arXiv:2611.31155v1 [cs.SC] 23 Nov 2026

计算 3×3 矩阵乘法的方法多种多样：秩 23 方案大型数据集

预印本

Andrew PerminovTAI 系
统编程研究所研究中心 莫斯科
perminov@ispras.ru

2026年11月24日

摘要

本文介绍了一个公开可用的数据集，其中包含 62 029 个用于 3×3 矩阵乘法的秩 23 方案。数据集中的每个方案都有独特的秩模式；秩模式是由方案的因子矩阵的秩构成的不变量，因此所有方案两两不等价。这些方案通过一种结合梯度下降与交替最小二乘的数值方法，以及翻转图搜索获得，其系数为三元、整数或有理数。秩 23 是这种格式目前已知的最佳秩，对于整数系数方案而言则是最优秩，但人们对这一秩下本质不同方案的数量知之甚少：Heule 等人和 Mårtensson 等人的现有集合包含许多秩模式数量很少的方案（分别有 17 376 个方案、387 种秩模式，以及 15 316 个方案、568 种秩模式）。该数据集表明，不同秩模式的数量远超此前所知。

关键词 快速矩阵乘法 · 3×3 矩阵乘法 · 秩 23 · 翻转图 · 数据集

1 引言

Strassen 算法 [1] 用七次乘法而非八次乘法来计算两个 2×2 矩阵，递归应用该算法可得到指数 $\log_2 7 \approx 2.807$。对于这种格式，七次乘法已是最优的 [2, 3]。下一个方形格式 3×3则难得多。Laderman 方案 [4] 使用 23 次乘法，对应的指数为 $\log_3 23 \approx 2.854$，因此并未优于 Strassen 算法。尽管如此，3×3 仍是秩在特征为零的域上尚未确定的最小方形格式：目前只知道其秩介于 21 [5] 和 23 之间。在 $\mathbb{F}_2$ 上的秩最近已被证明恰为 23 [6], ；通过模 2 约化，这意味着对于具有整数系数的方案，23 也是最优的。

Laderman 方案并非唯一。Johnson 和 McLoughlin 描述了秩 23 方案的参数化族 [7], ，Heule 等人 [8] 则使用基于 SAT 的局部搜索，获得了 17,376 个系数为三元值的两两不等价方案。Mårtensson 等人 [9, 10] 后来发表了一组通过神经网络流程获得的 15,316 个三元方案。近年来，计算机搜索已成为小型格式中的标准工具，其基础包括

强化学习 [11], 、约束编程 [12], 、翻转图 [13–19] 以及由大语言模型驱动的进化搜索 [20]。例如，Sedoglavic [21],、Perminov [22] 和 Lacelle [23]维护着已知最佳方案的目录。
大量的秩 23 方案十分有价值，因为秩相同的方案在实践中重要的属性上可能有所不同，而这些属性无法由秩决定。其中最显著的一项是计算方案所需的加法次数。Laderman 的原始构造需要 98 次加法，之后可将其减少到 62 [24]。此后，加法次数进一步降至 60、59、58、56 和 55 次；在替代基下，还降至 52 和 51 次。这些改进并非针对 Laderman 方案，而是针对其他秩 23 方案 [25–32] 获得的。因此，底层方案的选择很重要，方案集合越多样，就越有可能取得进一步改进。此外，多样化的

分解为更详细地研究矩阵乘法张量的结构和性质创造了机会。

这引出了一个自然的问题：本质上不同的秩 *23* 方案有多少种？如果两个方案可以通过矩阵乘法张量的对称性（第 3 节）相互转换，就认为它们等价。Heule 等人提出了这种等价关系的不变量，其中最基本的是秩模式：秩模式不同的方案必然不等价。将这些不变量应用于 [8, 33]，这一集合后，揭示出一个有趣的图景：其中 17,376 个两两不等价的方案仅对应 387 种不同的秩模式。 [9] 中的 15,316 个方案仅对应 568 种秩模式。换言之，在这两个集合中，许多方案共享同一种秩模式。本文探索相反的方向，并表明不同秩模式的数量也非常庞大。

贡献。

- 本文提出一个包含 62,029 个用于 3×3 矩阵乘法的秩 23 方案的数据集。这些方案通过梯度下降与交替最小二乘（GradALS）以及翻转图搜索（针对三元系数，以及针对 $\mathbb{F}_2$ 和 $\mathbb{F}_3$ 并结合亨泽尔提升）找到；其中超过一半由前者生成。每个方案都有其独特的秩模式，因此所有方案两两不等价。21,585 个方案（34.80%）的系数为三元系数，22,130 个（35.68%）为整数，18,314 个（29.52%）为有理数。
- Heule 等人和 Mårtensson 等人的方案并未作为搜索的输入，但这两个集合中的全部 771 种秩模式都包含在该数据集中。该数据集中的所有其他方案均与这两个集合中的每一个方案都不等价。加上 Heule 等人的两两不等价方案，该数据集至少包含 78634 个两两不等价的秩 23 方案（第 5.5 节）。
- 分析了数据集中非零系数数量的分布。GradALS 方法找到了一个含有 138 个非零系数的方案，而此前已知的公开集合中，没有任何方案的非零系数少于 139 个。该方案使用的系数为 $\{0, \pm 1/2, \pm 1, \pm 2\}$。后来，有人在一篇博客文章中发表了一个非零系数数量相同且秩模式相同的方案 [34]。由于具有这种秩模式的三元方案已纳入数据集，因此该方案在附录 A 中单独列出。
- 所有方案均提供 JSON、纯文本（.txt）和 Maple（.m）格式，并附有用于验证其正确性的脚本。包含数据集和工具的代码仓库见第 8 节。

本文结构如下。第 2 节确定符号记法，第 3 节介绍用于区分各方案的对称性和不变量。第 4 节说明方案的发现过程，第 5 节介绍数据集，第 6 节讨论结果和局限性。第 7 节和第 8 节分别给出结论，以及可用数据和代码的说明；附录 A 则收录了最稀疏方案。

2 预备知识

2.1 矩阵乘法方案

设 $\mathbb{F}$ 为一个域，设 $A, B \in \mathbb{F}^{n \times n}$ 为两个矩阵，其元素属于 $a_{i,j}$ 和 $b_{i,j}$。秩为 r 的一种方案，用于计算 $n \times n$ 矩阵的乘积，记作 $(n, n, n{:}r)$，是一种通过 r 次线性形式乘法来计算 $C = AB$ 各元素的方法：

$$m_l = \Big(\sum_{i,j=1}^{n} u_{i,j}^{(l)}\, a_{i,j}\Big) \cdot \Big(\sum_{i,j=1}^{n} v_{i,j}^{(l)}\, b_{i,j}\Big), \qquad l = 1, \ldots, r,$$

$$c_{i,j} = \sum_{l=1}^{r} w_{j,i}^{(l)}\, m_l, \qquad i, j = 1, \ldots, n,$$

其中 $u_i^{(l)}{}_{,\ j}$，$v_i^{(l)}{}_{,\ j}$，$w_i^{(l)}{}_{,\ j} \in \mathbb{F}$。

记号 2.1。 这些系数汇集在三个矩阵 U，V，$W \in \mathbb{F}^{r \times n^2}$ 中，其元素按照有序对 $(i,\ j)$ 的行优先顺序编索引：

$$u_{i,j}^{(l)} = U_{l,\,(i-1)n+j}, \qquad v_{i,j}^{(l)} = V_{l,\,(i-1)n+j}, \qquad w_{i,j}^{(l)} = W_{l,\,(i-1)n+j}.$$

不带下标的符号 $u^{(l)}$、$v^{(l)}$ 和 $w^{(l)}$ 表示 U、V 和 W 的第 l 行，它们也被视为元素为 $u_i^{(l)}{}_{,\ j}$、$v_i^{(l)}{}_{,\ j}$ 和 $w_i^{(l)}{}_{,\ j}$ 的 $n \times n$ 矩阵。此类矩阵的秩记为 $\mathrm{rk}\ u^{(l)}$，以区别于方案的秩 r。

令 vec 表示矩阵按行优先顺序进行的向量化。则该方案计算

$$\operatorname{vec}(C^T) = W^T\big((U\operatorname{vec}(A))\odot(V\operatorname{vec}(B))\big), \tag{1}$$

其中 $\odot$ 是逐元素（Hadamard）乘积。换言之，该方案由三元组 $(U,\ V,\ W)$ 完整描述。

备注 2.1。 矩阵 W 以转置形式存储：$w_{i}^{(l)}{}_{,\ j}$ 是 m_l 在乘积的条目 $c_{j,\ i}$ 中所对应的系数，而不是 $c_{i,\ j}$。这对应于通过迹形式 $\operatorname{tr}(ABC)$ 表示矩阵乘法的方式，也是大多数方案目录以及第 5 节所述数据集文件采用的编码。如下文所示，这种表示方式还使三个矩阵具有对称的地位。

例 2.1。 Strassen 的原始方案 [1] 是一个在 $\mathbb{Z}$ 上定义的方案 $(2,\ 2,\ 2:7)$:

$$\begin{aligned}
m_1 &= (a_{1,1}+a_{2,2})(b_{1,1}+b_{2,2}) & c_{1,1} &= m_1+m_4-m_5+m_7\\
m_2 &= (a_{2,1}+a_{2,2})\,b_{1,1} & c_{1,2} &= m_3+m_5\\
m_3 &= a_{1,1}(b_{1,2}-b_{2,2}) & c_{2,1} &= m_2+m_4\\
m_4 &= a_{2,2}(b_{2,1}-b_{1,1}) & c_{2,2} &= m_1-m_2+m_3+m_6\\
m_5 &= (a_{1,1}+a_{1,2})b_{2,2}\\
m_6 &= (a_{2,1}-a_{1,1})(b_{1,1}+b_{1,2})\\
m_7 &= (a_{1,2}-a_{2,2})(b_{2,1}+b_{2,2})
\end{aligned}$$

2.2 Brent 方程

三元组 (U,V,W) 定义了一个方案，当且仅当其系数满足 Brent 方程 [3 5]:

$$\sum_{l=1}^{r} u_{i_1,i_2}^{(l)}\, v_{j_1,j_2}^{(l)}\, w_{k_1,k_2}^{(l)} = \delta_{i_2 j_1}\,\delta_{j_2 k_1}\,\delta_{k_2 i_1} \tag{2}$$

对所有指标 $1\le i_1,i_2,j_1,j_2,k_1,k_2\le n$，其中 δ_{ij} 是克罗内克 δ，当 $i=j$ 时等于 1，否则等于 0。

2.3 张量表示

定义 2.1 (矩阵乘法张量)。 矩阵乘法张量 $T_{n,\ n,\ n}\in\mathbb{F}^{n^2\times n^2\times n^2}$ 是一个三维张量，其元素为

$$(T_{n,n,n})_{(i_1-1)n+i_2,\ (j_1-1)n+j_2,\ (k_1-1)n+k_2} = \delta_{i_2 j_1}\,\delta_{j_2 k_1}\,\delta_{k_2 i_1}.$$

定义 2.2（CP 分解 ）。 将张量 T 分解为 r 个秩一张量之和的表达式称为分解

$$T=\sum_{l=1}^{r} u^{(l)}\otimes v^{(l)}\otimes w^{(l)},$$

其中 $\otimes$表示外积 。存在这种分解时， r 的最小值称为 T的张量秩。

将（2）的右侧与 $T_{n,\ n,\ n}$ 的定义进行比较可知，对于 $n\times n$ 矩阵，秩为 r 的方案恰好对应于将 $T_{n,\ n,\ n}$ 分解为 r 个秩一张量；而方案的秩就是该分解的长度。

例 2.2。 上一个例子中的 Strassen 方案对应于以下分解

$$\begin{aligned}
T_{2,2,2} = {}& (a_{1,1}+a_{2,2})\otimes(b_{1,1}+b_{2,2})\otimes(c_{1,1}+c_{2,2})\\
&+(a_{2,1}+a_{2,2})\otimes(b_{1,1})\otimes(c_{1,2}-c_{2,2})\\
&+(a_{1,1})\otimes(b_{1,2}-b_{2,2})\otimes(c_{2,1}+c_{2,2})\\
&+(a_{2,2})\otimes(b_{2,1}-b_{1,1})\otimes(c_{1,1}+c_{1,2})\\
&+(a_{1,1}+a_{1,2})\otimes(b_{2,2})\otimes(c_{2,1}-c_{1,1})\\
&+(a_{2,1}-a_{1,1})\otimes(b_{1,1}+b_{1,2})\otimes(c_{2,2})\\
&+(a_{1,2}-a_{2,2})\otimes(b_{2,1}+b_{2,2})\otimes(c_{1,1}).
\end{aligned}$$

由于对 W 采用了转置编码，每一项的第三个因子都用基 $c_{j,\ i}$ 表示：例如，在 Strassen 方案中， m_2 进入 $c_{2,\ 1}$ 和 $-c_{2,\ 2}$，而第二项的第三个因子是 $c_{1,\ 2}-c_{2,\ 2}$。

2.4 系数类别与稀疏度

本文考察的方案的系数是有理数。根据它们所取值的集合，这些方案分为三类：系数位于 $\mathbb{Z}_T = \{-1, 0, 1\}$ 中的三元 方案、系数位于 $\mathbb{Z}$ 中的整数 方案，以及系数位于 $\mathbb{Q}$ 中的有理数 方案。一个方案的类别是包含其全部系数的最小类别，因此这些类别按 $\mathbb{Z}_T \subset \mathbb{Z} \subset \mathbb{Q}$ 排列。三元方案最为方便：它们适用于任意域，并且不需要乘以常数。

方案的稀疏度 由其非零系数的总数 $\mathrm{nnz}(U) + \mathrm{nnz}(V) + \mathrm{nnz}(W)$衡量。方案的非零系数越少，它所计算的线性形式就越简单，直接计算它所需的（朴素）加法也越少。

3 对称性与不变量

矩阵乘法具有多种对称性，因此可以从已知方案得到新方案。以下每种变换都会将一个秩为 r 的方案映射为秩相同的方案：

(a) 项的重排：三元组 $(u^{(l)}, v^{(l)}, w^{(l)})$ 可以任意置换。(b) 重新缩放：$(u^{(l)}, v^{(l)}, w^{(l)}) \mapsto (\alpha u^{(l)}, \beta v^{(l)}, \gamma w^{(l)})$，其中 $\alpha, \beta, \gamma \in \mathbb{F}$ 非零，且 $\alpha\beta\gamma = 1$。由于三个因子的乘积不变，因此每一项都可以分别选择这些常数。(c) 夹乘：$(u^{(l)}, v^{(l)}, w^{(l)}) \mapsto (Pu^{(l)}Q^{-1},\ Qv^{(l)}R^{-1},\ Rw^{(l)}P^{-1})$，对所有 l均成立，其中 $P, Q, R \in \mathbb{F}^{n\times n}$ 为可逆矩阵。它基于恒等式 $(PAQ^{-1})(QBR^{-1}) = P(AB)R^{-1}$：可以对变换后的矩阵应用方案，再将结果变换回来。(d) 循环移位：$(U, V, W) \mapsto (V, W, U)$。(e) 转置：(

$$u^{(l)}\ v^{(l)}\ w^{(l)} \quad \mapsto \quad v^{(l)\ T}\ \ u^{(l)\ T}\ \ w^{(l)\ T} \qquad l \qquad AB\ ^{T} =, ,)$$

$B^T A^T$
(( ) , ( ) , ( ))，对所有 均成立。它体现了恒等式 ( ) 。

变换 (d) 和 (e) 是 W 的转置编码的结果，这种编码使三个矩阵 U、V 和 W 发挥对称的作用：矩阵乘法张量对应于三线性形式 $\mathrm{tr}(ABC)$，该形式在循环移位 $\mathrm{tr}(ABC) = \mathrm{tr}(CAB)$ 下保持不变，并且在转置 $\mathrm{tr}(ABC) = \mathrm{tr}(C^T B^T A^T)$ 下也保持不变。

变换 (c)–(e) 是张量 $T_{n、\,n、\,n}$ 本身的对称性：夹乘是 de Groote [36], 所研究的群 $\mathrm{GL}_n^{\times 3}$的作用，而 (d) 和 (e) 则置换三个张量因子（在 (e) 的情况下，还要同时对矩阵 $u^{(l)}$、$v^{(l)}$、$w^{(l)}$进行转置），并生成一个与对称群 S_3 [37]同构的群。它们共同生成 $T_{n、\,n、\,n}$ [36, 38]的完整对称群。相比之下，变换 (a) 和 (b) 是将张量表示为秩一项之和时的平凡对称性。

定义 3.1 (方案的等价性)。 如果一个方案可以通过有限次应用变换 (a)–(e) 从另一个方案得到，那么这两个方案就是等价的。与给定方案等价的所有方案构成的集合称为它的轨道。

这些变换既不保持系数类别，也不保持非零系数的数量：用适当的矩阵进行夹乘，可以将三元方案变为整数或有理数方案，反之亦然。因此，夹乘是一种探索方案轨道、并在其中寻找更稀疏或“更优雅”代表元的方法；第4节所述的搜索正是以这种方式进行的。

对于 $n = 2$，所有秩为 7 的方案都属于同一个轨道，即 Strassen 方案 [36] 的轨道。对于 $n = 3$，情况则截然不同：秩 23 方案存在大量轨道，区分这些轨道正是本节余下内容的主题。

3.1 秩模式

区分轨道的一种便捷方法是考察矩阵 $u^{(l)}$、$v^{(l)}$ 和 $w^{(l)}$ 的秩，这是 Heule 等人 [8] 提出的方法。对于项为 $(u^{(l)},\ v^{(l)},\ w^{(l)})$ 的方案，考虑这些三元组

$$\rho_l = \big(\mathrm{rk}\, u^{(l)},\, \mathrm{rk}\, v^{(l)},\, \mathrm{rk}\, w^{(l)}\big), \qquad l = 1, \dots, r,$$

并对所有三元组应用相同的坐标排列。这样的排列共有六种（包括恒等排列），每种排列都会得到一个由 r 个三元组组成的列表，随后按字典序排序。

定义 3.2 （秩模式）。一个方案的秩模式是上述六个已排序列表中字典序最小的列表。

例 3.1。 Laderman 方案具有以下秩序列 ρ_l:

$$\begin{array}{lcccccccccccccccccccccc} \operatorname{rk} u^{(l)} & 3 & 1 & 1 & 2 & 1 & 1 & 2 & 1 & 1 & 3 & 1 & 2 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \operatorname{rk} v^{(l)} & 1 & 1 & 3 & 2 & 1 & 1 & 2 & 1 & 1 & 1 & 3 & 2 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \operatorname{rk} w^{(l)} & 1 & 1 & 1 & 2 & 1 & 3 & 2 & 1 & 1 & 1 & 1 & 2 & 1 & 3 & 1 & 2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}$$

按字典序排序并选取六种排列中最小的一种后，得到秩模式

$$\begin{array}{cccccccccccccccccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 3 & 3 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 3 & 3 & 2 & 2 & 2 & 2 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 3 & 3 & 1 & 1 & 2 & 2 & 2 & 2 & 1 & 1 \end{array}$$

等价方案具有相同的秩模式。事实上，项的重排、重新缩放和夹乘都不会改变矩阵秩，因此这些变换会保留三元组 ρ_l 的多重集。循环移位和转置会以相同的方式置换每个三元组的坐标（矩阵与其转置的秩相同），且二者生成全部六种排列，而选择最小列表的做法已将这些排列考虑在内。因此，秩模式不同的方案必然不等价。

反过来则不成立：不等价方案可能具有相同的秩模式。具体而言， [8] 的 17,376 个两两不等价方案仅实现了 387 种不同的秩模式。因此，秩模式相同是等价的必要条件，但不是充分条件。 [39], 提出了一种判定两个方案是否等价的算法，但本研究并未使用该算法：此处唯一用于断定两个方案不等价的依据，是它们具有不同的秩模式。第 5 节所述的数据集对每种秩模式恰好包含一个方案，因此其中所有方案两两不等价。换言之，该数据集为每种秩模式包含一个代表元，而不是为每个轨道包含一个代表元。

3.2 多项式不变量与方案的类型 heme

秩模式可以 方便地用一个多项式来编码。对于由上文定义的三元组 ρ_l 构成的方案，令 t

$$\operatorname{type}(x,y,z) = \sum_{l=1}^{r} x^{\operatorname{rk} u^{(l)}} y^{\operatorname{rk} v^{(l)}} z^{\operatorname{rk} w^{(l)}}, \tag{3}$$

这是 Sedoglavic 的目录 [21]中使用的不变量。重新排序、重新缩放和夹乘不会改变它，而循环移位和转置会排列变量 x、 y、 z。

Heule 等人 [8] 还引入了两个多项式不变量。它们所包含的信息少于秩模式，但更容易制表和比较：

$$f(x,y,z) = \sum_{\pi \in S_3} \pi \left(\sum_{l=1}^{r} x^{\operatorname{rk} u^{(l)}} y^{\operatorname{rk} v^{(l)}} z^{\operatorname{rk} w^{(l)}} \right), \tag{4}$$

$$g(\omega) = \omega^{\sum_{l=1}^{r} \operatorname{rk} u^{(l)}} + \omega^{\sum_{l=1}^{r} \operatorname{rk} v^{(l)}} + \omega^{\sum_{l=1}^{r} \operatorname{rk} w^{(l)}}, \tag{5}$$

其中，排列 π 用于排列变量 x、 y 和 z。这两个多项式在任何变换 (a)–(e) 下都不会改变。

Exam例 3.2。 对于 Laderman 方案，类型、多项式 f 和 g 如下所示 ms:

$$\operatorname{type}(x,y,z) = 4x^2y^2z^2 + 2x^3yz + 2xy^3z + 2xyz^3 + 13xyz.$$

$$f(x,y,z) = 24x^2y^2z^2 + 12x^3yz + 12xy^3z + 12xyz^3 + 78xyz,$$

and

$$g(\omega) = 3\omega^{31}.$$

多项式类型本身取决于三个因子的顺序，因此每个方案都会转换为排序形式：通过 (d) 和 (e) 对矩阵 U、 V、 W 进行排列，使三元组的排序列表 ρ_l 恰好等于秩模式，并按这些三元组递增的字典序对各项排序。相同三元组对应项的顺序，以及在产生相同列表的排列中如何选择，并未固定，因此排序形式并不唯一。与 [39], 的标准形式不同，它不会选出轨道的一个代表元，只是让秩模式能够直接从文件中读出。数据集中的所有方案都以这种形式存储。排序形式中方案的多项式类型与其秩模式一一对应，因此数据集中不同类型的数量等于不同秩模式的数量，而 f 和 g 的不同取值数量更少，因为这些不变量较为粗略。第 5 节使用这三个量，将该数据集与 Heule 等人和 Mårtensson 等人的集合进行比较。

4 搜索方法

该数据集中的方案由两类方法获得：数值方法 GradALS（第4.4节），它生成了数据集的一半以上，其中包括绝大多数有理数方案；以及翻转图搜索，该方法针对三元系数和有限域进行，并结合亨泽尔提升（第4.1–4.3节）。这两类方法找到的方案都与同一批已知方案进行比较（第4.2节）。

4.1 翻转图

部分方案是通过翻转图方法 [13] 得到的。在翻转图中，每个顶点都是一个方案（即张量 $T_{n,\,n,\,n}$ 分解为秩一项），边对应于保持正确性的变换。由于该图规模过大，无法穷尽式地遍历，因此采用随机游走进行搜索。使用了三种变换，每种变换都作用于一对项 $u^{(i)} \otimes v^{(i)} \otimes w^{(i)}$ 和 $u^{(j)} \otimes v^{(j)} \otimes w^{(j)}$。

定义 4.1（翻转）。若 $u^{(i)} = u^{(j)}$，则

$$u^{(i)} \otimes v^{(i)} \otimes w^{(i)} + u^{(j)} \otimes v^{(j)} \otimes w^{(j)} \;\to\; u^{(i)} \otimes (v^{(i)} + v^{(j)}) \otimes w^{(i)} + u^{(j)} \otimes v^{(j)} \otimes (w^{(j)} - w^{(i)}).$$

对 $v^{(i)} = v^{(j)}$ 和 $w^{(i)} = w^{(j)}$ 也定义了相同的变换，只需交换三个因子的角色即可。翻转既保持方案的正确性，也保持其秩。

定义 4.2 (约简)。若 $u^{(i)} = u^{(j)}$ 且 $v^{(i)} = v^{(j)}$，则

$$u^{(i)} \otimes v^{(i)} \otimes w^{(i)} \;+\; u^{(j)} \otimes v^{(j)} \otimes w^{(j)} \;\to\; u^{(i)} \otimes v^{(i)} \otimes (w^{(i)} + w^{(j)}).$$

其他任意一对因子中的等式也同样成立。一次约简会使秩降低一。若一次翻转产生含有零因子的项，秩也会降低，而该项会被直接删除。

定义 4.3 (加法)。若 $u^{(i)} \neq u^{(j)}$，$v^{(i)} \neq v^{(j)}$ 且 $w^{(i)} \neq w^{(j)}$，则

$$u^{(i)} \otimes v^{(i)} \otimes w^{(i)} + u^{(j)} \otimes v^{(j)} \otimes w^{(j)} \;\to\; u^{(i)} \otimes (v^{(i)} + v^{(j)}) \otimes w^{(i)} + u^{(j)} \otimes v^{(j)} \otimes (w^{(j)} - w^{(i)}) \;+\; (u^{(j)} - u^{(i)}) \otimes v^{(j)} \otimes w^{(i)}.$$

此变换使秩增加一，并保持正确性。

单独进行翻转绝不会改变秩；而不同项中没有相等因子的方案则没有可用的翻转。加号变换 [14] 有助于摆脱这类死胡同：它使秩增加一，并在因子之间产生新的重合，之后约简便可使秩降回去。由于翻转和约简都会产生仍然正确的方案，因此每个访问到的秩 23 方案都是 3×3 矩阵的有效方案。

4.2 搜索过程

方案池与数据集相同：其中为迄今通过任何方法找到的每种秩模式都恰好保存一个方案。翻转图搜索由多个独立运行器执行，每个运行器都从先前翻转图搜索得到的秩 23 方案初始池开始，重复以下步骤（算法1）[18, 19]:

1. 从方案池中随机选取一个方案。 2. 从该方案出发，在翻转图中开始随机游走。大多数步骤都是翻转，只要可以进行约简，就会应用约简。偶尔会以一定概率应用加号变换，使游走能够离开当前方案的邻域。 3. 定期对当前方案应用随机夹乘（第 3 节）。这不会改变秩模式，但会改变系数，并且通常有助于获得更稀疏的方案，也就是非零系数更少的方案。 4. 将每个访问到的秩 23 方案与方案池进行比较，具体如下所述。

具有新秩模式的方案会被加入候选池。如果遇到具有已知模式的方案，则在以下两种情况下，它会取代已存储的方案：其系数所属的类别更优，顺序为 $\mathbb{Z}_T \succ \mathbb{Z} \succ \mathbb{Q}$（第 2 节）；或者类别相同，但该方案的非零系数更少。类别优先于稀疏度，因此三元方案总会取代整数或有理数方案，即使后者更稀疏。最终

算法1： 用于构建数据集的翻转图搜索

输入： 已知秩 23 方案池，每种秩模式各一个
输出： 更新后的方案池 $\mathcal{D}$

```
当搜索未停止时执行
    S ← 来自 D的随机方案;
    重复
        对 S应用一次翻转，或以较小的概率进行一次加法;
        对 S应用所有可用的约化;
        如果该进行夹乘步骤那么
            对 S 应用一次随机夹逼;
        如果秩(S) = 23那么
            p ←的秩模式;
            如果 p不在 D 则将 S 加入 D;
            否则，如果 S 优于 D 中秩模式为 p 的方案则
                将该方案替换为 S;
    直到 达到遍历长度;
```

数据集中的每种秩模式，都对应搜索过程中找到的、系数类别最佳且非零系数数量最少的方案。

这条规则有两个需要牢记的后果。首先，目标并非让非零系数的数量尽可能少：使用这条规则是为了避免保留针对某种模式找到的第一个方案，因为它可能过于稠密，但完全可能存在具有相同秩模式的更稀疏方案。其次，数据集中的整数方案和有理数方案，是指尚未找到具有相同模式的三元方案的那些方案；这样的方案仍有可能存在。

4.3 系数集：三元搜索和亨泽尔提升

翻转图搜索主要在三元翻转图 [18], 中进行，其中方案的所有系数都属于 $\mathbb{Z}_T = \{-1, 0, 1\}$。不会执行产生该集合之外系数的变换，因此每个访问到的方案都是三元的。

此外，翻转图搜索在域 $\mathbb{F}_2$ 和 $\mathbb{F}_3$上进行。在 $\mathbb{F}_2$上，所有系数都是 0 或 1，符号也会消失；而在 $\mathbb{F}_3$ 上，值 1 和 -1 仍然彼此不同。在 $\mathbb{F}_p$上找到的方案（其中 $p \in \{2, 3\}$）通过亨泽尔提升提升到零特征：从 Brent 方程模 p的一个解出发，逐步将该解细化为模 p的更高次幂的解。对于 $p = 2$，最多执行三次提升；对于 $p = 3$，则只执行一次。最后一步之后，根据每个系数的剩余类将其重构为有理数（有理重构）；如果所得三元组 (U, V, W) 精确满足 Brent 方程，则接受该三元组。不允许在规定步数内完成此类重构的方案将被舍弃，因此只保留分子和分母较小的提升结果。

提升方案的类别取决于重构值。如果所有系数都属于 $\mathbb{Z}_T$，则该方案为三元方案。如果它们是整数，且其中一些位于 $\mathbb{Z}_T$ 之外（例如 2），则该方案为整数方案。否则，重构会产生分数，其分母不能被 p 整除，该方案则为有理数方案。

4.4 采用交替最小二乘法的梯度下降（GradALS）

数据集中超过一半的方案是通过一种称为 GradALS 的数值方法获得的，该方法将梯度下降与交替最小二乘（ALS）相结合。该方法通过 23 个秩一项之和来近似张量 $T_{3,\,3,\,3}$。通过梯度下降最小化近似误差，并利用 ALS 步骤加快收敛；在这些步骤中，固定另外两个矩阵后，交替求解矩阵 U、V 和 W 对应的线性最小二乘问题。若干正则化项会推动系数趋向集合 $\{0, \pm 1/2, \pm 1, \pm 2\}$。所得数值解会转换为精确方案并经过验证。解中可能出现该值集合的任意子集，因此 GradALS 会生成三元方案、整数方案（系数为 ± 2）以及分母为 2 的有理数方案；每个方案的类别由实际得到的值决定。数据集中绝大多数有理数方案都是通过这种方式获得的，而不是通过使用提升的 $\mathbb{F}_3$ 翻转图搜索获得的。该方法的详细说明将另行发表，其实现可在第 8 节所述的代码仓库中找到。

5 数据集

5.1 内容

该数据集包含 62029 个用于 3×3 矩阵乘法的秩 23 方案。每个方案都由矩阵 U、V、$W \in \mathbb{F}^{23\times 9}$ 给出，采用第 2 节中的编码。所有方案均以第 3.2 节所述的排序形式存储：对矩阵进行置换以匹配秩模式，并按照秩三元组的字典序递增排列矩阵的各行。

对于每种秩模式，数据集恰好包含一个方案，该方案按照第4.2节的规则选出：它具有最佳的系数类别；在该类别中，它是在搜索过程中发现的、具有此秩模式的方案里非零系数数量最少的方案。因此，数据集中的所有方案都两两不等价。同时，数据集并不完整：它既不包含秩 23 方案的所有轨道，也很可能没有涵盖所有可能的秩模式；而且，对于数据集中的每种秩模式，都可能存在多个不等价方案。
按照第 2 节的系数类别，21 585 个方案（34.80%）是三元方案，22 130 个（35.68%）是整数方案，18 314 个（29.52%）是有理数方案。整数方案和有理数方案是指未发现具有相同秩模式的三元方案的那些方案。

5.2 不变量的统计

表 1 展示了数据集中各方案所实现的第 3 节不变量的不同取值数量。由于每个方案都有其独有的秩模式，不同类型的数量与方案数量相同。多项式 f 和 g 是更粗略的不变量，因此其取值数量要少得多。各类别列中的数字分别在每个类别内统计；除了类型这一行外，它们的总和并不等于总数，因为不同类别的方案可能具有相同的 f 或 g 取值。

表 1：数量　　数据集中不变量的不同取值数量：总计及按系数类分别统计　　ss.

不变量	All	三元	整数	有理数
类型$(x,\ y,\ z)$	62 029	21 585	22 130	18 314
$f(x,\ y,\ z)$	7 930	3 421	5 072	4 774
$g(\omega)$	530	339	441	466

粗不变量呈现出与方案计数不同的图景。三个系数类别的规模相近（分别占数据集的 34.80%、35.68% 和 29.52%），但三元方案中 f 和 g 的不同取值最少：分别为 3 421 和 339；整数方案为 5 072 和 441，有理数方案为 4 774 和 466。这表明，数据集中非三元部分并未集中于少数几种秩模式，而是分布在范围广泛的秩模式中。

5.3 稀疏度

方案的稀疏度以 U、V 和 W 中非零系数的总数衡量。图 1 展示了该数值在整个数据集中的分布。

该分布呈单峰且右偏：大多数方案集中在峰值附近的狭窄区间内；从 139 开始的稀疏端样本较少，而密集端则延伸出长尾，最高达到 278。因此，接近最小值的方案十分罕见。

系数类别的差异主要体现在分布范围，而非位置。三元方案覆盖了整个范围，并且只有它们出现在最稀疏的部分：非零系数少于 149 个的方案全部是三元方案。整数方案和有理数方案集中在较窄的区间内，峰值几乎相同：三元、整数和有理数方案的众数分别为 175、174 和 178。

这些分布描述的是数据集，而非所有秩 23 方案。根据第4.2节的选择规则，只有在某个秩模式不存在三元方案时，才会保留更一般类别的方案；无论非零系数的数量多少，都会优先选择三元方案，因此数据集之外可能存在具有相同秩模式、但更稀疏的方案。第 5.6 节中描述的含有 138 个非零系数的方案就是一例：它是有理数的，而数据集中表示其秩模式的三元方案含有更多非零系数。

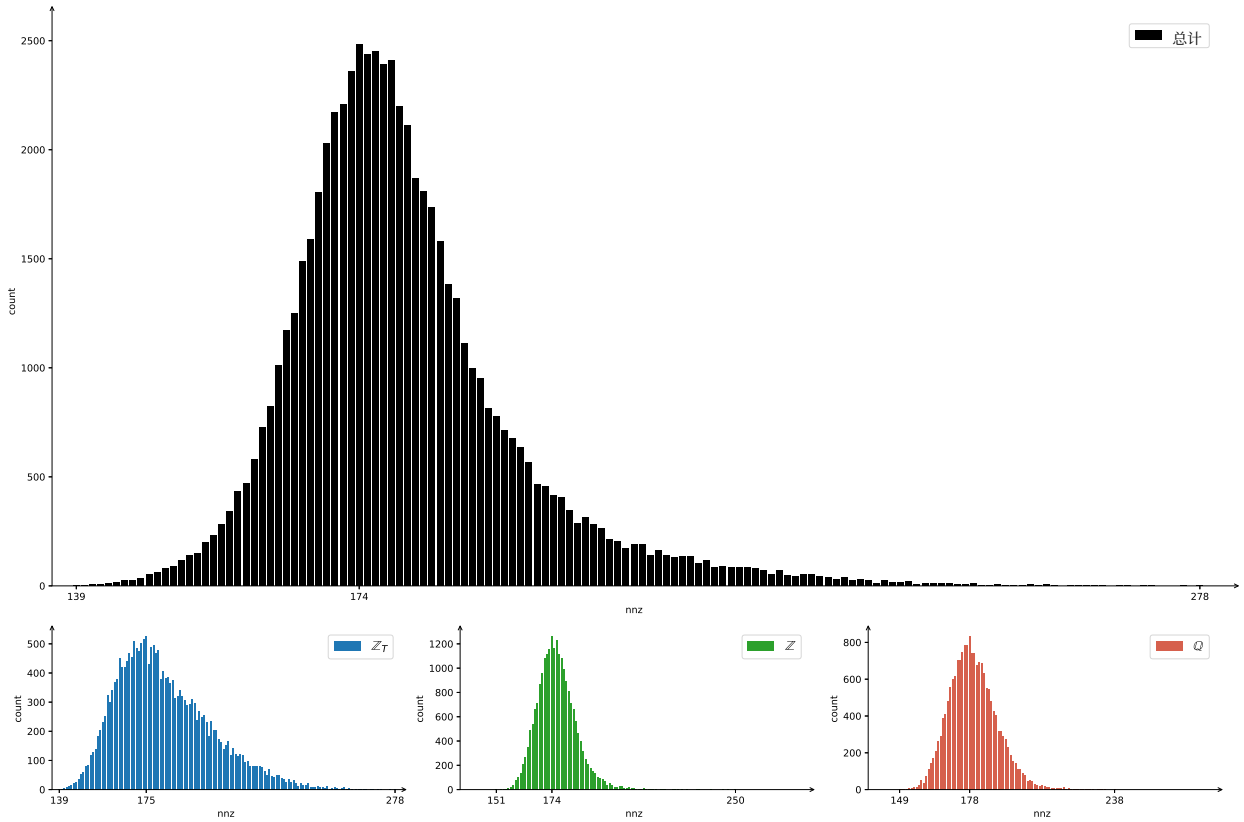

图 1：数据集中所有方案（上图）以及三元（$\mathbb{Z}_T$）、整数（$\mathbb{Z}$）和有理数（$\mathbb{Q}$）方案（下图）中非零系数数量的分布。横轴刻度标出每种分布的最小值、众数和最大值；各面板的纵轴刻度范围不同。

5.4 秩三元组

一个方案的秩模式是其 23 个项对应的三元组 $\rho_l = (\operatorname{rk} u^{(l)},\ \operatorname{rk} v^{(l)},\ \operatorname{rk} w^{(l)})$ 所组成的多重集，因此，一个方案可以看作是将其各项分配到 27 种可能的三元组中的一种方式。图 2 展示了数据集中的方案如何进行这种分配：对于每个三元组，它给出了属于该类型的项数分布。

这幅图中的几个特征解释了为何秩模式如此繁多。低秩因子的项占主导地位，而三元组 (1，1，1)、(1，1，2) 和 (2，2，2) 几乎存在于每个方案中，因此构成了秩 23 方案的共同核心。差异在于每个方案包含各类型项的数量：对于 (1，1，1)，这一数量变化范围很广；对于大多数其他常见三元组，这一数量也有多种取值。因此，秩模式是若干几乎相互独立变化的计数的组合，而这种组合自由度解释了大部分模式。另一方面，至少含有两个秩为 3 的因子的三元组则很少见：它们只出现在少数方案中，且在一个方案中最多出现两次。

这些面板对于三元组的排列并不对称，例如 (1，1，2) 比 (2，1，1) 常见得多。这是排序形式（第 3.2 节）导致的：每个方案中因子的顺序都经过选择，使其三元组列表达到字典序最小，这会优先排列首项较小的三元组。因此，该图展示的是秩模式的定义，而不是各个因子的秩。

5.5 与现有集合的比较

表 2 将该数据集与两个现有的秩 23 方案集合进行了比较：Heule 等人 [8], 的集合在性质上与其最为接近，而 Mårtensson 等人 [9]的集合则是另一个。第一个集合包含 17,376 个具有三元系数的两两不等价方案，它们仅实现了 387 种不同的秩模式和 303 个不同的 f 取值。第二个集合包含 15,316 个具有三元系数的方案，实现了 568 种秩模式和 215 个 f 取值。它们的

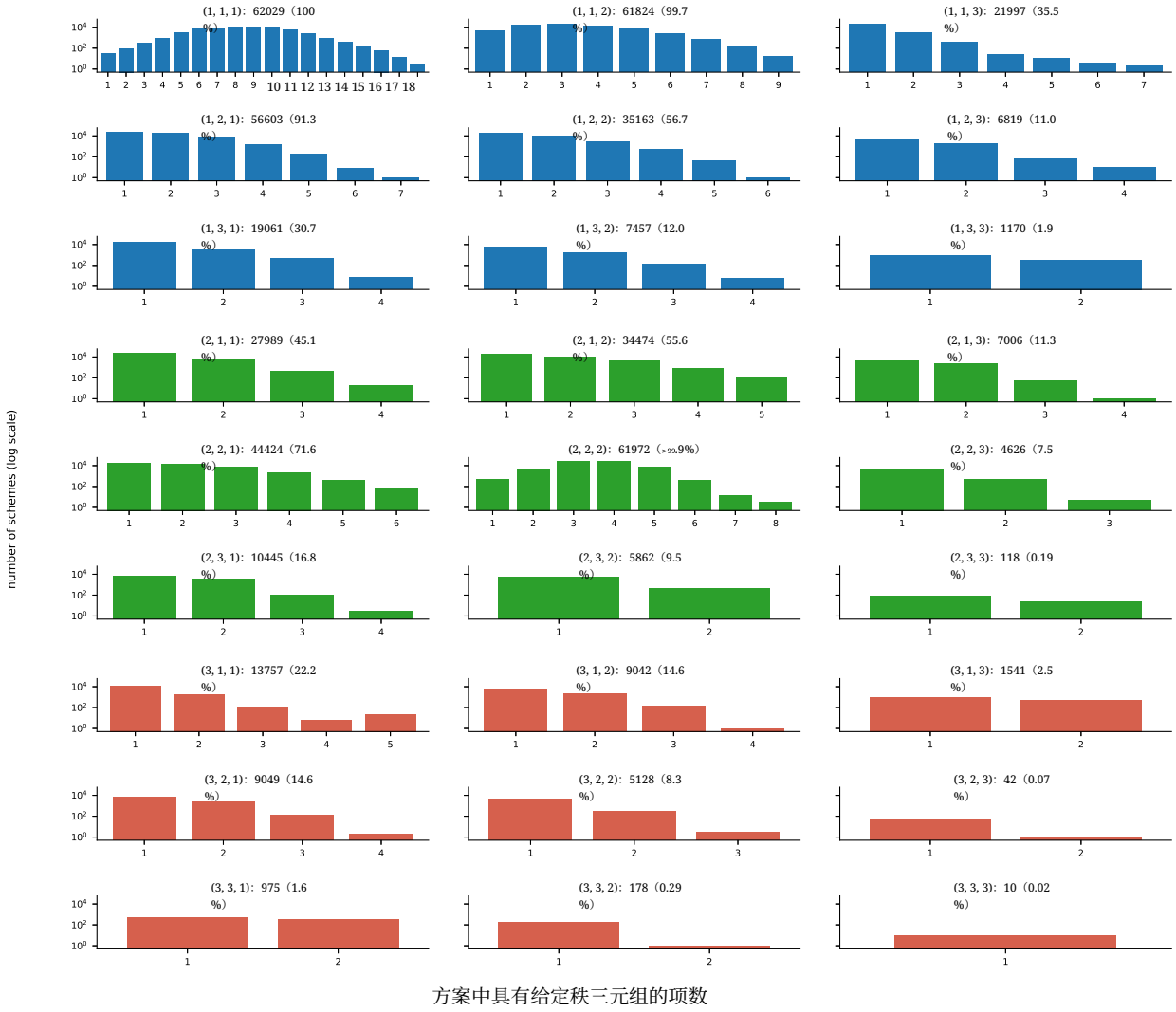

图 2：方案中具有给定秩三元组（$\mathrm{rk}\, u^{(l)}$、$\mathrm{rk}\, v^{(l)}$、$\mathrm{rk}\, w^{(l)}$）的项数分布，共有 27 个三元组。横轴表示方案中此类项的数量，纵轴（对数刻度）表示数据集中方案的数量。面板标题中的百分比表示至少包含一次该三元组的方案所占比例；未显示不含该三元组的方案。颜色对应第一个因子的秩。

两两不等价性尚未得到确立。这两个集合共有 184 种秩模式。相比之下，该数据集包含 62,029 个秩模式两两不同的方案，以及 f 的 7,930 个不同取值。

由于两个集合都由三元方案组成，表格还单独展示了数据集中的三元部分：仅这一部分就实现了 21585 种秩模式，超过了任一集合中的方案数量。

这两个集合中的方案在搜索过程中均未使用，但这两个集合的全部 771 种秩模式都出现在数据集中。数据集中的其余方案均与 [8, 9] 中的任何方案不等价。加上 Heule 等人的集合（其中的方案两两不等价），这些方案至少给出了 78 634 个两两不等价的秩 23 方案。 [9] 中的方案未计入这一数量，因为尚未确立它们两两不等价。对于与这些集合共享的 771 种模式，我们不作任何断言：数据集中的方案可能与某个集合中具有相同模式的方案等价，也可能不等价，因为并未直接检验等价性。

因此，数据集与现有集合相辅相成。Heule 等人的集合表明，同一种秩模式可能包含许多不等价方案，而该数据集则表明，不同秩模式的数量远远超过以往的发现。

表 2：与现有秩-23 3×3 方案集合的比较。最后两列分别展示数据集的三元部分和整个数据集。

	Heule 等人	Mårtensson 等人	此数据集（三元）	此数据集（完整）
方案	17 376	15 316	21 585	62 029
两两不等价	yes	未知	yes	yes
不同的秩模式	387	568	21 585	62 029
$f(x,\ y,\ z)$的不同取值	303	215	3 421	7 930
$g(\omega)$的不同取值	91	83	339	530
非零系数（范围）	139-186	139-200	139-278	139-278

5.6 最稀疏的方案

目前发现的最稀疏方案包含 138 个非零系数，由 GradALS 于 2026年6月 得出（第4节.4）。据作者所知，在此日期之前，无论是在 $[8, 9]$ 的集合中，还是在其他作者早期搜索的结果中，都没有已知方案包含少于 139 个非零系数。该方案并未作为独立作品发表，而仅通过电子邮件发送给了几位研究人员。2026年9月，KnotTheory.ai 博客上的一篇文章 [34] 介绍了一种包含 138 个非零系数的方案，该方案通过对 $\mathbb{F}_3$ 进行支持集上限局部搜索，再提升至有理数域而得到。该方案使用范围更广的系数集合 $\{0\} \cup \{\pm 2^k :\ -4 \le k \le 4\}$，并包含 30 个非整数系数，而本文介绍的方案仅包含 12 个。两种方案的秩模式相同，但尚未验证它们是否精确等价。

此处描述的方案使用系数 $\{0, \pm 1/2, \pm 1, \pm 2\}$，因此属于有理数类。它不属于该数据集：其秩模式与一个含有 146 个非零系数的三元方案相同；根据第4.2节的选择规则，该三元方案取代了它。因此，此方案单独列于附录 A。

5.7 格式与验证

这些方案提供三种格式：JSON、纯文本（.txt）和 Maple（.m）。文件名为 3x3x3_m23_<hash>_<class>.<ext>，其中 <hash> 是该方案的哈希值，<class> 是系数类别：ZT（三元）、Z（整数）或 Q（有理数）。JSON 文件存储方案的维数和秩（dimension、rank）、非零系数的数量（nnz）、矩阵 U、V、W（以行列表 (u, v, w) 的形式表示），其中有理数系数以字符串形式书写（“-1/2”），以及不变量 type、f 和 g。文本文件包含标题行 3 3 3 23，后接三行展平矩阵 U、V和 W。在有理数方案的文本文件中，每个系数均写成一对数字，即分子和分母，因此 -1 2 表示 $-1/2$，而 1 1 表示 1。Maple 文件将每一项的三个矩阵 $u^{(l)}$、$v^{(l)}$、$w^{(l)}$ 存储为嵌套列表，其中有理数系数写作 -1/2。在所有格式中，W 的系数均采用第 2 节所述的转置编码存储。

每个方案的正确性都通过在整数和有理数上使用精确算术验证 Brent 方程 (2) 来检查。验证脚本包含在该数据集的代码仓库中。相同的检查、从三种格式加载方案、计算不变量以及其他实用工具，都由主研究代码仓库 [22] 中的 Scheme 类实现。

6 讨论

6.1 秩 23 方案的实际价值

秩 23 大大高于与 Strassen 算法竞争所需的秩。将其递归应用于分块时，对于 3×3 矩阵，秩为 r 的方案给出的指数为 $\log_3 r$；只有当 $r < 3^{\log_2 7} \approx 21.85$（即 $r \le 21$）时，该指数才小于 Strassen 算法的 $\log_2 7 \approx 2.807$。秩 23 给出 $\log_3 23 \approx 2.854$，因此，该数据集中的方案都无法在递归设置下改进 Strassen 算法；在实践中，在速度至关重要的应用中，它们也不太可能取代 Strassen 算法。该数据集的价值在于其他方面：它展示了在已知最小秩下存在多少种结构上不同的方案，并为搜索方案的其他优化属性提供了一个规模大且多样化的初始候选集。

这些性质中最重要的是加法次数，其纪录值列于引言中。非零系数的数量只能粗略反映这一指标，因为它没有考虑中间和在线性形式之间的复用

，但计算成本低，可作为自然的初步筛选标准。未来工作的一个自然方向是对整个数据集而非少数几个方案进行加法优化。

6.2 下界与秩 23 的现状

近期的下界结果改变了秩 23 的现状。$\mathbb{F}_2$ 上的 3×3 矩阵乘法的秩恰为 23 [6]，而根据模 2 约化，这意味着对于具有整数系数的方案，23 是最优的。因此，数据集中的三元方案和整数方案在秩方面都是最优的，该数据集可视为不同最优方案的集合。但这不适用于分母为偶数的有理数方案，例如数据集中分母为 2 的方案，因为它们无法进行模 2 约化。在特征为零的域上，秩目前仅知介于 21 [5] 和 23 之间，因此，无法排除 $\mathbb{Q}$或 $\mathbb{C}$ 上存在秩为 22 或更低的方案。

6.3 局限性

该数据集存在若干局限性，这些局限性源于其构建方式以及区分其中方案的方法。

- 完备性与等价性。 该数据集包含搜索所发现的每种秩模式各一个方案。它既不是秩模式的完整列表，也不是轨道的完整列表。根据秩模式，可以保证其中各方案彼此不等价；但并未直接检验等价性，例如没有使用 [39], 的算法。因此，该数据集只能给出轨道数量的下界，也不对早期集合中与数据集某个方案具有相同秩模式的方案作任何判断。
- 选择规则。 系数类别优先于稀疏度（第4.2节）。因此，整数或有理数方案所表示的秩模式也可能存在三元方案；具有相同秩模式的方案也可能更加稀疏；第 5.6 节中的方案就是一个例子。

6.4 未来方向

由上述内容可以自然地引出若干后续方向。该数据集可用作方案来源，以优化加法、研究秩 23 方案的稀疏度与结构，以及考察其他取决于方案本身而非秩的性质，例如数值稳定性。同样的方法——寻找不变量两两不同的方案——也可应用于其他小型格式。最后，搜索本身还可以继续：数据集不断扩充，每种新的秩模式都会提高不等价方案数量的下界。若采用比秩模式更精细的等价性检验，例如对具有相同秩模式的方案应用 [39], 的算法，就能将轨道数量的下界转化为更精确的计数，并使数据集与现有集合能够逐个方案进行比较。

7 结论

本文展示了一个包含 62 029 个用于计算 3×3 矩阵乘法的秩 23 方案的数据集。每个方案都有其独特的秩模式，因此所有方案两两不等价。Heule 等人 [8] 和 Mårtensson 等人 [9] 的集合中的全部 771 种秩模式都包含在该数据集中，其他所有方案也都与这些集合中的任何方案不等价。结合 Heule 等人的集合，该数据集给出了至少 78 634 个两两不等价的秩 23 方案。因此，同一种秩模式可能包含许多不等价方案；正如本文所示，不同秩模式的数量远超以往所知。
此外，还发现了一个含有 138 个非零系数的方案，少于截至 2026年6月已知的任何公开可用集合中的方案。该方案见附录 A。随后，一篇博客文章 [34]中刊载了一个秩模式和非零系数数量均相同的方案。

由于在 $\mathbb{F}_2$ 上且对于整数系数方案而言，秩 23 是最优的，因此该数据集可视为大量不同最优方案的集合，可作为优化其他属性（首先是加法次数）的初始候选池。

8 可用性

与本研究相关的所有数据和代码均可在以下代码仓库中公开获取。

- 数据集：https://github.com/dronperminov/rank23_3x3_dataset。方案以 JSON、纯文本和 Maple 格式提供，并附带一个用于验证每个方案正确性的脚本。

•快速矩阵乘法方案的主要研究代码仓库 [22]：https://github.com/dronperminov/FastMatrixMultiplication。它包含 Scheme 类，可从数据集的所有格式中加载方案、使用精确算术验证 Brent 方程并计算第 3 节中的不变量，还提供方案目录和其他实用工具。

• 翻转图工具 [18, 19]：https://github.com/dronperminov/ternary_flip_graph。它实现了第4节中的翻转图搜索，包括对三元系数的搜索，以及结合亨泽尔提升在有限域上的搜索。

• *GradALS* 方法：https://github.com/dronperminov/numerical-fast-matmul。它实现了一种数值搜索，将梯度下降与交替最小二乘相结合，并使用多个正则化项来获得系数较小的精确方案（第4.4节）。

致谢

作者感谢 Kirill Khoruzhii、Isaac Wood、Erik Mårtensson、Paul Stankovski Wagner、Joshua Stapleton 和 AlexandreSedoglavic，感谢他们进行的有益交流与讨论。

参考文献

[1] Volker Strassen。高斯消元并非最优。《Numerische Mathematik》，13(4):354–356，1969。[2] Shmuel Winograd。关于矩阵乘法。《Linear Algebra and Its Applications》，4:381–388，1971。网址：https://api.semanticscholar.org/CorpusID:123566199。[3] John E Hopcroft 和 Leslie R Kerr。关于最小化矩阵乘法所需的乘法次数。《SIAM 应用数学杂志》，20(1):30–36，1971。[4] Julian D Laderman。一种使用23次乘法计算矩阵乘法的非交换算法。《美国数学会通报》，1976。[5] Andrey Ivanov。每个域上的矩阵乘法秩下界为21，2026年10月。网址：https://github.com/AIV5/mmrank21。[6] Tejasvi Singh Tomar。整数环上的矩阵乘法秩为23。arXiv预印本 arXiv:2610.11823，2026。[7] Rodney W Johnson 和 Aileen M McLoughlin。矩阵乘法的非交换双线性算法。《SIAM 计算杂志》，15(2):595–603，1986。[8] Marijn JH Heule、Manuel Kauers 和 Martina Seidl。乘法计算矩阵的新方法。《符号计算杂志》，104:899–916，2021。[9] Erik Mårtensson、Paul Stankovski Wagner 和 Joshua Stapleton。地狱钟声：用于三元快速矩阵乘法算法的神经网络流程。密码学电子印本档案，2026。网址：https://eprint.iacr.org/2026/1688。[10] Paul Stankovski Wagner。fmm_加法_约简，2025年5月。网址：https://github.com/werekorren/fmm_加法_约简。[11] Alhussein Fawzi、Matej Balog、Aja Huang、Thomas Hubert、Bernardino Romera-Paredes、Mohammadamin Barekatain、Alexander Novikov、Francisco J R. Ruiz、Julian Schrittwieser、Grzegorz Swirszcz 等。利用强化学习发现更快的矩阵乘法算法。《自然》，610(7930):47–53，2022。[12] Arnaud Deza、Chang Liu、Pashootan Vaezipoor 和 Elias B Khalil。轻松实现快速矩阵乘法：一种约束编程方法。arXiv预印本 arXiv:2306.01097，2023。[13] Manuel Kauers 和 Jakob Moosbauer。矩阵乘法的翻转图。载于《2023年国际符号与代数计算研讨会论文集》，第381–388页，2023。[14] Yamato Arai、Yuma Ic hikawa 和 Koji Hukushima。矩阵乘法的自适应翻转图算法。载于《2024年国际符号与代数计算研讨会论文集》，第292–298页，2024。[15] Jakob Moosbauer 和 Michael Poole。具有对称性的翻转图与新的矩阵乘法方案。载于《2025年国际符号与代数计算研讨会论文集》，第233–239页，2025。[16] Manuel Kauers 和 Isaac Wood。探索矩阵乘法的元翻转图。arXiv预印本 arXiv:2510.19787，2025。

[17] Kirill Khoruzhii、Patrick Gelß 和 Sebastian Pokutta。通过翻转图搜索实现更快的结构化矩阵乘法算法。载于《2026年国际符号与代数计算研讨会论文集》，第227–236页，2026。[18] Andrew Perminov。通过三元元翻转图实现快速矩阵乘法。arXiv预印本 arXiv:2511.20317，2025。[19] Andrew Perminov。小型格式下的快速矩阵乘法：利用开源翻转图框架发现新方案。arXiv预印本 arXiv:2603.02398，2026。[20] Alexander Novikov、Ngân V˜u、Marvin Eisenberger、Emilien Dupont、Po-Sen Huang、Adam Zsolt Wagner、Sergey Shirobokov、Borislav Kozlovskii、Francisco JR Ruiz、Abbas Mehrabian 等。Alphaevolve：用于科学与算法发现的编码代理。arXiv预印本 arXiv:2506.13131，2025。[21] Alexandre Sedoglavic。又一份快速矩阵乘法算法目录，2019。网址：http://fmm.univ-lille.fr，访问时间：2026年11月。[22] Perminov, Andrew。快速矩阵乘法研究代码仓库。https://github.com/dronperminov/FastMatrixMultiplication，2026。访问时间：2026年11月5日。[23] Benoit Chatain Lacelle。采用前沿闭包搜索的快速矩阵乘法算法目录。arXiv预印本 arXiv:2606.13408，2026。[24] Erik Mårtensson 和 Paul Stankovski Wagner。野兽之数：将维数不超过666的快速矩阵乘法算法中的加法次数降下来。载于《2025年应用与计算离散算法会议论文集》（ACDA），第47–60页。SIAM，2025。[25] Joshua Stapleton。用于精确矩阵乘法的60次加法、秩23方案。arXiv预印本 arXiv:2508.03857，2025。[26] Erik Mårtensson、Paul Stankovski Wagner 和 Joshua Stapleton。一种算术复杂度为59的秩23算法，用于矩阵乘法。arXiv预印本 arXiv:2601.05272，2025。[27] Andrew Perminov。一般3×3矩阵乘法的58次加法、秩23方案。arXiv预印本 arXiv:2512.21980，2025。[28] Yinqi Sun。一般矩阵乘法的精确56次加法、秩23方案。arXiv预印本 arXiv:2604.27645，2026。[29] Samurdhi Karunaratne 和 Anushka Idamekorala。秩23的矩阵乘法只需55次加法。arXiv预印本 arXiv:2607.28676，2026。[30] Greg Sidebottom。通过精确双侧最小化实现的55次加法、秩23矩阵乘法方案。arXiv预印本 arXiv:2610.27451，2026。[31] Hugo Nielsen。用于精确3×3矩阵乘法的52次加法、秩23方案。arXiv预印本 arXiv:2609.06588，2026。[32] Joshua Stapleton 和 Andrew Perminov。用于秩23矩阵乘法的51次加法替代基核。arXiv预印本 arXiv:2609.32388，2026。[33] Marijn J. Heule、Manuel Kauers 和 Martina Seidl。矩阵乘法代码库。http://www.algebra.uni-linz.ac.at/research/matrix-multiplication/，2019。[34] Chandragupt Sharma。追逐零：一个支集138的秩23矩阵乘法方案。KnotTheory.ai研究札记（博客文章），2026年9月。网址：https://knottheory.ai/blogs/matrix-multiplication-3x3-support-138。发布于2026年9月30日。访问时间：2026年11月5日。[35] Richard P Brent。矩阵乘法算法。技术报告，斯坦福大学，美国加利福尼亚州斯坦福，1970。[36] HF de Groot。关于双线性映射计算最优算法的簇。《理论计算机科学》，7(2):127–148，1978。[37] John Hopcroft 和 Jean Musinski。对偶性在矩阵乘法及其他双线性型复杂度中的应用。载于第五届年度 ACM计算理论研讨会论文集，第73–87页，1973。[38] Vladimir P Burichenko。关于Strassen算法的对称性。arXiv预印本 arXiv:1408.6273，2014。[39] Manuel Kauers 和 Jakob Moosbauer。矩阵乘法方案的标准形式。载于《国际代数信息学会议》，第149–160页。Springer，2022。

一个含有 138 个非零系数的秩 23 方案

第 5.6 节的方案采用第 2 节中的编码如下给出。矩阵 W 以转置形式打印为$W^T \in \mathbb{Q}^{9\times 23}$，因此其第 l 列是向量 $w^{(l)}$。各项按搜索产生的顺序列出，而非采用第 3.2 节的排序形式。它的 nnz(U) 为 $= 47$，nnz(V) 为 $= 47$，nnz(W) 为 $= 44$。Brent 方程 (2) 已通过精确有理数算术验证。

$$U = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 2 \\
1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & -\frac{1}{2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -\frac{1}{2} & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 2 \\
0 & 1 & 0 & 1 & 0 & 0 & -2 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{2} & 0 & 1 & 0 & 0 & -1 \\
0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0
\end{bmatrix}
\quad
V = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 \\
2 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \\
0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -\frac{1}{2} \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & -\frac{1}{2} & 0 \\
0 & 1 & 0 & 2 & 0 & 0 & 0 & -1 & 0 \\
2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & \frac{1}{2} & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & \frac{1}{2} & 0 \\
0 & \frac{1}{2} & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & 0
\end{bmatrix}$$

$$W^T = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -1 & 0 & 1 \\
-1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

不变量。 该方案的第 3 节不变量为

- 类型$(x, y, z) = 4x^2y^2z^2 + x^2yz + xy^2z + 2xyz^2 + 15xyz$;
- $f(x,\ y,\ z) = 24x^2y^2z^2 + 8x^2yz + 8xy^2z + 8xyz^2 + 90xyz$;
- $g(\omega) = \omega^{29} + 2\omega^{28}$。

按照第 3.1 节的秩模式，该方案包含 15 项，其中三个矩阵 $u^{(l)}$、$v^{(l)}$、$w^{(l)}$ 的秩均为 1；四项中三个矩阵的秩均为 2；另有四项中恰有一个矩阵的秩为 2。