arXiv:2302.02278v2 [quant-ph] 1 Feb 2024

作为量子性能基准的优化应用

Thomas Lubinski，[1,2] 卡尔顿大学，[3] Catherine McGeoch，[4]Pratik Sathe，[5,6,7]Joshua Apanavicius，[8,9]和 David E. Bernal Neira[6,10,11]（量子经济发展联盟 (QED-C) 合作）[*][1]*Quantum Circuits Inc，25 Science Park*，纽黑文，*CT 06511*[2]*QED-C*标准与性能基准技术咨询委员会[3]先进网络科学倡议，洛斯阿拉莫斯国家实验室，美国新墨西哥州洛斯阿拉莫斯，邮编*87545*[4]*D-Wave Systems*，加拿大不列颠哥伦比亚省本拿比，*V5G 4M9*[5]加州大学洛杉矶分校物理与天文学系，美国[6]高级计算机科学研究所，大学空间研究协会，美国加利福尼亚州芒廷维尤[7]洛斯阿拉莫斯国家实验室理论部 *(T4)*，美国新墨西哥州洛斯阿拉莫斯，邮编*87545*[8]印第安纳大学物理系，美国印第安纳州布卢明顿，邮编*47405*[9]印第安纳大学量子科学与工程中心，美国印第安纳州布卢明顿，邮编*47405*[10]量子人工智能实验室，*NASA*艾姆斯研究中心，美国加利福尼亚州芒廷维尤[11]戴维森化学工程学院，普渡大学，美国印第安纳州西拉法叶

（日期：2024年2月5日）

组合优化预计将在未来几年成为量子计算的主要应用领域之一。量子近似优化算法（QAOA）和量子退火（QA）有望展现出显著优于当前最先进解法的运行时性能。受用于表征经典优化算法的现有方法启发，我们分析了使用门模型量子设备和量子退火设备求解最大割问题所得解的质量。我们以此指导开发一种先进的量子计算机基准测试框架，用于评估迭代式量子-经典混合应用中运行时执行性能与解的质量之间的权衡。该框架通过直观易懂的可视化生成性能剖面图，展示不同问题规模下性能随时间的变化，并揭示基准测试方法所发现的算法局限性。作为示例，我们探究影响量子计算系统吞吐量的因素，并采用在各种量子模拟器和量子硬件系统上执行所得的结果进行分析。

目录

I. 引言 2

II. 背景 4
A. 最大割优化问题 4 B. 优化问题的量子算法 4 C. 量子计算机基准测试 4 D. 优化任务的量子基准 5
6

III. QED-C基准框架增强 7
A. QAOA 基准测试算法 7 B. QA基准算法 8

IV. 最大割基准指标分析 8
A. 特定应用结果质量指标 8 B. 结果质量与执行时间 10 C. 影响 QAOA 拟设态保真度的因素 11

V. 在量子硬件上执行 12
A. 离子阱系统 13 B. 超导 Transmon 系统 14 C. 量子退火系统 15 D. 硬件结果讨论 15

VI. 总结与结论 16

代码可用性 17 致谢 17
参考文献 17

A. 组合优化方法 19
1. 组合优化理论 24

B. 优化问题的量子启发式算法 24
1. 量子近似优化算法 25 2. 量子退火 25

C. 问题与实现细节 26

D. 执行时间分析 27

E. 结果质量与超参数 27
1. 初始角度和重启次数 28 2. 固定角度猜想 28 3. 参数选择策略 29

* 本研究由量子经济发展联盟（QED-C）（quantumconsortium.org）资助，该联盟由 SRI International 管理。本研究在 QED-C标准与性能基准技术咨询委员会的指导下开展。作者感谢多位委员会成员为本项目及本文提供意见和反馈。

一、引言

在许多应用领域，高效地求得涉及大量变量的问题的近似最优解至关重要，这些变量会影响某种操作或函数的成本。例如，在大型电网中，迅速确定电力分配的最佳方案可以防止大规模停电。这类问题称为组合优化问题，常被视为量子计算的一种潜在应用场景 [1–3]。

用于解决此类问题的经典计算机算法已相当成熟，并广泛应用于工业、政府和学术领域。它们在优化资源利用和降低成本方面发挥着关键作用。组合优化应用通常需要在严格的资源约束（例如时间、内存、能量或资金）下执行，因此，人们尤其重视量化有限预算内能够获得的结果质量。用于衡量和比较不同解法性能的标准技术已趋于成熟，并得到广泛应用 [1]。性能剖面图的一个示例如图 1所示。

量子计算引入了寻找此类组合优化问题解法的新技术，例如量子退火（QA） [4, 5] 和量子近似优化算法（QAOA） [6]，这些技术可能比经典方法更具优势。理论研究和经典模拟表明，对于某些问题，QAOA 有望胜过经典算法 [7, 8],，而针对 QA 系统的一些实证测试也表明，在有限场景下，其性能优于经典方法 [3, 9–11]。

大量工作致力于评估量子计算机在优化应用中的性能（见第二节）。然而，我们发现，若要让量子研究领域之外的用户也能使用此类基准测试，就必须既纳入量子计算基准测试的新兴方法，又以经典优化和运筹学（OR）等领域专家能够理解的方式呈现结果。

本文展示了如何通过精心构建的基准测试程序，监测并表征组合优化应用在量子计算系统上的执行过程，从而深入了解提升其性能和整体吞吐量的方案。

此外，分析和呈现方法可以采用量子计算专家熟悉的结构，同时借鉴运筹学评估求解器解决优化问题时结果质量的方式。这些改进后的分析和可视化技术能够提供有用的信息，说明量子计算解能够达到怎样的吞吐量，以及可以调整哪些因素来提升这些系统的性能。虽然组件级和简单的应用级基准测试能够提供有关总体性能特征的有用信息，但优化应用还能进一步深入分析量子优化应用的总拥有成本。尽管这些技术在运筹学领域早已得到应用，但将其有效应用于量子

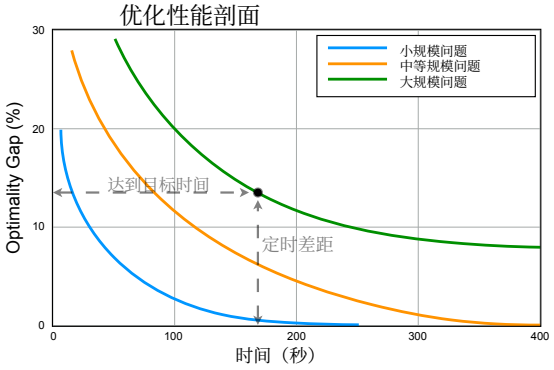
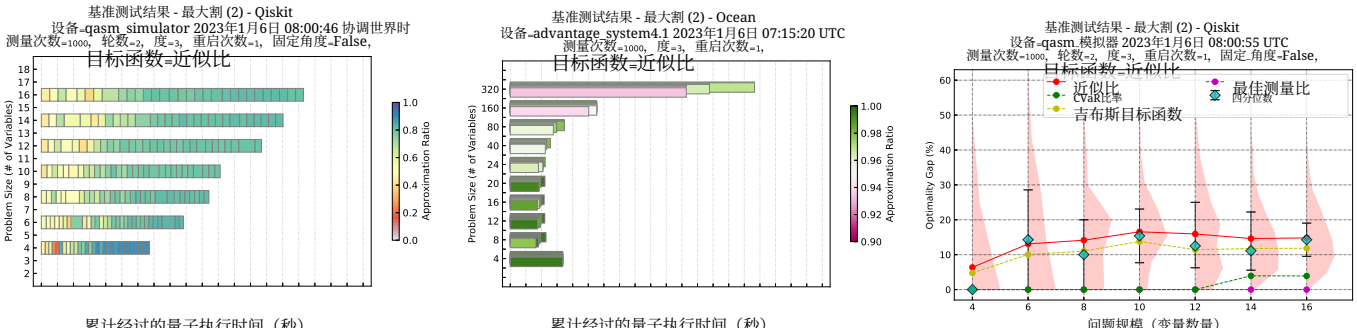


图 1：**用于基准测试优化方法的性能剖面图示例**。性能剖面图（如图所示）在运筹学界被广泛用于理解、传达和比较优化方法的性能。解的质量（相对于最优解的差异，即最优性差距）会随着优化算法的执行而不断变化。这使用户能够评估获得指定质量解所需的时间（达到目标时间），或经过指定时间后所达到的解的质量（定时差距）。定时差距指标是运筹学领域的事实标准，反映了大多数工业优化应用中的实际需求。性能剖面通常会随问题规模变化。变量数量较少的问题通常能够相当快地收敛到最优解。相比之下，对于规模较大的问题，要获得超过某一质量阈值的解可能很困难；这正是具有挑战性的优化任务属于 NP-困难问题的预期特征。

计算仍处于早期阶段。

具体而言，我们提出了一种方法和通用框架，用于表征基于不同底层技术的量子计算系统上执行的组合优化求解器的性能。我们以 QAOA 算法在门模型系统上的执行为例，展示该框架的特点并突出其优势。我们还通过在退火硬件上使用量子退火，展示该框架对其他类型求解器的适应能力。未来，我们计划将其扩展到其他技术，例如冷原子。我们使用广泛研究的最大割问题来展示该框架的能力，其目标是求出无向图的最大割大小。最大割问题提供了一个简单的早期目标，可用于评估量子计算解决方案的有效性。这些解决方案还可以扩展到规模更大的应用，并纳入约束及其他会增加挑战的问题特性。 [12, 13] 问题，其目标是求出无向图的最大割大小。最大割问题提供了一个简单的早期目标，可用于评估量子计算解决方案的有效性。这些解决方案还可以扩展到规模更大的应用，并纳入约束及其他会增加挑战的问题特性。

这项新的优化应用基准测试是对现有开源 QED-C 面向应用的基准测试套件 [14, 15] 的增强。该套件收录了多种算法基准测试，用于评估（门模型）量子计算机在当前与优化无关的问题上的性能，并支持在多个系统上执行、收集和分析性能指标，以及以统一的方式呈现这些指标。以这一现有框架为基础，使我们能够轻松扩展其功能



(a) (b) (c)

图 2：**量子计算解决方案的性能特征**。 这些新的性能剖面图展示了两种量子计算解决方案在无权最大割问题上的结果质量与执行时间之间的权衡：(a) 量子近似优化算法和 (b) 量子退火。每一行对应不同的问题规模。X轴显示累积执行时间，矩形的颜色表示由近似比定义的解的质量（= 1 –*opt_gap*/100）。对于 QAOA，连续的矩形展示其迭代执行过程，追踪其搜索合适参数并收敛至最优解的过程。对于 QA，每个堆叠矩形代表一次在逐渐增加的退火时间下进行的独立执行。(c) 展示了算法成功度的若干指标，即近似比的变体，这些指标绘制在各问题规模下最终测量结果的分布上。

，并使其易于使用且能让广大受众轻松获取。

这项新的基准测试涵盖了运行量子优化应用的集成式混合量子经典计算机系统的多个组件，并模拟其在现实世界中的使用情况。将现有的指标可视化工具与针对优化问题的新技术相结合，便能深入探究不同目标系统上的算法性能。为展示这些分析功能，我们在图 2中给出了由这项新基准测试生成的若干种可视化图表。尽管许多先前的基准测试研究都以最大割作为优化应用的示例，但我们的方法具有独特的灵活性、通用性和可定制性。

我们注意到，本文所述工作并未对不同类型的量子计算系统进行全面比较，也不涉及优化问题经典解的基准测试，因为这一领域已有大量深入研究（参见IID节）。本文的性能结果旨在主要展示我们的基准测试框架的特点和优势，而非满足运筹学领域对启发式方法性能研究的方法论要求。

在 QA 算法中纳入基准测试有其重要目的。数十年来，QA 一直受到广泛研究，其在退火硬件上的性能特征也已得到充分了解。纳入 QA 展示了我们的基准测试框架如何轻松适配门模型以外的量子计算技术。我们将 QA 用作其他求解器的替代代表；这些求解器可能采用量子技术，并在远程计算服务的硬件中执行算法的大部分内容。我们展示了该框架如何管理一系列基准问题的执行，并以一致的方式收集和分析不同技术的指标。

我们的研究发现，许多变量都会影响量子计算机解决优化问题的效果。

然而，我们并未对这些因素进行全面研究，也无法在所有情况下调整供应商特定的硬件设置以获得最佳结果。因此，本文呈现的性能结果不应被视为可推广至其他测试场景。全面研究影响量子性能的所有因素，并厘清算法和硬件各自的贡献，超出了本文的范围。

我们对量子基准测试的贡献主要有三方面：

- 提出一种评估异构量子平台上运行的量子计算机性能的方法，该方法借鉴了评估经典优化启发式方法的标准流程。

- 实现并演示了一套用于优化应用的开源基准测试流程。该流程可与不断演进的 QED-C 基准测试框架无缝集成，并使用户能够轻松开展性能研究。

- 通过一个应用领域关注的常见 NP-困难问题，展示该框架的能力及其能够支持的性能分析类型。作为示例，我们重点分析应用在多个量子硬件后端系统上的吞吐量，并将其作为使用量子解的总成本的一项影响因素。

我们希望这项工作能阐明在量子计算系统上实现组合优化求解器时涉及的实际考量，并鼓励和帮助其他人衡量和记录量子算法及计算系统开发方面的进展。我们提出，该框架可用于探索近期优化问题量子算法中的诸多创新 [16–24]。

本文其余部分的结构如下。第二节介绍量子计算机及其应用性能基准测试的基础知识。第三节介绍对 QED-C 面向应用的基准测试套件的改进，并描述基准测试算法。接下来，第四节讨论如何分析并呈现基准测试收集的指标。通过在经典实现的量子模拟器上执行测试，所得结果验证了结果符合预期，并凸显了这些基准测试所能提供的见解。

在第五节中，我们分析了采用上一节所述方法，在两种门模型量子硬件系统和一台量子退火处理器上运行这些基准测试所获得的结果。本文末尾附有若干附录，详细介绍组合优化问题的量子解，并重点说明影响结果质量的因素、参数选择中的权衡，以及量子算法固有的可扩展性挑战。

II. 背景

该基准测试框架衡量了用于求解组合优化问题的两种主流量子启发式算法的性能特征：量子近似优化算法使用门模型量子计算机，而量子退火则使用模拟量子计算机。

本文围绕这些算法在求解最大割问题中的应用，对其进行了基准测试。本节概述该问题的特征，随后介绍我们拟进行基准测试的量子解。我们还简要回顾了现有的基准测试方法，并讨论我们的方法与这些方法的相似之处和差异。

A. 最大割优化问题

最大割问题已成为量子优化领域广受欢迎的基准 [12, 25–27]原因有二：（1）它是最具挑战性的组合优化任务之一，即使是求得近似解也很困难，也就是说，它属于 APX-困难问题 [13, 28], ；（2）作为一项无约束离散优化任务，它可以自然地编码为二次无约束二进制优化（QUBO）[29, 30] 或伊辛模型 [5, 31], , 与当前的量子优化算法（量子近似优化算法 QAOA、量子退火 QA）十分契合。这些问题常见于将实际应用映射 [32, 33] 到计算硬件时，也可能作为复合算法中的子程序。

最大割问题的输入是一个由节点或顶点（V）和边（E）组成的无向图。（图的每条边都可以带有“权重”，但本文只考虑无权 3-正则图。）割是将图的节点划分为两个集合。割的规模定义为连接

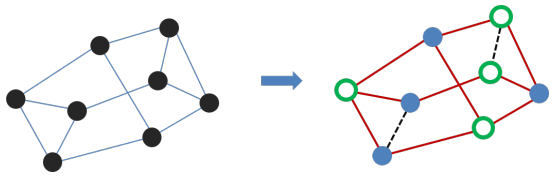


图 3：**最大割问题**。对于由节点（或顶点，V）和边（E）组成的无向图，应将顶点划分为互补的集合，使集合之间的边数最多。此图展示了一个包含八个节点的图中最大割问题某个实例的一种解，节点和边用不同颜色表示。颜色不同的节点分别属于解割的两个集合。连接不同集合中节点的实线红色边的数量，就是该图的最大割。

属于不同集合的节点的图边数量。最大割问题是从所有可能的割中找出规模最大的一个。（这里，我们考虑最大割问题的无权版本，因此图的每条边权重都相同。）

图3 展示了一个具有代表性的8节点图，其中每个节点都恰好通过三条边与其他节点相连。这类图称为3-正则图，或3度图。3度图被认为是最难求解的图之一 [34]。该图展示了最大割问题的一个解，其中节点和边用不同颜色标出。连接这些节点的红色边的数量就是该图的最大割。

由于在更大规模下，即使要为最大割问题寻找近似解也面临挑战（对于经典和量子算法皆如此），人们通常会计算一个称为近似比的量，以表征所得解的质量。例如，在 图3所示的问题中，最大割大小为10（总共12条边）。一个简单的优化算法随机尝试各种割，但因时间不够而未能全部测试，可能会得出最大割大小为9的结论。在这种情况下，近似比会小于9/10，即0。90，因为它是根据所有已找到解的分布计算出的统计量。

在针对优化问题对经典算法进行基准测试时，通常会根据结果与最优解之间的距离来报告结果的质量。最优性差距与近似比的关系如下：公式 1。

$$optimality_gap = (1.0 - approximation_ratio) \times 100 \quad (1)$$

这两种衡量量子系统性能的指标分别适用于不同的情境。在我们的工作中，我们同时采用这两种指标，以便讨论基准测试实现所取得的结果。

B. 优化问题的量子算法

量子退火和基于电路的量子计算机采用根本不同的策略来求解组合优化问题。为帮助理解我们的工作，我们简要概述这些量子算法如何运行以找到此类问题的解。关于这些算法的更多细节见B节。

优化问题（例如最大割）可以用一个哈密顿量 H_P 来描述，该哈密顿量因问题而异，并表示其变量和约束。问题的最优解对应于此哈密顿量的基态。量子近似优化算法和量子退火算法利用量子态演化计算H_P 的能量期望值，并确定变量 $\boldsymbol{\beta}$ 和 $\boldsymbol{\gamma}$ 的取值，从而得到能量最低的本征态

$$F_{\beta,\gamma} := \langle \boldsymbol{\beta}, \boldsymbol{\gamma} | H_P | \boldsymbol{\beta}, \boldsymbol{\gamma} \rangle . \tag{2}$$

解的质量，或近似比（AR），可以定义为计算所得能量状态$F_{\beta,\ \gamma}$与真实基态能量$E_{\min}$ 之比（假设已知）。在我们的案例中，哈密顿量的能量始终为负值或零。

$$AR = F_{\beta,\gamma}/E_{\min} \tag{3}$$

a. 量子近似优化算法量子近似优化算法（QAOA）可以说是使用门模型量子处理器求解组合优化问题的主要候选算法。QAOA属于变分量子算法（VQA） [35]，通常以迭代方式实现，由经典优化器对参数化量子电路进行“训练”。QAOA是一种启发式方法，旨在求解组合优化问题，例如QUBO问题。具体而言，该问题被编码为二元变量的指定二次函数，目标是找到一组变量赋值，使函数值最小。

QAOA的核心是“试探电路”，即参数化量子电路。在电路末端进行计算基测量，相当于从问题所有可能答案的概率分布中采样。通过改变参数值并反复测量电路，经典优化器可用于获得可能产生最优或近似最优解的参数值。

b. 量子退火 量子退火通过一种灵活的方法，以系统化的流程寻找函数的全局最小值，从而有效地解决优化问题。量子退火的算法方法受到量子力学中的绝热定理启发，将易于制备的初始哈密顿量的基态转化为编码组合优化问题的“目标”哈密顿量的基态。

从宏观层面来看，该协议通过对以下哈密顿量进行模拟插值过程，力求找出用户指定的H^{Target} 模型的低能态，最终得到演化结束时的最低能量态：

$$H(s) = (1-s)H_{\text{Init}} + (s)H_{\text{Target}}. \tag{4}$$

在量子退火中，从用户的视角来看，收敛到解的过程完全在量子系统内部完成。问题被映射到硬件上的一个初始态（相对于问题基的均匀叠加态），随后系统被设定为朝着解进行退火。较长的退火时间通常与较高的解的质量相关。

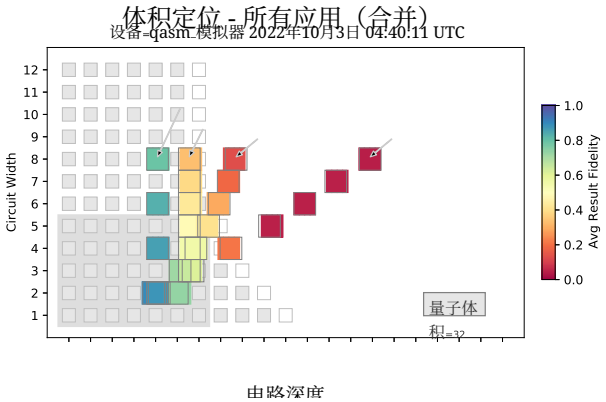


图4：**面向应用的基准测试**。在此，我们展示一种方法，用于呈现通过含噪量子计算机模拟器执行多个最多涉及 8 个量子比特的面向应用基准测试所获得的结果保真度。该图展示了平均结果保真度随电路宽度的变化，并将电路深度绘制在体积背景上，以直观呈现基准测试电路的“轮廓”和结果保真度。（由 *QED-C* 基准测试套件生成的图）

C. 量子计算机基准测试

本节回顾了以往基准测试工作中的概念和定义，本文后续内容将引用这些概念和定义。我们主要关注组合优化应用基准测试中面向应用层面的性能评估。

a. 系统级基准测试关于门模型计算系统的组件级和系统级基准测试，已有大量参考资料 [36–42]。我们采用两种知名的系统级性能基准测试——量子体积（QV） [43, 44] 和体积基准测试（VB） [45, 46] ——作为若干基准测试图的背景。虽然这两种方法都能表征量子电路执行的质量和规模，但都无法提供程序运行所需时间的信息，而这对于评估任何计算解决方案的总成本至关重要。

自 2011 [47–49]以来，量子退火系统一直可供实证研究。早期论文 [47, 50, 51] 以及近期针对大规模量子退火硬件的基准测试提案 [52–54]中，介绍了评估量子退火处理器的组件级和系统级方法。

b. 应用级基准测试 组件级和系统级指标有助于深入了解系统的整体能力，但要预测具有特定通用性能水平的机器在某一特定应用类别中的效用，可能颇具挑战 [46, 55]。为此，应用导向的基准测试会运行定义明确的程序，以提供应用专属性能指标。

由于量子退火相对成熟，其基准测试往往涉及使用超过 100 个量子比特模型的应用层测试，这给通过与理想量子系统的经典模拟进行比较来验证带来了重大挑战 [3, 56–58]。此类基准测试工作通常会比较

量子硬件的运行时间性能与经典方法的性能 [3, 59–62]。使用合成优化问题，将具有已知最优解的问题实例植入其中 [63–65]。

相比之下，早期阶段的门模型量子计算机需要针对规模较小的问题和量子比特数量较少的基准测试。面向应用的基准测试框架通常会创建使用公认量子门组合或算法的电路，提供输入和预期输出，并在量子模拟器或物理硬件上执行这些电路 [14, 15, 66–70]。结果质量指标通过比较预期测量结果与实际测量结果之间的统计差异，或根据测量结果推导出的特定应用指标的接近程度来计算。

尤其值得关注的是首个 QED-C 面向应用的基准测试套件 [14, 15], , 我们的工作正是以此为基础。QED-C 套件提供了一种实用方法，用于评估各类量子程序在一系列量子硬件和模拟器系统上的性能。其基准测试程序会遍历不同的问题规模和输入特征，同时系统地采集关键性能指标，例如结果质量、执行运行时间和所消耗的量子门资源，如图 4所示。支持性基础设施和抽象层让广大用户都能使用这些基准测试。该框架还提供了相应结构，以支持对量子近似优化算法（如 QAOA）等迭代算法进行基准测试，或像量子退火（QA）那样一次性执行算法。

QED-C 基准测试计算了若干重要的性能指标，我们在本文中始终使用这些指标。单个电路的结果质量以“归一化海林格保真度”表示，这是对标准“海林格保真度”的一种修正，通过对指标进行缩放，考虑完全噪声设备所产生的均匀分布。资源消耗以门层总数来量化，即“电路深度”，它可以是“算法型”或“归一化”的（转译到常规基）。执行时间则以“经过的量子执行时间”（挂钟时间）或“量子执行时间”（由量子服务提供商服务报告）来衡量；某些系统还可提供更细粒度的计时。这些指标在我们先前的工作中有详细定义 [15]。

用于采集运行时间指标的框架，是我们开展组合优化算法基准测试这项新工作的基础。尽管首批 QED-C 基准测试已纳入执行时间，但对迭代算法和混合算法进行基准测试，能够更全面地展现量子计算机的运行时性能。在量子应用中执行电路所需的时间，是一项重要的性能指标 [71–74], , 在比较量子计算和经典计算时经常被引用。例如，量子计算和经典计算所用的时间，是近期量子优势演示中的核心内容 [75, 76]。

D. 用于优化任务的 Quantum 基准测试

关于经典解的基准测试以及优化问题的比较，有大量背景资料可供参考 [3, 62, 77] ；也有不少资料将量子方法与经典解进行比较 [78–80]。这类工作通常着眼于提升量子算法的性能，而不涉及对特定硬件平台的分析。对于量子退火和量子近似优化算法，这种方法通常侧重于比较量子算法的不同实现或调优策略，而非将其运行时间与经典算法进行比较。

十多年来，基于量子退火技术的量子计算一直被用于解决组合优化问题 [3, 9, 11, 62, 80–83]。在这些研究中，主要关注的指标是预期解质量和操作计数 （即电路深度），它们被用作近似运行时间的衡量标准。在此背景下，人们采用了两种主要的执行时间分析思路，用于比较量子退火算法与经典启发式算法的性能。达到解所需时间（TTS）指标用于确定将问题精确求解至最优所需的预期实际运行时间 [48, 82–85]。相比之下，达到目标所需时间 （TTT）指标用于确定将问题求解至指定目标 解质量所需的预期实际运行时间 [3, 9, 56, 80]。典型做法是测量经典方法和量子方法的 TTS 或 TTT 随问题规模的变化情况（即尺度优势），从而预测在何种系统规模下，量子求解方法可能比经典方法更快。在本研究中，我们利用这些概念分析解的质量与执行量子算法所需时间之间的权衡。

在真实硬件上对量子解进行基准测试面临着重大挑战，因为量子计算机噪声特性和运行时间开销带来了额外要求。由于配置用于量子硬件基准测试的优化任务实现复杂，因此出现了多个软件框架，用以支持评估在不同平台上实现的相同（或相似）优化方法。SupermarQ [67] 和 QPack Scores [68] 等基准测试框架在其示例基准测试中纳入了一个或多个量子近似优化算法应用，而 QUARK [69]则考虑了工业领域中出现的特定优化问题。据称，Q-score 指标 [70]适用于多个类别的量子处理器；它通过测量求解器在固定时限内能够胜过随机猜测的最大图规模来进行评估。所有参考资料均展示了在多台后端量子计算机上测量解的有效性、可行性和运行时间的结果，其中一些同时涉及门模型和量子退火设备。我们借鉴了其中的大量研究成果，指导开发基于组合优化问题的 QED-C 基准测试套件新基准。

III. QED-C基准框架增强

本节介绍我们对 QED-C 基准测试套件所做的增强，这些增强用于收集并分析与混合量子应用（例如量子近似优化算法和量子退火）相关的特定于应用的质量和时间指标。在这类应用中，解的质量与资源利用率（此处指执行时间）之间的权衡至关重要。我们的工作有两个主要目标：1）将其他以优化为核心的基准测试工作中的关键概念整合到 QED-C 基准测试套件中并加以完善，使其成为标准功能；2）以运筹学领域从业者易于理解的方式呈现结果和分析，这些从业者已熟悉对优化问题的经典解进行基准测试。

本节介绍了我们为跨范式量子优化启发式算法基准测试开发的框架的具体特性。相较于初始基准测试套件中使用的简单测试场景，我们对原有基准测试框架所做的增强，源于优化问题和启发式算法性能评估所特有的特性与挑战。运筹学（OR）领域已开发出用于评估这一背景下计算性能的方法和工具，我们将其中一些方法和工具改编用于量子场景。理论基础详见附录A。

我们的基准测试框架有若干不同于其他框架的特点。用户可以详细评估执行时间和解的质量，并探索二者之间的权衡（而非固定使用特定的 TTS 或 TTT 指标）。该平台支持对能够运行量子退火或门模型算法的量子计算硬件进行基准测试。此外，用户还可以选择感兴趣的问题和输入，而不局限于我们使用最大割输入所做的简单示例。基准测试结果的呈现遵循运筹学和 QED-C 框架的标准方法。随着量子计算机规模不断扩大，基准测试框架将能够支持对各种优化问题进行测试。

我们重点关注其在组合优化问题中的应用，并以最大割为具体示例来展示关键改进。与初始基准测试套件中使用的简单算法不同，后者以电路执行保真度为关键指标；改进后的基准测试必须推导出特定于应用的解质量指标，并考虑到优化问题的解通常是近似解。通过在经典实现的基于电路的量子模拟器上执行该算法所得的结果，展示了如何收集和呈现关键指标。

QED-C 基准测试框架包含用于管理基准测试算法执行的共享函数，这些算法涵盖一系列问题定义；同时收集执行过程中的指标，并在不同后端目标之间一致地呈现结果。最大割量子近似优化算法和量子退火基准测试算法都在目标系统后端_*ID*上运行，遍历一系列问题规模 [最小 _ 规模，最大 _ 规模], , 以求解由输入问题定义的参数。由最大_重启次数参数控制的内层循环，使得基准测试算法能够在每个问题规模下多次执行。

算法1 QAOA 基准测试算法

```
目标 ←后端_标识符
初始化_指标()
对于规模 ← 最小_规模,最大_规模执行
    电路_定义 ← 定义_问题(问题, 规模, 参数, 轮数)
    针对重新启动 _ 标识符 ← 1, max_ 重启次数 执行
        成本_ 函数 ← 定义_成本_ 函数(问题)
        电路，数量_参数 ← 创建_电路(电路_定义)
        缓存_电路 ←编译_电路(电路)
        参数[β, γ] ← 随机（数量_参数）
        当 最小化器() 未完成时                          ▷ 最小化
            电路 ← 应用_参数（缓存_电路，参数）
            计数 ← 执行（目标，电路，数量_测量次数）
            能量，质量 ←成本_ 函数（计数）
            存储_迭代_指标(质量, 计时)
            参数[β, γ] ← 优化(参数[β, γ,能量])
            完成 ← 若最低值(能量) 已找到
            完成 ← 若迭代_限制_已达到()
        结束 while
        计算_并_存储_重新启动_指标()
    结束循环
    计算_和_，存储_组_指标()
结束循环
```

对于测试的每个问题规模，我们使用*NetworkX* 软件包生成一组随机的 3-正则图 [86] ，并使用*Gurobi* 软件包 [87]确定每个图的最大割大小。这些结果分别用于生成用于测试的量子电路，以及在执行后确定解的质量。

QAOA 与量子退火算法在实际应用中的一个关键区别体现在重启循环中：在此过程中，会将特定求解器应用于输入，并在不断增加的执行时间下评估解的质量。门模型设备会依次执行一系列量子电路，并测试不同的参数值，以找到能够产生低能态的一组参数。相比之下，量子退火系统则会逐渐尝试到达其最低能态，就像横向伊辛模型经历量子力学演化一样。要比较这些系统中状态随时间的演化，需要采用不同的数据收集和呈现方式。下文将详细介绍用于对这些解进行基准测试的相关算法，突出它们之间的差异及其对结果的影响；为简洁起见，省略部分细节。

A. QAOA基准测试算法

QAOA 基准测试方法定义于算法1中。嵌套在第一和第二个 for 循环中的，是 QAOA 算法，其中根据问题的具体情况定义一个代价_函数，以及一个实现与该问题相关联的哈密顿量、并由变量 $\boldsymbol{\beta}$ 和 $\boldsymbol{\gamma}$参数化的门模型量子电路。QAOA 所用的量子电路可以重复若干轮数 (通常在代码中称为 p)。

从一组随机或固定的参数开始，*params*（对应于 $|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle$ 来自方程 B1。量子电路执行*shots* 次，以获得测量计数并计算代价函数的值。经典优化器代码通过改变参数集来探索参数空间，以获得代表哈密顿量最低能态的测量计数；这一过程会不断迭代，直到确定最低能量或达到迭代上限。相关的

算法2 QA基准算法

```
目标 ←后端_标识
初始化_指标()
对于 规模 ← 最小_规模，最大_规模 执行
    对于重新启动_标识 ← 1, max_重启次数 执行
        计算_质量 ← 定义_计算_质量(问题)
        对于一个 _时间 ← 分钟_退火 _time, max_退火 _时间 执行
            嵌入 ← 定义_问题(问题, 规模, 参数)
            采样器 ← 创建_采样器(目标,嵌入)
            样本 ← 采样_伊辛(采样器,a_时间,读取次数)
            质量 ←计算_质量(样本)
            存储_迭代_指标(质量, 计时)
        结束循环
        计算_并_存储_重新启动_指标()
    结束循环
    计算_并_存储_分组_指标()
结束循环
```

质量指标会被计算并存储，同时还会记录量子和经典计时信息。尽管我们在基准测试中使用随机初始参数，但在实际应用中，用户可能掌握一些有关合理初始点的信息，从而获得更优或更快的解（参见E2 节）。

该算法的执行结果会受到若干 QAOA 特有因素的影响，例如测量次数和轮数、所采用的经典优化器类型，以及最重要的目标系统中的噪声水平。经典优化器执行的迭代次数也可能限制结果的质量。在 QAOA基准测试中，这是一个可配置选项，但我们默认将此限制设为 30，以避免在昂贵的硬件上出现失控执行。

最大割基准测试可以通过两种不同的方法执行，从而能够独立于完整的 QAOA 算法研究这些因素。方法（1）针对特定问题执行一次试探电路实例，并使用可配置的测量次数和轮数，以便详细分析这些因素。方法（2）执行完整的 QAOA 算法，并提供指定经典优化器的选项，默认使用 COBYLA。此外，还可以选择若干代价函数变体之一。这些变体将在下文第四节 A中介绍。

B. QA 基准测试算法

QA 基准测试方法在算法2中进行了描述。QA 基准测试算法的核心在前两个`for` 循环中。该算法采用了 QA 基准测试特有的指标收集循环。

从用户的角度来看，使用量子退火器收敛到一个解只需一个步骤，整个过程都在量子系统内完成。描述该问题的哈密顿量被嵌入设备的量子组件中，并随时间演化，最终达到代表最优解的最低能态。

在我们的量子退火基准测试中，QA算法会在该特定采集循环开始后多次执行。退火_时间初始化为 1µs，每次执行后加倍，直至达到 256µs（退火时间的范围可在基准测试中配置）。每次执行时，问题会映射到硬件上的一个初始状态（相对于问题基的等叠加态），并将系统设置为朝着某个解进行退火。较长的退火时间通常对应更高的解的质量。

采用这种方法，我们可以通过捕获每个退火时间下获得的解的质量，提供一个与 QAOA 基准测试相当的时间与质量权衡度量。在单次 QA 算法执行过程中，无法监测解质量的演变。

对于基准测试中执行的每次运行，算法都会查询（读取次数）样本，查询次数与我们在 QAOA 中进行测量的次数相同。与 QAOA 基准测试一样，每次执行后都会计算相关的质量指标，并与跟踪量子和经典计时信息的指标一并存储。

该算法的执行结果主要受测量次数或样本数、所用退火时间以及目标执行系统中的噪声水平影响。Max-Cut 基准测试可以通过两种不同的方法执行，从而能够独立于整个 QA 算法研究这些因素。方法（1）针对每种问题规模，以可配置的测量次数，在单个可配置的退火时间下执行，以便详细分析这些因素。方法（2）执行本节所述的完整 QA 算法。请注意，在这种情况下，方法（1）与方法（2）相同，只是将范围收缩为单个值。

IV. 最大割基准测试指标分析

本节将讨论我们对基准测试执行过程中收集的数据所进行的分析。首先，我们讨论针对最大割问题明确计算的应用特定质量指标，这些指标以统一方式适用于量子近似优化算法和量子退火算法。基准测试框架生成最优性差距图和割大小分布图，作为简洁直观地呈现这些附加指标的实用工具。此外，除近似比之外，还可以使用其他指标评估结果质量，本节将对此进行讨论。

随后，我们将通过可视化展示执行时间与所得结果质量之间的权衡。虽然使用简单折线图呈现这些数据是常见做法，但我们生成了信息更丰富的图表：使用颜色表示执行质量，使用水平长度表示单次迭代的执行时间，并使用垂直/水平位置分别表示问题规模和累计执行时间。本节最后分析测量次数和轮数对该基准测试的 QAOA实现结果质量的影响。

A. 特定应用的结果质量指标

QAOA 和量子退火算法试图通过在目标系统上进行一系列参数测试（QAOA）或退火操作（量子退火），寻找哈密顿量的最低能量态，从而收敛到一个优化问题的解。在

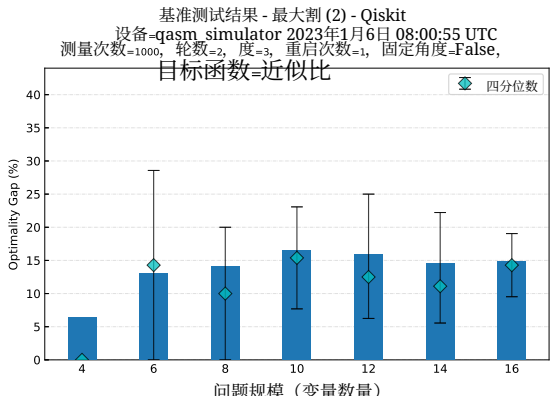


图 5：**展示最终最优性差距**。随着量子近似优化算法程序执行，最小化器会找到最大割问题的最优解，该解由最终迭代后计算出的近似比表示。根据这些数值计算每种问题规模对应的最优性差距，并在此条形图中展示，同时用四分位标记显示最终测量结果的分布。在某些情况下，延长执行时间可以提高结果质量，但为节省计算资源，我们将基准测试限制为 30 次迭代。

这两种情况下，结果都是一个能量值，根据执行结束时获得的能量样本分布计算得出。

该能量值可与预先计算的最优解（即哈密顿量的最低能量）进行比较。实际能量值与预期能量值之比，即近似比 [6, 30, 88], 是通常用于量化最大割问题结果质量的指标，相关讨论见第 IIA 节。其形式化定义如下。

设M 表示测量次数，因此对于给定的值（向量） $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$，我们得到M 个割大小，每个割大小对应于通过计算基测量得到的一个比特串。将这些能量记为E_1, 。 . . , E_M，并按非递减顺序排列。由于这些能量为≤ 0, $|E_1|$, . . . , $|E_M|$，它们是按非递增顺序排列的非负整数。因此，能量期望值近似为

$$F \approx \frac{\sum_{i=1}^{M} E_i}{M}. \tag{5}$$

将这项计算的结果归一化到范围 [0, 1]很方便。因此，我们在以下内容中定义近似比

$$\text{Approximation Ratio } r = \frac{F_{\beta,\gamma}}{|E_{\min}|}, \tag{6}$$

其中 $E_{\min} < 0$ 是问题哈密顿量的实际基态能量。

优化性能研究通常使用其补数，即最优性差距。在图5中，我们展示了针对每种受测问题规模，在基准测试算法执行结束时计算出的最终最优性差距。此图还包含四分位数线，除了均值之外，也展示了分布的宽度。尽管这些结果是使用 QAOA 算法在经典实现的量子模拟器上得出的，图中也以同样的方式绘出了 ourQA 算法的结果。

近似比是衡量量子计算系统求解最大割问题能力的重要指标。分布中找到的割大小均值越高，算法越有可能产生最大割大小的割。不过，我们注意到，这些量子算法的最终输出并非近似比，而是在算法的所有迭代过程中得到的测得的最佳割（即对应最大割大小的割）。

测得的最佳割往往无法很好地衡量结果质量，因为它在数值上不稳定，尤其是在样本数量较少时。不过，它启发了其他一些目标函数，例如条件风险价值（ CVaR） [89], 和吉布斯目标函数 [90]。这两种指标都关注分布的尾部，而不是对所有测量结果一视同仁。

为了说明这些不同的质量指标之间的关系，在 图6中，我们展示了在问题规模为 16、采集 1000 个样本（测量次数）的条件下，在无噪声量子模拟器上运行基准测试所得到的割大小分布。基准测试得到的分布以黑线表示。宽粉色线表示在计算系统上运行算法并返回均匀随机结果时得到的模拟分布。峰值越靠右，表示结果质量越高。在本例中，测得的最佳结果显示为 1.0，表明该算法返回了预期的最优最大割。条件风险价值比和 Gibbs 比介于近似比与测得的最佳比之间。

形式上，对于选定的参数值，CVaR [89]， $\alpha \in (0, 1]$, 定义如下

$$\text{CVaR}_\alpha(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \frac{1}{\lceil \alpha N \rceil} \sum_{i=1}^{\lceil \alpha N \rceil} E_i, \tag{7}$$

其中， $\lceil . \rceil$ 表示向上取整函数。CVaR_α 表示测得的能量分布中较低的 α 尾部所对应的（负）割大小均值。极限 $\alpha \to 0$对应基态能量值（即 E_1） ， 而$\alpha = 1$ 对应能量期望值 $F_{\beta,\ \gamma}$。该指标的值取决于 α 的选择。虽然可以配置此参数的值，但在所有图表和分析中，我们默认采用 $\alpha = 0.1$。

另一个选择是吉布斯目标函数 [90], , 其定义为

$$f_\eta(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \ln \langle \boldsymbol{\beta}, \boldsymbol{\gamma}| \, e^{-\eta H} \, |\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle, \tag{8}$$

其中 $\eta > 0$, H 表示所求基态的哈密顿量。参数 η 决定了表达式中H 的低能态的相对权重。参数 η 在两个极值之间调节f_η （类似于条件风险价值中的 α ） : $f_{\eta=0}(\boldsymbol{\beta}, \boldsymbol{\gamma}) = F_{\beta,\gamma}$, 而 $f_{\eta\to\infty} = E_{\min}$, 即$H$ 在状态 $|\boldsymbol{\beta}\boldsymbol{\gamma}\rangle$中的最低可测能量。

我们对这些目标函数分别进行归一化，使其

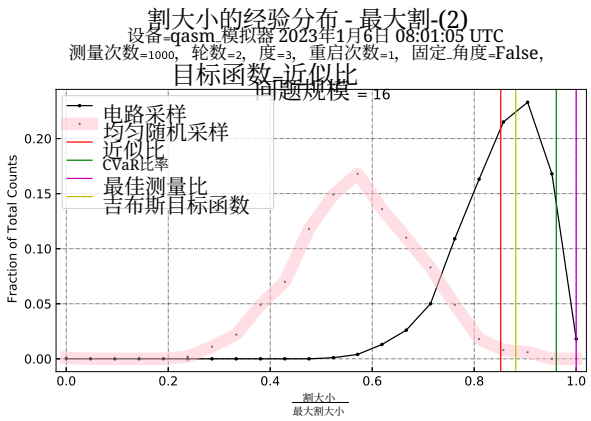


图 6：**割大小分布：** 通过检查最终优化器迭代中获得的割大小值分布，可以了解 QAOA 最终输出的质量。分布峰值越靠右，结果质量越高。图中还绘出了各项指标（竖线）以及对应于比特串均匀随机采样的分布（粉色线）。

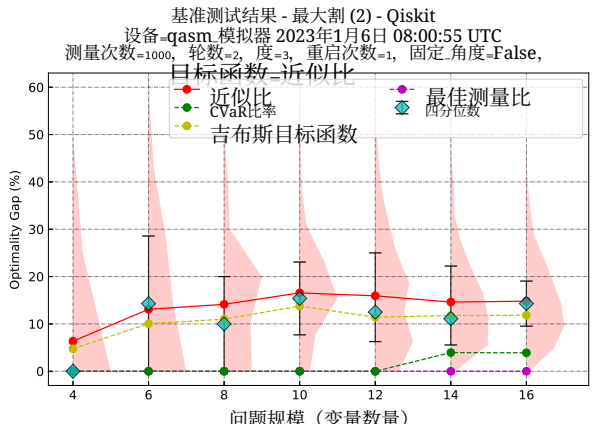


图 7：**详细的最优性差距图**。可使用多种指标评估 QAOA 输出最终分布的质量。对于每个已实现的问题规模或电路宽度（沿 X轴），我们绘制最优性差距（沿 Y轴）。所得的最优性差距分布以粉色半小提琴图表示。与条件风险价值比率、近似比、吉布斯比率和最佳割比率对应的最优性差距值则以折线图表示。

位于范围 $[0,1]$ 内，并由此定义以下量：

$$\begin{aligned} \text{CVaR}_\alpha \text{ Ratio} &= \frac{\text{CVaR}_\alpha}{|E_{\min}|}, \\ \text{Gibbs Ratio} &= \frac{f_\eta}{\eta |E_{\min}|}, \\ \text{Best Measurement Ratio} &= \frac{E_1}{|E_{\min}|}. \end{aligned} \tag{9}$$

在我们的基准测试框架中，目标函数可以设为其中任意一种。近似比常用于量子计算求解优化问题的研究中，而文献中较少使用其他比率。这些指标用于衡量解的质量，取值为 1.0 时表示最优。

我们的框架还会生成“详细的最优性差距图”（例如图 7）。对于每个问题规模，使用半小提琴图展示通过经验观察得到的割大小分布。（所绘制的分布是 1 – 割大小 最优割大小这一比值的分布，因此其归一化范围为 [0,1]）。公式 6-9中的四项指标以最优性差距的形式呈现，即 (1 – 指标_值) * 100。这展示了不同质量指标下，结果质量如何随问题规模变化的详细情况。

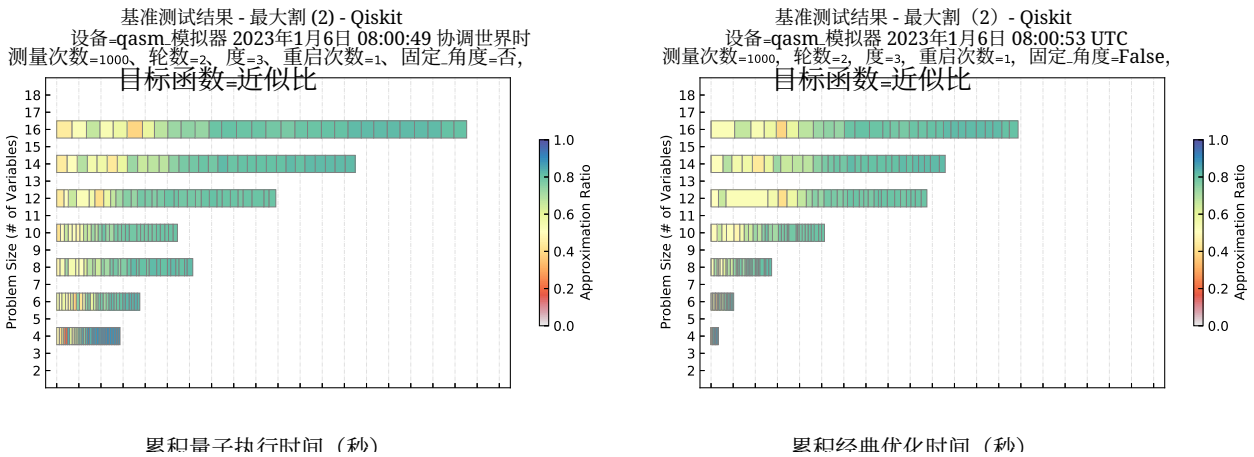
B. 结果质量与执行时间

在本节中，我们将介绍一种新方法，用于可视化最大割基准测试结果中解的质量与执行运行时间之间的关系。该方法的灵感来自运筹学中常见的可视化方式（例如图 1中的性能剖面图）。不过，我们对其进行了改进，使其能够揭示有关混合量子计算算法执行时间的独特而有价值的信息。这种方法的某些方面在量子计算成熟度的早期阶段尤其相关。它们提供了关键信息，帮助我们了解瓶颈及其他阻碍因素，而这些因素对系统吞吐量的影响比人们意识到的更大。

我们的方法既适用于 QAOA，也适用于量子退火，尽管图表呈现略有不同，以反映算法差异。在图 8中，我们展示了 QAOA 算法的时间与质量权衡关系，采用了一种称为“面积图”的新型性能剖面图。与图 4中的体积图类似，该图以电路宽度（问题规模）为 Y轴，并通过矩形的颜色表示指标得分。在面积图中，单个矩形的水平宽度表示某个试探态的执行时间，其在 X轴上的位置则表示累计执行时间，其中包括之前各次迭代的执行时间。

QAOA 和 QA 基准测试之间的一个关键差异在于如何评估绘制在 X轴上的执行时间。对于 QAOA，算法本身以一系列迭代的形式执行，执行时间会随着每次迭代累积，并以堆叠矩形表示。然而，对于 QA，整个算法只需一步即可执行。为了评估算法在不同时间下的表现，每次都从头开始执行算法，并采用不同的退火时间。图 2b展示了如何用相互重叠而非堆叠的矩形表示重新初始化。关于 QA 执行的更多细节，请参见第 V C 节。

在此处展示的面积图中，我们使用近似比作为默认的评价指标来衡量结果的质量。近似比的范围为 0 到 1.0，但对于量子近似优化算法（QAOA），其数值通常从 0.5 以上开始，并随着优化器收敛到一个解而不断振荡。由于退火计算机的特性，量子退火（QA）的初始点通常在 0.9 以上。我们对量子退火采用不同的颜色刻度，以强调这些算法之间的根本区别。不过，在运行基准测试时，



(a) (b)

图 8：**QAOA最大割算法的迭代执行**。在 QAOA 中，经典最小化函数会迭代执行试探态、改变其参数值并计算代价函数，以收敛至最优解。在 (a) 中，每一条水平行表示给定问题规模（量子比特数）下的连续迭代；X轴上的位置表示累积量子执行时间，颜色则表示优化器每次迭代的近似比。在 (b) 中，X轴表示累计经典执行时间（优化器）。这两种时间共同构成用户体验到的总耗时。

可以选择上一节中的任何目标函数，作为图中显示的评价指标。

基准测试框架会收集多项执行时间指标。图 图 8 中的第一幅图展示累积量子执行时间，即量子处理器执行所耗费的时间。第二幅图展示累计经典执行时间，主要包括 QAOA 中优化器所花费的时间，或 QA 中的设置时间。在图 2a中，我们展示累计经过的量子执行时间，即包含上述两项以及其他设置时间（例如编译或将程序载入量子处理器所需时间）在内的总墙钟时间。附录 D中对这些时间以及与 QAOA 和 QA 算法相关的其他重要时间进行了详细分析。

此处采用的质量与时间可视化方式——面积图——显著增强了向用户呈现的信息，帮助用户了解执行 QAOA 或量子退火等混合量子算法的情况。例如，这种 QAOA 和量子退火的演化分析可以提供有关执行所消耗的增量时间单位的信息。对于一些较新的混合系统以及误差缓解的应用，向用户说明执行过程中的瓶颈或异常极具价值。

例如，在图 8的图中，表示每次迭代所用时间的若干矩形，其宽度并不一致。量子计算尚处于发展初期，执行流水线通常包含许多涉及非确定性经典计算的步骤，其中一些是量子计算特有的，例如误差缓解。这些图有效地传达了执行时间不可预测程度的衡量结果，而这种不可预测性可能导致吞吐量下降。效地传达了执行时间不可预测程度的衡量结果，而这种不可预测性可能导致吞吐量下降。

这些图中还传达了其他量子计算所特有的信息。例如，量子近似优化算法可能需要经典预处理，具体包括将程序从中间表示编译并转译为目标拓扑和门集。相比之下，量子退火需要在执行前将问题图嵌入设备拓扑中。对于门模型量子计算机，间歇性的校准过程有时会中断程序执行，并以宽度较大的矩形呈现。此外，执行这些算法的任何电路组件通常还会产生启动成本。

虽然此处仅展示了少数示例，但面积图可让用户一目了然地查看所有这些信息，并帮助他们快速理解执行时间如何影响量子算法的结果质量和整体吞吐量。

C. 影响 QAOA 试探态保真度的因素

执行 QAOA 算法会受到若干因素的影响。在此，我们使用 QAOA 基准测试中的方法（1），分析测量次数和轮数如何影响量子计算系统执行该算法所用试探电路的能力。

QAOA 算法的每次迭代都涉及反复测量一个参数化电路，该电路使用由经典优化器程序确定的参数值 $|\boldsymbol{\beta},\boldsymbol{\gamma}\rangle$ 执行。算法的成功取决于量子子程序能否准确计算目标函数值。如果量子子程序得到的测量概率与理想分布 $P_{\text{ideal}}(s) = |\langle s\ |\boldsymbol{\beta},\boldsymbol{\gamma}\rangle|^2$ 中的概率不够匹配，就可能对经典优化器的效果产生负面影响。

即使在无噪声模拟器中，也只有在测量次数趋于无限时才能达到完美保真度。在量子硬件上，噪声和退相干会加剧保真度的下降，量子比特之间有限的连接性也会如此。为了量化

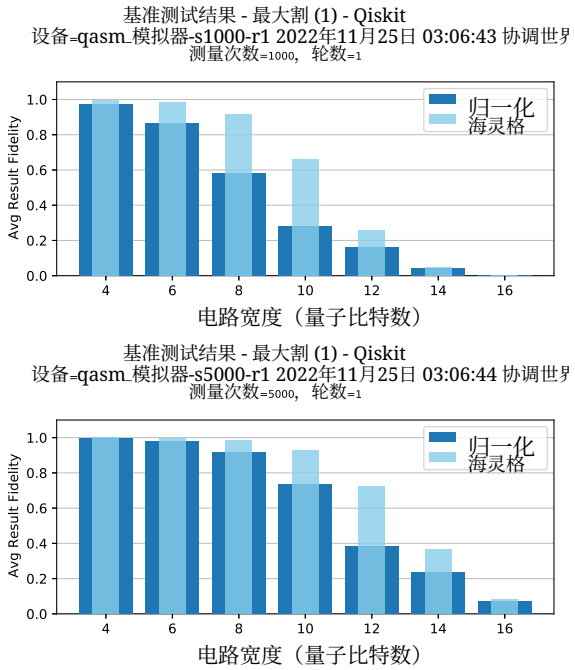


图 9：**测量次数对试探态执行保真度的影响。** 此处，我们展示了在理想量子模拟器上，以 1000 次测量和 5000 次测量执行相同的最大割试探态电路时，电路宽度从 4 到 16 个量子比特的情况下，保真度的差异。对于每种问题规模，我们使用一个图来定义最大割问题的实例 。使用更多测量次数时，所得保真度更高。

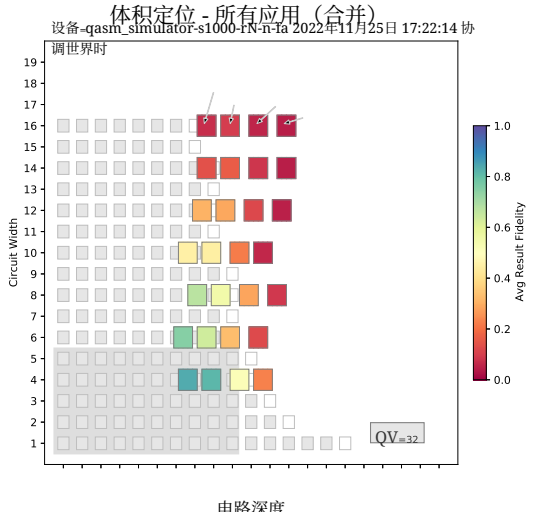


图 10：**保真度的体积化呈现及轮数的影响。** 执行任何量子电路所获得的保真度指标，都会受到电路宽度及其深度（即量子门层总数）的影响。此处，在量子体积为 32 的有噪声量子模拟器上执行具有不同轮数的最大割试探态电路（单量子比特门和双量子比特门的错误率分别为 0.003 和 0.03）。特定宽度和深度下的（归一化）结果保真度以对应位置矩形的颜色表示，并会随着轮数（深度）或问题规模（宽度）的增加而降低。

电路保真度，我们采用海灵格保真度和归一化海林格保真度，这两者的定义见我们最初关于面向应用的基准测试的研究 [15]。在我们的语境中，归一化保真度最为实用；需要指出的是，电路保真度为 0 对应均匀随机概率分布，而保真度为 1 则对应理想分布。

在图 9中，每次迭代的测量次数（我们称为“shots”）对电路保真度有显著影响。例如，在此处使用的无噪声模拟器上，使用 1000 次测量时，电路归一化保真度在八个量子比特时降至 0.6 以下，而使用 5000 次测量时，则在十二个量子比特时才降至该值。随着电路宽度增加，区分理想分布与随机分布所需的测量次数也会增加。对于此处使用的 QAOA 变体，更大的问题需要更多测量次数，才能有效捕捉所得更大规模分布中的割大小。

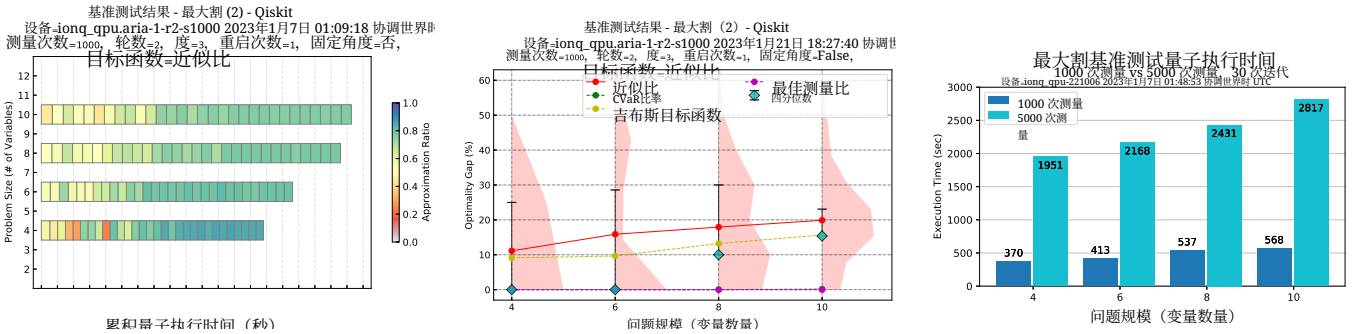
图 10 展示了随着量子近似优化算法试探态定义中的一个参数——轮数 （在代码中称为 p ）从 1 增加到 8，电路保真度如何受到影响。随着电路变得更宽（即包含更多量子比特）且更深（即门层数增加），执行保真度预计都会下降。轮数增加会使电路变得更深。体积图使用颜色刻度表示在每种受测电路宽度和深度下的保真度。在本例中，电路在量子模拟器上执行了 1000 次测量；该模拟器的噪声特性模拟了典型量子计算机（单量子比特门和双量子比特门错误率分别为 0.003 和 0.03），且量子体积为 32（深色矩形所示区域）。随着“轮数”参数增大，电路也相应变得更深，结果保真度会随深度增加而下降。因此，增加轮数所带来的理论优势，会被执行门数更多的深电路所导致的较低保真度抵消。

V. 在量子硬件上执行 RE

本节展示了在若干具有代表性的量子硬件系统上执行最大割基准测试的结果，这些系统基于不同的底层量子技术。我们的目标是证明该基准测试框架具备稳健能力，能够准确捕捉关键性能指标，从而凸显不同技术之间的根本差异。

本演示可为这些系统的提供者提供宝贵的资源，帮助他们了解不同代硬件之间性能的逐步提升。它还可为用户提供工具，使其全面理解使用这些新兴技术时固有的权衡。本文特别关注解的质量与执行运行时间之间的关系。

不过，我们强调，本节中的结果不应被视为这些量子平台整体比较性能的代表。这些结果旨在用于说明，并非用于对量子系统进行正式比较。此外，我们使用



(a) (b) (c)

图 11：**在离子阱系统上的执行**。这些图展示了在不同问题规模下，于 IonQ Aria QPU 上运行最大割基准测试的结果（30 次迭代，每次进行 1000 次测量）。面积图 (a) 显示，随着累积量子执行时间增加（十个量子比特时约为 2400 秒），每种问题规模的近似比均有所提升。小提琴图 (b) 显示每个计算所得近似比对应的最终最优性差距，并说明随着问题规模增大，近似比会下降（十个量子比特时约为 20 %）。图 (c) 展示了另一种系统 IonQ Harmony 的数据，对比了使用 1000 次测量（十个量子比特时为 568 秒）和 5000 次测量（十个量子比特时为 2817 秒）时的总量子执行时间。这表明，在这些系统上执行最大割算法的成本几乎与使用的测量次数成正比。(数据通过云服务收集。)

各制造商的软件默认设置和参数设置。与量子系统一样，这些软件工具集也在迅速发展；因此，我们对量子系统性能的结论仅代表某一时点的进展情况。我们希望用户利用该框架创建更全面的基准测试，并据此得出结论。

当前量子计算硬件上的基准算法结果会受到若干关键因素的影响。门保真度不足和退相干造成的误差可能导致测得的分布与理想系统之间出现显著差异，尤其是在电路规模较大时。这些误差会在量子近似优化算法等迭代算法中累积，从而降低结果的质量。量子退火（QA）中的噪声（例如热能和控制线波动）可能会扰乱量子比特态。低效的映射或针对特定硬件拓扑的嵌入会加剧这些影响。

除了纯粹的量子计算之外，量子近似优化算法返回的解的质量还显著取决于经典计算的质量，例如每轮对 beta 和 gamma 的编译与优化。同样，量子退火返回的解的质量取决于次嵌入和后处理等经典操作，以及控制量子计算的用户参数。从这个意义上说，我们的基准测试框架应被视为一种工具，用于评估量子系统性能与算法选择和参数设置相结合时的表现，而非独立电路的性能。

在硬件上运行量子优化应用时，由于程序会受到噪声的负面影响，结果质量会比在模拟器上运行时有所下降。为展示硬件执行的实际极限，我们在三类不同的量子计算机上执行这些基准测试：离子阱、超导传输子和量子退火系统。

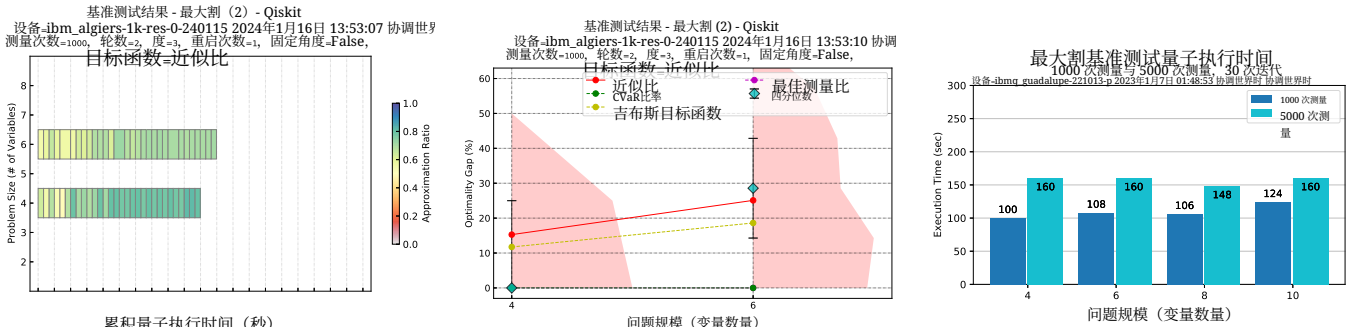
A. 离子阱系统

我们首先展示在两台通过远程访问、采用离子阱技术的门模型量子计算系统上执行最大割基准测试的结果。基于离子阱的量子计算机具有全连接和高保真度的优势，但执行时间比其他技术更长，抵消了这一优势。

图 11展示了使用 IonQ Aria QPU（一台第二代离子阱计算机）及其上一代产品 IonQ Harmony 得出的结果。对于从 4 个量子比特到 10 个量子比特的每种问题规模，我们都在这两个系统上执行了基准测试，最多迭代 30 次，每次使用 1000 次测量。

第一张图（a）使用第 IV B 节中的面积图，展示对于每种问题规模，近似比如何随着每次迭代累积量子执行时间的增加而提升（在 Aria 上，十个量子比特时为 ~ 2400s）。问题规模较大时，试探电路会更深，从而增加总执行时间并降低质量。小提琴图（b）显示了结果质量的下降，其中每种问题规模下计算出的比率对应的最终最优性差距会随问题规模增加而增大（十个量子比特时为 ~ 20%）。不过请注意，最佳测量比的差距为 0%。

本图的 (c) 图展示了使用 IonQ Har- 生成的数据，mony，我们在该系统上执行了两次最大割基准测试，一次使用 1000 次测量，另一次使用 5000 次测量。对于所有问题规模，累计量子执行
时间大致与测量次数的增加成正比。
例如，10 个量子比特时，耗时从 568 秒增加到
2817 秒，增至原来的 4.959 倍。
特定问题规模下柱形之间的高度差表示所花费时间的差异，即额外执行 4000 次测量所需的时间，
因为两次运行都使用相同的初始化时间，并且各执行 30 次迭代。
由此，我们可以计算每次量子测量的执行时间。对于六个量子比特的情况，例如，该时间为（2168 – 413）/（4000 ∗ 30），即14.6



(a) (b) (c)

图 12：**在超导传输子系统上执行**。这些图展示了在 IBM Quantum *ibm_algiers* 系统上以不同问题规模执行最大割基准测试的结果（迭代 30 次，每次测量 1000 次）。执行时使用了 Sampler 原语，未进行误差缓解（韧性级别 0 ）。面积图 (a) 显示，随着累积量子执行时间增加，每种问题规模的近似比均有所提升（六个量子比特时达到 ~33s）。小提琴图 (b) 显示了每个计算所得近似比的最终最优性差距，并展示了问题规模增大时近似比如何下降（六个量子比特时降至 ~25%）。图 (c) 展示了来自另一系统 IBM Quantum *ibm_guadalupe*的数据，对比了使用 1000 次测量（十个量子比特时为 124s）和 5000 次测量（十个量子比特时为 160s）时的总量子执行时间。这表明，在该系统上执行最大割算法时，增加测量次数仅会使成本略微上升。（数据通过云服务收集。）

IonQ Harmony 上为 ms/shot。这些结果表明，用户在这类量子计算机上执行最大割或类似算法的总成本，不仅包括取决于测量次数的财务成本，还包括因执行时间较长而导致的总吞吐量下降。

我们观察到，在 IonQ Aria 上的执行时间比在 IonQ Harmony 上更长。例如，对于 10 个量子比特，前者约需 2400 秒，后者约需 568 秒。Aria 设备会对测量数据应用误差缓解方案，以改善结果。不过，我们没有探究 Aria 执行时间增加且质量提升的原因。

B. 超导传输子系统

这里，我们展示了在可远程访问、采用超导 Transmon 技术的双门模型量子计算系统上执行最大割基准测试的结果。这些结果突显并量化了基于该技术构建的硬件实现中量子程序执行的特性。基于超导 Transmon 量子比特的量子计算机执行速度比其他技术更快。然而，由于需要引入交换操作来弥补量子比特之间连接性的局限，这一优势会被保真度降低所抵消。

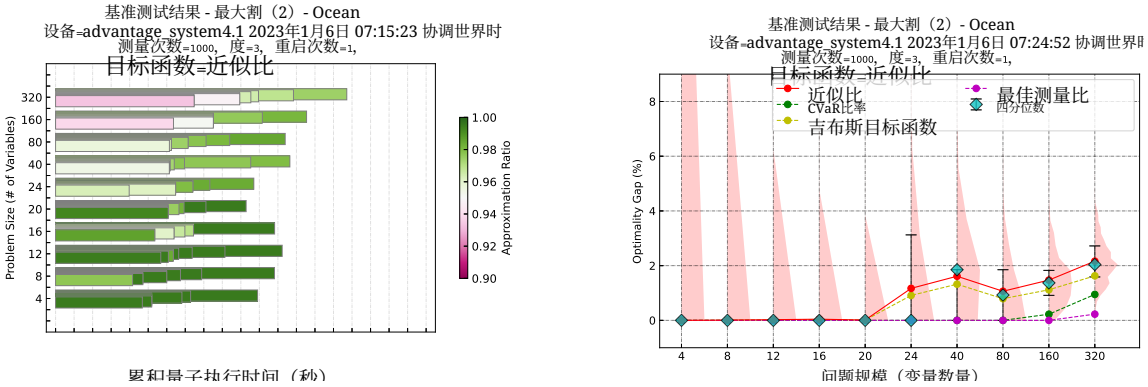
两个系统上的执行均通过 IBM Cloud Qiskit Runtime 服务运行 Qiskit Sampler 原语完成 [91]。通过将“resiliencelevel” 执行参数设为 0，关闭了误差缓解_。虽然 Sampler 原语提供自动误差缓解功能，但在硬件演示中，我们选择不启用该功能。建议用户自行评估选择此选项对结果质量和总执行时间的影响。

图中的曲线图 12展示了执行在 IBM Quantum *ibm_algiers* 系统上进行最大割基准测试的结果，涵盖不同的问题规模（30 次迭代，每次 1000 次测量）。面积图 (a) 显示，随着每次迭代的累积量子执行时间增加（六个量子比特时达到 ~33 秒），各问题规模下的近似比均有所提升。问题规模较大时，试探电路更深，执行总时间更长，结果质量也会下降。不过，请注意该系统整体的运行时性能。每次迭代进行 1000 次测量，在四个量子比特时需要 ~1 秒，在六个量子比特时需要 ~1.1 秒。因此，在最多进行 30 次迭代的限制下，完成量子近似优化算法的执行分别需要 30 秒和 33 秒。

小提琴图 (b) 显示，随着近似比下降，每个问题规模下计算出的最终最优性差距都会随问题规模增大而增加（六个量子比特时为 ~25%）。不过，最佳测量比差距为 0%，表明该算法在这两种问题规模下都能找到最大割。

图 (c) 展示了在另一台系统 IBM Quantum *ibm_guadalupe* 上执行基准测试所得的数据。这是一台较早一代的系统，其执行时间长于 *ibm_algiers*。不过，我们使用这些结果来说明超导传输子计算机上量子程序执行的一个重要方面。这里，我们比较了使用 1000 次测量（十个量子比特时为 124 秒）和 5000 次测量（十个量子比特时为 160 秒）运行基准测试时得到的总量子执行时间。由于两次运行执行了类似的初始化处理，因此每种问题规模下柱形的高度差代表了在 30 次迭代中额外执行 4000 次测量所需的时间。这使我们能够确定在此量子计算设备上执行一次测量所需的时间。例如，对于六个量子比特，每次测量的执行时间可计算为 (160−108)/(4000×30)，即 0.43 毫秒/次测量。

这些结果表明，增加测量次数来执行量子程序，只会使在该系统上执行最大割优化或类似算法的成本略有增加。用户可以利用更高的吞吐量



(a) (b)

图 13：**在量子退火系统上执行**。这些图展示了在不同问题规模下，于 D-Wave *advantage_system4.1* 量子退火系统上执行最大割基准测试的结果（每个规模进行 1000 次测量）。问题嵌入一次后，在 1 毫秒至 256 毫秒的退火时间范围内执行，以评估量子退火时间对结果质量的影响。面积图 (a) 显示，随着累积量子执行时间增加，各问题规模下的近似比均有所提升（在 320 个变量时达到 ~ 0.31 ）。小提琴图 (b) 显示每个计算所得近似比对应的最终最优性差距，并说明随着问题规模增大，近似比会下降（在 320 个变量时降至 ~2%）。（通过云服务收集的数据。）

在这类设备上反复执行电路以搜索最优结果，同时仍能比使用其他技术缩短总执行时间。此外，还可以探索通过执行更多测量次数来获得更高质量的结果。

C. 量子退火系统

我们工作的一个重要重点是构建 QED-C 基准测试套件，使其能够在采用门模型量子电路以外的量子技术实现的后端系统上执行部分基准测试。本节将介绍如何通过 LEAP Solver“advantage_system4.1”访问 D-Wave Advantage 系统，并在其上执行最大割基准测试，以此说明该框架如何支持测试编排、采用不同参数、采集相关性能指标，以及以一致的方式呈现不同量子技术的结果。

图 13 中的图表展示了在 D-Wave *advantage_system4.1*量子退火系统上执行最大割基准测试的结果，问题规模从 4 到 320 个变量（每次执行均进行 1000 次测量）。与相应的门模型图表一样，面积图 (a) 显示，随着累积量子执行时间的增加（在 320 个变量时达到 ~ 0.31 秒），每种问题规模下的近似比都会提高。类似地，小提琴图 (b) 显示了每个计算所得比率对应的最终最优性差距；我们可以看到，随着问题规模增大，近似比会下降（在 320 个变量时降至 ~2%）。

不过，QA 基准测试的实现（详见III B 节）与 QAOA 基准测试不同，这会影响结果的呈现方式。对于每个问题规模，都会从头重新执行 QA算法，按步长将退火时间加倍，从 1 增至 128μs。在退火版本的图 (a) 中，每个问题规模对应的矩形会彼此叠加绘制，以突出其与 QAOA基准测试算法的区别。每个矩形代表一次完整执行，但退火时间更长。

这样，我们便以与展示 QAOA 算法相同的方式，呈现了量子退火中质量与时间的权衡，同时清晰地传达了不同算法之间的差异。我们的目的是让用户能够快速了解特定退火时间下预期可达到的质量水平，这对于评估量子退火解决方案的总体成本至关重要。

随着问题规模增大，退火算法的总量子执行时间仅略有增加：从 4 个变量时的 ~0.22s 增至 320 个变量时的 ~0.32s。这是因为图中报告的量子执行时间除了实际退火时间外，还包括芯片编程和读出等非量子操作，而实际退火时间相较之下很短。这表明，虽然增加计算的退火时间可能会提高财务成本，但不会影响使用量子退火时可实现的吞吐量。

在此基准测试所测的问题规模下，QA 基准测试所得的近似比在所有情况下都高于 0.90。因此，QA 面积图采用了不同的颜色刻度来表示近似比，以便在这一不同的数值范围内直观呈现质量的变化。另请注意，QA 算法基准测试能够为最多包含 160 个变量的问题确定最佳割大小。

D. 硬件结果讨论

QED-C 基准测试套件的最大割扩展功能允许用户控制问题的定义和规模、测量次数、轮数、重启次数、初始角度，以及优化器的选择及其参数。这些设置可能的组合数量很多，我们只探索了其中有限的一部分。在

上述硬件测试中，我们在三种不同类型的量子计算机上执行不同规模的基准问题，并探究解的质量如何随执行时间增加而变化。在其中两项测试中，我们还调整了测量次数（1000 和 5000），以评估该参数对系统吞吐量的影响。

在其他测试中，我们使用根据这些目标系统的噪声特性实现的模拟，发现将轮数参数设为 2，能在 QAOA 算法的有效性与较长电路中噪声造成的性能下降之间取得良好平衡。其他测试表明，在使用门模型硬件运行 QAOA 的小规模测试中，将测量次数设为 1000、轮数设为 2，是在量子硬件上执行时最合适的默认配置，可最大限度地减少基准测试期间对硬件资源的占用。

感兴趣的读者可参阅附录 E 节，其中我们讨论了更全面的演示，这些演示旨在确定各种不同参数如何影响结果质量。结果表明，多次重启或可改善基准测试结果。值得注意的是，采用固定初始角度并结合多次重启，可能是一种有效的方法，能够提供标准的优化基准测试，无需完整执行 QAOA 即可评估目标系统的有效性，从而减少基准测试所需的资源。本节还介绍了一种借助这些基准测试开发的参数调优策略，用于找出 QAOA 执行过程中若干参数的最佳组合。

在上述结果中，我们使用面积图呈现累计量子执行时间，以说明随着执行过程的推进，结果质量如何得到改善。硬件提供商报告的这一时间反映了量子处理器（或模拟器）的使用情况。这一点至关重要，因为它会直接影响使用量子计算机所需的财务成本，而该成本在不同系统之间存在显著差异。我们的分析揭示了不同类别量子计算技术在执行吞吐量方面存在显著差异。在评估量子计算机的实用性时，必须同时考虑财务成本和完成任务所需的时间。

评估量子计算解总成本时，另一个需要考虑的关键因素是量子处理中执行时间之外的其他时间成本。为执行准备量子程序会产生资源开销。这可能包括编译时间、针对目标拓扑和门集进行转译、加载已编译的程序，以及向量子系统传入和传出数据。这些开销组成部分会直接影响优化应用的总吞吐量，因为每次执行都会包含其中一部分或全部开销。

这些吞吐量因素在不同供应商之间可能存在很大差异，尤其是在执行环境方面，并且也会因用户而异。例如，程序的预编译，或经典处理器与量子处理器的共址，都可能显著影响总吞吐量。此外，还有许多我们未曾测试的供应商特定硬件设置，可能有助于改善结果。

此外，为适配各自对应的门模型和基于退火的硬件范式而设计的量子近似优化算法和量子退火算法，具有不同的属性和结构，会显著影响计算时间和解的质量。这种复杂性可能会掩盖硬件和系统性能的影响。对于量子近似优化算法，优化器及其参数的选择可能会显著影响基准测试结果。

如前所述，我们研究中的性能结果仅用于说明，不应被视为通常情况下相对性能的代表。影响解的质量和总运行时间的变量过多，因此这些结果无法被视为对这些系统性能的确切描述。基于这些原因，我们不对这些因素进行详尽分析。全面研究所有影响量子性能的因素，以厘清硬件的贡献，超出了本文的范围。相反，我们建议用户在使用此处介绍的基准测试套件进行研究时，将执行的总成本（包括这些额外开销）纳入考量。

值得注意的是，固有变化可能会显著影响这些基准测试所采用的量子算法的结果质量。尽管我们的研究未深入探讨此类变化，但我们的基准测试框架提供了一个探索优化算法最新进展的平台。例如，可以配置该框架，纳入预先计算对角矩阵等创新方法 [16],，研究多轮 [17, 18], ，采用多角度变分形式 [19、22],使用高表达能力变分形式 [23],，或采用仅需少量量子比特的大规模求解器 [24]。随着量子计算硬件逐步具备容错和纠错功能，同步推进算法发展势在必行。采用此处展示的测试的更精简版本，对于评估性能的同步提升至关重要。

VI. 总结与结论

尽管当前一代量子计算机的计算能力可能有限，但预计这些系统将迅速发展，并能够执行日益复杂的任务。因此，对于这一发展进程而言，建立准确且经过验证的进展测量方法至关重要；这些方法应可供此类系统的开发者以及利用该资源解决现实世界问题的用户随时使用。

为此，我们基于现有的开源 QED-C 面向应用的基准测试套件进行了扩展，使其能够支持对组合优化问题的量子-经典混合解进行基准测试；组合优化问题常被视为量子计算的一种应用场景。多种因素会影响量子计算机有效求解组合优化问题的能力。用于求解这些问题的算法只能给出近似答案，而结果的质量通常取决于可用于处理的时间，且时间往往受到严格约束。我们的

基准测试旨在提供一种评估这些因素的机制，并为如何提升其性能和整体吞吐量提供宝贵且关键的见解。

我们通过在门模型系统上执行量子近似优化算法，展示了该框架的特性并突显其优势；同时，通过在模拟量子计算机系统上执行量子退火算法，展示了该框架对其他类型求解器的适应性。我们使用广泛研究的最大割问题展示框架的能力。该问题为评估量子计算解提供了一个简单的早期目标，同时也可扩展至未来更大规模的应用挑战。在未来的工作中，我们计划将这一演示扩展至其他技术，例如冷原子。

本研究的主要目标之一，是构建分析与呈现方法，使其采用量子计算专家熟悉的方法，同时借鉴运筹学评估求解器解决优化问题时结果质量的视角。针对量子解的特定特征，这些方法也进行了增强，例如统计抽样或 QAQA 算法的迭代特性。这些方法能够向用户指出执行过程中的瓶颈或异常，具有极高价值。

这些改进后的分析与可视化技术能够提供有关量子计算解决方案吞吐量的有用信息，并帮助深入了解其总拥有成本。这些基准测试的结果能够让用户了解可调整哪些因素，以提升这些系统的性能。虽然这些技术在运筹学领域已使用多年，但将其有效应用于量子计算仍处于起步阶段。

本文不对不同类型的量子计算系统进行全面比较，也不涉及优化问题经典解的基准测试，因为这一领域已有许多深入研究。本文中的性能结果旨在展示我们基准测试框架的特点和优势。我们将退火硬件上的量子退火算法基准测试作为使用不同量子技术的其他求解器的替代示例。我们的研究发现了许多会影响量子计算机高效求解优化问题的参数。不过，我们没有对这些参数或所有可能产生最优结果的选项进行详尽研究。我们不试图涵盖或分析算法参数或特定供应商执行选项的所有可能设置。

虽然我们通常会指出，一类设备可能提供质量更高的结果，而另一类设备可能执行得更快，但本文重点关注基准测试框架的结构，以及如何利用该框架探索量子算法的执行与性能。用户可以在自己有权使用的设备上执行这些基准测试程序，并自行评估这项技术的总拥有成本。这对于了解量子计算机何时以及如何能够带来可衡量的价值至关重要。

我们认为，考虑到现有技术水平，这项工作具有前瞻性。离散优化问题的复杂性促使人们开发各种方法，例如量子近似优化算法、量子退火等。这些方法在最坏情况复杂度分析中可能面临挑战，但在实践中却能发挥价值。基于性能的指标和基准测试工具可以量化这些方法取得的进展，并提供一种比较替代方案的途径；这些方案的能力超出现有技术所能达到的水平。随着量子计算机不断发展，我们在此定义的基准测试方法对于衡量性能提升至关重要。

代码可用性

本文介绍的基准测试套件代码可在以下网址获取：https://github.com/SRI-International/QC-App-Oriented-Benchmarks。仓库中提供了详细说明。

致谢

作者感谢在本项工作中使用 IBM Quantum 服务。文中所表达的观点仅代表作者本人，并不反映 IBM 或 IBM Quantum 团队的官方政策或立场。IBM Quantum, https://quantum-computing.ibm.com/，2021年。

我们感谢 IonQ 提供硬件访问权限。文中观点仅代表作者，不代表 IonQ 的官方政策或立场。

我们感谢 D-Wave Systems 提供硬件和软件工具的访问权限。文中观点仅代表作者，不代表 D-Wave Systems 的官方政策或立场。

洛斯阿拉莫斯国家实验室对本工作的贡献是在美国能源部国家核安全管理局的监督下开展的，合同编号为 89233218CNA000001。本研究使用了洛斯阿拉莫斯国家实验室机构计算计划提供的资源，并由实验室指导研究与开发计划项目 20210114ER 资助。

D.B. 感谢美国国家航空航天局学术任务服务（合同 NNA16BD14C，由 SAA2-403506 资助）。P.S. 感谢美国国家航空航天局/美国大学空间研究协会费曼量子学院实习项目的支持。D.B. 和 P.S. 均获美国国家科学基金会计算领域远征计划 CCF #1918549 资助。本研究使用了加州大学洛杉矶分校数字研究与教育研究所研究技术组提供的 Hoffman2共享集群相关计算和存储服务。

D-Wave®、OceanTM 和 AdvantageTM 是 D-Wave Systems, Inc. 的商标。IBM®、Qiskit®、IBM Q® 和 IBM Quantum System TwoTM 是 International Business Machines Corporation 的商标。IonQTM、IonQ HarmonyTM 和 IonQ AriaTM 是 IonQ, Inc. 的商标。

我们感谢 Verizon公司的 Jerry Gamble 对代码开发和编辑工作所作的贡献，

并感谢达特茅斯学院物理与天文学系的 Jason Necaise（此前任职于 D-Wave Systems）对代码开发所作的贡献。

感谢 Mark Johnson（D-Wave）、Andrew Wack（IBM）、DavidMcKay（IBM）、Paul Nation（IBM）、Luning Zhao（IonQ）、Ananth Kaushik（IonQ）、Farshud Sorouifar（俄亥俄州立大学）、Amos Anderson（Quantum Circuits）、Steve Reinhardt（Quantum Machines）、Davide Venturelli（美国大学空间研究协会/美国国家航空航天局）、Filip Maciejewski（美国大学空间研究协会/美国国家航空航天局）以及其他人为本文提供意见。

[1] Ehsan Zahedinejad 和 Arman Zaribafiyan。门模型量子计算机上的组合优化：综述，2017年。网址：https://arxiv.org/abs/1708.05294。[2] Daniel J. Egger、Claudio Gambella、Jakub Marecek、Scott Mc-Faddin、Martin Mevissen、Rudy Raymond、Andrea Simon-etto、Stefan Woerner 和 Elena Yndurain。金融领域的量子计算：现状与未来展望。《IEEE量子工程汇刊》，1:1–24，2020年。doi: 10.1109/TQE.2020.3030314。[3] Catherine C. McGeoch 和 Cong Wang。组合优化中绝热量子系统的实验评估。载于 ACM 国际计算前沿会议论文集，CF '13，纽约，纽约州，美国，2013年。计算机协会。ISBN 9781450320535。doi:10.1145/2482767.2482797。网址：https: //doi.org/10.1145/2482767.2482797。[4] A.B. Finnila、M.A. Gomez、C. Sebenik、C. Stenson 和 J.D. Doll。量子退火：最小化多维函数的新方法。《化学物理快报》，219(5):343 – 348，1994年。ISSN 0009-2614。doi: https://doi.org/10.1016/0009-2614(94)00117-0。网址：http://www.sciencedirect.com/science/article/pii/0009261494001170。[5] Tadashi Kadowaki 和 Hidetoshi Nishimori。横向伊辛模型中的量子退火。《物理评论E》，58:5355–5363，1998年11月。doi:10.1103/PhysRevE.58.5355。[6] Edward Farhi、Jeffrey Goldstone 和 Sam Gutmann。一种量子近似优化算法。（arXiv:1411.4028），2014年11月。doi:10.48550/arXiv.1411.4028。[7] Gavin E. Crooks。量子近似优化算法在最大割问题上的性能，2018年11月。[8] Jonathan Wurtz 和 Peter Love。$p>1$ 下最大割量子近似优化算法的性能保证。《物理评论A》，103(4):042612，2021年4月。doi: 10.1103/PhysRevA.103.042612。[9] Immanuel Trummer 和 Christoph Koch。D-Wave 2X 绝热量子计算机上的多查询优化，2015年。网址：https://arxiv.org/abs/1510.06437。[10] Yuchen Pang、Carleton Coffrin、Andrey Y. Lokhov 和 Marc Vuffray。量子退火用于快速解结构识别的潜力。《约束》，26(1):1–25，2021年10月。ISSN 1572-9354。doi:10.1007/s10601-020-09315-0。网址：https://doi.org/10.1007/s10601-020-09315-0。[11] Byron Tasseff、Tameem Albash、Zachary Morrell、Marc Vuffray、Andrey Y. Lokhov、Sidhant Misra 和 Carleton Coffrin。量子退火硬件在组合优化领域不断显现的潜力，2022年。网址：https://arxiv.or g/abs/2210.04291。[12] M. R. Garey、D. S. Johnson 和 L. Stockmeyer。若干简化的 NP-完全图问题。《理论计算机科学》，1(3):237–267，1976年2月。ISSN 0304-3975。doi: 10.1016/0304-3975(76)90059-1。[13] Christos H. Papadimitriou 和 Mihalis Yannakakis。优化、近似与复杂性类。《计算机与系统科学杂志》，43(3):425–440，1991年。ISSN 0022-0000。doi：https://doi.org/10.1016/0022-0000(91)90023-X。网址：https://www.sciencedirect.com/science/article/pii/002200009190023X。[14] 面向应用的量子计算性能基准，2015年。网址：https://github.com/SRI-International/QC-App-Oriented-Benchmarks。

[15] Thomas Lubinski、Sonika Johri、Paul Varosy、Jeremiah Cole- man、Luning Zhao、Jason Necaise、Charles H. Baldwin、Karl Mayer 和 Timothy Proctor。《面向应用的量子计算性能基准》。IEEE量子工程汇刊，4:1–32，2023年。doi: 10.1109/TQE.2023.3253761。[16] Danylo Lykov、Ruslan Shaydulin、Yue Sun、Yuri Alexeev 和 Marco Pistoia。高深度QAOA电路的快速模拟。载于《高性能计算、网络、存储与分析国际会议 SC '23研讨会论文集》，SC-W 2023。ACM，2023年11月。doi: 10.1145/3624062.3624216。URL http://dx.doi.org/10. 1145/3624062.3624216。[17] Yingyue Zhu、Zewen Zhang、Bhuvanesh Sundar、Alaina M Green、C Huerta Alderete、Nhung H Nguyen、Kaden R A Hazzard 和 Norbert M Linke。囚禁离子量子计算机上的多轮QAOA和高级混合器。《量子科学与技术》，8(1):015007，2022年11月。doi:10.1088/2058- 9565 205 8 91 10 1088 9565 91/ac ef。URL https://dx.doi.org/ . / - /ac ef。[18] Ritajit Majumdar、Dhiraj Madan、Debasmita Bhoumik、Dhinakaran Vinayagamurthy、Shesha Raghunathan 和 Susmita Sur-Kolay。优化QAOA中最大割问题的试探态设计，2021年。URL https://arxiv.org/abs/2106.02812。[19] Rebekah Herrman、Phillip C. Lotshaw、James Ostrowski、Travis S. Humble 和 George Siopsis。多角度量子近似优化算法，2021年。URL https: // www.nature.com/articles/s41598-022-10555-8。[20] Rebekah Herrman。动态图上的多角度量子近似优化算法与连续时间量子游走之间的关系，2022年。URL https://arxiv.org/abs/2209.00415。[21] Kaiyan Shi、Rebekah Herrman、Ruslan Shaydulin、Shouvanik Chakrabarti、Marco Pistoia 和 Jeffrey Larson。多角度QAOA并非总是需要用到所有角度。载于《2022年IEEE/ACM第七届边缘计算研讨会（SEC）》。IEEE，2022年12月。doi:10.1109/sec54971.2022.00062。URL http://dx.d oi.org/10.1109/SEC54971.2022.00062。[22] Michelle Chalupnik、Hans Melo、Yuri Alexeev 和 Alexey Galda。通过多参数、问题无关层扩展QAOA试探态，2022年。URL https://ieeexplore.ieee. org/document/9951267。[23] V. Vijendran、Aritra Das、Dax Enshan Koh、Syed M. Assad 和 Ping Koy Lam。用于低深度量子优化的高表达能力试探态，2023年。URL https://arxiv.org/abs/2302 .04479。[24] Marco Sciorilli、Lucas Borges、Taylor L. Patti、Diego García-Martín、Giancarlo Camilo、Anima Anandkumar 和 Leandro Aolita。迈向用少量量子比特构建大规模量子优化求解器，2024年。[25] Daniel Beaulieu 和 Anh Pham。使用IBM运行时环境评估混合量子优化算法在最大割聚类中的性能，2021年。URL https://arxiv. org/abs/2112.03199。[26] David Amaro、Carlo Modica、Matthias Rosenkranz、Mattia Fiorentini、Marcello Benedetti 和 Michael Lubasch。面向组合优化的变分量子算法筛选。《量子科学与技术》，7(1):015021，2022年1月。doi: 10 1 0881088 2058 9565 3 54 10 2 2058 9565 2 3 54 . / - / ac e 。URL https://doi.org/ . % F - % Fac e。[27] Linghua Zhu、Ho Lun Tang、George S. Barron、F. A. Calderon-

Vargas、Nicholas J. Mayhall、Edwin Barnes 和 Sophia E. Economou。一种用于在量子计算机上求解组合问题的自适应量子近似优化算法，2020 102 58 2005。URL https://arxiv.org/abs/ . .[28] Johan Håstad。一些最优不可近似性结果。《ACM学报》，48(4):798–859，2001年7月。ISSN 0004-5411。doi: 10.1145/502090.502098。[29] Fred Glover、Gary Kochenberger 和 Yu Du。QUBO模型的构建与应用教程，2018年。URL https://ar xiv.org/abs/1811.11538。[30] Leo Zhou、Sheng-Tao Wang、Soonwon Choi、Hannes Pichler 和 Mikhail D. Lukin。量子近似优化算法：近期量子设备上的性能、机制与实现。《物理评论X》，10(2)，2020年6月。doi: 10 1 1031103 10 021067 10 2 10 021067 . /physrevx. . . URL https://doi.org/ . % Fphysrevx. . . [31] Andrew Lucas。许多NP问题的伊辛形式化。《物理学前沿》，2，2014年。doi:10.3389/fphy.2014.00005。URL https://doi.org/10.3389%2Ffphy.2014.00005。[32] Francisco Barahona、Martin Grötschel、Michael Jünger 和 Gerhard Reinelt。组合优化在统计物理学和电路布局设计中的应用。《运筹学》，36(3):493–513，1988年6月。ISSN 0030-364X。doi:10.1287/opre.36.3.493。[33] Bahram Alidaee、Gary A. Kochenberger 和 Ahmad Ahma- dian。利用0-1二次规划求解两个调度问题的最优解。《国际系统科学杂志》，25(2):401–408，1994年2月。ISSN 0020- 7721。doi:10.1080/00207729408928968。[34] Cristina Bazgan 和 Zsolt Tuza。最大度为 3 的图上的最大割组合5/6近似。《离散算法杂志》，6(3):510–519，2008年。ISSN 1570-8667。doi:https://doi.org/10.1016/j.jda.2007.02.002。URL https://www.sciencedirect.com/science/article/pii/S1570866707000822。[35] M. Cerezo、Andrew Arrasmith、Ryan Babbush、Simon C. Ben- jamin、Suguru Endo、Keisuke Fujii、Jarrod R. McClean、Kosuke Mitarai、Xiao Yuan、Lukasz Cincio 和 Patrick J. Coles。变分量子算法。《自然评论：物理学》，3(9):625–644，2021年8月。doi:10.1038/s42254-021-00348-9。URL https://doi.org/10.1038%2Fs42254-021-00348-9。[36] E. Knill、D. Leibfried、R. Reichle、J. Britton、R. B. Blakestad、J. D. Jost、C. Langer、R. Ozeri、S. Seidelin 和 D. J. Wineland。量子门的随机化基准测试。Phys. Rev. A，77: 012307，2008年1月。doi:10.1103/PhysRevA.77.012307。URL https://link.aps.org/doi/10 0 12307 1103 77 . /PhysRevA. . .[37] Easwar Magesan、J. M. Gambetta 和 Joseph Emerson。可扩展且稳健的量子过程随机化基准测试。Phys. Rev. Lett.，106:180504，2011年5月。doi: 10.1103/PhysRevLett.106.180504。URL https://link.aps .org/doi/10.1103/PhysRevLett.106.180504。[38] Robin Blume-Kohout、John King Gamble、Erik Nielsen、Kenneth Rudinger、Jonathan Mizrahi、Kevin Fortier 和 Peter Maunz。通过门集层析证明量子比特操作达到严格容错阈值以下。Nat. Com- mun.，8:14485，2017年2月。doi:10.1038/ncomms14485。URL https://www.nature.com/articles/ncomms1448 5。[39]Jay M Gambetta、AD Córcoles、Seth T Merkel、Blake R Johnson、John A Smolin、Jerry M Chow、Colm A Ryan、Chad Rigetti、S Poletto、Thomas A Ohki 等。通过同时进行随机化基准测试来表征可寻址性标记。《物理评论快报》，109(24):240504，2012年。网址：https://journals.aps.org/prl/abstract/10 110 3 109 240504 . / PhysRevLett. . .[40] Mohan Sarovar、Timothy Proctor、Kenneth Rudinger、Kevin Young、Erik Nielsen 和 Robin Blume-Kohout。检测量子信息处理器中的串扰错误。Quantum，4:321，2020年。网址：https://quantum-journal.org/pap ers/q-2020-09-11-321/.[41] Timothy Proctor、Stefan Seritan、Kenneth Rudinger、Erik Nielsen、Robin Blume-Kohout 和 Kevin Young。使用镜像电路对量子计算机进行可扩展的随机化基准测试。《物理评论快报》，129(15)，2022年10月。ISSN 1079-7114。doi:10.1103/physrevlett.129.150502。网址：http: //dx.doi.org/10.1103/PhysRevLett.129.150502。[42] David C. McKay、Ian Hincks、Emily J. Pritchett、Malcolm Carroll、Luke C. G. Govia 和 Seth T. Merkel。大规模基准测试量子处理器性能，2023年。网址：https: //arxiv.org/abs/2311.05933。[43] Andrew W. Cross、Lev S. Bishop、Sarah Sheldon、Paul D. Nation 和 Jay M. Gambetta。使用随机化模型电路验证量子计算机。《物理评论A》，100(3)，2019年9月。ISSN 2469-9934。doi:10.1103/physreva.100.032328。网址：http://dx.doi.org/10 03 2328 1103 100 . /PhysRevA. . .[44] Qiskit 团队。测量量子体积，2021年8月。网址：https://qiskit.org/textbook/ch-quantum-hardwar e/measuring-quantum-volume.html。[45] Robin Blume-Kohout 和 Kevin C. Young。量子计算机基准测试的体积框架。Quantum，4:362，2020年11月。ISSN 2521-327X。doi:10.22331/q-2020- 11 1 1 15 362 10 22331 2020 15 362- - . 网址：https://doi.org/ . /q- - - - .[46] Timothy Proctor、Kenneth Rudinger、Kevin Young、Erik Nielsen 和 Robin Blume-Kohout。测量量子计算机的能力，2020年。[47] Mark Johnson、Mohammad Amin、S Gildert、Trevor Lant-ing、F Hamze、N Dickson、R Harris、Andrew Berkley、Jan Johansson、Paul Bunyk、E Chapple、C Enderud、Jeremy Hilton、Kamran Karimi、E Ladizinsky、Nicolas Ladizinsky、T Oh、I Perminov、C Rich 和 Geordie Rose。使用人造自旋进行量子退火。《自然》，473:194–8，2011年5月。doi:10.1038/nature10012。[48] Alejandro Perdomo-Ortiz、Alexander Feldman、Asier Oza-eta、Sergei V. Isakov、Zheng Zhu、Bryan O'Gorman、Hel-mut G. Katzgraber、Alexander Diedrich、Hartmut Neven、Jo-han de Kleer、Brad Lackey 和 Rupak Biswas。量子优化机器用于工业应用的准备情况。Physical Review Applied，12(1)，2019年7月。doi: 10.1103/physrevapplied.12.014004。网址：https://doi.or g/10.1103%2Fphysrevapplied.12.014004。[49] Bikas K Chakrabarti 和 Sudip Mukherjee。量子退火与计算，2022 22 03 15839。网址：https://arxiv.org/abs/ . .[50] Neil G. Dickson、M. William Johnson、Mohammad H. S. Amin、R. Harris、F. Altomare、Andrew J. Berkley、Paul I. Bunyk、J. Cai、E. M. Chapple、P Chavez、Florentin Cioat˘a、T Cirip、P Debuen、Marshall Drew-Brook、C. Enderud、S. Gildert、Firas Hamze、Jeremy P. Hilton、E. Hoskinson、Kamran Karimi、Eric Ladizinsky、Nicolas Ladizinsky、Trevor Lanting、Timothy Ma-hon、Richard Bryon Neufeld、Travis Oh、I. G. Perminov、C P Petroff、Anthony J. Przybysz、Chris Rich、P. Spear、Adi Tcaciuc、Murray C. Thom、Elena Tolkacheva、Sergey Uchaikin、J. Wang、A. B. Wilson、Zeeya Merali 和 Geordie Rose。对一个16量子比特问题进行热辅助量子退火。《自然》

通讯，4:1903，2013年。[51] T. Lanting、A. J. Przybysz、A. Yu. Smirnov、F. M. Spedalieri、M. H. Amin、A. J. Berkley、R. Harris、F. Altomare、S. Boixo、P. Bunyk、N. Dickson、C. Enderud、J. P. Hilton、E. Hoskin-son、M. W. Johnson、E. Ladizinsky、N. Ladizinsky、R. Neufeld、T. Oh、I. Perminov、C. Rich、M. C. Thom、E. Tolkacheva、S. Uchaikin、A. B. Wilson 和 G. Rose。在量子退火处理器中的纠缠。《物理评论X》，4(2)，2014年5月。doi:10.1103/physrevx.4.021041。网址：https: //doi.org/10.1103%2Fphysrevx.4.021041。[52] Tristan Zaborniak 和 Rogério de Sousa。对 D-Wave 量子退火器中的哈密顿量噪声进行基准测试。IEEE量子工程汇刊，2:1–6，2021年。doi: 10.1109/TQE.2021.3050449。[53] Jon Nelson、Marc Vuffray、Andrey Y. Lokhov 和 Carleton Coffrin。量子退火硬件的单量子比特保真度评估。IEEE量子工程汇刊，2:1–10，2021年。doi:10.1109/TQE.2021.3092710。[54] Marc Vuffray、Carleton Coffrin、Yaroslav A. Kharkov 和 Andrey Y. Lokhov。可编程量子退火器作为含噪吉布斯采样器。PRX Quantum，3:020317，2022年4月。doi: 10.1103/PRXQuantum.3.020317。网址：https://link.aps .org/doi/10.1103/PRXQuantum.3.020317。[55] Daniel C Murphy 和 Kenneth R Brown。通过控制误差方向提高量子算法成功率。《物理评论A》，99(3):032318，2019年。网址：https://journals.aps .org/pra/abstract/10.1103/PhysRevA.99.032318。[56] James King、Sheir Yarkoni、Mayssam M. Nevisi、Jeremy P. Hilton 和 Catherine C. McGeoch。使用达到目标时间指标对量子退火处理器进行基准测试，2015年。网址：https://arxiv.org/abs/1508.05087。[57] David Subires、Fernando J. Gó mez-Ruiz、Antonia Ruiz-García、Daniel Alonso 和 Adolfo del Campo。量子退火动力学的基准测试：自旋矢量朗之万模型。Physical Review Research，4(2)，2022年5月。doi: 10.1103/physrevresearch.4.023104。网址：https://doi.org/ 10.1103%2Fphysrevresearch.4.023104。[58] Antika Sinha。基于 Google Scholar 数据构建量子退火计算与信息研究网络，2022年。网址：https://arxiv.org/abs/2206.02176。[59] S. Kirkpatrick、C. D. Gelatt 和 M. P. Vecchi。模拟退火优化。Science，220(4598):671–680，1983年。doi:10.1126/science.220.4598.671。网址：https://www.science.org/doi/abs/10.1126/science.220.4598.671。[60] Charlie J. Geyer。并行回火。载于 E. M. Keramidas 和 S. M. Kaufman 编，《计算科学与统计：第23届交叉领域研讨会论文集》，第156页，纽约，1991年。美国统计学会。[61] Bor1928Zheng Zhu、Andrew J. Ochoa 和 Helmut G. Katz-graber。适用于任意空间维数中自旋玻璃的高效团簇算法。《物理评论快报》，115:077201，2015年8月。doi:https://doi.org/10.1103/PhysRevLett.115.077201。网址：https://link.aps.org/doi/10.1103/PhysRevLett .115.077201。[62] Carleton Coffrin、Harsha Nagarajan 和 Russell Bent。使用整数规划评估伊辛处理单元。载于 Louis-Martin Rousseau 和 Kostas Stergiou 编，《约束规划、人工智能与运筹学的融合》，第163–181页，Cham，2019年。Springer International Publishing。ISBN 978-3-030-19212-9。[63] Itay Hen、Joshua Job、Tameem Albash、Troels F. Røn-now、Matthias Troyer 和 Daniel A. Lidar。利用植入解探测自旋玻璃问题中的量子加速。《物理评论A》，92:042325，2015年10月。doi: 10.1103/PhysRevA.92.042325。网址 https://link.aps .org/doi/10.1103/PhysRevA.92.042325。[64] Dilina Perera、Inimfon Akpabio、Firas Hamze、Salvatore Man- dra、Nathan Rose、Maliheh Aramon 和 Helmut G. Katz- graber。《Chook——用于生成具有植入解的二元优化问题的综合套件》，2020年。网址 https://arxiv.org/abs/2005.14344。[65] Matthew Kowalsky、Tameem Albash、Itay Hen 和 Daniel A. Lidar。《3-正则3-XORSAT植入解：经典与量子启发式优化器的基准测试》，2021年。doi:10.1088/2058- 9565/ac4d1b。[66] Ang Li、Samuel Stein、Sriram Krishnamoorthy 和 James Ang。《Qasmbench：用于NISQ评估和模拟的低级QASM基准测试套件》，2020年。网址 https://arxiv.org/ abs/2005.13018。[67] Teague Tomesh、Pranav Gokhale、Victory Omole、Gokul Sub- ramanian Ravi、Kaitlin N. Smith、Joshua Viszlai、Xin-Chuan Wu、Nikos Hardavellas、Margaret R. Martonosi 和 Frederic T. Chong。《SupermarQ：可扩展量子基准测试套件》，2022年。网址 https://arxiv.org/abs/2202.11045。[68] Huub Donkers、Koen Mesman、Zaid Al-Ars 和 Matthias Möller。《Qpack Scores：面向应用的量子计算机基准测试定量性能指标》，2022年。网址 https://arxiv.org/abs/2205.12142。[69] Jernej Rudi Finžgar、Philipp Ross、Leonhard Hölscher、Jo- hannes Klepsch 和 Andre Luckow。《Quark：量子计算应用基准测试框架》，2022年。网址 https://arxiv.org/abs/2202.03028。[70] Ward van der Schoot、Daan Leermakers、Robert Wezeman、Niels Neumann 和 Frank Phillipson。《评估量子退火机的Q-score》。载于2022年IEEE量子软件国际会议（QSW）。IEEE，2022年7月。doi: 10.1109/qsw55613.2022.00017。网址 https://doi.org/10 .1109%2Fqsw55613.2022.00017。[71] Blake Johnson 和 Ismael Faro。《IBM Quantum借助Qiskit Runtime使量子工作负载提速120倍》，2021年5月。网址 https://research.ibm.com/blog/120x-quantum -speedup?lnk=ushpv18re2。[72] K. Bertels、A. Sarkar、T. Hubregtsen.、M. Serrao、A. A. Moue- denne、A. Yadav、A. Krol 和 I. Ashraf。《量子计算机体系结构：迈向全栈量子加速器》。2020年欧洲设计、自动化与测试会议暨展览会（DATE），2020年3月。doi:10.23919/date48585.2020.9116502。网址 http://dx.doi.org/10.23919/DATE48585.2020 .9116502。[73] Matthias Möller 和 Cornelis Vuik。《量子计算技术对未来高性能科学计算发展的影响》。Ethics and Information Technology，19(4):253–269，2017年8月。ISSN 1572-8439。doi: 10.1007/s10676-017-9438-0。网址 http://dx.doi.org/10.1007/s10676-017-9438-0。[74] Yudong Cao 和 Timothy Hirzel。《2020年的量子加速》，2020年12月。网址 https://www.infoq.com/articles /quantum-acceleration-2020/。[75]Frank Arute、Kunal Arya、Ryan Babbush、Dave Bacon、Joseph C Bardin、Rami Barends、Rupak Biswas、Sergio Boixo、Fernando G S L Brandao、David A Buell、Brian Burkett、Yu Chen、Zijun Chen、Ben Chiaro、Roberto Collins、William Courtney、Andrew Dunsworth、Edward Farhi、Brooks Foxen、Austin Fowler、Craig Gidney、Marissa Giustina、Rob Graff、Keith Guerin、Steve Habegger、Matthew P Harrigan、Michael J Hartmann、Alan Ho、Markus Hoffmann、Trent Huang、Travis S Humble、Sergei V Isakov、Evan Jeffrey、Zhang Jiang、Dvir Kafri、Kostyantyn Kechedzhi、Julian Kelly、Paul V Klimov,

Sergey Knysh、Alexander Korotkov、Fedor Kostritsa、David Landhuis、Mike Lindmark、Erik Lucero、Dmitry Lyakh、Salvatore Mandrà、Jarrod R McClean、Matthew McEwen、An- t hony Megrant、Xiao Mi、Kristel Michielsen、Masoud Mohseni、Josh Mutus、Ofer Naaman、Matthew Neeley、Charles Neill、Murphy Yuezhen Niu、Eric Ostby、Andre Petukhov、John C Platt、Chris Quintana、Eleanor G Rieffel、Pedram Roushan、Nicholas C Rubin、Daniel Sank、Kevin J Satzinger、Vadim Smelyanskiy、Kevin J Sung、Matthew D Trevithick、Amit Vainsencher、Benjamin Villalonga、Theodore White、Z Jamie Yao、Ping Yeh、Adam Zalcman、Hartmut Neven 和 John M Martinis。《使用可编程超导处理器实现量子霸权》。Nature，574(7779):505–510，2019年。ISSN 0028-0836，1476-4687。doi:10.1038/s41586-019-1666-5。网址 http://dx.doi.org/10.1038/s41586-019-1666-5。[76] Han-Sen Zhong、Hui Wang、Yu-Hao Deng、Ming-Cheng Chen、Li-Chao Peng、Yi-Han Luo、Jian Qin、Dian Wu、Xing Ding、Yi Hu、Peng Hu、Xiao-Yan Yang、Wei-Jun Zhang、Hao Li、Yuxuan Li、Xiao Jiang、Lin Gan、Guangwen Yang、Lixing You、Zhen Wang、Li Li、Nai-Le Liu、Chao-Yang Lu 和 Jian- Wei Pan。《利用光子实现量子计算优势》。Science，370(6523):1460–1463，2020年。ISSN 0036-8075，1095- 9203。doi:10.1126/science.abe8770。网址 http://dx.doi.o rg/10.1126/science.abe8770。[77] Juneseo Lee、Alicia B. Magann、Herschel A. Rabitz 和 Christian Arenz。《量子组合优化中趋向有利景观的进展》。《物理评论A》，104(3)，2021年9月。doi:10.1103/physreva.104.032401。网址 https://doi.org/10.1103%2Fphysreva.104.032401。[78] Jonathan Ward、Johannes Otterbach、Gavin Crooks、Nicholas Rubin 和 Marcus da Silva。《使用最大割对量子近似优化算法进行性能基准测试》。载于美国物理学会三月会议摘要，2018年卷，APS会议摘要，第R15.007页，2018年1月。[79] Gavin E. Crooks。《量子近似优化算法在最大割问题上的性能》，2018年。网址 https://arxiv.org/abs/1811.08419。[80] Byron Tasseff、Tameem Albash、Zachary Morrell、Marc Vuffray、Andrey Y. Lokhov、Sidhant Misra 和 Carleton Coffrin。《量子退火硬件在组合优化中的潜在优势初现》，2022年。网址 https://arxiv.org/abs/2210.04291。[81] Hristo N. Djidjev、Guillaume Chapuis、Georg Hahn 和 Guil- laume Rizk。《利用量子退火实现高效组合优化》，2018年。网址 https://arxiv.org/abs/1801 .08653。[82] Tameem Albash 和 Daniel A. Lidar。《展示量子退火器相对于模拟退火的尺度优势》。《物理评论X》，8(3)，2018年7月。doi:10.1103/physrevx.8.031016。网址 https://doi.org/10 .1103%2Fphysrevx.8.031016。[83] S. Ebadi、A. Keesling、M. Cain、T. T. Wang、H. Levine、D. Bluvstein、G. Semeghini、A. Omran、J.-G. Liu、R. Sama- jdar、X.-Z. Luo、B. Nash、X. Gao、B. Barak、E. Farhi、S. Sachdev、N. Gemelke、L. Zhou、S. Choi、H. Pichler、S.-T. Wang、M. Greiner、V. Vuleti´c 和 M. D. Lukin。《利用里德堡原子阵列对最大独立集进行量子优化》。Science，376(6598):1209–1215，2022年。doi: 10.1126/science.abo6587。网址 https://www.science.org/doi/abs/10.1126/science.abo6587。[84] Troels F. Rønnow、Zhihui Wang、Joshua Job、Sergio Boixo、Sergei V. Isakov、David Wecker、John M. Martinis、Daniel A. Lidar 和 Matthias Troyer。《定义并检测量子加速》。Science，345(6195):420–424，2014年7月。doi: 10.1126/science.1252319。网址 https://doi.org/10.112 6%2Fscience.1252319。

[85] Salvatore Mandrà 和 Helmut G Katzgraber. 迈向量子加速检测的一个误导性步骤。《量子科学与技术》，3(4):04LT01，2018年7月。doi:10.1088/2058- 9565 9 565 8 2 10 1088 2 2058 2 8 2/aac b . URL https://doi.org/ . % F - % Faac b .[86] NetworkX。NetworkX——Python 网络分析。https: // networkx.org/，2023。NetworkX 网站。[87] Gurobi。Gurobi 优化。https://www.gurobi.com/，2023。Gurobi 网站。[88] Madita Willsch、Dennis Willsch、Fengping Jin、Hans De Raedt 和 Kristel Michielsen。量子近似优化算法的基准测试。《量子信息处理》，19(7):197，2020年7月。ISSN 1570-0755，1573-1332。doi: 10.1007/s11128-020-02692-8。[89] Panagiotis Kl Barkoutsos、Giacomo Nannicini、Anton Robert、Ivano Tavernelli 和 Stefan Woerner。使用条件风险价值改进变分量子优化。《Quantum》，4:256，2020年4月。doi:10.22331/q-2020-04-20-256。[90] Li Li、Minjie Fan、Marc Coram、Patrick Riley 和 Stefan Leichenauer。采用新型吉布斯目标函数和试探态架构搜索的量子优化。《物理评论研究》，2(2):023074，2020年4月。doi: 10.1103/PhysRevResearch.2.023074。[91] IBM Cloud Qiskit Runtime。https://cloud.ibm.com/qu antum，2023。访问日期: 2023-05-15。[92] Endre Boros、Peter L Hammer 和 Gabriel Tavares。二次无约束二元优化（QUBO）的局部搜索启发式算法。《启发式杂志》，13(2):99–132，2007。[93] Gary Kochenberger、Jin-Kao Hao、Fred Glover、Mark Lewis、Zhipeng Lü、Haibo Wang 和 Yang Wang。无约束二次二元规划问题：综述。《组合优化杂志》，28(1):58–81，2014年。[94] M. Born 和 V. Fock。绝热定理的证明。《物理学杂志》，51(3):165–180，1928。doi:10.1007/BF01343193。URL https://doi.org/10.1007/BF01343193。[95] Tosio Kato。论量子力学的绝热定理。《日本物理学会杂志》，5(6):435–439，1950年。doi:10.1143/JPSJ.5.435。URL https://doi.org/10.1143/JPSJ.5.435。[96] Sabine Jansen、Mary-Beth Ruskai 和 Ruedi Seiler。绝热近似的界及其在量子计算中的应用。《数学物理学杂志》，48 102111，2007 10 10 1063 1 2798382 10 ( ): 。doi: . / . . URL https://doi.org/ .1063/1.2798382。[97] Vicky Choi。绝热量子计算中的小图嵌入：I. 参数设置问题。《量子信息处理》，7:193–209，2008年。[98] Andrew Wack、Hanhee Paik、Ali Javadi-Abhari、Petar Jurce- vic、Ismael Faro、Jay M. Gambetta 和 Blake R. Johnson。质量、速度和规模：衡量近期量子计算机性能的三个关键属性，2021年。URL https://arxiv.org/abs/2110.14108。[99] Lennart Bittel 和 Martin Kliesch。训练变分量子算法是 NP-困难问题。《物理评论快报》，127(12):120502，2021年9月。doi: 10.1103/PhysRevLett.127.120502。[100] Jarrod R. McClean、Sergio Boixo、Vadim N. Smelyanskiy、Ryan Babbush 和 Hartmut Neven。量子神经网络训练景观中的贫瘠高原。《自然-通讯》，9(1):4812，2018年11月。ISSN 2041-1723。doi: 10.1038/s41467-018-07090-4。[101] M. Cerezo、Akira Sone、Tyler Volkoff、Lukasz Cincio 和 Patrick J. Coles。依赖于代价函数的贫瘠高原

存在于浅层参数化量子电路中。《自然-通讯》，12(1):1791，2021年3月。ISSN 2041-1723。doi: 10.1038/s41467-021-21728-w。[102] Samson Wang 、 Enrico Fontana 、 M. Cerezo 、 Kunal Sharma、Akira Sone、Lukasz Cincio 和 Patrick J. Coles。变分量子算法中的噪声诱导贫瘠高原。《自然-通讯》，12(1):6961，2021年11月。ISSN 2041- 1723。doi:10.1038/s41467-021-27045-6。[103] Carlos Ortiz Marrero、Mária Kieferová 和 Nathan Wiebe。纠缠诱导的贫瘠高原。（arXiv:2010.15968），2021年3月。doi:10.48550/arXiv.2010.15968。[104] 量子近似优化算法在正则最大割图上的固定角度猜想。Jonathan Wurtz 和 Danylo Lykov。《物理评论*A*》，104(5):052419，2021年11月。doi:10.1103/PhysRevA.104.052419。[105] V. Akshay、D. Rabinovich、E. Campos 和 J. Biamonte。量子近似优化中的参数集中现象。《物理评论*A*》，104(1):L010401，2021年7月。ISSN 2469- 9926，2469-9934。doi:10.1103/PhysRevA.104.L010401。[106] David Bernal Neira、Davide Venturelli、Filip Wudarski 和 Eleanor Rieffel。量子启发式算法和伊辛机运行情况的基准测试：针对真实世界优化应用评估参数设置策略。载于 美国物理学会三月会议摘要，第2022卷，第F38–005页，2022年。[107] 随机基准测试。https://github.com/usra-riacs/stochastic-benchmark，2022年。

附录 A：组合优化方法

我们将概述组合优化问题的理论基础，以及这些基础对开发用于研究求解器性能的硬件演示的影响（求解器是在软件或硬件中实现的算法或启发式方法）。此处讨论的问题为我们在设计 QED-C 基准测试框架时选择输入和性能指标提供了依据。

1. 组合优化理论

具体而言，我们考虑定义在n 个整数值变量$x = \{x_1, \ldots x_n\}$ 上的一类组合优化问题，其中包含 m 个约束函数 $c(x)$ 和一个目标函数f；该目标函数是x的多项式，具体如下。

$$\begin{aligned} &\min : f(x) \\ &\text{s.t.: } c_i(x) \le 0 \quad \forall i \in \{1, \ldots, m\} \\ &x_i \in \mathbb{Z} \quad \forall i \in \{1, \ldots, n\} \end{aligned} \tag{A1}$$

对于上述问题，算法型目标是为x 找到一组整数值赋值，使其满足所有约束，并使$f(x)$的值最小。例如，下面这个简单问题

$$\begin{aligned} f(x) &= x_1 + 2x_2 \\ c_1(x) &: -x_1 + 1 \le 0 \\ c_2(x) &: -x_2 + 1 \le 0, \end{aligned}$$

要求找出两个正整数，使$f(x)$最小；一个最优解$x_1 = 1$、$x_2 = 1$ 的目标值为$f(x) = 3$。

这种表示法足够通用，能够涵盖所有行业领域关注的各种优化问题。以下仅举几例：

- 在工业调度问题中，**作业车间调度** 问题及其各种变体十分常见，涉及如何高效地将多种资源分配给多项任务。
- **投资组合优化** 问题是金融领域关注的问题。例如，给定一份待购买商品清单，选择其中一部分商品，以使利润最大化并将风险降至最低。
- 交通运输行业中的**机场登机口调度** 问题如下：给定一系列机场到达时间和乘客中转信息，为飞机分配登机口，以尽量缩短乘客步行前往中转登机口的总距离。
- **机器学习（ML）**是人工智能的核心问题。大多数 ML 技术都需要在其内部流程中使用高质量的优化启发式方法

  可能需要执行数百次或数千次的循环计算。该启发式方法会找到输入/输出对，构成给定优化问题近似最优解空间的多样化样本。

复杂性类 NP-OPT 包含优化问题（包括 (A1)），这类问题以具有数值结果的目标函数来定义；与之相对的是结果为二元值（例如“是/否”或“真/假”）的判定问题，后者属于更著名的 NP 类。

NP 中的每个问题 P 都可以重新表述（也称为转换）为 NP-OPT 中的问题 P-OPT。例如，在二元可满足性问题（SAT）中，给定布尔表达式 B，是否存在一种变量赋值，使得 B 的计算结果为真？可以转换为 NP-OPT 中的等价问题：给定 B-OPT，找出一种变量赋值，使满足的子句数量最大化。该转换保证 SAT-OPT 的最优解对应于 SAT 的肯定答案。在这种情况下，满足的子句最大数量等于$B - OPT$中的子句总数，此时二元问题的答案为是。

转换后的 SAT-OPT 问题称为 NP-困难，因为一个求解 SAT-OPT 的多项式时间算法可用于解决 SAT，进而使 NP 中的每个问题都能在多项式时间内解决。既是 NP-困难又属于 NP 的问题（即二元判定问题）称为 NP-完全。著名的悬而未决问题 P 是否 $=$ 等于 NP？体现了当前对这些问题的认知困境：目前尚无已知的多项式时间算法，也没人能证明这样的算法不可能存在。

通过转换求解问题在求解 NP 和 NP-OPT 中的问题时，有许多方法；对于少数问题和算法，人们研究了通过转换求解的方式。

当针对目标问题 T 的单个求解器可用于解决实践中出现的各种问题时，这种方法对从业者很有吸引力：也就是说，将各个输入转换为符合 T 的表述所需的开销，低于为每个新出现的问题分别实现专用求解器的开销。

这种方法中研究最广泛的版本涉及一类表示为 (A1) 的问题，称为整数线性问题。当目标函数 $f(x)$ 和约束 $c_i(x)$ 均为线性时，这类问题可在多项式时间内求解。另一种常见方法是将问题转换为 SAT 或 SAT-OPT；有时可以为这些问题找到高效的启发式算法。

二次无约束二元优化（QUBO）问题也被视为一种通用的目标表示形式，尤其适用于定义在图上的问题；在量子计算兴起之前，这种做法就已存在 [92, 93]。能够在硬件上原生求解 QUBO 的量子退火处理器问世后，QUBO 及其变体——伊辛模型（IM）问题——近来备受关注，后者常用于物理学应用。这两个问题完全相同，区别仅在于变量取值域有所不同：二元变量 $x \in \{0, 1\}$（QUBO）与自旋变量 $s \in \{-1, +1\}$（IM）。

NP 完全性理论告诉我们，原则上，任何可表述为 (A1) 的问题输入，都可以在多项式时间内转换为一种可直接使用量子计算机求解的形式。有关如何将由 (A1) 表示的一般优化问题表述为 QUBO 和伊辛模型的教程，请参见 [29, 31]。然而，由于规模较小，

目前在量子平台上测试的问题受到很大限制。

附录 B：优化问题的量子启发式算法

该基准测试框架评估了两种用于求解组合优化问题的主流量子启发式算法的性能特征：量子近似优化算法（QAOA），它使用门模型量子计算机；以及量子退火（QA），它使用模拟量子计算机。本文对这些算法在求解最大割问题中的应用进行了基准测试。

最大割问题的输入是一个由节点或顶点（V）和边（E）构成的无向图。通常，图中的每条边都可以带有“权重”，但本文只考虑无权 3-正则图。一个割将图的节点划分为两个集合。其规模定义为连接属于不同集合的节点的图边数量。最大割问题旨在从所有可能的割中找出规模最大的一个。

最大割已成为量子优化的一项热门基准测试 [12, 25–27]原因有二：（1）即便只是求得近似解，它也是最具挑战性的组合优化任务之一，即 APX-困难 [13, 28]，（2）作为一项无约束离散优化任务，它可以自然地编码为二次无约束二元优化（QUBO） [29, 30] 或伊辛模型 [5, 31]，非常适合当前的量子优化算法（QAOA、QA）。虽然最大割为当前方法的基准测试提供了一个合理的起点，但未来的工作还需测试更复杂的优化任务，包括带约束的问题，从而证明量子加速优化能够影响广泛的优化应用。

1. 量子近似优化算法

量子近似优化算法 [6]可以说是使用门模型量子处理器求解组合优化问题的主要候选算法。QAOA 属于变分量子算法（VQA） [35]，通常以迭代方式实现：经典优化器对参数化量子电路进行“训练”。QAOA 是一种启发式方法，旨在求解 QUBO问题等组合优化问题。具体而言，问题被编码为二元变量的指定二次函数，目标是找到这些变量的一组赋值，使函数值最小。

量子近似优化算法的核心是“试探电路”，即一种参数化量子电路。在电路末端进行的计算基测量，对应于从问题可能答案的概率分布中抽样。经典优化器会获取一组参数值，使其有很大概率产生最优解或近似最优解。最后，使用优化器确定的参数值反复测量该电路，

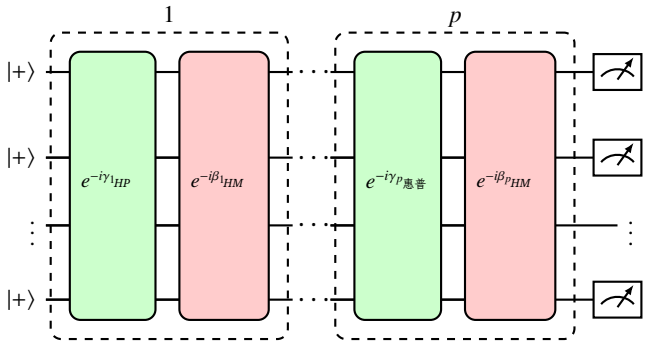


图 14：**QAOA电路**由p 个重复的参数化模块组成。首先，对每个量子比特施加哈达玛门，以获得 $|+\rangle$ 态。每个模块还包含一次涉及问题哈密顿量H_P的旋转，随后是一次涉及混合哈密顿量H_M的旋转。最后，在计算基 $\{|0\rangle, |1\rangle\}$中测量所有量子比特。

即可得到问题的近似解。

首先将问题编码为哈密顿量H_P，使得问题的最优解对应于基态或最低能量本征态。对于给定的“轮数”p，量子近似优化算法试探态定义为

$$|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_P} \dots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_P} |+\rangle, \tag{B1}$$

其中 $H_M = \sum_i X_i$是所谓的混合哈密顿量，而$|+\rangle = \otimes_i H|0\rangle$是等幅叠加态。在这里，$X$ 是泡利 X 矩阵，定义为 $X|0\rangle = |1\rangle$ 以及 $X|1\rangle = |0\rangle$。因此，试探态是通过围绕 H_P 和 H_M 进行重复交替旋转而获得的，如 图 14所示。

最大割问题可以表述为求解以下哈密顿量的基态

$$H_P = \frac{-1}{2} \sum_{\langle i,j\rangle \in E} (1 - Z_i Z_j), \tag{B2}$$

其中 E 表示图的（无向）边集合，而Z是满足以下条件的泡利 Z 矩阵 $Z|0\rangle = |0\rangle$ 以及 $Z|1\rangle = -|1\rangle$。每个计算基矢量对应一种可能的割，其能量表示割大小的负值。注意，H_P 的特征值都是非负整数，并且 公式 B2 对应于一个所有耦合常数均设为 1/2 的伊辛模型。

通常会计算一个称为近似比的量，以刻画解的质量。近似比r 定义为能量期望值$F_{\beta,\gamma} := \langle \boldsymbol{\beta}、\boldsymbol{\gamma}| H_P |\boldsymbol{\beta}、\boldsymbol{\gamma}\rangle$与基态能量值$E_{\min}$之比：

$$r_{\beta,\gamma} = \frac{F_{\beta,\gamma}}{E_{\min}} = \frac{\langle \boldsymbol{\beta}, \boldsymbol{\gamma}| H_P |\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle}{E_{\min}}. \tag{B3}$$

请注意，分子小于或等于零，而分母（即最大割大小的相反数）严格小于零。因此，$0 \le r \le 1$。经典优化器程序旨在获得角度 $\boldsymbol{\beta}$ 的最优值

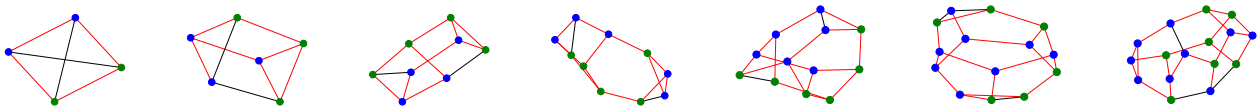


图15：**为基准实现选取的图实例**。对于从 4 到 16、间隔为 2 的每个问题规模，我们都使用 QAOA 和 QA，对一个具有相应节点数的 3-正则图求解最大割问题。（对于 QA，还生成了节点数最多达 320 的较大图。）这些图通过不同颜色的节点和边展示了最大割问题的一个解。不同颜色的节点属于解割的两个集合。连接不同集合中节点的红色边的数量，就是该图的最大割。

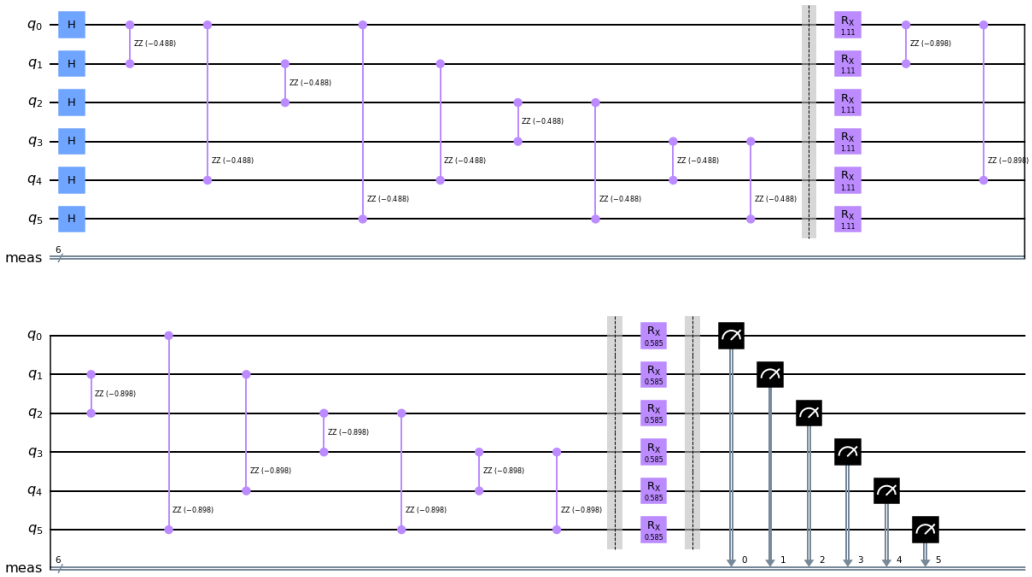


图16：**量子近似优化算法试探电路图示例**。此处展示的量子电路是为上方所示的、含 6 个变量的最大割问题构建的试探态，使用 6 个量子比特实现，共 2 轮。两组各包含 9 个参数化RZZ门，将图中的边映射到电路中，并表示问题哈密顿量。两组参数化RX门表示混合哈密顿量。此电路实现了图 14 中针对基准测试所用特定图的内容。

和 $\boldsymbol{\gamma}$，即对应于最高近似比的值。$F_{\beta,\,\gamma}$，因此 $r_{\beta,\,\gamma}$ 无法精确计算，而是通过在电路末端的计算基中进行 $|\boldsymbol{\beta},\boldsymbol{\gamma}\rangle$ 多次（例如 M 次）测量（即测量次数）来近似（见图 14）。具体而言，$F_{\beta,\,\gamma}$ 通过能量经验平均值进行近似。

2. 量子退火

通过量子退火，优化问题会被编码到机器中，随后通过量子绝热演化确定解，从而得到近最优末态。量子退火的算法型方法是利用量子系统的动态演化，将一个初始基态（易于制备）转变为目标哈密顿量的基态，而该哈密顿量未知，且难以通过其他方式计算。从较高层面来看，该协议旨在通过对用户指定的H^{Target} 模型执行模拟插值过程，演化以下哈密顿量：

$$H(s) = (1-s)H_{\text{Init}} + (s)H_{\text{Target}}. \tag{B4}$$

插值过程从$s = 0$ 和基态开始：H_{Init}。退火过程包括对s 从 0 平滑插值到 1。对于足够长的退火时间，绝热定理表明，量子系统会保持在插值哈密顿量的最小本征向量 $H(s)$ [94–96], 因此，在演化结束时到达H^{Target} 的最低能量态。

目前，可用的量子退火硬件主要针对方程 B4 的一种特殊情况，其仅限于横向场伊辛模型，

$$H(s) = A(s)\left(\sum_i X_i\right) + B(s)\left(\sum_i h_i Z_i + \sum_{i,j} J_{ij} Z_i Z_j\right). \tag{B5}$$

其中X_i 表示作用于量子比特i的泡利X算符，Z_i表示作用于量子比特i的泡利Z算符，而Z_iZ_j 表示作用于量子比特i 和j上的Z算符的张量积。两个插值函数$A(s)$和$B(s)$控制着从一种状态到另一种状态的过渡，

从强H_{Init}和弱H_{target}过渡到弱H_{Init}和强H_{target}；也就是说，$A(0) \gg B(0)$，且$A(1) \ll B(1)$。硬件通过这些函数实现默认的退火“路径”，用户参数可以对其进行修改。这样，硬件和QA算法便能通过参数h和J，在Z基上寻找用户指定的经典伊辛模型的基态和低能态，其中h和J分别编码局域场和耦合强度。请注意，本文所研究的最大割问题通过设置$h = 0$和$J_{ij} = +1$来编码到该模型中，适用于给定最大割图中出现的每条边$(i, j) \in E$。如果最大割图无法直接编码到量子退火硬件中（即，最大割问题的边集不是量子退火硬件中Z_iZ_j项所构成的子图），则会使用一种称为次嵌入 [97] 的过程，将最大割问题映射为数学上等价且适用于硬件的原生问题。

值得注意的是，B1 节 中概述的 QAOA 算法可以解释为公式 B5 的特罗特化版本，其中轮数p决定了特罗特阶数。也就是说，当$p \to \infty$时，QAOA 电路的极限可以模拟平滑的模拟量子退火转变。QA 的近似比计算方式与 QAOA 相同，即将退火后得到的样本转换为等效分布。

附录C：问题与实现细节

为生成本文所展示的结果，我们针对每种问题规模（或量子比特数）仅在一个问题实例上执行最大割基准测试。每种规模下的问题实例均为随机选取的 3-正则图。定义各问题实例的数据集存放在 QED-C 基准测试代码库中，地址为 https://github.com/SRI-International/QC-App-Oriented-Benchmarks。如有需要，可修改基准测试以使用其他图。

作为参考，在 图 15中，我们展示了用于不同规模问题的一些图。每个图都展示了最大割问题的一个解，使用不同颜色的节点和边表示，如图注所述。

在图 16中，我们展示了此基准测试中用于规模为 6（变量/量子比特数量）的问题所生成的 QAOA试探电路。图注说明了电路的各个组成部分如何表示定义该问题的哈密顿量。

下面展示了J 矩阵，该矩阵是根据方程 B5指定量子退火硬件的 6 变量最大割问题所必需的。

$$J = \begin{bmatrix} 0 & -1 & 0 & 0 & -1 & -1 \\ -1 & 0 & -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 & 0 & -1 \\ 0 & 0 & -1 & 0 & -1 & -1 \\ -1 & -1 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 & & \end{bmatrix} \tag{C1}$$

附录 D：执行时间分析

a. ***QAOA*** 中的执行时间QAOA 算法消耗的总时间包括量子和经典两部分。在量子处理器上执行量子试探电路需要一定时间。此外，还需要在经典处理器上运行最小化函数，根据每次执行试探态后获得的测量结果计算新的参数。

算法本身量子部分的执行时间可分解为几个组成部分。其中之一是“量子执行时间”，定义为在量子处理器中执行量子电路 N 次测量所需的时间。量子计算机硬件提供商通常会在结果记录中报告这一时间，其中包括执行前初始化量子系统所需的时间以及每次测量之间的延迟[98]，如 方程 D1所示。

$$t_{\text{quantum}} = t_{\text{init}} + N_{\text{shots}} \cdot (t_{\text{shot}} + t_{\text{delay}}) \tag{D1}$$

使用 QAOA 时，量子电路会在参数不断变化的情况下重复执行。电路执行的总时间，即“经过的量子执行时间”，包括执行前编译电路或应用参数所需的时间，以及验证并加载已编译电路以供执行所需的时间。另一个变化很大的组成部分是在队列中等待执行所耗费的时间。经过的量子执行时间定义见 方程 D2，并且必须作为基准测试算法的一部分进行采集，因为我们发现大多数系统都无法直接提供这一指标。

$$t_{\text{elapsed_quantum}} = t_{\text{queue}} + t_{\text{compile}} + t_{\text{load}} + t_{\text{quantum}} \tag{D2}$$

QAOA 算法所有迭代的量子时间与经过的量子时间之和，分别称为“累积量子执行时间”和“累计经过的量子执行时间”，定义见公式 D3和D4。在许多硬件系统中，量子计算的财务成本通常与累积量子执行时间相关。每次迭代的经过时间取决于执行参数，并可能受到系统是否支持参数化执行，或是否包含隐藏的经典后处理时间（例如误差缓解）的影响。

$$t_{\text{cum_quantum}} = \sum_{iter=1}^{N_{\text{iter}}} t_{\text{quantum(iter)}} \tag{D3}$$

$$t_{\text{cum_elapsed_quantum}} = \sum_{iter=1}^{N_{\text{iter}}} t_{\text{elapsed_quantum(iter)}} \tag{D4}$$

QAOA 的“经典执行时间”是指在特定迭代中，创建具有特定参数的试探态所需时间，以及最小化器处理测量结果并生成新参数所用时间之和迭代，如公式 D5所示。“累计经典执行时间”是所有迭代的经典执行时间之和，如公式 D6所示。对于小规模问题，这段时间通常

微不足道，但会随着问题规模和轮数的增加而增长。此外，它还可能受到系统支持参数化执行并缩短创建时间的能力影响。

$$t_{\text{classical}} = t_{\text{create}} + t_{\text{optimize}} \tag{D5}$$

$$t_{\text{cum_classical}} = \sum_{iter=1}^{N_{\text{iter}}} t_{\text{classical(iter)}} \tag{D6}$$

QAOA 的总执行时间是累计经过的量子时间$t_{\text{cum_elapsed_quantum}}$与经典计算时间$t_{\text{cum_classical}}$之和。由于处理选项或经典优化器的选择不同，结果质量和执行时间可能会有很大差异。

b. ***QA*** 中的执行时间 对于 QA，“量子执行时间”定义为量子处理单元（QPU）使用指定的退火时间执行 N 次读取（样本）所需的时间。硬件报告的这一时间称为“qpu_access_time”，其中包括 _programming_time”，即~ 16ms，以及“qpu_sampling_time”，即“退火_时间”加上“读出_时间”（~ 0。25ms），再乘以读取次数。量子执行时间的定义见方程 D7。

$$t_{\text{quantum}} = t_{\text{qpu_access}} = t_{\text{qpu_programming}} + t_{\text{qpu_sampling}} \tag{D7}$$

“经过的量子执行时间”包括向（远程）后端硬件系统发出采样命令、等待其完成并接收完全解析的样本集所需的时间。其定义见 方程 D8。它包括量子执行时间，以及计算输入的次嵌入以匹配 QPU 内部特定量子比特连接结构所需的时间，还有将解映射回原始（未嵌入）问题所需的时间。对于每个测试的退火时间，嵌入成本只测量一次；在实际应用中，这并非总是必要的，因为嵌入可以重复使用。

$$t_{\text{elapsed_quantum}} = t_{\text{queue}} + t_{\text{embed}} + t_{\text{sample}} + t_{\text{quantum}} + t_{\text{resolve}} \tag{D8}$$

对于量子退火，图2 中报告的累计时间反映了在t_{quantum} 不断增加的退火时间下进行的测量结果，这些时间如基准测试代码第 6 行所指定。$t_{\text{elapsed_quantum}}$指标包含执行退火操作并获取最终样本集所需的全部时间。这与量子近似优化算法中的累计时间相当。为展示两者的差异，我们以略有不同的样式可视化数据。每个条形代表一个不同的退火时间，并相对于前一个条形略微垂直错开，以表示其所代表的时间从 0 开始。有关量子退火硬件执行过程中收集的这些指标，请参见第 V C 节。

附录E：结果质量与超参数

优化算法的性能通常通过所获得结果的质量与实现该结果所需资源之间的权衡来研究。在许多实际应用中，需要在有限时间内获得高质量结果。我们希望预测能否在可用时间预算内获得质量可接受的解，并确定能够产生高质量输出的参数值。

不过，QAOA 的执行可以通过许多选项（即“超参数”）进行控制，因此不能仅根据有限探索所得的一组结果得出结论。尽管多种超参数（例如测量次数、经典优化器的选择、迭代次数和轮数）都会影响结果质量，本节将重点讨论初始角度取值对性能的影响。本节最后将介绍一种参数调优策略，以帮助确定适用于 QAOA 执行的优良超参数。（类似的策略也可以用于量子退火，但本文不作讨论。）

虽然本节是在最大割问题的语境下讨论 QAOA，但其中大多数结论对 QAOA 更普遍地适用。在本节中，我们交替使用“割大小”和“能量”这两个术语，其中能量指的是公式 B2中哈密顿量的特征值。

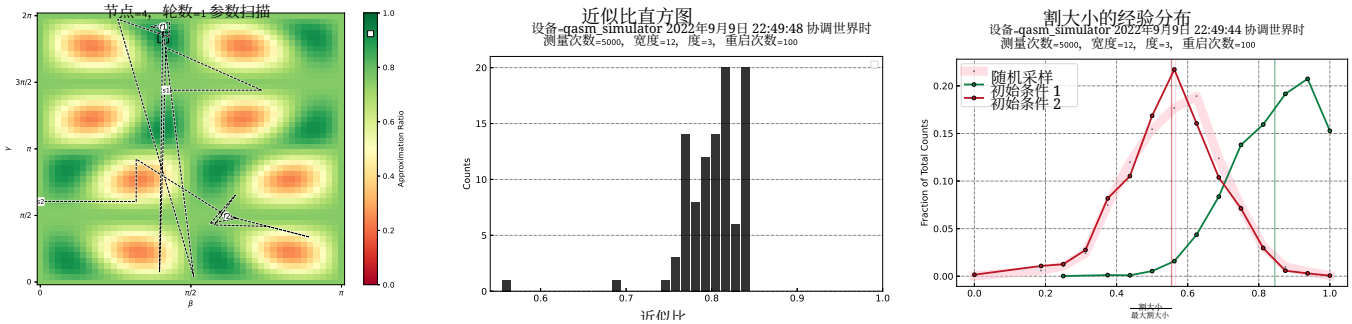
1. 初始角度和重启次数

尽管多个超参数（例如轮数、测量次数、优化器迭代次数等）都会影响迭代式量子近似优化算法执行获得高质量输出的能力，但最关键且最具挑战性的选择或许是角度初始值的设定。

经典优化程序面临着多项挑战。已有研究表明，为 QAOA 找到最优角度是一个 NP-困难问题 [99]。此外，目标函数的景观还会遭遇“贫瘠高原”现象，即目标函数的梯度接近于零，从而阻碍角度的训练 [100]。目标函数的选择 [101],以及量子硬件中的噪声 [102], 或试探态中的大规模纠缠 [103]也可能加剧贫瘠高原现象。

这些挑战带来的一个结果是，初始角度（即 $\boldsymbol{\beta}$和 $\boldsymbol{\gamma}$）的选择会显著影响优化器达到目标函数最优值的能力。例如，图 17a展示了优化器针对两个随机选取的初始角度所探测到的角度轨迹。经过 30 次优化器迭代后得到的割大小分布存在显著差异，如图17c所示。一种选择得到的输出几乎与随机采样比特串无法区分，而另一种选择则得到高质量的割分布。我们还在图 17b 中绘制了 100 次随机初始化所得近似比的直方图。

这些问题促使人们开展了大量研究，以解决并克服这些挑战。尽管已有许多方案被提出，但考虑到我们的目标是对终端用户可用的量子解的性能进行基准测试，我们着重实现了缓解其中一些影响的最基本方法。具体而言，我们实施多次“重新启动”，即使用



(a) (b) (c)

图 17：**初始角度会影响结果质量**。(a) 参数轨迹和近似比景观：COBYLA 优化器会根据初始参数值，以不同方式探索参数空间。背景中的近似比景观来自状态向量模拟，其中显示了两组随机选取的初始角度值（标记为$s1$ 和$s2$）所对应的轨迹。经过 30 次迭代后，参数分别标记为$f1$ 和$f2$。(b) 近似比直方图简洁地展示了初始条件所导致的差异：该图根据 QAOA 模拟，显示了 100 次重启在 30 次 COBYLA 迭代结束时得到的近似比。尽管大多数重启得到的近似比在 0.75 到 0.85 之间，但有些结果的近似比显著偏低。(c) 两种初始条件下，经过 30 次迭代后割大小的分布差异显著；初始条件 2 的分布几乎与比特串的随机采样重合，而初始条件 1 得到的近似比相对较高，为 ≈ 0.83。

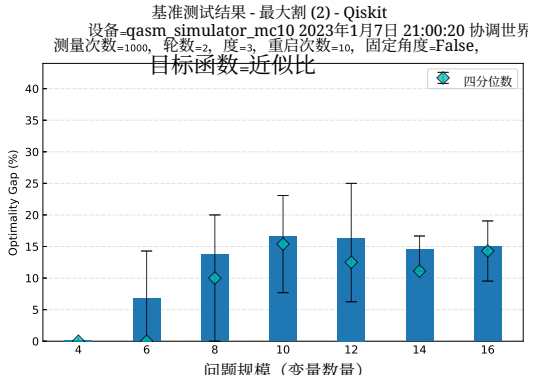


图18：**多次重启下的最优性差距**。我们展示了在每种问题规模下运行最大割基准测试10次所获得的结果。每种问题规模所示的结果，是该规模下所有10次重启中获得的“最佳”结果，定义为近似比最高的结果。

随机角度作为优化器的初始角度，多次运行量子近似优化算法，并报告效果最佳的那次重新启动所对应的输出。

为此，我们的基准测试框架允许用户通过一个名为`max_circuits`的参数指定重启次数。该参数设为 1，所有初始β 和 γ 角度也都设为 1。因此，第 V 节中的所有结果均使用这些初始角度。对于重启次数为 $>$ 1的情况，每种问题规模的图中都会显示最佳重启对应的输出。图18展示了与图5采用相同参数、进行 10 次重启时对应的输出。对于较小的问题规模，结果质量明显更好。用户还可以使用`thetas_array` 参数手动指定初始角度。

2. 固定角度猜想

尽管多次初始化或重启可以缓解优化器面临的一些困难，但多次运行优化器程序的实现成本可能相当高昂，需要耗费数倍的 Quantum 处理单元访问时间。

近期一项研究 [104] 提出了一种无需优化的QAOA实现：对每个问题只执行一次试探态，使用所谓的“固定角度”。作者表明，对于3-正则图的每个问题实例，采用这些角度所得的近似比都高于阈值。尽管这些角度并非近似比景观的全局最大值，但无需执行代价高昂的优化器循环，也能保证接近最优的性能。例如，图19 显示，除规模为4的问题外，所有问题规模在固定角度下的最优性差距，实际上与图18中的结果相同；而图18中的结果需要进行10次重启，每次包含30次优化器迭代。这使QPU访问时间减少了 ≈300 倍，同时仍能获得质量相近的结果。

因此，基准测试框架提供了一项设置，可通过将 use_fixed_angles 标志设为 True，选择将初始角度设为固定角度。还可以同时将优化器迭代次数设为 1，以避免使用优化器例程。

在图20中，我们展示了一次测试运行的结果，使用此基准测试功能来探索“参数集中”现象 [105]。对于规模从4到20个量子比特的3-正则图，使用 Max-Cut 基准测试对100组随机初始角度进行了测试，每组执行30次优化器迭代。此图显示了 γ 优化器最终得到的值及其对应的近似比。优化器得到的角度集中在四个值附近，其中大多数与 [104]中提出的值相匹配。初始角度的选择会影响算法的结果，因此，选择这些角度的策略对于实现最佳性能至关重要。

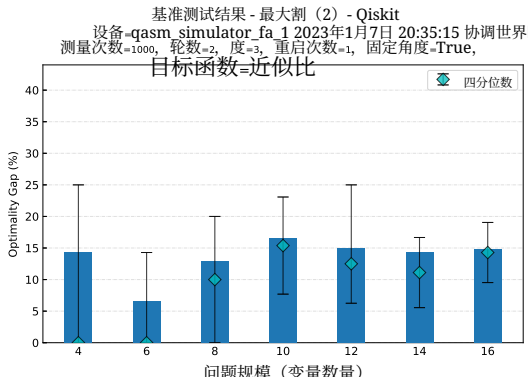


图19：**使用固定角度的免优化量子近似优化算法实现。** 为避免重启和开销高昂的优化循环，可以使用能够保证产生高质量输出的“固定角度” [104]。以下是在不实现最小化程序的情况下，使用固定角度时轮数=2 对应的最优性差距。

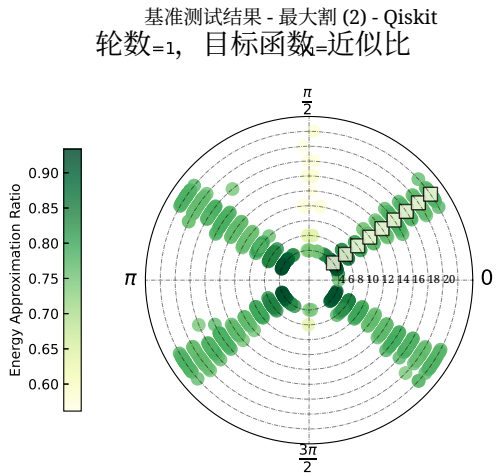


图20：**角度聚集在某些值附近。** 对于规模从4个量子比特到20个量子比特的3-正则图，我们选择100组随机初始角度，并分别运行30次（COBYLA）优化器迭代。优化器得到的最终角度聚集在四个值附近。这里展示了这些γ 值，以及相应的近似比。

3. 参数选择策略

此前，我们展示了初始角度以及轮数、测量次数和重启次数的选择如何影响 QAOA实现的性能。此外，性能还可能因问题实例而异。这些因素引出了以下问题：对于此前未见过的问题实例，我们能否预测出有望实现最佳性能的参数值？具体而言，给定资源（例如 QPU访问时间）和结果质量指标（例如近似比）的定义，在资源预算给定的情况下，应使用哪些参数值才能获得最佳性能？

基于这一目标，目前正在并行开发一个用于参数化随机优化算法的基准测试框架

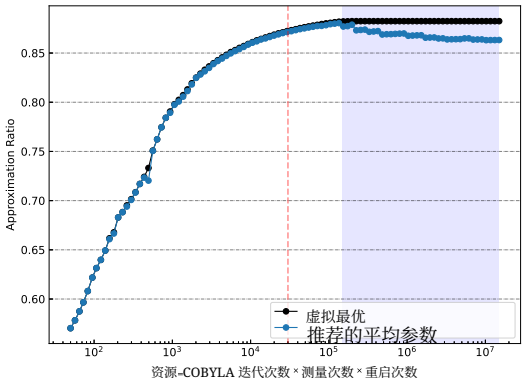


图21：**解的质量与总资源利用率：** 虚拟最优给出了任何参数策略所能达到的最佳性能上界。这里展示了虚拟最优的性能剖面，以及采用随机基准测试框架所推荐参数 [107]时获得的性能。红色虚线对应于硬件部分中始终使用的资源值（30次迭代、1000次测量、1次重启）。阴影区域突出显示了这样一个区间：在此之后，随着资源增加，近似比会下降。

，涵盖量子近似优化算法和量子退火 [106]。尽管该框架 [107] 也适用于其他算法，我们将其应用于 QAOA 模拟，并采用 QED-C基准测试框架获得的结果。该框架会针对一系列资源值生成参数建议，并绘制相应的性能

-

与数据中观测到的最佳表现相比。

该框架的输入是通过在各种问题实例上实施算法而实证获得的性能数据。数据包括结果质量（我们称之为性能指标），对应于在一系列参数值设置下进行的多次算法执行。框架还提供一个函数，用于计算每次执行所消耗的资源。

该框架将问题实例划分为测试集和训练集。随后，利用训练集数据进行统计分析，以确定应用于测试集时可能带来高性能的参数值。另一方面，对于测试集中的每个实例，框架会根据所有可用资源网格值的数据，找出对应于迄今为止最高结果质量的资源值的参数值。这些参数值汇总为一条称为“虚拟最优”的曲线。

因此，与虚拟最优对应的参数值模拟了这样一种情况：对于每个实例，都能预先知道在任意资源值下哪些参数是最佳的。虚拟最优为使用所提供的数据进行分析的任何参数选择策略所能达到的性能提供了上限。因此，推荐参数和虚拟最优参数会一同绘制在“策略图”中。虚拟最优性能则单独绘制，并与在测试集上使用推荐参数所获得的性能一同展示。

现在，我们使用这一框架分析量子近似优化算法。图 21 展示了得到的性能剖面图，而 图 22展示了该框架生成的策略图，

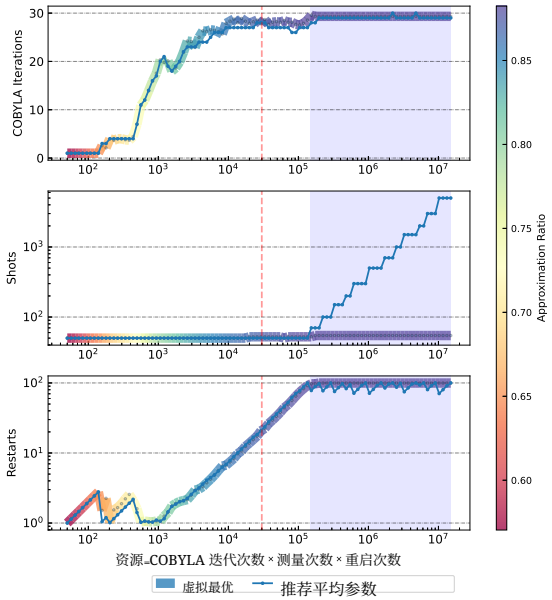


图 22：**策略图：**对于每个资源值，该框架都会为最小化器迭代次数、测量次数和重启次数推荐参数值，这些参数值可能带来最佳性能。作为比较，图中还以彩色曲线绘制了虚拟最优参数，颜色表示相应的近似比。红色虚线竖线对应于硬件部分始终使用的资源值（30 次迭代、1000 次测量、1 次重启）。阴影区域突出显示了这样一个范围：超过该范围后，近似比会随着资源增加而下降。

采用 80%-20% 的训练/测试实例划分。量子近似优化算法使用无噪声模拟，对规模为 12 的 50 个不同 3-正则图进行两轮计算。我们针对重启次数 $[1,\ldots,100]$，、经典优化器（COBYLA）的迭代次数 $[1,\ldots,30]$，以及测量次数 $[50,\ldots,5000]$ 的一系列取值进行了运行。我们将资源定义为这些参数的乘积，以此统计处理单元被访问的次数。

图 21 可用于确定推荐参数值相对于虚拟最优给出的乐观上界的相对性能

。在此示例中，我们观察到两条线几乎重合，这表明训练实例（推荐参数值）和测试实例（虚拟最优参数值）共享优质参数值。具体而言，这些结果表明，随着对处理单元的访问增加，响应质量也会提高；这一点可通过近似比衡量，但提升会达到一定限度。图中的阴影区域表示资源数量所处的一个范围，在此范围内，性能指标会随着资源增加而下降。这种现象有悖直觉，并表明，根据用于生成此分析的数据，最佳参数值对应于 COBYLA 迭代次数设为 30、重启次数设为 100 且测量次数仅为 50 的情况。导致资源使用量更大的参数值组合可能会降低近似比。这一观察结果表明，如果允许更多地访问处理单元，增加测量次数会使性能下降。

此外，这表明，当可用资源增加时，应先增加经典最小化器的迭代次数，再增加重启次数，同时始终尽量保持较少的测量次数。红色虚线表示剩余E 节中模拟所使用的等效资源量。请注意，如图 22所示，推荐的参数值，即 27 次 COBYLA 迭代、20 次重启和 50 次测量，与其他硬件演示中使用的参数并不相同，后者为 30 次 COBYLA 迭代、1 次重启和 1000 次测量。利用这一分析，并指定性能指标和资源函数，可以使用实证数据来确定参数设置值。此外，这些结果还可以帮助了解问题实例本身。在此示例中，问题实例的规模相对较小，即包含 12 个节点的图，其解空间规模为 $2^{12} = 4096$。使用量子近似优化算法求解问题时，以大量测量次数对输出分布进行广泛采样，并不能像重新初始化问题（重启）或允许更多经典优化迭代那样显著提高近似比。

这些参数设置策略分析为如何使用本文讨论的此类算法提供了实用建议。