

УДК 004.01:006.72 (470.22)

ПРОВЕРКА ВЫПОЛНИМОСТИ БУЛЕВЫХ ФОРМУЛ С ПОМОЩЬЮ КВАНТОВОГО ОТЖИГА

М. А. Мальцева, А. С. Румянцев*

*Институт прикладных математических исследований КарНЦ РАН,
ФИЦ «Карельский научный центр РАН» (ул. Пушкинская, 11, Петрозаводск,
Республика Карелия, Россия, 185910), *ar0@krc.karelia.ru*

Проверка выполнимости булевых формул (SAT) широко применяется при решении современных задач автоматизации проектирования микроэлектроники. Важным приложением такой задачи является проектирование логических цепей с заданными характеристиками из заданного набора логических вентилях. Для решения SAT-задачи можно применять как стандартные решатели, так и современные оптимизационные инструменты, в том числе относительно новый инструмент аппаратного решения задачи квантового отжига. Решение данной задачи может выполняться на системе квантового отжига с помощью построения по особым правилам так называемой булевой сети и выбора соответствующей целевой функции. В статье рассматривается практическое применение квантового отжига к построению коротких логических цепей с помощью решения SAT-задачи на основе предложенных в литературе методов, а также проводится анализ практических аспектов применения таких методов к построению квантовых цепей.

Ключевые слова: задача выполнимости булевых формул; квантовый отжиг; логические цепи; квантовые цепи; бинарное представление

Для цитирования: Мальцева М. А., Румянцев А. С. Проверка выполнимости булевых формул с помощью квантового отжига // Труды Карельского научного центра РАН. 2023. № 4. С. 41–49. doi: 10.17076/mat1783

Финансирование. Финансовое обеспечение исследований осуществлялось из средств Российского научного фонда (грант № 21–71–10135).

M. A. Maltseva, A. S. Rumyantsev*. BOOLEAN SATISFIABILITY VERIFICATION BY QUANTUM ANNEALING

*Institute of Applied Mathematical Research, Karelian Research Centre, Russian Academy of Sciences (11 Pushkinskaya St., 185910 Petrozavodsk, Karelia, Russia), *ar0@krc.karelia.ru*

The Boolean satisfiability problem (SAT) is widely applied in modern tasks of microelectronics design automation. An important application of such a task is the design of logic circuits with specified characteristics from a given set of logic gates. To solve an SAT problem, both standard solvers and modern optimization tools can be used, including a relatively new hardware tool for solving the quantum

annealing problem. The solution of this problem can be performed on a quantum annealing system by constructing a so-called Boolean network according to special rules and selecting an appropriate objective function. The article discusses the practical application of quantum annealing to the construction of short logic circuits by solving an SAT problem based on the methods proposed in the literature, and also analyzes the practical aspects of applying such methods to the construction of quantum circuits.

Key words: Boolean satisfiability problem; quantum annealing; logical circuits; quantum circuits; binary representation

For citation: Maltseva M. A., Rumyantsev A. S. Boolean satisfiability verification by quantum annealing. *Trudy Karelskogo nauchnogo tsentra RAN = Transactions of the Karelian Research Centre RAS*. 2023;4:41–49. doi: 10.17076/mat1783

Funding. The study was funded under Russian Science Foundation (project # 21–71–10135).

ВВЕДЕНИЕ

Проверка выполнимости булевых формул (Boolean satisfiability problem, SAT-задача) широко применяется при решении современных задач автоматизации проектирования микроэлектроники [2, 9, 21, 24] и при проектировании классических логических цепей [3]. В общем случае SAT-задача относится к классу так называемых NP-полных [5] (задач, для которых не доказано существование алгоритма, решающего их за полиномиальное относительно размера входных данных время). В связи с быстрым ростом сложности таких задач относительно размера исходных данных широко применяются методы ускорения расчетов с помощью распределенных [1] и параллельных [14] вычислений. В этой связи перспективным является применение квантовых вычислительных систем, для которых известны алгоритмы, превосходящие алгоритмы на классических системах с точки зрения вычислительной сложности (например, алгоритм Шора [17]).

Среди разработанных прототипов квантовых вычислителей следует выделить системы, работающие по принципу квантового отжига (КО). Такие системы позволяют выполнять решение задач оптимизации (в частности, при квадратичной целевой функции, бинарных переменных и отсутствии ограничений – задачи QUBO) и не являются универсальными (не позволяют выполнять произвольные вычисления) [12, 20], однако обладают большей вычислительной мощностью в сравнении с универсальными квантовыми процессорами (например, система, основанная на КО, D-Wave 2000Q имеет 2048 кубитов (квантовых битов, вычислительных единиц системы) [6], в

то время как универсальный квантовый процессор IBM Osprey имеет 433 кубита [11]). В то же время возможностей КО достаточно для решения SAT-задачи, что объясняет интерес исследователей к построению алгоритмов для указанного класса задач на системах типа КО [4, 18, 19].

Важным приложением SAT-задачи является проектирование логических цепей с заданными характеристиками из заданного набора логических вентилях (кодирование с помощью SAT-задачи функционала логической цепи [3]). Решение данной задачи может выполняться на системе КО с помощью построения по особым правилам так называемой булевой сети и выбора соответствующей целевой функции [18]. Предполагается, что такой метод может применяться и для построения некоторых классов квантовых цепей [3, 13, 16, 22], т. е. цепей из так называемых квантовых вентилях (эта модель квантовых вычислений является универсальной [15]). Однако при этом необходимо обеспечить бинарное представление всех возможных промежуточных состояний квантовой цепи, что является открытым вопросом [3]. Целью данной статьи является практическое применение КО к построению коротких логических цепей с помощью решения SAT-задачи на основе методов, предложенных в [4, 18], а также анализ практических аспектов применения метода к построению квантовых цепей.

В работе выполнена экспериментальная проверка метода кодирования логической цепи и решения SAT-задачи на ее основе с помощью симулятора машины D-Wave и рассмотрены особенности применения данного метода к кодированию квантовых цепей.

КОДИРОВАНИЕ ЛОГИЧЕСКИХ ЦЕПЕЙ С ПОМОЩЬЮ КВАНТОВОГО ОТЖИГА

КО – эмпирический метод решения задач оптимизации, который позволяет найти глобальный минимум заданной целевой функции [7]. КО является вариантом широко применяемого эмпирического оптимизационного алгоритма имитации отжига [12]. В отличие от последнего КО использует так называемое квантовое туннелирование (квантовые флуктуации) для преодоления локальных минимумов целевой функции при осуществлении поиска решения [20, 23], что отличает КО от большинства алгоритмов, основанных на так называемом восхождении к вершине (generalized hill climbing). Реализация КО возможна как на традиционной вычислительной архитектуре (в режиме имитации), так и (физически) на специализированных квантовых вычислительных системах, таких как системы D-Wave. Особенности расчета на физических системах заключаются в необходимости учета топологии графа возможных связей (так называемой квантовой запутанности) между парами кубитов, а также в ограниченности числа кубитов.

Одним из видов задач оптимизации, решаемых с помощью КО, является поиск конфигурации так называемых спинов, минимизирующей энергию в модели Изинга, т. е. минимизация целевой функции

$$\mathcal{E}(\underline{z}) = \sum_{i \in V} h_i z_i + \sum_{(i,j) \in E} J_{ij} z_i z_j, \quad (1)$$

где

- $G = (V, E)$ – неориентированный граф возможных взаимодействий между кубитами (подграф топологии квантовой системы),
- $\underline{z} = \{-1, 1\}^{|V|}$ – конфигурация спинов, значения которых указывают энергетическое состояние соответствующих кубитов,
- симметричная матрица $J = \|J_{ij}\|_{(i,j) \in E}$ задает энергию взаимодействия пар кубитов,
- вектор $h = \|h_i\|_{i \in V}$ задает внешнее поле, воздействующее на отдельные кубиты.

В системах D-Wave имеются следующие ограничения на используемые коэффициенты целевой функции: $h_i \in [-2, 2]$, $J_{ij} \in [-1, 1]$ [18]. Заметим, что задачу минимизации $\mathcal{E}(\underline{z})$ в эквивалентной форме в булевом базисе

$\{0, 1\}^{|V|}$ принято называть задачей квадратичной неограниченной (безусловной) бинарной оптимизации (двоичной оптимизации в виде квадратической функции с неограниченным диапазоном значений, quadratic unconstrained binary optimization, QUBO) [18]. Обе указанные формы задачи могут быть решены с помощью эвристики КО, для чего требуется задать матрицу J и вектор h .

Напомним, что SAT-задача заключается в поиске такого набора значений $\underline{x} = (x_1, \dots, x_n) \in \{0, 1\}^n$ булевой формулы F , при котором формула является выполнимой, т. е. F истинна [8].

Пример 1. Рассмотрим булеву формулу с двумя переменными $F = x_1 \wedge x_2$, где \wedge – операция логического «И». Она выполнима, так как при $x_1 = 1$ и $x_2 = 1$ F истинна.

В общем случае решение SAT-задачи сводится к перебору всех возможных значений переменных, входящих в заданную булеву формулу, для поиска набора, выполняющего формулу, либо доказательства отсутствия такового.

Подход к решению SAT-задач с помощью КО предложен в работах [4, 18, 19]. Суть подхода состоит в следующих этапах:

1. Представление булевой формулы в виде логической цепи с соответствующими логическими операциями;
2. Построение элементов матрицы J и вектора h для функции \mathcal{E} с помощью решения специальным образом построенной задачи линейного программирования (ЛП) в отдельности для каждой логической операции, входящей в цепь;
3. Иерархическое построение итоговых матрицы J и вектора h путем объединения логических операций цепи, связанных отношением предшествования (если выходное значение логической операции является входным значением другой операции) по особым правилам;
4. Добавление выходной переменной цепи (например, z_n) с присвоением максимального (положительного) коэффициента h_n и максимального по модулю отрицательного веса соответствующей дуги.

Таким образом, на этапах 1–3 фактически выполняется кодирование логической цепи с помощью матрицы J и вектора h , а на заключительном этапе 4 – постановка SAT-задачи. Рассмотрим далее указанные этапы более подробно.

Для экономии обозначений положим здесь, что цепь состоит из одной логической операции с n операндами. Согласно [18], для заданной логической операции следует таким образом подобрать коэффициенты матрицы J и вектора h , чтобы обеспечить достижение минимального значения энергии (1) на наборах $\underline{z} \in C$, входящих в таблицу истинности C указанной операции (это значение энергии обозначим l). При этом для значений $\underline{z} \in \{-1, 1\}^n \setminus C$ обеспечивается энергетический штраф не менее $g > 0$, причем необходимо максимизировать указанный штраф. Таким образом, решается следующая задача линейного программирования [18]:

$$\begin{aligned} g &\rightarrow \max \\ \mathcal{E}(\underline{z}) &= l, & \underline{z} &\in C, \\ \mathcal{E}(\underline{z}) &\geq g + l, & \underline{z} &\in \{-1, 1\}^n \setminus C, \\ -2 &\leq h_i \leq 2, & i &\in V, \\ -1 &\leq J_{ij} \leq 1, & (i, j) &\in E, \\ g &> 0. \end{aligned} \quad (2)$$

Отметим, что l есть энергия системы в так называемом основном состоянии (состоянии с минимальной энергией), а величина g представляет собой промежуток, который отделяет основное состояние от остальных (возбужденных) состояний. Чем больше g , тем больше вероятность получить основное состояние в результате выполнения КО и меньше время, требуемое для выполнения КО. Таким образом, если указанная задача (2) имеет решение, то результатом выполнения КО с соответствующей целевой функцией (1) при h, J , являющихся решением (2), будет с большой вероятностью один из элементов таблицы истинности C . Заметим, что, в связи со спиновым представлением энергии (1), значения булевых переменных кодируются следующим образом: «-1» – ИСТИНА, «1» – ЛОЖЬ.

Поясним кодирование одиночной логической операции (этап 2) на следующем примере.

Пример 2. Рассмотрим в качестве примера логическую операцию ИЛИ-НЕ (NOR), соответствующую булевой формуле $F = \neg(x_0 \vee x_1)$. Обозначим x_2 выходное значение логической операции NOR, имеющей на входе значения x_0, x_1 . Таблица истинности для соответствующей логической операции представлена в таблице 1. Таким образом, для логической операции ИЛИ-НЕ множество троек C , соответствующих таблице истинности, в задаче оптимизации (2) имеет вид

$$C = \{(1, 1, -1), (1, -1, 1), (-1, 1, 1), (-1, -1, 1)\}.$$

Таблица 1. Таблица истинности операции ИЛИ-НЕ $F = \neg(z_0 \vee z_1)$ в спиновой форме записи
Table 1. Truth table of the NOR operation $F = \neg(z_0 \vee z_1)$ in spin notation

z_0	z_1	z_2
1	1	-1
1	-1	1
-1	1	1
-1	-1	1

Таким образом, нужно решить задачу ЛП (2) с переменными $h_0, h_1, h_2, J_{01}, J_{02}, J_{12}, l, g$. Для решения задачи ЛП воспользуемся пакетом **highs** для среды вычислений R. Решение имеет вид: $h = (-0.5, -0.5, -1)$, $J_{01} = 0.5, J_{02} = 1, J_{12} = 1, g = 2, l = -1.5$. На рис. 1 представлено полученное решение в виде графа, где соответствующие веса h_i указаны для вершин $i = 0, 1, 2$, а веса J_{ij} указаны для ребер $(i, j), i, j = 0, 1, 2$ (отметим, что граф полносвязный). Как показывает экспериментальная проверка на симуляторе машины D-Wave, решение задачи минимизации энергии (1) с помощью КО для полученных вектора h и матрицы J возвращает конфигурации спинов, которые составляют таблицу истинности (табл. 1).

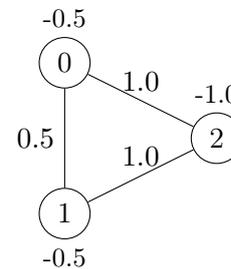


Рис. 1. Граф с весовыми коэффициентами, полученными путем решения задачи ЛП (2) для операции ИЛИ-НЕ

Fig. 1. Graph with weight coefficients obtained by solving the LP problem (2) for the NOR operation

Заметим, что задача (2) может не иметь решений. В этом случае может потребоваться добавление дополнительных вспомогательных переменных [4], однако обсуждение этой техники лежит за рамками данной статьи.

Для построения цепи из нескольких логических операций в [18] предлагается получить соответствующие коэффициенты для отдельных операций, а затем обеспечить, чтобы переменные в связанных логических операциях (например, выходная переменная одной операции, являющаяся входной для другой операции) принимали одно и то же значение. Для

этого устанавливается наибольший по модулю отрицательный весовой коэффициент $J_{ij} = -1$ для всех пар i, j переменных, для которых требуется обеспечить равенство $z_i = z_j$ (действительно, в этом случае $J_{ij}z_i z_j$ принимает наименьшее значение).

Приведем пример объединения двух логических операций в логическую цепь (этап 3).

Пример 3. Рассмотрим булеву формулу $F_2 = \neg(\neg(z_0 \vee z_1) \vee z_4)$. Такая формула эквивалентна системе логических операций

$$\begin{cases} z_2 = \neg(z_0 \vee z_1), \\ z_5 = \neg(z_3 \vee z_4), \\ z_3 = z_2. \end{cases} \quad (3)$$

Таким образом, логическая цепь строится на основе полученных в Примере 2 логических операций ИЛИ-НЕ путем добавления ребра с весовым коэффициентом $J_{23} = -1$ между выходной переменной z_2 первого логического элемента и входной переменной z_3 второго логического элемента. Граф логической цепи для формулы F_2 , построенный путем объединения графов, представлен на рис. 2.

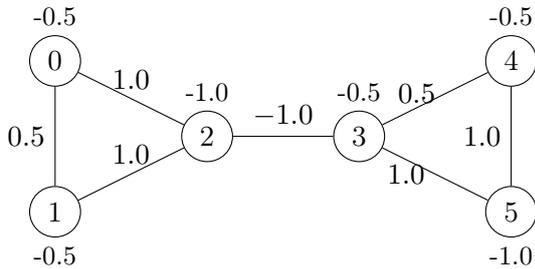


Рис. 2. Граф с весовыми коэффициентами, полученными путем решения задачи ЛП (2) и применения правил этапа 3 для булевой формулы F_2
Fig. 2. Graph with weights obtained by solving the LP problem (2) and applying the rules of stage 3 for the Boolean formula F_2

Наконец, покажем пример постановки SAT-задачи (этап 4) на примере булевой формулы F_2 , кодирование которой выполнено в Примере 3.

Пример 4. Для получения конфигурации спинов, при которой F_2 истинна, следует добавить переменную z_6 , эквивалентную выходной переменной z_5 формулы F_2 . Для того чтобы обеспечить минимизацию энергии при $z_6 = -1$, коэффициент h_6 следует выбрать максимальным, т. е. $h_6 = 2$ (для машины D-Wave). Наконец, аналогично этапу 3,

для обеспечения равенства $z_5 = z_6$ выбираем $J_{56} = -1$. Граф логической цепи для SAT-задачи на основе формулы F_2 представлен на рис. 3. Экспериментальная проверка на симуляторе машины D-Wave показывает, что решение задачи КО возвращает конфигурации переменных, которые составляют те строки таблицы истинности формулы F_2 в форме (3) (табл. 2), при которых F_2 истинно ($F_2 = -1$). При 100 запусках КО возвращает конфигурацию $\{-1, -1, 1, 1, 1, -1, -1\}$ 30 раз, $\{1, -1, 1, 1, 1, -1, -1\}$ – 34 раза, $\{-1, 1, 1, 1, 1, -1, -1\}$ – 36 раз.

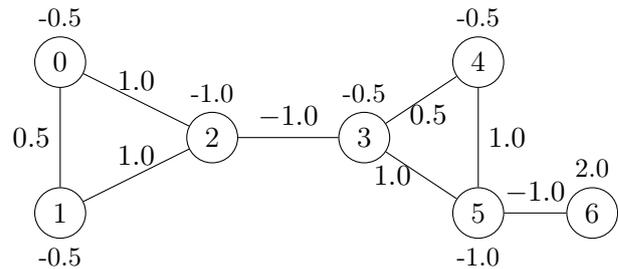


Рис. 3. Граф логической цепи для SAT-задачи на основе формулы F_2
Fig. 3. Logical circuit graph for SAT-problem based on formula F_2

Таблица 2. Таблица истинности логической цепи, записанной в виде системы (3)
Table 2. Truth table of a logical chain written as a system (3)

z_0	z_1	z_2	z_3	z_4	z_5
1	1	-1	-1	1	1
-1	1	1	1	1	-1
-1	-1	1	1	1	-1
-1	-1	1	1	-1	1
1	-1	1	1	1	-1
1	-1	1	1	-1	1
-1	1	1	1	-1	1
1	1	-1	-1	-1	1

ОСОБЕННОСТИ КОДИРОВАНИЯ ЦЕПИ ИЗ КВАНТОВЫХ ВЕНТИЛЕЙ

В отличие от классических логических цепей квантовые цепи состоят из квантовых вентилях, оперирующих на кубитах. Помимо базисных значений, традиционно обозначаемых $|0\rangle$ и $|1\rangle$, одиночные кубиты могут находиться в смешанном состоянии $\alpha|0\rangle + \beta|1\rangle$, где $\alpha, \beta \in \mathbb{C}$ – комплексные коэффициенты, такие что $\alpha^2 + \beta^2 = 1$ (т. е. множество возможных значений кубита непрерывно). Кроме того,

несколько кубитов могут находиться в так называемом запутанном состоянии, т. е. состоянии, при котором значения кубитов при измерении являются зависимыми. Для таких цепей непосредственное применение метода кодирования, предложенного в [18], представляет ряд сложностей. В то же время такое кодирование позволило бы обеспечить решение задачи построения квантовой цепи минимальной длины, реализующей заданную функцию, например, с помощью метода, предложенного в работе [10]. Однако решение такой задачи потребует создания бинарного представления для всех возможных промежуточных значений кубитов квантовой цепи, что является открытой проблемой [3].

В данном разделе рассмотрим применение метода, представленного в разделе «Кодирование логических цепей с помощью квантового отжига», к кодированию простой квантовой цепи с бинарным входом и квантовыми вентилями, выходные значения которых также являются бинарными при бинарных входных значениях. В качестве таких вентилях выберем вентиль отрицания NOT и вентиль контролируемого отрицания CNOT. Заметим, что вентиль CNOT широко применяется в квантовых цепях и в так называемой обратимой логике. Вентиль имеет два входа и два выхода, при этом значение первого входа копируется на первый выход, а значение второго входа инвертируется (т. е. вместо $|0\rangle$ на выход поступает $|1\rangle$ и наоборот) в случае, если первый вход имеет значение истина, т. е. первый кубит находится в состоянии $|1\rangle$, в противном случае (если первый кубит равен $|0\rangle$) второй выход копирует второй вход. Таблица истинности вентиля CNOT представлена в таблице 3.

Таблица 3. Таблица истинности вентиля CNOT с входными битами z_0, z_1 , выходными битами z_2, z_3 и дополнительным битом $z_4 = z_0 \wedge z_1 \wedge z_2$

Table 3. Truth table of a CNOT gate with input bits z_0, z_1 , output bits z_2, z_3 and an extra bit $z_4 = z_0 \wedge z_1 \wedge z_2$

z_0	z_1	z_2	z_3	z_4
1	1	1	1	1
1	-1	1	-1	1
-1	1	-1	-1	1
-1	-1	-1	1	-1

В следующем примере выполним кодирование квантовой цепи с двумя кубитами, состоящей из последовательного применения вентиля CNOT с контрольным первым кубитом, а затем – применения вентиля отрицания NOT к первому кубиту. Поскольку рассматриваемая

цепь при бинарном входе имеет бинарный выход, удобно представить ее значения в виде таблицы истинности, представленной в таблице 4. Заметим, что нумерация входных и выходных кубитов в таблице 4 выполнена в соответствии с Примером 5.

Таблица 4. Таблица истинности цепи, состоящей из последовательного применения вентиля CNOT с контрольным первым кубитом, а затем применения вентиля отрицания NOT к первому кубиту

z_0	z_1	z_6	z_3
1	1	-1	1
1	-1	-1	-1
-1	1	1	-1
-1	-1	1	1

Пример 5. Выполним кодирование вентиля CNOT с двумя входами. Для значений кубитов примем следующие соглашения. Обозначим $z_0 = 1$, если первый входной кубит находится в состоянии $|0\rangle$, и $z_0 = -1$, если иначе. Аналогично закодируем значение второго входного и выходных кубитов. Заметим, что соответствующая задача ЛП (2) не имеет решения для переменных z_0, \dots, z_3 и требует введения дополнительной переменной $z_4 = z_0 \wedge z_1 \wedge z_2$, полученной путем перемножения трех первых битов (о данном методе подробнее см. [4]). Решение (2) имеет следующий вид:

$$h = (0.75, 0.5, 0.75, -0.5, -2.0), \quad g = 1, \quad l = -4,$$

$$J = \begin{bmatrix} 0.25 & -1 & 0.5 & -1 \\ & 0.25 & -0.5 & -1 \\ & & -1 & -1 \\ & & & 1 \end{bmatrix}.$$

(Для матрицы J указаны только значения верхнетреугольной подматрицы вследствие симметричности.) Экспериментальная проверка на 100 запусках КО показала, что значение $\{-1, -1, -1, 1, -1\}$ было получено 21 раз, $\{-1, 1, -1, -1, 1\}$ – 32 раза, $\{1, -1, 1, -1, 1\}$ – 19 раз, $\{1, 1, 1, 1, 1\}$ – 28 раз, т. е. кодирование вентиля выполнено верно.

Аналогично выполним кодирование операции NOT с входным кубитом z_5 и выходным z_6 . Решая задачу ЛП (2), получаем, что $h_5 = h_6 = 0, J_{56} = 1, g = 2, l = -1$.

Наконец, для кодирования цепи, состоящей из последовательных вентилях CNOT с контрольным первым кубитом и NOT, примененным к первому кубиту, приравняем первый

выходной кубит CNOT, z_2 , и первый входной кубит, z_5 , для чего, в соответствии с этапом 3 метода, представленного в разделе «Кодирование логических цепей с помощью квантового отжига», положим $J_{25} = -1$. Выходными кубитами квантовой цепи при этом будут кубиты z_3 и z_6 .

Используем полученные коэффициенты h, J для запуска КО. В результате 100 запусков КО значение $\{1, -1, 1, -1, 1, 1, -1\}$ встречается 35 раз, $\{-1, -1, -1, 1, -1, -1, 1\} - 27$ раз, $\{1, 1, 1, 1, 1, 1, -1\} - 21$ раз, $\{-1, 1, -1, -1, 1, -1, 1\} - 17$ раз. Как можно убедиться, указанные значения соответствуют таблице истинности квантовой цепи, представленной в таблице 4. Таким образом, выполнено кодирование искомой квантовой цепи с бинарными входными и бинарным выходными значениями.

ЗАКЛЮЧЕНИЕ

В данной работе рассмотрены подходы к кодированию функциональности логических цепей и SAT-задач на их основе в виде соответствующих коэффициентов в модели Изинга и дальнейшего поиска решения с помощью КО. Экспериментально показано, что указанный подход может быть адаптирован для квантовых цепей с бинарным входным и бинарным выходным значением в случае, если все промежуточные состояния цепи также являются бинарными. Для обобщения данного подхода на универсальные квантовые цепи необходимо обеспечить конечность возможных промежуточных состояний цепи, однако такая возможность требует дальнейших исследований.

ЛИТЕРАТУРА

1. *Заикин О. С., Семенов А. А., Посыпкин М. А.* Процедуры построения декомпозиционных множеств для распределенного решения SAT-задач в проекте добровольных вычислений SAT@HOME // Управление большими системами. 2013. Вып. 43. С. 138–156.
2. *Заплетина М. А., Жуков Д. В., Гаврилов С. В.* Методы анализа выполнимости булевых формул для современных задач систем автоматизации проектирования в микроэлектронике // Изв. вузов. Электроника. 2020. Т. 25, № 6. С. 525–538. doi: 10.24151/1561-5405-2020-25-6-525-538
3. *Berent L., Burgholzer L., Wille R.* Towards a SAT encoding for quantum circuits: A journey from classical circuits to Clifford circuits and beyond // Schloss Dagstuhl - Leibniz-Zentrum für Informatik. 2022. Vol. 236. P. 1–17. doi: 10.4230/LIPICS.SAT.2022.18
4. *Bian Z., Chudak F., Macready W., Aidan R., Sebastiani R., Varotti S.* Solving SAT and MaxSAT with a quantum annealer: Foundations, encodings, and preliminary results // Inf. Comput. 2018. Vol. 275. Art. 104609. doi: 10.1016/j.ic.2020.104609
5. *Cook S. A.* The complexity of theorem-proving procedures // STOC'71: Proceedings of the third annual ACM symposium on Theory of computing. Association for Computing Machinery, New York, USA. 1971. P. 151–158. doi: 10.1145/800157.805047
6. D-Wave upgrades quantum chip to 2,000 qubits, gets first customer for its \$15 million machine // ZME Science [Электронный ресурс]. URL: <https://www.zmescience.com/science/news-science/d-wave-quantum-chip/> (дата обращения: 14.05.2023).
7. *de Falco D., Apolloni B., Cesa-Bianchi N.* A numerical implementation of quantum annealing // Conference: Stochastic Processes, Physics and Geometry. July 1988.
8. Formal logic [Электронный ресурс]. URL: https://en.wikibooks.org/wiki/Formal_Logic (дата обращения: 14.05.2023).
9. *Goldberg E. I., Prasad M. R., Brayton R. K.* Using SAT for combinational equivalence checking // Proceedings Design, Automation and Test in Europe. Conference and Exhibition 2001, Munich, Germany. 2001. P. 114–121. doi: 10.1109/DATE.2001.915010
10. *Große D., Chen X., Dueck G., Drechsler R.* Exact sat-based toffoli network synthesis // Proceedings of the 17th ACM Great Lakes Symposium on VLSI 2007, Stresa, Lago Maggiore, Italy. Association for Computing Machinery, New York, USA. 2007. P. 96–101. doi: 10.1145/1228784.1228812
11. IBM Unveils 400 Qubit-Plus Quantum Processor and Next-Generation IBM Quantum System Two // IBM Newsroom [Электронный ресурс]. URL: <https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two> (дата обращения: 14.05.2023).
12. *McGeoch C. C.* Adiabatic quantum computation and quantum annealing: Theory and practice // Synthesis Lectures on Quantum Computing. 2014. Vol. 5, no. 2. P. 1–93. doi: 10.1007/978-3-031-02518-1
13. *Meuli G., Soeken M., De Micheli G.* SAT-based {CNOT, T} Quantum circuit synthesis // Lecture Notes in Computer Science / Eds. J. Kari, I. Ulidowski. Vol. 11106. Cham: Springer Int. Publ., 2018. P. 175–188. doi: 10.1007/978-3-319-99498-7_12
14. *Nabeshima H., Inoue K.* Reproducible efficient parallel SAT solving // SAT 2020: Theory and Applications of Satisfiability Testing. 2020. Vol. 12178. P. 123–138. doi: 10.1007/978-3-030-51825-7_10

15. *Nielsen M., Chuang I.* Quantum computation and quantum information. 10th Anniversary Edition. New York: Cambridge Univ. Press, 2010. 704 p.
16. *Schneider S., Burgholzer L., Wille R.* A SAT Encoding for optimal Clifford circuit synthesis // ASPDAC'23: Proceedings of the 28th Asia and South Pacific Design Automation Conference. Association for Computing Machinery, New York, USA. 2023. P. 190–195. doi: 10.1145/3566097.3567929
17. *Shor P. W.* Algorithms for quantum computation: Discrete logarithms and factoring // Proceedings 35th Annual Symposium on Foundations of Computer Science, Santa Fe, USA. 1994. P. 124–134. doi: 10.1109/SFCS.1994.365700
18. *Su J.* Towards quantum computing: Solving satisfiability problem by quantum annealing // UCLA [Электронный ресурс]. 2018. URL: <https://escholarship.org/uc/item/8qp5200s> (дата обращения: 14.05.2023).
19. *Su J., Tu T., He L.* A quantum annealing approach for boolean satisfiability problem // DAC'16: Proceedings of the 53rd Annual Design Automation Conference. New York, 2016. Art. 148. doi: 10.1145/2897937.2897973
20. *Venegas-Andraca S. E., Cruz-Santos W., McGeoch C., Lanzagorta M.* A cross-disciplinary introduction to quantum annealing-based algorithms // Contemp. Phys. 2018. Vol. 59, no. 2. P. 174–197. doi: 10.1080/00107514.2018.1450720
21. *Vizel Y., Weissenbacher G., Malik S.* Boolean satisfiability solvers and their applications in model checking // Proceedings of the IEEE. 2015. Vol. 103, no. 11. P. 2021–2035. doi: 10.1109/JPROC.2015.2455034
22. *Wille R., Przigoda N., Drechsler R.* A compact and efficient SAT encoding for quantum circuits // 2013 Africon Pointe-Aux-Piments, Mauritius: IEEE. 2013. P. 1–6. doi: 10.1109/AFRCON.2013.6757630
23. *Yarkoni S., Raponi E., Bäck T., Schmitt S.* Quantum annealing for industry applications: Introduction and review // Reports on Progress in Physics. 2022. Vol. 85, no. 10. Art. 104001. doi: 10.1088/1361-6633/ac8c54
24. *Zhang S., Malik S., McGeer R.* Verification of computer switching networks: An overview // Automated Technology for Verification and Analysis - 10th International Symposium, ATVA 2012, Proceedings. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2012. Vol. 7561. P. 1–16. doi: 10.1007/978-3-642-33386-6_1

REFERENCES

1. *Zaikin O. S., Semenov A. A., Posypkin M. A.* Constructing decomposition sets for distributed solution of SAT problems in volunteer computing

project SAT@HOME. *Upravlenie bolshimi sistemami = Large-Scale Systems Control.* 2013;43:138–156. (In Russ.)

2. *Zapletina M. A., Zhukov D. V., Gavrilov S. V.* Boolean satisfiability methods for modern computer-aided design problems in microelectronics. *Izvestia vuzov. Elektronika = Proceedings of Universities. Electronics.* 2020;25(6):525–538. doi: 10.24151/1561-5405-2020-25-6-525-538 (In Russ.)

3. *Berent L., Burgholzer L., Wille R.* Towards a SAT encoding for quantum circuits: A journey from classical circuits to Clifford circuits and beyond. *Schloss Dagstuhl - Leibniz-Zentrum für Informatik.* 2022;236:1–17. doi: 10.4230/LIPICS.SAT.2022.18

4. *Bian Z., Chudak F., Macready W., Aidan R., Sebastiani R., Varotti S.* Solving SAT and MaxSAT with a quantum annealer: Foundations, encodings, and preliminary results. *Inf. Comput.* 2018;275:104609. doi: 10.1016/j.ic.2020.104609

5. *Cook S. A.* The complexity of theorem-proving procedures. *STOC'71: Proceedings of the third annual ACM symposium on Theory of computing.* Association for Computing Machinery, New York, USA. 1971. P. 151–158. doi: 10.1145/800157.805047

6. D-Wave upgrades quantum chip to 2,000 qubits, gets first customer for its \$15 million machine // ZME Science. URL: <https://www.zmescience.com/science/news-science/d-wave-quantum-chip/> (accessed: 14.05.2023).

7. *de Falco D., Apolloni B., Cesa-Bianchi N.* A numerical implementation of quantum annealing. *Conference: Stochastic Processes, Physics and Geometry.* July 1988.

8. Formal logic. URL: https://en.wikibooks.org/wiki/Formal_Logic (accessed: 14.05.2023).

9. *Goldberg E. I., Prasad M. R., Brayton R. K.* Using SAT for combinational equivalence checking. *Proceedings Design, Automation and Test in Europe. Conference and Exhibition 2001, Munich, Germany.* 2001;114–121. doi: 10.1109/DATE.2001.915010

10. *Große D., Chen X., Dueck G., Drechsler R.* Exact sat-based toffoli network synthesis // Proceedings of the 17th ACM Great Lakes Symposium on VLSI 2007, Stresa, Lago Maggiore, Italy. Association for Computing Machinery, New York, USA. 2007. P. 96–101. doi: 10.1145/1228784.1228812

11. IBM Unveils 400 Qubit-Plus Quantum Processor and Next-Generation IBM Quantum System Two // IBM Newsroom. URL: <https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two> (accessed: 14.05.2023).

12. *McGeoch C. C.* Adiabatic quantum computation and quantum annealing: Theory and practice. *Synthesis Lectures on Quantum Computing.* 2014;5(2):1–93. doi: 10.1007/978-3-031-02518-1

13. Meuli G., Soeken M., De Micheli G. SAT-based CNOT, T Quantum circuit synthesis. *Kari J., Ulidowski I. (eds.). Lecture Notes in Computer Science*. Vol. 11106. Cham: Springer International Publishing, 2018. P. 175–188. doi: 10.1007/978-3-319-99498-7_12
14. Nabeshima H., Inoue K. Reproducible efficient parallel SAT solving. *SAT 2020: Theory and Applications of Satisfiability Testing*. 2020;12178: 123–138. doi: 10.1007/978-3-030-51825-7_10
15. Nielsen M., Chuang I. Quantum computation and quantum information. 10th Anniversary Edition. New York: Cambridge Univ. Press; 2010. 704 p.
16. Schneider S., Burgholzer L., Wille R. A SAT Encoding for optimal Clifford circuit synthesis. *ASPAC'23: Proceedings of the 28th Asia and South Pacific Design Automation Conference*. Association for Computing Machinery, New York, USA. 2023. P. 190–195. doi: 10.1145/3566097.3567929
17. Shor P. W. Algorithms for quantum computation: discrete logarithms and factoring // *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, USA. 1994. P. 124–134. doi: 10.1109/SFCS.1994.365700
18. Su J. Towards quantum computing: Solving satisfiability problem by quantum annealing. *UCLA*. 2018. URL: <https://escholarship.org/uc/item/8qp5200s> (accessed: 14.05.2023).
19. Su J., Tu T., He L. A quantum annealing approach for boolean satisfiability problem. *DAC'16: Proceedings of the 53rd Annual Design Automation Conference*. New York; 2016. Art. 148. doi: 10.1145/2897937.2897973
20. Venegas-Andraca S. E., Cruz-Santos W., McGeoch C., Lanzagorta M. A cross-disciplinary introduction to quantum annealing-based algorithms. *Contemp. Phys.* 2018;59(2):174–197. doi: 10.1080/00107514.2018.1450720
21. Vizek Y., Weissenbacher G., Malik S. Boolean satisfiability solvers and their applications in model checking. *Proceedings of the IEEE*. 2015;103(11): 2021–2035. doi: 10.1109/JPROC.2015.2455034
22. Wille R., Przigoda N., Drechsler R. A compact and efficient SAT encoding for quantum circuits. *2013 Africon Pointe-Aux-Piments, Mauritius: IEEE*. 2013. P. 1–6. doi: 10.1109/AFRCON.2013.6757630
23. Yarkoni S., Raponi E., Bäck T., Schmitt S. Quantum Annealing for Industry Applications: Introduction and Review // *Reports on Progress in Physics*. 2022;85(10):104001. doi: 10.1088/1361-6633/ac8c54
24. Zhang S., Malik S., McGeer R. Verification of computer switching networks: An overview. *Automated Technology for Verification and Analysis - 10th International Symposium, ATVA 2012, Proceedings. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2012;7561:1–16. doi: 10.1007/978-3-642-33386-6_1

Поступила в редакцию / received: 15.05.2023; принята к публикации / accepted: 14.06.2023.

Авторы заявляют об отсутствии конфликта интересов / The authors declare no conflict of interest.

СВЕДЕНИЯ ОБ АВТОРАХ:

Мальцева Мария Алексеевна
младший научный сотрудник

e-mail: masha.mariam.maltseva@mail.ru

Румянцев Александр Сергеевич
д-р физ.-мат. наук, старший научный сотрудник

e-mail: ar0@krc.karelia.ru

CONTRIBUTORS:

Maltseva, Mariya
Junior Researcher

Rumyantsev, Alexander
Dr. Sci. (Phys.-Math.), Senior Researcher