Математические структуры и моделирование
2007, вып. 17, с. 112–120

УДК 003.26:519.713

ПРИМЕНЕНИЕ ТЕОРЕТИКО-АВТОМАТНЫХ МОДЕЛЕЙ В КРИПТОГРАФИИ

Н.Ф. Богаченко

В статье обсуждаются вопросы применения классических моделей теории автоматов в криптографии. Проведен обзор методов построения симметричных криптосистем и криптосистем с открытым ключом на основе последовательных автоматов.

В настоящее время «Теория автоматов» в классическом ее представлении (абстрактный и структурный синтез последовательных автоматов [1, 3]) – это достаточно изученная и методически обеспеченная дисциплина. Но вместе с тем представляет интерес применение такого математического объекта, как «автомат», в различных смежных отраслях. Остановимся на вопросах, связанных с возможностью использования автоматного подхода в криптографии.

Изложение материала не будет исчерпывающим. Целью работы является обсуждение нескольких криптосистем на идейном уровне и демонстрация принципиальной возможности использования в них последовательных автоматов (автоматов с памятью). Несмотря на то что предложенные криптосистемы являются скорее иллюстративными, они служат отправной точкой для дальнейших исследований.

1. Классические модели теории автоматов

Последовательным автоматом (или *автоматом с памятью*) называется система из пяти элементов

$$S = (A, X, Y, \delta, \lambda), \tag{1}$$

где $A = \{a_1, \ldots, a_p\}$ – множество внутренних состояний; $X = \{x_1, \ldots, x_n\}$ – множество входных сигналов (или входной алфавит); $Y = \{y_1, \ldots, y_m\}$ – множество выходных сигналов (или выходной алфавит); $\delta : D_\delta \subseteq A \times X \to A$ – функция переходов: $\lambda : D_\lambda \subseteq A \times X \to Y$ – функция выходов.

Если необходимо, в систему (1) добавляется шестой элемент $a_1 \in A$ – начальное состояние автомата.

В дальнейшем будем рассматривать только детерминированные последовательные автоматы с конечными множествами A, X и Y [3].

Омский государственный университет.
E-mail: nfbogachenko@mail.ru

Предполагается, что автомат функционирует в дискретные моменты времени $t = 1, 2, \ldots$ В каждый момент времени t автомат, находясь в определенном состоянии a_t множества A, воспринимает на входном канале сигнал $x_t \in X$, выдает на выходном канале сигнал $y_t \in Y : y_t = \lambda(a_t, x_t)$ и переходит в новое состояние $a_{t+1} \in A : a_{t+1} = \delta(a_t, x_t)$.

Другими словами, автомат S, находясь в начальном состоянии a_1, индуцирует *автоматное отображение* S_{a_1}, которое ставит в соответствие входному слову $\xi = x_1, \ldots, x_l$ выходное слово $\psi = y_1, \ldots, y_l$. При этом автомат проходит последовательность внутренних состояний $\alpha = a_1, \ldots, a_{l+1}$.

В зависимости от области определения функции выходов различают две функциональные модели последовательных автоматов:

1. *Модель Мили* (или *автомат Мили*):

$$\begin{cases} a_{t+1} = \delta(a_t, x_t); \\ y_t = \lambda(a_t, x_t). \end{cases} \tag{2}$$

2. *Модель Мура* (или *автомат Мура*):

$$\begin{cases} a_{t+1} = \delta(a_t, x_t); \\ y_t = \lambda(a_t). \end{cases} \tag{3}$$

Для задания автоматов удобно использовать табличное представление (см. табл. 1, 2, 3).

Таблица 1. Таблица переходов автомата Мили

	a_1	$\ldots$	a_p
x_1	$\delta(a_1, x_1)$	$\ldots$	$\delta(a_p, x_1)$
$\ldots$	$\ldots$	$\ldots$	$\ldots$
x_n	$\delta(a_1, x_n)$	$\ldots$	$\delta(a_p, x_n)$

Таблица 2. Таблица выходов автомата Мили

	a_1	$\ldots$	a_p
x_1	$\lambda(a_1, x_1)$	$\ldots$	$\lambda(a_p, x_1)$
$\ldots$	$\ldots$	$\ldots$	$\ldots$
x_n	$\lambda(a_1, x_n)$	$\ldots$	$\lambda(a_p, x_n)$

Таблица 3. Таблица переходов автомата Мура

	$\lambda(a_1)$	$\ldots$	$\lambda(a_p)$
	a_1	$\ldots$	a_p
x_1	$\delta(a_1, x_1)$	$\ldots$	$\delta(a_p, x_1)$
$\ldots$	$\ldots$	$\ldots$	$\ldots$
x_n	$\delta(a_1, x_n)$	$\ldots$	$\delta(a_p, x_n)$

На множестве состояний автомата определим отношение эквивалнтности: $a_i \sim a_j$, если $\forall \xi \in X^*$ (X^* – множество всевозможных слов над алфавитом X) выполняется равенство: $S_{a_i}(\xi) = S_{a_j}(\xi)$. Объединяя состояния автомата в классы эквивалентности, можно перейти к автомату, эквивалентному исходному, с минимально возможным числом состояний [3]. Далее будем рассматривать минимизированные последовательные автоматы.

2. Автоматная модель шифратора

Следуя [2], построим модель шифрующего автомата. Для этого необходимо дать несколько определений.

Шифром (или *криптосистемой*) называется система из пяти элементов

$$\Sigma = (K, \widetilde{X}^*, \widetilde{Y}^*, E, D), \tag{4}$$

где K – множество ключей; $\widetilde{X}^* \subseteq X^*$ – множество открытых текстов; $\widetilde{Y}^* \subseteq Y^*$ – множество закрытых текстов; $E_k : \xi \to \psi$ ($\xi \in \widetilde{X}^*$, $\psi \in \widetilde{Y}^*$) – правило зашифрования для $k \in K$; $D_k : \psi \to \xi$ – правило расшифрования для $k \in K$. При этом должны выполняться следующие свойства:

1. $\forall \xi \in \widetilde{X}^*$, $\forall k \in K$: $D_k(E_k(\xi)) = \xi$ (отсюда следует, что отображение E_k инъективно).
2. $\widetilde{Y}^* = \bigcup E_k(\xi)$, где объединение берется по всем $k \in K$ и $\xi \in \widetilde{X}^*$.

Очевидно, что функции переходов и выходов автомата, моделирующего работу шифра, должны зависить от ключа k. В качестве такого *шифрующего автомата* можно рассмотреть следующую систему:

$$S = (A \times K, X, Y, \delta, \lambda), \tag{5}$$

функциональная модель которой представляется автоматом Мили:

$$\begin{cases} \delta((a_t, k), x_t) = \delta_k(a_t, x_t) = (a_{t+1}, k); \\ \lambda((a_t, k), x_t) = \lambda_k(a_t, x_t) = y_t. \end{cases} \tag{6}$$

Перейдем теперь к вопросам расшифрования. С точки зрения автоматной модели для этого необходимо построить обратный автомат или, более строго, обратить автоматное отображение.

Автомат $S^{-1} = (\widetilde{A}, Y, X, \widetilde{\delta}, \widetilde{\lambda})$ называется *обратным* (*слева*) к автомату $S = (A, X, Y, \delta, \lambda)$, если $\forall a \in A$ $\exists \widetilde{a} \in \widetilde{A}$ такое, что $\forall \xi \in X^*$ выполняется равенство $S_{\widetilde{a}}^{-1}(S_a(\xi)) = \xi$. Далее, пока не оговорено противное, будем рассматривать левую обратимость.

Очевидно, что для существования обратного автомата необходимо и достаточно потребовать инъективность автоматного отображения S_a для любого начального состояния a. Инъективность автоматного отображения достигается требованием инъективности функции выходов λ при фиксированном состоянии a, то есть частичной функции $\lambda_a : X \to Y$. Это, в свою очередь, означает, что $|\lambda_a^{-1}(y)| \leq 1$ [2].

Итак, для однозначности расшифрования на шифрующий автомат (5), (6) необходимо наложить условие инъективности частичной функции выходов λ_a.

Инъективность λ_a приводит к следующему алгоритму построения обратного автомата [2]:

1. Если $|\lambda_a^{-1}(y)| = 1$, то $\widetilde{\lambda}(a, y) = \lambda_a^{-1}(y)$ и $\widetilde{\delta}(a, y) = \delta(a, \lambda_a^{-1}(y))$.
2. В противном случае, когда $|\lambda_a^{-1}(y)| = 0$, $\widetilde{\lambda}(a, y)$ – произвольный $x \in X$, $\widetilde{\delta}(a, y)$ – произвольное $a \in A$.

Если мощности алфавитов X и Y совпадают, то обратный автомат строится однозначно. В общем случае должно выполняться $|X| \leqslant |Y|$. Если неравенство строгое, то обратный автомат является частичным.

Заметим, что для обратимости автомата необходимо и достаточно, чтобы в его табличном представлении в каждом столбце таблицы выходов все выходные сигналы были различны.

Для иллюстрации предложенных алгоритмов и моделей рассмотрим автомат Мили S, заданный таблицами 4, 5. Если произвольное значение сигнала или состояния обозначить прочерком, то автомат S^{-1} представляется таблицами 6, 7.

Таблица 4. Переходы автомата S

	a_1	a_2	a_3
x_1	a_3	a_1	a_1
x_2	a_1	a_3	a_2

Таблица 5. Выходы автомата S

	a_1	a_2	a_3
x_1	y_3	y_1	y_2
x_2	y_1	y_3	y_1

Таблица 6. Переходы автомата S^{-1}

	a_1	a_2	a_3
y_1	a_1	a_1	a_2
y_2	$-$	$-$	a_1
y_3	a_3	a_3	$-$

Таблица 7. Выходы автомата S^{-1}

	a_1	a_2	a_3
y_1	x_2	x_1	x_2
y_2	$-$	$-$	x_1
y_3	x_1	x_2	$-$

В простейшем случае ключом можно считать сам автомат. Пусть a_3 – начальное состояние. В качестве открытого текста рассмотрим последовательность $\xi = x_1x_1x_2x_1x_1$. Тогда закрытый текст $\psi = S_{a_3}(\xi)$ очевидным образом определяется по таблицам 4, 5:

$$\begin{array}{lccccc|c} \xi = & x_1 & x_1 & x_2 & x_1 & x_1 & \\ \alpha = & a_3 & a_1 & a_3 & a_2 & a_1 & a_3 \\ \psi = & y_2 & y_3 & y_1 & y_1 & y_3 & \end{array}$$

Процесс расшифрования проводится по таблицам 6, 7:

$$\begin{array}{lccccc|c} \psi = & y_2 & y_3 & y_1 & y_1 & y_3 & \\ \alpha = & a_3 & a_1 & a_3 & a_2 & a_1 & a_3 \\ \xi = & x_1 & x_1 & x_2 & x_1 & x_1 & \end{array}$$

Далее сформулируем ряд замечаний.

Одним из преимуществ представленного «автоматного» способа шифрования является то, что одинаковые фрагменты входной последовательности в общем случае шифруются различными блоками. Это видно даже на нашем примере: подпоследовательность x_1x_1 встречается во входном слове дважды, и ей соответствуют подпоследовательности y_2y_3 и y_1y_3 закрытого текста.

Если потребовать, чтобы частичная функция λ_x также была инъективной, то таблица выходов автомата (без строки состояний и столбца входных сигналов) будет представлять собой латинский прямоугольник – элементы в линиях (как в столбцах, так и в строках) не повторяются. Существующие оценки числа латинских прямоугольников [9] могут служить основой для анализа криптостойкости алгоритмов «автоматного» шифрования.

Следует отметить, что итогом теоретико-автоматной модели шифра является принципиальная возможность построения криптосистемы в виде двух последовательных автоматов (шифратора и дешифратора), блоки памяти которых совпадают [2].

3. Криптосистемы с открытым ключом, основанные на последовательных автоматах

Идея использования автоматой модели в криптографии с открытым ключом не нова. Так, в Китае с 80-х годов прошлого века ведутся работы в области создания асимметричных криптосистем, основанных на конечных автоматах [11]. Здесь следует остановиться на терминологии. *Конечный автомат* - это модель распознавателя, являющаяся эквивалентом автоматной грамматики относительно порождаемых языков [4,10]. С точи зрения классической теории автоматов, в частности последовательных абстрактных моделей, конечный автомат - это автомат Мура, в общем случае частичный, выходные сигналы которого заданы в алфавите $\{0,1\}$ [4].

В работе [8] предложена криптосистема с открытым ключом, основным элементом которой является последовательный автомат. Как уже отмечалось, произвольный последовательный автомат S реализует автоматное отображение $S(\xi) = \psi$. При этом и входное, и выходное слова имеют одинаковую длину: $|\xi| = |\psi|$.

Автомат S^{-1} с входным алфавитом Y и выходным алфавитом X, переводящий выходное слово ψ обратно во входное ξ, – это, как и ранее, обратный автомат, но теперь нам понадобится обратимость справа.

Автомат $S^{-1} = (\widetilde{A}, Y, X, \widetilde{\delta}, \widetilde{\lambda})$ называется *обратным* (*справа*) к автомату $S = (A, X, Y, \delta, \lambda)$, если $\forall a \in A\ \exists \widetilde{a} \in \widetilde{A}$ такое, что $\forall \psi \in Y^*$ выполняется равенство $S_a(S_{\widetilde{a}}^{-1}(\psi)) = \psi$. Отметим, что для доказательства правой обратимости необходимо и достаточно потребовать, чтобы автоматное отображение S_a было сюръективным для любого начального состояния a, или предъявить эквивалентное требование – сюръективность частичной функции выходов λ_a [2]. В табличном представлении автомата для правой обратимости необходимо и достаточно, чтобы в каждом столбце таблицы выходов каждый выходной сигнал встречался как минимум один раз.

Алгоритм построения правого обратного автомата достаточно прост [2]: $\widetilde{\lambda}(a, y) = x$ и $\widetilde{\delta}(a, y) = \delta(a, x)$, где x – произвольный элемент из множества $\lambda_a^{-1}(y)$, которое в силу сюръективности не пусто.

Определим теперь *инверсию с задержкой* z, где z – некоторое натуральное число. Пусть $\psi = S(\xi)$. Инверсия с задержкой S_z^{-1}, получая на входе слово $\psi\mu$,

где μ – произвольное слово длины z, порождает на выходе слово $\nu\xi$, где ν – также слово длины z.

Представленная в [8] криптосистема с открытым ключом строится следующим образом. В качестве ключа зашифрования открываются z и S_z^{-1}, а S хранится как секретный ключ расшифрования. Зашифрование исходного текста ψ происходит с помощью выбора произвольного слова μ длины z и применения автомата S_z^{-1} к слову $\psi\mu$. При расшифровании криптотекста легальный получатель игнорирует первые z букв и применяет автомат S к оставшемуся слову. Отметим, что в данном методе используется правая инверсия с задержкой.

Обсуждаемый в [6] асимметричный алгоритм шифрования также основан на паре автоматов S и S_z^{-1}. На шифрующий автомат S накладывается требование отсутствия инъективности частичной функции выходов λ_a (тогда автомат S необратим слева). При дополнительных условиях возможно построение левой инверсии с задержкой S_z^{-1}, что в общем случае является трудной задачей. Знание особенностей структуры автомата S дает легкий алгоритм получения S_z^{-1} (секретный ключ).

4. Шифрование с помощью «раскрашивания» на автоматах Мура

Достаточно много криптосистем используют подход, который может быть сформулирован как *шифрование с помощью «раскрашивания»* [8]. «Краска» связывается с каждой буквой исходного текста. Для простоты будем полагать, что исходные тексты являются двоичными последовательностями. Тогда нам необходимы только две краски - белая (бит «0») и черная (бит «1»).

Ключ зашифрования должен давать метод, который порождает произвольно много элементов, раскрашенных белой краской, и произвольно много элементов, раскрашенных черной краской. Примером таких элементов могут являться слова некоторого фиксированного алфавита. Биты шифруются как слова, раскрашенные соответствующим образом. Для различных появлений бита «0» (соответственно «1») должны выбираться различные слова, раскрашенные белой (соответственно черной) краской.

Для генерации «белых» и «черных» слов представляется возможным использовать автоматную модель. Пусть $S = (A, X, Y, \delta, \lambda, a_{\text{н.с.}})$ – автомат Мура, в котором выходной алфавит является двоичным: $Y = \{0, 1\}$ и $a_{\text{н.с.}}$ – начальное состояние.

Процесс зашифрования открытого двоичного текста заключается в следующем. Текущий бит i ($i \in \{0, 1\}$) шифруется на автомате S как произвольное входное слово ξ алфавита X с тем ограничением, что последний символ выходного слова $\psi = S(\xi)$ равен «i». Следует отметить, что в автомате Мура, в отличие от модели Мили, выходное слово строится со сдвигом на один такт, то есть не учитывается выходной сигнал, приписанный начальному состоянию.

При расшифровании текущее слово ξ криптотекста подается на вход автомата S: последний символ выходного слова $\psi = S(\xi)$ и будет являться искомой

«краской». Однозначность расшифрования очевидна. Естественно, что сам автомат S необходимо держать в секрете.

Таблица 8. Переходы автомата Мура S

	1	0	1	0	1
	a_1	a_2	a_3	a_4	a_5
a	a_4	a_4	a_1	a_2	a_2
b	a_1	a_1	a_5	a_3	a_1
c	a_2	a_1	a_3	a_5	a_4

Рассмотрим предложенный способ шифрования на примере. Пусть автомат Мура S задан таблицей 8 и в качестве начального состояния выбрано состояние a_1. Зашифруем последовательность «00101». Одним из вариантов закрытого текста является последовательность слов: «$a\ bc\ b\ c\ ab$». Расшифрование продемонстрируем на примере слова «ab»:

$$\begin{array}{lc|c|c} \xi = & a & b & \\ \alpha = & a_1 & a_4 & a_3 \\ \psi = & & 0 & 1 \end{array}$$

Так как последний символ выходного слова равен «1», то и соответствующий бит открытого текста – это единица.

Очевидно, что все криптосистемы, основанные на шифровании с помощью раскрашивания, имеют тенденцию к недопустимо большому росту длины криптотекста, так как чем больше различных слов, пригодных для шифрования одного бита (а следовательно, чем больше длина каждого слова), тем более стойким является криптоалгоритм.

Этот недостаток не столь существенен, если подобные криптосистемы использовать не для шифрования сообщений, а для шифрования ключей некоторого симметричного алгоритма, который в дальнейшем и будет применяться для передачи сообщений [11].

По этой же причине шифрование с помощью раскрашивания наиболее широко используется в асимметричных криптосистемах. Открытый ключ зашифрования – это метод выбора слов, сопоставимых каждому биту. В идеальной ситуации задача определения «краски» для заданного слова является трудноразрешаемой, в то время как знание секретного ключа позволяет легко решать задачу расшифрования [8].

Рассмотренный алгоритм шифрования также может быть преобразован в криптосистему с открытым ключом. Как уже отмечалось, автомат Мура с двоичным выходным алфавитом является конечным автоматом.

Пусть дан автомат Мура $S^0 = (A, X, Y = \{0, 1\}, \delta, \lambda^0, a_{\text{н.с.}})$. Это конечный автомат, определяющий автоматный язык $L(S^0)$ – множество таких слов входного алфавита X, что последний символ соответствующих выходных слов равен «1». Построим автомат $S^1 = (A, X, Y, \delta, \lambda^1, a_{\text{н.с.}})$ по правилу: $\lambda^1(a, x) = \neg\lambda^0(a, x)$. Этот автомат задает автоматный язык $L(S^1)$.

Согласно [8], выберем две грамматики Γ^0 и Γ^1 с одним и тем же терминальным алфавитом X так, чтобы языки $L(\Gamma^0)$ и $L(\Gamma^1)$, порождаемые этими грамматиками, удовлетворяли условию: $L(\Gamma^i) \cap L(S^i) \neq \varnothing$ $(i \in \{0,1\})$.

Построим две новые грамматики $\widetilde{\Gamma}^0$ и $\widetilde{\Gamma}^1$ такие, что $L(\widetilde{\Gamma}^i) = L(\Gamma^i) \cap L(S^i)$ $(i \in \{0,1\})$. Построение пересечения языков – стандартное. Отметим лишь, что в общем случае языки не замкнуты относительно операции пересечения, то есть тип результирующего языка в иерархии Хомского может быть произвольным [10].

Пара $(\widetilde{\Gamma}^0, \widetilde{\Gamma}^1)$ открывается в качестве ключа зашифрования – бит i шифруется как произвольное слово в $L(\widetilde{\Gamma}^i)$ $(i \in \{0,1\})$. Автоматы S^0 и S^1 хранятся в качестве секретного ключа.

Перехватчик для расшифрования должен определить принадлежность каждого слова криптотекста языку $L(\widetilde{\Gamma}^i)$ $(i \in \{0,1\})$, то есть решить задачу распознавания, являющуюся трудной для произвольного контекстного языка, а для языка типа 0 – в общем случае неразрешимой.

Легальный получатель решает легкую задачу принадлежности слова криптотекста одному из автоматных языков $L(S^0)$ или $L(S^1)$.

В силу построения автоматов S^0 и S^1, языки $L(S^0)$ и $L(S^1)$ не пересекаются (они являются дополнениями друг друга) [8, 10]. Тем самым расшифрование всегда будет однозначным.

Рассмотренные в статье криптосистемы не являются единственно возможной сферой применения «автоматного» подхода в криптографии.

Автоматные модели используются в формальном анализе криптографических протоколов. Существующие алгоритмы основаны на следующем подходе. Если предположить, что злоумышленник может удалять сообщения из канала связи и помещать в канал связи созданные им сообщения, то любое сообщение, посланное легальным пользователем, можно рассматривать как сообщение, отправленное злоумышленнику, и любое сообщение, принятое легальным пользователем, как сообщение, полученное от злоумышленника. Тогда систему можно интерпретировать как автомат, используемый злоумышленником для генерации слов [5].

Представляет интерес идея построения однонаправленных хэш-функций на основе последовательных автоматов. Основное свойство однонаправленной функции – простота вычисления и сложность (практическая невозможность) инвертирования. Тогда в качестве такой функции можно рассмотреть необратимое (или трудно обратимое) автоматное отображение. Требования на хэш-функцию накладывают ограничения и на «однонаправленный» автомат S: трудно подобрать два разлиных входных слова ξ_1 и ξ_2 таких, что $S(\xi_1) = S(\xi_2)$, а изменение одного символа во входном слове ξ ведет к изменению в среднем половины символов выходного слова $\psi = S(\xi)$.

В работах [6, 7] рассматриваются автоматные однонаправленные отображения, для которых сложность построения обратного автомата приближается к экспоненциальной зависимости.

В заключение отметим, что для представленных в обзоре криптосистем не обсуждались такие вопрсы, как оценка сложности, область применения, сопоставимость с другими шифрами. Важно было показать принципиальную возможность приложения классических теоретико-автоматных моделей к задачам криптографии.

Литература

1. Баранов С.И. Синтез микропрограммных автоматов. Ленинград: Энергия, 1974. 216 с.
2. Белов Е.Б., Зубов А.Ю., Погорелов Б.А., Проскурин Г.В., Черемушкин А.В., Шанкин Г.П., Шурупов А.Н. Криптографические методы защиты информации. Учебно-методическое пособие для курсов повышения квалификации преподавателей вузов учебно-методического объединения по образованию в области информационной безопасности. Часть 2. Москва. 2003.
3. Богаченко Н.Ф., Файзуллин Р.Т. Синтез дискретных автоматов. Учебно-методическое пособие. Омск: Издательство Наследие. Диалог-Сибирь, 2006. 150 с.
4. Богаченко Н.Ф., Файзуллин Р.Т. Автоматы, грамматики, алгоритмы. Учебно-методическое пособие. Омск: Издательство Наследие. Диалог-Сибирь, 2006. 140 с.
5. Давыдов А.Н. Формальный анализ криптографических протоколов: методы, основанные на моделях конечных автоматов // Труды научно-технической конференции «Безопасность информационных технологий». Пенза. 2005. – http://beda.stup.ac.ru/rv-conf/v06/009/index.html.
6. Копыленко В.М. Криптографическая конечно-автоматная модель. – http://zhurnal.lib.ru/k/kopylenko_w_m/afr2.shtml.
7. Копыленко В.М. Однонаправленная функция с «секретом» (trap-door function) на базе КАМСИ (Конечно-автоматная модель, сохраняющая информацию). – http://zhurnal.lib.ru/k/kopylenko_w_m/af_7.shtml.
8. Саломаа А. Криптография с открытым ключом. М.: Мир, 1995.
9. Холл М. Комбинаторика. М.: Мир, 1970.
10. Шамашов М.А. Теория формальных языков. Грамматики и автоматы. Учебное пособие. Самара: Университет Наяновой, 1996.
11. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. М.: ТРИУМФ, 2003.