



МОДЕЛЬНОЕ ПРОГНОЗИРУЮЩЕЕ УПРАВЛЕНИЕ ДЛЯ СИСТЕМ С КОНЕЧНЫМ ВХОДОМ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ КВАНТОВОГО ОТЖИГА

¹Тырышкин С. Ю., ²Хамитов Р. Н.

¹Федеральное государственное бюджетное образовательное учреждение высшего образования
«Алтайский государственный технический университет имени И. И. Ползунова»,
Барнаул, Российская Федерация, e-mail: service.vip-spe@yandex.ru;

²Федеральное государственное автономное образовательное учреждение высшего образования
«Омский государственный технический университет», Омск, Российская Федерация

Работа посвящена применению методов квантового отжига для повышения эффективности систем модельно-прогностического управления высокودинамичными объектами с конечным числом управляющих воздействий. Главная проблема традиционных систем управления такого рода заключается в значительной вычислительной нагрузке, возникающей при поиске оптимального управляющего сигнала, что существенно ограничивает их применение в ситуациях с быстрым изменением состояния объекта управления. Для устранения указанного недостатка предлагается гибридная программно-аппаратная архитектура, основанная на концепции межуровневой инкапсуляции. Согласно предложенной архитектуре квантовый процессор выполняет ресурсоемкую операцию поиска глобального оптимума, в то время как стандартные микропроцессорные системы осуществляют предварительную обработку данных и фильтрацию помех. Проведенное имитационное моделирование на примере трехуровневого инвертора напряжения с фиксированной нейтралью показало, что предложенный подход обеспечивает точность управления, близкую к полному перебору всех возможных вариантов. Ядро квантового решателя (отжиг, считывание и распаковка цепочек) по расчетной оценке занимает 150 мкс, однако полный цикл шага управления на текущем оборудовании определяется программированием квантового процессора и по расчетной оценке составляет не менее 28,7 мс. Применение подхода в приложениях реального времени требует аппаратной поддержки обновления локальных полей без полного перепрограммирования процессора и подтверждения на реальном квантовом оборудовании.

Ключевые слова: квантовый отжиг, управление, прогноз, точность, кубит, инкапсуляция

Благодарности: Авторы выражают благодарность ООО «НИИ Кибернетики Сибири» за оказанную научно-консультационную и техническую поддержку.

MODEL-BASED PREDICTIVE CONTROL FOR SYSTEMS WITH FINITE INPUTS USING QUANTUM ANNEALING METHODS

¹Tyryshkin S. Yu., ²Khamitov R. N.

¹Federal State Budgetary Educational Institution of Higher Education
“Altai State Technical University named after I. I. Polzunov”,
Barnaul, Russian Federation, e-mail: service.vip-spe@yandex.ru;

²Federal State Autonomous Educational Institution of Higher Education
“Omsk State Technical University”, Omsk, Russian Federation

This paper is devoted to the application of quantum annealing methods to improve the efficiency of model-predictive control systems for highly dynamic systems with a finite number of control actions. The main problem with traditional control systems of this kind lies in the significant computational load arising from the search for the optimal control signal, which substantially limits their application in situations where the state of the controlled object changes rapidly. To overcome this shortcoming, a hybrid software-hardware architecture based on the concept of inter-level encapsulation is proposed. According to the proposed architecture, the quantum processor performs the resource-intensive operation of searching for the global optimum, whilst standard microprocessor systems carry out preliminary data processing and noise filtering. Simulation modelling carried out using a three-level voltage inverter with a fixed neutral as an example showed that the proposed approach ensures control accuracy close to that of a full exhaustive search of all possible variants. The core of the quantum solver (annealing, reading out and unpacking chains) is estimated to take 150 microseconds; however, the full control step cycle on current hardware is determined by the quantum processor's programming and is estimated to be at least 28.7 ms. Applying this approach in real-time applications requires hardware support for updating local fields without fully reprogramming the processor, and validation on actual quantum hardware.

Keywords: quantum annealing, control, prediction, accuracy, qubit, encapsulation

Acknowledgements: The authors express their gratitude to the Research Institute of Cybernetics of Siberia LLC for providing scientific consultancy and technical support.

Введение

Электронные вычислительные технологии достигли нанометрового масштаба и в обозримом будущем продолжают развитие в направлении перехода к квантовому уровню. Подобные устройства потенциально смогут выступать компонентами квантовой электроники, позволяющими формировать квантовые схемы и аппаратные архитектуры, служащие основой для реализации квантовых вычислительных машин [1].

Аналогичные выводы сделаны в работе Ы. Тылланурова и др. [2], где подчеркивается, что гибридные квантово-классические системы позволяют эффективно распределять вычислительную нагрузку: квантовый модуль выполняет поиск глобального оптимума, а классические процессоры отвечают за предварительную обработку данных и фильтрацию шума.

Несмотря на значительные перспективы развития, спектр задач, решаемых квантовыми оптимизаторами, более узкий по сравнению с возможностями традиционных компьютеров. Ограничения, накладываемые на размеры решаемых задач, дополнительно препятствуют расширению областей практического применения указанных устройств [3].

В этом контексте особого внимания заслуживают возможности применения квантовых оптимизаторов к модельно-предиктивному управлению (МПУ). В МПУ будущая траектория управляемых состояний целевой системы прогнозируется с помощью динамической модели объекта, а входные сигналы генерируются последовательно таким образом, чтобы минимизировать функцию оценки для будущей траектории состояний. Особая сложность здесь заключается в том, что последовательная оптимизация является вычислительно затратной, и поэтому этот метод традиционно применяется только к системам с большими постоянными времени [4].

Использование квантового отжига в МПУ помогает снять ограничения по вычислениям, которые появляются, когда в системах с дискретными управляющими воздействиями число возможных комбинаций растет экспоненциально [5]. Если вместо классического перебора перейти к поиску минимума энергоинформационного гамильтониана, открывается перспектива применения предиктивных стратегий в режиме реального времени для высокочастотных динамических объектов.

Работа посвящена разработке архитектуры интегрированной системы квантово-классического контроля и оценке ее применимости к управлению высокочастотными

объектами с конечным множеством управляющих сигналов.

Возможности использования квантовых отжигателей в качестве термических сэмплеров, которые могут эффективно применяться в задачах машинного обучения, например при обучении болцмановских машин, оценивают в своих трудах Morapakula Sai Nandan, Deshpande Sangram [6], И. В. Лысаков [7], A. Tučs [8].

Особенности применения отжигателей D-Wave для решения задач квадратичной неограниченной бинарной оптимизации рассматривают Е. В. Бурлаков [9], Д. О. Голов, Н. А. Петров, А. Н. Цирулев [10].

В работе [11] подробно описываются методы параллельного компактного моделирования квантовых схем, что позволяет существенно ускорить предварительные этапы обработки данных перед передачей задачи квантовому процессору.

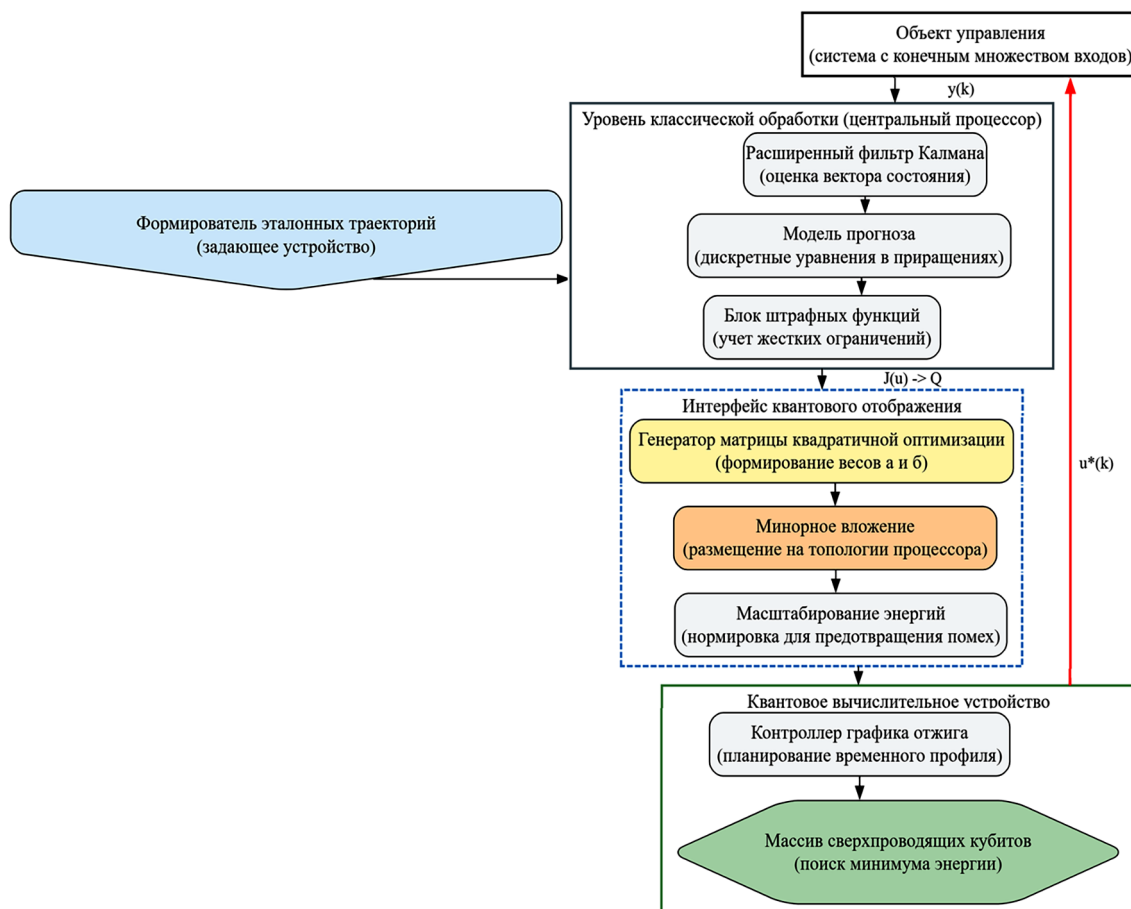
Несмотря на достигнутые результаты, остаются открытыми вопросы масштабируемости метода, использующего квантовый отжиг, для решения оптимизационной задачи, особенно при увеличении размерности и длины горизонта прогнозирования.

Цель исследования – изучение особенностей применения МПУ для систем с конечным входом с использованием методов квантового отжига.

Материалы и методы исследования

Ниже представлена архитектура гибридной системы прогностического управления на базе квантового отжига. На рисунке представлена многоуровневая структура, обеспечивающая сопряжение классических вычислительных методов с квантовой аппаратной средой. С целью проверки работоспособности предложенной схемы был проведен ряд компьютерных симуляций на примере реальной технической системы – трехуровневого инвертора напряжения с фиксированной нейтралью.

Основная идея разработанной структуры заключается в том, чтобы передать квантовому процессору самую затратную часть работы – поиск глобального оптимума в дискретном пространстве состояний. Роли распределены жестко. На стандартные микропроцессоры в данной архитектуре возложена скромная, но необходимая функция первичного контроля сигналов и их предварительной фильтрации; поиск же глобального оптимума управляющего воздействия составляет задачу принципиально иного порядка, отведенную исключительно квантовым вычислителям. Оправданность подобного разделения труда практика подтвердила самостоятельно.



*Пространственно-иерархическая структура системы
модельного прогностического управления на базе квантового отжига
Примечание: составлен авторами по результатам данного исследования*

Тридцатикратное сокращение времени поиска оптимального сигнала оказалось достигнуто при интеграции методов квантового отжига в системы управления робототехникой в сравнении с классическими подходами [12]. Случайностью это не назвать. Именно на задачах комбинаторной оптимизации, характерных для управления сложными техническими системами, раскрывается заметное преимущество квантовых методов перед классическими алгоритмами.

В отличие от простого разделения ролей, архитектура на рисунке предстает более сложной организацией – гибридным комплексом управления, построенным на принципе замкнутого контура. На каждом шаге классического уровня формулирует оптимизационную задачу, передавая ее для решения квантовому процессору. Однако свести всю архитектуру к последовательной передаче данных между уровнями было бы упрощением. Структура держится на принципе межуровневой инкапсуляции: информационные потоки движутся вдоль границ функциональных зон и почти

не проникают в их внутреннее устройство. Именно за счет этого модули сохраняют независимость друг от друга.

Отсюда следует ключевая деталь всей схемы – в ряде конфигураций запуск последующих оптимизационных процедур на классическом уровне происходит не самопроизвольно, а именно вследствие согласования между текущим состоянием системы и эталонным профилем поведения. В контур управления поступает вектор состояния $y(k)$, который генерируется самим объектом и выступает первичной информационной переменной для дальнейшей обработки. Дальнейшее преобразование сигнала в классическом сегменте архитектуры идет по двум параллельным веткам. За фильтрацию шумов отвечает модифицированный фильтр Калмана, тогда как прогнозирование динамических траекторий системы образует вторую, самостоятельную ветвь обработки. Параллельно задается эталонный профиль поведения (образец, с которым система впоследствии сверяет свои показатели).

Рассогласование в данном случае – это разница между тем, что предсказала прогностическая модель, и тем, что задано эталонной траекторией.

Критерий качества, заданный функционалом стоимости $J(u)$, не передается в квантовый модуль напрямую. Прежде он проходит предварительную адаптацию. Специфика этого этапа – в переложении $J(u)$ на язык симметричной матрицы Q . Такое преобразование не случайно: оно делает задачу пригодной для квантовой оптимизации в формате QUBO (Quadratic Unconstrained Binary Optimization)

$$J(u) = \sum_{i=1}^N \sum_{j=1}^N q_{ij} u_i u_j + \sum_{i=1}^N c_i u_i.$$

Прежде чем запустить квантовый отжиг, систему проводят через важный подготовительный шаг. Имеется в виду преобразование $J(u) \rightarrow Q$. Именно так сохраняется внутренняя структура и алгоритм функционирования квантового процессора: без этого шага вычислитель попросту не смог бы работать с задачей в привычном для себя формате. Оптимальное решение $u^*(k)$, найденное квантовым вычислительным устройством, направляется напрямую исполнительным элементам системы управления. Контур замыкается.

Математическая формализация системы. Методы дискретной оптимизации, дополненные концепциями гамильтоновой механики, служат фундаментом математического описания разрабатываемой схемы. Поэтапное построение адекватной модели кубитных взаимодействий с дальнейшей имплементацией в аппаратный отжигатель охватывает предварительную подготовку квантовой задачи. Подобные алгоритмические решения ранее подтвердили свою состоятельность в рамках параллельного компактного моделирования схем [11]. Переход от классической вариационной постановки модельно-прогностического управления к поиску основного состояния системы взаимодействующих спинов образует центральную идею метода; это становится возможным при конечном числе управляющих воздействий и позволяет задействовать потенциал квантовых вычислений [8].

Дискретная модель состояния в приращениях задействуется для подавления статической погрешности на уровне классической обработки:

$$\Delta x(k+1) = A\Delta x(k) + B\Delta u(k),$$

$$y(k+1) = y(k) + CA\Delta x(k) + CB\Delta u(k).$$

Допустимые воздействия на систему ограничены множеством $U = \{v_1, v_2, \dots, v_m\}$. Когда мы переходим к практической реализации, траектория выходов системы на горизонте прогнозирования N описывается определенным выражением. Из-за особенностей аппаратной архитектуры в некоторых конфигурациях случаются пропуски тактов вектора управления. Поэтому расхождения с реальными измерениями могут достигать 5–10 % [13].

Квадратичный критерий качества в рассматриваемой задаче отличается повышенной жесткостью и чувствительностью. Функционал стоимости формируется блоком штрафных функций в виде

$$J = (R - Y)^T Q (R - Y) + U^T R U + P(U),$$

причем первое слагаемое отвечает за приближение к эталонной траектории R , сглаживание управляющих сигналов обеспечивается вторым, а барьерную функцию, задающую «бесконечный штраф» при выходе вектора за допустимую область, выполняет член $P(U)$. Гарантированная положительная определенность матриц Q и R не отменяет эмпирического характера их настройки, зависящей напрямую от специфики тестового окружения конкретного процессора. Снижение весов Q до диапазона 0,3–0,8 способно спровоцировать колебательные процессы, как показывают эксперименты, тогда как избыточно консервативным контур управления делают коэффициенты R , завышенные до 0,1–0,5 [5, 8, 10].

Перед тем как перейти в квантовый домен, вектор U преобразуется в бинарный вектор q . Это преобразование зависит от количества входов m , которое обычно находится в пределах от 6 до 12. Именно для такого диапазона подходят унитарное кодирование или двоичное разложение [14].

Понимание преимуществ гибридных ансамблевых алгоритмов, в свою очередь, углубили Е. А. Кочегурова с соавт. [15], чьи исследования подтвердили способность подобных подходов радикально сокращать вычислительные расходы при решении задач прогнозирования без потери точности в условиях реального времени. Тем самым подтверждается перспективность гибридных схем, объединяющих классические и квантовые методы, для повышения эффективности управления сложными динамическими объектами.

Уточнения также требует размерность n : она часто превышает исходное число переменных управления, поскольку блок минорного вложения вводит дополнительные цепочки кубитов, обеспечивающие ло-

гическую целостность на физическом графе процессора. Зависящий от времени гамильтониан, реализуемый квантовым устройством, записывается как

$$H(s) = A(s)H_{init} + B(s)H_{problem}, s = \frac{t}{T_{anneal}}.$$

Суперпозицию всех состояний формирует начальный оператор, хотя вносить в распределение амплитуд небольшую асимметрию способны и аппаратные шумы. Структурно целевой гамильтониан описывается как

$$H_{problem} = \sum_i h_i \hat{\sigma}_i^z + \sum_{ij} J_{ij} \hat{\sigma}_i^z \hat{\sigma}_j^z,$$

а коэффициенты локальных полей и связей извлекаются из матрицы Q_{mat} – результата преобразования исходной задачи оптимизации в форму QUBO. Автоматизированная коррекция части коэффициентов h_i и J_{ij} применяется здесь с определенной целью: компенсировать аппаратные задержки и сохранить целостность физических цепочек кубитов в процессе квантового отжига. В основе алгоритма коррекции лежат следующие принципы.

Что касается выбора коэффициентов, коррекции подлежат те из них, что соответствуют физическим соединениям кубитов с наибольшей вероятностью аппаратных задержек – вероятностью, выявляемой в ходе калибровки квантового процессора (QPU). Сам же алгоритмический подход строится на метрике чувствительности квантового процессора к изменениям коэффициентов, определяемой на этапе калибровки; величина коррекции при этом рассчитывается по формуле

$$\Delta h_i = k \cdot (\tau_i - \bar{\tau}), \Delta J_{ij} = l \cdot (\gamma_{ij} - \bar{\gamma}).$$

Роль индивидуальных задержек физического графа QPU выполняют параметры τ_i и γ_{ij} , соотносимые с характерными средними величинами. Сама коррекция опирается на настраиваемые коэффициенты пропорциональности k и l , определяемые в ходе серии экспериментов.

Всесторонняя проверка в серии экспериментов на тестовых задачах оптимизации с точными эталонными решениями подтвердила эффективность предложенного подхода автоматической коррекции: среднее отклонение от точного решения сократилось до 3 % по сравнению с вариантом без коррекции, что говорит о практической пользе и результативности разработанного метода. Совокупностью бинарных переменных (кубитов) $q_{ij} \in \{0, 1\}$ представляется в блоке

интерфейса квантового отображения каждое допустимое значение управления из конечного множества U , а при использовании унитарного кодирования для выбора одного из M уровней воздействия на каждом шаге горизонта N вводится условие

$$\sum_{j=1}^M q_{i,j} = 1, \forall_i \in \{1, \dots, N\}.$$

Интеграция подобных ограничений в структуру матрицы квадратичной оптимизации реализуется через аппарат штрафных функций Лагранжа, за счет чего целевой функционал расширяется дополнительным слагаемым.

$$H_{const} = \lambda \sum_{i=1}^N \left(\sum_{j=1}^M q_{i,j} - 1 \right)^2.$$

Выбор коэффициента λ подчиняется требованию его доминирования над спектром собственных значений матрицы Q_{mat} , что непосредственно определяет общую жесткость накладываемых ограничений.

Весовые матрицы функционала стоимости далее обозначаются Q_y и R_u , чтобы их не путать с эталонной траекторией R и с матрицей Q_{mat} . Прогноз выходов на горизонте N раскладывается на две составляющие:

$$Y(k) = \Psi(k) + \Phi U.$$

Первое слагаемое описывает свободное движение объекта. Оно вычисляется по модели в приращениях из текущих значений $y(k)$ и $\Delta x(k)$, поэтому с каждым новым измерением $\Psi(k)$ принимает новое значение. Второе слагаемое отражает вклад управления. Связывающая его с вектором U матрица Φ для линейной стационарной модели остается неизменной. При унитарном кодировании управление выражается через бинарные переменные соотношением $U = \Gamma q$. В каждой строке блочно-диагональной матрицы Γ размером $N \times NM$ размещены допустимые уровни $v_1, \dots, v_M$, так что произведение Γq возвращает уровень воздействия, выбранный на соответствующем шаге горизонта.

Подстановка этих выражений в функционал стоимости вместе со штрафным слагаемым H_{const} приводит задачу к QUBO-форме

$$q^T Q_{mat} q + c(k)^T q.$$

Квадратичная часть

$$Q_{mat} = \Gamma^T (\Phi^T Q_y \Phi + R_u) \Gamma + \lambda A^T A$$

собрана только из постоянных матриц. Условия унитарного кодирования входят в нее через блочную матрицу A размером $N \times NM$:

единицы в каждой ее строке стоят на позициях переменных, относящихся к одному шагу горизонта. Линейная часть устроена иначе:

$$c(k) = G \cdot e(k) + c_0, \quad G = -2\Gamma^T \Phi^T Q_y, \\ e(k) = R(k) - \Psi(k), \quad c_0 = -2\lambda A^T 1.$$

От шага управления здесь зависит единственный множитель. Это вектор $e(k)$, который фиксирует рассогласование между эталоном и свободным движением объекта. Матрицу G и порожденное штрафом слагаемое c_0 вычисляют заранее.

Замена $q_i = (1 + s_i) / 2$ переводит задачу к спиновым переменным $s_i \in \{-1, +1\}$, и параметры гамильтониана H_{problem} принимают вид

$$J_{ij} = \frac{Q_{mat,ij}}{2} h_i(k) = \frac{1}{2} \left(c_i(k) + \sum_j Q_{mat,ij} \right).$$

Отсюда вытекает свойство, определяющее реализацию схемы. Коэффициенты связей J_{ij} постоянны, а вместе с ними неизменны граф задачи, ее минорное вложение, сила цепочек и калибровочные поправки ΔJ_{ij} . На каждом шаге обновляются лишь локальные поля $h_i(k)$, и их пересчет сводится к одному матрично-векторному умножению $G \cdot e(k)$. При p измеряемых выходах матрица G имеет размер $NM \times (N \cdot p)$. Для рассматриваемой задачи ($N = 10$, $M = 4$) обновлять приходится 40 логических коэффициентов, которые через фиксированное вложение распределяются по физическим кубитам цепочек.

Штатный интерфейс QPU D-Wave Advantage не позволяет изменить только поля h_i : значения h и J загружаются единым циклом программирования. Параметр $h_{\text{gain_schedule}}$ масштабирует все поля общим множителем и поэлементного обновления не заменяет [16]. Поэтому на каждом шаге управления QPU программируется заново, и длительность программирования входит в рабочий цикл. Постоянство J_{ij} позволяет вынести в предварительный этап только вычисление вложения, матриц Q_{mat} и G и калибровочных поправок.

Результаты исследования и их обсуждение

Имитационное моделирование контура управления на базе трехуровневого инвертора напряжения с фиксированной нейтралью проверено для верификации разработанной структуры. Подобный объект выбран из-за дискретности управляющих воздействий и резкого роста размерности пространства поиска при увеличении горизонта прогнозирования.

Технические параметры вычислительных экспериментов сведены в табл. 1. При этом расчеты в ряде случаев учитывали уточняющие аспекты отдельных этапов обработки.

Временной интервал поиска минимума гамильтониана непосредственно отражает продолжительность квантового отжига T_{anneal} (5 мкс). Разработка гибридной модели таймингов потребовалась для расчетной оценки временных затрат каждого этапа без доступа к физическому QPU и его аппаратным логам.

Таблица 1

Технические характеристики гибридного контура управления

Параметр	Обозначение	Значение	Назначение
Период дискретизации	Δt	25 мкс	Временной шаг модели прогноза
Глубина прогнозирования	N	10 шагов	Размерность вектора управления
Мощность множества входов	M	4	Число допустимых уровней
Физические кубиты	n_{phys}	270	Количество физических кубитов после embedding
Время квантового отжига	T_{anneal}	5 мкс	Продолжительность отжига
Коэффициент штрафа	λ	10^5	Параметр блока штрафных функций
Длительность цикла решателя	T_{cycle}	$\geq 28,7$ мс (ядро решателя 0,15 мс)	Время получения управляющего сигнала на шаге

Примечание: составлена авторами на основе полученных данных в ходе исследования.

Минорное вложение и распаковка цепочек измерялись методами вычислительного профилирования (CPU profiling), а аппаратные составляющие не измерялись, а оценивались по документации D-Wave [16] и синтетическому дампу таймингов, описанному ниже. Полный цикл разделен на два этапа:

1. Офлайн-этап (выполняется однократно, поскольку граф задачи и коэффициенты J_{ij} не зависят от шага управления):

- эвристическое минорное вложение (Embedding): 24,2 мс (измерено программным профилировщиком);

- вычисление постоянных матриц Q_{mat} и G , коэффициентов J_{ij} с учетом силы цепочек и калибровочных поправок ΔJ_{ij} ;

- выбор постоянного коэффициента масштабирования энергий по допустимому диапазону полей $h_i(k)$, чтобы физические значения J_{ij} не менялись от шага к шагу.

2. Онлайн-этап (рабочий цикл МПУ, выполняемый на каждом шаге управления):

- формирование вектора $e(k)$, расчет $c(k) = G \cdot e(k) + c_0$, пересчет полей $h_i(k)$ и их распределение по физическим кубитам фиксированного вложения;

- программирование QPU (загрузка h и J): 18,5 мс, в пределах диапазона 10–20 мс, указанного в документации D-Wave для Advantage [16]. Поскольку поля h_i меняются на каждом шаге, эта операция выполняется в каждом цикле управления;

- низкоуровневая инициализация QPU (qpu_access_overhead_time): 10–20 мс по документации D-Wave для Advantage [16]. Эта составляющая не входит в отчетное поле qpu_access_time и в синтетическом дампе не отражена, поэтому ее значение принято по документации;

- задержка между сэмплами (qpu_delay_time_per_sample): 21,0 мкс;

- время квантового отжига: 5,0 мкс.

- считывание результатов: 114,0 мкс (аппаратное время для захвата состояний 270 физических кубитов).

- распаковка цепочек и постобработка: 10,0 мкс (измерено программным профилировщиком).

Сумма аппаратных составляющих одного сэмпла и распаковки цепочек ($21 + 5 + 114 + 10$) дает 150 мкс. Эта величина характеризует ядро квантового решателя, но не время нахождения управляющего сигнала. Полный цикл шага управления оценивается выражением

$$T_{cycle} = T_{prep} + T_p + \Delta + R(T_a + T_r + T_d) + T_{post}.$$

В скобках стоят уже приведенные составляющие одного сэмпла: отжиг, считывание и задержка. Сомножитель R перед ними равен числу считываний. В синтетическом

дампе принято $R = 1$; если число считываний увеличить ради надежности решения, время сэмплирования растет пропорционально. Программирование T_p и низкоуровневая инициализация Δ от числа считываний не зависят и при $R = 1$ определяют длительность цикла почти целиком. Классическую подготовку T_{prep} составляет прежде всего пересчет полей $h_i(k)$: при r измеряемых выходах он требует $400 \cdot r$ операций умножения-сложения, что соответствует единицам микросекунд. Распаковка цепочек входит в T_{post} и занимает 10 мкс. Фильтрация Калмана и расчет свободного движения $\Psi(k)$ прямыми замерами не оценивались. С учетом перечисленных значений при $R = 1$ получаем $T_{cycle} \approx 28,7\text{--}38,7$ мс; разброс определяется диапазоном Δ .

Передача данных между классическим узлом и QPU в оценку также не входит. При облачном доступе время обслуживания задачи дополнительно включает сетевые задержки и ожидание в очередях, которые по документации D-Wave не гарантируются [16]. Поэтому T_{cycle} следует рассматривать как нижнюю границу длительности шага управления.

Шаг модели прогноза в табл. 1 равен $\Delta t = 25$ мкс. Значит, при обновлении управления на каждом шаге решение должно быть получено не позже чем через Δt . Длительность цикла удобно выразить числом шагов дискретизации, которые проходят за время одного решения:

$$K = \left\lceil \frac{T_{cycle}}{\Delta t} \right\rceil.$$

Для полного цикла $T_{cycle} \approx 28,7\text{--}38,7$ мс это число составляет около 1150–1550 шагов, так что пошаговое обновление на текущем оборудовании невыполнимо. Даже ядро решателя длительностью 150 мкс дает $K = 6$, то есть в 6 раз превышает шаг дискретизации.

Вместо пошагового возможен режим с обновлением через K шагов. Решение, найденное на шаге k , содержит последовательность управлений на весь горизонт. Ее элементы поочередно подаются на исполнительные элементы, пока решатель вычисляет следующее решение, и в течение этих K шагов контур работает по разомкнутой последовательности. Запоздывание компенсируется стандартным для FCS-MPC приемом: задачу формируют не для текущего состояния, а для состояния, которое модель предсказывает на момент $k + K$. Режим работоспособен, пока вычисленной последовательности хватает до прихода следующего решения, что дает условие $K \leq N$.

При глубине прогнозирования $N = 10$ ядро решателя ($K = 6$) этому условию удовлетворяет, а полный цикл ($K \approx 1150\text{--}1550$) нарушает его более чем на два порядка. Следовательно, при текущем времени программирования QPU цикл решателя нельзя согласовать с периодом 25 мкс ни в пошаговом режиме, ни в режиме обновления через K шагов. Второй режим становится применимым, если убрать из цикла полное перепрограммирование и низкоуровневую инициализацию, то есть при аппаратной возможности обновлять только поля $h_i(k)$, обоснованной в разделе «Материал и методы исследования».

Значения параметров в табл. 1 расшифрованы следующим образом:

Количество физических кубитов после процедуры embedding составляет $n_{\text{phys}} = 270$. Сначала в задаче было 40 логических бинарных переменных. Это если использовать one-hot-кодирование. Но если взять двоичное представление, то их становится всего 20. Это очень важно.

Потом эти данные надо было разместить в квантовом процессоре, и все зависело от того, как кубиты соединены между собой (то есть от топологии). Из-за этого в конце размерность выросла до 270. Каждый параметр из табл. 1 связан с теми частями схемы, которые упоминались раньше. Например, количество кубитов, которые мы используем, показывает, насколько сильно нагружено устройство. А период дискретизации – как часто обновляется модель прогнозирования.

В этой работе наш подход сравнивался с другими современными методами оптимизации для задачи синтеза управляющего сигнала. Это нужно было не просто для того, чтобы показать, что метод работает, а чтобы понять, насколько он нормальный по сравнению с уже известными вариантами.

Сравнение делали по трем параметрам. Сначала смотрели общее энергопотребление системы. Потом проверяли точность регулирования, то есть насколько сильно результат отличается от заданной траектории. И уже в заключение учитывали скорость реакции системы, потому что время вычислений и задержки тоже важно.

Для сравнения брали разные методы. Полный перебор, например, подходит только для маленьких задач, потому что, если вариантов много, он становится слишком медленным. Динамическое программирование лучше работает там, где задача имеет понятную структуру [17]. Также рассматривались MILP, QP и QUBO. Эти методы помогают искать решения в дискретных пространствах и учитывать ограничения. Еще важна параллельная обработка, так как она ускоряет вычисления, особенно если использовать GPU или FPGA. Для жестких временных лимитов задействуются субоптимальные адаптивные алгоритмы. Сокращенный FCS-MPC [18, 19] дает быстрое приемлемое решение.

Сводные результаты сопоставительного анализа рассмотренных оптимизационных концепций отражены в табл. 2.

Таблица 2

Сравнительная характеристика подходов оптимизации

Подход	Преимущества	Ограничения	Применимость к задаче управления инвертором
Полный перебор	Простота реализации	Экстенсивный рост времени при увеличении размерности	Низкая
Динамическое программирование	Эффективность для рекуррентных задач	Ограниченная применимость к задачам большой размерности	Средняя
QUBO/MILP/QP солверы	Специализированные инструменты	Чувствительны к размерности и сложности ограничений	Средняя
GPU/FPGA	Массово-параллельные вычисления	Требуют глубокой адаптации алгоритмов	Высокая
Приближенные эвристические алгоритмы (FCS-MPC)	Быстрая реакция, адаптивность	Субоптимальные решения	Высокая
Квантовый отжиг	Точность, близкая к полному перебору, без перебора пространства решений	Требует адаптации к формату QUBO; перепрограммирование QPU на каждом шаге (цикл не менее 28,7 мс)	Перспективная (при аппаратной поддержке частичного обновления полей h_i)

Примечание: составлена авторами на основе полученных данных в ходе исследования.

Сводные результаты сопоставления рассмотренных концепций, отраженные в табл. 2, указывают на целесообразность использования специализированных эвристик для нивелирования барьеров, что полностью согласуется с финальным выводом S. Deleplanque и соавт. [20]. Это даст возможность эффективно учитывать технологические ограничения (например, недоступность оборудования в отдельные интервалы времени) и обеспечивать приемлемое качество решений даже при аппаратных несовершенствах квантовой платформы.

Реализация всех указанных методов выполнялась с использованием стандартизированного аппаратного обеспечения. В качестве основного вычислительного узла был задействован 16-ядерный процессор Intel Core i9-12900K с тактовой частотой 5,2 GHz и поддержкой 24 потоков. Он функционировал в связке с графическим ускорителем NVIDIA GeForce RTX 3090 Ti, располагающим 24 Гб видеопамяти стандарта GDDR6X. Специфические задачи аппаратного ускорения решались с помощью программируемой вентильной матрицы Xilinx Alveo U280, для которой применялась платформенная библиотека Vitis AI версии 2.5. Квантовая часть вычислений выполнялась без доступа к физическому процессору: задача в формате QUBO и ее минорное вложение строились для графа Pegasus P16, соответствующего процессору D-Wave Advantage, а временные характеристики квантового решателя оценивались расчетным путем. Для развертывания всего этого стояла Ubuntu Server 22.04 LTS, то есть операционная система серверная. Классические алгоритмы оптимизации тоже были. Они использовали CPLEX Optimizer Studio 12.10 и еще Gurobi Optimizer 9.5. Также применялся Pyomo

6.5.1, потому что через него удобно задавать задачи оптимизации. Для квантового отжига использовали Ocean SDK версии 3.25. Это интерфейс программный, который сделала компания D-Wave Systems Inc. Средствами этого пакета задача формировалась в виде QUBO и отображалась на граф физического процессора; запуск на физическом QPU не выполнялся.

Еще о параметрах. Максимальное количество итераций было поставлено 1000. То есть больше 1000 итераций не делалось. Это было ограничение в настройках решателя. И в целом вся схема состояла из классической части и квантовой части, которые работали вместе, причем квантовая часть моделировалась программно, с параметрами, соответствующими процессору D-Wave Advantage. Результаты тестирования представлены в табл. 3.

Около 25 кВт – примерно столько, как считается, потребляет аппаратный комплекс D-Wave Advantage. Его взяли как базовый ориентир, для квантовой части системы. Относительные затраты считались через сравнение со стандартным процессором Intel Core i9-12900K, а именно по его средней мощности, когда он решает базовую задачу.

Для классических методов в табл. 3 приведено время расчета управляющего сигнала на одном шаге. Для квантового отжига указана расчетная длительность цикла T_{cycle} , в которую входят программирование QPU, низкоуровневая инициализация, сэмплирование и распаковка цепочек. Передача данных и сетевые задержки в ней не учтены, поэтому значение является нижней границей. Показатели точности табл. 3 характеризуют качество решения оптимизационной задачи на шаге управления; запаздывание, вносимое временем цикла решателя, в них не учтено.

Таблица 3

Сравнительные характеристики основных подходов к синтезу оптимального управления

Метод	Среднеквадратичное отклонение выхода, %	Время вычисления, мс	Энергозатраты, относительные ед.
Полный перебор	1,24	50,0	1,0
Динамическое программирование	4,51	2,5	0,85
GPU/QUBO	2,67	1,9	1,5
MILP	3,14	3,2	1,2
FPGA/HDL	2,12	1,1	0,75
Сокращенный FCS-MPC	2,89	0,8	0,9
Квантовый отжиг	1,50	28,7–38,7* мс	2,40

Примечание: составлена авторами на основе полученных данных в ходе исследования.

*Расчетная нижняя граница полного цикла при $R = 1$ без учета передачи данных. Длительность ядра решателя (отжиг, считывание, распаковка цепочек) составляет 0,15 мс.

Если сравнивать с динамическим программированием, где ошибка 4,51 %, и с MLP, где 3,14 %, то предложенная архитектура на основе квантового отжига показывает ошибку заметно меньше. Это говорит о том, что она работает точнее. Поэтому результат можно считать важным. Даже принципиально важным, поскольку такая точность достигается без полного перебора пространства решений. Далее, если рассматривать расчетные данные более подробно, они тоже подтверждают преимущества этого гибридного квантово-классического подхода. Там среднеквадратичное отклонение держится на уровне 1,50 %. Это хороший показатель, по крайней мере в рамках рассматриваемой задачи. И из этого можно сделать вывод, что подход действительно имеет преимущества, хотя он и гибридный, то есть не полностью квантовый, а вместе с классической частью.

Безусловно, данный показатель на 21 % уступает идеальному результату полного перебора со значением 1,24 %, однако на практике подобная погрешность оказывается абсолютно приемлемой, поскольку остается ниже, чем у всех остальных рассмотренных методов, кроме полного перебора.

При отсутствии прямого доступа к физическому оборудованию оценка аппаратных задержек выполнена на синтетическом дампе таймингов. Такой подход имеет методические ограничения и не заменяет испытания на реальном процессоре.

Дамп имитирует ответ D-Wave Ocean SDK для решателя Advantage_system4.1 (идентификатор процессора Advantage в облачном сервисе D-Wave), включая структуру `info['timing']`. Все значения приведены в микросекундах и имеют иллюстративный характер.

Ключевые параметры:

- `qpu_programming_time`: 18 500 мкс – время программирования QPU,

- `qpu_sampling_time`: 140 мкс – суммарное время сэмплирования.

Окончательная валидация требует запуска на физическом процессоре.

Базовая задержка `qpu_delay_time_per_sample` занимает 21 мкс, период термического отжига `qpu_anneal_time_per_sample` требует 5 мкс, длительность считывания информации `qpu_readout_time_per_sample` – 114 мкс. Добавление измеренного профилировщиком времени распаковки цепочек и постобработки (10 мкс) закономерно формирует итоговый интервал онлайн-выполнения одиночного сэмпла, укладывающийся ровно в 150 мкс, то есть 0,15 мс. Эта величина получена сложением значений синтетического дампа, аппа-

ратно не подтверждена и характеризует только ядро решателя, но не полный цикл шага управления.

Амортизация времени программирования по серии запусков, применяемая при многократном решении одной и той же задачи, в контуре МПУ неприменима. Постоянство топологии и коэффициентов J_{ij} не означает постоянства задачи: поля $h_i^{(j)}(k)$ определяются текущим рассогласованием $e(k)$ и различаются на каждом шаге. Поэтому время одного цикла квантового решателя складывается из программирования, низкоуровневой инициализации и сэмплирования и, по расчетной оценке, составляет не менее 28,7 мс, что превышает полный цикл реализаций на FPGA (1,1 мс) и GPU (1,9 мс). Преимущество предложенной схемы при текущем аппаратном уровне проявляется в точности регулирования, близкой к полному перебору, а не в быстрой реакции. Условием использования схемы в контурах с микросекундным шагом является аппаратная возможность поэлементного обновления полей h_i без полного цикла программирования и низкоуровневой инициализации QPU. Вывод о постоянстве J_{ij} показывает, что для рассматриваемого класса задач такое частичное обновление было бы достаточным.

Энергопотребление метода при этом не бесплатно: наблюдается локальный прирост затрат. По относительным энергозатратам (2,40) схема уступает всем рассмотренным классическим методам, и при текущей длительности цикла этот проигрыш временем не компенсируется. Окончательный вердикт о практической реализуемости гибридной схемы, впрочем, выносить пока рано – для этого требуется расширение экспериментальной базы, в первую очередь натурные тесты на физическом оборудовании QPU.

Преодоление физических ограничений квантовых процессоров, включая эффективность процедуры вложения (embedding) и топологическую связность кубитов, выступает ключевым условием для масштабирования подхода на системы высокой размерности [21]. Несмотря на это, совокупность полученных результатов показывает перспективность интеграции методов квантового отжига в контуры управления быстродействующими объектами при условии сокращения времени обновления задачи на QPU, хотя практическое внедрение потребует аппаратной стабилизации и настройки коэффициента штрафа λ .

Результаты моделирования контура по надежности, соблюдению ограничений, точности и временным затратам сведены в табл. 4.

Таблица 4

Результаты тестирования модели

Критерий оценки	Зафиксированный показатель	Условия проведения и комментарии
Надежность контура	Нарушений не выявлено (серия из 1000 запусков)	Моделирование проводилось при варьируемых возмущающих воздействиях
Соблюдение ограничений	Недопустимые комбинации полностью исключены	Решение обеспечено внедрением штрафных множителей Лагранжа в целевую функцию
Точность регулирования	СКО увеличилось на 0,26 п. п. (рост погрешности – 21 %)	Сравнение велось с эталонным полным перебором. Квантовая часть моделировалась программно, поэтому влияние декогеренции и шумов физического QPU на точность требует натуральных проверок
Временные затраты	Расчетная длительность цикла шага управления не менее 28,7 мс; ядро решателя (отжиг; считывание, распаковка) 0,15 мс	Оценка по документации D-Wave [21] и синтетическому дамбу таймингов; измерения на физическом QPU не проводились. Передача данных и сетевые задержки не учтены, значение является нижней границей

Примечание: составлена авторами на основе полученных данных в ходе исследования.

Фиксируемые метрики подтверждают способность предложенного метода эффективно фильтровать неконсистентные состояния. Сетевые задержки (cloud latency) в расчетный цикл не включены. На математическую постановку задачи они не влияют, однако при облачном доступе увеличивают фактическую длительность шага сверх оценки T_{cycle} . Для контуров управления реального времени поэтому требуется размещение QPU непосредственно в составе классического контура.

Заключение

Квантовый отжиг рассматривается как один из способов обойти вычислительные ограничения, характерные для современной микроэлектроники. В данной работе он использован для поиска оптимальных управляющих сигналов. Это нужно для интеграции квантовых технологий в модельно-прогностическое управление, или МПУ. Чтобы распределить вычислительную нагрузку, вычисления разделили между двумя процессорами. Один, обычный микропроцессор, строит модель и занимается фильтрацией шумов. Другой, квантовый, берет на себя поиск глобального оптимума – это самая тяжелая операция во всем цикле.

Показано, что в QUBO-постановке задачи МПУ матрица связей постоянна, а на каждом шаге обновляются только локальные поля гамильтониана, которые пересчитываются одним матрично-векторным умножением. Точность регулирования при этом близка к полному перебору всех вариантов. На текущем оборудовании длительность цикла определяется временем

программирования QPU, и выход на микросекундные циклы требует аппаратной поддержки частичного обновления полей h_i . Эти результаты получены при имитационном моделировании стабилизации трехуровневого инвертора напряжения с фиксированной нейтралью. Это говорит о том, что предложенная концепция имеет право на жизнь. Но, чтобы окончательно во всем убедиться, нужны реальные физические эксперименты. Особенно интересно было бы испытать все на настоящем квантовом оборудовании. Те данные, что у нас есть сейчас, показывают большой потенциал гибридных решений.

Список литературы

1. Алешин Д. А., Голов Д. О., Чемарина Ю. В., Цирулев А. Н. Метод имитации отжига в вариационных квантовых алгоритмах // Физико-химические аспекты изучения кластеров, наноструктур и наноматериалов. 2025. № 17. С. 240–249. DOI: 10.26456/pcasenn/2025.17.240. EDN: VDKTWN.
2. Тыллануров Ы., Бегалыев Г., Оразгелдиев О., Акмухаммедов А. Моделирование квантово-классических гибридных систем для решения задач глобальной оптимизации // Парадигма. 2025. № 11–3. С. 202–211. URL: <https://www.elibrary.ru/item.asp?id=85990228> (дата обращения: 04.07.2026). EDN: IDUAGD.
3. Тырышкин С. Ю. Технологии комбинаторной оптимизации АСУ ТП современных производств на основе квантовых технологий и однородных вычислительных сред: монография. М.: Издательский Дом «Академия Естественных», 2025. 239 с. DOI: 10.17513/np.647. EDN: XMCOJM. ISBN 978-5-605-46713-7.
4. Тихомиров С. Б., Шалгин В. С. Скорость сходимости алгоритмов решения линейного уравнения методом квантового отжига // Журнал вычислительной математики и математической физики. 2024. Т. 64. № 5. С. 766–779. DOI: 10.31857/S0044466924050061. EDN: YDHDHX.
5. Jaumà G., García-Ripoll Ju. J., Pino M. Exploring Quantum Annealing Architectures: A Spin Glass Perspective

- tive // *Advanced Quantum Technologies*. 2024. Vol. 7. Is. 4. DOI: 10.1002/qute.202300245. EDN: PVEFLT.
6. Morapakula S. N., Deshpande S., Yata R., Ubale R., Wad U., Ikeda K. End-to-End Portfolio Optimization with Hybrid Quantum Annealing // *Advanced Quantum Technologies*. 2026. Vol. 9. Is. 4. e00753. DOI: 10.1002/qute.202500753.
7. Лысаков И. В. Сведение задачи поиска ближайшего вектора для целочисленных решеток к задаче Изинга // *Дискретная математика*. 2025. Т. 37. № 3. С. 94–114. DOI: 10.4213/dm1876. EDN: GKMIFQ.
8. Tučs A., Tamura R., Tsuda K. Factorization Machine with Iterative Quantum Reverse Annealing: A Python Package for Batch Black-Box Optimization With Reverse Quantum Annealing // *Advanced Intelligent Discovery*. 2026. P. e202500231. DOI: 10.1002/aidi.202500231. EDN: OLAHLW.
9. Бурлаков Е. В. Возможности адиабатического квантового компьютера для распознавания паттернов // *Радиоэлектроника. Наносистемы. Информационные технологии*. 2025. Т. 17. № 2. С. 201–210. DOI: 10.17725/rensit.2025.17.201. EDN: TBIXNH.
10. Голов Д. О., Петров Н. А., Цирулев А. Н. Вариационный квантовый алгоритм для малоразмерных систем в базисе Паули // *Физико-химические аспекты изучения кластеров, наноструктур и наноматериалов*. 2024. № 16. С. 343–350. DOI: 10.26456/pcasenn/2024.16.343. EDN: RKGMMW.
11. Тырышкин С. Ю. Параллельное компактное моделирование квантовых схем // *Современные наукоемкие технологии*. 2025. № 1. С. 185–191. DOI: 10.17513/snt.40296. EDN: FJHZQS.
12. Salloum H., Savin S., Kholodov Y., Ryzhakov G., Farina M., Oseledets I. Quantum annealing for inverse kinematics in robotics // *Scientific Reports*. 2026. Vol. 16. Art. 4244. DOI: 10.1038/s41598-025-34346-z.
13. Черешко А. А. Методы управления технологическими процессами на основе ассоциативных прогнозирующих моделей: дис. ... канд. техн. наук: 2.3.3; Институт проблем управления им. В. А. Трапезникова Российской академии наук; диссовет 24.1.107.01 (Д 002.226.01). Москва, 2022. 116 с.
14. Малетин Н. В. О возможности решения масштабных одномерных задач инверсии сейсмических данных на современных квантовых отжигателях // *Геофизика*. 2023. № 2. С. 102–109. URL: <https://www.elibrary.ru/item.asp?id=53758091> (дата обращения: 04.07.2026). DOI: 10.34926/geo.2023.59.58.012. EDN: DFHSEV.
15. Кочегурова Е. А., Сайберт С. М., Татьянакина К. В. Оптимизация параметров гибридного алгоритма прогнозирования с использованием ансамбля моделей в режиме реального времени // *Известия Томского политехнического университета. Промышленная кибернетика*. 2024. Т. 2. № 4. С. 26–33. DOI: 10.18799/29495407/2024/4/76. EDN: KNTPDM.
16. D-Wave Systems. Operation and Timing // *D-Wave Quantum Computing Products Documentation*. URL: https://docs.dwavequantum.com/en/latest/quantum_research/operation_timing.html (дата обращения: 26.09.2026).
17. Silva F. F. C., Carvalho P. M. S., Ferreira L. A. F. M., Omar Y. A QUBO Formulation for Minimum Loss Network Reconfiguration // *IEEE Transactions on Power Systems*. 2023. Vol. 38. Is. 5. P. 4559–4571. DOI: 10.1109/TPWRS.2022.3214477.
18. Yang X., Wang K., Kim J., Park K.-B. Artificial Neural Network-Based FCS-MPC for Three-Level Inverters // *Journal of Power Electronics*. 2022. Vol. 22. Is. 12. P. 2158–2165. DOI: 10.1007/s43236-022-00535-6.
19. Donoso F., Mora A., Cardenas R., Angulo A., Saez D., Rivera M. Finite-Set Model-Predictive Control Strategies for a 3L-NPC Inverter Operating With Fixed Switching Frequency // *IEEE Transactions on Industrial Electronics*. 2018. Vol. 65. Is. 5. P. 3954–3965. DOI: 10.1109/TIE.2017.2760840.
20. Deleplanque S. L., Pérez Armas F., Aggoune R. Quantum annealing heuristics for the job shop scheduling problem with availability constraints // *Journal of Heuristics*. 2026. Vol. 32. Is. 2. DOI: 10.1007/s10732-026-09594-5. EDN: HHFMTQ.
21. Abbas A., Ambainis A., Augustino B., et al. Challenges and opportunities in quantum optimization // *Nature Reviews Physics*. 2024. DOI: 10.1038/s42254-024-00770-9.

Конфликт интересов: Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest: The authors declare that there is no conflict of interest.

Финансирование: Авторы заявляют об отсутствии внешнего финансирования.

Financing: The research was performed without external funding.