Санкт-Петербургский государственный университет

Черников Артем Александрович

Выпускная квалификационная работа

Применение генетических алгоритмов и градиентного спуска для поиска схем запутывающих преобразований в линейной квантовой оптике

Уровень образования: бакалавриат

Направление *02.03.03 «Математическое обеспечение и администрирование информационных систем»*

Основная образовательная программа *СВ.5006.2019 «Математическое обеспечение и администрирование информационных систем»*

Научный руководитель:
к.ф.-м.н., старший преподаватель кафедры системного программирования СПбГУ
С.С. Сысоев

Рецензент:
заместитель генерального директора ООО «ПетроМС» Ю.Г. Велюхов

Санкт-Петербург
2023

Saint Petersburg State University

Artem Chernikov

Bachelor's Thesis

Genetic algorithms and gradient descent application to the search of linear optics schemes for entangling quantum gates

Education level: bachelor

Speciality *02.03.03 "Software and Administration of Information Systems"*

Programme *CB.5006.2019 "Software and Administration of Information Systems"*

Scientific supervisor:
C.Sc., senior lecturer of the software engineering chair S.S. Sysoev

Reviewer:
deputy CEO at PetroMS LLC Y.G. Veliuchov

Saint Petersburg
2023

Оглавление

Введение

Квантовые вычислительные устройства уже долгое время вызывают интерес у научного сообщества благодаря своей способности решать некоторые задачи намного быстрее, чем классические компьютеры [10, 13], тем самым демонстрируя так называемое квантовое превосходство. Впервые идея квантовых вычислений была предложена независимо Юрием Маниным и Ричардом Фейнманом в начале 1980-х [15, 17], но исследования в этой области начались ещё раньше [12]. С тех пор и по сей день активно рассматриваются способы создания квантовых компьютеров, а также открываются и изобретаются новые квантовые алгоритмы, то есть алгоритмы, исполняемые квантовым вычислительным устройством.

Одним из первых квантовых алгоритмов, демонстрирующих квантовое превосходство, является алгоритм Дойча-Джозы [5]. Впоследствии был предложен также алгоритм Бернштейна-Вазирани [1]. Несмотря на то что их практическое применение может вызывать сомнения, само их существование бросило вызов научному сообществу искать новые алгоритмы, демонстрирующие квантовое превосходство и решающие при этом более животрепещущие задачи.

Квантовый алгоритм, решающий очень важную задачу факторизации целого числа за полиномиальное время, был открыт американским учёным Питером Шором в 1994-м году [13]. На предположении о том, что такая задача нерешаема за обозримое количество времени, основаны многие даже современные криптосистемы. Отсюда стало ясно, что с помощью квантового компьютера можно взломать любую такую систему, и это открытие вызвало сильный интерес к квантовым компьютерам.

Спустя время в 1996-м году американский математик Лов Гровер изобрёл алгоритм поиска в неотсортированной базе данных [10], имеющий квадратичное ускорение по сравнению с лучшими известными классическими алгоритмами, решающими эту задачу. Алгоритм Гровера может быть использован для решения широкого спектра задач, в

частности, NP-полных задач.

Впоследствии американским физиком-теоретиком Дэвидом П. ДиВинченцо были сформулированы критерии [4], которым должно соответствовать вычислительное устройство для того, чтобы по праву называться квантовым компьютером. Было предложено множество архитектур квантовых компьютеров: основанные на явлении ядерного магнитного резонанса; использующие электроны, запертые в квантовых точках; основанные на ядерных спинах идентичных молекул; использующие запутанные фотоны.

Идея создать квантовый компьютер на фотонах вызывает особый интерес благодаря тому, что практически все критерии ДиВинченцо выполнимы в этой архитектуре без особых усилий. Единственная трудность остаётся в осуществлении преобразований, запутывающих пару фотонов. Исследования в этой области на текущий момент привели к некоторым любопытным результатам [7, 9, 11], однако, к сожалению, все они не масштабируемы, и задача нахождения запутывающего преобразования остаётся открытой.

В связи с этим возникло предположение о том, что для поиска таких преобразований стоит использовать неочевидные техники. Одним из таких подходов является “выращивание” интересующего преобразования с использованием генетических алгоритмов.

1. Постановка задачи

Целью данной работы является реализация алгоритмов поиска схем запутывающих преобразований в линейной квантовой оптике и нахождение преобразований лучше, чем известные на данный момент.

Для достижения данной цели были поставлены следующие задачи.

- Оптимизировать существующий генетический алгоритм поиска.
- Реализовать алгоритм поиска с помощью градиентного спуска.
- Сравнить два подхода и сделать выводы.

2. Описание решения

В данном разделе представлено описание контекста решаемой задачи, технических подробностей и реализации решения. Код доступен в репозитории[1], размещённом на веб-сервисе GitHub.

2.1. Математическая модель

Согласно квантовой теории поля, фотон, как и некоторые другие элементарные частицы, способен пребывать в нескольких состояниях одновременно до тех пор, пока неопределённость относительно его состояния нисколько не влияет на окружающую его вселенную. Такое сложное состояние называется квантовой суперпозицией.

Как только состояние окружающей вселенной начинает зависеть от состояния фотона (например, состояние фотона детектируется датчиком), с точки зрения её жителей неопределенность состояния фотона исчезает. При этом фотон переходит из суперпозиции в какое-нибудь одно из возможных состояний с некоторой определённой вероятностью. Этот процесс называется коллапсом волновой функции частицы, или измерением состояния частицы.

В описанном выше примере фотон и его окружение до измерения представляли собой две независимые системы. После измерения одна система стала зависеть от другой, поэтому их теперь невозможно рассматривать по отдельности. Фотон и его окружение теперь являют собой единую неразделимую систему. С этой точки зрения коллапса волновой функции после измерения не произошло, вместо этого сама вселенная стала пребывать сразу в нескольких состояниях, причем жители вселенной из одного состояния никак не могут провзаимодействовать с ними же из другого состояния. Поэтому приверженцы этой теории, в парадигме которой в данной работе и рассматриваются физические явления, называют такии версии вселенных параллельными.

Аналогичным образом данные рассуждения переносимы с системы

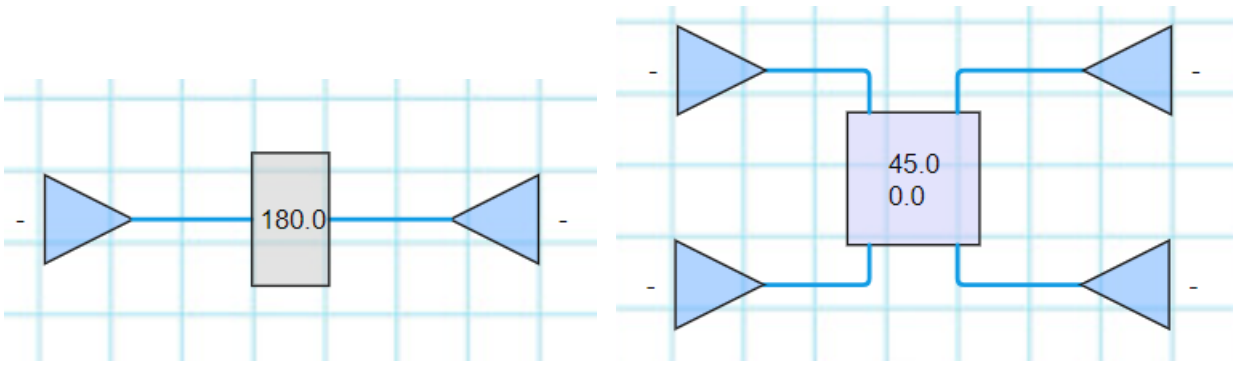
[1]Репозиторий проекта — https://github.com/sysoevss/galopy (дата обращения: 15.05.2023), пользователь artemgl

“фотон-окружение” на систему “фотон-фотон”. Если состояние двух фотонов не представимо в виде двух независимых систем (по одной для каждого фотона), то эти фотоны находятся в так называемом запутанном состоянии, в котором состояние одного фотона зависит от состояния второго. Существенным отличием данного примера от предыдущего является процесс запутывания, — на практике перевести два фотона из незапутанного состояния в запутанное оказывается нетривиальной задачей.

Квантовый алгоритм в достаточно широком смысле представляет собой некоторое обратимое преобразование над системой, способной находиться в N различных состояниях. Для решения содержательных задач зачастую число N должно быть катастрофически большим, поэтому часто рассматриваются системы, состоящие из n подсистем, называемых кубитами, каждая из которых способна находиться всего в двух состояниях — $|0\rangle$ или $|1\rangle$. Тогда общее количество состояний зависит экспоненциально от количества простых подсистем (кубитов) и равно 2^n. Доказано [3, 6], что в такой архитектуре произвольное преобразование над всей системой представимо в виде последовательности одно- и двухкубитных преобразований (гейтов), причём среди двухкубитных достаточно наличие всего одного запутывающего гейта.

В данной работе рассматривается архитектура квантового компьютера на фотонах, называемая протоколом KLM [9]. Каждый кубит кодируется ровно одним фотоном и парой пространственных мод, в которых может находиться этот фотон. Если он находится в первой моде, то состояние кубита считается равным $|0\rangle$, если во второй — $|1\rangle$.

Однокубитные преобразования в протоколе KLM осуществимы с помощью таких оптических элементов как фазовая пластина и светоделитель. Фазовая пластина имеет единственную входную и выходную моду и параметризуется всего одним углом, отвечающим за сдвиг фазы волновой функции проходящего через эту пластину фотона (рис. 1а). Светоделитель имеет два входных и два выходных плеча, что позволяет обеспечить взаимодействие двух мод между собой. Его принято параметризовать парой углов, один из которых отвечает за коэффици-



(a) Фазовая пластина. (b) Светоделитель.

Рис. 1: Схематичные изображения оптических элементов, используемых в квантовой линейной оптике. Треугольниками обозначены источники и детекторы фотонов, линиями — моды, в которых могут находиться фотоны. Параметры оптических элементов (углы) указываются в градусах.

ент пропускания, а другой — за фазовый сдвиг после отражения или пропускания фотона. Так, если первый угол равен нулю, то светоделитель вырождается в идеальное зеркало, что эквивалетно отсутствию каких-либо оптических элементов; если же он равен 90°, то светоделитель наоборот пропускает все проходящие через него лучи, тем самым меняя фотоны в двух модах местами (рис. 1b).

Двухкубитный запутывающий гейт в протоколе KLM неосуществим при помощи указанных выше оптических элементов. Однако исследования в этой области на данный момент пришли к тому, что с использованием дополнительных фотонов запутывающее преобразование становится осуществимо [7, 9]. Дополнительные фотоны взаимодействуют посредством оптических элементов с сигнальными (то есть теми, над которыми требуется произвести преобразование), после чего на дополнительных детекторах ожидается измерение определённого паттерна (к примеру, два фотона на первом датчике, один на втором и ноль на третьем). Считается, что гейт сработал правильно, тогда и только тогда, когда заданный паттерн оказался на датчиках в результате измерения. Измерение на дополнительных модах называется оповещением о работе гейта, а удачное измерение — правильным или верным оповещением.

Неотъемлемой частью гейтов с вышеописанной логикой работы яв-

ляется ненулевая вероятность неверного оповещения, что мешает использовать их на практике, поскольку для работы всего алгоритма, состоящего из сотен или тысяч таких гейтов, требуется, чтобы сработали все гейты до единого, вероятность чего оказывается крайне мала. Максимально известная на данный момент вероятность правильного оповещения одного гейта составляет $2/27$ [7].

Рассмотрим данную архитектуру подробнее. На вход гейта подаётся пара фотонов в некотором двухкубитном состоянии и несколько дополнительных фотонов, которые запускаются в дополнительные моды. Состояние, в котором находятся дополнительные фотоны, задано заранее и не зависит от состояния сигнальных фотонов. Сама схема представляет собой некоторым образом расставленные на всех модах оптические элементы с заданными углами. Также к схеме прилагается верное оповещение, ожидаемое на дополнительных модах. На выходе схемы в сигнальных модах ожидается пара фотонов в новом двухкубитном состоянии, получившемся в результате запутывающего преобразования при верном оповещении.

Итак, запутывающее преобразование в линейной квантовой оптике задаётся следующими параметрами.

- Количество дополнительных мод и состояние дополнительных фотонов, подаваемое на эти моды.
- Расстановка светоделителей и фазовых пластин на модах (далее — топология преобразования).
- Углы (параметры) расставленных светоделителей и фазовых пластин.
- Измерение, считаемое за правильное оповещение.

Топология схемы может быть представлена как набор оптических элементов, поставленных последовательно друг за другом, где для каждого из них указаны моды, на которых они действуют (один для фазовой пластины и два для светоделителя). Однако такое представление

оказывается неудобным для использования на практике, а также избыточным. Эквивалентными преобразованиями любая схема может быть приведена к виду последовательности лишь светоделителей, после которых следуют фазовые пластины по одной на каждую моду. В связи с этим для удобства представления топологии схемы рассматривается только расстановка светоделителей, а фазовые пластины в конце преобразования всегда подразумеваются.

Вышеперечисленных параметров достаточно для однозначного определения оптической схемы, среди которых требуется найти такую, которая реализует запутывающий гейт с максимально возможной вероятностью правильного оповещения. Было рассмотрено два подхода для решения этой задачи — применение генетического алгоритма и градиентного спуска. После этого были проанализированы результаты и произведено сравнение подходов.

2.2. Расчёт состояний

В разделе 2.1 была упомянута система из двух мод и одного фотона, которая может находиться в квантовой суперпозиции из двух состояний, тем самым кодируя один кубит. Состояние, в котором находится система, описывается следующим образом.

$$\alpha|0\rangle + \beta|1\rangle,$$

$$\text{где} \quad \alpha, \beta \in \mathbb{C} \quad \text{и} \quad |\alpha|^2 + |\beta|^2 = 1$$

Рассмотрим условия, накладываемые на α и β. Модуль квадрата коэффициента α (или β) равен вероятности измерения системы в состоянии $|0\rangle$ (или $|1\rangle$). Отсюда возникает второе условие, поскольку система может быть измерена либо в состоянии $|0\rangle$, либо в состоянии $|1\rangle$. Комплекснозначность коэффициентов требуется для отражения информации о фазовом сдвиге волновой функции системы. Благодаря наличию фазы у волновой функции возможно такое явление квантовой физики как интерференция, играющее ключевую роль в квантовых алгоритмах.

Предположим теперь, что имеется двухкубитная система. Если два данных кубита находятся в незапутанном состоянии, то вся система описывается как тензорное произведение двух подсистем.

$$(\alpha|0\rangle + \beta|1\rangle) \otimes (\gamma|0\rangle + \delta|1\rangle)$$

Часто для упрощения записи опускается знак тензорного произведения, а также сокращается запись базисных состояний.

$$(\alpha|0\rangle + \beta|1\rangle) \otimes (\gamma|0\rangle + \delta|1\rangle) = (\alpha|0\rangle + \beta|1\rangle)(\gamma|0\rangle + \delta|1\rangle) =$$

$$= \alpha\gamma|0\rangle|0\rangle + \alpha\delta|0\rangle|1\rangle + \beta\gamma|1\rangle|0\rangle + \beta\delta|1\rangle|1\rangle =$$

$$= \alpha\gamma|00\rangle + \alpha\delta|01\rangle + \beta\gamma|10\rangle + \beta\delta|11\rangle$$

Если же двухкубитная система находится в запутанном состоянии, то она не представима в виде тензорного произведения двух однокубитных систем. Например, запутанным является одно из состояний Белла: $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

Таким образом, все возможные двухкубитные состояния (запутанные и незапутанные) имеют следующий вид.

$$\widetilde{\alpha}|00\rangle + \widetilde{\beta}|01\rangle + \widetilde{\gamma}|10\rangle + \widetilde{\delta}|11\rangle$$

Итак, в протоколе KLM система из двух кубит представляет собой четыре моды и два фотона, каждый из которых может находиться только в отведённой для него паре мод. Однако в схеме возможно преобразование, результатом которого будет присутствие обоих фотонов в первой паре мод, тогда как оставшаяся пара мод не будет содержать ни одного фотона. Это состояние не является двухкубитным, следовательно, двухкубитного пространства недостаточно для описания всех возможных состояний системы.

Поэтому будем рассматривать расширенное пространство: для схемы с m модами и p фотонами состояние, в котором в k-той моде находится i_k фотонов, обозначим за $|i_1 \dots i_m\rangle$, причем $\sum_{k=1}^{m} i_k = p$, поскольку утечка и добавление фотонов в процессе преобразования не предусмат-

риваются. На примере, рассмотренном в предыдущем абзаце, базисные двухкубитные состояния примут вид $|1010\rangle$, $|1001\rangle$, $|0110\rangle$ и $|0101\rangle$, а любая их суперпозиция, соответственно, $\widetilde{\alpha}|1010\rangle+\widetilde{\beta}|1001\rangle+\widetilde{\gamma}|0110\rangle+\widetilde{\delta}|0101\rangle$.

Также для расчета выходного состояния требуется переводить состояние системы из вышеописанной нотации (далее — форма Дирака[2]) в нотацию операторов рождения[3] (далее — операторная форма) и наоборот. Для одной моды переход определён в соответствии со следующей формулой ($a^\dagger$ — оператор рождения).

$$a^\dagger|n-1\rangle = \sqrt{n}|n\rangle, \quad n \in \mathbb{N}$$

$$\Rightarrow \left(a^\dagger\right)^n|0\rangle = \sqrt{n!}|n\rangle$$

$$\Rightarrow |n\rangle = \frac{1}{\sqrt{n!}}\left(a^\dagger\right)^n|0\rangle$$

На систему из нескольких мод операция переносится естественным образом. Например, если $a_k^\dagger$ — оператор рождения на k-той моде, то

$$|1352\rangle = \frac{1}{\sqrt{1!}}a_1^\dagger \cdot \frac{1}{\sqrt{3!}}\left(a_2^\dagger\right)^3 \cdot \frac{1}{\sqrt{5!}}\left(a_3^\dagger\right)^5 \cdot \frac{1}{\sqrt{2!}}\left(a_4^\dagger\right)^2 \cdot |0000\rangle$$

Операторная форма представляет собой действие нескольких операторов рождения с определёнными коэффициентами на вакуум (т.е. на состояние вида $|0\dots0\rangle$), который для простоты будет опускаться.

Форма Дирака используется для человекочитаемой записи состояний, а операторная форма — для расчета состояния на выходе схемы.

Так как схема состоит из последовательности светоделителей и фазовых пластин, достаточно определить действие этих двух оптических элементов на состояние системы.

Фазовая пластина с параметром ϕ действует всего на одну моду с соответствующим ей оператором рождения $a^\dagger$ следующим образом.

$$a^\dagger \to e^{i\phi}a^\dagger$$

[2]Статья про нотацию Дирака — https://en.wikipedia.org/wiki/Bra-ket_notation (дата обращения: 15.05.2023)

[3]Статья про операторы рождения и уничтожения в квантовой оптике — https://en.wikipedia.org/wiki/Creation_and_annihilation_operators (дата обращения: 06.05.2023)

Светоделитель с параметрами θ и ϕ действует на две моды с соответствующими им операторами рождения $a^\dagger$ и $b^\dagger$ следующим образом.

$$a^\dagger \to a^\dagger \cos\theta + b^\dagger e^{-i\phi} \sin\theta$$

$$b^\dagger \to -a^\dagger e^{i\phi} \sin\theta + b^\dagger \cos\theta$$

Данные преобразования в силу своей унитарности могут быть представлены в виде матриц $U = \{u_{ij}\}$ так, что преобразование примет вид ($a_k^\dagger$ — оператор рождения на k-той моде, m — количество мод в схеме).

$$a_k^\dagger \to u_{1k}a_1^\dagger + u_{2k}a_2^\dagger + \cdots + u_{mk}a_m^\dagger, \quad k = 1, 2, \ldots, m$$

Таким образом, для подсчёта матрицы полного преобразования сначала составляются матрицы светоделителей и фазовых пластин, которые затем перемножаются.

Рассмотрим теперь расчет состояния на примере схемы, изображённой на рис. 2. Для начала определим исходное состояние — $|112\rangle$ (моды считаются сверху вниз). Произведем перевод этого состояния в операторную форму — $\frac{1}{\sqrt{2}}a_1^\dagger a_2^\dagger \left(a_3^\dagger\right)^2$. Составим матрицы оптических элементов и посчитаем их произведение.

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{i}{\sqrt{2}} & 0 \\ -\frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & e^{i\pi/6} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{i}{\sqrt{2}} & 0 \\ -\frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & e^{i\pi/6} \end{pmatrix}$$

Заменим каждый оператор рождения в исходном состоянии на суперпозицию, в которую он преобразуется.

$$\frac{1}{\sqrt{2}}a_1^\dagger a_2^\dagger \left(a_3^\dagger\right)^2 \to \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}a_1^\dagger - \frac{i}{\sqrt{2}}a_2^\dagger\right)\left(-\frac{i}{\sqrt{2}}a_1^\dagger + \frac{1}{\sqrt{2}}a_2^\dagger\right)\left(e^{i\pi/6}a_3^\dagger\right)^2 =$$

$$= -\frac{i}{2\sqrt{2}}\left(\left(a_1^\dagger\right)^2 + \left(a_2^\dagger\right)^2\right)e^{i\pi/3}\left(a_3^\dagger\right)^2 = \frac{\sqrt{3}-i}{4\sqrt{2}}\left(\left(a_1^\dagger\right)^2 + \left(a_2^\dagger\right)^2\right)\left(a_3^\dagger\right)^2 =$$

$$= \frac{\sqrt{3}-i}{4\sqrt{2}}\left(\left(a_1^\dagger\right)^2\left(a_3^\dagger\right)^2 + \left(a_2^\dagger\right)^2\left(a_3^\dagger\right)^2\right)$$

После преобразования в форму Дирака получим следующее выход-

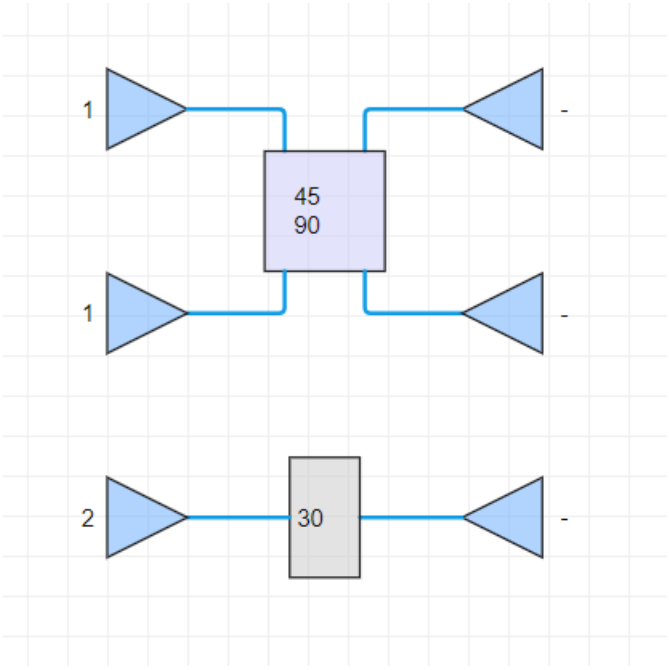


Рис. 2: Пример схемы с тремя модами и четырьмя фотонами. Число около источника фотонов означает количество фотонов, подаваемых в данную моду.

ное состояние.

$$\frac{\sqrt{3}-i}{2\sqrt{2}}(|202\rangle + |022\rangle)$$

2.3. Расчёт верности

Верность преобразования — это мера, отражающая, насколько одно преобразование похоже на другое. Во время поиска гейта верность требуется для оценки того, насколько точно имеющаяся схема реализует заданное искомое преобразование.

Одним из самых простых способов задания верности является матричная норма разницы матриц двух преобразований.

$$F(U,V) = ||U - V||$$

Главной проблемой такого задания является то, что квантовая система не чувствительна к глобальному фазовому сдвигу, то есть сдвиг фазы волновой функции всей системы никак на эту систему не влияет с физической точки зрения. Это означает, что унитарная матрица, по-

лученная из искомой домножением всех её элементов на одно и то же комплексное число, задаёт существенно то же самое преобразование. Верность должна учитывать это, с чем данная мера не справляется.

Другие способы задания верности гейтов основаны на верности, определённой на отдельных состояниях. Мерой схожести (верностью) двух состояний между собой принято считать квадрат их скалярного произведения. Особенностью данной меры является то, что она не является математической метрикой — чем более похожи состояния друг на друга, тем выше значение верности. Наибольшее значение достигается на паре идентичных состояний и равно 1. Наименьшее — на паре ортогональных состояний и равно 0.

$$F(|\phi\rangle, |\psi\rangle) = |\langle\phi|\psi\rangle|^2$$

Перенести эту меру на гейты можно следующим образом.

1. Задать некоторый набор состояний (случайный для каждого вычисления или фиксированный).

2. К каждому из элементов этого набора применить искомое и найденное преобразования, посчитать меру схожести между полученным и ожидаемым состоянием.

3. Взять минимум/медиану/среднее среди посчитанных верностей.

В первоначальной версии кода использовался подобный вариант подсчёта верности гейта, в котором на первом шаге использовался определённый фиксированный набор состояний, а на последнем считался минимум верностей. Первое решение обосновано желанием наличия воспроизводимости предыдущих результатов вычислений для одних и тех же гейтов. Второе решение делает функцию более жёсткой и чувствительной к погрешностям, тем самым повышая её надёжность.

В текущей версии было принято решение использовать другую формулу подсчёта верности по двум причинам. Во-первых, требовалось ускорить вычисления, упростив подсчёт результата. Во-вторых, возникло желание использовать меру, не зависящую от заданного внешне на-

бора состояний, который для некоторых гейтов может, например, оказаться неудачно подобранным. Такая формула верности была найдена в статье [14]. Её идея остаётся такой же как и у вышеописанной, только на третьем шаге вычисляется среднее арифметическое верностей, а на первом рассматривается не конечный набор, а все возможные состояния пространства. Поскольку теперь мы имеем дело с континуальным множеством, формула записывается через интеграл.

$$F(U,V) = \int\limits_{S^{2n-1}} |\langle\psi|V^\dagger U|\psi\rangle|^2 dV =$$

$$= \frac{1}{n(n+1)}\Big(Tr(MM^\dagger) + |Tr(M)|^2\Big), \text{ где } M = V^\dagger U$$

Подробный вывод формулы описан в статье [14]. Записью $M^\dagger$ обозначена эрмитово сопряжённая к M матрица; число n — количество столбцов или строк квадратной матрицы M. Из вида формулы можно заметить, что основная её вычислительная сложность заключается в двух матричных произведениях, а также её результат зависит только от её аргументов.

2.4. Расчёт оповещения

Состояние на выходе схемы, подсчитанное по шагам, изложенным в разделе 2.2, описывает полную суперпозицию расположений фотонов, получившуюся в результате их прохождения через оптические элементы. Однако, если в схеме есть вспомогательные моды, на дополнительных детекторах ожидается измерить паттерн, соответствующий верному оповещению. Только после этого преобразование считается удачно сработавшим. Для того, чтобы расчитать преобразование, реализуемое схемой, и вероятность срабатывания гейта, нужно вычислить состояние в сигнальных модах, оказавшееся в результате коллапса волновой функции при верном оповещении. Это происходит путём редуцирования всего пространства на соответствующее подпространство.

Рассмотрим расчёт редуцированного состояния на примере. Предпо-

ложим, что в результате преобразования в выходных модах оказалось следующее состояние.

$$\frac{1}{\sqrt{5}}\Big(|010101\rangle - |010100\rangle + i|100101\rangle + |001110\rangle - |001101\rangle\Big)$$

Пусть верным оповещением является измерение ровно одного фотона в последней моде и ни одного в предпоследней. В этом случае удалим все состояния из суперпозиции кроме тех, что оканчиваются на $|01\rangle$. Получим следующую суперпозицию.

$$|s\rangle = \frac{1}{\sqrt{5}}\Big(|010101\rangle + i|100101\rangle - |001101\rangle\Big) =$$

$$= \frac{1}{\sqrt{5}}\Big(|0101\rangle + i|1001\rangle - |0011\rangle\Big)|01\rangle$$

Вероятность верного оповещения равна сумме квадратов модулей коэффициентов при базисных векторах. На языке векторных операций эта вероятность выражается через скалярный квадрат полученного вектора.

$$\langle s|s\rangle = \Big|\frac{1}{\sqrt{5}}\Big|^2 + \Big|\frac{i}{\sqrt{5}}\Big|^2 + \Big|-\frac{1}{\sqrt{5}}\Big|^2 = \frac{3}{5}$$

В зависимости от изначального состояния посчитанная вероятность может принимать различные значения. Аналогично расчёту верности, описанному в разделе 2.3, было принято решение вычислять среднюю вероятность по всему пространству. Формула вероятности верного оповещения принимает следующий вид (средний скалярный квадрат вектора, полученного после преобразования).

$$P(U) = \int\limits_{S^{2n-1}} \langle\psi|U^\dagger U|\psi\rangle dV$$

Докажем следующее утверждение подобно тому, как это было сделано в статье [14] для формулы верности.

$$\int\limits_{S^{2n-1}} \langle\psi|M|\psi\rangle dV = \frac{1}{n}Tr(M)$$

Пусть M — эрмитова матрица ($M^\dagger = M$). Тогда существует A — унитарная матрица, такая что $M = A^\dagger \Lambda A$, где

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{pmatrix}, \quad \lambda_k \in \mathbb{R}$$

Тогда

$$\int\limits_{S^{2n-1}} \langle\psi|M|\psi\rangle dV = \int\limits_{S^{2n-1}} \langle\psi|A^\dagger \Lambda A|\psi\rangle dV = \int\limits_{S^{2n-1}} \langle\phi|\Lambda|\phi\rangle dV$$

Подынтегральное выражение приняло вид взвешенной суммы чисел λ_k. Разложим полученный интеграл на сумму интегралов и вынесем за знаки интегралов числа λ_k.

$$\int\limits_{S^{2n-1}} \langle\phi|\Lambda|\phi\rangle dV = \sum_{k=1}^{n} a_k \lambda_k$$

Значение вычисляемого интеграла не зависит от перестановки λ_k ввиду того, что матрицы, переставляющие строки или столбцы местами, являются унитарными. Таким образом, все a_k равны одному и тому же значению. Обозначим его за a.

$$\sum_{k=1}^{n} a_k \lambda_k = a \sum_{k=1}^{n} \lambda_k = a \cdot Tr(\Lambda) = a \cdot Tr(A^\dagger \Lambda A) = a \cdot Tr(M)$$

Число a не зависит от матрицы M. Найдём его, подставив в формулу вместо M единичную матрицу.

$$\int\limits_{S^{2n-1}} \langle\psi|I_n|\psi\rangle dV = a \cdot Tr(I_n) \Rightarrow 1 = a \cdot n \Rightarrow a = \frac{1}{n}$$

Утверждение доказано для случая, когда матрица M — эрмитова.

Предположим теперь, что M — антиэрмитова ($M^\dagger = -M$). В этом случае эрмитовой является матрица iM, поскольку $(iM)^\dagger = (-i)M^\dagger = (-i)(-M) = iM$.

Тогда

$$\int\limits_{S^{2n-1}} \langle\psi|M|\psi\rangle dV = \frac{1}{i} \int\limits_{S^{2n-1}} \langle\psi|iM|\psi\rangle dV = \frac{1}{i} \cdot \frac{1}{n} \cdot Tr(iM) =$$

$$= \frac{1}{n} \cdot Tr\left(\frac{i}{i}M\right) = \frac{1}{n} \cdot Tr(M)$$

Утверждение доказано для случая, когда M — антиэрмитова матрица.

Рассмотрим произвольную матрицу M.

$$M = \frac{M + M^\dagger}{2} + \frac{M - M^\dagger}{2}$$

Нетрудно заметить, что $S = \frac{M+M^\dagger}{2}$ — эрмитова, $K = \frac{M-M^\dagger}{2}$ — антиэрмитова.

$$\int\limits_{S^{2n-1}} \langle\psi|M|\psi\rangle dV = \int\limits_{S^{2n-1}} \langle\psi|S|\psi\rangle + \langle\psi|K|\psi\rangle dV =$$

$$= \frac{1}{n} \cdot Tr(S) + \frac{1}{n} \cdot Tr(K) = \frac{1}{n} \cdot Tr(S + K) = \frac{1}{n} \cdot Tr(M)$$

Утверждение доказано для произвольной матрицы M.

Таким образом, формула расчёта средней вероятности верного оповещения принимает следующий вид.

$$P(U) = \frac{1}{n} \cdot Tr(U^\dagger U)$$

Поскольку от произведения матриц $U^\dagger$ и U сразу берётся след, полностью считать это произведение не нужно. Для вычисления следа достаточно просуммировать все элементы матрицы, полученной в результате поэлементного умножения $U^\dagger$ на U, что и было использовано в реализации.

2.5. Генетический алгоритм

Генетический, или эволюционный, алгоритм является эвристическим подходом к задаче глобальной оптимизации. Он вдохновлён теорией эволюции, которая подробно и достаточно убедительно объясняет процесс приспособления живых организмов к окружающей среде в природе. Упрощенно идея алгоритма такова.

1. Инициализация. Породить набор случайных особей — элементов пространства поиска (первое поколение).

2. Селекция. Оценить каждую особь в поколении (в соответствии с некоторой функцией приспособленности) и выбрать подмножество наилучших особей.

3. Скрещивание. Породить потомство (новое поколение) для выбранных особей с помощью случайной рекомбинации их генетического материала.

4. Мутации. Применить случайные изменения к особям нового поколения.

5. Если лучшая особь недостаточно приспособлена, перейти к шагу 2 с новым поколением.

Эта идея применима и к задаче поиска запутывающего преобразования в квантовой линейной оптике: в роли особи выступает оптическая схема гейта, устройство которой подробно описано в разделе 2.1.

Итак, каждая схема представляется набором чисел — геномом особи. Геном состоит из нескольких частей, несущих информацию о различных параметрах схемы.

- Номера мод для светоделителей: пары целых чисел, кодирующие входные моды для каждого светоделителя.

- Углы (параметры) светоделителей, по паре углов на каждый.

- Углы (параметры) фазовых пластин, по одному на каждую.

- Состояние вспомогательных фотонов, подаваемое на вход: номера дополнительных мод, куда запускаются фотоны, по одному на каждый вспомогательный фотон.

- Паттерн, считаемый за верное оповещение: номера дополнительных мод, где ожидается измерить фотоны, по одному на каждый вспомогательный фотон.

Каждый запуск алгоритма может быть настроен с помощью следующих параметров.

- Количество родителей — число наиболее приспособленных особей, которые переходят в следующее поколение.

- Количество потомков — число новых особей, которые создаются на этапе скрещивания из родительских генов.

- Интенсивность мутаций — вероятность гена изменить своё значение на некоторую небольшую величину.

- Глубина (сложность) схемы — количество оптических элементов в каждой схеме.

- Количество вспомогательных мод.

- Количество вспомогательных фотонов.

Значения количества родителей и количества потомков влияют только на разнообразие популяции. Общее число особей в популяции (то есть сумма этих параметров), равное 10-15 тысячам, оказывалось достаточным для успешной работы алгоритма. Более актуальным показателем является доля родителей в популяции. Он был подобран за несколько запусков и оказался равен 40%.

Слишком большая интенсивность мутаций ухудшает работу алгоритма, поскольку в таком случае в каждом поколении многие гены изменяются случайным образом, что делает метод очень похожим на полный перебор, замедляя его работу. Нулевая интенсивность мутаций,

наоборот, не добавляет в имеющуюся популяцию новых генов, отчего алгоритм ищет нужную особь, которая может быть получена только рекомбинацией генов имеющихся особей. Интенсивность, равная 10%, показала оптимальную сходимость метода.

Остальные параметры должны подбираться индивидуально для каждой задачи поиска.

Говоря о глобальных шагах генетического алгоритма, стоит остановиться на селекции и скрещивании. Селекция происходит следующим образом: для каждой особи (оптической схемы) вычисляется значение некоторой функции приспособленности, после чего особи упорядочиваются по этим значениям и из популяции удаляются схемы с самыми низкими значениями.

Для того чтобы судить о приспособленности схемы, оказывается недостаточно одного параметра. Во-первых, требуется верность преобразования — насколько похоже преобразование, реализуемое схемой, на искомое (см. раздел 2.3). Во-вторых, требуется вероятность правильного оповещения (см. раздел 2.4). Если схема обладает низкой верностью, она не реализует желаемое преобразование, так что функция приспособленности должна возвращать низкие значения для таких схем. С другой стороны, при высокой верности вероятность правильного оповещения может быть очень мала, и тогда такой гейт должен иметь небольшое значение функции приспособленности, поскольку всё ещё не представляет интереса для научного сообщества ввиду существования схем с бóльшими вероятностями (до $2/27$).

В данной связи было принято решение определить функцию приспособленности следующим образом.

```
def fitness(p, f):
  if f < f_min:
        return f
  return 1 + p
```

Аргументы функции p и f — соответственно вероятность верного оповещения и верность преобразования. Параметр f_min настраивае-

мый и означает такое значение верности, при котором гейт считается исправно выполняющим искомое преобразование с пренебрежимой погрешностью. Он был взят за 0.999.

Из определения функции можно заметить, что особи на первых поколениях соревнуются по верности, после чего со временем появляются схемы с высокими верностями, которые начинают соревноваться уже между собой по вероятности.

Для реализации скрещивания были рассмотрены два подхода: потомок получает случайные гены от отца, а остальные — от матери; потомок получает левую часть схемы от отца, а правую — от матери. На практике явного лидера среди этих двух подходов не нашлось. В окончательную версию был включён второй вариант как более естественный.

2.5.1. Оптимизации

В первоначальной реализации генетического алгоритма расчёт преобразования (см. раздел 2.2) происходил посредством символьной алгебры с использованием библиотеки sympy[4]. Состояние хранилось в виде символьной записи, что позволяло экономить много памяти, поскольку размерность пространства состояний преобразования на m модах с p фотонами равна $C^p_{m+p-1} = \frac{(m+p-1)!}{p!(m-1)!}$, и рассматриваемый вектор зачастую оказывается разреженным.

Однако профилирование показало, что вызовы функций библиотеки sympy отнимают большое количество времени, из-за чего было решено сменить математическое представление состояний и механизм расчёта преобразований. Было предложено использовать векторную алгебру: представлять состояние в виде вектор-столбца и выражать действия над ним через матричные операции. Для этого подошла библиотека PyTorch[5]. Кроме того, PyTorch предоставляет возможность выполнять вычисления на графическом процессоре, что позволяет использовать распараллеливание для ускорения расчётов. Стоит отметить, что за

[4]Сайт с описанием библиотеки sympy — https://www.sympy.org/en/index.html (дата обращения: 15.05.2023)

[5]Официальный сайт PyTorch — https://pytorch.org (дата обращения: 15.05.2023)

эти оптимизации пришлось поплатиться: поскольку векторы обладают большой размерностью, алгоритм использует много памяти в процессе вычислений.

Помимо этого были изменены формулы верности и вероятности гейта, что подробно описано в разделах 2.3 и 2.4. Это также дало прирост в скорости расчётов.

Достигнутые результаты см. в разделе 3.1.

2.6. Градиентный спуск

Градиентный спуск повзоляет находить точку экстремума дифференцируемой функции. Если имеется набор параметров и определённая на них функция, называемая в таких случаях функцией потерь и возвращающая оценку, насколько эти параметры подходят для решения рассматриваемой задачи, то при определённых условиях в таком случае применим градиентный спуск. В результате работы алгоритма будет найден локально наиболее оптимальный набор параметров.

Условия применимости градиентного спуска заключаются в дифференцируемости функции потерь, из чего возникает требование наличия только вещественных параметров. Препятствием к применению этого метода выступает то, что многие параметры, задающие оптическую схему, целочисленны.

Однако это препятствие удалось обойти путём избавления от таких параметров как топология схемы и верное оповещение. Немногие оставшиеся целочисленные параметры были вынесены в так называемые гиперпараметры, то есть настраиваемые вручную, перед запуском алгоритма поиска.

Оказалось, что существуют топологии схем, в которых возможно задать любое унитарное преобразование, меняя только углы закреплённых на своих местах оптических элементов [2, 16]. Назовём такие топологии универсальными. Таким образом, если зафиксировать некоторую универсальную топологию схемы, пространство поиска нисколько не сузится. В качестве универсальных топологий рассматривались так

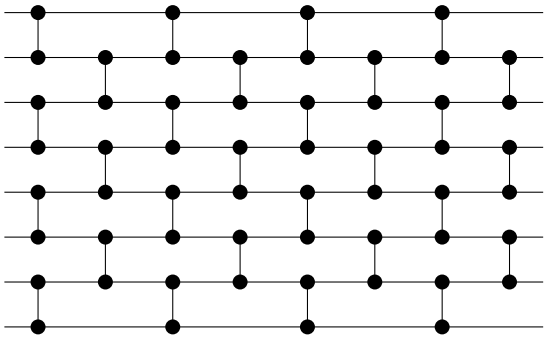
называемые схемы Река, Клементса, а также ещё две новых, предложенных вместо них (рис. 3).

Использование топологий Река и Клементса не привело к сходимости метода к оптимальным значениям верности и вероятности. Это связано с тем, что данные топологии хорошо подходят для изготовления и эксплуатации интерферометра на практике, но не для задачи вычисления функции потерь. Из рисунка 3 видно, что в этих топологиях светоделители сильно связаны друг с другом — чтобы заставить провзаимодействовать, например, первую и последнюю моды, нужно поменять углы в большом количестве светоделителей. Такие зависимости между оптическими элементами порождают лишние паразитные локальные максимумы в функции потерь.

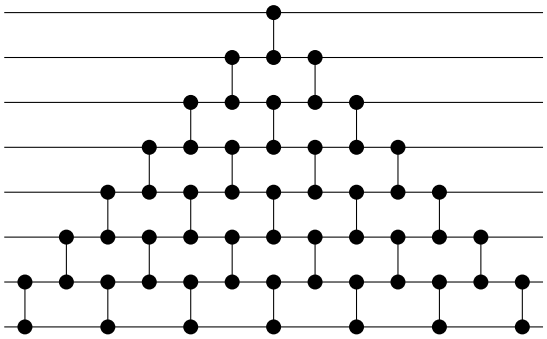
От сильной связи между светоделителями можно избавиться благодаря использованию универсальных топологий, в которых для каждой пары мод существует ровно один светоделитель, входами и выходами которого они являются. В этом случае за взаимодействие отдельной пары мод отвечает всего один оптический элемент. На основе этой идеи были составлены две топологии, названные соответственно устойчивой и параллельной (рис. 3c и 3d).

Устойчивая топология спроектирована с идеей упростить подбор такого параметра как количество вспомогательных мод. Она обладает следующим свойством: любое подмножество мод со светоделителями, установленными только на моды из этого подмножества, образуют схему с точно такой же топологией. Отсюда, если установить количество дополнительных мод в относительно большое число, может найтись схема, для реализации которой достаточно меньшего количества вспомогательных мод. Светоделители, осуществляющие взаимодействие нужных мод с лишними, вырождаются в идеальные зеркала, то есть в отсутствие каких-либо преобразований. В схеме, найденной в результате работы алгоритма, оказывается нетрудно убрать лишние неиспользуемые моды.

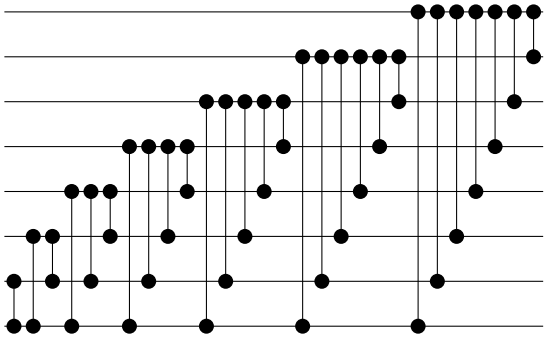
Параллельная топология спроектирована по принципу ”разделяй и властвуй” — схема делится на две равные, для каждой из которых па-



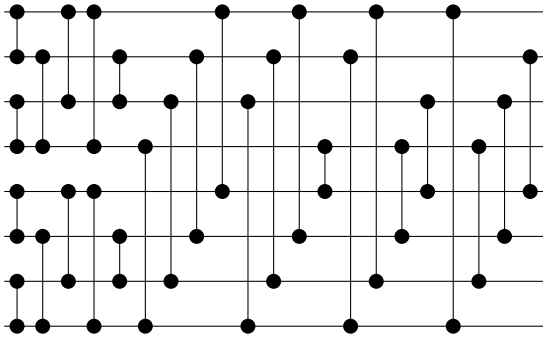
(a) Топология Клементса



(b) Топология Река



(c) Устойчивая топология



(d) Параллельная топология

Рис. 3: Рассматриваемые универсальные топологии схем для восьми мод. Горизонтальными линиями обозначены моды. Каждый вертикальный отрезок представляет собой светоделитель, действующий на модах, отмеченных жирными точками.

раллельно применяется преобразование, после чего все моды первой части последовательно взаимодействуют со всеми модами второй. Для построения топологии подсхем принцип сохраняется и уходит в глубину до тех пор, пока рассматриваемая подсхема не будет состоять из двух мод, топология преобразования для которой очевидна и единственна (состоит из единственного светоделителя). Параллельная топология обладает следующим свойством: луч света проходит через одинаковое количество светоделителей вне зависимости от того, в какую входную моду он был запущен. Это свойство делает функцию потерь максимально сбалансированной относительно того, какой вклад вносят её аргументы при вычислении её значения, что ускоряет работу алгоритма.

После нескольких запусков функция потерь была подобрана следующим образом. Параметр p_min означает минимальную приемлемую вероятность работы гейта.

```
def loss(p, f):
  if p < p_min:
        return p
  return f
```

Таким образом, в процессе работы алгоритма в первую очередь настраивается наиболее приемлемая вероятность работы гейта, после чего параметры обновляются в направлении повышения верности гейта. Также можно заметить, что алгоритм максимизирует функцию потерь, в отличие от классического подхода, в котором она минимизируется. Концептуально это изменение ни на что не влияет.

Вместо подсчёта функции потерь для конкретного оповещения, трактуемого как верное, как это было в генетическом алгоритме, теперь эта функция считается для любого возможного оповещения, среди которых потом выбирается то, что с наилучшим значением. Выбранное оповещение трактуется далее как верное.

3. Результаты

В данном разделе представлены результаты применения рассматриваемых методов поиска и сделанные на их основе выводы. Все эксперименты проводились в среде выполнения Google Colab[6] с поддержкой графического процессора.

В качестве запутывающего гейта был выбран CZ[7], поскольку он является одним из стандартных двухкубитных гейтов. Также он изменяет фазу волновой функции пары фотонов, не меняя друг с другом сами состояния, что проще осуществимо с помощью линейной оптики. На рис. 4a изображена лучшая известная на данный момент схема такого преобразования, описанная в статье Эмануэля Книлла [7]. На рис. 4b представлена схема, найденная генетическим алгоритмом и приведённая к более простому виду в результате анализа.

3.1. Генетический алгоритм

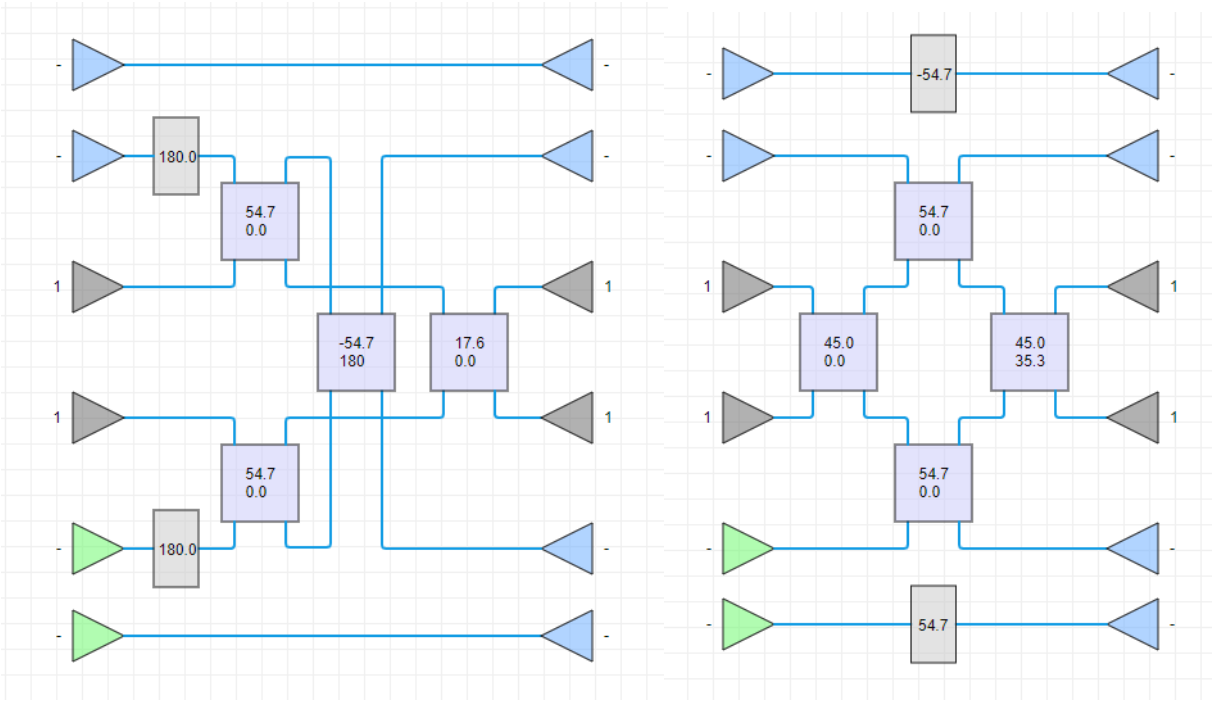
В результате оптимизации имеющегося кода было достигнуто сильное ускорение: алгоритм стал находить за несколько секунд тот гейт, который раньше находил за несколько дней. Это позволило продвинуть исследования дальше, предоставив возможность подбирать гиперпараметры и запускать поиск на более продолжительное время с расширенным пространством поиска.

Запуск генетического алгоритма производился 10 раз с различными случайно сгенерированными начальными популяциями (рис. 5) и со следующими параметрами.

- Количество родителей: 4000.
- Количество потомков: 6000.
- Количество поколений: 1000.

[6]Стартовая страница с описанием среды выполнения Google Colab — https://colab.research.google.com (дата обращения: 15.05.2023)

[7]Статья про квантовые гейты, в том числе гейт CZ (Controlled Z) — https://en.wikipedia.org/wiki/Quantum_logic_gate (дата обращения: 15.05.2023)



(a) Предложенная Эмануэлем Книллом. (b) Найденная генетическим алгоритмом.

Рис. 4: Эквивалентные схемы гейтов CZ с вероятностью верного оповещения 2/27. Парой голубых треугольников в левой части схемы обозначены моды первого кубита, парой зелёных — моды второго. Серыми треугольниками обозначены входы и выходы вспомогательных мод. Число около каждого из треугольников слева означает количество фотонов, подаваемое на данную моду. Число справа означает количество фотонов в данной моде, которое ожидается измерить для верного оповещения. На рис. 4b приведена схема, полученная после упрощения вручную результата генетического алгоритма.

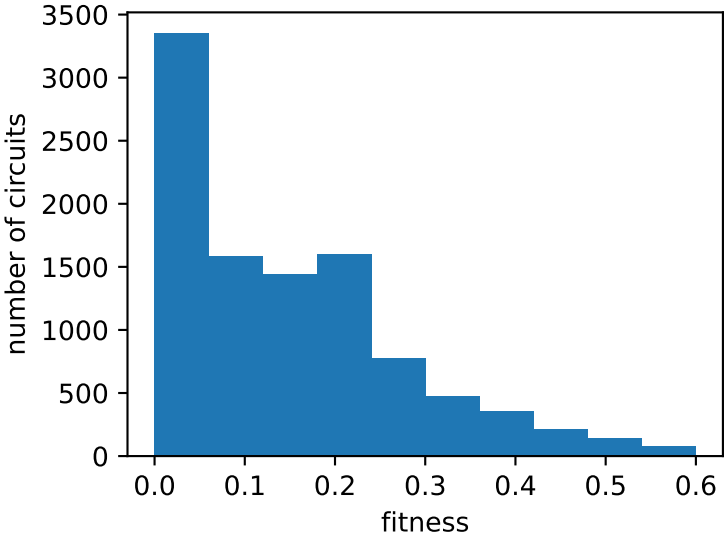


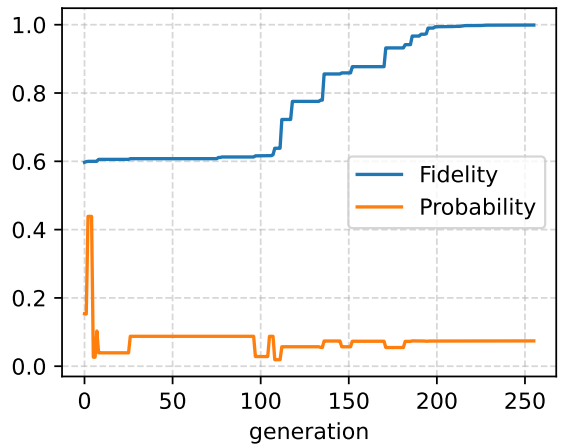
Рис. 5: Распределение случайно сгенерированных особей по значению функции приспособленности в рамках одной популяции.

- Вероятность мутаций: 10%.
- Приемлемая вероятность верного оповещения: 2/27.
- Глубина схемы: 4.
- Количество вспомогательных мод: 2.
- Количество вспомогательных фотонов: 2.

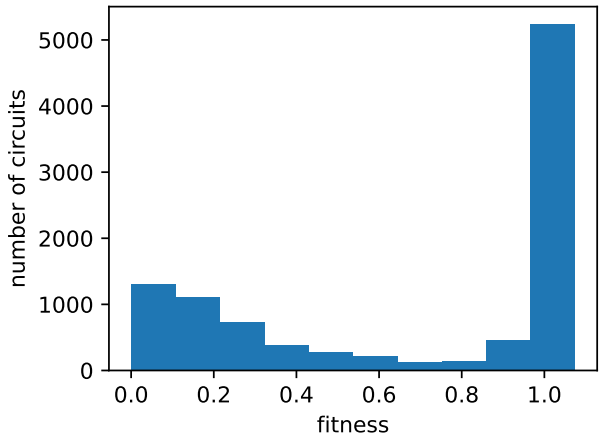
В 60% запусков алгоритм успешно находил схему, эквивалентную наилучшей известной научному сообществу на данный момент, менее чем за 500 поколений (в среднем, за 315 поколений). В остальных запусках алгоритм находил неоптимальный локальный экстремум. На рис. 6 приведены данные о работе одного успешного запуска. На рис. 7 изображена схема в том виде, в котором её нашёл алгоритм.

3.2. Градиентный спуск

Запуск градиентного спуска производился по 5 раз для каждой топологии из приведённых на рисунке 3 и описанных подробно в разделе 2.6 со следующими параметрами.



(a) Характеристики лучшей особи.



(b) Распределение особей.

Рис. 6: Результат работы генетического алгоритма. На рисунке (a) приведена динамика изменения характеристик наиболее приспособленной особи для примера успешного нахождения схемы за 255 поколений. На рисунке (b) изображено распределение особей по функции приспособленности в окончательной популяции.

- Количество градиентных шагов: 1000.
- Приемлемая вероятность верного оповещения: 2/27.
- Количество вспомогательных мод: 2.
- Количество вспомогательных фотонов: 2.

На рисунке 8 отражены результаты запусков для каждой топологии. Использование топологий Река и Клементса не привело к сходимости метода, в отличие от двух предложенных топологий. Также, использование параллельной топологии вместо устойчивой сокращает количество градиентных шагов, требующихся для нахождения локального экстремума (в среднем, с 812 до 517). На рис. 9 приведена схема для параллельной топологии в том виде, в котором её нашёл алгоритм.

3.3. Дополнительные результаты

Реализованное решение способно искать не только запутывающие преобразования, но и произвольные, действующие на заданный набор

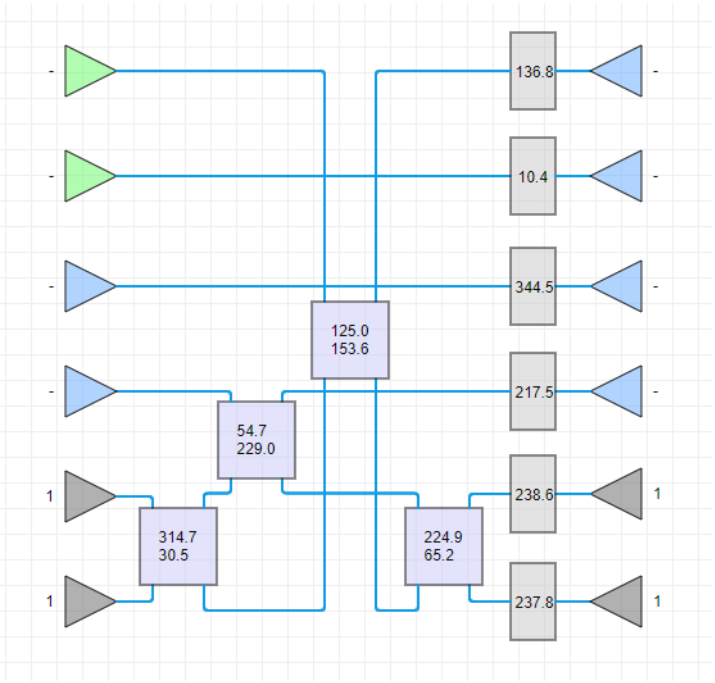


Рис. 7: Найденная схема в результате успешного запуска генетического алгоритма.

исходных состояний. В частности, может быть найдена схема, генерирующая состояния Белла[8]. Генерация таких состояний также является важной задачей в квантовой линейной оптике [8].

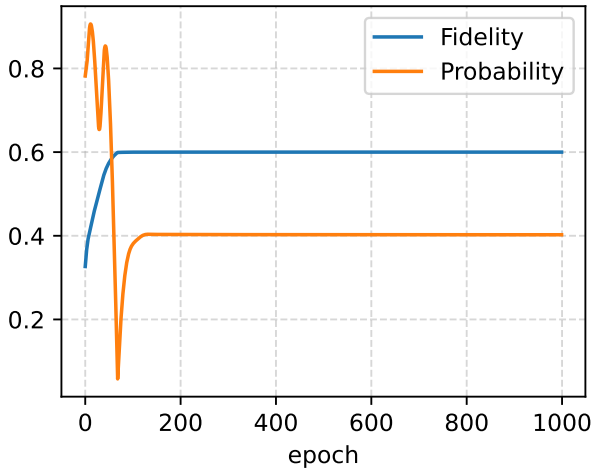
3.4. Выводы

В результате анализа экспериментов были сделаны следующие выводы.

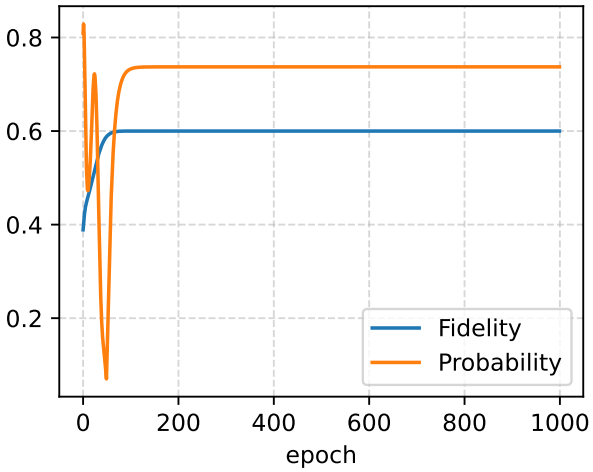
- Градиентный спуск работает быстрее, чем генетический алгоритм. При этом использование параллельной топологии вместо устойчивой ещё больше ускоряет поиск.

- Генетический алгоритм позволяет найти оптимальную топологию схемы (то есть состоящую из наименьшего количества оптических элементов), тогда как топология в градиентном спуске всегда фиксирована.

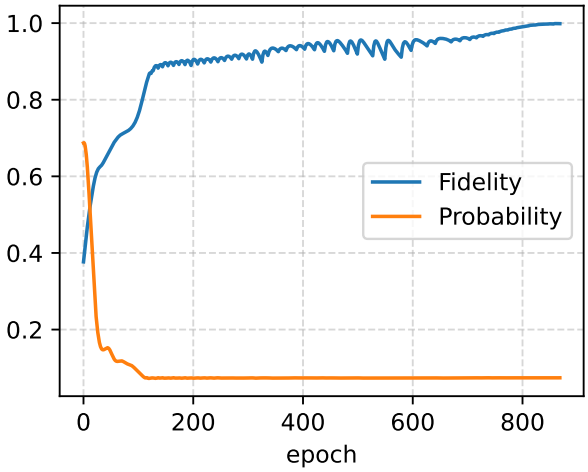
[8]Статья про состояния Белла — https://en.wikipedia.org/wiki/Bell_state (дата обращения: 15.05.2023)



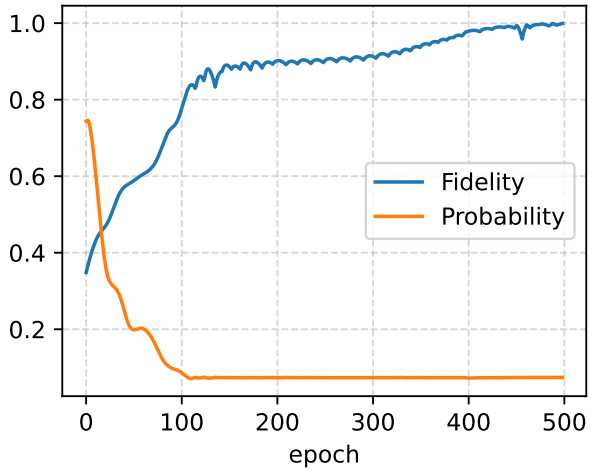
(a) Топология Клементса



(b) Топология Река



(c) Устойчивая топология.



(d) Параллельная топология.

Рис. 8: Результаты работы градиентного спуска для различных топологий схемы. Для топологий Клементса (a) и Река (b) алгоритм останавливается в неоптимальных локальных экстремумах. Для устойчивой топологии (c) приведён пример нахождения схемы за 870 градиентных шагов, для параллельной топологии (d) — за 500 шагов.

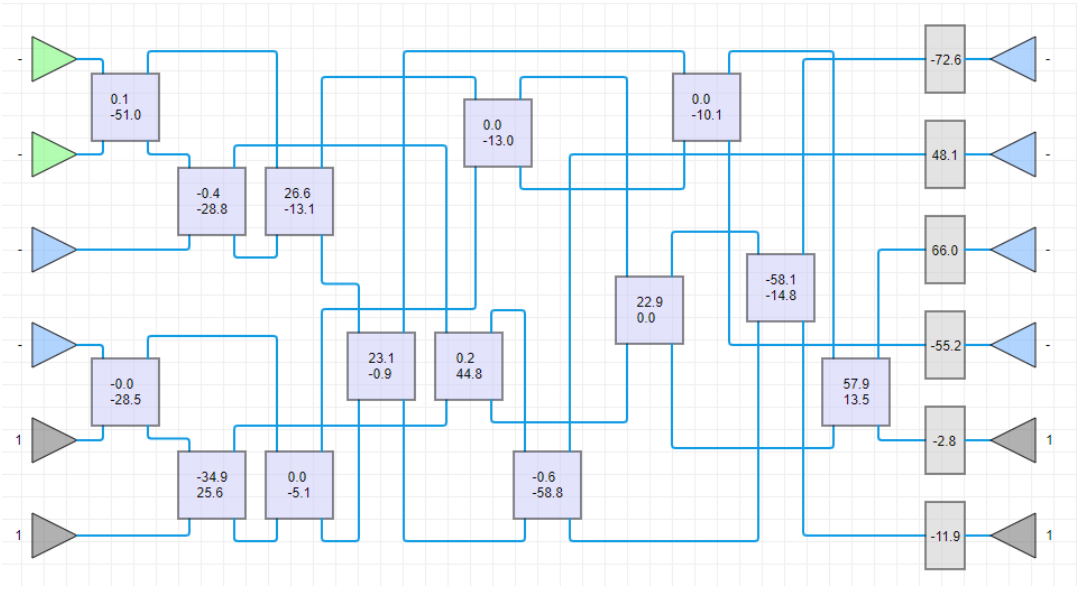


Рис. 9: Найденная схема в результате успешного запуска градиентного спуска.

- Генетический алгоритм является методом глобальной оптимизации в отличие от градиентного спуска, что делает предположение о несуществовании лучших схем, чем найденные им, заслуживающим больше доверия.

- Оба метода применимы к задаче поиска запутывающего преобразования в линейной квантовой оптике. С помощью каждого из методов независимо была найдена лучшая известная на данный момент схема гейта CZ с вероятностью верного оповещения $2/27$.

Таким образом, каждый из методов обладает своими достоинствами и недостатками. Для оптимального поиска запутывающих преобразований целесообразно комбинировать друг с другом оба подхода.

Тот факт, что не нашлось схемы лучше, чем уже известная научному сообществу, навевает на мысль о несуществовании в пространстве поиска схемы с большей вероятностью верного оповещения, чем $2/27$. Однако это пространство может быть расширено, например, допущением нескольких верных оповещений и корректировкой состояния на сигнальных модах в зависимости от состояния, в котором были измерены вспомогательные фотоны. В данной работе не рассматривались

подобные расширения пространства поиска, поэтому эта идея может быть включена в планы по продолжению работы.

Заключение

В ходе выполнения данной работы были достигнуты следующие результаты.

- Оптимизирован существующий генетический алгоритм поиска.
- Реализован алгоритм поиска с помощью градиентного спуска.
- Произведено сравнение двух подходов и сделаны выводы.
- Расширены возможности поискового фреймворка.

Код доступен в репозитории[9], размещённом на веб-сервисе GitHub.

По данной работе была подготовлена и отправлена на рецензирование совместная публикация “Heralded gate search with genetic algorithms for quantum computation by A. Chernikov, S. S. Sysoev, E. A. Vashukevich, et al.” в журнал Physical Review A[10].

[9]Репозиторий проекта — https://github.com/sysoevss/galopy (дата обращения: 15.05.2023), пользователь artemgl

[10]Сайт журнала Physical Review A — https://journals.aps.org/pra/ (дата обращения: 15.05.2023)

Список литературы

[1] Bernstein E. Vazirani U. Quantum complexity theory. — 1993. — Access mode: https://dl.acm.org/doi/pdf/10.1145/167088.167097 (online; accessed: 15.05.2023).

[2] Clements W.R. et al. Optimal design for universal multiport interferometers. — 2016. — Access mode: https://opg.optica.org/optica/fulltext.cfm?uri=optica-3-12-1460 (online; accessed: 15.05.2023).

[3] D.P. DiVincenzo. Two-bit gates are universal for quantum computation. — 1995. — Access mode: https://arxiv.org/pdf/cond-mat/9407022.pdf (online; accessed: 15.05.2023).

[4] D.P. DiVincenzo. The physical implementation of quantum computation. — 2000. — Access mode: https://arxiv.org/pdf/quant-ph/0002077.pdf (online; accessed: 15.05.2023).

[5] Deutsch D. Jozsa R. Rapid solution of problems by quantum computation. — 1992. — Access mode: https://www.isical.ac.in/~rcbose/internship/lectures2016/rt08deutschjozsa.pdf (online; accessed: 15.05.2023).

[6] Deutsch D.E. Barenco A. Ekert A. Universality in quantum computation. — 1995. — Access mode: https://arxiv.org/pdf/quant-ph/9505018.pdf (online; accessed: 15.05.2023).

[7] E. Knill. Quantum gates using linear optics and postselection. — 2002. — Access mode: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.66.052306 (online; accessed: 15.05.2023).

[8] Fldzhyan S.A. Saygin M.Y. Kulik S.P. Compact linear optical scheme for Bell state generation. — 2021. — Access mode: https://journals.aps.org/prresearch/pdf/10.1103/PhysRevResearch.3.043031 (online; accessed: 15.05.2023).

[9] Knill E. Laflamme R. Milburn G.J. A scheme for efficient quantum computation with linear optics. — 2001. — Access mode: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=7955e6b51070395b97b70ddb8d7e2ae797528a81 (online; accessed: 15.05.2023).

[10] L.K. Grover. A fast quantum mechanical algorithm for database search. — 1996. — Access mode: https://dl.acm.org/doi/pdf/10.1145/237814.237866 (online; accessed: 15.05.2023).

[11] O'Brien J.L. et al. Demonstration of an all-optical quantum controlled-NOT gate. — 2003. — Access mode: https://arxiv.org/pdf/quant-ph/0403062.pdf (online; accessed: 15.05.2023).

[12] P. Benioff. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. — 1980. — Access mode: https://link.springer.com/article/10.1007/BF01011339 (online; accessed: 15.05.2023).

[13] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. — 1994. — Access mode: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2273d9829cdf7fc9d3be3cbecb961c7a6e4a34ea (online; accessed: 15.05.2023).

[14] Pedersen L.H. Møller N.M. Mølmer K. Fidelity of quantum operations. — 2007. — Access mode: https://arxiv.org/pdf/quant-ph/0701138.pdf (online; accessed: 15.05.2023).

[15] R.P. Feynman. Simulating physics with computers. — 1982.

[16] Reck M. et al. Experimental realization of any discrete unitary operator. — 1994. — Access mode: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.73.58 (online; accessed: 15.05.2023).

[17] Ю.И. Манин. Вычислимое и невычислимое. — 1980.