凸多边形最小覆盖：CG:SHOP 挑战赛 2023

SándorP. Fekete计算机科学系，布伦瑞克工业大学，德国 s.fekete@tu-bs.de

Phillip Keldenich
计算机科学系，布伦瑞克工业大学，德国
p.keldenich@tu-bs.de

Dominik Krupke
计算机科学系，布伦瑞克工业大学，德国 d.krupke@tu-bs.de

Stefan Schirra
仿真与图形系，马格德堡奥托·冯·格里克大学，德国
stschirr@isg.cs.uni-magdeburg.de

arXiv:2303.07007v1 [cs.CG] 13 Mar 2023

摘要

我们概述了 2023 年计算几何挑战赛，该赛事聚焦于凸多边形最小覆盖问题：用数量最少的凸子集覆盖给定的多边形区域（可能含有孔洞）。这一问题在计算几何领域由来已久。

2012 年 ACM 学科分类计算理论 → 计算几何；计算理论 → 算法设计与分析

关键词和短语计算几何、几何优化、最小覆盖、凸性、算法工程、竞赛

资助 布伦瑞克工业大学的相关工作部分得到了德国研究联合会（DFG）项目“计算几何：求解困难优化问题”（CG:SHOP）的资助，项目编号为 FE407/21-1。

致谢 我们感谢 CG:SHOP 挑战顾问委员会成员提供的宝贵意见：William J. Cook、Andreas Fabri、Dan Halperin、Michael Kerber、Philipp Kindermann、Joe Mitchell 和 Kevin Verbeek。此外，我们感谢 Efi Fogel 就 CGAL 中的一个问题为我们提供咨询，也感谢 Rouven Kniep 实现了一种简单的贪心方法，作为评估的基线。

1 引言

“CG:SHOP 挑战赛”（计算几何：求解困难优化问题）最初是 2019 年 6 月在俄勒冈州波特兰举行的 2019 年计算几何周（CG 周）上的一场研讨会。其目标是举办一场计算挑战竞赛，聚焦于一个特定的困难几何优化问题，鼓励研究人员设计并实现求解方法，并根据这些方法在精心挑选、各具特点的实例数据库上的表现，进行科学比较。计算几何研究大多具有理论性质，往往致力于为 NP 难优化问题寻找可证明的近似算法；而挑战赛的目标，则是以特定一组基准几何实例上的计算结果作为衡量成功与否的标准。2019 年挑战赛聚焦于这样一个问题：对于给定的平面顶点集，计算面积最小和最大的简单多边形。这次挑战赛

得到了众多研究团队的积极响应，这些团队来自计算几何和组合优化两个领域，并促成了求解思路的热烈交流。

在 2020、2021 和 2022 年的 CG 周上，挑战赛问题分别是最小凸划分、协调运动规划和最小平面子图划分。CG:SHOP 挑战赛成为 CG 周计划中的一项活动，表现最佳的解会发表于计算几何研讨会（SoCG）论文集。挑战赛的时间提前了，以便让更多人有机会参与，尤其是学生，例如将其作为课程项目的一部分。

2023 年的第五届挑战赛延续了这一形式，并促成了相关论文发表于 SoCG 论文集。共有 22 支团队报名，其中 18 支至少提交了一个有效解。

2 挑战赛：凸多边形最小覆盖

合适的竞赛问题应具备若干理想特性。

- 该问题具有几何性质，具有普遍的科学意义，并且此前已受到关注。优化问题通常比可
- 行性问题更适合作为竞赛题；原则上，可行性问题也可以采用，但必须适合进行足够细
- 粒度的评分，才能形成有趣的竞赛。对于规模合理的实例，计算最优解十分困难。这种困难根源于算法性质，而不仅仅是编码问题，或无法使用复杂软件、硬件所致。验证所提供解的可行性相对容易。
-
-

-

在举办的第五年，征集适合问题的通知于2022年6月发出。对此，共有六个有趣的问题被提议作为2023年挑战赛的题目。评选时考虑了问题的难度、与往届题目的差异，以及现有文献和相关工作。最终，顾问委员会选定了该问题。特别感谢 Dan Halperin（特拉维夫大学）提出这一问题，其灵感源于机器人学领域的应用 [4]。

2.1 问题

构成2023年 CG 挑战赛基础的具体问题如下；简单示例见图1。

问题： 凸多边形最小覆盖
给定： 在平面上给定一个几何区域 P，它可以是简单多边形，也可以是带孔多边形。

目标： 任务是用一组凸多边形 $C_1, \ldots, C_k$ 覆盖 P，其中每个凸多边形都包含在 P 内，并使覆盖中凸多边形的数量 k 最少。

这个问题的变体在计算几何领域有着悠久的历史；事实上，年度 SoCG 会议的重复徽标（见图2）表明，即使对于边平行于坐标轴的简单多边形区域，最小凸覆盖也可能需要使用不位于多边形边的延长线所形成的排列上的顶点；参见 [15]，由 O'Rourke 撰写，他是1985年 首届会议的程序委员会主席；以及

`https://www.computational-geometry.org/logo.html`。

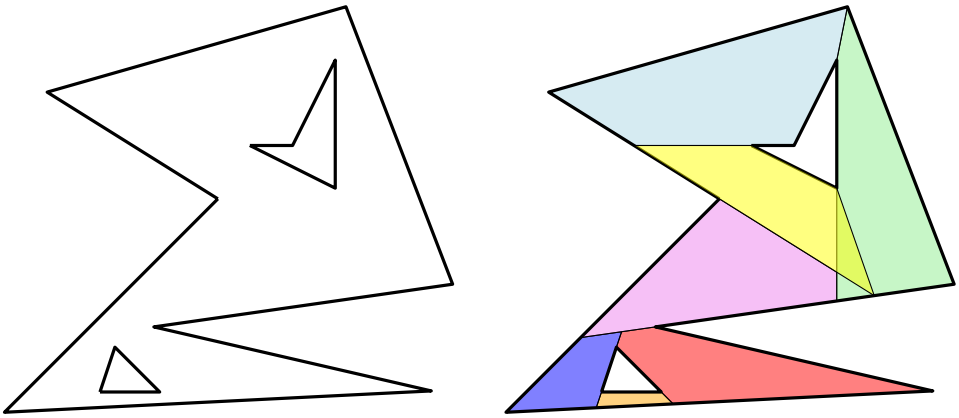


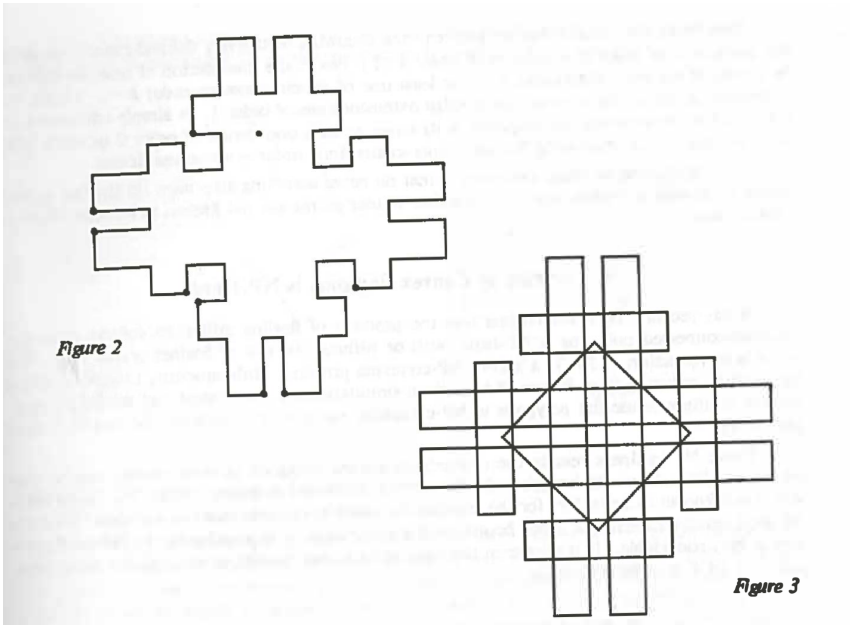
图1 一个可能的实例：平面中由一个（非简单）多边形区域给出（左），以及由凸集构成的一个可行覆盖（右）。

2.2 相关工作

即使在计算几何发展的早期，凸覆盖问题也备受关注 [13]。与之密切相关的多边形最小凸划分问题亦是如此，也就是将多边形覆盖为内部两两不相交的凸多边形。早在首届计算几何研讨会上，就有人研究了它的一个变体 [14]。当时，研究覆盖问题的动力来自形状分析与模式识别、计算机图形学以及超大规模集成电路设计等应用。研究重点多在于问题本身的内在复杂性，而较少关注算法解决方案的设计。

O'Rourke 首先证明了该问题的判定版本确实是可判定的 [16]。他展示了如何构造一个关于实数的存在公式：当且仅当给定多边形具有由 k 个凸集构成的凸覆盖时，该公式为真。尽管针对无孔洞简单多边形的最小凸划分问题，已有多项式时间算法被提出 [6], ，O'Rourke 和 Supowit [18] 证明了带孔洞多边形的最小凸覆盖问题是 NP困难的。几年后，Culberson 和 Reckhow [8] 证明，即使对于无孔洞的简单正交多边形，矩形最小凸覆盖仍然是 NP困难的。直到最近，Abrahamsen [1] 才成功证明凸覆盖问题是 $\exists\mathbb{R}$-完全的（即使是用三角形覆盖一个简单多边形），这意味着除非 NP $= \exists\mathbb{R}$，否则长期悬而未决的"该问题是否属于 NP"这一问题的答案是否定的。尽管 O'Rourke [15] 早已猜想最优解可能需要无理数坐标，但 Abrahamsen [1] 最终证明，对于具有整数坐标的多边形，在最优解中，各个部件的角点可能必需使用代数次数任意高的、具有无理数坐标的施泰纳点。这使得采用精确方法（例如整数规划或约束满足）变得不太可能直接实现。

在挑战赛中，只允许使用坐标为有理数的凸多边形片，一方面是为了避免就代数数的表示格式达成一致，另一方面是为了便于验证提交的结果。即使在这一受限版本中，仍允许使用坐标为有理数的施泰纳点。（据我们所知，对于有孔洞或无孔洞的一般简单多边形，目前尚无考虑这一限制的复杂性结果。）由于已知一般多边形问题是 NP困难，因此人们考虑了一些特殊情况，其中最著名的是用矩形覆盖正交多边形；引出 SoCG 标志的版本便是如此，见图2。然而，覆盖问题仍然



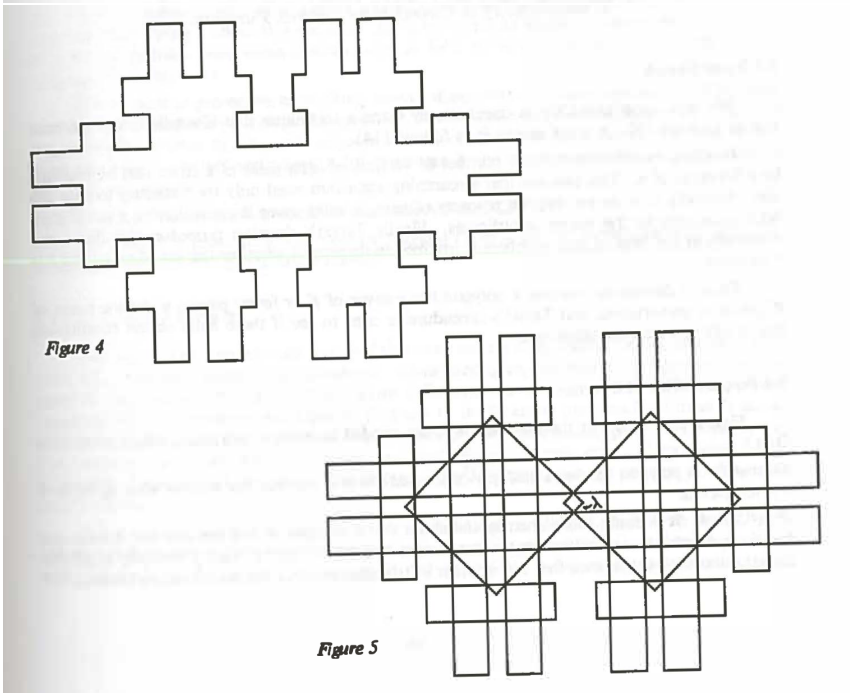


图2 SoCG标志与最小凸覆盖。一个简单多边形区域（左上，“图2”）以及由九个凸多边形组成的最小凸覆盖，其中一个是菱形（即旋转正方形），且其顶点不在该区域的任何边上（右上，“图3”）。将两个这样的多边形合并为一个（左下，“图4”），会得到一个这样的区域：其最优解需要用到甚至不在边的延长线上的施泰纳点（右下，“图5”）。所有图片均来自 O’Rourke [15]。

即使对于带孔洞的正交多边形，也属于 NP困难；为了寻找该问题的多项式时间可解版本，人们还研究了其他特殊类型的正交多边形 [12]。

除了将允许的覆盖形状从任意凸多边形限制为矩形和轴对齐矩形之外，也可以考虑三角形。然而，Christ [7]已证明，这一问题版本同样是 NP困难的，而 Abrahamsen [1] 还证明它是 $\exists\mathbb{R}$完全问题。

除了凸多边形，也可以考虑使用其他非凸多边形进行覆盖。如果考虑星形多边形，我们处理的就是美术馆问题 [17], ——另一个 $\exists\mathbb{R}$完全问题 [2], ，其最优解可能涉及复杂的代数坐标。

2.2.1 简单启发式

一种最初且简单的启发式方法是完全不使用施泰纳点。这样，所有凸片的顶点也必须是待覆盖多边形的顶点。此外，只需考虑满足以下极大性条件的凸片 C ：无法再添加另一个多边形顶点 v，使 $C \cup \{v\}$ 的凸包变得更大，同时仍包含于 P之中。因此，一种直接的方法是构造由 P的顶点集形成的所有极大凸多边形（或其中足够大的子集），然后使用某种搜索启发式算法选出一个子集来覆盖 P。对于更一般的方法，可以添加特定类型的施泰纳点。O'Rourke [15] 建议使用 P 在 P内各边的极大延伸端点。进一步而言，可以在下一阶段添加这些延伸线段的交点，或者更一般地，添加先前所考虑的施泰纳点对所确定直线的交点。

2.2.2 近似解

处理困难问题的另一种方法是寻找近似解。Eidenbenz 和 Widmayer [11] 证明，最小凸覆盖问题是 APX难问题，并提出了一种具有对数近似比的近似算法。该算法采用离散化和动态规划。

近似解有不同类型。除了近似最小化完全覆盖一个多边形所需的凸多边形数量外，也可以放宽覆盖要求，寻找近似覆盖该多边形的凸多边形，例如允许一定比例的区域未被覆盖，或允许凸多边形在一定程度上延伸到外部。在机器人学应用中，采用凸片进行保持连通性的近似覆盖通常就足够了。

2.3 实例

任何挑战赛的重要组成部分都是创建合适的实例。如果实例很容易求得最优解，挑战赛就会变得毫无难度；另一方面，如果实例需要大量计算，才能完成重要且常见的预处理步骤或找到任何尚可的解，那么挑战赛可能会严重偏向那些有能力购置更好计算设备的团队。如果实例集大到无法用一台（或少数几台）计算机管理，情况也一样。

我们使用以下生成器来生成实例；图3展示了相应的实际竞赛实例示例。

cheese cheese 生成器的目标是创建一个相对简单的外边界，其中包含大量小孔洞。为此，我们首先在一个较大的矩形区域内均匀随机地生成孔洞中心点。然后，我们从一个较小的可能取值范围内均匀随机地选择边界点的数量，通常为 3–6 个点；所有这些点都在孔洞中心附近的范围内均匀随机选取。随后，按照初始随机顺序依次访问这些点，将它们连成巡回路径；只要存在相交的边，就通过应用 2-opt 步骤缩短巡回路径，从而将其转化为没有自相交的多边形。为确保孔洞之间不会相交，也不会位于其他孔洞内部，每个新孔洞都会被加入一个二维排列中；如果它与先前加入的孔洞相交，就会被忽略，并在新的起始点周围生成一个新孔洞。生成所需数量的孔洞后，通过求取所有孔洞中心点的凸包来生成外边界，并在与孔洞相交的必要位置向外扩展。请注意，这可能会使外边界变为非凸。

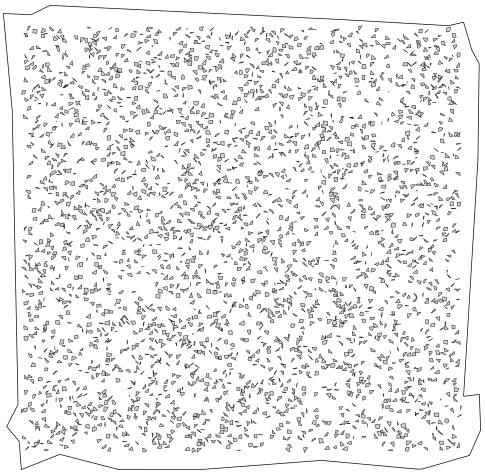
ccheese ccheese 生成器的工作方式与 cheese 生成器类似，但额外要求只生成凸孔洞；具体做法是将每个生成的孔洞替换为其顶点的凸包。进行这一修改的原因如下。尤其是对于较大的孔洞，非凸顶点往往需要用单独的凸片来覆盖。在许多情况下，这些凸片可以以唯一的方式转化为可行区域中的极大凸子区域；这可能会无意间使实例中的很大一部分变得容易求解。

srpg 有几个实例取自萨尔茨堡多边形数据数据库 [10]。这些实例从数据库中提取后经过归一化处理，因此所有坐标均为非负数。如果坐标不全为整数，我们会将所有坐标乘以一个较大的系数，再四舍五入到最近的整数，从而得到一个相似的整数坐标多边形。

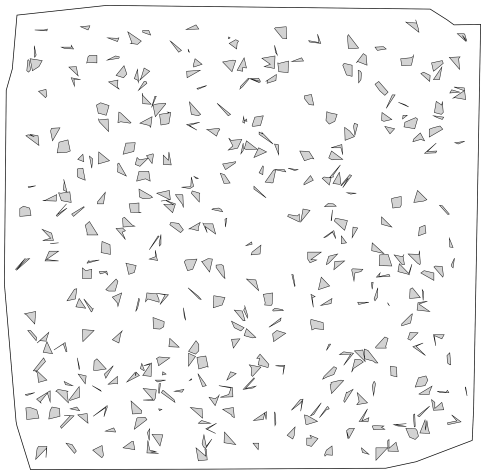
> `srpg`系列实例使用其超随机多边形生成器生成。前缀 `srpg_iso` 表示正交多边形；前缀 `srpg_iso_aligned`表示具有整数坐标的正交多边形，其中许多点具有相同的 x坐标或 y 坐标。前缀 `srpg_iso_octa` 表示八边形，即所有角度均为 45° 的倍数的多边形。最后，前缀 `srpg_smo` 和`srpg_smr` 表示带有平滑角的随机多边形，即通过添加额外顶点使角变得更平滑的多边形； `smr` 表示比 `smo`更强的平滑处理。

fpg 与 `srpg` 系列一样， `fpg` 系列实例取自萨尔茨堡多边形数据数据库 [10], ，并采用相同的方法对坐标进行归一化和整数化。这些实例使用 FPG（三角剖分扰动）生成器生成，该生成器通过移动初始多边形（例如正多边形）的顶点，同时保持边界的连通分量数量不变，对其进行变异。这通常会产生比 `srpg` 系列中常见情况更狭长的多边形部分。

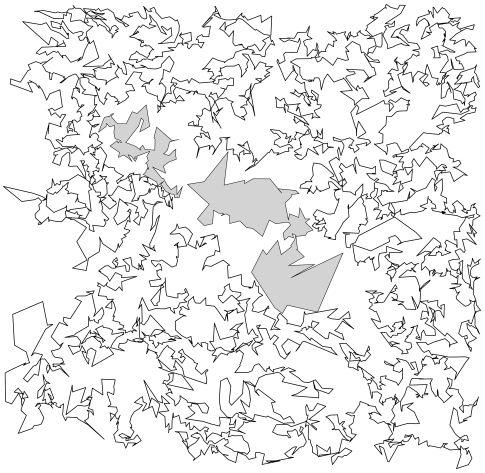
迷宫 迷宫生成器生成外边界相对简单的多边形，并以网格状排列的方式在其中放置大量方形障碍物，从而留下长长的自由空间走廊。要获得良好的解，需要使用高度重叠的多边形来覆盖这些走廊。随后会移除部分障碍物；另一些障碍物则会通过向外移动其部分顶点而略微扰动。这些扰动旨在排除一些简单的方法，例如用单个凸片覆盖每条走廊。



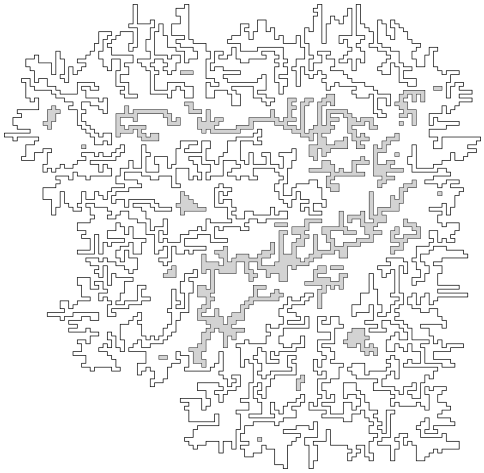
(a) 实例 `ccheese11045`。



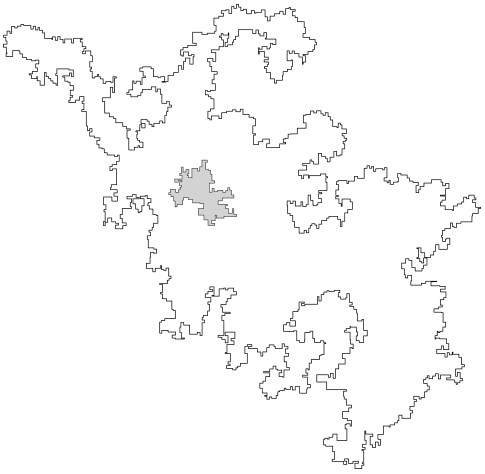
(b) 实例 `cheese1516`。



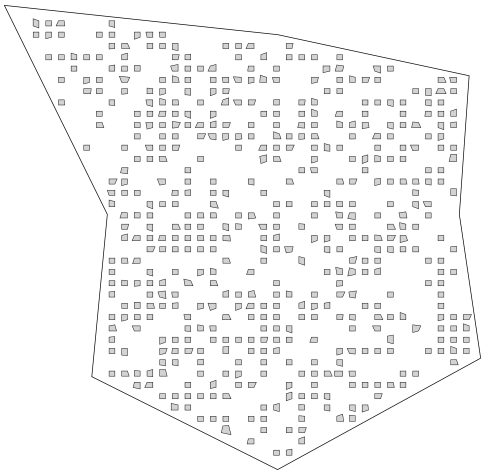
(c) `fpg-poly_0000004900_h2`。



(d) `srpg_iso_aligned_mc03255`。



(e) `srpg_iso_mc0001964`。



(f) `maze_2534_250_05_01`。

图3 由不同生成器生成的一组选定的实际竞赛实例；灰色区域为孔洞。

2.4 评估

本次竞赛共使用了 206 个实例。

对于许多优化问题，找到最优值的解，往往比找到一个接近最优值的解要困难得多，例如，其值为 OPT $+ c$，其中 c 是某个较小的常数。

我们猜想竞赛问题也是如此。为了在解的评分中体现这一点，我们没有采用与多边形数量呈线性关系的得分，而是引入了二次评分函数。对于一个实例 I，令 $B(I)$ 表示该实例提交的所有解中，所需凸片数量最少的解所用的凸片数。此外，令$T(I)$表示团队 T 为 I提交的最佳解中的凸片数量。团队 T 在实例 I 上的得分$S_T(I)$为

$$S_T(I) := \frac{B(I)^2}{T(I)^2}.$$

因此，与已知最佳解相比，凸片数量翻倍会使得分降至 0.25；对于某个实例 I，所有提交的解均未被其他团队超越的团队，在 I上均得 1 分。对于某个实例 I，未提交任何有效解的团队将获得默认得分 0（对应于使用无限多个凸片的解）。随后，将各团队 T 在所有实例 I上的得分 T 相加，计算出每个团队的总得分 $S_T = \sum_I S_T(I)$。得分最高的团队获胜。如果出现平局，则以达到特定总分的时间作为决胜规则。与往年一样，最终并未用到这一规则。

2.5 类别

竞赛分为公开组，参赛者可以使用任意计算设备、任意时长的计算时间（在竞赛持续期间内），并自行组队。青少年组要求团队成员必须全部符合 CG:YRF（CG 周青年研究者论坛）的规则，即在 2021 年之前尚未通过正式博士学位答辩。

2.6 服务器与时间线

竞赛通过布伦瑞克工业大学的一台专用服务器运行，托管于 `https://cgshop.ibr.cs.tu-bs.de/competition/cg-shop-2023/`。服务器于 2022 年 9 月 30 日 00:00（UTC）开放，提供若干测试实例；完整的竞赛实例集于 2022 年 10 月 28 日发布，并于 2023 年 1 月 27 日 24:00（午夜，AoE）关闭。

竞赛期间，服务器用于验证提交内容的代码也以 Python 包的形式发布在 Python 包索引（PyPI）上，供参赛者使用[1]。除了针对编码错误进行简单的有效性检查外，提交内容还会经过严格验证，以确认它们是各自实例的有效解。这包括检查每个解中所有区域的凸性、确认这些区域的并集覆盖整个区域，以及确保没有覆盖额外区域。借助 CGAL [20] 库，这些检查相对容易实现；不过，将作为解提交的浮点数、整数或有理数转换为 CGAL 的精确数值类型时，必须格外谨慎。对各种多边形进行的大量布尔运算，最终成了

[1] 参见 `https://pypi.org/project/cgshop2023-pyutils/`。

一次严峻的压力测试，并且实际上帮助发现了 CGAL 中一个基于折线的新 `join`算法的漏洞[2]，该漏洞后来已得到修复。

3 项结果

共有22支团队报名参加比赛，其中18支团队至少提交了一个有效解。最终，排名前10的团队排行榜如表1所示。有两支团队（DIKU (AMW) 和 Shadoks）的表现远超其他所有参赛者。

Rank	Team	得分	青少年
1	DIKU (AMW)	201.571	
2	Shadoks	198.347	
3	BX23	144.150	
4	SmartLab	142.925	✓
5	agr	122.400	
6	rkPlayground	121.311	✓
7	Karteflan	103.392	✓
8	Ofir	103.223	
9	pjgblt	103.127	✓
10	cgl@tau	100.893	

表1 最终得分排名前10的团队，得分四舍五入至小数点后三位。符合初级团队认定标准的团队会在“初级”栏中标记勾号。

图4展示了各团队得分随时间的变化；图5则展示了所有实例的最佳解（按得分显示）。排名前两位的团队受邀为2023年SoCG会议论文集撰稿，具体如下。

1. DIKU团队（AMW）：Mikkel Abrahamsen、William Bille Meyling、André Nusser [3]。2. Shadoks团队：Guilherme D. da Fonseca [9]。
他们的方法细节及所作的工程决策，详见各自的论文。图6简要评估了他们各自的优势与不足。下面简要介绍他们采用的方法。

DIKU团队（AMW） [3] 以给定多边形 P 顶点的约束德劳内三角剖分为基础，并加入一些额外点，例如 P线段延长线的交点。在此三角剖分上，他们计算出一个可见性图，其中每个三角形对应一个顶点；若一对三角形的凸包完全包含在 P内，则在它们之间连一条边。他们注意到，该图中的团对应于由团内三角形诱导出的凸多边形。有时，这些诱导出的多边形未必完全包含在 P内，但他们认为这种情况相对少见。随后，他们使用 Strash 和 Thompson 为顶点团覆盖问题开发的现有实现 ReduVCC [19]，计算出数量较少且覆盖所有三角形的团。最后，他们修补未完全包含在 P 内的凸片，并移除那些仅覆盖 P 中已被其他凸片覆盖部分的凸片。

Shadoks团队 [9] 采用了不同的方法。从整体上看，他们的方法包括生成一个所谓的集合 C，也就是一组包含在 P中的凸片，它们合起来覆盖所有 P。虽然不能让 C 变得过大很重要，但目标是

[2] https://github.com/CGAL/cgal/issues/7235

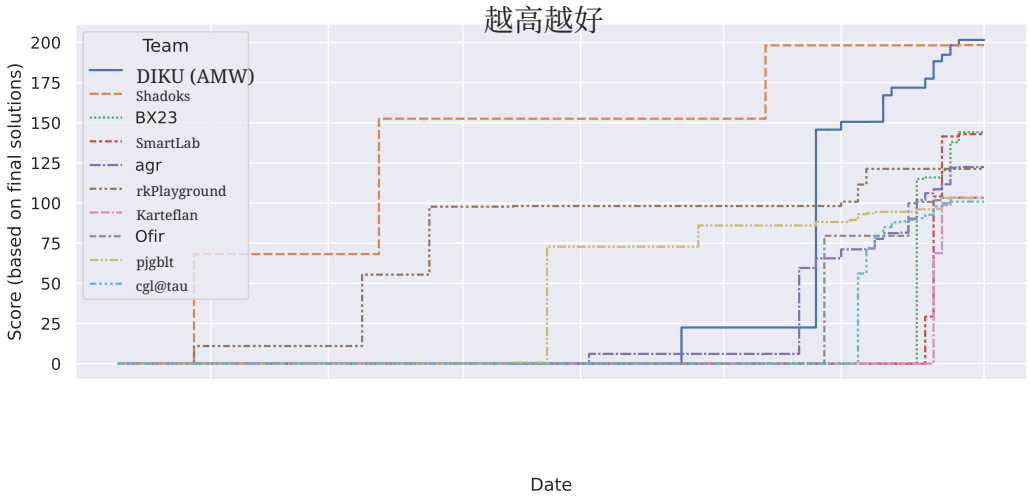


图4 前 10 个团队随时间变化的得分。得分根据最终提交计算。*Shadoks* 团队一直保持领先，直到最后几天被*DIKU (AMW)* 团队以微弱优势反超。大多数团队直到最后两周才开始提交正式作品。许多团队都能看到迭代改进的过程。

生成一个优质的 C 时，只需得到一个较小的解 $S \subseteq C$，同时覆盖所有 P。有了优质的 C 后，Shadoks会将寻找优质解 $S \subseteq C$ 的任务转化为一个规模适中的集合覆盖问题实例，并使用整数规划求解器求得最优解，或使用模拟退火进行启发式求解；为此，他们在 P 中引入见证点，并要求每个见证点至少被S中的一个集合覆盖。他们提出了几种生成 C的方法；一种基于经过修改的Bron-Kerbosch 算法 [5]来枚举极大团，另一种是他们称为随机膨胀的过程。

为了评估相对简单、直接的方法能够取得怎样的得分，我们实现了以下类型的启发式算法。从多边形 P 的顶点构造约束德劳内三角剖分开始，我们通过贪心方式合并当前细分中的任意选定面，形成用于覆盖P的凸片，直到无法再合并任何面。按照这一方案实现的程序本可获得 85.3 分，从而在挑战赛中排名第 12。

这表明顶尖团队的方法相较于简单、直接的方法具有巨大优势；同时也说明，大多数实际参赛者确实提出了能够胜过这类方法的算法。

4 结论

2023 年 CG:SHOP 挑战赛促使众多团队开展了广泛的优化研究。这些成果有望进一步揭示这一基础性重要优化问题的内在规律。此外，初级团队的踊跃参与表明，挑战赛本身激励了大量学生和青年研究人员投身于实际的算法问题。

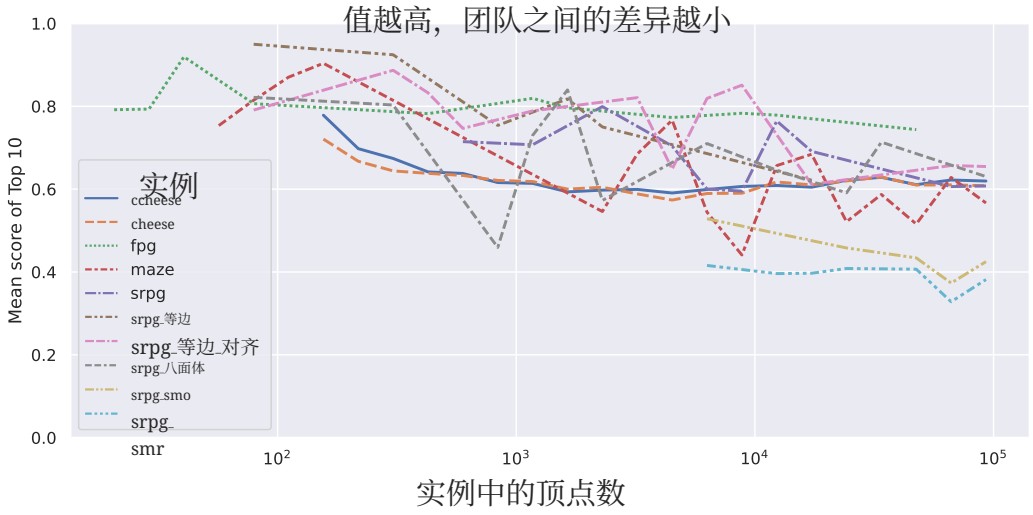


图5 不同实例中排名前十的团队的平均得分，可以反映这些实例的难度。平均得分越高，说明团队越能跟上顶尖团队。若平均得分较低，则表明要跟上顶尖团队还需要额外且不平凡的思路。团队可能遇到的首个明显挑战是可扩展性，因为实例规模越大，平均得分越低。此外， `ccheese` 和 `cheese` 实例即使在中等规模下似乎也格外棘手，且两者难度相近。对于更大规模的实例， `srpg` 和`maze` 实例也颇具挑战性；经过平滑处理的 `srpg`-实例 `smo` 和 `smr` 则是所有实例中最具挑战性的。 `fpg` 实例似乎相对容易。

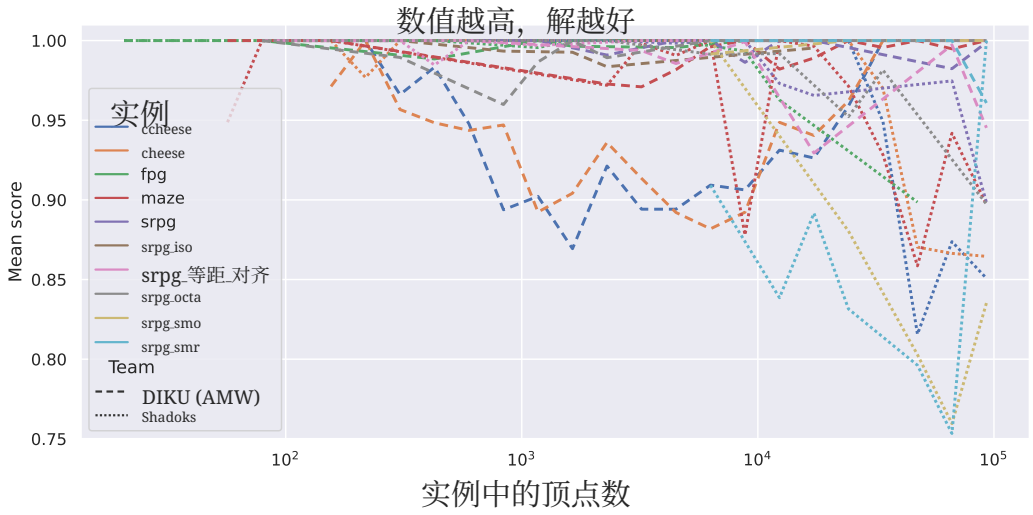


图6 排名前两位的团队得分相近，但各有所长。团队*Shadoks* 在 `cheese` 和 `ccheese` 实例上表现更好，而*DIKU(AMW)* 的方法似乎具有更好的可扩展性，因此在大型实例中领先。在 `maze` 实例中，不同实例的领先团队有所不同。

参考文献

1 M. Abrahamsen. 覆盖多边形甚至更加困难。载于计算机科学基础研讨会（FOCS），第 375–386 页，2021。 2 M. Abrahamsen、A. Adamaszek 和 T. Miltzow. 艺术画廊问题是 ∃ℝ-完全的。《ACM 杂志》，69(1):4:1–4:70，2022。 3 M. Abrahamsen、W. B. Meyling 和 A. Nusser. 通过团覆盖构造简洁凸覆盖。载于计算几何研讨会（SoCG），2023。 4 A. Amice、H. Dai、P. Werner、A. Zhang 和 R. Tedrake. 为机器人机械臂在构形空间中寻找并优化经认证的无碰撞区域。载于机器人算法基础第十五届会议（WAFR），Springer《先进机器人学会议论文集》第 25 卷，第 328–348 页，2022。 5 C. Bron 和 J. Kerbosch. 算法 457：寻找无向图的所有团。《ACM 通讯》，16(9):575–577，1973。 6 B. Chazelle 和 D. P. Dobkin. 最优凸分解。载于 G. T. Toussaint 编，《计算几何》，机器智能与模式识别第 2 卷，第 63–133 页。North-Holland，1985。 7 T. Christ. 超越三角剖分：用三角形覆盖多边形。载于算法与数据结构研讨会（WADS），第 231–242 页，2011。 8 J. C. Culberson 和 R. A. Reckhow. 覆盖多边形是困难的。《算法杂志》，17(1):2–44，1994。 9 G. D. da Fonseca. 凸覆盖的 Shadoks 方法。载于计算几何研讨会（SoCG），2023。 10 G. Eder、M. Held、S. Jasonarson、P. Mayer 和 P. Palfrader. 萨尔茨堡多边形数据数据库：多边形及其生成器。《简报数据》，31:105984:1–8，2020。 11 S. J. Eidenbenz 和 P. Widmayer. 一种具有对数级性能保证的最小凸覆盖近似算法。《SIAM 计算杂志》，32(3):654–670，2003。 12 J. M. Keil. 多边形分解。载于 J. Sack 和 J. Urrutia 编，《计算几何手册》，第 491–518 页。North Holland / Elsevier，2000。 13 J. M. Keil 和 J.-R. Sack. 多边形对象的最小分解。载于 G. T. Toussaint 编，《计算几何》，机器智能与模式识别第 2 卷，第 197–216 页。North-Holland，1985。 14 A. Lubiw. 将多边形区域分解为凸四边形。载于计算几何研讨会（SoCG），第 97–106 页，1985。 15 J. O’Rourke. 计算多边形最小凸覆盖的复杂性。载于第二十届年度阿勒顿会议论文集，1982年10月，第 75–84 页，1982。 16 J. O’Rourke. 凸多边形覆盖的可判定性。技术报告 JHU-EECS 82-4，约翰斯·霍普金斯大学，1982。 17 J. O’Rourke.《艺术画廊定理与算法》。牛津大学出版社，1987。 18 J. O’Rourke 和 K. J. Supowit. 一些 NP 难多边形分解问题。《IEEE 信息论汇刊》，29(2):181–189，1983。 19 D. Strash 和 L. Thompson. 顶点团覆盖问题的有效数据缩减。载于算法工程与实验研讨会（ALENEX），第 41–53 页，2022。 20 CGAL 项目。《CGAL 用户与参考手册》。CGAL 编辑委员会，5.5.2 版，2023。