arXiv:2301.05146v1 [quant-ph] 12 Jan 2023

基于量子退火的多样化肯梅尼排序聚合启发式算法

Sven Fiergolla[1], Kevin Goergen[1],Patrick Neises[1], 和 Petra Wolf[2]

[1]德国特里尔大学， {s4svfier, s4kegoer,s4paneis}@uni-trier.de[2]挪威卑尔根大学， mail@wolfp.net

摘要

Kemeny Rank Aggregation（KRA）问题是社会选择领域中一个经过深入研究的问题，在数据库和搜索引擎等许多不同领域都有广泛应用。直观而言，给定一组候选项上的选票，该问题要求找到一个候选项的汇总排序，使其关于这些选票的总体不满程度最小。近期，有研究提出了 KRA 的多样化版本，旨在寻找一组足够多样且质量足够好的解。解的多样性框架是人工智能领域一个新兴且蓬勃发展的研究主题。其核心思想是不只向用户提供一个解，而是提供一组不同的解，让用户能够挑选出一个足够好的解，以满足难以或无法建模的其他主观标准。

在本研究中，我们使用量子退火机求解 KRA 问题，并计算一个具有代表性的解集。量子退火是一种元搜索启发式方法，不仅在当前已有的原型机上展现出良好的运行时间表现，还以本质上不同的方式对解空间进行采样，并利用量子效应。我们说明如何使用量子退火机求解 KRA 实例，并提供了相应的实现及实验评估。由于现有量子退火机的量子比特数量仍然有限，我们还实现了两种不同的数据约简规则，可将一个实例拆分为一组更小的实例。在评估中，我们将能够对多个解进行采样的经典启发式方法（如模拟退火和局部搜索）与在物理量子退火机上执行的量子退火进行比较。我们比较了运行时间、解的质量和解的多样性，并考察了是否预先应用数据约简规则的影响。尽管我们的目标是计算一个多样化的解集，实验中也纳入了仅计算单个解的经典启发式方法。

关键词：社会选择、解的多样性、量子退火、二次无约束二元优化、启发式搜索

1 引言

随着量子计算机技术的迅速发展，评估其潜力变得越来越重要。利用量子效应解决计算问题有两种不同的方法：量子电路模型（谷歌和美国国家航空航天局采用的方法 [Arute等人， 2019], 以及IBM采用的方法 [Steffen等人， 2011]）和绝热量子模型（由D-Wave Systems主导的方法 [King等人， 2022]）。从能够以多项式级开销相互模拟的意义上说，这两种模型在计算能力上是等价的 [Farhi等人，2000，Aharonov等人， 2008] ，但它们的物理实现截然不同。量子电路模型旨在构建通用量子计算机，这需要设计新的量子算法来利用量子效应。由于量子算法相当反直觉，目前已知的、相较于经典算法能带来显著优势的量子算法寥寥无几（例如，Grover算法可在非结构化数据中搜索，并实现二次加速 [Grover， 1996], 以及

Shor算法可在多项式时间内完成因数分解 [Shor, 1994]）。相比之下，D-Wave已将绝热量子模型实现为一种称为量子退火（QA）的求解方法，用于解决NP难优化问题。D-Wave的量子退火器专为求解一种原生问题——伊辛模型[1] 而设计，该模型运行于其硬件芯片上。伊辛模型与二次无约束二元优化（QUBO）问题密切相关；从计算机科学家的角度来看，后者更易于处理，因此本文将着重讨论QUBO问题。QUBO问题为我们提供了一种便捷的接口，使我们能够将问题表述为经典优化问题，同时仍可利用量子效应加快计算。我们将在后文详细讨论QUBO问题。

直观地说，在绝热量子模型中，系统会从一个易于制备的初始低能态转变为另一个低能态，使最终状态对应于目标函数的最小化。如果这一转变足够缓慢，并且在极低温下进行（≤20mK），系统在转变过程中就会保持在最低能态，我们便可以从最终状态测得一个最优解。绝热量子模型最初由 [Farhi等人 2001] 提出，很快便应用于神经网络领域 [Kinjo等人 2003]。

量子退火是一种元搜索启发式方法，可通过绝热量子模型的原生指令集实现。目前最成熟的量子退火机实现是 D-Wave Advantage系统，拥有 5,000+个量子比特 [Willsch等人 2022]。D-Wave 的量子计算机还支持量子机器学习应用 [Hu等人, 2019年，Adachi 和 Henderson, 2015]。该公司迅速研发用于量子退火器的新型硬件芯片，但这也带来了一些与 API稳定性相关的困难。关于技术实现的更多详情，请参阅 [Bunyk等人，2014年，Harris等人，2010年，Johnson等人， 2011]；关于绝热量子模型的一般介绍，请参阅 [McGeoch， 2014]。

绝热量子模型有所不同它与热退火的经典模拟不同 [Kirkpatrick等人, 1983] 其区别在于，计算从所有能量最低状态的叠加态开始，从而利用量子并行性；此外，它还能在解空间中穿越高能区域，而经典方法则需要越过这些高能区域。模拟结果表明，绝热量子模型因此有望优于经典模拟退火[Morita和Nishimori，2008年，Ohzeki和Nishimori，2011年，Crosson 和 Harrow，2016]。此外，实验结果表明，量子退火器能够找到解空间中的局部谷地，而模拟退火器无法找到这些谷地 [Koshka 和 Novotny, 2020]。这表明，无法通过经典模拟退火高效求解的实例，或许可以在量子退火机上高效求解。

由于量子退火过程具有随机性，必须多次执行并测量，才能获得较好的解。但这看似是该方法的弱点，实际上却带来了一项优势：在系统冷却过程中，可以将其视为处于所有局部极小值的叠加态中。只有在测量后，系统才会坍缩到其中一个极小值。通过重复这一过程，我们可以对解空间中的不同极小值进行采样，从而避免像梯度下降方法那样陷入某个局部极小值。自动采样得到的多个解还使我们能够对解空间进行表征；这一目标在经典方法中通过解的多样性这一概念建模，而它是人工智能领域近期兴起的趋势 [Petit 和 Trapp, 2019, Baste 等人, 2019b, Baste 等人, 2020年, Ingmar 等人, 2020年, Fomin 等人，2020]。

传统上，人们通常关注如何为某个计算问题找到一个最优解；但在实践中，如果存在难以建模的侧约束，这可能还不够。这些主观标准的例子包括审美、经济、政治或环境标准。由于这类标准具有主观且模糊的性质，因此可能无法在问题表述层面将其建模。因此，用户或许更愿意选择一个稍逊于最优、但更符合自身主观标准的解。为了让用户能够选择最符合其需求的解

[1]伊辛模型问题是 NP 完全的 [Istrail, 2000, Cipra, 2000] 而且，即使将问题限制在可由 D-Wave 量子退火机实现的实例类别中，它仍然是 NP 完全的 [Bunyk等人, 2014]。

，最好提供一个规模适当且包含足够多样化解的解集。不过，将解集的多样性形式化并非易事，参见 [Baste 等人，2020 年，Baste 等人，2019a] 中的讨论。直观而言，多样性可用于衡量一个解集对解空间的代表性。量子退火机在对解空间进行采样时，自然会为我们提供一组解。随后，可以通过经典后处理，从这些解中选出一个适用于当前标准的子集。

在挑选多个优秀委员会时，考虑多样化的备选方案十分重要，这样可以确保每位委员会成员的工作负担不过重，正如 [Bredereck 等人所述，2020]。选举委员会，或寻找一个尽可能与一组选票相符的排名，是社会选择领域的关键问题。该领域研究最多的问题之一是 Kemeny Rank Aggregation (KRA) 问题。给定一组候选项及其对应的一组投票，任务是找出一个候选项排名，使选民的满意度最大化。这个问题在许多不同领域都很重要，例如高维数据库中的相似性搜索和分类 [Fagin 等人，2003], 生物信息学中的遗传图谱构建 [Jackson 等人，2008], 搜索引擎中的搜索结果聚合和垃圾邮件检测 [Dwork 等人，2001], 等等。KRA 之所以广受欢迎，是因为 Kemeny 规则具有独特的性质：中立、一致，并满足孔多塞性质 [Young 和 Levenglick，1978]。

KRA 问题是 NP 困难的 [Bartholdi 等人，1989] 即使限制为只有四张选票，仍然是 NP 困难的 [Dwork 等人，2001，Biedl 等人，2009]。这一复杂性促使人们研究用于 KRA的启发式算法和近似算法，参见 [Dwork 等人，2001，Ailon 等人，2008，van Zuylen 和 Williamson，2009], 以及参数化算法 [Betzler 等人，2009，Simjour，2009，Arrighi 等人，2020，Betzler 等人，2014]。一系列关于 KRA 问题的启发式算法和近似方法的实验研究，也凸显了其在人工智能领域的实际意义 [Ali 和 Meila，2012，Davenport 和 Kalagnanam，2004] 以及关于 KRA [精确解下界的实验研究 KRA [Conitzer 等人，2006，Schalekamp 和 van Zuylen，2009]。针对 KemenyRank Aggregation 多样化版本的参数化算法近期发表于 IJCAI [Arrighi 等人，2021]。

在这项工作中，我们利用绝热量子模型为 Diverse Ke-meny Rank Aggregation 问题设计启发式算法，从而获得一组解空间样本。为此，我们将 KRA 建模为二次无约束二元优化问题。随后，我们将基于量子的启发式算法与经典启发式方法进行比较：后者既包括能够采样不同解的方法，如模拟退火和局部搜索，也包括只能确定性地计算单个解的经典启发式方法。

由于量子计算机目前可用的量子比特数量仍然相当有限（目前为 5,000+个量子比特），大型实例对任何量子计算机来说仍是一项挑战。为应对这一问题，我们在预处理步骤中实现了若干数据约简规则。这些规则使我们能够将一个实例分解为若干更小的实例，并分别求解。随后，将这些较小实例的解汇总，得到最终解。

本文结构如下。首先，我们给出 QUBO 和 KRA 问题的形式化定义。然后，我们介绍所用的数据约简规则，以及针对KRA得到的 QUBO构造。接下来进入论文的实验部分，并介绍所考察的求解器。在实验中，我们比较所有求解器的运行时间和所得解的质量，并比较支持计算多个解的求解器所得解的多样性。为此，我们比较三种不同的预处理模式：不进行预处理、穷尽应用 $\leq_{3/4}$多数规则，以及应用扩展孔多塞规则。本文最后介绍我们的观察结果。我们的实现和所用数据集可作为补充材料获取。

2 预备知识

设 Q 是一个 $n \times n$ 矩阵。我们用 Q_{ij} 表示 Q 中第 i 行、第 j 列的矩阵元素。设 n 是一个非负整数。我们记作 $[n] = \{1, 2, \ldots n\}$。

因此，$[0] = \emptyset$。

2.1 二进制二次无约束优化

现在，我们给出 Quadratic Unconstrained Binary Optimization（QUBO）问题的形式化定义。设 Q 是一个权重 $n \times n$ 矩阵，且 x_1、x_2、...、x_n 是二进制变量。那么，QUBO问题要求为变量 x_i 赋值，其中 $i \in [n]$ 使得以下表达式最小化

$$\sum_{i,j} Q_{ij} x_i x_j.$$

2.2 Kemeny 排名聚合

设 C 为一个有限候选人集合。一个投票（或排名）π 是对 C 的一个全序。如果对于两个候选人 a、$b \in C$，候选人 a 在投票 π中的排名优于 b，则记作 $a <_\pi b$。因此，按照 $<_\pi$ 排名最靠前的候选人 a 是投票 π 的获胜者，可视为排名第 1 位。对于两个投票 π_i 和 π_j，在这两个投票中排序不同的候选人对数称为π_i和π_j的 Kendall Tau 距离，记作 KT-distance(π_i,π_j)。形式化地说：

$$\text{KT-distance}(\pi_i, \pi_j) = |\{(c_1, c_2) \in C \times C \mid c_1 <_{\pi_i} c_2 \wedge c_1 >_{\pi_j} c_2\}|$$

给定对 C 的一个排名 π 以及一个由 C上的投票组成的集合 Π，排名 π 相对于 Π的肯梅尼得分，是 π 与每个$\pi_i \in \Pi$之间 Kendall Tau 距离的总和。KRA 的目标是计算 C的一个排名$\pi_{\min}$，使其肯梅尼得分尽可能小。这样的排名称为 Kemeny 排名。

3 框架

本节旨在从理论上解释我们所用的概念。首先，我们介绍针对 KRA 的数据约简规则，这些规则取自 [Betzler 等人, 2014], 并由我们作为预处理步骤实现。[2] 然后，我们介绍针对 KRA 的 QUBO 构造。

[2]在 [Betzler 等人, 2014] 中，选票以反向方式解读，也就是说，若候选人 a 在一张选票中的排名优于 b，则 $a > b$。因此，在原始研究中，根据 $\geq_s$-多数规则定义了以下规则。

3.1 数据约简规则

由于量子退火机上可用的量子比特数量有限，我们采用一些已知的数据约简规则作为预处理步骤，将一个实例划分为一组更小、可独立求解的实例。

3.1.1 $\leq_{3/4}$-多数规则

设 C 为候选人集合，Π 为选票集合。此外，设 a、$b \in C$ 为候选人。若对于 C 上的所有选票 π_i，均有 $a <_{\pi_i} b$，则在 Kemeny 排名中，$a < b$。直观而言，如果 a在每一张选票中的排名都低于 b，那么在 Kemeny 排名中，a 显然必须排在 b 之后。我们可以将这一思想推广如下。设 $s \in [0, 1]$。如果至少在 $s \cdot |\Pi|$张选票中，$a <_{\pi_i} b$ π_i，那么我们称有序对（a，b）是关于 $\leq_s$-多数规则的干净对，并将其记为 $a \leq_s b$。如果候选人 a 与每个其他候选人 $b \in C \setminus \{a\}$都形成关于 $\leq_s$-多数规则的干净对，则称 a 是关于 $\leq_s$-多数规则的干净候选项。Betzler 等人，2014，引理 1]证明，对于关于 $\leq_{3/4}$-多数规则（$s = 3/4$）的干净候选项，任何 Kemeny 排名都会遵循涉及 a 的干净对。基于这一观察，针对参数 d_a 得到了一个线性部分核，其中$d_a = \sum_{\pi_i,\ \pi_j \in \Pi}$ KT 距离(π_i,π_j)/(|C| · (|C| − 1))为平均 KT 距离。该核基于以下 $\leq_{3/4}$-多数规则：

对于一个 KRA 实例 (C, Π)，令 $N := \{n_1, \ldots, n_t\}$ 为相对于 $\leq_{3/4}$-多数规则的干净候选项集合，使得对于 $i \in [t-1]$，$n_i \leq_{3/4} n_{i+1}$。然后，定义

$$D_0 := \{c \in C \setminus N \mid c \leq_{3/4} n_1\},$$
$$D_i := \{c \in C \setminus N \mid n_i \leq_{3/4} c \wedge c \leq_{3/4} n_{i+1}\},$$
$$D_t := \{c \in C \setminus N \mid n_t \leq_{3/4} c\},$$

对于 $i \in [t-1]$。将原实例替换为由 D_i 对于 $0 \leq i \leq t$ 所诱导的 $t+1$ 子实例。

正如 Betzler 等人所示，这一数据约简规则是安全的，并且能够保留所有 Kemeny 排名；对于计算多种不同的 Kemeny 排名而言，这是一项理想的性质。该规则的另一个优点是，各个子实例可以在量子退火机上分别求解，所需的量子比特比原始实例少得多。

3.1.2 扩展孔多塞规则

在扩展孔多塞规则中，我们不再比较某个候选人与所有其他候选项之间的相对位置，而是考察一组候选项 C' 与其余候选项 $C \setminus C'$之间的相对位置。由此，该规则根据多数图的强连通分量对实例进行划分。

定义 1（多数图）。设 (C, Π) 是 KRA 的一个实例。(C, Π) 的弱（严格）多数图是一个有向图，其顶点集为 C，且当且仅当 $u < v$ 出现在至少（超过）半数选票中时，存在一条从 u 指向 v 的弧。

由于我们关注的是计算一个多样化的解集，因此我们采用扩展孔多塞规则的一个版本，该版本基于弱多数图，并保留所有 Kemeny 排名。如果上下文明确，我们就称此规则为孔多塞规则。其工作方式如下 [Betzler 等人， 2014]。设 (C, Π) 是一个选举，并令 $C_1, C_2, \ldots, C_t \subseteq C$ 为 (C, Π) 的弱多数图中强连通分量的顶点集，按拓扑序排列。将原始实例替换为由 C_i 诱导的子实例，其中包含 $|C_i| \geq 2, i \in [t]$。

尽管该规则保留了所有 Kemeny 排名，但如 [Betzler 等人 2014]的例 2 所示，应用该规则后得到的实例仍可能通过 $\leq_{3/4}$多数规则进一步缩减。如果我们改为考虑严格多数图的强连通分量，那么相对于参数 d_a得到的核便无法再通过$\leq_{3/4}$多数规则进一步压缩，这意味着核大小更小；但我们可能会丢失一些 Kemeny 排名，因为在两个方向上恰好各获得 $1/2$ 选票的候选项对，在严格多数图中并不相连，因此必须在拓扑排序过程中确定它们之间的顺序。由于我们的目标是将多个解的计算留给量子退火器，因此接下来我们采用基于弱多数图的孔多塞规则。另一种做法是只在子实例上计算多个解，并使用经典后处理来计算严格多数图的所有拓扑排序，再据此拼接各子实例的解。不难看出，一旦我们得到强连通分量，就可以将问题视为一个所有代价均设为 1 的 Comple-tion of an Ordering 问题，并使用 [Arrighi 等人 2021]提出的多样化版本的参数化算法，从而计算出一组多样化的拓扑排序。

3.2 Kemeny 排名聚合的 QUBO构造

现在我们讨论 KRA 的 QUBO构造：给定一组 n 候选项 C 的排名，我们使用 n^2 二进制变量，记作 $c_{i,\,j}$ 其中 $1 \leq i,\ j \leq n$。直观地说，如果变量 $c_{i,\,j} = 1$，这意味着候选人 i 的位置为 j。二进制变量只能取 0 或 1，可将其解释为布尔值。我们将变量 $c_{i,\,j}$ 视为一个二维网格。为获得有效的 Kemeny 排名，我们必须确保每个候选人在最终排名中恰好占据一个位置（$\forall i : \sum_{j=1}^{n} c_{i,\,j} = 1$，我们通过所谓的行惩罚项来实现这一点），并确保每个位置恰好由一个候选人占据（$\forall j : \sum_{i=1}^{n} c_{i,\,j} = 1$，这一点则通过列惩罚来实现）。惩罚项是一个很大的正权重，它会乘以目标函数中的某一项；当变量赋值违反我们希望禁止的条件时，该项会取正值。将侧约束转化为惩罚项后，我们便可得到一个无约束 QUBO 实例。因此，我们需要确保惩罚项的代价高到任何最优解都不会选择承担它。我们将 $P = |C|^2 \cdot |\Pi|$ 选作惩罚项。稍后便会说明这一选择的原因。

3.2.1 行惩罚项

行惩罚项确保每位候选人恰好被分配到一个位置。这通过最终目标函数中的以下项实现。对于每一行 $i \in [n]$，我们定义

$$\mathrm{row}_i = P\left(1 - (\sum_{j=1}^{n} c_{i,j})\right)^2$$

该方程的最小值为 0，当且仅当总和中的 $c_{i,j}$ 恰有一个为真时取得。对于固定的 i，如果变量 $c_{i,j}$ 中没有任何一个（其中 $j \in [n]$ 为真），那么第 row_i 项的值

为 P。另一方面，如果有 $k > 1$ 许多变量$c_{i,j}$ 为真，那么第 row_i 项的值为 $P(1-k)^2 = P(-(k-1))^2 = P(k-1)^2 \geq P$。

3.2.2 列惩罚

类似地，列惩罚确保每个位置 j 都恰好分配给一个候选人。对于每个 $j \in [n]$，我们设定：

$$\text{col}_j = P\left(1 - (\sum_{i=1}^{n} c_{i,j})\right)^2$$

3.2.3 排序惩罚

最后，我们需要表示实际的投票。这通过对每对变量 $c_{i,\ j}$、$c_{i',\ j'}$ 施加惩罚来实现，其中 $i \neq i'$ 且 $j \neq j'$。我们的目标是为二次项 $c_{i,\ j}c_{i',\ j'}$ 赋予一个权重，以反映当候选人 i 位于位置 j 且候选人 i' 位于位置 j'时，会违背多少张选票。 [Arrighi 等人，2020，第 5 节1]证明，可以通过对每对候选项的代价求和来计算某个排名的肯梅尼得分；该代价即为该候选项对的相对位置被违背的选票数。这样，对所有候选项对的惩罚求和，就能得到与某个变量赋值相关联的排名的肯梅尼得分。

设 $w_{i,\ j}$ 为候选人 c_j 排名优于候选人 c_i的选票总数。更正式地说[3]，$w_{i,\ j} = \sum_{\pi \in \Pi}[c_j <_\pi c_i]$。如果我们在汇总排名中违反候选人的这一相对位置关系，那么$w_{i,\ j}$ 就是将候选人 c_i 排在候选人c_j之前所需付出的代价。因此，我们得到以下表示将候选人 c_i 排在候选人 c_j之前所需代价的项。对于每个 i，$j \in [n]$，我们定义：

$$\text{rank}_{i,j} = w_{i,j} \cdot \sum_{k=1}^{n-1} \sum_{l>k}^{n} c_{i,k} c_{j,l}$$

3.2.4 最终表述

我们现在可以构造依赖于所有变量 $c_{i,\ j}$ 的最终无约束目标函数

[3]括号记法的含义如下：如果 p 是一个逻辑命题，那么当 p 为真时，$[p]$的值为 1；否则，$[p]$ 的值为 0。

对于 $i, j \in [n]$:

$$f(c_{1,1} \ldots c_{n,n}) = \sum_{i=1}^{n} \left(\text{row}_i + \text{col}_i + \sum_{j=1}^{n} \text{rank}_{i,j} \right)$$

回到用于行惩罚和列惩罚的惩罚项 P：由于我们将其设为$P = |C|^2 \cdot |\Pi|$，现在可以看出，即使每一张选票都与所有可能的相对位置安排相矛盾（这是一个无法达到的上界），违反每位候选人恰好占据一个位置、且每个位置恰好由一位候选人占据的规则，所产生的惩罚仍然会更高，因此在最优解中绝不会发生。

4 实现

该项目使用 Python 编写，并采用 D-Wave 的DIMOD API。

4.1 数据集

我们在评估中使用了两种不同的数据集。第一种是 Betzler 等人 [的数据约简规则实验评估中使用的一级方程式数据集 2014]。该数据集中的候选项数量从 6 到 28 不等。他们还考察了其他真实世界的数据集，例如滑雪排名和网页搜索结果。但这些数据集分别包含多达 69 个和 200 个候选项的实例，对于所考虑的量子退火器而言，规模太大，无法映射到其物理硬件上。因此，我们不将它们纳入评估。

相比之下，我们纳入了第二个由随机生成的实例组成的数据集，候选项数量从 3 个到 100 个不等，旨在逐步增加实例规模，从而找出不同求解器的局限性。我们首先从一个有序列表开始，然后使用标准的 Python 方法将其打乱，以创建该实例。正如预期，随机生成的实例并未为数据约简规则提供太多结构，因此用于比较不同参数下不同求解器的运行时间和解的质量。相比之下，一级方程式数据用于比较解的多样性以及数据约简

规则的影响。有关数据编码的详细信息、我们实现的文档，以及如何运行实验的说明，可在补充材料中找到；补充材料还包含代码和数据集。

4.2 预处理

我们实现了上述数据约简规则，将实例划分为子实例。如果目标是获得多样化的解集，我们会以笛卡尔积的方式拼接各子实例中的最优解。可用模式包括：不进行数据约简、 $\leq_{3/4}$-多数规则和孔多塞规则。

4.3 使用不同求解器求解

完成问题表述后，可以使用多种不同的求解器来求解二次模型，各有优缺点。一般来说，D-Wave 提供的求解器可分为三类：数学求解、模拟退火和真正的量子退火。

4.3.1 最速下降

我们考察的 D-Wave 求解器中，第一个是 SteepDes。它采用最速下降方法，这是梯度下降的离散类比，但它通过局部最小化来计算最佳移动，而非计算梯度。对于一个包含 n 个变量的二次无约束二元优化输入，底层 C++ 实现的运行时间复杂度在初始化阶段为 $\mathcal{O}(n^2)$，每次下降步骤为 $\mathcal{O}(n)$。在我们进行实验时， SteepDes 不支持计算多个解。

4.3.2 模拟退火

模拟退火是一种用于寻找给定目标函数局部最优解的概率技术。D-Wave 提供的 SimAnl 求解器实现了一种模拟退火算法，用于近似玻尔兹曼采样。他们的方法遵循 [Kirkpatrick 等人的工作， 1983]。

4.3.3 量子退火

D-Wave 提供两种不同类型的量子退火器：量子处理单元（QPU）和混合模型。在混合模型中，Ocean SDK 会负责将二次无约束二元优化实例映射为硬件图上的量子比特偏置和耦合值，但其缺点是无法自由访问参数，例如样本数量（num_read）或输出解的数量。相比之下，QPU 型号允许直接访问量子机器，但用户需要自行确定参数的最优值，并在硬件上寻找嵌入方式。后者并非易事；如果没有自行实现嵌入算法，只能使用通用启发式算法，而该算法所能嵌入的实例规模明显小于混合模型所能处理的规模。目前发展最成熟的 QPU 系列是 Advantage 型号，拥有超过 5,000 个量子比特。与其前代产品 2,000Q 量子处理单元相比，它采用了新的拓扑结构 Pegasus 图，用以描述芯片上量子比特之间的连接关系。在我们的实验中，我们使用 Advantage 型号。我们将基于混合模型的求解器称为 Hybrid，将 Advantage QPU 称为 QAnl。

局部搜索 我们实现了 [Schalekamp 和 van Zuylen, 2009] 中描述的局部搜索算法。 [Ali 和 Meila, 2012] 的实验中也考虑了局部搜索，并发现它计算出的解明显优于 Borda 等速度更快的算法。该算法以随机顺序遍历候选项。考察位置 i 时，会将位置 i 上的候选人移至能使当前肯梅尼得分提升最多的位置。如果没有元素移动且所有位置都已考察，该过程便会停止。由于该算法采用随机顺序，我们会多次执行局部搜索算法，以计算多个解。

博尔达 我们实现了 [Ali 和 Meila, 2012] 提出的博尔达算法。按照我们的记号，该算法为每个候选人 a 计算值 $q_{c_i} = \sum_{c_j \in C} w_{i,j}$ ，然后返回按 q_{c_i} 降序排列候选项所得的排列。由于该算法以确定性方式计算单个解，因此无法计算多样化的解

集。在 [Ali 和 Meila，2012] 中，研究者发现，在没有共识或共识很强的情况下，BORDA 在所考察的求解器中表现最佳。在 [Schalekamp 和 van Zuylen，2009] BORDA 的比较中，表现也非常出色。

快速排序 Ali 和 Meila 在 [2012] 中考虑了多种基于排序的算法，其中基于快速排序的算法在我们的实验中表现最佳。该算法根据谓词 $w_{ij} < w_{ji}$ 对候选人进行排序：如果该谓词为真，则将候选人 c_i 排在候选人 c_j 左侧。此外，QUICKSORT 只能计算一个解。

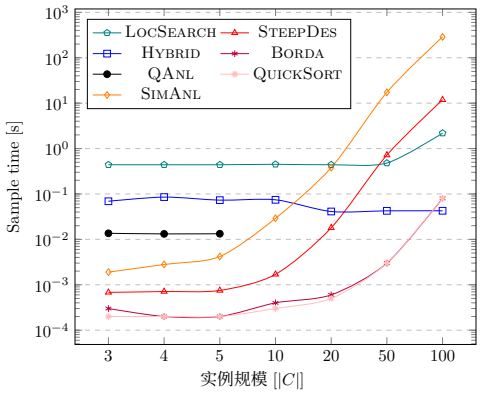

图 1：随着候选项数量增加，样本数量和量子处理单元时间分别呈现的变化；数据为 10 个随机实例的平均值。

4.4 参数

我们重点分析两个参数的影响。参数 num_reads 决定求解器执行的样本读取次数，其标准值为 1。增大该参数可获得更好的解，但会导致运行时间增加。我们在实验 2 中考察了它的影响。在我们进行实验时，除 HYBRID 外，D-Wave 的所有求解器都支持 num_reads 参数。参数 max_answers 决定求解器输出的解的数量。虽然在本项目初期，HYBRID 求解器也支持 max_answers，但在我们进行实验时，只有 QANL 和 SIMANL 支持该参数。

5 实验结果

所有本地求解器基准测试均在运行 Linux PopOS（Ubuntu 22.4 LTS）的系统上进行，该系统使用 5.17.5 内核，配备 AMD Ryzen 5 2600X 六核处理器（12 个线程），基准时钟频率为 3.6 GHz、加速时钟频率为 4.2 GHz，并配有 32GB、3200MHz 内存。

5.1 实验 1：运行时间表现（图 1）

我们测量了每个求解器在一组规模递增的随机生成的实例上的运行时间，每种规模（候选项数量）包含 10 个实例。每个实例包含与候选项数量相同的选票。所有求解器均使用默认参数初始化，测量的仅为纯采样时间。对于 HYBRID 求解器，测量的是 QPU 时间。

我们观察到，QANL、HYBRID 和 LOCSEARCH 的采样时间基本保持恒定；而 SIMANL、STEEPDES、BORDA 和 QUICKSORT 的运行时间起初优于其他求解器，但随着实例规模增大，随后显著增加。

5.2 实验 2：解的质量（图 2）

由于 QANL、SIMANL和 STEEPDES使用的默认采样数为 1，因此较大的实例无法得到最优解。事实上，随着实例规模增大，解的质量会大幅下降。为提高解的质量，可以通过参数 num_read 设置采样次数。为了衡量解的质量，我们使用整数线性规划计算最优解的成本，并将其与不同求解器找到的最佳解的 Kemeny 得分进行比较。我们测量了所有规模为 20 的随机实例在 num_read 逐渐增大时的平均解质量和平均运行时间。作为对比，我们也展示了其他求解器的结果，尽管它们不支持该参数。由于 QANL solver 无法求解规模为 20 的实例，且在较小规模的实例上也只表现出平凡的行为，因此本实验未纳入该求解器。

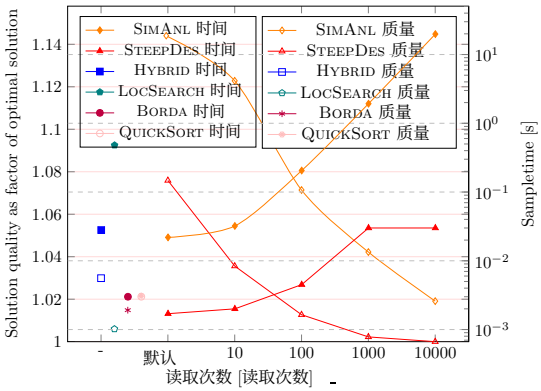

图2：解的质量占最优解的比例，对 10 个包含 20 个候选项的随机实例取平均。

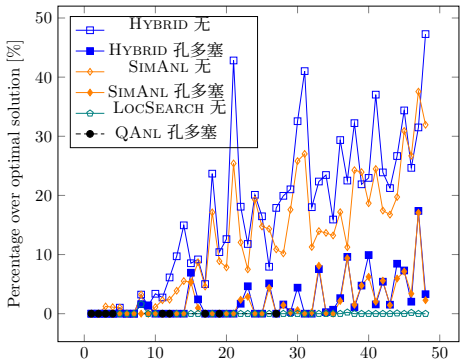

图 3：图中显示了找到的最佳解的肯梅尼得分高于最优解的百分比。

SimAnl和 SteepDes 在解的质量与采样时间之间呈现明显的权衡。使用默认参数时，Hybrid 在解的质量方面优于SimAnl 和 SteepDes 。随着 num_read 增大， SimAnl和 SteepDes 在解的质量方面确实优于 Hybrid 。对于 SimAnl ，只有采样时间大幅增加时才会出现这种情况。当 num_read = 100 时，SteepDes在解的质量和采样时间方面均优于 Hybrid 。增大 num_read 后， SteepDes 便能找到最优解，其采样时间仅略高于 Hybrid。在允许计算多个解的求解器中， LocSearch 在标准参数下的解质量最佳，但运行时间也最长。

5.3 实验3：多样性与预处理（图3–5）

接下来，我们测量能够一次性计算出一个解集的求解器的解的质量、解的多样性和运行时间。具体而言，我们比较基于退火的求解器 SimAnl、Hybrid、 QAnl和 LocSearch 在有预处理和无预处理两种情况下的表现。请注意， Borda、Quick-Sort和 SteepDes 不支持在单次求解器运行中输出多个解，因此不纳入本次实验。由于此前考虑的随机化数据不包含足够的数据约简规则结构，我们使用一级方程式数据。我们发现，在一级方程式数据上， $\leq_{3/4}$多数规则从未

适用。因此，我们将其排除在此处所示的评估之外。对于 SimAnl 和 D-Wave，所有实验均设置 num_read =10, 000，并将至多 10 个最佳解纳入解集；我们以两两最小 KT 距离和平均 KT 距离衡量该解集的多样性。对于 LocSearch，我们运行求解器 50 次，并将找到的 10 个最佳解纳入考量。图中所示的运行时间为全部 50 次运行的总运行时间。并非所有实例都能在量子退火器上求解。若缺少某个数据点，则表示该实例无法嵌入硬件，或者返回了无效解（违反行约束或列约束）。实例按候选项数量升序排列。与 Betzler 等人 [一样，我们删除了包含重复候选项以及未出现在每张选票中的候选项的无效投票。与 SimAnl不同，LocSearch 并非总能找到 10 个不同的解。只要 D-Wave 能够求解该实例，它几乎总是会输出 10 个不同的解。

观察结果 我们发现，$\leq_{3/4}$- 多数规则不会缩减任何实例。相比之下，孔多塞规则会将 $1/3$ 的实例拆分为两个子实例，并将 $1/6$ 的实例拆分为 3 个子实例。此外，它几乎缩减了每个实例的规模。虽然 Hybrid 求解器在两种预处理模式下都能求解几乎所有实例，但 QAnl只有在应用孔多塞规则后才能求解部分实例。应用孔多塞

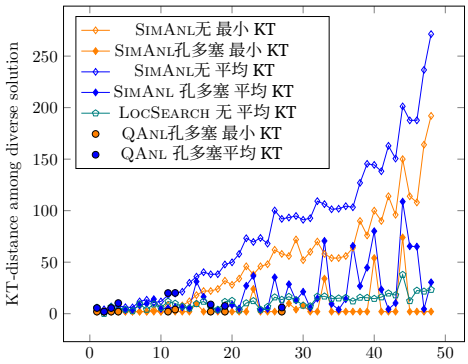

图 4：使用和不使用孔多塞规则时，找到的前 10 个解之间的最小及平均两两 KT 距离。

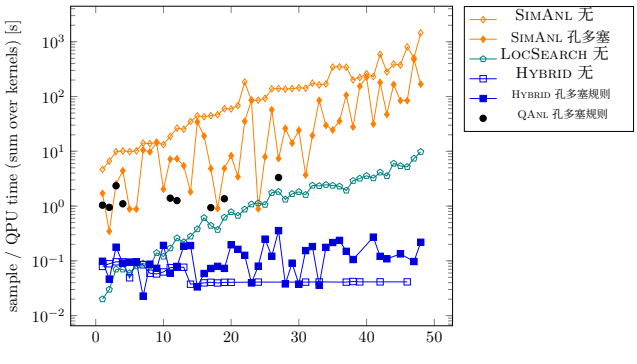

图5：采样时间，以及采用和未采用孔多塞规则时的 QPU 时间。

规则显著提升了所有求解器的解的质量。应用该规则后，所有求解器都能最优地求解大多数实例。令人意外的是，就解的质量而言，在不进行预处理的情况下，模拟退火器和非模拟退火器的表现相近。在所考察的 10 个最佳解中，各解的肯梅尼得分差异很小。这意味着我们找到了质量相近但彼此不同的解。就多样性而言，应用孔多塞规则显著降低了 SimAnl 的最小和平均 KT 距离。相比之下，对于QAnl 求解器，孔多塞规则似乎不会影响平均 KT 距离。可以观察到，孔多塞规则使更多实例能够通过 QAnl 求解，且解的质量更好、多样性更高；不过，这些观察结果应谨慎看待，因为我们只能在普通量子退火机上求解少数实例。就采样时间而言， Hybrid 的 QPU 时间显著短于SimAnl的采样时间。孔多塞规则缩短了 SimAnl 的总采样时间，却增加了 Hybrid 的总 QPU 时间。我们认为，这是由于硬件耦合器的充电时间增加所致，而充电时间占 QPU 时间的最大部分。由于需要求解多个子实例，硬件必须多次充电。与 Hybrid 相比， QAnl的 QPU 时间更长，我们认为这是因为QAnl 的读取次数较多，而 Hybrid的标准值无法获取。将 num_read= 100 设置为，QAnl 的 QPU 时间与 Hybrid 相近，但解的质量略有下降。

即使没有进行预处理， LocSearch 几乎总能找到最优解，但仅在 60% 的实例中找到 10 个不同的解，而 SimAnl在 90% 的实例中做到了这一点。 SimAnl 和 LocSearch找到的第 10 优解的肯梅尼得分仅略有差异。就运行时间而言， LocSearch 比 SimAnl 快一个数量级，而 Hybrid 在几乎所有实例中运行时间恒定，是速度最快的。如果将孔多塞规则作为预处理步骤应用于 Hybrid，那么在候选项超过 15 个的实例中，有 50% 找到了与 LocSearch质量相近的解。

6 结论

快速对解空间进行大量采样的特性，可能成为量子退火的主要优势之一。我们的实验表明，利用量子退火，我们能够计算出一组多样性相当高且运行时间表现良好的解。不过，只有等硬件进一步发展，能够求解更大规模的实例后，才能充分检验量子退火的潜力。在此之前，将量子退火与数据约简规则结合起来，作为预处理步骤，可能带来颇具前景的近期应用。

参考文献

[Adachi 和 Henderson， 2015] Adachi, S. H. 和 Henderson, M. P.（2015年）。将量子退火应用于深度神经网络的训练。arXiv 预印本 arXiv:1510.06356。

[Aharonov 等人， 2008] Aharonov, D., van Da m, W., Kempe, J., Landau, Z., Lloyd, S. 和 Regev, O.（2008年）。绝热量子计算等价于标准量子计算。《工业与应用数学学会评论》，50(4):755–787。

[Ailon 等人， 2008] Ailon, N., Charikar, M. 和 Newman, A.（2008年）。汇总不一致的信息：排名与聚类。《ACM 期刊》，55(5):23:1–23:27。

[Ali 和 Meila， 2012] Ali, A. 和 Meila, M.（2012）。Kemeny 排名实验：何时采用何种方法？《数学与社会科学》，64(1):28–40。

[Arrighi 等人， 2020] Arrighi, E., Fernau, H., de Oliveira Oliveira, M., and Wolf, P.（2020年）。排序相关问题的宽度概念。载于 Saxena, N. 和 Simon, S. 编，FSTTCS，LIPIcs 第 182 卷，第 9:1–9:18 页。Schloss Dagstuhl - Leibniz-Zentrum f¨ur Informatik。[Arrighi 等人， 2021] Arrighi, E., Fernau, H., Lok- shtanov, D., de Oliveira Oliveira, M., and Wolf, P.（2021）。Kemeny 排名聚合中的多样性：一种参数化方法。载于 Zhou, Z. 编，IJCAI，第 10–16 页。ijcai.org。[Arute 等人， 2019] Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., Biswas, R., Boixo, S., Brandao, F. G., Buell, D. A., 等（2019）。使用可编程超导处理器实现量子霸权。《自然》，574(7779):505–510。[Bartholdi 等人， 1989] Bartholdi, J., Tovey, C. A., and Trick, M. A.（1989）。难以判定谁赢得选举的投票方案。Social Choice and Welfare，6(2):157–165。[Baste 等人， 2020] Baste, J., Fellows, M. R., J af- fke, L., Masar´ık, T., de Oliveira Oliveira, M., Philip, G., and Rosamond, F. A.（2020）。解的多样性：从固定参数可处理性理论视角进行探索。载于 IJCAI，第 1119–1125 页。

[Baste 等人，2019a] Baste, J., Jaffke, L., Masaˇr´ık, T., Philip, G., and Rote, G.（2019a）。面向多样化击中集集合的 FPT 算法。Algorithms，12(12):254。

[Baste 等人，2019b] Baste, J., Jaffke, L., Masaˇr´ık, T., Philip, G. 和 Rote, G.（2019b）。用于多样化击中集集合的 FPT 算法。《Algorithms》，12:254。

[Betzler 等人， 2014] Betzler, N., Bredereck, R. 和 Niedermeier, R.（2014）。精确 Ke meny 排名聚合的数据约简：理论与实证评估。Auton. Agents Multi Agent Syst., 28(5):721–748。[Betzler 等人， 2009] Betzler, N., Fellows, M. R., Guo, J., Niedermeier, R. 和 Rosamond, F. A.（2009）。Kemeny 排名的固定参数算法。Theor. Comput. Sci., 410(45):4554–4570。[Biedl 等人， 2009] Biedl, T. C., Brandenburg, F. 和 Deng, X.（2009）。关于排列中的交叉问题的复杂性。Discret. Math., 309(7):1813–1823。[Bredereck 等人， 2020] Bredereck, R., Kaczmar- czyk, A. 和 Niedermeier, R.（2020）。选举连续委员会：复杂性与算法。载于 AAAI，第 1846–1853 页。AAAI出版社。[Bunyk等人， 2014] Bunyk, P. I., Hoskinson, E. M., Johnson, M. W., Tolkacheva, E., Al- tomare, F., Berkley, A. J., Harris, R., Hilton, J. P., Lanting, T., Przybysz, A. J. 等（2014）。超导量子退火处理器设计中的架构考量。IEEE Transactions on Applied Superconductiv- ity, 24(4):1–10。[Cipra， 2000] Cipra, B. A.（2000）。伊辛模型是 NP 完全问题。SIAM News, 33(6):1–3。[Conitzer 等人， 2006] Conitzer, V., Davenport, A. J. 和 Kalagnana m, J.（2006）。计算 Kemeny 排名的改进界限。载于 AAAI，第 620–626 页。AAAI出版社。[Crosson 和 Harrow， 2016] Crosson, E. 和 Har- row, A. W.（2016）。量子模拟退火的速度可能比经典模拟退火快一个指数级。载于 FOCS，第 714–723 页。IEEE。

[Davenport 和 Kalagnanam， 2004] Davenport, A. J. 和 Kalagnanam, J.（2004）。偏好聚合中 Kemeny 规则的计算研究。载于 McGuinness, D. L. 和 Fergu- son, G.（编），AAAI，第 697–702 页。AAAI出版社 / 麻省理工学院出版社。[Dwork 等人， 2001] Dwork, C., Kumar, R., Naor 和 Sivakumar, D.（2001）。万维网的排序聚合方法。载于 Shen, V. Y., Saito, N., Lyu, M. R. 和 Zurko, M. E.（编），WWW，第 613–622 页。ACM。[Fagin 等人， 2003] Fagin, R., Kumar, R. 和 Sivakumar, D.（2003）。SIGMOD。第 301–312 页。ACM。[Farhi 等人， 2001] Farhi, E., Goldstone, J., Gut- mann, S., Lapan, J., Lundgren, A. 和 Preda, D.（2001）。一种应用于 NP 完全问题随机实例的量子绝热演化算法。《科学》, 292(5516):472–475。[Farhi 等人， 2000] Farhi, E., Goldstone, J., Gut- mann 和 Sipser, M.（2000）。通过绝热演化实现量子计算。arXiv 预印本 quant-ph/0001106。[Fomin 等人， 2020] Fomin, F. V., Golovach, P. A., Jaffke, L., Philip, G. 和 Sagunov, D.（2020）。多样化匹配对。载于 ISAAC。Schloss Dagstuhl-Leibniz-Zentrum f¨ur Informatik。[Grover， 1996] Grover, L. K.（1996）。一种用于数据库搜索的快速量子力学算法。载于 Miller, G. L.（编），STOC，第 212–219 页。ACM。[Harris 等人， 2010] Harris, R., Johansson, J., Berkley, A., Johnson, M., Lanting, T., Han, S., Bunyk, P., Ladizinsky, E., Oh, T., Perminov, I. 等（2010）。稳健且可扩展磁通量子比特的实验演示。《物理评论B》, 81(13):134510。[Hu等人， 2019] Hu, F., Wang, B.-N., Wang, N. 和 Wang, C.（2019）。利用 D-Wave量子计算机进行量子机器学习。《量子工程》, 1(2):e12。[Ingmar 等人， 2020] Ingmar, L., de la Banda, M. G., Stuckey, P. J. 和 Tack, G.（2020）。解的多样性建模。载于 AAAI，第 1528–1535 页。AAAI出版社。

[Istrail， 2000] Istrail，S.（2000）。统计力学、三维性与 NP 完全性：I. 非平面曲面上伊辛模型配分函数不可解性的普适性（扩展摘要）。载于 Yao, F. F. 和 Luks, E. M. 编，STOC，第 87–96 页。ACM。

[Jackson 等， 2008] Jackson，B. G.、Schnable, P. S. 和 Aluru，S.（2008年）。将共识遗传图谱视为来自不一致来源的中位数序。IEEE ACM《计算生物学与生物信息学汇刊》，5(2)：161–171。

[Johnson等人， 2011] Johnson, M. W., Amin, M. H., Gildert, S., Lanting, T., Hamze, F., Dickson, N., Harris, R., Berkley, A. J., Johansson, J., Bunyk, P., 等人（2011年）。使用制造自旋进行量子退火。《自然》，473(7346):194–198。[King等人， 2022] King, A. D., Suzuki, S., Ray- mond, J., Zucca, A., Lanting, T., Altomare, F., Berkley, A. J., Ejtemaee, S., Hoskinson, E., Huang, S., 等人（2022年）。在可编程2000量子比特伊辛链中进行相干量子退火。arXiv预印本 arXiv:2202.05847。[Kinjo等人， 2003] Kinjo, M., Sato, S., 和 Naka- jima, K.（2003年）。量子神经网络的量子绝热演化算法。载于 Kaynak, O., Alpaydin, E., Oja, E., 和 Xu, L.（编），ICANN/ICONIP，第2714卷《计算机科学讲义》，第951–958页。施普林格。[Kirkpatrick等人， 1983] Kirkpatrick, S., Gelatt Jr, C. D., 和 Vecchi, M. P.（1983年）。通过模拟退火实现优化。《科学》，220(4598):671–680。[Koshka和 Novotny， 2020] Koshka, Y. 和 Novotny, M. A.（2020年）。比较D-Wave量子退火与经典模拟退火在局部极小值判定中的表现。IEEE信息论选定领域期刊，1(2):515–525。[McGeoch， 2014] McGeoch, C. C.（2014年）。《绝热量子计算与量子退火：理论与实践》。《量子计算综合讲义》。Morgan & Claypool出版社。

[Morita和Nishimori, 2008] Morita, S. 和 Nishimori, H.（2008年）。量子退火的数学基础。《数学物理学杂志》，49(12):125210。

[Ohzeki 和 Nishimori, 2011] Ohzeki, M. 和 Nishimori, H.（2011年）。量子退火：入门介绍与最新进展。《计算与理论纳米科学杂志》，8(6):963–971。

[Petit 和 Trapp, 2019] Petit, T. 和Trapp, A. C.（2019）。通过解工程丰富组合问题的解。《INFORMS计算杂志》，31(3):429–444。

[Schalekamp 和 van Zuylen, 2009] Schalekamp, F. 和 van Zuylen, A.（2009）。排名聚合：众人拾柴火焰高。载于 Finocchi, I. 和 Hershberger, J.（编），ALENEX，第38–51页。工业与应用数学学会。

[Shor, 1994] Shor, P. W.（1994）。量子计算算法：离散对数与因数分解。载于 FOCS，第124–134页。IEEE计算机学会。

[Simjour, 2009] Simjour, N.（2009）。改进的 Kemeny 聚合问题参数化算法。载于 Chen, J. 和 Fomin, F. V.（编），IWPEC，《计算机科学讲义》第 5917 卷，第 312–323 页。施普林格。

[Steffen 等, 2011] Steffen, M., DiVincenzo, D. P., Chow, J. M., Theis, T. N. 和 Ketchen, M. B.（2011）。量子计算：IBM 视角。《IBM研究与开发杂志》，55(5)：13–1。

[van Zuylen 和 Williamson, 2009] van Zuylen, A. 和 Williamson, D. P.（2009）。用于约束排序和聚类问题的确定性枢轴算法。《数学运筹学》，34(3):594–620。[Willsch等人, 2022] Willsch, D., Willsch, M., Gonzalez Calaza, C. D., Jin, F., De Raedt, H., Svensson, M. 和 Michielsen, K.（2022年）。使用精确覆盖问题对 D-Wave 2000Q 量子退火器的优势进行基准测试。《量子信息处理》，21(4):1–22。

[Young 和 Levenglick, 1978] Young, H. P. 和 Levenglick, A.（1978）。孔多塞选举原则的一种一致性扩展。《SIAM应用数学杂志》，35(2):285–300。