arXiv:2512.20325v1 [cs.CG] 23 Dec 2025

Top-K 外幂持久同调：算法、结构与稳定性

Yoshihiro Maruyama

日本名古屋大学信息学部
maruyama@i.nagoya-u.ac.jp

摘要。 外幂在计算几何中的持久同调研究里发挥着重要作用。本文研究如何直接从温和持久模的条形码中提取外幂层中 K 个最长区间，而无需枚举整个 B。我们证明了一个结构分解定理，将 B组织成按锚点划分的单调流，并明确给出重数，从而能够采用最佳优先算法。对于任意固定的 $i \geq 2$，我们通过分组最佳优先搜索给出一种$O\big((M+K)\log\ M\big)$ 时间算法。我们还证明，Top-K 长度向量在输入条形码的瓶颈扰动下是 2-Lipschitz 连续的，并证明一个比较模型下界，表明 $O(M \log M)$预处理在信息论意义上不可避免。实验结果验证了理论分析，并显示在高重叠情形下，该方法比完全枚举更快。通过高效提取最显著的特征，我们的方法使大型数据集上的高阶持久性分析成为可能，因此可广泛应用于机器学习、数据科学和科学计算。

关键词： 计算几何 · 持久同调 · 稳定性

1 引言

持久同调中的外幂 Λ^i 能够捕捉拓扑特征之间的高阶相互作用，这是普通持久性无法记录的 [14, 9, 25].[1]标准持久性追踪单个圈的生命周期，而外幂 Λ^i 则编码圈群如何共存，从而提供更丰富、稳定且可计算的不变量 [7, 4, 5]。这些高阶特征在理论拓扑学到机器学习等领域的应用中日益重要，因为简洁且具有区分性的摘要至关重要。

然而，快速计算完整的外幂条形码很快就会变得难以实现：即使对于 Λ^2，其大小也可能与输入规模呈二次关系。因此，枚举所有区间的直接算法会造成计算浪费，尤其是在只需要最显著特征时。这促使我们提出$Top\text{-}K$ 问题：无需枚举整个结构，只提取 $B(\Lambda^i M)$ 中最长的 K 个区间。这样的

[1] Λ 表示外积/外幂。 Λ 是在持久模上逐点定义的（即在定义域区间的每个元素处取外积；更多详情见下文）。

视角与实际需求相符：用户需要简洁的可视化摘要、稳健的统计量，或供下游学习使用的定长特征。

本文表明，针对外幂的 Top-K 存在一种高效且稳定的解法，将经典持久同调算法 [10, 26, 1] 与选择算法 [11, 13] 和持久数据结构 [8] 中的技术相结合。这使高阶持久性更具可扩展性，并能广泛应用于计算几何和拓扑数据分析。

本文的主要贡献如下：（i）我们将 $B(\Lambda^i M)$ 结构分解为简单的单调流，明确说明高阶区间的来源及其重数的产生方式；（ii）我们设计了一种最佳优先算法，能够以近似输出敏感的时间提取精确的 Top-K 区间，避免完全枚举的成本；（iii）我们证明，Top-K 长度向量在瓶颈扰动下保持稳定，从而提供一种简洁且抗噪的摘要；（iv）我们在比较模型下确立了一个 $\Omega(M \log M)$ 下界，证明我们的预处理成本在常数因子范围内是最优的。

在本文余下部分，我们首先建立数学基础，并给出带有复杂度结果的结构定理和算法。随后，我们证明稳定性和最优性结果。最后，我们给出实验验证。

2 基本概念和基本定理

在全文中，令 $i \geq 2$ 为一个固定常数。

令 M 为定义在 $I \subset \mathbb{R}$ 上的良态、逐点有限维持久模，其条形码为 $B(M) = \{ J_r = [b_r, d_r) \}_{r=1}^{M}$；为方便记号起见，我们对该持久模及其区间数量均使用同一符号 M（有关持久模的基础知识，参见 [25, 5]）。按照惯例，我们将 M 在 $t \in I$ 处的取值写作 M_t。$\Lambda^i M$ 按逐点方式定义为 $(\Lambda^i M)_t = \Lambda^i M_t$。[2]

除非另有说明，我们假设所有条均为有限条（$d_r < \infty$）；这涵盖有限复形/图上的常见滤过。在某些滤过中（例如 H_0 的 Vietoris–Rips 或 ˇCech 复形），部分条会延伸至 $+\infty$。如果按照以下任一种标准方式处理这些条，我们的算法和分解均可保持不变：(i) 截断：固定一个全局时间范围 $t_{\max}$，并将每个无限条 $[b_r, \infty)$ 替换为 $[b_r, t_{\max})$，从而使 Λ^i 区间遵守有限范围；(ii)相对表述：将无限条视为持续到形式符号 ∞，并注意到在外幂区间演算（下文定理 1）中只会出现 $\min\{d_r, \dots\}$，因此截断到任意足够大的有限截止值都会得到相同的 Top-K 结果。因此，不失一般性，我们可以假设所有条均为有限条。

我们采用左闭右开约定，并处理一个包含所有b_r 和 d_r 、按时间排序的全局事件列表；若时间并列，则先处理死亡，再处理出生 [9, 5].[3]

[2] 注意，右侧的 $\Lambda^i M_t$ 表示向量空间 M_t 的 i 次外幂。对于态射，对于 I 中的每个 $s \leq t$，我们定义 $(\Lambda^i M)_{s \leq t} = \Lambda^i(M_{s \leq t}) : \Lambda^i(M_s) \longrightarrow \Lambda^i(M_t)$。有关持久同调的基础知识，另请参阅 [9, 25].[3] 在持久同调中，条形码的 $birth$ 是同调类首次出现时的参数值，其 $death$ 则是该类消失时的参数值 [9, 5]。

对于同一时刻发生的出生，我们固定任意一个在整个扫描过程中保持一致的全序（参见 [9, 10]）。对于从 $-\infty$ 到 $+\infty$的事件扫描，在条形码 r 于时刻 b_r出生之前，定义存活集合 $A_r := \{\, s : [b_s, d_s) \text{ 在 } b_r \,\}$之前存活，且 $c_r := |A_r|$. 按死亡时间非递增排序；将它们记为 $d_r(1) \geq d_r(2) \geq \cdots \geq d_r(c_r)$。对于 $j \in \{1, \ldots, c_r\}$ 集合

$$\ell_r(j) \;:=\; \max\Big\{\, 0,\; \min\{d_r,\; d_r(j)\} - b_r \Big\}, \tag{1}$$

因此 $j \mapsto \ell_r(j)$ 是非递增的。

2.1 条形码层面的外幂

我们首先证明一个基本定理，以阐明外幂的条形码结构。

定理 1（外幂区间演算）。对于任意 $i \geq 1$, $\Lambda^i M$ 的条形码是多重集

$$B(\Lambda^i M) \;=\; \Big\{\, \big[\max_j b_{\ell_j},\, \min_j d_{\ell_j}\big) \;:\; \ell_1 < \cdots < \ell_i,\; \max_j b_{\ell_j} < \min_j d_{\ell_j} \Big\}.$$

证明。根据温和逐点有限维模的条形码分解（见 [25, 5]），存在一个（非典范）同构 $M \;\cong\; \bigoplus_{r=1}^{M} I_{[b_r, d_r)}$，其中 $I_{[b_r,d_r)}$ 是支撑在 $[b_r, d_r)$ 上的（一维）区间模。

固定 $i \geq 1$。在 $t \in \mathbb{R}$ 中逐点应用外幂函子。对于任意有限族向量空间，$\Lambda^i\Big(\bigoplus_{r=1}^{M} V_r\Big) \;\cong\; \bigoplus_{\substack{\ell_1<\cdots<\ell_i \\ \ell_j \in \{1,\ldots,M\}}} V_{\ell_1} \wedge \cdots \wedge V_{\ell_i}$关于 V_r 是自然的。在时间 t 处取值，并令 $V_r = (I_{[b_r,d_r)})_t$，得到

$$\Lambda^i(M_t) \;\cong\; \bigoplus_{\ell_1<\cdots<\ell_i} (I_{[b_{\ell_1},d_{\ell_1})})_t \wedge \cdots \wedge (I_{[b_{\ell_i},d_{\ell_i})})_t.$$

由于每个 $(I_{[b_\ell,d_\ell)})_t$ 要么为 0（若 $t \notin [b_\ell, d_\ell)$），要么是 1 维向量空间（若 $t \in [b_\ell, d_\ell)$），因此，当且仅当 t 位于交集$\bigcap_{j=1}^{i}[b_{\ell_j}, d_{\ell_j})$中时，$\{\ell_1, \ldots, \ell_i\}$ 对应的直和项为 k；否则为 0。因此，随着 t 变化，由该直和项生成的子函子是支撑在 $\bigcap_{j=1}^{i}[b_{\ell_j}, d_{\ell_j})$ $=[\max_j b_{\ell_j}, \min_j d_{\ell_j})$ 上的区间模，当且仅当 $\max_j b_{\ell_j} < \min_j d_{\ell_j}$ 时非零。

上述同构相对于M 的结构映射具有自然性，这表明这些逐点分解可组装成持久模的同构 $\Lambda^i M \;\cong\; \bigoplus_{\ell_1<\cdots<\ell_i} I_{[\text{最大值}_j\, b_{\ell_j},\ \text{最小值}_j\, d_{\ell_j})}$，其中约定空交集贡献零模，因此不产生任何条形码。因此，$\Lambda^i M$ 的条形码恰好就是所述的区间多重集。

2.2 带重数的 Top-K

将 $B(\Lambda^i M)$ 的长度多重集（按重数计数）按非递增顺序排列为 $L_1 \geq L_2 \geq \cdots$。对于 $K \geq 1$, *Top-K 多重集* 为$\{L_1, \ldots, L_K\}$（按重数计），而 *Top-K 长度向量* 为

$$\mathbf{L}_K(M, i) \;:=\; (L_1, \ldots, L_K) \in \mathbb{R}^K_{\geq 0},$$

必要时用零补齐。[4]

对于 $j \geq i-1$，定义二项式权重 $w_i(j) := \binom{j-1}{i-2}$. 在第3节中，我们证明：对于固定的锚点 r，所有最大所选秩等于 j 的 Λ^i 区间都具有相同的长度 $\ell_r(j)$ 和总重数 $w_i(j)$；此外，对所有锚点取并集可得到完整的多重集 $B(\Lambda^i M)$。

2.3 瓶颈距离与稳定性

令 $\Delta = \{(t,t) \in \mathbb{R}^2 : t \in \mathbb{R}\}$ 表示对角线。对于条形码 X, Y（由点 (b,d) 构成的有限多重集，且满足 $b < d$），ε-匹配 是 $X \cup \Delta$ 与 $Y \cup \Delta$ 之间的部分匹配，使得匹配的点对之间的距离不超过 L_∞，未匹配的点到对角线的距离不超过 ε。瓶颈距离为

$$d_B(X,Y) := \inf\{\, \varepsilon \geq 0 \;:\; \text{there exists an } \varepsilon\text{-matching} \,\}.$$

定理 2（外幂的稳定性）。对于 $i \geq 1$ 和温和持久模 M, M',

$$d_B\big(B(\Lambda^i M),\, B(\Lambda^i M')\big) \;\leq\; d_B\big(B(M),\, B(M')\big).$$

证明。假设 M 和 M' 是 ε-交错的，即存在满足交换结构映射并满足移位 2ε下通常之字形关系的线性映射 $f_t : M_t \to M'_{t+\varepsilon}$ 和 $g_t : M'_t \to M_{t+\varepsilon}$。由于 Λ^i 是一个作用于向量空间、且保持线性映射的函子，对每个 f_t, g_t 应用 Λ^i 可得到自然变换

$$\Lambda^i f_t : \Lambda^i M_t \;\to\; \Lambda^i M'_{t+\varepsilon}, \qquad \Lambda^i g_t : \Lambda^i M'_t \;\to\; \Lambda^i M_{t+\varepsilon}.$$

这些自然变换与 $\Lambda^i M$ 和 $\Lambda^i M'$所诱导的结构映射交换，因为函子保持交换图。此外，之字形恒等式在 Λ^i作用下仍然成立，因为若 $g_{t+\varepsilon} \circ f_t$ 等于移位映射 $M_t \to M_{t+2\varepsilon}$，则 $\Lambda^i g_{t+\varepsilon} \circ \Lambda^i f_t$ 等于 $\Lambda^i M_t$上的移位映射。因此，$\Lambda^i M$ 和 $\Lambda^i M'$也互为 ε-交错。根据持久性的基本等距定理（[4, 5]），条形码上的交错距离等于瓶颈距离。因此$d_B\big(B(\Lambda^i M)\; B(\Lambda^i M')\big) \;\leq\; d_B\big(B(M)\; B(M')\big)$.

这一结果表明，外幂保留了持久性的经典稳定性，确保高阶交互特征在输入数据受到扰动时依然稳健。

2.4 计算模型与数据结构

我们在带比较操作的 RAM（随机存取机）模型中分析运行时间；对 $O(M)$ 个端点进行排序需要 $O(M \log M)$，从信息论角度看，这在该模型中是无法避免的（参见第 6 节；另见 [15, 6]）。我们使用不同死亡时间的坐标压缩 来 $\{1, \ldots, N\}$。

[4]当多个区间长度相同时，并列项的顺序可以任意排列，因为我们的结果涉及 Top-K 多重集和排序后的长度向量 $\mathbf{L}_K(M,\ i)$，它们不受平局处理的影响（参见 top-K 聚合 [11]）。

持久化顺序统计（*OS*）树。 我们将使用一种标准的持久化线段树，在 $\{1,\dots,N\}$ 上存储每个压缩死亡坐标处的存活条形计数（参见 [8]）。它支持:

– Update（根，位置 ±1）在 $O(\log M)$ 时间内返回一个新根，并保持旧根不可变；– KthFromRight（根 k）：按非递增顺序返回第 k 个存活条形码的死亡值，重数也计入其中。等价地，该树维护每个坐标处存活条形的累积计数，并通过查询遍历这些计数，从右侧定位第 k 个条形；– Size（根）在 $O(1)$ 时间内完成。

扫描过程中，对于 r 的每个出生，我们存储插入 r （编码 A_r）之前的快照根节点 T_r ，以及整数 $c_r = \textsc{Size}(T_r)$ 。总空间为：堆缓冲区占用 $O(M)$ ，持久化占用 $O(M \log M)$ 个节点。

3 算法结构分解定理

现在，我们推导出 $B(\Lambda^i M)$ 的出生锚定、按秩分组的描述，并以此作为最佳优先算法的基础。本节始终采用第 2 节的扫描／平局约定（条形码操作的背景知识另见 [14, 25] ）。

3.1 锚点与存活集合

给定一个 i 元组 $I = \{\ell_1,\dots,\ell_i\} \subseteq \{1,\dots,M\}$，满足 $\max_j b_{\ell_j} < \min_j d_{\ell_j}$，令

$$t^\star := \max_j b_{\ell_j}, \qquad A^\star := \{\, \ell_j : b_{\ell_j} = t^\star \,\}.$$

根据我们的事件顺序（死亡先于出生，并在相同时间的出生之间固定一个全序），存在一个唯一的索引

$$r^\star \in A^\star \quad \text{that is processed last at time } t^\star.$$

我们称 $r^\star$ 为 I 的锚点。在 $b_{r^\star} = t^\star$ 之前的那一刻， $I \setminus \{r^\star\}$ 的所有元素都存活；因此 $I \setminus \{r^\star\} \subseteq A_{r^\star}$，其中 $A_{r^\star}$ 是第 2 节中的存活集合。按死亡时间非递增顺序排列 $A_{r^\star}$ ，将死亡时间记为 $d_{r^\star}(1) \geq d_{r^\star}(2) \geq \cdots \geq d_{r^\star}(c_{r^\star})$。

引理 1。假设扫描顺序规定相同时间的死亡先于出生，并在同时发生的出生之间固定一个全序。对于定理 *1* 中由 $I = \{\ell_1,\dots,\ell_i\}$ 产生的任意 Λ^i 区间，令 $t^\star = \max_j b_{\ell_j}$。那么存在唯一的锚点 $r^\star$，即在 $t^\star$ 时出生的元素中最后处理的索引，并且 $I \setminus \{r^\star\} \subseteq A_{r^\star}$。反过来，对于任意 r 以及 A_r 的任意$(i-1)$子集， i 元组 $\{r\} \cup S$ 会产生一个出生时间为 b_r 的（长度可能为零的） Λ^i 区间。

证明。 根据定理 1，来自 $I = \{\ell_1,\dots,\ell_i\}$ 的 Λ^i 区间为 $[\max_j b_{\ell_j},\ \min_j d_{\ell_j})$。令 $t^\star = \max_j b_{\ell_j}$。在出生时间为 $t^\star$ 的索引中，恰有一个会按照平局规则最后处理；记为 $r^\star$。在时间 $b_{r^\star}$，所有其他 ℓ_j 都存活，因此 $I \setminus \{r^\star\} \subseteq A_{r^\star}$。反过来，对于任意锚点 r 和任意 $(i-1)$ 子集 $S \subseteq A_r$， i 元组 $\{r\} \cup S$ 根据同一公式给出区间 $[b_r, \min\{d_r, d_s : s \in S\})$。

3.2 固定锚点处的秩分组

固定一个锚点 r，其存活集合为 A_r，大小为 c_r。令 $S = \{j_1 < \cdots < j_{i-1}\} \subseteq \{1, \ldots, c_r\}$ 为选定邻居的秩（因此，相应的死亡时间为 $d_r(j_1) \ldots, d_r(j_{i-1})$）。由 $(r,\ S)$ 产生的 Λ^i 区间为

$$I(r, S) \ = \ \big[b_r,\ \min\{d_r,\ d_r(j_1), \ldots, d_r(j_{i-1})\}\big),$$

其长度按定义 (1) 等于 $|I(r,\ S)| \ = \ \ell_r(\max S)$。因此，长度仅取决于 S中的最大秩。

命题 1。 固定 r和一个秩 $j \in \{i-1, \ldots, c_r\}$。秩的$(i-1)$- 子集 S 中最大秩为$\max S = j$ 的子集数为 $w_i(j) \ = \binom{j-1}{i-2}$。所有这些子集产生相同的长度 $\ell_r(j)$，并按 (1) 的方式将下限截为 0 。特别地，当 $d_r(j) \le b_r$时，所得值为 $\ell_r(j) \ = 0$，这对 $B(\Lambda^i M)$没有贡献。

证明。 要使最大值为 $\max S = j$，必须包含秩 j，并从 $\{1, \ldots, j-1\}$ 中选择其余的$i-2$ 个秩，共有 $\binom{j-1}{i-2}$ 种选择。长度相同是因为 $|I(r,\ S)| = \min\{d_r, d_r(j)\} - b_r$ 仅取决于最大秩（并按 (1) 的方式将下限截为 0）。

定理 3（锚点秩分组）。假设条形码为左闭右开，并列情况通过先处理死亡时间、再处理出生时间来打破，且同时出生的条具有固定的全序。对于每个出生时间 r，令 A_r 为在b_r之前仍存活的条的集合，并按死亡时间 d_r（1） $\ge \cdots \ge d_r$（c_r）非递增排序。定义

$$\ell_r(j) := \max\{0,\ \min\{d_r, d_r(j)\} - b_r\}, \qquad J_r := \{\, j \in \{i-1, \ldots, c_r\} : \ell_r(j) > 0 \,\}.$$

于是， B（$\Lambda^i M$）的每个区间都唯一对应于一对 （r, S），其中 r 是锚点（在 $t^\star = \max_j b_{\ell_j}$ 处最后处理的出生时间），且 $S \subseteq A_r$ 具有 $|S| = i-1$，其长度为 $|I$（r, S）$| = \ell_r(\max S)$。因此，对于每个锚点 r，锚定多重集等于

$$\{\, \ell_r(j) \ \textit{with multiplicity}\ w_i(j) = \tbinom{j-1}{i-2} : j \in J_r \,\},$$

$j \mapsto \ell_r$（j）非递增，且整体上

$$B(\Lambda^i M) \ = \ \bigsqcup_{r=1}^{M} \{\, \ell_r(j) \ \textit{with multiplicity}\ w_i(j) : j \in J_r \,\},$$

有限（正长度）区间的多重集的不交并。

证明。固定 $i \ge 2$。根据区间演算（定理 1），任意满足 $\max_j b_{\ell_j} < \min_j d_{\ell_j}$ 的 i-元组 I $=\{\ell_1, \ldots, \ell_i\}$ 都会生成 Λ^i–区间 $[\max_j b_{\ell_j},\ \min_j d_{\ell_j})$。令 $t^\star = \max_j b_{\ell_j}$，并选取满足 $b_{\ell_j} = t^\star$ 的元素中最后处理的唯一索引 $r^\star$。那么 $I \setminus \{r^\star\} \subseteq A_{r^\star}$，且 $r^\star$ 是 I 的锚点。反过来，对于任意锚点 r 和 $(i-1)$-子集 $S \subseteq A_r$， i-元组 $\{r\} \cup S$ 给出

$$I(r, S) = \big[\, b_r,\ \min\{d_r,\ d_s : s \in S\}\,\big).$$

因此， $B(\Lambda^i M)$ 中的每个元素都唯一地来自某个对 $(r,\ S)$。

现在按死亡时间 $d_r(1) \geq d_r(2) \geq \cdots \geq d_r(c_r)$从大到小对 A_r 排序。若 $S = \{j_1 < \cdots < j_{i-1}\}$ 是选定邻居的秩，则$I(r, S)$的长度仅取决于最大秩： $|I(r, S)| = \ell_r(\max S)$。因此，具有相同最大秩 j 的所有 $(i-1)$-子集都会得到相同的长度 $\ell_r(j)$。

此时命题 1 适用：它告诉我们，此类子集的数量恰为 $w_i(j) = \binom{j-1}{i-2}$，且它们的贡献值 $\ell_r(j)$（截断为 0）都相同。因此，对于每个锚点 r，锚点对应长度的多重集为

$$\big\{\, \ell_r(j) \text{ with multiplicity } w_i(j) \ :\ j \in \{i-1, \ldots, c_r\} \,\big\}.$$

最后，定义 $J_r = \{j : \ell_r(j) > 0\}$。限制到 $j \in J_r$ 可去除长度为零的区间，因为它们不属于 $B(\Lambda^i M)$。由于每个区间都有唯一的锚点，全局条形码就是对各锚点取不相交的多重集并：

$$B(\Lambda^i M) \ = \ \bigsqcup_{r=1}^{M} \big\{\, \ell_r(j) \text{ with multiplicity } w_i(j) \ :\ j \in J_r \,\big\}.$$

$j \mapsto \ell_r(j)$ 的单调性可直接由排序 $d_r(1) \geq d_r(2) \geq \cdots$ 得出。这证明了该定理。

当 $i = 2$, $w_2(j) = \binom{j-1}{0} = 1$ 时，每个秩恰好贡献一个元素，因此锚定流会成为一个简单的非递增序列$\ell_r(1) \geq \ell_r(2) \geq \cdots \geq \ell_r(c_r)$。

上述定理将 $B(\Lambda^i M)$ 完整且无冗余地分解为按锚点划分的单调流，并给出其重数的闭式表达，为高效的 Top-K 算法奠定了结构基础，同时准确揭示了高阶区间的组织方式。

4 Top-K–Λ^i 算法

现在，我们给出一种最佳优先算法，它无需枚举整个多重集，即可输出 K 个最长的 $B(\Lambda^i M)$而元素。

4.1 预处理：扫描与持久化顺序统计

构建包含所有出生时间和死亡时间的全局事件列表，并按时间排序；时间并列时，死亡时间排在出生时间之前（第 2 节；参见 [10, 9]）。将不同的死亡时间坐标压缩为 $\{1, \ldots, N\}$，并在该轴上维护一棵持久化顺序统计树，用于存储存活死亡时间的计数。在扫描过程中：

– 当条形码 x 发生死亡时，在 d_x 的索引处执行更新 -1。– 当条形码 r 在时间 b_r 出生时，在插入 r 之前：存储当前快照根节点 T_r，以编码 A_r，并记录 $c_r = \textsc{Size}(T_r)$；然后在 d_r 的索引处插入 $+1$，使 r 对之后的锚点保持存活。

这需要 $O(M \log M)$ 的时间和 $O(M \log M)$ 个持久化节点。

4.2 最优优先 Top-K 提取算法

根据定理 3，$B(\Lambda^i M)$ 是按秩分组的流的多重集并。我们运行分组式最佳优先搜索，其中每个堆条目表示锚点 r 的流在秩 j 处当前的队首（$r,\ j$），其键为 $\ell_r(j)$，权重为 $w_i(j) = \binom{j-1}{i-2}$。这与结构化集合上的 top-$K$ 聚合和选择中的经典最佳优先范式相呼应（参见 [11, 13]）。

完整过程见下方的算法 1。该算法利用 $B(\Lambda^i M)$ 按秩分组的结构，以接近输出敏感时间的复杂度计算精确的 Top-K 区间；当 $K \ll |B(\Lambda^i M)|$ 时，无需进行完全枚举。

算法 1 TopK–Λ^i（分组的最优优先；固定 $i \geq 2$）

输入： 条形码 $B(M)$、层 $i \geq 2$、目标 K

扫描与快照： 如上所述，获取 $\{(T_r, c_r)\}_{r=1}^{M}$。
初始化一个空最大堆 H，以长度为键。
对于 每个锚点 r，若 $c_r \geq i-1$ **则执行**
    $j \leftarrow i-1$；在 T_r 上查询 $d_r(j)$；通过 (1) 设置 $L \leftarrow \ell_r(j)$
    如果 $L > 0$ **则**
        将 $(L, r, j, w_i(j))$ 压入 H
    结束如果
结束循环
$S \leftarrow \emptyset$ {S 收集输出长度（含重数）}
当 $|S| < K$ 且 H 非空 **时**
    从 H中弹出（$L,\ r,\ j,\ w$）
    将 $\{w,\ K - |S|\}$ 个 L 的副本追加到 S {按锚点批量输出并列项；参见 §2.2}
    如果 $j < c_r$，**则**
        $j \leftarrow j+1$；查询 $d_r(j)$；$L \leftarrow \ell_r(j)$
        如果 $L > 0$，**则**
            将 $(L,\ r,\ j,\ w_i(j))$ 压入 H
        如果结束
    如果结束
结束循环
返回 S（按非递增顺序排列的 Top-K 多重集）

区间标识。 算法 1 直接输出 Top-K 长度多重集 。如果需要实际的区间标识 , 则可将第 12 行的每次批量输出展开为实现该重数$w_i(j)$的秩的显式（$i-1$）-子集, 并在产生 K 个区间后截断。对于固定的 i，这种细化保留了渐近复杂度界。

未捆绑（余字典序）变体。 或者，也可以将状态表示为严格递增的 $(i-1)$ 元秩元组，并在每次弹出时至多扩展 i 个余字典序邻居；这样得到的输出相同，但循环开销会增加一个 i 因子（参见 [13, 11]）。为获得最紧的界，我们聚焦于分组变体。

定理 4（正确性与复杂度）。对于固定的 $i \geq 2$，算法 *1*（分组变体）按非递增顺序输出 $B(\Lambda^i M)$ 中恰好 K 个最长元素（计入重数）

$$O\big((M+K)\log M\big)$$

时间，使用 $O(M)$ 堆空间，以及 $O(M$对数 $M)$ 个持久化节点。另一种未捆绑（余字典序）变体的运行时间为 $O\big($（$M+iK$）对数 $M\big)$。

证明。我们先证明正确性。根据定理 3，全局多重集是单调流 $\{\ell_r(j)\}$ 的不相交并集，其权重为 $w_i(j)$。堆中恰好存储所有非空流当前的首项。由于每个流都是非递增的，且堆键是首项长度，因此一旦弹出 $\ell_r(j)$，就不会有尚未出现的元素大于它，因为其余元素都不超过各自流的首项，而所有首项都在堆中。因此，算法会一次性输出全部 $w_i(j)$ 个副本；所以，并列情况可以按锚点分组，这与第 2 节中的并列策略一致。由于前 -K 个向量在相等值的排列下保持不变，因此批量输出是安全的。推进该流后，不变式仍然成立。由归纳法可知，输出恰好是按顺序排列的全局前-K 个元素。

我们证明这些复杂度结论。预处理的成本为 $O(M\log M)$。在初始化期间，每个具有 $c_r \geq i-1$ 的锚点至多贡献一个堆条目，该条目通过在 $j=i-1$ 处进行一次 KthFromRight 查询获得。具有 $c_r < i-1$ 的锚点不贡献任何条目，因此初始堆条目的数量至多为 M，从而总体耗时为 $O(M\log M)$。每次弹出至少输出一个元素，因此至多进行 K 次弹出（如果总输出为 $< K$ 时 H 变为空，则次数更少）。一次弹出执行 $O(1)$ 次堆操作和一次顺序统计查询，二者各需 $O(\log M)$，因此分组变体中的循环复杂度为 $O(K\log M)$，非捆绑变体中的复杂度为 $O(iK\log M)$。空间复杂度界由堆的大小 $O(M)$ 和持久化树得出。在 RAM 模型下，所有 $\log M$ 因子均成立，其中基本算术运算和内存访问耗时 $O(1)$。

在特殊情况 $i=2$ 下：由于对所有 j 都有 $w_2(j)=1$，每次弹出都会输出单个元素并推进 $j \mapsto j{+}1$，从而得到所述的 $O((M+K)\log M)$ 界。

5 Top-K 长度向量的稳定性

我们证明，在输入条形码 $B(M)$ 发生瓶颈扰动时，$B(\Lambda^i M)$ 的 Top-K 长度向量会以 Lipschitz 连续方式变化（常数为 2）。本节中，$i \geq 1$ 固定不变。

回顾第 2 节，$\mathbf{L}_K(M,i) = (L_1 \geq \cdots \geq L_K)$ 表示 $B(\Lambda^i M)$ 的非递增 Top-K 长度向量（带重数），必要时以零补齐。

定理 5（Top-K稳定性）。如果 $d_B\big(B(M)\ B(M')\big) \leq \varepsilon$，那么对于每个固定的$i \geq 1$ 和 $K \geq 1$，

$$\big\|\mathbf{L}_K(M,i) - \mathbf{L}_K(M',i)\big\|_\infty \ \leq\ 2\varepsilon.$$

证明。令 $X = B(\Lambda^i M)$ 和 $Y = B(\Lambda^i M')$。根据外幂的稳定性（定理 2），有 $d_B(X,\ Y) \le \varepsilon$。因此，在$X \cup \Delta$ 与 $Y \cup \Delta$ 之间存在一个 ε-匹配（参见 [7, 5]）。

构造长度多重集 $S = \{\, d-b : (b,d) \in X \,\}$和 $T = \{\, d'-b' : (b',d') \in Y \,\}$. 。由 ε-匹配可得双射 π，它在较短的多重集中补入零后建立于 S 和 T之间：对于任意匹配对 $(b,d) \leftrightarrow (b',d')$，令 $\pi(d-b) = d'-b'$；对于与对角线匹配的任意区间，则将其长度与 0 配对。对于匹配的区间，有 $\big|(d-b)-(d'-b')\big| \le |d-d'|+|b-b'| \le 2\varepsilon$；若 (b,d) 在 ε内与 Δ匹配，则根据与对角线进行瓶颈匹配的定义，$|d-b| \le 2\varepsilon$。因此：对所有 $x \in S.,\ |x-\pi(x)| \le 2\varepsilon$

令 $(s_1 \ge s_2 \ge \cdots)$ 和 $(t_1 \ge t_2 \ge \cdots)$ 分别为 S 和 T 的非递增重排（补零至长度相等）。我们声称，对所有 k，都有 $|s_k - t_k| \le 2\varepsilon$。反设 $s_k > t_k + 2\varepsilon$。那么 S 至少有 k 个元素$\ge s_k$，因此它们在 π 下的像为 $\ge s_k - 2\varepsilon > t_k$，这意味着 T 至少有k 个元素严格大于 t_k，与 t_k的定义矛盾。反向不等式 $t_k > s_k + 2\varepsilon$ 同理可证。因此，对所有 k，都有 $|s_k - t_k| \le 2\varepsilon$。最后，限制到前 K 个坐标，得到 $\big\|\mathbf{L}_K(M,\ i)\ -\mathbf{L}_K(M', i)\big\|_\infty \le 2\varepsilon.$

注：系数 2 是紧的：通过平移 $b \mapsto b+\varepsilon$ 和$d \mapsto d-\varepsilon$ 来扰动一个区间 $[b,d)$；瓶颈距离为 ε，而长度恰好变化 2ε。

6 最优性：比较模型下界

我们证明，定理 4 中的 $O(M\log\ M)$ 预处理项在比较模型下不可避免，即使目标仅仅是寻找最长的单个 Λ^2 区间。

命题 2（下界）。任何基于比较的算法，若输入为条形码 $B(M)$ 未排序的端点，并计算 $B(\Lambda^2 M)$ 的 *Top*-1 长度，则在最坏情况下必须执行 $\Omega(M\log\ M)$ 次比较。

证明。我们将元素唯一性（或者等价地，一维最小间距的判定版本）归约为计算 Top-1 Λ^2 长度。考虑一个实数输入集合 $\{x_1, \ldots, x_M\} \subset [0, \frac{1}{2}]$（其中的数不必互不相同）。构造一个包含条形码 $J_r := [\, x_r,\ x_r + 1\,)$ $(r = 1, \ldots, M)$。的条形码。对于任意一对 $(, s)$，Λ^2 交集长度等于 $|J_r \cap J_s| = \max\{0,\ 1 - |x_r - x_s|\}$. 由于所有$|x_r - x_s| \le \frac{1}{2} < 1$，任意一对都相交，且 $\max_{r\neq s} |J_r \cap J_s| = 1 - \min_{r\neq s} |x_r - x_s|$. 因此，Top-1 Λ^2 长度等于 1 当且仅当存在重复元素（$\min_{r\neq s} |x_r - x_s| = 0$）。所以，任何能够计算 Top-1 Λ^2 长度的算法都能判定元素唯一性。后者在代数决策树/比较模型中需要 $\Omega(M\log\ M)$ 次比较；因此，计算 Top-1 Λ^2 长度同样需要$\Omega(M\log\ M)$ 次比较。

7 实验验证

我们在合成条形码上评估所提出的最佳优先算法，在这些条形码中，我们可以直接控制重叠 （即期望并发数），而这又决定了

输出大小 $K_{\text{all}} = |B(\Lambda^2 M)|$。我们的目标是：(i) 根据完全枚举基线验证结果的精确性；(ii) 量化运行时间随 K 和重叠的变化而产生的改进。

每次试验中，我们按以下方式在 $[0,1]$ 上抽取 M 条：出生时间 $b \sim \mathsf{Unif}[0,1)$，以及独立的指数分布长度 $L \sim \mathsf{Exp}(\lambda)$，并截断至 $[0, 1-b]$；均值参数由 $\mathbb{E}[L] = \mathsf{Lmean} \in \{0.03, 0.05\}$ 设定。这样得到的期望并发数为 $\approx M \cdot \mathsf{Lmean}$，因此 Lmean 越大，重叠越严重，$K_{\text{all}}$ 也越大。

我们重点关注 $i = 2 (\Lambda^2)$，此时分组最佳优先界最紧。[5]基线 $=$ Enum–Λ^2（完整的两两枚举）$+$ TopK-Select（基于堆的选择；参见经典持久同调流水线 $[10, 26]$ 和快速工具链，例如 $[1]$）。我们的方法 $=$ TopK–Λ^2（算法 1）。我们测量每次查询的总实际运行时间，并报告加速比：$=$ 基线时间 我们的方法时间 $\cdot$ 通过检查前 K 个长度的多重集是否相等（允许数值舍入误差）来验证正确性。

我们使用 $M \in \{3000, 5000\}$、$\mathsf{Lmean} \in \{0.03, 0.05\}$ 和 $K = 10$、000。对于每种设置，我们报告一次代表性运行；在所有运行中，TopK–Λ^2的输出都与 Enum–Λ^2 完全一致。

表 1 总结了结果（墙钟时间，单位为秒）。在高重叠情形下：当 K_{all} 较大时，TopK–Λ^2 始终更胜一筹（在 K=10^4 时为 2.1×，对应M=3000、Lmean=0.05），因为它避免了基线固定的“枚举所有内容”开销。在中等重叠情形下：对于M=5000、Lmean=0.03，我们获得了稳定的提升（1.45×）。所有运行的 Top-K 长度都与基线完全一致。

表 1。墙钟时间结果与加速比。$K_{\text{all}} = |B(\Lambda^2 M)|$ 随重叠增加而增长。

设置	K	K_{all}	基线（秒）	我们的方法（秒）	加速比
M=3000, Lmean= 0.05 （高）	10,000	422,272	0.262	0.125	**2.10**×
M=5000, Lmean= 0.03 （中等）	10,000	733,622	0.468	0.322	1.45×

8 结论

我们提出了对$B(\Lambda^i M)$进行按出生时间锚定、按秩分组的结构分解（定理 3），并以此设计了一种最佳优先 Top-K 算法（定理 4），无需 完全枚举即可返回精确的 Top-K 多重集。对于固定的 $i \geq 2$，分组算法的运行时间为 $O((M + K) \log M)$，而未打包的字典序变体运行时间为 $O((M + iK) \log M)$。我们证明，Top-K 长度向量相对于瓶颈扰动是 2-利普希茨的（定理 5），并证明了一个比较模型下界（命题 2），表明 $O(M \log M)$预处理在信息论意义上不可避免。实验验证了理论结果：在高重叠区间，加速比最高可达 $2.1\times$，且输出精确。

除了理论贡献之外，这项工作还表明，高阶持久性构造可以实现实际上的可扩展性，从而使其能够被系统地

[5] $i = 3$（及更高阶）的实证表现与 $i = 2$的表现一致，这表明观察到的提升并非 $i = 2$ 这一情形所独有。

用作数据分析、机器学习和计算几何中的稳定且具有区分性的特征（例如，用于分子图分类任务）。我们还计划开发本方法的变体，以应用于逻辑、范畴论与机器学习的交叉领域 [18–24, 2]。

参考文献

1. Bauer, U.: Ripser：高效计算 Vietoris–Rips 持久条形码。 *Journal of Applied and Computational Topology* 5, 391–423 (2021). 2. Bennett, M. T., Maruyama, Y.: 共情伦理人工智能的哲学规范。 *IEEE Transactions CDS* 14(2), 292–300 (2021). 3. Ben-Or, M.: 代数计算树的下界。载于: *Proc. 15th ACM* , – ( ). 4. *Symposium on Theory of Computing (STOC)* 80 86 1983 Chazal, F., de Silva, V., Oudot, S.: 几何复形的持久性稳定性。*Geometriae Dedicata* 173(1), 193–214 (2014). 5. Chazal, F., de Silva, V., Glisse, M., Oudot, S.: *The Structure and Stability of Persistence Modules* 2016 .

Springer 数学简明读本，Springer ( ). 6. Cormen, T.H., et al.: *Introduction to Algorithms*, MIT 出版社 (2009). 7. Cohen-Steiner, D., Edelsbrunner, H., Harer, J.: 持久图的稳定性。 *Discrete & Computational Geometry* 37(1), 103–120 (2007). 8. Driscoll, J.R., Sarnak, N., Sleator, D.D., Tarjan, R.E.: 构造持久化数据结构。 *J. Comp. Sys. Sci.* 38(1), 86–124 (1989). 9. Edelsbrunner, H., Harer, J.: *Computational Topology: An Introduction*. 美国数学会 (2010). 10. Edelsbrunner, H., Letscher, D., Zomorodian, A.: 拓扑持久性与简化。 *Discrete & Computational Geometry* 28(4), 511–533 (2002). 11. Fagin, R., Lotem, A., Naor, M.: 中间件的最优聚合算法。*Journal of Computer and System Sciences* 66(4), 614–656 (2003). 12. Fenwick, P.M.: 一种新的累积频数表数据结构。 *Software: Practice and Experience* 24 3 327 336 1994 ( ), – ( ).

13. Frederickson, G.N., Johnson, D.B.: $X+Y$ 及列已排序矩阵中的选择与排序复杂度。 *J. Comp. Sys. Sci.* 24(2), 197–208 (1982). 14. Ghrist, R.: 条形码：数据的持久拓扑。 *Bulletin of the American Mathematical Society* 45 1 61 75 2008 ( ), – ( ).

15. Knuth, D.E.: *The Art of Computer Programming, Vol. 3: Sorting and Searching*，第 2 版，1998 年。Addison–Wesley ( ). 16. Lampret, L.: 持久模的张量、对称、外幂及其他幂。 *arXiv preprint* arXiv:1503.08266 (2015). 17. Lyle, J., Monta˜no, J., Sather-Wagstaff, K.: 外幂与 Tor 持久性。*Journal of Pure and Applied Algebra* 226(4), 106890 (2022). 18. Maruyama, Y.: 格值逻辑与格值模态逻辑的代数研究。 *Proc. ICLA*, 170–184 (2009). 19. Maruyama, Y.: 无点凸几何的基本结果。 *Annals of Pure and Applied Logic* 161 12 1486 1501 2010 ( ), – ( ).

20. Maruyama, Y.: 自然对偶、模态与余代数。 *Journal of Pure and Applied Algebra* 216 3 565 580 2012 ( ), – ( ).

21. Maruyama, Y.: 全 Lambek 超教义：一阶次结构逻辑的范畴语义。 *Proc. WoLLIC*, 211–225 (2013).

22. Maruyama, Y.: 范畴对偶理论：及其在域、凸性和分布单子中的应用。 *Proc. CSL*, 500–520 (2013). 23. Maruyama, Y.: 认知的符号与统计理论：迈向整合式人工智能。 *Proc. SEFM*, 129–146 (2020). 24. Maruyama, Y.: 通用人工智能的条件。 *Proc. AGI*, 242–251 (2020). 25. Oudot, S.Y.: *Persistence Theory: From Quiver Representations to Data Analysis*. 数学调查与专著 209，美国数学会 (2015). 26. Zomorodian, A., Carlsson, G.: 计算持久同调。 *Discrete & Computational Geometry* 33 2 249 274 2005 ( ), – ( ).