



Проект программного комплекса генерации машин Тьюринга, решающих NP-трудные задачи

В.И. Мартьянов¹

Иркутский национальный исследовательский технический университет, г. Иркутск, Россия

Байкальский государственный университет, г. Иркутск, Россия

Иркутский государственный университет, г. Иркутск, Россия

Аннотация. Целью исследования является обобщение промежуточных результатов программной реализации концепции генерации машин Тьюринга, решающих NP-сложные задачи. Обсуждаются планы по разработке программного пакета для генерации машин Тьюринга, который может стать открытой платформой для обучения студентов теории алгоритмов и информационным технологиям; проведения всеми желающими экспериментов по определению неполиномиальной сложности решения определенной серии задач, сгенерированными машинами Тьюринга; участия всех желающих в генерации машин Тьюринга и решении NP-трудных задач на личных компьютерах; передачи результатов просчетов на платформу, являющуюся подобием генерации криптовалют, но осуществляющую поиск нетленных математических объектов, а не каких-то искусственно созданных на короткий период времени блоков чисел. Отмечается, что идея использования методов удовлетворения ограничений для генерации машин Тьюринга, решающих NP-трудные задачи, записанные на ленте, является расширением возможностей программирования в ограничениях и, возможно, будет успешной для решения вопроса о равенстве полиномиальной и NP-сложности ($P = ? NP$), что является одной из семи пока не решенных проблем третьего тысячелетия математики.

Ключевые слова: NP-трудные задачи, удовлетворение ограничений, программирование в ограничениях, искусственный интеллект, машины Тьюринга

Для цитирования: Мартьянов В.И. Проект программного комплекса генерации машин Тьюринга, решающих NP-трудные задачи // Известия вузов. Инвестиции. Строительство. Недвижимость. 2023. Т. 13. № 2. С. 285–297. <https://elibrary.ru/uaewdt>. <https://doi.org/10.21285/2227-2917-2023-2-285-297>.

Original article

Draft program complex for generating Turing machines solving NP-hard problems

Vladimir I. Martyanov

Irkutsk National Research Technical University, Irkutsk, Russia

Baikal State University, Irkutsk, Russia

Irkutsk State University, Irkutsk, Russia

Abstract. This paper provides an overview of the progress made in the software implementation of the concept for generating Turing machines that solve NP-hard problems. Plans are discussed for developing a software package for generating Turing machines, with the potential to serve as an open-source educational platform for learning algorithm theory and information technologies. The proposed program complex also encompasses the opportunity for interested individuals to conduct experiments aimed at

determining the NP-hardness of specific problem series generated by Turing machines; participating in the generation of Turing machines and solving NP-hard problems on personal computers; transferring the calculation results to a platform, similar to the generator of cryptocurrencies, which focuses on searching for timeless mathematical objects, instead of creating artificially generated blocks of numbers for a short period. It is noted that the idea of using constraint satisfaction methods to generate Turing machines for solving tape NP-hard problems extends the boundaries of constraint programming. Furthermore, it holds the potential for contributing to the resolution of the long-standing question regarding the equality of polynomial and NP-hardness ($P = ? NP$) – one of the seven unsolved problems of the third millennium in mathematics to this day.

Keywords: NP-hard problems, satisfying constraints, programming in constraints, artificial intelligence, Turing machines

For citation: Martyanov V.I. Draft program complex for generating Turing machines solving NP-hard problems. *Izvestiya vuzov. Investitsii. Stroitel'stvo. Nedvizhimost' = Proceedings of Universities. Investment. Construction. Real estate*. 2023;13(2):285-297. (In Russ.). <https://elibrary.ru/uaewdt>. <https://doi.org/10.21285/2227-2917-2023-2-285-297>.

ВВЕДЕНИЕ

Программный комплекс генерации машин Тьюринга (ПКГМТ), решающих NP-сложные задачи, основан на методах, предложенных в работах [1–5], функционирует до некоторой степени подобно генераторам криптовалют [6–8], но формирует машины Тьюринга (МТ), решающие математические задачи. С точки зрения программной реализации ПКГМТ является результатом применения расширенной концепции программирования в ограничениях [9–10].

Автор данной работы считает, что, может быть, это станет путем отрицательного решения проблемы $P = ? NP$ [11–13].

1. Структура программного комплекса ПКГМТ генерации МТ.

1.1. Начальные данные пишутся на ленте МТ [14, 15] и как математический объект представляют формулы исчисления высказываний. Напомним, что именно алгоритмическая сложность проверки выполнимости высказываний $\alpha(x_1, x_2, \dots, x_n)$ объявлена NP-трудной (полной) задачей. В табл. 1 в 1-ой клетке прописан начальный элемент ленты 0000 и эту клетку считывает головка МТ готовая к исполнению прописанной команды МТ.

1.2. Алфавит МТ $ho_alphabet.slw = \{0; 1; b; e; \#; 5; 6\}$, который в ПКГМТ представлен следующим образом (табл. 2).

1.3. Совокупность состояний МТ $ho_state.slw = \{q; q0, q1, q2, q3\}$,

где q – начальное состояние; $q0$ – стоп (конец работы); $q1, q2, q3$ – без свойств, в ПКГМТ представлена в табл. 3.

1.4. Совокупность движений головки МТ $ho_displace.slw = \{right; left, stop\}$ в ПКГМТ представлена в табл. 4.

1.5. Программа МТ $ho_DMT.slw$ в ПКГМТ представлено в табл. 5.

1.6. Редактор программы МТ $ho_DMT.slw$ в ПКГМТ представлен в табл. 6.

Редактирование в данном случае может осуществляться для 2-ой команды вызовом одного из трех словарей. Покажем редактирование состояния в табл. 7.

1.7. Проектирование расписания занятий является одним из классов NP-трудных задач.

Основные множества: Дисц = $\{d1, d2, \dots, ds\}$ – дисциплины, Груп = $\{g1, g2, \dots, gu\}$ – учебные группы, Преп = $\{l1, l2, \dots, ls\}$ – преподаватели, Мн-во занятий: Зан $\subseteq$ Дисц $\otimes$ Груп $\otimes$ Преп. Врем = $\{t1, t2, \dots, tu\}$ – пары в двухнедельном цикле, Ауд = $\{a1, a2, \dots, aw\}$ – аудитории.

Необходимо построить Распис : Зан $\rightarrow$ Врем $\otimes$ Ауд, т.е. отображение, которое для (дисциплина, группа, преподаватель) определяет время проведения занятия и аудиторию.

Программа генерации МТ в ПКГМТ реализована во многом на технических решениях:

- проекта системы управления региональной сетью автомобильных дорог (СУРАД) Иркутской области [16];

- проекта построения графа автомобильных дорог для системы взимания платы с большегрузного транспорта [17];

- приложений логико-эвристических методов для расчета комбинаторных задач высокой сложности [3];

- программы сетевого планирования содержания сети автомобильных дорог Иркутской области [18];

- теоретико-множественные модели данных в задаче расчета вторичных структур РНК [19];

- программы планирования работ по нормативному содержанию сети автомобильных дорог [20];
- программы обработки видеорядов, снятых дорожными лабораториями на автомобильных дорогах [21];
- программы расчета (генерации) расписания [22], используемой с 2006 г. в ИРНТУ. Аналоги организации данных и требования

к расчету можно найти в итоговых результатах расчета расписания (табл. 8).

Карточка (табл. 8) показывает, что:

- занятия ставятся в первую смену или на стыковочную 4-ю пару;
- лекций (красный цвет) ставится не более 2-х в день;
- отсутствуют единственные занятия и «окна».

Таблица 1. Начальный фрагмент ленты MT в ПКГМТ

Table 1. The initial fragment of the MT tape in the software package for generating Turing machines

Выберите объект из списка:

0000 (b begin,q1)->(b begin,q2,right)

☐ 0000 (b begin,q1)->(b begin,q2,right)

☐ 0001 0

☐ 0002 0

☐ 0003 0

☐ 0004 0

☐ 0005 0

☐ 0006 e end

✓ Объект выбран Убрать отметки объектов ✗ Отменить выбор

1. Общее количество объектов : 7

Таблица 2. Алфавит MT в ПКГМТ

Table 2. The MT alphabet in the software package for generating Turing machines

Выберите объект из списка:

1

☐ 0

☒ 1

☐ b begin

☐ e end

☐ # devide

☐ 5

☐ 6

☐ 7

✓ Объект выбран Убрать отметки объектов ✗ Отменить выбор

1. Общее количество объектов : 8

Таблица 3. Совокупность состояний MT в ПКГМТ

Table 3. The totality of MT states in the software package for generating Turing machines

Выберите объект из списка:

q0 stop

☐ q begin

☒ q0 stop

☐ q1

☐ q2

☐ q3

☐ q4

☐ q5

☐ n6

✓ Объект выбран Убрать отметки объектов ✗ Отменить выбор

1. Общее количество объектов : 8

Таблица 4. Совокупность движений головки МТ в ПКГМТ

Table 4. The totality of the movements of the MT head in the software package for generating Turing machines

Выберите объект из списка:

left

☐ right

☒ left

☐ stop

✓ Объект выбран Убрать отметки объектов Отменить выбор

1. Общее количество объектов : 3

Таблица 5. Программа МТ в ПКГМТ

Table 5. Program MT in the software package for generating Turing machines

Выберите объект из списка:

(0,q0 stop)->(?,?,?)

☐ (0,q1)->(1,q begin,left)

☐ (0,q2)->(1,q begin,right)

☐ (0,q3)->(?,?,?)

☐ (1,q begin)->(5,q begin,left)

☐ (1,q0 stop)->(?,?,?)

☐ (1,q1)->(1,q1,left)

☒ (1,q2)->(e_end,q1,left)

☐ (1,q3)->(??,?)

✓ Объект выбран Убрать отметки объектов Отменить выбор

1. Общее количество объектов : 35

Таблица 6. Редактор программы МТ в ПКГМТ

Table 6. The editor of the MT program in the software package for generating Turing machines

Список объектов

Настройка экрана Операции проектирования Выход

ДМТ: Поиск покрытия	Команда 1-ая	Команда 2-ая	Команда 3-ая	Команда 4-ая	Команда 5-ая	Команда 6-ая	Команда 7-ая
Аргументы команды	0	0	0	0	0	1	1
Значение Символ	q begin	q0 stop	q1	q2	q3	q begin	q0 stop
Значение Состояние	0	???	1	1	???	5	???
Значение Сдвиг	right	???	left	right	???	left	???

ДМТ: Поиск покрытия	Команда 8-ая	Команда 9-ая	Команда 10-ая	Команда 11-ая	Команда 12-ая	Команда 13-ая	Команда 14-ая
Аргументы команды	q2	q3	# device	q begin	q0 stop	q1	q2
Значение Символ	e_end	???	0	???	???	???	???
Значение Состояние	q1	???	q begin	???	???	???	???
Значение Сдвиг	right	???	left	???	???	???	???

☐ Символ (аргумент) : 0

☐ Состояние (аргумент) : q0 stop

☐ Символ (значение): ???

☐ Состояние (значение): ???

☐ Двиг головки (значение): ???

Таблица 7. Редактор состояния во второй команде программы МТ в ПКГМТ
Table 7. Status editor in the second command of the program MT in the software package for generating Turing machines

С:\prg\яя СПИСОК ОБЪЕКТОВ

Настройка экрана Операции проектирования Выход

ДМТ: Поиск покрытия	Команда 1 - ая	Команда 2 - ая	Команда 3 - ая	Команда 4 - ая	Команда 5 - ая	Команда 6 - ая	Команда 7 - ая
Аргументы команды	0	0	0	0	0	1	1
Значение Символ	q begin	q0 stop	q1				
Значение Состояние	q begin	???	q begin				
Значение Сдвиг	right	???	left				

Выберите объект из списка:

q0 stop

☐ q begin
☒ q0 stop
☐ q1
☐ q2
☐ q3
☐ q4
☐ q5
☐ n6

☒ Объект выбран

1. общее количество объектов : 8

ДМТ: Поиск покрытия	Команда 19 - ая	Команда 20 - ая	Команда 21 - ая	Команда 22 - ая	Команда 23 - ая	Команда 24 - ая	Команда 25 - ая
Аргументы команды	e end	e end	# devide	# devide	# devide	# devide	# devide
Значение Символ	q2	q3	q begin	q0 stop	q1	q2	q3
Значение Состояние	q1	???	q begin	???	???	???	???
Значение Сдвиг	right	???	left	???	???	???	???

☐ Символ (аргумент) : 0
☐ Состояние (аргумент) : q0 stop
☐ Символ (значение) : ???
☒ Состояние (значение) : ???
☐ Двиг головки (значение) : ???

Таблица 8. Карточка подгруппы АСУб-22-1-2
Table 8. Subgroup card ASUb-22-1-2

D:\2022\22_23\2333-07\2333-07П ОБЪЕКТ: АСУб-22-1(И,О)

Настройка экрана Операции проектирования Выход

29.08-03.09	ПОНЕДЕЛЬНИК	ВТОРНИК	СРЕДА	ЧЕТВЕРГ	ПЯТНИЦА	СУББОТА
1 - с 8:15 по 9:45	Информатика; Иностранный язык	Физика; АСУб-22-1(И,О); АСУб-22-1-2(И,О)	Физика; АСУб-22-1(И,О)	Дискретная математика; АСУб-22-1(И,О)		
2 - с 10:00 по 11:30	Физика; АСУб-22(И,О); И(АСУб-22(И,О); И(АСУб-22-1(И,О); И(АСУб-22-1(И,О)	Математика; АСУб-22-1(И,О); АСУб-22-1(И,О)	Элективные курсы	Математика; АСУб-22(И,О); АСУб-22-1(И,О)	Дискретная математика; АСУб-22(И,О); АСУб-22-1(И,О)	Введение в программирование; АСУб-22-1(И,О)
3 - с 11:45 по 13:15	Иностранный язык	Программирование; АСУб-22-1(И,О); АСУб-22-1(И,О)	Физическая культура; АСУб-22(И,О); И(АСУб-22-1(И,О); И(АСУб-22-1(И,О)	Программирование; АСУб-22-1(И,О); АСУб-22-1(И,О)	Введение в программирование; АСУб-22-1(И,О)	
4 - с 13:45 по 15:15		Физика; АСУб-22-1-2(И,О)	Основы делового общения; АСУб-22-1(И,О)	Введение в программирование; АСУб-22-1(И,О)		
5 - с 15:30 по 17:00						
6 - с 17:10 по 18:40						

05.09-10.09	ПОНЕДЕЛЬНИК	ВТОРНИК	СРЕДА	ЧЕТВЕРГ	ПЯТНИЦА	СУББОТА
1 - с 8:15 по 9:45	Информатика; АСУб-22-1(И,О)	Программирование; АСУб-22(И,О); АСУб-22-1-2(И,О)	Иностранный язык; АСУб-22-1(И,О)			
2 - с 10:00 по 11:30	Физика; АСУб-22(И,О); И(АСУб-22(И,О); И(АСУб-22-1(И,О); И(АСУб-22-1(И,О)	Математика; АСУб-22(И,О); АСУб-22-1(И,О)	Элективные курсы	Математика; АСУб-22(И,О); АСУб-22-1(И,О)	Дискретная математика; АСУб-22(И,О); АСУб-22-1(И,О)	Введение в программирование; АСУб-22-1(И,О)
3 - с 11:45 по 13:15	Иностранный язык	Программирование; АСУб-22-1(И,О); АСУб-22-1(И,О)	Физическая культура; АСУб-22(И,О); И(АСУб-22-1(И,О); И(АСУб-22-1(И,О)	Дискретная математика; АСУб-22(И,О); АСУб-22-1(И,О)	Программирование; АСУб-22-1(И,О); АСУб-22-1(И,О)	Введение в программирование; АСУб-22-1(И,О)
4 - с 13:45 по 15:15			Основы делового общения; АСУб-22-1(И,О)		Программирование; АСУб-22-1(И,О)	
5 - с 15:30 по 17:00						
6 - с 17:10 по 18:40						

☐ Предмет : Физика;
☐ Преподаватель(и) : Баранов Е.О.;
☐ Контингент(и) : АСУб-22-1-2(И,О);
☐ Аудитория(и) : Б-206;

☐ день недели : ВТОРНИК
☐ пара : 4 - с 13:45 до 15:15
☐ Четность недели : НЕЧЕТНАЯ НЕДЕЛЯ
☐ Номер занятия : 6509;

☐ Вид работы : 2. лабораторные работы
☐ кафедра : Физики
☐ дата проведения : 30.08
☐ период проведения : 29.08 до 25.12

КОЛ-ВО УЧАЩИХСЯ: 12
 ВМЕСТИМОСТЬ : 30

Таблица 9. Карточка преподавателя Е.С. Друговой
Table 9. Teacher's card by E.S. Drugova

D:\2022\22_23\233-070 ОБЪЕКТ: Другова Е.С.

Настройка экрана Операции проектирования Выход

29.08-03.09	ПОНЕДЕЛЬНИК	ВТОРНИК	СРЕДА	ЧЕТВЕРГ	ПЯТНИЦА	СУББОТА
1-с 8:15 по 9:45	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ
2-с 10:00 по 11:30		Управление общ ЖР6-21(И,О);				ОСВОБОДИТЬ
3-с 11:45 по 13:15		Жанры PR-текст ЖР6-20-2(И,О);	Жанры PR-текст ЖР6-20(И,О);			ОСВОБОДИТЬ
4-с 13:45 по 15:15		Пользовательск ЖР6-19(И,О);	Управление общ ЖР6-21-1(И,О);	Пользовательск ЖР6-19-1-1(И,О);		ОСВОБОДИТЬ
5-с 15:30 по 17:00		Жанры PR-текст ЖР6-20-1(И,О);		Пользовательск ЖР6-19-1-2(И,О);		ОСВОБОДИТЬ
6-с 17:10 по 18:40	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ

05.09-10.09	ПОНЕДЕЛЬНИК	ВТОРНИК	СРЕДА	ЧЕТВЕРГ	ПЯТНИЦА	СУББОТА
1-с 8:15 по 9:45	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ
2-с 10:00 по 11:30		Управление общ ЖР6-21(И,О);				ОСВОБОДИТЬ
3-с 11:45 по 13:15		Реклама в меди ЖР6-21(И,О);	Жанры PR-текст ЖР6-20(И,О);	Реклама в меди ЖР6-21-1(И,О);		ОСВОБОДИТЬ
4-с 13:45 по 15:15			Управление общ ЖР6-21-1(И,О);	Пользовательск ЖР6-19-1-1(И,О);		ОСВОБОДИТЬ
5-с 15:30 по 17:00				Пользовательск ЖР6-19-1-2(И,О);		ОСВОБОДИТЬ
6-с 17:10 по 18:40	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ	ОСВОБОДИТЬ

☐ Предмет :
☐ Преподаватель(и) :
☐ Контингент(ы) :
☐ Аудитория(и) :

КОЛ-ВО УЧАЩИХСЯ: 16

☐ День недели : ПЯТНИЦА
☐ Пара : 5 – с 15:30 до 17:00
☐ Четность недели : ЧЕТНАЯ НЕДЕЛЯ
☐ Номер занятия :

☐ Вид работы :
☐ Кафедра :
☐ Дата проведения :
☐ Период проведения : ДО

Карточка (табл. 9) показывает, что:

- лекций (красный цвет) ставится не более 2-х в день;
- отсутствуют единственные занятия и «окна»;
- учитываются пожелания преподавателя на освобождение времени от занятий.

Кроме того, размерность задачи проектирования расписания для ИРНТУ имеет очень большую размерность, а именно $(400 \times 72) \times 10000$ (практически сравнима с размерностью задачи генерации МТ при количестве элементов алфавита и совокупности состояний не более 10). Следует из числа занятий примерно 10 000, числа аудиторий бо-

лее 400 и количества пар 72 в 2-недельном цикле. Расчет расписания занимает чуть более недели на настольном компьютере, причем качество расписания недостижимо для ручного проектирования.

2. Результаты просчетов программного комплекса ПКГМТ генерации МТ.

2.1. Пока не ставится цель провести предельно большие просчеты, так как еще не реализованы многие компоненты ПКГМТ и многие стратегии оптимизации переборов (например, применение параллельных вычислений). Самый глубокий просчет проводился в течение 2-х сут. на настольном компьютере, полученные результаты приведены в табл. 10.

Таблица 10. Описание результатов просчета

Table 10. Description of the calculation results

Параметр	Кол-во шагов или этапов генерации	Параметр	Кол-во шагов или этапов генерации
Общее число шагов генерации	624 425 886	Общее число выполнимых формул	3 733 146
Общее число не выполнимых формул	7 264 362	Количество сгенерированных МТ	8797

Таблица 11. Последняя сгенерированная МТ

Table 11. The last generated MT

Аргументы команд			Значения команд (функции: печати, перехода состояний, сдвига головки)			Аргументы команд			Значения команд (функции: печати, перехода состояний, сдвига головки)		
№ команд	Символ	Состояние	Символ	Состояние	Двиг головки	№ команд	Символ	Состояние	Символ	Состояние	Сдвиг головки
1	0	q begin	0	q begin	right	19	e end	q2	# divide	q1	left
2	0	q0 stop	?	?	?	20	e end	q3	# divide	q3	left
3	0	q1	1	q begin	left	21	# divide	q begin	0	q begin	left
4	0	q2	1	q begin	right	22	# divide	q0 stop	?	?	?
5	0	q3	1	q2	right	23	# divide	q1	# divide	q1	left
6	1	q begin	5	q begin	left	24	# divide	q2	5	q1	left
7	1	q0 stop	?	?	?	25	# divide	q3	5	q3	left
8	1	q1	1	q1	left	26	5	q begin	6	q begin	left
9	1	q2	# divide	q1	left	27	5	q0 stop	?	?	?
10	1	q3	6	q2	left	28	5	q1	e end	q2	left
11	b begin	q begin	?	?	?	29	5	q2	6	q1	left
12	b begin	q0 stop	?	?	?	30	5	q3	# divide	q2	left
13	b begin	q1	b begin	q2	right	31	6	q begin	0	q1	left
14	b begin	q2	b begin	q3	right	32	6	q0 stop	?	?	?
15	b begin	q3	b begin	q1	right	33	6	q1	?	?	?
16	e end	q begin	0	q begin	left	34	6	q2	0	q2	right
17	e end	q0 stop	?	?	?	35	6	q3	e end	q3	left
18	e end	q1	5	q2	left						

В настоящем варианте ПКГМТ промежуточные варианты МТ не сохраняются и доступен для рассмотрения только последний вариант сгенерированной МТ, *которая не является решателем NP-трудных задач*. Отметим, что МТ с программой, определенной табл. 11, определяет не одну, а целый спектр МТ, получающийся доопределением команд с неопределенными значениями (???). И все

эти МТ также *не будут решателями NP-трудных задач*.

Без комментариев приведем пример команд МТ-решателя NP-трудных задач (табл. 12), который был получен с использованием эвристик, исключающих возможность найти *все решатели NP-трудных задач* (именно это и необходимо для решения проблемы $P = ? NP$).

Таблица 12. Команды МТ-решателя NP-трудных задач

Table 12. MT-solver commands for NP-hard problems

Состояние	Буква		Состояние	Буква	Сдвиг
0 S r D p 0 n o	2 ☀		1 S r D a 0 n o	2 ☀	1 вправо
1 S r D a 0 n o	0 0 p 0 n		0 S r D p 0 n o	7 0 a 0 n	1 вправо
1 S r D a 0 n o	1 1 p 1 n		2 S r D p 1 n o	13 1 a 1 n	1 вправо
0 S r D p 0 n o	0 0 p 0 n		0 S r D p 0 n o	0 0 p 0 n	1 вправо
0 S r D p 0 n o	1 1 p 1 n		0 S r D p 0 n o	1 1 p 1 n	1 вправо
0 S r D p 0 n o	4 # p 0 n		3 S r R a 0 n o	6 # a 0 n	1 вправо
2 S r D p 1 n o	4 # p 0 n		4 S r R a 1 n o	6 # a 0 n	1 вправо
3 S r R a 0 n o	0 0 p 0 n		5 S R p 0 n o	7 0 a 0 n	0 влево
4 S r R a 1 n o	0 0 p 0 n		6 S R p 1 n o	11 0 a 1 n	0 влево
3 S r R a 0 n o	1 1 p 1 n		5 S R p 0 n o	14 1 a 1 y	0 влево
4 S r R a 1 n o	1 1 p 1 n		7 S R p 0 n c	13 1 a 1 n	0 влево
5 S R p 0 n o	6 # a 0 n		8 S D p 0 n o	6 # a 0 n	0 влево
6 S R p 1 n o	6 # a 0 n		8 S D p 0 n o	6 # a 0 n	0 влево
7 S R p 0 n c	6 # a 0 n		9 S D p 0 n c	6 # a 0 n	0 влево
8 S D p 0 n o	0 0 p 0 n		8 S D p 0 n o	0 0 p 0 n	0 влево
9 S D p 0 n c	0 0 p 0 n		9 S D p 0 n c	0 0 p 0 n	0 влево
8 S D p 0 n o	1 1 p 1 n		8 S D p 0 n o	1 1 p 1 n	0 влево
9 S D p 0 n c	1 1 p 1 n		9 S D p 0 n c	1 1 p 1 n	0 влево
8 S D p 0 n o	7 0 a 0 n		1 S r D a 0 n o	0 0 p 0 n	1 вправо
9 S D p 0 n c	7 0 a 0 n		10 S r D a 0 n c	0 0 p 0 n	1 вправо
8 S D p 0 n o	13 1 a 1 n		12 S r D a 1 n o	1 1 p 1 n	1 вправо
9 S D p 0 n c	13 1 a 1 n		11 S r D a 1 n c	1 1 p 1 n	1 вправо
1 S r D a 0 n o	0 0 p 0 n		0 S r D p 0 n o	7 0 a 0 n	1 вправо
1 S r D a 0 n o	1 1 p 1 n		0 S r D p 0 n o	13 1 a 1 n	1 вправо
10 S r D a 0 n c	0 0 p 0 n		13 S r D p 0 n c	7 0 a 0 n	1 вправо
10 S r D a 0 n c	1 1 p 1 n		13 S r D p 0 n c	13 1 a 1 n	1 вправо
12 S r D a 1 n o	0 0 p 0 n		0 S r D p 0 n o	7 0 a 0 n	1 вправо
12 S r D a 1 n o	1 1 p 1 n		0 S r D p 0 n o	13 1 a 1 n	1 вправо
11 S r D a 1 n c	0 0 p 0 n		14 S r D p 1 n c	7 0 a 0 n	1 вправо
11 S r D a 1 n c	1 1 p 1 n		14 S r D p 1 n c	13 1 a 1 n	1 вправо
1 S r D a 0 n o	6 # a 0 n		15 S r R p 0 y o	6 # a 0 n	1 вправо
10 S r D a 0 n c	6 # a 0 n		16 S r R p 0 n c	6 # a 0 n	1 вправо
12 S r D a 1 n o	6 # a 0 n		17 S r R p 1 y o	6 # a 0 n	1 вправо
11 S r D a 1 n c	6 # a 0 n		18 S r R p 1 y c	6 # a 0 n	1 вправо
15 S r R p 0 y o	0 0 p 0 n		19 S r R p 0 n o	0 0 p 0 n	1 вправо
15 S r R p 0 y o	1 1 p 1 n		15 S r R p 0 y o	1 1 p 1 n	1 вправо
15 S r R p 0 y o	10 1 p 1 y		15 S r R p 0 y o	10 1 p 1 y	1 вправо

16 S <u>r</u> R <u>p</u> 0 <u>y</u> c	0 0 p 0 n		16 S <u>r</u> R <u>p</u> 0 n c	0 0 p 0 n	1 вправо
16 S <u>r</u> R <u>p</u> 0 <u>y</u> c	1 1 p 1 n		16 S <u>r</u> R <u>p</u> 0 n c	1 1 p 1 n	1 вправо
16 S <u>r</u> R <u>p</u> 0 <u>y</u> c	10 1 p 1 y		16 S <u>r</u> R <u>p</u> 0 n c	1 1 p 1 n	1 вправо
17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	0 0 p 0 n		20 S <u>r</u> R <u>p</u> 1 n o	0 0 p 0 n	1 вправо
17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	1 1 p 1 n		17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	1 1 p 1 n	1 вправо
17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	10 1 p 1 y		17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	10 1 p 1 y	1 вправо
18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	0 0 p 0 n		18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	0 0 p 0 n	1 вправо
18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	1 1 p 1 n		18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	1 1 p 1 n	1 вправо
18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	10 1 p 1 y		18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	1 1 p 1 n	1 вправо
15 S <u>r</u> R <u>p</u> 0 <u>y</u> o	4 # p 0 n		20 WIN		СТОП
16 S <u>r</u> R <u>p</u> 0 n c	4 # p 0 n		8 S <u>l</u> D <u>p</u> 0 n o	4 # p 0 n	0 влево
17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	4 # p 0 n		20 WIN		СТОП
18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	4 # p 0 n		8 S <u>l</u> D <u>p</u> 0 n o	4 # p 0 n	0 влево
20 S <u>r</u> R <u>p</u> 1 n o	4 # p 0 n		8 S <u>l</u> D <u>p</u> 0 n o	4 # p 0 n	0 влево
8 S <u>l</u> D <u>p</u> 0 n o	6 # a 0 n		1 S <u>r</u> D a 0 n o	4 # p 0 n	1 вправо
15 S <u>r</u> R <u>p</u> 0 <u>y</u> o	3 ☐		20 WIN		СТОП
16 S <u>r</u> R <u>p</u> 0 n c	3 ☐		22 B <u>l</u> D <u>p</u> 0 n o	3 ☐	0 влево
17 S <u>r</u> R <u>p</u> 1 <u>y</u> o	3 ☐		20 WIN		СТОП
18 S <u>r</u> R <u>p</u> 1 <u>y</u> c	3 ☐		22 B <u>l</u> D <u>p</u> 0 n o	3 ☐	0 влево
20 S <u>r</u> R <u>p</u> 1 n o	3 ☐		22 B <u>l</u> D <u>p</u> 0 n o	3 ☐	0 влево
22 B <u>l</u> D <u>p</u> 0 n o	0 0 p 0 n		22 B <u>l</u> D <u>p</u> 0 n o	0 0 p 0 n	0 влево
22 B <u>l</u> D <u>p</u> 0 n o	1 1 p 1 n		22 B <u>l</u> D <u>p</u> 0 n o	1 1 p 1 n	0 влево
22 B <u>l</u> D <u>p</u> 0 n o	4 # p 0 n		22 B <u>l</u> D <u>p</u> 0 n o	4 # p 0 n	0 влево
22 B <u>l</u> D <u>p</u> 0 n o	10 1 p 1 y		23 B <u>l</u> D a 0 n o	5 1 p 0 n	0 влево
23 B <u>l</u> D a 0 n o	10 1 p 1 y		23 B <u>l</u> D a 0 n o	5 1 p 0 n	0 влево
22 B <u>l</u> D <u>p</u> 0 n o	0 0 p 0 n		23 B <u>l</u> D a 0 n o	0 0 p 0 n	0 влево
23 B <u>l</u> D a 0 n o	1 1 p 1 n		23 B <u>l</u> D a 0 n o	1 1 p 1 n	0 влево
23 B <u>l</u> D a 0 n o	2 ☀		24 B <u>r</u> D a 0 n c	2 ☀	1 вправо
24 B <u>r</u> D a 0 n c	15 ►		23 B <u>l</u> D a 0 n o	15 ►	0 влево
23 B <u>l</u> D a 0 n o	4 # p 0 n		24 B <u>r</u> D a 0 n c	4 # p 0 n	1 вправо
24 B <u>r</u> D a 0 n c	0 0 p 0 n		24 B <u>r</u> D a 0 n c	0 0 p 0 n	1 вправо
24 B <u>r</u> D a 0 n c	1 1 p 1 n		24 B <u>r</u> D a 0 n c	1 1 p 1 n	1 вправо
24 B <u>r</u> D a 0 n c	4 # p 0 n		1 S <u>r</u> D a 0 n o	4 # p 0 n	1 вправо

2.2. Уже на данном начальном этапе обнаруживаются достаточно интересные закономерности, которые, наверное, трудно предсказать (или предвидеть), пытаясь решать проблему умозрительно. Например, все найденные *MT-решатели NP*-трудных задач (блок-тестов ниже) демонстрируют неполиномиальный рост количества шагов решения на следующем блоке тестов:

$$\begin{aligned} & (\neg x_1 \vee x_2 \vee \neg x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6) \wedge (\neg x_1 \vee \neg x_2 \vee \\ & \quad x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6) \wedge \\ & (\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4 \vee \neg x_5 \vee x_6) \wedge (\neg x_1 \vee \neg x_2 \vee \end{aligned}$$

$$\neg x_3 \vee \neg x_4 \vee x_5 \vee x_6) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee \neg x_4 \vee \neg x_5 \vee x_6); \quad (1)$$

$$\begin{aligned}
& (\neg x_1 \vee x_2 \vee \neg x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6 \vee \neg x_7) \wedge (\neg x_1 \vee \\
& \quad \neg x_2 \vee x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6 \vee \neg x_7) \wedge \\
& (\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4 \vee \neg x_5 \vee x_6 \vee \neg x_7) \wedge (\neg x_1 \vee \\
& \quad \neg x_2 \vee \neg x_3 \vee \neg x_4 \vee x_5 \vee \neg x_6 \vee x_7) \wedge \\
& (\neg x_1 \vee \neg x_2 \vee \neg x_3 \vee \neg x_4 \vee \neg x_5 \vee x_6 \vee x_7) \wedge (\neg x_1 \vee \\
& \quad \neg x_2 \vee \neg x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6 \vee x_7).
\end{aligned} \tag{2}$$

Следующие формулы строятся по принципу перехода от формулы (1) к формуле (2). Конкретно, МТ с программой по табл. 12 проверяет выполнимость формулы (1) за 101 шаг;

формулы (2) за 240 шагов и так далее, т.е. рост количества шагов решения идет по экспоненте, хотя, конечно, будут сгенерированы другие решатели, где это будет происходить иначе. Следует из того, что МТ реализуют любые алгоритмы [9] и значит, может быть, придется работать с отгенерированными МТ «индивидуально».

3. Математические аспекты создания программного комплекса ПКГМТ генерации МТ.

Данный раздел практически цитата из статьи [1]. В работе [1] генерация МТ для решения NP-трудных задач была рассмотрена достаточно бегло, причем перебор программ МТ будем оптимизировать методами УО.

Машина Тьюринга имеет следующие части:

1) *Конечная лента* L , разделенная на ячейки. В процессе работы ДМТ с обеих сторон ленты могут пристраиваться новые ячейки, поэтому ленту L можно считать *потенциально бесконечной*. В каждой из ячеек прописан некоторый символ из алфавита $A = \{a_0, a_1, a_2, \dots, a_n\}$, т.е. в ячейках ленты a_{ij} являются символами алфавита A . В процессе работы МТ символы в ячейках ленты L могут меняться (или не меняться). Ленту L называют также *внешней памятью МТ*. При пристраивании новой ячейки слева или справа предполагаем, что в ней прописывается один и тот же выделенный символ алфавита $A = \{a_0, a_1, \dots, a_n\}$, а именно, a_0 , причем в этом случае ячейка считается пустой.

2) *Множество состояний МТ* $Q = \{q_0, q_1, q_2, \dots, q_m\}$, элементы которого называются *внутренней памятью*. В дальнейшем предполагаем, что МТ *всегда находится только в одном состоянии* (почему называется детерминированной, а не детерминированная машина Тьюринга может одновременно находиться более чем в одном состоянии). Состояние q_0 называется *заключительным* (или *стоп-состоянием*) при получении которого работа МТ прекращается.

3) *Управляющая головка*. Это устройство считывает информацию *только* из одной ячейки ленты L .

4) *Схема работы МТ*. Один такт работы МТ состоит в считывании головкой информации из ячейки ленты МТ и прописывания в эту ячейку некоторого символа алфавита $A = \{a_0, a_1, \dots, a_n\}$ и сдвига головки (*обязательно!*) МТ в соседнюю левую или правую ячейку, а ДМТ переходит в некоторое состояние из множества $Q = \{q_0, q_1, q_2, \dots, q_m\}$. Отметим, что МТ может работать *бесконечно* (или *вечно*), так как заключительное состояние q_0 мо-

жет оказаться *недостижимым*. Состоянием (машинным словом или конфигурацией) МТ будем называть совокупность, образованную последовательностью $a_{j1}, a_{j2}, \dots$, символов всех ячеек ленты, состоянием внутренней памяти q_i и номером k воспринимаемой ячейки a_{jk} .

Таким образом, каждое машинное слово содержит только *одно* вхождение символа состояния q_i . Для рассматриваемого здесь машинного слова символ q_i стоит на позиции k , что позволяет записать его следующей последовательностью:

$$a_{j1} a_{j2} \dots a_{jk-1} q_i a_{ik} \dots a_{js}. \quad (3)$$

Важно отметить, что каждое машинное слово всегда заканчивается символом алфавита (если головка сдвигается вправо от последнего символа машинного слова, то справа приписывается символ a_0 , т.е. справа к ленте приписывается пустая ячейка), но может начинаться символом состояния q_i .

5) *Функции перехода, печати и сдвига*.

Функция перехода состояний:

$Pass : (Q \times A) \rightarrow Q$;

Функция печати:

$Print : (Q \times A) \rightarrow A$;

Функция сдвига головки:

$Displ : (Q \times A) \rightarrow \{-1, 1\}$.

Рассмотрим преобразование машинного слова (16) после применения выше определенных функций. Пусть $Pass(q_i, a_{ik}) = q_\xi$; $Print(q_i, a_{ik}) = a_\lambda$; $Displ(q_i, a_{ik}) = t$. Тогда в зависимости от значения t (-1 соответствует сдвигу головки по ленте влево, а 1 – сдвигу головки вправо) возможны следующие два варианта.

1 *вариант*. Пусть $t = -1$. Тогда машинное слово (3) будет преобразовано в последовательность

$$a_{j1} a_{j2} \dots q_\xi a_{jk-1} a_\lambda \dots a_{js}.$$

2 *вариант*. Пусть $t = 1$. Тогда машинное слово (3) будет преобразовано в последовательность

$$a_{j1} a_{j2} \dots a_{jk-1} a_\lambda q_\xi a_{jk+1} \dots a_{js}.$$

Функции $Pass$, $Print$, $Displ$ называют *программой МТ*, а начальную ленту L – *задачей*. На ленте МТ можно записывать классические NP-трудные задачи: выполнимости (SAT), раскраски графов, существования покрытия и др. [5].

Методы УО можно применять для ускорения перебора входных данных (задачи на ленте МТ), а также для ускорения перебора программ МТ (функций $Pass$, $Print$, $Displ$), что, возможно, обещает весьма неожиданные результаты. В статье [1] отмечается, что *генерация программ* (хотя бы только для МТ) является *принципиально новым* шагом, а

в настоящей работе показаны результаты, которые могут быть значительно продвинуты вперед в ближайшее время.

ЗАКЛЮЧЕНИЕ

1. Создание открытой платформы по генерации программ МТ, решающих *NP*-трудные задачи, будет интересным экспериментом, создающим возможность участия как студентов, изучающих информационные технологии, так и продвинутых исследователей проблемы $P = ? NP$.

2. Кроме того, возможны эксперименты по созданию решателей для других *NP*-трудных

задач, коих, имеющих большое прикладное значение, очень и очень много. Например, [23–28]:

- приближительные вычисления;
- конфигурация;
- криптография;
- сбор данных;
- поддержка при принятии решения;
- филогенетика;
- планирование;
- расписания;
- маршрутизация.

СПИСОК ИСТОЧНИКОВ

1. Мартьянов В.И. *NP-трудные задачи: автоматическое доказательство теорем и машины Тьюринга* // Baikal Research Journal. 2021. Т.12. № 4. С. 11–20. [https://doi.org/10.17150/2411-6262.2021.12\(4\).11](https://doi.org/10.17150/2411-6262.2021.12(4).11).
2. Мартьянов В.И. Логико-эвристические методы сетевого планирования и распознавание ситуаций // Проблемы управления и моделирования в сложных системах. Самара, 2001. С. 209–215.
3. Мартьянов В.И., Архипов В.В., Каташевцев М.Д., Пахомов Д.В. Обзор приложений логико-эвристических методов решения комбинаторных задач высокой сложности // Современные технологии. Системный анализ. Моделирование. 2010. № 4 (28). С. 205–211. EDN: NRBKXP.
4. Мартьянов В.И., Сухорутченко В.В., Окунцов В.В. Планирование информационных потоков в иерархической системе // Прикладные системы. М.: ИАП РАН, 1992. С. 46–58.
5. Мартьянов В.И., Вяткин И.В., Могильницкий Э.Ю. Теоретический и реализационный аспекты некоторых классов задач сетевого управления // Проблемы управления и моделирования в сложных системах. Самара, 1999. С. 203–208.
6. Лагутенков А.В. Криптовалюты. Правила применения // Наука и жизнь. 2018. № 2. С. 22–26.
7. Kharpal A. All you need to know about the top 5 cryptocurrencies // Yahoo! [Электронный ресурс]. URL: <https://finance.yahoo.com/news/know-top-5-cryptocurrencies-093231173.html> (14.03.2023).
8. Wilson-Nunn D., Zenil N. On the Complexity and Behaviour of Cryptocurrencies Compared to Other Markets // Research Gate [Электронный ресурс]. URL: https://www.researchgate.net/publication/268079329_On_the_Complexity_and_Behaviour_of_Cryptocurrencies_Compared_to_Other_Markets (14.03.2023).
9. Щербина О.А. Удовлетворение ограничений и программирование в ограничениях (препринт). Vienna: University of Vienna, 2012. 118 с.
10. Hentenryck van P. Constraint Satisfaction in Logic Programming. Cambridge: The MIT Press, 1989. 356 p.
11. Стюарт И. Величайшие математические задачи / Пер. с англ. Н. Лисовой. М.: Альпина нон-фикшн, 2015. 460 с.
12. Разборов А.А. Алгебраическая сложность. М.: Московский центр непрерывного математического образования, 2016. 32 с.
13. Fortnow L. The Golden Ticket: P, NP, and the Search for the Impossible. Princeton: Princeton University Press, 2013.
14. Мальцев А.И. Алгоритмы и рекурсивные функции. М.: Наука, 1965. 392 с.
15. Корольков Ю.Д., Мартьянов В.И. Дискретные модели: представление конечными деревьями и разрешимость формальных теорий. Иркутск: Изд-во Иркутского национального исследовательского технического университета, 2017. 160 с.
16. Кулик Н.С., Мартьянов В.И., Пахомов Д.В., Большаков Э.А. Проект системы управления региональной сетью автомобильных дорог (СУРАД) Иркутской области // Вестник Иркутского государственного технического университета. 2014. № 4 (87). С. 118–123. EDN: SBNFNJ.
17. Кулик Н.С., Мартьянов В.И., Пахомов Д.В. Построение графа автомобильных дорог для системы взимания платы с большегрузного транспорта // Вестник Иркутского государственного технического университета. 2016. № 4 (111). С. 96–101. EDN: VVSOFF. <https://doi.org/10.21285/1814-3520-2016-4-96-101>.
18. Мартьянов В.И., Пахомов Д.В., Архипов В.В. Сетевое планирование содержания сети автомобильных дорог Иркутской области // Новые технологии в инвестиционно-строительной сфере и ЖКХ: сб. науч. трудов: в 2 т. Т. 1. Иркутск: Иркутский государственный технический университет, 2005. С. 123–129.
19. Мартьянов В.И. Теоретико-множественные модели данных в задаче расчета вторичных структур РНК // System Analysis & Mathematical Modeling. 2022. Т. 4. № 4. С. 343–357. EDN: VUNLFX. [https://doi.org/10.17150/2713-1734.2022.4\(4\).343-357](https://doi.org/10.17150/2713-1734.2022.4(4).343-357).
20. Мартьянов В.И., Корольков Ю.Д. Теоретико-множественные модели сложных систем и алгоритмы интеллектуальной поддержки. Иркутск:

Изд-во Байкальского государственного университета, 2020. 188 с.

21. Mart'yanov V.I., Katashevsev M.D. Computer processing of distorted video sequences obtained by mobile road laboratories // Investments, Construction, Real Estate: New Technologies and Special-Purpose Development Priorities: IOP Conference Series: Materials Science and Engineering (Irkutsk, 25 April 2019). 2019. Vol. 667. <https://doi.org/10.1088/1757-899X/667/1/012060>.
22. Мартьянов В.И. Теоретико-множественный анализ организации данных учебного процесса и алгоритмы проектирования расписания с элементами искусственного интеллекта // Прикладные проблемы дискретного анализа: сб. науч. тр. 2021. С. 90–100. EDN: BVANFU.
23. Leeuwen J.V. Handbook of Theoretical Computer Science. Vol. A, Algorithms and complexity. Amsterdam: Elsevier, 1998. 1020 p.

24. Knuth D. Postscript about NP-hard problems // ACM SIGACT News. 1974. Vol. 6. No. 2. P. 15–16. <https://doi.org/10.1145/1008304.1008305>.
25. Aaronson S. The Scientific Case for P≠NP // Shtetl-Optimized [Электронный ресурс]. URL: <https://scottaaronson.blog/?p=1720> (25.09.2016).
26. Aaronson S. PHYS771 Lecture 6: P, NP, and Friends // Scott Aaronson.ru [Электронный ресурс]. URL: <https://www.scottaaronson.com/democritus/lec6.html> (25.09.2016).
27. Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B. The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. John Wiley & Sons, 1985. 75 p.
28. Wegener I. Complexity Theory: Exploring the Limits of Efficient Algorithms. Springer, 2005. P. 189. <https://doi.org/10.1007/3-540-27477-4>.

REFERENCES

1. Mart'yanov V.I. NP-difficult tasks: automatic proof of theorems and Turing's machine. *Baikal Research Journal*. 2021;12(4):11. (In Russ.). [https://doi.org/10.17150/2411-6262.2021.12\(4\).11](https://doi.org/10.17150/2411-6262.2021.12(4).11).
2. Mart'yanov V.I. Logical-heuristic methods of network planning and situation recognition. *Problemy upravleniya i modelirovaniya v slozhnykh sistemakh*. Samara; 2001. p. 203-215. (In Russ.).
3. Mart'yanov V.I., Arkhipov V.V., Katashevsev V.I. Mart'yanov, V.V. Arkhipov, M., Pakhomov D.V. Logic-heuristic methods for solving combinatorial problems of high complexity applications preview. *Sovremennye tekhnologii. Sistemnyi analiz. Modelirovanie = Modern technologies. System analysis. Modeling*. 2010;4:205-211. (In Russ.). EDN: NRBKXP.
4. Mart'yanov V.I., Sukhorutchenko V.V., Okuntsov V.V. Planning of information flows in a hierarchical system. In: *Prikladnye sistemy*. Moscow: Institute of Design Automation of the Russian Academy of Sciences; 1992. p. 46–58. (In Russ.).
5. Mart'yanov V.I., Vyatkin I.V., Mogil'nitskii E.Yu. Theoretical and implementation aspects of some classes of network management tasks. *Problemy upravleniya i modelirovaniya v slozhnykh sistemakh*. Samara; 1999. p. 203-208. (In Russ.).
6. Lagutenkov A.V. Cryptocurrencies. Application rules. *Nauka i zhizn'*. 2018;2:22-26. (In Russ.).
7. Kharpal A. All you need to know about the top 5 cryptocurrencies. *Yahoo!* Available from: <https://finance.yahoo.com/news/know-top-5-cryptocurrencies-093231173.html> [Accessed 14th March 2023].
8. Wilson-Nunn D., Zenil H. On the Complexity and Behaviour of Cryptocurrencies Compared to Other Markets. Research Gate. Available from: https://www.researchgate.net/publication/268079329_On_the_Complexity_and_Behaviour_of_Cryptocurrencies_Compared_to_Other_Markets [Accessed 14th March 2023].
9. Shcherbina O.A. Uдовлетворение ограничений i programmirovaniye v ogranicheniyakh (preprint). Vienna: University of Vienna; 2012. 118 p. (In Russ.).
10. Hentenryck van P. Constraint Satisfaction in Logic Programming. Cambridge: The MIT Press; 1989. 356 p.
11. Stewart I. The great mathematical problems. Moscow: Al'pina non-fikshn; 2015. 460 p. (In Russ.).
12. Razborov A.A. Algebraic complexity. Moscow: Moscow Center for Continuing Mathematical Education; 2016. 32 p. (In Russ.).
13. Fortnow L. The Golden Ticket: P, NP, and the Search for the Impossible. Princeton: Princeton University Press; 2013.
14. Mal'tsev A.I. Algorithms and recursive functions. Moscow: Nauka; 1965. 392 p.
15. Korolkov Yu.D., Mart'yanov V.I. Discrete models: representation by finite trees and solvability of formal theories. Irkutsk: Irkutsk National Research Technical University; 2017. 160 p. (In Russ.).
16. Mart'yanov V.I., Kulik N.I., Pakhomov D.V., Bolshakov E.A. Project of the system to control Irkutsk regional automobile road network. *Vestnik Irkutskogo gosudarstvennogo tekhnicheskogo universiteta = Proceedings of Irkutsk State Technical University*. 2014;4:118-123. EDN: SBNFNJ.
17. Kulik N.S., Mart'yanov V.I., Pakhomov D.V. Building a motorway graph for the heavy trucks tolling system. *Vestnik Irkutskogo gosudarstvennogo tekhnicheskogo universiteta = Proceedings of Irkutsk State Technical University*. 2016. № 4 (111). С. 96–101. (In Russ.). EDN: VVSOF.
18. Mart'yanov V.I., Pakhomov D.V., Arkhipov V.V. Network planning of the maintenance of the highway network of the Irkutsk region. *Novye tekhnologii v investitsionno-stroitel'noi sfere i ZhKKh: sb.*

nauch. trudov: in 2 vol. Vol. 1. Irkutsk: Irkutsk State Technical University; 2005. p. 123-129. (In Russ.).

19. Martyanov V.I. Set-theoretical data models in the problem of calculating secondary structures of RNA. *System Analysis & Mathematical Modeling*. 2022;4(4):343-357. EDN: VUNLFX. [https://doi.org/10.17150/2713-1734.2022.4\(4\).343-357](https://doi.org/10.17150/2713-1734.2022.4(4).343-357).

20. Martyanov V.I., Korolkov Yu.D. Set-theoretic models of complex systems and algorithms of intellectual support. Irkutsk: Baikal State University; 2020. 188 p. (In Russ.).

21. Martyanov V.I., Katashevsev M.D. Computer processing of distorted video sequences obtained by mobile road laboratories. *Investments, Construction, Real Estate: New Technologies and Special-Purpose Development Priorities: IOP Conference Series: Materials Science and Engineering* (Irkutsk, 25 April 2019). 2019. Vol. 667. <https://doi.org/10.1088/1757-899X/667/1/012060>.

22. Martyanov V.I. Set-theory analysis of the data organization and algorithms for net planning with elements artificial intelligence. *Prikladnye problemy diskretnogo analiza: sbornik nauchnyh trudov*.

2021;90-100. (In Russ.). EDN: BVANFU.

23. Leeuwen J.V. Handbook of Theoretical Computer Science. Vol. A, Algorithms and complexity. Amsterdam: Elsevier, 1998. 1020 p.

24. Knuth D. Postscript about NP-hard problems. *ACM SIGACT News*. 1974;6(2):15-16. <https://doi.org/10.1145/1008304.1008305>.

25. Aaronson S. The Scientific Case for P≠NP. *Shtetl-Optimized*. Available from: <https://scottaaronson.blog/?p=1720> (25.09.2016). [Accessed 14th March 2023].

26. Aaronson S. PHYS771 Lecture 6: P, NP, and Friends. *Scott Aaronson.ru*. Available from: <https://www.scottaaronson.com/democritus/lec6.html> [Accessed 14th March 2023].

27. Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B. The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, John Wiley & Sons, 1985.

28. Wegener I. Complexity Theory: Exploring the Limits of Efficient Algorithms. Springer, 2005. P. 189. <https://doi.org/10.1007/3-540-27477-4>.

Информация об авторе

Мартьянов Владимир Иванович,
д.ф.-м.н., профессор кафедры
автомобильных дорог,
Иркутский национальный исследовательский
технический университет,
664074, г. Иркутск, ул. Лермонтова, 83, Россия,
Байкальский государственный университет,
664003, г. Иркутск, ул. Ленина, 11, Россия
Иркутский государственный университет,
664003, г. Иркутск, ул. Карла Маркса, 1, Россия
e-mail: MartVilv@mail.ru
<https://orcid.org/0000-0003-2659-0355>

Information about the author

Vladimir I. Martyanov
Dr. Sci. (Phys.-Math.), Professor
of the Department of Highways,
Irkutsk National Research
Technical University,
83 Lermontov St., Irkutsk 664074, Russia,
Baikal State University,
11 Lenin St., Irkutsk 664003, Russia
Irkutsk State University,
1 Karl Marx St., Irkutsk 664003, Russia
e-mail: MartVilv@mail.ru
<https://orcid.org/0000-0003-2659-0355>

Вклад автора

Автор провел исследование, подготовил рукопись к печати и несет ответственность за плагиат.

Contribution of the author

Autor has conducted the study, prepared the manuscript for publication and bears the responsibility for plagiarism.

Конфликт интересов

Автор заявляет об отсутствии конфликта интересов.

Conflict of interests

The author declare no conflict of interests regarding the publication of this article.

Автор прочитал и одобрил окончательный вариант рукописи.

The final manuscript has been read and approved by the author.

Информация о статье

Статья поступила в редакцию 16.03.2023.
Одобрена после рецензирования 10.04.2023.
Принята к публикации 11.04.2023.

Information about the article

The article was submitted 16.03.2023.
Approved after reviewing 10.04.2023.
Accepted for publication 11.04.2023.