УДК 519.71

КВАДРАТИЧНОЕ ПРОГРАММИРОВАНИЕ В МОДЕЛИ РАСПРЕДЕЛЕНИЯ РЕСУРСОВ В СЕЛЬСКОМ ХОЗЯЙСТВЕ НА ОСНОВЕ КВАНТОВОГО АЛГОРИТМА

[1]***Мухамедиева Д.Т.***, [2]***Раупова М.Х.***

r.mokhinur@gmail.com

[1]НИУ Ташкентский институт инженеров ирригации и механизации сельского хозяйства, 100000, Узбекистан, Ташкент, улица Кари Ниязова 39;

[2]Чирчикский государственный педагогический университет, 111700, Узбекистан, Ташкент, г. Чирчик, Проспект Амира Темура, 104.

В данной работе рассматриваются применение квантовых алгоритмов в задаче оптимизации распределения ресурсов в сельском хозяйстве. В частности, исследуется возможность использования Quantum Approximate Optimization Algorithm для решения данной задачи. Этот является алгоритмом квантовой оптимизации, который может быть эффективно применен к комбинаторным оптимизационным задачам. Работа предполагает адаптацию алгоритма к задаче оптимизации распределения ресурсов в сельском хозяйстве, включая моделирование оптимизационной функции и учет ограничений на использование ресурсов. Предполагается, что использование квантового алгоритма может предложить новые подходы к решению сложных оптимизационных задач в сельском хозяйстве и способствовать повышению эффективности использования ресурсов и улучшению устойчивости производства.

Ключевые слова: квантовые алгоритмы, QAOA, оптимизация, распределение ресурсов, математическая модель, комбинаторная оптимизация.

Цитирование: *Мухамедиева Д.Т., Раупова М.Х.* Квадратичное программирование в модели распределения ресурсов в сельском хозяйстве на основе квантового алгоритма // Проблемы вычислительной и прикладной математики. – 2025. – № 2(64). – С. 101-113.

DOI: https://doi.org/10.71310/pcam.2_64.2025.09.

1 Введение

Оптимизация распределения ресурсов в сельском хозяйстве играет ключевую роль в повышении эффективности производства, увеличении выхода продукции и улучшении устойчивости аграрного сектора. Сложность этой задачи заключается в необходимости учета множества факторов, таких как доступность ресурсов, требования культур, климатические условия, рыночные требования и др. Традиционные методы оптимизации, такие как линейное или квадратичное программирование, могут столкнуться с ограничениями в эффективности решения задачи в силу ее сложности и размерности. В последние годы интерес к применению квантовых алгоритмов в оптимизационных задачах растет. Quantum Approximate Optimization Algorithm (QAOA) - один из перспективных квантовых алгоритмов, предназначенных для решения комбинаторных оптимизационных задач. Применение QAOA к задаче распределения ресурсов в сельском хозяйстве представляет собой новый подход, который может привести к более эффективному и устойчивому использованию ресурсов в аграрном секторе. В данной работе исследуется возможность применения QAOA к задаче оптимизации распределения ресурсов в сельском хозяйстве. Будут рассмотрены основные принципы работы QAOA, а также специфика его применения к задачам оптимизации в сельском хозяйстве. Предполагается, что использование квантового алгоритма может привести к более точным и быстрым решениям задачи оптимизации и способствовать повышению эффективности сельскохозяйственного производства. Использование квантовых алгоритмов в задачах оптимизации, таких как распределение ресурсов в сельском хозяйстве, является активной областью исследований. Один из квантовых алгоритмов, который может быть применен к таким задачам, это Quantum Approximate Optimization Algorithm (QAOA) [1–4].

Для применения QAOA к задаче распределения ресурсов в сельском хозяйстве, сначала требуется представить задачу в виде оптимизационной задачи, которую можно решить при помощи QAOA. Затем применяются методы квантовой оптимизации для решения этой задачи. Конкретный алгоритм QAOA включает в себя создание квантовой цепи, которая представляет оптимизационную задачу в виде оператора Шредингера. Затем применяются операторы эволюции и измерения, чтобы найти состояние, соответствующее оптимальному решению задачи. Применение QAOA к задаче распределения ресурсов в сельском хозяйстве требует дополнительной работы по адаптации алгоритма к конкретным условиям и требованиям этой задачи. Это может включать в себя разработку специальных функций стоимости, учета ограничений на ресурсы и т. д. В целом, использование квантовых алгоритмов, таких как QAOA, для решения задач распределения ресурсов в сельском хозяйстве может предложить новые методы оптимизации и позволить находить более эффективные решения в сравнении с классическими методами оптимизации [5–7].

Научная новизна данной работы заключается в применении квантового алгоритма Quantum Approximate Optimization Algorithm (QAOA) к задаче оптимизации распределения ресурсов в сельском хозяйстве. В настоящее время квантовые алгоритмы активно исследуются в контексте решения комбинаторных оптимизационных задач, однако их применение к задачам оптимизации в сельском хозяйстве относительно ново. Применение QAOA к задаче распределения ресурсов в сельском хозяйстве представляет собой новый подход, который может привести к более эффективному использованию ресурсов и улучшению производства в аграрном секторе. Данное исследование также включает в себя адаптацию QAOA к конкретным условиям сельского хозяйства, что представляет собой дополнительную научную ценность [8–10].

2 Материалы и методы

Рассмотрим нелинейные модели при наличии ограничений [1]:

$$F(\overline{x}) \to min, x \in E^n, \tag{1}$$

$$h_i(\overline{x}) = 0; i = \overline{1, m}, \tag{2}$$

$$g_i(\overline{x}) \geqslant 0; i = \overline{m+1, p}. \tag{3}$$

Для решения подобных задач широко используются методы нелинейного программирования. Один из подходов заключается в том, чтобы свести нелинейную задачу к задаче линейного программирования. Для этого используется процедура линеаризации, которая преобразует нелинейные функции в линейные приближения в окрестности некоторой точки. Это позволяет затем применить методы решения задач линейного программирования для нахождения оптимального решения.

В этом случае задача нелинейного моделирования в общей постановке может быть модифицирована путем записи каждой из нелинейных функций этой задачи двумя первыми числами в соответствующем разложении в ряд Тейлора в окрестности $\bar{x}^{(k)}$ и сведена к задаче линейного программирования [1, 2]:

$$\{f(\overline{x}^{(k)}) + \bigtriangledown^r h_i(\overline{x}^{(k)})(\overline{x} - \overline{x}^{(k)})\} \Rightarrow min, \overline{x} \in E^n \tag{4}$$

при ограничениях

$$h_i(\overline{x}^k) + \bigtriangledown^T h_i(\overline{x}^k)(x - \overline{x}^k) = 0, i = \overline{1, m}, \tag{5}$$

$$g_i(\overline{x}^k) + \bigtriangledown^T g_i(\overline{x}^k)(x - \overline{x}^k) \geqslant 0, i = \overline{m + 1, p}. \tag{6}$$

Процедура линеаризации часто используется в итерационном процессе. Она позволяет на каждой итерации линеаризовать нелинейные функции в текущей точке и решать соответствующую линейную задачу. Затем процесс итерации продолжается до достижения условия останова, такого как сходимость к оптимальному решению или достижение максимального числа итераций. Такой подход широко применяется в задачах управления, где нелинейные модели могут быть сведены к линейным задачам приближенно.

Сходимость методов линейной аппроксимации к решению гарантируется при выполнении условий:

Функции $f(\bar{x}), h_1(\bar{x}), ..., h_m(\bar{x}), g_{m+1}(\bar{x}), ..., g_p(\bar{x})$ должны быть непрерывными и дифференцируемыми.

Функция $f(\bar{x})$ должна быть выпуклой, а ограничения $\sum h_j^2(\bar{x})$ должны быть выпуклыми в допустимой области D.

Функции $g_{m+1}(\bar{x}), ..., g_p(\bar{x})$ должны быть вогнутыми.

Область D должна быть замкнутой, выпуклой и непустой.

Все параметры, встречающиеся в условиях функции, должны быть ограничены.

При невыполнении этих условий методы, использующие линейную аппроксимацию, могут найти только локальный оптимум.

В контексте задач линейного программирования (ЛП) для координации решений и учета допущений добавляется дополнительное условие: ограничение длины шага при перемещении в различных направлениях до небольшой величины $\delta_j^{(k)} > 0$. Это помогает предотвратить слишком большие шаги, которые могут привести к выходу из области, где аппроксимация действительна

$$\delta_j^{(k)} - |\overline{x}_j^{k+1} - \overline{x}_j^k| \geqslant 0, j = \overline{1, n}. \tag{7}$$

Решение задачи (4)-(6) позволяет вычислить следующую точку:

$$\bar{x}^{(k+1)} = \bar{x}^{(k)} + F^0(\hat{x}, \bar{x}^{(k)}, k). \tag{8}$$

Повторное применение этой процедуры с меньшим значением $\delta^{(k)}$ происходит до тех пор, пока минимизирующая поправка к найденному на предыдущем шаге значению оказывается меньше предварительно заданного числа. Это позволяет получить более точное решение задачи.

Если начальный вектор $\bar{x}^{(0)}$ оказывается вне пределов допустимой области D, вводятся искусственные переменные $\omega_i^{(k)} > 0$, которые добавляются в ограничения равенства и неравенства. Это позволяет привести начальную точку к допустимой области и начать процесс оптимизации.

Метод обобщенного приведенного градиента, который входит в класс алгоритмов оптимизации и применяется в работах [1, 5]. Этот метод эффективен при решении задач с непрерывной и дифференцируемой функцией цели по управляющим переменным, а также в случае, когда параметры задачи находятся в непрерывных диапазонах. Метод обобщенного приведенного градиента является итерационным методом оптимизации, который использует информацию о градиенте целевой функции для поиска оптимального решения. Он сочетает в себе идеи градиентного спуска с методом поиска оптимального шага, что делает его эффективным при решении различных задач оптимизации.

Однако стоит отметить, что этот метод не всегда применим в случаях, когда функция цели не является непрерывной и дифференцируемой, либо когда параметры задачи имеют дискретные значения или находятся в дискретных диапазонах. В таких случаях могут потребоваться другие методы оптимизации, специально адаптированные для этих условий.

Когда управляющие или координирующие параметры имеют дискретные значения, применяются другие методы оптимизации. Одним из таких методов является динамическое программирование, которое широко используется для решения задач оптимизации в условиях дискретности параметров.

Декомпозиция задачи позволяет использовать при решении каждой локальной задачи тот метод, который наиболее эффективен для данной ситуации. Параметрическая декомпозиция включает фиксацию некоторых параметров и решение основной задачи с учетом этих фиксированных параметров. Это позволяет упростить задачу и сосредоточиться на оптимизации конкретных параметров. Параметрическая декомпозиция - это подход к решению сложных задач оптимизации, который заключается в разбиении основной задачи на более простые подзадачи с последующим объединением их результатов. Основная идея состоит в том, чтобы фиксировать некоторые параметры задачи и решить ее в условиях этих фиксированных параметров. Сначала определяются параметры, которые будут фиксироваться в процессе декомпозиции. Выбор этих параметров может зависеть от конкретной задачи и целей оптимизации. Задача разбивается на более мелкие подзадачи, учитывая фиксированные параметры. Это может быть выполнено различными способами, такими как разбиение по времени, пространственное разбиение, или любой другой подход, который учитывает специфику задачи. Каждая подзадача решается независимо с учетом фиксированных параметров.

Для решения этих задач могут применяться различные методы оптимизации, такие как динамическое программирование для дискретных параметров или методы оптимизации для непрерывных параметров. После решения всех локальных задач и получения оптимальных значений для каждой подзадачи, результаты объединяются для получения решения основной задачи. Параметрическая декомпозиция позволяет эффективно решать сложные задачи оптимизации, разбивая их на более простые компоненты. Этот подход особенно полезен, когда управляющие или координирующие параметры имеют дискретные значения, и может быть эффективно применен в широком спектре прикладных задач.

При параметрической декомпозиции [1, 4] производится фиксация r параметров $(x_{n+1}, ..., x_{n+r})$ и решается основная задача (1):

$$\sum_{j=1}^{n} a_{ij}x_j + g_i(x_{n+1}, ..., x_{n+r}) = b_i, i = \overline{1,m},$$

$$\underline{U_k} \leqslant x_{n+k} \leqslant \overline{U_k}, k = \overline{1,r},$$

$$x_j \geqslant 0, j = \overline{1,n}, \tag{9}$$

$$F = \sum_{j=1}^{n} a_{0j} x_j + g_0(x_{n+1}, ..., x_{n+r}) \to min, \tag{10}$$

размерность которой сокращается на r [1]. В результате решения этой задачи линейного программирования находятся базисные, небазисные переменные и двойственные оценки $\prod_1, ..., \prod_m$ для ограничений (9). После этого решается вторая задача

$$\sum_{i=1}^{n} \prod_i g_i(x_{n+1}, ..., x_{n+r}) + g_0(x_{n+1}, ..., x_{n+r}) \to min, \tag{11}$$

при ограничениях

$$\overline{x}^{k+1} = \overline{x}^k + F^0(\widehat{x}, \overline{x}^k, k),$$

$$\underline{U_k} \leqslant x_{n+k} \leqslant \overline{U_k}, k = \overline{1, r}. \tag{12}$$

Находим все небазисные переменные в задаче (12), которые были найдены при решении задачи (9)-(10), и приравниваем их к нулю. Осуществляем линейную аппроксимацию ограничений (12) и целевой функции, чтобы преобразовать исходную нелинейную задачу в линейную задачу. Решаем полученную линейную задачу. Полученное решение используется для нахождения нового решения исходной задачи (9)-(10). Процедура повторяется, пока не будет достигнута заданная точность решения. Этот процесс позволяет решить сложные нелинейные задачи оптимизации путем линеаризации и последующей итерационной оптимизации. Проведение структурной декомпозиции исходной задачи путем линеаризации исходной модели открывает новые возможности для использования линейных моделей с переменными коэффициентами. Это позволяет эффективно решать разнообразные задачи оптимизации, так как линейные модели часто имеют хорошо разработанные методы решения и могут быть решены с высокой точностью и эффективностью.

Применение этих моделей дает возможность решить задачу следующего вида [1,9]:

$$G_i = \sum_{j=1}^{n} f_{ij}(\overline{u}) x_j - b_i = 0, i = \overline{1, n},$$

$$x_j \geqslant 0, j = \overline{1, R}, \tag{13}$$

$$F = \sum_{j=1}^{n} F_j(\overline{u}) x_j \to min, \tag{14}$$

где f_{ij}, F_j- заданные функции управлений $\underline{u} = \{u_1, ..., u_t\}, u \in U$.

Такую модель можно записать, когда работа каждого элемента описывается линейной моделью с переменными коэффициентами вида

$$y_i^{(k)} = \sum_{j=1}^{R_i} f_{ij}^{(k)}(\overline{u}^{(k)}) x_j^{(k)}, i = \overline{1, n},$$

$$x_j^{(k)} \geqslant 0, j = \overline{1, R_i}, \tag{15}$$

$$F^{(k)} = \sum_{j=1}^{R_i} F_j^{(k)}(\overline{u}^{(k)}) x_j^{(k)} \to min. \tag{16}$$

Модель отдельного элемента (агрегата) можно рассматривать как результат линеаризации нелинейной модели

$$y_i^{(k)} = f_i^{(k)}(\bar{x}^{(k)}, \bar{u}^{(k)}), i = \overline{1, m_k} \tag{17}$$

по входным параметрам $\bar{x}^{(k)}$, например, путем разложения функции $f_i^{(k)}(\bar{x}^{(k)}, \bar{u}^{(k)})$ в ряд Тейлора в окрестности некоторого значения входных параметров $\bar{x}_0^{(k)}$ и пренебрегая в этом разложении членами второго порядка и выше [1–3].

Понятие граничного режима используется для аппроксимации производственных возможностей агрегатов, которые могут работать в различных режимах в пределах интервала времени $[\underline{u}_i, \bar{u}_i]$. Граничные режимы представляют собой предельно возможные условия работы технологических агрегатов, которые определяются для каждого агрегата. Промежуточные режимы работы агрегатов могут быть аппроксимированы с использованием линейных комбинаций граничных режимов. Это позволяет учесть различные варианты работы агрегатов внутри заданного интервала времени. Путем последовательной фиксации значений параметра в интервале $[\underline{u}_i, \bar{u}_i]$для каждого из этих значений с использованием принципа параметрической декомпозиции можно получить последовательность оптимальных значений функционала F. Затем из этой последовательности выбирается решение, дающее наименьшее значение функционала F.

Таким образом, принцип параметрической декомпозиции позволяет рассматривать различные варианты работы агрегатов в пределах заданного интервала времени и выбирать оптимальное решение с учетом всех возможных вариантов. Этот подход является эффективным способом учета разнообразных условий и ограничений при оптимизации производственных процессов.

Если объект обладает свойством внутренней оптимизации и стремится максимизировать свою полезность или другой функционал в изменяющихся условиях, то это может способствовать саморегуляции его поведения. Это свойство может позволить объекту выбирать устойчивые и рациональные стратегии в ответ на изменения в ресурсном обеспечении и расходных коэффициентах. В случае, если объект способен самостоятельно анализировать и оценивать окружающую среду, а также свое внутреннее состояние, он может адаптироваться к изменяющимся условиям и выбирать оптимальные действия без прямого внешнего управления. Однако, необходимо учитывать, что быстрые и неточные изменения условий, а также ограничения по доступу к информации могут затруднить процесс саморегуляции и принятия решений. Для решения подобных задач могут применяться методы нелинейного программирования, которые позволяют учитывать сложные взаимосвязи и ограничения в системе. Моделирование поведения объекта и оптимизация его стратегий могут помочь понять, как объект будет реагировать на различные сценарии изменений в среде.

Эта задача нелинейного программирования записывается следующим образом [1,4-7] [1, 4–7]:

Пусть x – вектор продуктов, R – вектор ресурсов. Мы имеем аддитивную функцию полезности $f(x, \alpha)$

$$y = f(x, \alpha) = \sum_{j=1}^{n} \alpha_j f_j(x_j) \to \max \tag{18}$$

и аддитивную ограничения по ресурсам $g(x)$

$$g(x) = \sum_{j=1}^{n} g_j(x_j, \beta_j) \leqslant R, \tag{19}$$

$$x \in Q, \tag{20}$$

где $j = 1, \ldots, n$ - индекс продукта; $R = (R_1, ..., R_m)$ - вектор ресурсов; $i = 1, \ldots, m$ – индекс ресурса; $f_j(x_j)$ - функция полезности производства j – го продукта x_j; $f(x, \alpha)$- функция полезности общего объема j- продуктов; $g_j(x_j, \beta_j)$ - векторная строго выпуклая функция потребления ресурсов при производстве j – го продукта x_j; β_j- вектор параметров j –й технологии, $\beta_j \in B_j$ - выпуклые множества; Q -выпуклое множество.

Пусть x^*- решение задачи (18) – (20), а $f(x, \alpha)$- измеряется как общий объем материальных благ.

Модель должна учитывать неопределенность в значениях параметров, а не просто принимать их как фиксированные числа. Такой подход позволяет более адекватно отражать реальные условия и учет возможных вариаций параметров, что важно для точности и надежности модели. Моделирование неопределенности параметров в задачах оптимизации приводит к понятию стохастического программирования или программирования с неопределенными данными. В таких моделях параметры могут принимать значения из некоторого вероятностного распределения, а целью является нахождение оптимальной стратегии управления при неопределенных условиях. Использование стохастического программирования позволяет учесть неопределенность и риски при принятии решений, что особенно важно в условиях переменной среды и различных факторов, влияющих на производственные процессы. Моделирование неопределенности также может включать в себя различные методы анализа рисков, такие как анализ чувствительности, сценарный анализ и вероятностный анализ, чтобы оценить возможные последствия различных вариантов значений параметров.

Такой подход может помочь в создании более гибких и адаптивных стратегий управления, которые могут эффективно работать в условиях неопределенности и изменчивости внешней среды $[1, 11]$.

Распишем Лагранжиан задачи (18)-(20):

$$L = \sum_{j=1}^{n} \alpha_j f(x_j) - \sum_{j=1}^{n} \lambda_j [R - g_j(x_j, \beta_j)]^2.$$

Имеем систему уравнений:

$$\begin{cases} \frac{\partial L}{\partial x_j} = \alpha_j \frac{\partial f}{\partial x_j} - 2\lambda_j \frac{\partial g_j}{\partial x_j} = 0, \\ \frac{\partial L}{\partial \lambda_j} = R - g_j\left(x_j, \beta_j\right) = 0. \end{cases} \tag{21}$$

Всякое решение системы уравнений определяет точку x^0, в которой может иметь место экстремум функции $f(x, \alpha)$. Следовательно решив систему уравнений (22) получают все точки, в которых функция (18) может иметь экстремальные значения. Среди точек, подозрительных на экстремум, находят такие x^*, в которых достигается экстремум, и вычисляют значения функции (18) в этих точках.

3 Результаты

В качестве исходной задачи оптимизации рассматривается задача распределения ресурсов в сельском хозяйстве. Это включает в себя определение целевой функции (максимизация прибыли), ограничения на использование ресурсов (линейные и/или квадратичные) и другие факторы, характеризующие производственные процессы в сельском хозяйстве. В данном исследовании используются квантовые алгоритмы оптимизации, QAOA (Quantum Approximate Optimization Algorithm), для решения подзадачи квантовой оптимизации, которая включена в процесс оптимизации ADMM.

На основе полученных данных разрабатывается математическая модель задачи оптимизации распределения ресурсов. В модели учитываются целевая функция (максимизация прибыли), ограничения на использование ресурсов и другие факторы, характеризующие производственные процессы.Для учета квадратичного программирования в модели распределения ресурсов в сельском хозяйстве, мы можем изменить целевую функцию таким образом, чтобы она учитывала квадратичные зависимости между переменными. Предположим, что у нас есть квадратичная зависимость между количеством использованных ресурсов и прибылью, то есть мы стремимся максимизировать квадратичную функцию прибыли относительно использования ресурсов.

Пусть Q - это квадратичная матрица, представляющая квадратичные зависимости между переменными x_j (продукция) и b_i (ресурсы). Тогда целевая функция будет иметь вид:

$$F = \sum_{j=1}^{n} c_j x_j - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} Q_{ij} x_i x_j \to \max .$$

Эта функция максимизирует прибыль, вычитая из нее штраф за использование ресурсов, учитывая квадратичные зависимости между использованием ресурсов и прибылью.

При этом ограничения на использование ресурсов останутся линейными, как описано ранее:

$$\sum_{j=1}^{n} a_{ij} x_j \leqslant b_i, \quad i = 1, 2, ..., m.$$

Разрабатывается алгоритм применения QAOA к задаче оптимизации распределения ресурсов. Этот алгоритм включает в себя создание квантовой цепи, представляющей оптимизационную задачу, применение операторов эволюции и измерений для нахождения состояния, соответствующего оптимальному решению задачи [11, 12].

1. Задание начальных параметров алгоритма. Начальные параметры алгоритма, такие как глубина квантовой цепочки p и начальные значения углов γ и β, могут быть выбраны на основе эмпирических данных и опыта исследователя. Важно учитывать, что оптимальные значения этих параметров могут зависеть от конкретной задачи оптимизации и свойств применяемой квантовой аппаратуры. Глубина цепочки определяет количество слоев в QAOA. Обычно начальное значение выбирается в диапазоне от 1 до 5. Большие значения p могут повысить точность решения, но при этом увеличивается вычислительная сложность. Начальные значения углов могут быть выбраны случайным образом в заданном диапазоне или с использованием методов оптимизации, таких как случайный поиск или методы оптимизации на основе градиента. Эти углы обычно выбираются из диапазона $[0, \pi]$. После выбора начальных параметров алгоритм QAOA выполняется для генерации решения задачи оптимизации. Полученные результаты могут затем быть оценены и, при необходимости, параметры могут быть дополнительно настроены для улучшения качества решения [13, 14].

2. Создание начального квантового состояния $|\psi(\gamma,\beta)\rangle$ является важным шагом в алгоритме QAOA и может зависеть от конкретной задачи оптимизации. Обычно это начальное состояние выбирается таким образом, чтобы оно было суперпозицией различных базисных состояний, чтобы обеспечить равномерное покрытие пространства состояний и увеличить вероятность нахождения оптимального решения.

В качестве примера, для задачи оптимизации распределения ресурсов в сельском хозяйстве, начальное квантовое состояние может быть сформировано суперпозицией всех возможных состояний, представляющих различные варианты распределения ресурсов. Это может быть реализовано путем применения оператора Адамара ко всем кубитам, что приведет к созданию равновероятной суперпозиции всех базисных состояний.

Таким образом, начальное квантовое состояние $|\psi(\gamma,\beta)\rangle$может быть представлено следующим образом [15]:

$$|\psi(\gamma,\beta)\rangle = H^{\otimes n}|0\rangle^{\otimes n},$$

где H - оператор Адамара, n - количество кубитов, а $|0\rangle$- базисное состояние кубита.

Это начальное суперпозиционное состояние позволяет исследовать широкий диапазон возможных решений задачи оптимизации в алгоритме QAOA.

3.Применение оператора эволюции $U(\gamma,\beta)$. Оператор эволюции играет ключевую роль в алгоритме QAOA. Этот оператор зависит от задачи оптимизации и от текущих значений параметров γ_i и β_i. Обычно он представляется как последовательное применение гейтов, описываемое следующим образом:

$$U(\gamma,\beta) = e^{-i\beta_p B}e^{-i\gamma_p C}...e^{-i\beta_1 B}e^{-i\gamma_1 C},$$

где B_i и C_i - это гамильтонианы, соответствующие задаче оптимизации, а γ_i и β_i - текущие значения параметров на шаге i.

В алгоритме QAOA обычно используются два типа гамильтонианов:

Гамильтонианы C_i представляют собой гамильтонианы, связанные с целевой функцией задачи оптимизации.

Гамильтонианы B_i представляют собой гамильтонианы, которые помогают вращать состояния кубитов вокруг оси Шредингера и увеличивают вероятность обнаружения оптимального решения. Обычно они выбираются как сумма по всем кубитам операторов Паули X или Z.

После применения оператора эволюции $U(\gamma,\beta)$к начальному состоянию $|\psi(\gamma,\beta)\rangle$, получается новое состояние. Это состояние содержит информацию о текущем решении задачи оптимизации и будет использоваться для вычисления целевой функции и последующей оптимизации параметров γ_i и β_i на следующей итерации алгоритма [16].

4. Шаги 2 и 3 алгоритма QAOA обычно повторяются несколько раз с различными значениями параметров γ_i и β_i, образуя итерационный процесс. Это позволяет постепенно улучшать качество решения задачи оптимизации и приближаться к оптимальному решению. Процесс повторяется до тех пор, пока не будет достигнут критерий остановки, такой как заданное количество итераций или достижение определенной точности решения.

5. После завершения итераций алгоритма QAOA проводится измерение квантового состояния, которое содержит информацию о текущем решении задачи оптимизации. Измерение приводит к коллапсу квантового состояния в классическую конфигурацию, в которой каждый кубит принимает определенное значение. Затем на

основе результатов измерения оцениваются значения целевой функции и других характеристик задачи оптимизации. Полученные результаты измерения представляют собой строки бит, которые соответствуют различным возможным решениям задачи оптимизации. Для каждого измеренного решения вычисляется значение целевой функции, которое позволяет оценить качество решения. Например, в задаче максимизации прибыли это может быть общая прибыль от распределения ресурсов, а в задаче минимизации затрат - общие затраты или потери. Из всех измеренных решений выбирается то, которое соответствует оптимальному значению целевой функции. Это решение является решением задачи оптимизации, полученным с использованием алгоритма QAOA. Полученное оптимальное решение и его значение целевой функции анализируются для понимания эффективности алгоритма и его пригодности для решения конкретной задачи оптимизации. Измерение квантового состояния и оценка значений задачи оптимизации на основе результатов измерения позволяют получить конечный результат работы алгоритма QAOA и оценить его эффективность в решении задач оптимизации [17].

6. Параметры алгоритма γ и β обновляются в соответствии с полученными результатами после завершения каждой итерации алгоритма QAOA. Это позволяет уточнить значения параметров и улучшить качество решения задачи оптимизации. Процесс обновления параметров может осуществляться различными способами, в зависимости от выбранной стратегии оптимизации.

7. Процесс продолжается до достижения критерия остановки, который определяется заранее и может быть различным в зависимости от конкретной задачи и требований пользователя. Алгоритм может быть остановлен после выполнения определенного числа итераций. Это позволяет контролировать время выполнения алгоритма и ограничить его вычислительную сложность. Алгоритм может быть остановлен, когда достигнуто решение, которое удовлетворяет определенным критериям качества. Например, значение целевой функции может быть близким к оптимальному или удовлетворять другим заданным условиям.

Оператор $e^{-i\beta_p B}$может быть реализован в виде квантового гейта, используя однокубитные вращения. Форма гейта поворота $R_z(\theta)$в виде матрицы:

$$R_z(\theta) = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}.$$

Таким образом, оператор эволюции $e^{-i\beta_p B}$может быть реализован следующим образом:

$$e^{-i\beta_p B} = R_z(-2\beta_p B).$$

Этот гейт $R_z(-2\beta_p B)$действует на одиночный кубит и поворачивает его вокруг оси Z на угол, пропорциональный параметру β_p и весу B.

Приведем следующий пример.

Максимизировать: $-2 * x_0 * x_2 - x_1 * x_2 - x_0 + 2 * x_1 - 3 * x_2$.

При ограничениях:

- линейные ограничения (1): $x_0 + x_1 + x_2 <= 1$;

- бинарные переменные (3): $x_0 x_1 x_2$.

Значение целевой функции: -6.0.

Значения переменных: $x_0 = 1.0, x_1 = 0.0, x_2 = 1.0$.

Статус: УСПЕШНО РЕШЕНО

Квантовая схема для решения задачи приведена на рис. 1.

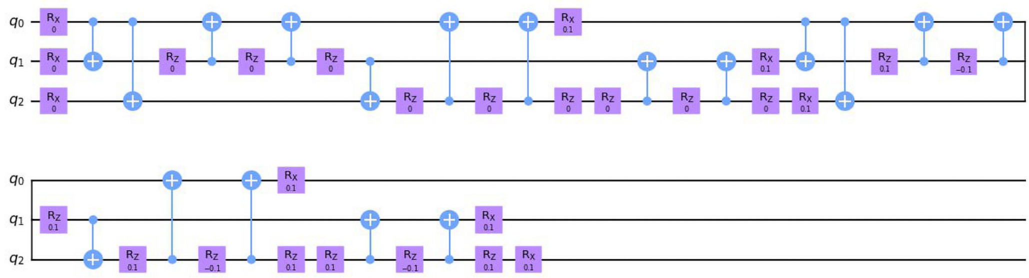

Рис. 1 Квантовая схема

Был успешно применен квантовый алгоритм Quantum Approximate Optimization Algorithm (QAOA) к задаче оптимизации распределения ресурсов в сельском хозяйстве. QAOA позволил решить задачу оптимизации, учитывая сложные зависимости между ресурсами, культурами и другими факторами производства. Были получены оптимальные решения задачи оптимизации, которые позволяют максимизировать прибыль от производства сельскохозяйственной продукции при ограниченных ресурсах. Решения учитывают различные факторы, такие как доступность ресурсов, требования культур к ресурсам и другие ограничения.

4 Заключение

В данной работе был исследован подход применения квантового алгоритма Quantum Approximate Optimization Algorithm (QAOA) к задаче оптимизации распределения ресурсов в сельском хозяйстве. Применение QAOA показало свою эффективность в решении сложных оптимизационных задач в сельском хозяйстве. QAOA способен учитывать множество факторов, таких как доступность ресурсов, требования культур и другие ограничения, что позволяет получать оптимальные решения задачи распределения ресурсов. Полученные решения при применении QAOA демонстрируют высокое качество и эффективность. Они позволяют максимизировать прибыль от производства сельскохозяйственной продукции при ограниченных ресурсах. Результаты исследования свидетельствуют о перспективности применения квантовых алгоритмов в сельском хозяйстве. Это открывает новые возможности для решения сложных оптимизационных задач в аграрном секторе и повышения эффективности производства.

Дальнейшие исследования могут быть направлены на расширение применения квантовых алгоритмов в сельском хозяйстве, включая адаптацию других квантовых алгоритмов, учет дополнительных факторов в моделировании и улучшение методов оптимизации. Также возможно исследование применения квантовых алгоритмов в других отраслях, связанных с оптимизацией ресурсов. В целом, результаты данного исследования подтверждают потенциал квантовых алгоритмов в решении сложных задач оптимизации в сельском хозяйстве и открывают новые перспективы для развития аграрного сектора с использованием современных технологий.

Литература

[1] *Zahedinejad E. and A. Zaribafiyan* Combinatorial optimization on gate model quantum computers: A survey, – 2017. arXiv:1708.05294.

[2] *Kandala A. et al.,* Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets,Nature, vol. 549, no. 7671, – P. 242–246. – 2017.

[3] *Farhi E. and Neven H.* Classification with quantum neural networks on near term processors, 2018. arXiv:1802.06002.

[4] *Sun K. and Sun X.A.,* A two-level distributed algorithm for general constrained non-convex optimization with global convergence, – 2019. arXiv:1902.07654.

[5] *Farhi E., Goldstone J., Gutmann S. and Neven H.* Quantum algorithms for fixed qubit architectures, – 2017. arXiv:1703.06199.

[6] *Nannicini G.* Performance of hybrid quantum-classical variational heuristics for combinatorial optimization, Phys. Rev. E, vol. 99, no. 1, Art. no. 013304.– 2019.

[7] *Braine L., Egger D.J., Glick J., Woerner S.* Quantum algorithms for mixed binary optimization applied to transaction settlement, – 2019. arXiv:1910.05788.

[8] *Vyskocil T., Pakin S., Djidjev H.N.* Embedding inequality constraints for quantum annealing optimization, in Proc. Int. Workshop Quantum Technol. Optim. Problems – 2019. – P. 11–22.

[9] *Davis D. and Yin W.* Convergence rate analysis of several splitting schemes, in Splitting Methods in Communication and Imaging, Science and Engineering, R. Glowinski, S. Osher, and W. Yin, Eds. Cham, Switzerland: Springer, – 2017.

[10] *Ghadimi E., Teixeira A., Shames I., Johansson M.,* Optimal parameter selection for the alternating direction method of multipliers (ADMM): Quadratic problems, IEEE Trans. Autom. Control, vol. 60, no. 3, – 2015. – P. 644–658.

[11] *Giselsson P., Boyd S.* Linear convergence and metric selection for Douglas-Rachford splitting and ADMM, IEEE Trans. Autom. Control, vol. 62, no. 2, – 2017. – P. 532–544.

[12] *Wang Y., Yin W., Zeng J.* Global convergence of ADMM in nonconvex nonsmooth optimization, J. Sci. Comput., vol. 78, no. 1, – 2019. – P. 29–63.

[13] *Wu B. and Ghanem B.* p-box ADMM: A versatile framework for integer programming, IEEE Trans. Pattern Anal Mach. Intell., vol. 41, no. 7, – 2019. – P. 1695–1708.

[14] *Themelis A. Patrinos P.* Douglas-Rachford splitting and ADMM for nonconvex optimization: Tight convergence results, SIAM J. Optim., vol. 30, no. 1, – 2020. – P. 149–181.

[15] *Kueng R., Stilck França D.* Faster quantum and classical SDP approximations for quadratic binary optimization, – 2019. arXiv:1909.04613.

[16] *Romero J., Olson J.P., Aspuru-Guzik A.* Quantum autoencoders for efficient compression of quantum data, Quantum Sci. Technol., vol. 2, no. 4, – 2017.

[17] *Gilliam A., Woerner S., Gonciulea C.* Grover adaptive search for constrained polynomial binary optimization, – 2019. arXiv:1912.04088.

Поступила в редакцию 03.04.2025

UDC 519.71

QUADRATIC PROGRAMMING IN THE RESOURCE ALLOCATION MODEL IN AGRICULTURE BASED ON THE QUANTUM ALGORITHM

[1]***Mukhamedieva D.T.***, [2]***Raupova M.Kh.***

r.mokhinur@gmail.com

[1]Tashkent Institute of Irrigation Engineers and Agricultural Mechanization NRU,
39, Kari Niyazov Street, Tashkent, 100000, Uzbekistan;
[2]Chirchik State Pedagogical University,
104, Amir Temur Avenue, Chirchik, 111700 Uzbekistan.

This paper discusses the application of quantum algorithms to the problem of optimizing resource allocation in agriculture. In particular, the possibility of using the Quantum Approximate Optimization Algorithm to solve this problem is investigated. This is a quantum optimization algorithm that can be effectively applied to combinatorial optimization problems. The work involves adapting the algorithm to the problem of optimizing resource allocation in agriculture, including modeling the optimization function and taking into account resource use constraints. It is assumed that the use of a quantum algorithm can offer new approaches to solving complex optimization problems in agriculture and contribute to increasing the efficiency of resource use and improving the sustainability of production.

Keywords: quantum algorithms, QAOA, optimization, resource allocation, mathematical model, combinatorial optimization.

Citation: Mukhamediyeva D.T., Raupova M.H. 2025. Quadratic programming in the resource allocation model in agriculture based on the quantum algorithm. *Problems of Computational and Applied Mathematics.* 2(64): 101-113.

DOI: https://doi.org/10.71310/pcam.2_64.2025.09.

ПРОБЛЕМЫ ВЫЧИСЛИТЕЛЬНОЙ И ПРИКЛАДНОЙ МАТЕМАТИКИ

№ 2(64) 2025

Журнал основан в 2015 году.
Издается 6 раз в год.

Учредитель:
Научно-исследовательский институт развития цифровых технологий и искусственного интеллекта.

Журнал зарегистрирован в Агентстве информации и массовых коммуникаций при Администрации Президента Республики Узбекистан.
Регистрационное свидетельство №0856 от 5 августа 2015 года.

ISSN 2181-8460, eISSN 2181-046X

Адрес редакции:
100125, г. Ташкент, м-в. Буз-2, 17А.
Тел.: +(998) 712-319-253, 712-319-249.
Э-почта: `journals@airi.uz`.
Веб-сайт: `https://journals.airi.uz`.

Дизайн и вёрстка:
Шарипов Х.Д.

Отпечатано в типографии НИИ РЦТИИ.
Подписано в печать 25.04.2025 г.
Формат 60x84 1/8. Заказ №2. Тираж 100 экз.

PROBLEMS OF COMPUTATIONAL AND APPLIED MATHEMATICS

No. 2(64) 2025

The journal was established in 2015.
6 issues are published per year.

Founder:
Digital Technologies and Artificial Intelligence Development Research Institute.

The journal is registered by Agency of Information and Mass Communications under the Administration of the President of the Republic of Uzbekistan.
The registration certificate No. 0856 of 5 August 2015.

ISSN 2181-8460, eISSN 2181-046X

Address:
100125, Tashkent, Buz-2, 17A.
Tel.: +(998) 712-319-253, 712-319-249.
E-mail: journals@airi.uz.
Web-site: https://journals.airi.uz.

Layout design:
Sharipov Kh.D.

DTAIDRI printing office.
Signed for print 25.04.2025
Format 60x84 1/8. Order No. 2. Print run of 100 copies.

Содержание

Contents

№ 2(64) 2025 ISSN 2181-8460

HISOBLASH VA AMALIY MATEMATIKA MUAMMOLARI

ПРОБЛЕМЫ ВЫЧИСЛИТЕЛЬНОЙ И ПРИКЛАДНОЙ МАТЕМАТИКИ

PROBLEMS OF COMPUTATIONAL AND APPLIED MATHEMATICS

RAQAMLI TEXNOLOGIYALAR VA
SUN'IY INTELLEKTNI RIVOJLANTIRISH
ILMIY-TADQIQOT INSTITUTI