arXiv:2603.29825v1 [cs.DS] 31 Mar 2026

H子式自由图中的模式稀疏树分解*

Dániel Marx† Marcin Pilipczuk‡ Michał Pilipczuk§

摘要

给定一个 H子式自由图 G 和一个整数 k，我们的主要技术贡献是在随机多项式时间内采样一个 G 的诱导子图 G' 以及 G' 的树分解，其宽度为 $\mathcal{O}(k)$，使得对于每个大小为 k的 $Z \subseteq V(G)$，以至少 $\left(2^{\widetilde{\mathcal{O}}(\sqrt{k})}|V(G)|^{\mathcal{O}(1)}\right)^{-1}$的概率，我们有 $Z \subseteq V(G')$，并且该树分解的每个袋至多包含 Z的 $\widetilde{\mathcal{O}}(\sqrt{k})$个顶点。这样的树分解使我们能够以 $2^{\widetilde{\mathcal{O}}(\sqrt{k})}n^{\mathcal{O}(1)}$ 的（随机）时间解决一大类问题，其中解是一个大小为 k的模式Z，例如有向 k-路径、 H-打包等。特别是，我们的结果涵盖了 Fomin 等人关于模式覆盖结果的所有算法应用 [*SIAM* 计算杂志 2022]（该结果要求模式连通），以及 Nederlof [计算理论研讨会 2020]上提出的平面子图查找算法。

此外，对于 $K_{h,\,3}$-无关图（其中包括有界亏格图）以及固定常数 d，我们通过确保不仅 Z 与每个袋的交集为 $\widetilde{\mathcal{O}}(\sqrt{k})$，甚至其距离 -$d$ 邻域 $N_G^d[Z]$ 也满足这一性质，显著强化了该结果。这一扩展使我们能够处理更广泛的问题：在这些问题中，模式的邻域也会对解产生影响，例如部分支配问题和涉及距离约束的问题。

*本研究是项目 BUKA（Ma. P.）和 BOBR（Mi. P.）的一部分，这两个项目分别获得了欧洲研究理事会（ERC）根据欧盟地平线 2020研究与创新计划提供的资助（资助协议编号分别为 948057 和 101126229）。

†CISPA 亥姆霍兹信息安全中心，德国。marx@cispa.de
‡华沙大学信息学研究所，波兰。m.pilipczuk@uw.edu.pl§华沙大学信息学研究所，波兰。
michal.pilipczuk@mimuw.edu.pl

目录

11引言 引言 111.1 1 算法应用 . 算法应用 .3

2 2 概述概述 6

22.1 1 对偶结果 . 对偶结果 .7 22.2 2 转向稀疏分离 . 转向稀疏分离 .8 22.3 3 主算法 . 主算法 .11 22.4 4 扩展到有界距离版本（$d \geqslant 1$） . 扩展到有界距离版本（ ） .13

33预备知识 预备知识 16

33.1 1 图小 minors。图小 minors.17 33.2 2 乘积结构理论中的工具。乘积结构理论中的工具.18 33.3 3 簇与簇族。簇与簇族.20 4 4 流-割对偶 流-割对偶 20 44.1 1 距离版本。距离版本.21 5 5 分隔集改进过程 分隔集改进过程 30 6 6 算法 算法 39 66.1 1 不含顶点的近可嵌入图。不含顶点的近可嵌入图.396 2 H $d = 0$. 将其提升到 -子式-自由图中，其中 . 将其提升到 -子式-自由图中，其中.52

6 3 $K_{3\,h}$ $d \geqslant 1$. 将其提升到 , -子式-自由图中，其中 . 将其提升到 , -子式-自由图中，其中.56 A 定理 4 的证明 244 24 2..2 定理 .. 的证明

1 引言

人们投入了大量研究精力，致力于为平面图设计参数化算法，使其运行时间随解的规模或输入实例的其他某个参数呈次指数级增长[12, 14, 16, 17, 19–2121，2626，2929，3333，3434，3838–43]。虽然其中许多算法采用针对具体问题的论证，但人们也发现了若干有用的通用技术和设计模式。在某些情况下，这些技术还可推广到有界亏格图或 H-子式-自由图。

双维性理论为某些问题提供了一种非常简洁的双赢方法：对于参数为 k 的实例，如果树宽为 $\Omega(\sqrt{k})$，答案便是显而易见的。然而，这类问题的范围有限，而且对问题定义进行自然推广，往往会破坏这一性质；涉及终端点和有向图的问题尤其如此。Marx 等人 [40]和 Bandyapadhyay 等人 [3]提出了处理环命中问题的一般技术：任务是删除至多 k 条边/个顶点，使所得图的连通分量或二连通分量满足某种性质（例如，不存在经过某个终端点的环）。然而，这一框架本质上针对的是删除类问题，只有当剩余图所需满足的性质能够以特定方式用约束描述时，才能处理相应问题。

Fomin 等人 [27] 提出了一种强大的技术，可用于许多解为大小 k 的连通模式的问题。对于这类问题，将以下结果作为黑箱，并结合一个求解有界树宽实例的算法，即可得到一个次指数参数化算法。顶点图是指删除一个顶点后便可变为平面图的图。注意，$K_{3,\ h}$ 是一个顶点图，并且对于每个 $g \geqslant 0$，都存在一个 h，使得欧拉亏格至多为 g 的图均为 $K_{3,\ h}$-子式自由图；因此，该结论尤其适用于有界亏格图。

定理 1.1 (Fomin 等人 [27])。 设 $\mathcal{C}$ 是一类排除固定顶点图作为子式的图。则存在一种随机多项式时间算法：给定一个来自 $\mathcal{C}$ 的 n 顶点图G 和一个整数 k，该算法对顶点子集 $A \subseteq V(G)$ 进行采样，并满足以下性质：

(P1) 诱导子图 $G[A]$ 的树宽为 $\mathcal{O}(\sqrt{k}\log k)$。

(P2) 对于每个顶点子集 $Z \subseteq V(G)$，若 $|Z| \leqslant k$ 诱导出 G 的一个连通子图，则 Z 被 A 覆盖的概率，即 $Z \subseteq A$，至少为

$$\left(2^{\mathcal{O}(\sqrt{k}\log^2 k)} \cdot n^{\mathcal{O}(1)}\right)^{-1}.$$

假设我们要研究一个问题，其解是一个连通集合 Z，大小为 k，也就是说，$G[Z]$ 是连通的。作为一个贯穿始终的例子，我们考虑在有向图中寻找长度恰为 k 的有向环这一问题。我们调用定理 1 11。.1 中的算法（作用于底层无向图），并令 A 为该算法返回的集合。由于 $G[A]$ 的树宽为$\mathcal{O}(\sqrt{k}\log k)$，我们可以使用标准的动态规划技术，在时间 $2^{\widetilde{\mathcal{O}}(\sqrt{k})}n^{\mathcal{O}(1)}$内测试 $G[A]$ 中是否存在有向 k-环。如果找到 k-环，就可以停止，因为它也是 G 中的 k-环。否则，我们再次调用定理 1 11..1 来抽取一个新的集合 A，并重复这一过程。如果存在有向 k-环 Z，那么根据 (P2)，经过 $\mathcal{O}(2^{\mathcal{O}(\sqrt{k}\log^2 k)} \cdot n^{\mathcal{O}(1)})$ 次重复后，我们以高概率遇到一个满足 $Z \subseteq A$ 的集合 A，从而在 G中找到有向k-环。

为了将这种方法用于其他算法问题，该问题需要满足三个主要性质：

~1. 该问题可在树宽为 t 的图上于 $2^{\mathcal{O}(t)}n^{\mathcal{O}(1)}$ 时间内求解。2. 解 Z 需要是连通的。3. 解 Z 的有效性仅取决于 $G[Z]$，不受 Z 之外顶点的影响。

第一点的限制并不严格：在有界树宽图上设计算法已有相当成熟的方法，而在树分解上使用标准的动态规划技术即可得到这类算法。值得指出的是，这些动态规划技术非常灵活稳健，因此通常只需付出很少的额外工作，就能处理不同的问题变体和扩展（如带颜色的图、有向图、权重等）。另一方面，第二点和第三点则大幅限制了这种方法能够处理的问题范围。要求解具有连通性，会排除诸如寻找多个互不相交的小型模式之类的问题；而将问题定义为关于 $G[Z]$ 的某种性质，则尤其会排除支配类问题和涉及距离的问题。

Nederlof 提出了一种用于在平面图中检测子图的亚指数算法的另一种技术[43]。这项工作建立在 Fomin 等人 [27]，的一项关键组合结果之上，该结果揭示了互不相交分隔子的链与几乎不相交路径之间某种类型的对偶关系（见下文定理 2 12..1 ）。Nederlof [43] 的一个关键新见解（见 [43, 引理 3.7]）是：我们的目标并非寻找大小为 $\widetilde{\mathcal{O}}(\sqrt{k})$ 的分隔集（这是构造宽度为 $\widetilde{\mathcal{O}}(\sqrt{k})$ 的树分解所必需的），而是只需寻找大小为 k 的多项式级别、且与解相交于 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点的分隔集。（在本文中，我们将这种性质称为“模式稀疏”；例如，[15] 中也出现了类似的思想。）事实上，如果动态规划算法考察解相对于此类分隔集的行为，那么枚举解与分隔集之间所有 $k^{\widetilde{\mathcal{O}}(\sqrt{k})} = 2^{\widetilde{\mathcal{O}}(\sqrt{k})}$ 种可能的交集，以及该交集的所有 $2^{\widetilde{\mathcal{O}}(\sqrt{k})}$ 种可能状态，所需时间仍在允许的运行时间范围内。Nederlof [43] 的算法能够处理不连通模式，具有确定性，而且为使其也能在相同运行时间内适用于问题的计数版本，研究者付出了大量技术努力。不足之处在于，该算法的呈现方式不便于将其轻松复用于其他类型的问题。此外，该算法只适用于平面图：环分离集以及关于环内外的论证，是算法不可或缺的组成部分。因此，将其推广到有界亏格图或 H-子式自由图可能会很困难。

我们的结果。基于 Nederlof 的工作 [43]，，我们提出了一种新的通用技术，其理念更接近定理 1 11。.1，但适用范围显著更广。具体而言，我们的结果具有以下优点：

- 适用于 H-子式自由图。
- 能够处理非连通解。
- 返回一个宽度关于 k呈多项式级的树分解，并且（以一定概率）该解与每个袋的交集包含 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点。
- 对于 $K_{3,\ h}$-子式自由图以及固定的 d，我们甚至可以给出概率下界，使得解的距离-d 邻域 与每个袋的交集都有 $\widetilde{\mathcal{O}}(\sqrt{k})$。

因此，例如，这项技术可用于在给定的 H-子式无关图 G 中，以 $2^{\widetilde{\mathcal{O}}\ (\sqrt{|V\ (P)\ |})}|V(G)|^{\mathcal{O}(1)}$的时间内，查找模式图 P作为子图，即使 P不连通也可以。对于 $K_{3,\ h}$-子式无关图，我们将距离-d邻域纳入考量，从而有望解决一系列新的问题。例如，我们可以解决涉及部分支配的问题，即需要找到一个顶点集/模式，使其

邻域中的顶点数最大化。此外，我们还能处理涉及距离约束的问题，例如，寻找彼此距离至少为 k 的三角形。在下文的第 1 1节中，我们将详细介绍一些可以解决的潜在问题。

形式化地说，我们关于 H-子式自由图的主要结果如下。

定理 1.2。对于每个图 H，存在一种随机多项式时间算法：给定一个 H-子式自由图 G 和一个整数 $k \geqslant 2$，输出 G 的一个诱导子图 G' 以及 G' 的一个树分解 (T, β)，其宽度为 $\mathcal{O}_H(k \log k)$，使得对于任意大小至多为 k的 $Z \subseteq V(G)$，其概率至少为

$$\left(k^{\mathcal{O}_H\left(\sqrt{k}\log k\right)}|V(G)|^{\mathcal{O}_H(1)}\right)^{-1}$$

我们有 $Z \subseteq V(G')$，并且 (T, β) 的每个袋都包含 Z的 $\mathcal{O}_H(\sqrt{k}\log^2 k)$ 个顶点。

距离版本（仅针对 $K_{3,h}$-子式自由图进行了表述）本质上是相同的：区别在于，我们限制的是与 $N_G^d[Z]$ 的交集，而不是与 Z的交集。

定理 1.3。对于任意整数 $d, h \geqslant 1$，存在一种随机多项式时间算法：给定一个 $K_{3,h}$-子式自由图 G 和一个整数 $k \geqslant 2$，输出 G 的一个诱导子图 G' 以及 G' 的一个树分解 (T, β)，其宽度为 $\mathcal{O}_{d,h}(k \log k)$，使得对于任意大小至多为 k的 $Z \subseteq V(G)$，其概率至少为

$$\left(k^{\mathcal{O}_{d,h}\left(\sqrt{k}\log k\right)}|V(G)|^{\mathcal{O}_{d,h}(1)}\right)^{-1}$$

我们有 $N_G^d[Z] \subseteq V(G')$，并且 (T, β) 的每个袋都包含 $\mathcal{O}_{d,h}(\sqrt{k}\log^6 k)$ 个 $N_G^d[Z$ 的顶点 $]$.

经过一些技术处理，定理 1 31..3 或许可以推广到顶点子式自由图。然而，这个结论显然并不适用于一般的 H-子式自由图。考虑一个树宽很大的平面图 G_0，并令 G 由引入一个新的全邻接顶点 z得到，该顶点与 G_0的每个顶点相邻。注意， G 是 K_6-子式自由的。对于 $d = 1$ 和 $k = 1$，若令 $Z = \{z\}$，则 $N_G^d[Z]$ 包含整个图 G。因此，诱导子图 G' 应当就是 G 本身，而且它不可能具有宽度为关于 $k = 1$ 的多项式的树分解。

本文聚焦于判定版本，并将计数扩展（这是[43]的主要研究主题）留待未来工作。

1.1 算法应用

本节概述一些由我们的主要结果推导出的亚指数算法。所有这些算法都要求能够在有界树宽图上求解给定问题；更准确地说，给定一个宽度为 $k^{\mathcal{O}(1)}$ 的树分解，并保证k顶点的解与每个袋相交于 $\widetilde{\mathcal{O}}(\sqrt{k})$个顶点，我们需要能够在时间$2^{\widetilde{\mathcal{O}}(\sqrt{k})} \cdot n^{\mathcal{O}(1)}$内求解该问题。有界树宽图上的算法可以使用树分解上的标准动态规划技术来设计，但文献中通常不会考虑这种确切的设定和运行时间。在这篇概述中，我们不详述这些动态规划算法：描述此类动态规划算法出了名地繁琐，而且这些细节与我们的主要技术贡献完全无关。

对于那些需要调用 $d \geqslant 1$ 定理 1 31..3的情形的应用，我们只能得到适用于固定 h 的 $K_{3,h}$-子式自由图的算法，而无法得到适用于任意固定 H 的 H-子式自由图的算法。能否将这些结果推广到 H-子式自由图，仍是一个值得探讨的开放问题

采用某种针对具体问题的方法，或者更理想地，通过对该框架作适当推广。然而，很容易找到一些（人为构造的）问题，在这些问题中，定理 1 31..3 可用于任意 h 的 $K_{3,h}$-子式自由图，但该问题在 K_6- 子式自由图上是 NP-hard。例如，若有一种算法能够在满足 $k=1$ 的 K_6- 子式自由图中解决这样的问题：“找出一个包含 k 个顶点的集合 S，使得每个顶点的度都恰好为 d，且 S 的闭邻域所诱导的图是 4-可着色图”，那么就可以用它来测试平面图是否为 3-可着色图。[1]

子图同构定理 1 11..1 被 Fomin 等人 [26] 用于判定连通有界度图 P 是否为不含顶点 minor 的图 G 的子图，运行时间为 $2^{\tilde{\mathcal{O}}(\sqrt{|V(P)|})}|V(G)|^{\mathcal{O}(1)}$。证明利用了这样一个事实：如果给定 G 的宽度为 t 的树分解~，那么对于连通有界度 P，该问题可在 $|V(P)|^{\mathcal{O}(t)}|V(G)|^{\mathcal{O}(1)}$时间内求解。如果给定一个宽度为 t_1 的树分解，并保证存在一个解，使得每个袋中包含的模式 P t_2 顶点不超过 t_2，那么很容易修改该算法~，使其运行时间为 $(|V(P)|+t_1)^{\mathcal{O}(t_2)}\cdot|V(G)|^{\mathcal{O}(1)}$。因此，定理 1 21..2 将这一结果（及其在带颜色、有向、加权图上的扩展）推广到了不含 H-子式的图。

连通且度有界这一要求不能去除：Bodlaender 等人 [7] 证明，对于连通、度无界的 P，或非连通、度有界的 P，若存在一个运行时间为 $2^{o(|V(P)|/\log|V(P)|)}|V(G)|^{\mathcal{O}(1)}$ 的算法，就会违背指数时间假设 [30]。不过，有一种值得关注的非连通情形：模式的每个连通分量都相同，换句话说，目标是寻找与固定模式 P 同构的 k 个顶点不相交子图。对于固定的 P，该问题可在一般图中用 $2^{\mathcal{O}_P(k)}|V(G)|^{\mathcal{O}_P(1)}$ 时间解决，例如使用颜色编码 [14] 技术；给定一个宽度为 t 的 G 的树分解时，也可在 $t^{\mathcal{O}_P(t)}|V(G)|^{\mathcal{O}_P(1)}$ 时间内解决。由于定理 1 21..2 不再要求模式是连通的，因此可用它在 H-子式自由图中以 $2^{\tilde{\mathcal{O}}_{P,H}(\sqrt{k})}|V(G)|^{\mathcal{O}_{P,H}(1)}$ 的时间解决该问题。

斯坦纳树 给定一个图 G，其终端点集合为 $T\subseteq V(G)$，斯坦纳树问题要求求出一棵包含 T 中每个顶点的最小规模树 G_T。该问题可以自然地扩展到边加权图或点加权图。如果以终端点数量 $|T|$ 为参数，那么假设 ETH 成立，即使在平面图上，也不存在亚指数时间的参数化算法，也就是说，不存在运行时间为 $2^{o(|T|)}|V(G)|^{\mathcal{O}(1)}$ [41] 的算法。然而，如果以解的规模 G_T 为参数，则可以利用定理 1 11..1，在顶点极小子式自由图中得到一个 $2^{\tilde{\mathcal{O}}(\sqrt{|V(G_T)|})}|V(G)|^{\mathcal{O}(1)}$ 时间算法。我们的定理 1 21..2 将这一结果推广到了 H-子式自由图。

由于终端点集 T 始终是解树 G_T的一部分，斯坦纳树问题的另一个自然参数化方式是以非终端顶点的数量为参数，即以$\nu:=|V(G_T)-T|$为参数。我们可以利用定理 1 31..3 和 $d=1$，在 $K_{3,h}$-子式自由图上得到使用该参数的次指数参数化算法。不失一般性地，假设 T 是一个独立集，因为可以收缩 $G[T]$ 的任意边而不改变问题的答案。于是，关键观察是（如果 $|T|\geqslant 2$），若 X 是解中的非终端顶点集，则整个树 G_T 都包含于 $N_G[X]$中。因此，定理 1 31..3 与 $d=1$ 相结合，给出了一个宽度关于 ν 为多项式的树分解，使得每个~袋至多包含 $\mathcal{O}(\sqrt{\nu})$个 $N_G[X]$的顶点，从而也至多包含 G_T的顶点。这使我们能够得到一个~$2^{\mathcal{O}(\sqrt{\nu})}\cdot|V(G)|^{\mathcal{O}(1)}$-时间算法。这重新证明了 [45, 49]的结果。

[1] 给定一个平面图 G，通过引入一个全邻接顶点，构造 K_6-子式自由图 G'。在 G' 上调用该算法，并使用 $k=1$ 和 $d=|V(G)|$。

不相交路径 在不相交路径问题中，给定一个图 G ，其中有 k 对终端顶点（s_1, t_2）、……、（s_k, t_k），任务是寻找 k 条两两顶点不相交路径 P_1、……、 P_k，使得 P_i 连接 s_i 和 t_i。Robertson 和 Seymour [47] 的著名结果表明，该问题以终端点对的数量 k 为参数时属于固定参数 tractable（FPT）（另见 [32, 35, 37]）。然而，带长度约束的版本 [1,4–6,36,46,48] 是否也能以 k为参数实现固定参数 tractable，仍是一个重要的未决问题；例如，我们希望最小化路径的总长度 ℓ 。如果考虑以总长度 ℓ为参数这一更温和的问题，那么可以使用颜色编码等标准技术 [14] 来获得固定参数 tractable 算法。此外，我们的定理 1 21..2 给出了在 H-子式自由图上的 $2^{\widetilde{\mathcal{O}}(\sqrt{\ell})}|V(G)|^{\mathcal{O}(1)}$ 时间算法：解由 $\mathcal{O}(\ell)$个顶点组成，而带有各种长度约束的不相交路径问题可以利用标准技术在有界树宽图上求解。由于解不一定诱导出一个连通图，因此定理 1 21..2 与定理 1 11..1不同，并不要求连通性；这一点至关重要。

最稠密子图 给定一个图 G 和整数 k，最稠密子图问题要求找出一个集合S，其中包含 k 个顶点，使得 $G[S]$ 包含的边数最多。在一般图上，该问题显然至少与 k-团一样难。由于给定宽度为 t的树分解后，可以在时间 $2^{\mathcal{O}(t)}|V(G)|^{\mathcal{O}(1)}$ 内解决该问题，定理 1 21。.2 可用于为该问题在 H-子式自由图上构造亚指数参数化算法。请注意，尽管该问题要求寻找“稠密”子图，但没有理由假设最优 $G[S]$ 是连通的。因此，定理 1 21。.2 不要求连通性这一点再次至关重要。

部分覆盖问题 优化问题的部分版本要求寻找一个解集S ，其大小为 k ，它可能无法满足所有约束，但会使满足的约束数量最大化。这种推广会显著增加问题的难度：例如，部分顶点覆盖问题（寻找 k 个顶点，使其覆盖至少 ℓ条边）以 [1]为参数时是 W-难的。然而，该问题在平面图和 k-子式自由图上变为固定参数可 tractable，甚至存在亚指数参数化算法 [2,29]。我们指出，定理 1 21。.2 也可用于构造此类算法，因为使用标准技术即可在树分解上解决该问题（例如，注意由 S 覆盖的边数等于 S的总度数减去 $G[S]$中的边数，因此可以采用类似于最稠密子图问题的技术）。该算法的优点是易于推广到其他问题变体，例如顶点/边带有权重或颜色等情形。

可以将部分顶点覆盖问题推广如下：我们不再要求覆盖尽可能多的边，而是希望覆盖尽可能多的大小为 c 的团；显然， $c = 2$ 与部分顶点覆盖问题相同。对于 $c > 2$，仅凭 K_c无法确定被 S 覆盖的 $G[S]$ 子图数量，但可以通过 $G[N_G[S]]$ 来确定。因此，对于 $K_{3,\ h}$-子式自由图，我们可以利用定理 1 31。.3 以及 $d = 1$ 构造一个树分解，使得每个袋都包含 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个 $N_G[S]$ 的顶点。在这样的树分解上，我们可以使用标准技术，在 $2^{\widetilde{\mathcal{O}}(\sqrt{k})}|V(G)|^{\mathcal{O}(1)}$ 时间内求解该问题。

另一个研究充分的部分优化问题是部分支配集问题，其任务是找到一个包含 S 个 k 顶点的集合，使得闭邻域 $N_G[S]$的大小至少为 ℓ [18,29]。另一个自然的推广是，对于某个固定的 d ，最大化距离- $N_G^d[S]$邻域的大小。对于固定的 d，在 $K_{3,\ h}$-子式自由图上，定理 1 31..3 可用于构造一个$2^{\widetilde{\mathcal{O}}_{d,\ h}(\sqrt{k})}|V(G)|^{\mathcal{O}(d,h)}$-时间算法。

值得指出的是，这些结果所采用的稳健方法可以结合不同类型的问题。例如，我们可以考虑寻找长度为 k 的环，使其支配至少 ℓ 个顶点；或者寻找 k 个互不相交的三角形，使覆盖的边数最多，等等。

距离约束 我们的主要贡献之一是，定理 1 31。.3 对于 $K_{3,h}$-子式自由图，使我们能够表达涉及距离约束的问题。将（部分）支配集推广到其距离-d 版本，就是一个例子。不过，在其他问题中考虑距离约束也很自然，例如：

- 寻找 G 中固定模式 P 的 k 个副本，使它们两两之间的距离至少为 d。
- 寻找总长度为 k 的顶点不相交路径，连接给定的终端点对，并使这些路径彼此之间的距离至少为 d 。
- 寻找一条长度恰为 k 的环，且不存在长度至多为 d、使用环外的边连接环上两个顶点的“捷径”。

对于这类问题，如果 S 是大小为 k的解，那么仅凭 $G[S]$无法验证该解是否有效：我们需要知道$G[N_G^d[S]]$ 才能验证解的有效性。因此，正确理解这个问题的方式如下：

1. 我们要寻找两个集合：一个大小为 k的集合 S ，以及一个大小无界的集合 S^* 。2. 我们要寻找一个赋值 δ： $S^* \to \{0, 1, \ldots, d\}$ ，使得 $\delta(v)$是 v到 S的距离。3. 我们要验证 S^* 恰好是 $N_G^d[S]$。4. 我们要验证 $G[S^*]$ 表明 S 确实是一个正确的解。

定理 1 31..3 给出了一个集合 Z ，满足 $S^* \subseteq Z$ ，以及 $G[Z]$ 的一个树分解，使得每个袋都包含 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个 S^* 的顶点。如果我们找到了集合 S 和 S^*，以及赋值 δ，那么就可以验证 S^* 是否确实是 $N_G^d[S]$（特别是，我们需要验证：如果 $\delta(v) < d$，那么 v 在 S^* 之外没有邻点）。接下来，例行验证即可确认 S 满足要求。

地图图 平面图的一种广为研究的推广是地图图[11], ，它们是平面中有限个单连通且内部互不相交区域的交图。描述地图图的一种方法是取一个平面图 G 和一个 N 面的集合；地图图 M 的顶点集为 N ，当对应的两个面在 N 中共享一个顶点时，这两个顶点相邻。尽管地图图可以包含任意大的团，但平面图的一些算法结果可以推广到地图图 [9–11, 18, 24, 25, 28]。

给定 G 和 N 如上所述，令 G' 为二部平面图，其中一侧为 N，另一侧为 G的顶点集合，而 N 中的一个面在 G' 中与其所有顶点相邻。于是x， $y \in N$ 在 M 中相邻，当且仅当它们在平面图 G'中的距离为二。利用这一等价性，关于 M 的算法问题可以改写为关于平面图 G'的问题。特别地，可以从图 $M[S]$ 确定图 $G'[N_{G'}[S]]$ （更一般地，$M[N_M^d[S]]$ 可以从 $G'[N_{G'}^{2d+1}[S]]$确定）。因此，定理 1 31。.3 使我们也能在地图图中得到上述所有算法结果。

2 概述

本节将简要概述本文的主要结果和技术。本文的主要技术内容分为三个部分：

- 为了处理 $d \geqslant 1$ 这种情况，我们需要对 Fomin 等人的对偶结果进行大幅改进， [27], 并考虑模式的 d-邻域（第 4 节4）。

- 我们提出一种略显技术性的分隔集改进过程，并在主算法中反复使用（第 5 节5）。

- 我们首先针对“带顶点的近可嵌入图”介绍主算法（第 6161..节），然后将其推广到 H-子式自由图（第 6262..节），以及推广到 $K_{3,h}$-子式自由图（第 6363..节）。

尽管我们用同一个证明证明定理 1211..2 和 1 311..3，但我们先介绍定理 1 211..2 证明背后的思路，然后讨论将证明推广到定理 1 3 $d \geqslant 1$ 情形13..所面临的一些挑战。

2.1 对偶结果

Fomin 等人开发（Nederlof 也使用）的一项主要工具是以下对偶定理：

定理 2.1 （ [27]中的定理 9）。 存在一个多项式时间算法，给定一个连通图 G、一对不同顶点 $s, t \in V(G)$ 以及正整数 p, q，输出以下结构之一于 G：

- 一个（$C_1, \ldots, C_p$） 分隔集链，由（s, t）分隔集组成，其中 $|C_j| \leqslant 2q$ 对每个 $j \in [p]$都成立；或者
- 一个（$P_1, \ldots, P_q$） 路径序列，由（s, t）路径组成，使得对每个 $i \in [q]$， P_i中至多有 $4p$ 个顶点位于其他路径 P_j, $j \in [q] - \{i\}$上。

第一种情况下，($C_1, \ldots, C_p$)是分隔集链，意为每个 C_i 都将$\{s\} \cup C_1 \cup \ldots \cup C_{i-1}$ 与 $C_{i+1} \cup \ldots \cup C_p \cup \{t\}$分隔开。在第二种情况下， $P_i - \{s, t\}$中出现在其他路径 P_j 上的顶点称为公共顶点，其余所有顶点称为私有顶点。顶点 s 和 t 以特殊方式处理，既不是私有顶点，也不是公共顶点。

引理 2 112..1 中的一种典型用法是 [27, 43], ：当宿主图中隐藏着一个至多有 k个顶点的未知模式图时，令 $p \sim \sqrt{k}$ 且 $q = \text{poly}(k)$。在第一种情况下，某个分隔集 C_j 与模式的交集会很小（大小至多为 $\sqrt{k}$），并且当$|C_j|$ 在 k中以多项式为界时，可以从次指数数量的选择中猜出该交集。在第二种情况下，随机选取的路径 P_i 以高概率仅在公共顶点处与模式相交；通过缩减私有顶点（例如删除或收缩），可将 P_i 的长度降至至多 $4p \sim 4\sqrt{k}$。

我们将定理 2 112..1 改写为以下形式，以便与后文提出的距离版本保持一致。首先，我们强调存在一个与未知集合 Z交集很小的分隔集／路径。其次，作为一项技术性推广，我们还希望它与已知集合 Z_0的交集也很小。关键在于，未知集合 Z 会$|Z|$ 对 q的下界作出贡献，而已知集合 Z_0 仅贡献 $|Z_0|/p$。

推论 2.2. 存在一个多项式时间算法，给定一个图 G、一对不同的顶点 $s, t \in V(G)$、一个大小为 k_0 的集合 $Z_0 \subseteq V(G)$，以及满足 $q \geqslant k + k_0/p + 1$ 的正整数 $p, q \geqslant 2, k$，输出以下结构之一于 G 中：

- 一条链 $(C_1, \ldots, C_p)$，由 G 中大小至多为 $2q$ 的 (s, t)-分隔集组成，使得对于任意大小至多为 k 的 $Z \subseteq V(G)$，都存在一个 $j \in [p]$，满足 $|Z_0 \cap C_j| \leqslant 2k_0/p$ 且 $|Z \cap C_j| \leqslant 2k/p$。
- 一组$(P_1, \ldots, P_q)$ 条 G 中的(s, t)-路径，以及一组$(Q_1, \ldots, Q_q)$ 大小至多为 $5p + 2$ 的集合，满足 $Q_i \subseteq V(P_i)$；对于任意大小至多为 k的 $Z \subseteq V(G)$，都存在一个$j \in [q]$，使得$(Z_0 \cup Z) \cap (V(P_j) - Q_j) = \emptyset$。

证明。将定理 2 12..1 应用于 G、 s、 t、 p和 q。如果得到一条链$(C_1, \ldots, C_p)$，就将其作为第一个结果返回：由于它们两两不交，少于 $p/2$ 个分隔集含有超过$2k_0/p$ 个 Z_0 的元素，且少于 $p/2$个分隔集含有超过 $2k/p$ 个 Z的元素。如果返回的是一个序列$(P_1, \ldots, P_q)$，则连同 Q_i一起返回，其中 Q_i等于 $\{s,t\}$ 加上 P_i 的公共顶点集合，再加上至多 p 个属于 Z_0的 P_i 私有顶点。由于$q \geqslant k + k_0/p + 1$，存在 $i \in [q]$ ，使得 P_i 中没有私有顶点属于 Z ，且 P_i 中至多有 p个私有顶点属于 Z_0，因此 $V(P_i) \cap (Z_0 \cup Z) \subseteq Q_i$，如所需。 □

2.2 转向稀疏分离

Nederlof 的算法中的一个关键步骤是[43] ，如果我们有一个平衡分隔集，且它与一个大小为 Z k 的未知解有较大的交集，那么通过准多项式(k) 次猜测，我们可以转向一个与 $\widetilde{\mathcal{O}}(\sqrt{k})$ 只有 Z交集的平衡分隔集。Nederlof 的算法 [43] 在略有不同的情形下多次使用了本质相同的论证。我们的主要结果也采用了类似的论证。我们提出一个抽象论述，以涵盖该论证的所有必要用法，并证明它适用于 H-子式自由图，以及针对 $d \geqslant 1$ 上的 $K_{3,\ h}$-子式自由图的一个推广（定理 5 151。。）。由于 $d \geqslant 1$ 情形涉及许多额外的技术细节，这里我们给出一个与 $d = 0$相关的简化版本。

回顾一下， G的一个分离是指一对顶点子集（A， B），满足 $V(G) = A \cup B$，$A - B \neq \emptyset$，并且 G与 $A - B$和 $B - A$之间没有边。分离（A， B）的阶为 $|A \cap B|$。如前所述，一个关键目标是找到大小为 poly(k) 的分隔集，使其与解 Z的交集仅包含 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点。但由于我们要寻找 G 的一个诱导子图的树分解，如果能找到一个由两组顶点构成的分隔集也会很有帮助：一组“稀疏”部分，大小为 poly(k)，与解 Z 的交集仅包含 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点；另一组“无关”部分，大小无界，但与 Z不相交。如果我们有这样一个分隔集及其划分，就可以从图中移除第二部分。

定理 2.3. 对于每个固定的整数 $h \geqslant 4$，都存在一种随机化多项式时间算法，该算法以一个 K_h-子式自由图 G、一个整数 $k \geqslant 2$、G 中的一个分离(A°, B°) 以及 $G[A^\circ \cap B^\circ]$ 的一个生成森林 F° 作为输入，其中该生成森林的连通分量少于 h 个且最大度至多为 h，并从 $V(G)$ 的子集中抽取一个元组(A, B, C)，使其满足以下性质：

1. (A, B) 是 G 中的一个分离，
2. $C \subseteq A \cap B$，3. $|C| \leqslant |A^\circ \cap B^\circ| + \mathcal{O}_h(k \log k)$，
4. 对于每个正整数 θ，集合 $W \subseteq V(G)$，以及一个集合 $Z \subseteq V(G)$，使得
   - $|W| \geqslant 4\theta$，
   - $|A^\circ \cap W| \geqslant \theta$、 $|B^\circ \cap W| \geqslant \theta$，以及
   - $|Z| \leqslant k$，

   以（$\mathcal{O}_h$（$k|A^\circ \cap B^\circ|$））$^{-\mathcal{O}_h}$（对数 k）的概率，我们有如下结论：
   - $|A \cap W| \geqslant \theta$、 $|B \cap W| \geqslant \theta$，
   - $A \cap B \cap Z \subseteq C$，
   - $|A \cap B \cap Z| = |C \cap Z|$ 由 $\mathcal{O}_h$（$\sqrt{k}$ 对数 k）界定，
   - 此外，若 $|Z \cap A^\circ \cap B^\circ| > h(h+1)\sqrt{k}$，则$^{-}|Z \cap A \cap B| \geqslant \sqrt{k}$。

让我们试着解析这个陈述。Nederlof [43] 只考虑平面图中的环分隔集，也就是说， $G[A^\circ \cap B^\circ]$ 具有一个 2-正则生成子图。这个环的存在表明，

例如，可以将分隔集划分为多个连通部分，使每个部分包含数量大致相同的解顶点。在我们的推广中，我们用一个更抽象的性质取代对环分隔集的要求：一个连通分量数量有界且度有界的生成森林。这样的森林的存在表明了类似的性质：每棵树都可以划分为多个部分，使不同部分中的解顶点数量相差至多一个常数因子（或者是 F°的连通分量之一，但这样的连通分量只有常数个，不会造成问题）。这样的分隔集及其合适的生成树可以通过 [22]的乘积结构理论结果得到。

给定 G 的一个分离 (A°, B°)，其阶为 ℓ。我们假设，关于某个大小任意且未知的 W，该分离是平衡的（即 W 中至少有 1/4 位于 A° 和 B° 中）。在不知道 W 或大小至多为 k 的解 Z 的情况下，我们需要以概率 $(k\ell)^{-\mathcal{O}(\log k)}$ 采样一个分隔集 (A, B)，使得 (A, B) 关于 W 仍满足相同的平衡要求，但它与未知解 Z 的交集大小仅为 $\mathcal{O}(\sqrt{k}\log k)$。更准确地说，$(A, B)$ 的阶可以任意大，但除了 (A, B) 之外，我们还需要输出一个猜测 $C \subseteq A \cap B$，覆盖 $Z \cap A \cap B$ 中的每个顶点；这会将集合 $A \cap B$ 划分为稀疏部分 C 和无关部分 $(A \cap B) - C$。该猜测的大小 $|C|$ 至多为 (A°, B°) 的阶 ℓ 加上 $\mathcal{O}(k \log k)$。注意，(A, B) 的阶没有上界，但 $(A \cap B) - C$ 中的顶点不属于解，因此它们无关，可以从图中删除。

我们为定理 2 32 3.. 给出一个非常稳健的证明，不需要使用任何拓扑论证：我们的算法要么按要求采样 (A, B)，要么在 G 中展示一个 H-子式。为简单起见，假设 $A^\circ \cap B^\circ$ 的生成森林可以划分为 h 个连通集 $J_1, \ldots, J_h$，且每个集合与 Z 的交集所含顶点数大致相同。我们可以通过猜测生成树中的 $h-1$ 条边来猜出这个划分（也就是说，有 $(\ell-1)^{h-1}$ 种可能）。我们猜测 $|A \cap W|$ 和 $|B \cap W|$ 中哪一个更大（有两种可能）。因此，在下文中，不失一般性地假设 $|A \cap W|$ 更大，因而至少为 $|W|/2$。

我们的目标是猜出某个分隔集 S，它能将两个集合 J_i 和 J_j 分开，其中 $i, j \in [h]$。为简化记号，我们尝试为 $i = 1$ 和 $j = h$ 寻找这样的分隔集：我们希望找到一个集合 S，它是 $G -$ 中的一个 $J_1 - J_h$ 分隔集（见图11），大小为 poly(k)，且满足 $|S \cap Z| = \widetilde{\mathcal{O}}(\sqrt{k})$。如果有这样一个集合 S，就意味着 G 存在一个分离 (A', B')，满足 $A' \cap B' = (J_2 \cup \cdots \cup J_{h-1}) \cup S$, $J_1 \subseteq A'$, $J_h \subseteq B'$。我们猜测集合 J_1 和 J_h 中哪一个位于与 $A \cap W$ 交集较大的那一侧（有两种可能）；不失一般性地，假设 $|A' \cap (A \cap W)| \geqslant |A \cap W|/2 \geqslant |W|/4$。现在，我们解交叉这两个分离 (A, B) 和 (A', B')，并通过 $A'' = A \cap A'$ 和 $B'' = B \cup B'$ 定义分离 (A'', B'')。注意，我们有 $A'' \cap B'' \subseteq ($
$J_1 \cup J_2 \cup \cdots \cup J_{h-1}) \cup S$。我们观察到：

- $|A'' \cap W| = |A' \cap (A \cap W)| \geqslant |W|/4$，因为我们假设 A' 包含了 $A \cap W$ 中至少一半的顶点。
- $|B'' \cap W| \geqslant |W|/4$，因为根据假设，我们有 $B \subseteq B''$ 和 $|B \cap W| \geqslant |W/4|$。
- 从 $(A,\ B)$ 移至 $(A'',\ B'')$ 时，我们从分隔集中移除了 J_h（该分隔集按设想应包含与 Z 的交集中约 $1/h$ 的部分），代价是引入了 S _(按设想，它只包含 Z中的 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点）。除非该交集的大小已经是 $\widetilde{\mathcal{O}}(\sqrt{k})$，否则这会使与 Z 的交集缩小一个常数因子。

因此，我们可以对这个新的分离 $(A'',\ B'')$ 重复这一过程。经过 $\mathcal{O}(\log k)$ 次迭代后，交集大小降至 $\widetilde{\mathcal{O}}(\sqrt{k})$。通过反复引入集合 S，分离的阶总共增加 poly(k)，而与 Z 的交集增加 $\widetilde{\mathcal{O}}(\sqrt{k})$。

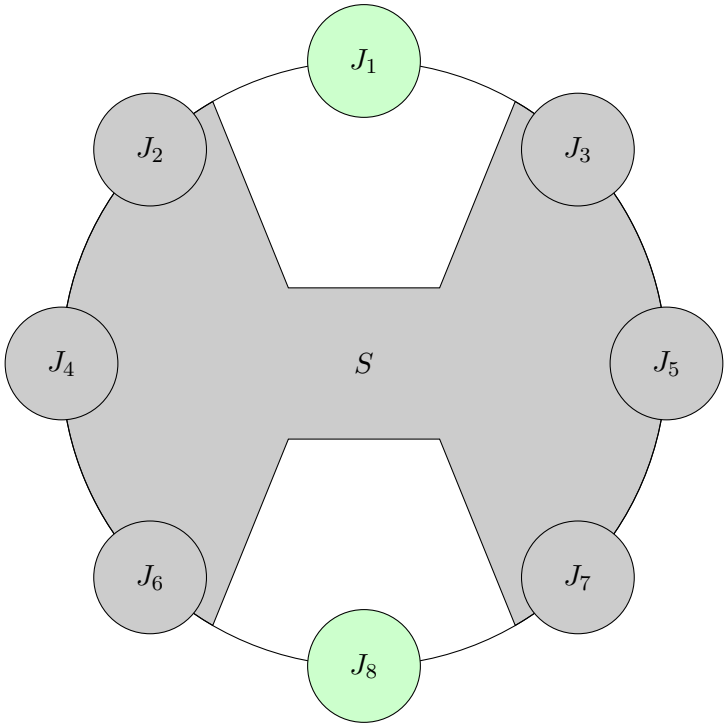

图1：分离两个集合 J_1 和 J_h。

但我们如何找到所需的分隔集，使其与 Z的交集为 $\widetilde{\mathcal{O}}(\sqrt{k})$ 呢? 这里就用到了推论 2 22..2 的对偶结果。基本思路是，对于每个$i,j \in [h]$，我们在由 G 得到的图上应用推论 2 22..2：将集合 J_i 收缩为单个顶点 s，将集合 J_j 收缩为单个顶点 t，并移除所有其他集合 J_s。如果对于某个 $i,j \in [h]$，推论 2 22..2 返回一个互不相交的分隔集序列，那么其中一个与 Z的交集大小为 $\widetilde{\mathcal{O}}(\sqrt{k})$，我们便猜测这个分隔集。如果对于每个 $i,j \in [h]$，我们都得到一条路径 $P_{i,j}$，那么我们就尝试利用这些路径证明该图包含一个 K_h-子式。如果这些路径两两不交，那么将每个 J_i 收缩为单个顶点，并将每条路径收缩为单条边，显然就能得到一个 K_h-子式。然而，这些路径不一定互不相交，因此我们必须以更谨慎的迭代方式运用这一论证。

我们按任意顺序遍历这些点对 $(i,j) \in \binom{h}{2}$。如果推论 2 22..2 给出一条路径 $P_{i,j}$，那么我们就从图中移除这条路径的内部顶点，并在剩余图中继续处理下一个点对 (i,j)。这样一来，收集到的路径确实会两两不交；因此，如果这次迭代遍历了每个点对 (i,j)，那么由此得到的连通集 $J_1,\ldots,J_h$ 以及两两不交的路径 $\binom{h}{2}$ 将在图中构成一个 K_h-子式。然而，如果迭代在某个点对 (i,j) 处停止，并得到一个分隔集 S，那么 S 仅是剩余图中的分隔集。换言之，只有将迭代过程中移除的所有路径 $P_{i,j}$ 的并集 P 加入其中，S 才是原图中的分隔集。不幸的是，这些路径的长度没有上界。但请回想，推论 2 22..2 中的每条路径 $P_{i,j}$ 都配有一个大小为 $\widetilde{\mathcal{O}}(\sqrt{k})$ 的集合 $Q_{i,j}$；推论 2 22..2 不仅指出，对于某条路径 $P_{i,j}$，其与 Z 的交集有界，还指出 $P_{i,j}$ 仅在 $Q_{i,j}$ 中与 Z 相交。因此，如果我们在迭代至多 $\binom{h}{2}$ 步的每一步都选择这样一条路径，那么可以假设这些路径的并集 P 与 Z 的交集仅包含所有 $Q_{i,j}$ 的并集 Q，而该并集的大小为 $\widetilde{\mathcal{O}}(h^2\sqrt{k})$。这意味着 $S \cup P$ 正是我们需要的那类分隔集：它包含一个大小为 poly(k) 的稀疏部分 $S \cup Q$，其中包含 Z 的 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点；此外还有一个与 Z 不相交、大小无界的部分 $P - Q$。因此，

我们可以按照上述方式，继续使用这个分隔集 $P \cup S \quad J_i - J_j$。

2.3 主算法

在本概述中，我们描述定理 1 1..2 中针对有界亏格图的算法，它已经展示了主要的技术思想。此外，我们假设图的直径为 $\mathcal{O}(k)$：通过标准地应用 Baker 分层方法，我们可以作出这一假设。已知，若一个有界亏格图的直径为 $\mathcal{O}(k)$，则其树宽为 $\mathcal{O}(k)$，因此存在阶为 $\mathcal{O}(k)$的平衡分离。但为了能够使用定理 2 2..3，我们需要能够找到这样的分隔集：它们具有有界度生成树，且这些生成树仅由有界数量的连通分量构成。根据 Dujmović 等人的乘积结构定理（一个合适的变体），我们推得连通有界亏格图（更一般地说，近似可嵌入不带顶点的曲面中的图）存在这样的平衡分隔集。 [22]

推论 2.4。 对于每个 $g \geqslant 0$，都存在一个常数 c，使得以下结论成立：存在一个多项式时间算法，给定一个亏格至多为 g、直径为 Δ 的图以及一个权重函数 $\mu\colon V(G) \to \mathbb{R}_{\geqslant 0}$，该算法可以计算出一个相对于 μ 的平衡分隔集 X，其阶为 G，以及一个 $G[X]$ 的生成森林 F，使得

1. $|X| \leqslant c^2 \cdot (\Delta + 1)$，2. F 至多有 c 个连通分量，且

3. F 的最大度至多为 $3c^2 - 1$。

在定理 1 2 1..2中，我们需要构造一个宽度为 poly（k）的诱导子图 G' 的 $\underline{G}$ 树分解，使得 G' 完全包含 Z，并且每个袋都包含 $\widetilde{\mathcal{O}}$（$\sqrt{k}$）个 Z的顶点。从概念上说，可以更容易地这样理解这一目标：我们想要找到原图 G 本身的一个树分解，其宽度没有上限，但我们需要将每个袋划分为“有用”顶点和“禁用”顶点，使得没有解顶点被标记为禁用，每个袋中标记为有用的顶点至多有 poly（k）个，并且每个袋只包含 $\widetilde{\mathcal{O}}$（$\sqrt{k}$）个解中的顶点。然后，我们只需从任意袋中移除所有被标记为禁用的顶点，就能得到G' 及其树分解。

主算法的基本思路是执行一个递归过程，尝试寻找图 G 的一个平衡分离 (A, B)，递归地为 $G[A]$ 和 $G[B]$ 分别寻找树分解，然后将这两个分解组合起来，得到 G 的树分解。由于 (A, B) 相对于 G 的顶点数是平衡的，递归深度为 $\mathcal{O}(\log n)$。

为了能够将 $G[A]$ 的分解与 $G[B]$ 的分解组合起来，这类算法通常会求解一个更一般的问题：输入中给定一个集合 $R \subseteq V(G)$，并且分解的根袋应包含 R。我们还要求不能将 R 中的任何顶点设为禁用。如果在递归构造 $G[A]$ 和 $G[B]$ 的分解时，将 $A \cap B$ 并入 R，那么这两个分解就可以组合起来。然而，这样一来，每次递归调用中 R 的规模（因此袋的规模也）都会增大：即使 $A \cap B$ 仅为 poly(k)，集合 R 最终也可能达到 $\mathcal{O}(\text{poly}(k) \cdot \log n)$ 的规模。解决这个问题的自然思路是加入一些递归步骤，在这些步骤中，选择 (A, B) 作为相对于 R 而非 $V(G)$ 的平衡分隔集。如果足够频繁地加入此类步骤，那么 R 就不会无限增大。

由于我们希望构造一种树分解，使得每个袋与解 Z中的 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点相交，因此集合 R 以及分隔集 $A \cap B$ 必须满足

这一性质。该算法的主要挑战在于确保我们考虑的每个分离 (A, B) 都以高概率具有这一性质。我们尝试通过以下方式确保这一点。首先，我们使用推论 2 42..4，得到一个平衡分离 (A, B)（相对于 $|V(G) - R|$ 或 R）。然后，以概率 $1 - 1/k$，我们猜测 $|(Z - R) \cap A \cap B| \leqslant \widetilde{\mathcal{O}}(\sqrt{k})$，因此可以如上所述直接使用分离 (A, B)。否则，以概率 $1/k$，我们使用定理 2 32..3，以 $k^{-\mathcal{O}(\log k)}$ 的成功概率增强 (A, B)，并使用这个增强后的分离继续递归。回想一下，在定理 2 32..3 中，采样分离 (A, B) 附带一个大小为 $\mathrm{poly}(k)$ 的集合 C，该集合应该覆盖 $A \cap B$ 中所有的解顶点。因此，我们可以使用 C 将 $A \cap B$ 的一些顶点标记为有用，并将其余顶点标记为禁用。

不过，上述方法有一个缺陷。虽然偶尔使用相对于 R 的平衡分隔集可以控制递归中 R 的规模，但无法充分控制 $Z \cap R$ 的规模。即使每一步只向 R 中加入 Z 的 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个顶点，经过 $\Theta(\log n)$ 层深度后，它们也会累积到 $\log n \cdot \widetilde{\mathcal{O}}(\sqrt{k})$，这实在太多了。我们通过以下方式解决这个问题。偶尔，我们会进入“模式模式”，以概率 $\frac{1}{k}$ 寻找一个分离 (A, B)，使算法未知的集合 $Z \cap R$ 得到平衡分离。为此，我们注意到，首先，我们可以借助 [22] 的工具，从仅有的 $\mathcal{O}(|R|)$ 种选择中采样初始分离 (A, B)；其次，我们可以利用定理 2 32 3.. 在不知道 W 和 θ 的情况下也能发挥作用这一事实来改进它。因此，根据当前的递归深度，算法以轮流方式在三种模式之间循环：

1. 寻找一个关于顶点数平衡的分离（以保持最大深度为 $\mathcal{O}(\log n)$），
2. 寻找一个关于 R 平衡的分离（以保持 R 的大小有界），or
3. 进入模式模式（以保持交集 $R \cap Z$ 有界）。

我们来尝试分析该方案生成所需树分解的概率。算法可能会因错误猜测 $|(Z-R) \cap A \cap B| = \widetilde{\mathcal{O}}(\sqrt{k})$是否成立，以及错误地从定理 2 32..3提供的列表中选择分离（即选择与 Z不兼容的分离）而犯错。为了分析出错的概率，我们可以将递归调用分为三类：

无效： $V(G) - R$与 Z不相交。

子式： $V(G) - R$与 Z相交，但其中包含的 Z顶点至多为 $\sqrt{k}$ 个。

主要： $V(G) - R$与 Z 不相交，且包含的 Z顶点超过 $\sqrt{k}$ 个。

重要的技巧在于，对于无效类型的递归调用，不会出错：只要每个袋中只有 $\mathrm{poly}(k)$个有用顶点，且根袋包含 R，那么任何树分解都是正确的。因此，无论我们是否决定调用定理 2 32。.3，也无论选择哪种分离，该算法成功的概率都是 1。

注意到，如果一次递归调用无效，那么它的所有后代递归调用也都无效。换言之，子式递归调用和主要递归调用构成递归树的一个前缀。此外，在递归过程中使用分离 (A, B) 时，两次递归调用中的集合 $V(G) - R$互不相交：一个是 $A - B$ 的子集，另一个是 $B - A$ 的子集。因此，考察非无效递归调用前缀的叶子时，集合$V(G) - R$ 两两不交。这意味着其中至多有 $|Z| = k$ 个集合能与 Z 有非空交集，因此该前缀至多有 k 个叶子。由于递归树的深度为 $\mathcal{O}(\log n)$，可知子式递归调用和主要递归调用共有 $\mathcal{O}(k \log n)$ 次。

在每次子式或主要调用中，我们都希望正确猜出 $|(Z-R)\cap A\cap B|\leqslant\widetilde{\mathcal{O}}(\sqrt{k})$。回想一下，我们以概率 $1-1/k$ 进入这种情况；由于子式和主要递归调用共有 $\mathcal{O}(k\log n)$ 次，因此其概率为

$$(1-1/k)^{\mathcal{O}(k\log n)}=n^{-\mathcal{O}(1)}$$

我们在每次子式调用和主要调用中都正确猜测了 $|(Z-R)\cap A\cap B|\leqslant\widetilde{\mathcal{O}}(\sqrt{k})$，若情况确实如此。接下来，我们使用推论 2 4$(A,\ B)$ 给出的分离2。.4，而不调用定理 2 323。。因此，这些递归调用的行为是正确的。

对于其他递归调用，如果 $|(Z-R)\cap A\cap B|>\widetilde{\mathcal{O}}(\sqrt{k})$，那么该调用必须是主要调用。在这种情况下，定理 2 32..3 保证 $A\cap B$ 中包含 $\Omega(\sqrt{k})$ 个 $Z-R$ 的顶点。这些顶点会被移入 R，因此这一步会“消耗” $Z-R$ 中 $\Omega(\sqrt{k})$ 个顶点。也就是说，如果这样一次针对 (G,R) 的“丰富”递归调用创建了两个实例 (G_1,R_1) 和 (G_2,R_2)，那么 $V(G_1)-R_1$ 和 $V(G_2)-R_2$ 不相交，因此 $\sum_{i\in[2]}|(V(G_i)-R_i)\cap Z|\leqslant|(V(G)-R)\cap Z|-\Omega(\sqrt{k})$。由此可知，这样的递归调用最多只有 $\mathcal{O}(\sqrt{k})$ 次。因此，在每次递归调用中，我们都从定理 2 32..3 返回的、大小为 $k^{\mathcal{O}(\log k)}$ 的列表中选到合适分离的概率为

$$\left(k^{-\mathcal{O}(\log k)}\right)^{\mathcal{O}(\sqrt{k})}=2^{-\widetilde{\mathcal{O}}(\sqrt{k})},$$

符合要求。

剩下要界定“模式模式”正确运行的概率。在这里，无效调用同样无关紧要。对于子式调用，或者 $|Z\cap R|\leqslant\widetilde{\mathcal{O}}(\sqrt{k})$ 我们希望绝不在模式模式下执行操作的主要调用；由于共有 $\mathcal{O}(k\log n)$ 次要调用和主要调用，且我们以 $1-1/k$的概率不执行操作，因此在此成功的概率为 $n^{-\mathcal{O}(1)}$。对于 $Z\cap R$较大的主要调用，这种调用会将模式非平凡地划分到其子调用之间；因此，这类调用的数量至多为 $Z\ \widetilde{\mathcal{O}}(\sqrt{k})$，从而只需要求定理 2 3 的应用在所有此类调用中均成功2。.3 。

考虑所有类型的递归调用，我们可以得出结论：以概率 $2^{-\widetilde{\mathcal{O}}(\sqrt{k})}n^{-\mathcal{O}(1)}$，每个递归调用都能正确执行，从而为 G 的诱导子图得到所需的树分解。

2.4 扩展到有界距离版本（$d\geqslant 1$）

现在我们概述证明定理 1 31..3（即我们希望将与 $N_G^d[Z]$ 的交集限制在某个固定 $d\geqslant 1$内）的主要挑战。相比仅证明定理 1 21..2，我们还需要解决一些根本性问题。此外，还会出现一些需要谨慎处理并需要新思路的技术难题。

改进的对偶结果 由于我们的目标现在是限制与 $N_G^d[Z]$的交集，因此我们需要一个推论 2 22..2 的版本，不仅能限制与 Z的交集，也能限制与 $N_G^d[Z]$的交集。然而，推论 2 22..2 证明中所述的平均论证不再适用：即使有 p个互不相交的分隔集，原则上 Z 中的单个顶点也可能与某个分隔集中的许多顶点相邻，从而使与 $N_G^d[Z]$ 的交集变得非常大。类似地，对于几乎不相交路径的集合，原则上 Z 中的一个顶点可能与每条路径的一个私有顶点相邻，因此我们无法断言存在一条路径，其私有顶点与 $N_G^d[Z]$不相交。

不过，对于$K_{3,\ k}$-子式自由图，我们成功证明了推论 2 22。.2 的一个加强版本，以界定与 $N_G^d[Z]$ 的交集（下文的定理 2 52。.5 ）。事实上，我们需要一个更强的结果：在算法执行过程中，我们经常在$G-X$ 的 G诱导子图上应用推论 2 22。.2 。对于 $d\geqslant 1,\ v$ 中的一个顶点可能属于 $G-X$，这是因为 $N_G^d[Z]$，也可能是因为

顶点 $z \in X \cap Z$，或者是因为顶点 $z \in Z \cap (V(G) - X)$ 与 v 的距离超过 d，且位于 $G - X$ 中（即，最短 $z - v$ 路径经过 X中的顶点）。因此，即使我们想在$G - X$中寻找分隔集/不相交路径，也不能完全忽略删除集 X。我们给出一个推论 2 22。.2 的版本，它可以考虑这样的集合 X，前提是该集合诱导出的连通分量数量有界。在我们的应用中，我们可以对 $G[X]$的连通分量数量给出上界。

定理 2 52..5 的另一个方面在于，它相较于推论 2 22 2.. 提供了技术上的强化。一个简单的论证表明，如果 Z 是一个大小为 k 的已知集合，那么我们要么可以找到一条经过 Z 中至多 $\sqrt{k}$ 个顶点的 $s - t$ 路径，要么可以找到一个大小为 $\sqrt{k}$ 的 $s - t$ 分隔集。这可以看作是对推论 2 22..2 的改进；后者表明，存在一条经过 $\mathcal{O}(\sqrt{k})$ 个公共顶点的路径，或者存在一个 $s - t$ 分隔集，该分隔集具有较弱的性质：其大小为 $\mathcal{O}(k)$，并且与 Z 在 $\mathcal{O}(\sqrt{k})$ 个顶点处相交。定理 2 52..5 考虑的是这样一种情形：存在一个大小为 k_0 的已知集合 Z_0，以及一个大小为 k 的未知集合 Z。我们希望同时控制其与 $N_G^d[Z_0]$ 及 $N_G^d[Z]$ 的交集，使得相对于 k_0 能得到比相对于 k 更好的界。

在我们的算法中，集合 Z 将是在当前图 G 中剩余的未知解，而 Z_0 将是粘合 R。控制 $N_G^d[Z_0]$ 将使我们能够控制解中位于当前递归调用范围之外、但与粘合 R 的距离小于 d，因而 $N_G^d[x]$ 仍与当前递归调用范围相交的顶点 x 的 $N_G^d[x]$。

定理 2.5。 存在一个多项式时间算法，给定一个连通的 $K_{3,h}$-子式自由图 G、一对不同的顶点 $s, t \in V(G)$、满足 $p \geqslant 100d \log_2 q$, $q \geqslant (p + k_0/p + k_1)(10d^2h)^{d+3}\lambda$ 的正整数 $p, q \geqslant 2, \lambda, d, k_0, k$、一个大小为 k_0 的集合 Z_0，以及一个集合 $X \subseteq V(G) - \{s, t\}$，其中 $G[X]$ 至多有 λ 个连通分量，并输出以下结构之一：

- 一条（$C_1, \ldots, C_p$）的链，由 $G - X$ 中的（s, t）-分隔集组成，且每个分隔集的大小至多为 $2q$，满足

  1. 对每个 $j \in [p]$ 和 $v \in V$（G），都有

  $$\left|V(C_j) \cap N_G^d[v]\right| \leqslant (10d^dh)^{2d+1}(1 + \lambda \log_2(4q));$$

  2. 对任意大小至多为 k的 $Z \subseteq V$（G），存在一个 $j \in [p]$，使得

  $$\left|N_G^d[Z] \cap C_j\right| \leqslant ((k_0 + k)/p) \cdot \log_2 q \cdot 50d(10d^dh)^{2d+1} \cdot \lambda.$$

- 一组（$P_1, \ldots, P_q$）的 $G - X$ 中的（s, t）-路径，以及一个序列（$Q_1, \ldots, Q_q$），其中各集合的大小至多为 $10dp$，且 $Q_i \subseteq V$（P_i），满足：对于任意大小至多为 k的 $Z \subseteq V$（G），存在一个$j \in [q]$，使得 $N_G^d[Z] \cap$（V（P_i）$- Q_i$）$= \emptyset$。

为了证明定理 2 52。.5，我们先应用定理 2 12。.1，得到一条分隔集链或一组几乎不相交路径，然后对其进行修改以满足要求。事实上，我们需要使用定理 2 121。。的一个加强版本。我们注意到，在返回一组几乎不相交路径的情况下，Fomin 等人对定理 2 121。。的原始证明实际上证明了一个稍强的结论：公共顶点可以按层组织，且路径依次经过这些层（定理 4 242。。）。

我们注意到，如果确实存在两两不相交的 $s - t$ 路径，且某个顶点 $z \in Z$ 与这些路径中的 h 一条相邻（但不在任何一条路径上），那么进行收缩就会产生一个$K_{3,\ h}$-子式，这与我们关于 G的假设相矛盾（见图 22）。我们希望用类似的论证说明，单个顶点的 d-邻域不可能产生太大的影响，但公共顶点的存在使论证显著复杂化。直观地说，我们要证明的是，如果一个

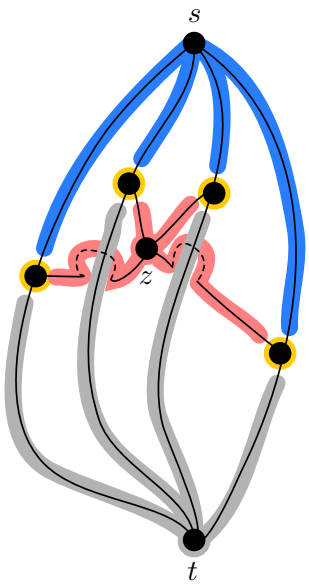

图 2：如果顶点 z 与 h 条内部不相交的 $s-t$ 路径相邻，且它不在这些路径上，那么就存在一个 $K_{3,\ h}$-子式：我们将蓝色子路径收缩为 s，将灰色子路径收缩为 t。更一般地，如果 z 无需经过任何其他 $s-t$ 路径（红色标出的路径）就能到达每条 $s-t$ 路径，同样的结论也成立；此时我们将这些路径收缩为 z..

顶点 Z 与许多路径 P_i 都很接近，具体而言，对每条路径 P_i，都存在一条从 z 出发、长度为 d 的路径，无需经过P_i的公共顶点即可到达一个私有顶点 P_i，那么我们就能构造一个 $K_{3,\ h}$-子式。此外，我们在 P_i 中引入“捷径”，以获得一条新的路径 P_i'，使其在 P_i的公共顶点的 d-邻域中仅包含有界数量的顶点。这确保修改后的路径之一与 $N_G^d[Z]$ 相交的顶点数量有界。

对于分隔集链，由于每个分隔集 C_j 都是极小的，对于每个 $u \in C_j$，都存在一条 $s-t$ 路径 P_u，它与 C_j 的交集仅为 u。这些路径当然可能相交。同样，如果存在一个顶点 z，它与 h 个 C_j 的顶点相邻，且不在任何路径 P_u 上，那么我们可以通过收缩构造一个 $K_{3,\ h}$-子式。然而，原则上，一个顶点 z 现在可能位于许多路径 P_u 上，使得这一论证难以应用。直观地说，我们证明在这种情况下，可以通过替换其中的许多顶点并将 z 加入其中，来减小分隔集 C_j 的大小。形式上，我们定义权重函数 $w_j(v) = 2^{\mathrm{dist}_{G-X}(v,\ C_j)}$，并根据权重函数 w_j，用一个权重最小的 $s-t$ 分隔集替换 C_j。我们证明，修改后的分隔集 C_j' 现在具有这样的性质：顶点 $z \in Z$ 只能接近 C_j' 中有界数量的顶点。我们还注意到，这些分隔集中有许多彼此不相交：如果 $|i-j| = \Omega(\log k)$，那么指数递增权重以及 $|C_i|,\ |C_j| \leqslant \mathrm{poly}(k)$ 这一事实保证了 C_i' 和 C_j' 不相交。

大邻域情况下的进展 另一个根本困难在于（如第 2.3 节所述），在调用定理 2.3 的主要递归调用中，我们之所以取得进展，是因为 Z 中的 $\Omega(\sqrt{k})$ 个顶点被放入集合 R，因此 $V(G)$ 与 Z 的交集显著缩小。然而，在 $d \geqslant 1$ 情况下，$N_G^d[Z]$ 的大小可能没有上界，因此将交集 $N_G^d[Z]$ 缩小 $\Omega(\sqrt{k})$ 并不算真正取得进展。例如，R 可能不包含 Z 中的任何顶点，却包含某个 Z 顶点的 $\Omega(\sqrt{k})$ 个邻居。这意味着，反复进行富递归调用可能并未取得任何进展

除了反复削去同一个顶点 $z \in Z$ 的$\widetilde{\Omega}(\sqrt{k})$ 个邻居之外。

为了解决这个问题，我们需要一些新的技巧。首先，我们注意到推论 2 42 4.. 可以加强。基于 Dujmović 等人 [22] 的乘积结构定理的论证实际上使我们能够证明：对于任意顶点 z 和整数 d，分隔集在距离 d 以内、距 Z 不超过该距离的顶点中只包含 $\mathcal{O}$（d）个。因此，在应用（ $d \geqslant 1$ 变体的）定理 2 32..3 之前，初始分离（A， B）具有良好的性质： $|A \cap B \cap N_G^d[Z]|$ 与靠近 $A \cap B$的 Z 顶点数量相当。

其次，我们从定理 2 52。.5 中提取出一条性质：输出的分隔集链也可以满足类似性质。遗憾的是，我们无法为输出的路径序列确立这一性质。不过，我们可以依靠这样一个事实：公共顶点只有 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个，而分隔集上的顶点有 $\mathcal{O}(k)$ 个。因此，我们可以将分隔集 $A \cap B$中的顶点划分为三类，而非之前的两类：

- 一个困难集 $\widetilde{C} \subseteq A \cap B$ 没有任何 $N_G^d[Z] \cap \widetilde{C}$结构，但具有界 $|\widetilde{C}| \leqslant \widetilde{\mathcal{O}}(\sqrt{k})$；
- 一个大小为 $\widetilde{\mathcal{O}}(k)$ 的稀疏集 $C \subseteq A \cap B$，满足 $|C \cap N_G^d[Z]| \leqslant \widetilde{\mathcal{O}}(\sqrt{k})$，且每个顶点$v \in V(G)$ 都满足 $|C \cap N_G^d[v]| \leqslant \widetilde{\mathcal{O}}(1)$；
- 一个大小无界且满足 $D \cap N_G^d[Z] = \emptyset$ 的无关集 $D \subseteq A \cap B$。

~ ~集合 C 的存在会延续到递归中：现在我们需要维护一个大小为 $\widetilde{\mathcal{O}}(\sqrt{k})$ 的集合 $R \subseteq R$，使得每个顶点 $v \in V(G)$ 都满足 $|N_G^d[v] \cap (R - \widetilde{R})| \leqslant \widetilde{\mathcal{O}}(1)$。为了控制前一句中的两个界，避免出现不希望有的 $\log n$ 项，我们需要定期清理 R 和 R 中较早的顶点，就像我们控制 R 的大小一样。因此，算法必须循环经过的不只是三种模式，而是五种不同的模式。

上述工具使我们能够通过控制 $Z \cap N_G^d[R]$ 的顶点数来控制 $R \cap N_G^d[Z]$ 的大小。这个量——即距离 R 至多为 d 的 Z 顶点数量——成为了 $d = 0$ 情形中度量 $R \cap Z$ 的对应物。这里有两个问题。首先，我们偶尔需要进入模式模式，以减小 $Z \cap N_G^d[R]$ 的大小。其次，我们需要以下对应于“丰富”猜测的做法。如果分离 (A, B) 在子调用中带来了大量此类新 Z 顶点——也就是说， $N_G^d[A \cap B]$ 中有超过 $\widetilde{\mathcal{O}}(\sqrt{k})$ 个 Z 顶点不属于 $N_G^d[Z]$——那么我们可以应用引理 2 32..3 的 $d \geqslant 1$ 版本来控制它们的数量，并将调用成功的概率计入子调用中加入 $N_G^d[R]$ 的新 Z 顶点数量。

3 预备知识

对于正整数 p，我们记作 $[p] := \{1, \ldots, p\}$。我们还使用记号 $[d \geqslant 1]$，当 $d \geqslant 1$ 时其值为 1，当 $d = 0$时其值为 0。

我们使用标准的图论记号。本文所讨论的所有图都是无向且简单的，即不含自环或重边。

对于图 G 和顶点 u，我们用 $N_d^G[u]$ 表示 u 的 d-邻域：与 u 的距离至多为 d 的所有顶点构成的集合。对于顶点集合，我们将记号扩展为$N_d^G[X] := \bigcup_{u \in X} N_d^G[u]$。如果上下文已明确图 G，则省略上标。

图中的独立集 是由两两不相邻的顶点构成的集合。我们通过考察底层无向图中的独立集，将这一概念扩展到有向图。

对于图 G 和不同的顶点 s， $t \in V(G)$，(s， t)-路径 是一个端点为 s和 t的路径，而(s， t)-分隔集是一个集合 $C \subseteq V(G) - \{s, t\}$，它与每条($s$， t)-路径都相交。由(s， t)-分隔集组成的序列(C_1， C_2、 ... ...、 C_p)称为链，如果 C_i 两两不交，并且对于每个$1 \leqslant i < j \leqslant p$， C_i 与从 s 到 C_j 的每条路径相交，且 C_j 与从 C_i 到 t的每条路径相交。

图 G μ V G $\to \mathbb{R}_{\geqslant 0}$ μ G $= \sum_{v \in V\ G} \mu\ v$ $>$中的权重函数 是一个函数 : ( ), 满足( ) : ( )。集合( )是一个平衡分隔集，相对于而言，如果$G - X$ 的每个连通分量的总权重至多为 $\mu(G)/2$。

图 G 的树分解是一个二元组（T, β），其中 T是一棵树，并且 β: V（T）$\to 2^V$（G）满足:（1）对于每个 $v \in V$（G），集合 $\{t \in V$（T）$\mid v \in \beta$（t）$\}$ 在 T中诱导出一个非空连通子图;（2）对于每个 $uv \in E$（G），存在 $t \in V$（T），使得 u, $v \in \beta$（t）。集合 β（t）通常称为 t 的袋。关于树分解的一个标准事实是，对于任意权重函数 μ，都存在 $t \in V$（T），使得 β（t）是 μ的平衡分隔集；例如参见引理 7.19]。

设（T, β）是一个树分解。对于一条边 $st \in E$（T），记 σ（st）：$= \beta$（s）$\cap \beta$（t），并称 σ（st）为边 st的粘合。如果 T 在上下文中以根为根，则简记 σ（t）：$= \sigma$（st），其中 s是 T中 t的父节点；对于 T 的根 r，则记 σ（r）$= \emptyset$。 $t \in V$（T）的躯干是子图 $G[\beta$（t）$]$，其中每个 $s \in N_T$（t）对应的粘合 σ（st）都被转化为一个团。

顶点图是指满足以下条件的图 H：存在某个 $v \in V$（H），使得 $H - v$ 是平面图。

我们的许多算法都是递归过程。对递归过程的一次调用的深度，是递归栈上的调用次数，其中包括当前调用。也就是说，初始根调用的深度为 1。对于参数 p，记号 $\mathcal{O}_p(\cdot)$ 隐去了可能依赖于 p 的乘法因子。

3.1 图小 minors

我们将使用近似可嵌入图的概念。不过，我们不需要采用例如 [22] 中给出的精确（复杂且具有拓扑性质的）定义。我们只需指出，对于每个由非负整数组成的元组 g, p, k, a，都存在一个 (g, p, q, a)-近似可嵌入图类，且该类对取子图封闭。（熟悉这一概念的读者会知道，g 是曲面的欧拉亏格，p 是涡旋的数量，q 是涡旋的最大宽度，a 是顶点的数量。）我们将 (p, p, p, p)-近似可嵌入图简称为p-近似可嵌入图，并将 $(p, p, p, 0)$-近似可嵌入图简称为无顶点的 p-近似可嵌入图。我们将用到这样一个事实：每个 p-近似可嵌入图 G 都包含一个至多有 p 个顶点的集合 A，使得 $G - A$ 是无顶点的 p-近似可嵌入图。

采用这一记号，可以将排除某个子式的图类的结构定理表述如下。[2]

定理 3.1。对于每个图 H，都存在一个常数 p_H 和一个多项式时间算法：给定一个不含 H-子式的图 G，该算法可以计算出一个树分解 (T, β)，其 G 的最大粘合度大小至多为 p_H；并且对于每个 $t \in V(T)$，都有一个集合 $A_t \subseteq \beta(t)$，使得每个节点 t的躯干都是 p_H-近似可嵌入的，并且移除 A_t后变为不含顶点的 p_H-近似可嵌入图。此外，如果 H 是一个顶点图，那么可以令 $A_t = \emptyset$，也就是说，整个躯干都是不含顶点的 p_H-近似可嵌入图。

我们将需要定理 3 1 的以下部分逆命题31。。。第一部分来自 [31]；第二部分是民间结果，例如可在 [13]中的引理 B.1 找到其明确表述。

定理 3.2。对于每个 p，都存在一个图 H，使得每个 p-近似可嵌入图都是 H-子式自由的。此外，对于每个 p，都存在一个顶点图 H，使得每个不含顶点的 p-近似可嵌入图都是 H-子式自由的。

[2]遗憾的是，我们未能在文献中找到关于这一（在学界众所周知的）陈述的合适引文。它可以由 [23] 中的定理 12 推导如下。对于作为顶点图的 H，顶点集 A_i 中没有顶点是主顶点。我们可以通过在曲面上增加若干个把手，将所有涡旋合并为一个涡旋，然后将所有顶点都加入所得超涡旋的所有袋中。

我们还将用到这样一个事实：在排除某个固定顶点图作为子式的图中，树宽关于直径呈线性有界。

定理 3.3 ([20])。对于每个顶点图 H，都存在一个常数 c_H，使得给定一个H-子式自由图 G，其直径为 Δ，可以在多项式时间内计算出 G 的一个树分解，其宽度至多为 $c_H\Delta$。

推论 3.4。对于每个整数 p，都存在一个常数 c_p，使得给定一个直径为 Δ 的 p-近似可嵌入且不含顶点的图 G，可以在多项式时间内计算出 G的树分解，其宽度至多为 $c_p\Delta$。

证明。该结论由定理 3.3 的第二个陈述3。.2 和定理 3.333。。 . □

3.2 乘积结构理论的工具

在我们的证明中，需要用到 Dujmović 等人 [22] 的一项结果：不含顶点子式的图具有所谓的乘积结构。为表述这一结果，我们需要先介绍几个定义。

设 G 是一个图。若划分 $\mathcal{P}$ 将 $V(G)$ 划分后，其每个部分 $P \in \mathcal{P}$ 都在 G 中诱导出一个连通子图，则称该划分为连通划分。G 的一个分层是一个序列 $\mathcal{L} = (V_1, V_2, \dots, V_m)$，由 $V(G)$ 的两两不交子集组成，并满足 $\bigcup_{i=1}^{m} V_i = V(G)$；对于每条边 $uv \in E(G)$，例如其端点为 $u \in V_i$ 和 $v \in V_j$，都有 $|i-j| \leqslant 1$。集合 V_i 称为层。对于 $V(G)$ 的划分 $\mathcal{P}$ 和 G 的分层 $\mathcal{L}$，$(\mathcal{P}, \mathcal{L})$ 的宽度是 $|P \cap V_i|$ 的最大值，其中 P 遍历$\mathcal{P}$ 的各部分，V_i 遍历 $\mathcal{L}$的各层。对于 $V(G)$ 的划分 $\mathcal{P}$，商图 $G/\mathcal{P}$ 的顶点集为 $\mathcal{P}$；当且仅当存在 $u \in P$ 和 $v \in Q$ 使得 $uv \in E(G)$ 时，部分 P 和 Q 之间有一条边。

排除顶点图作为子式的图的乘积结构理论，其主要结果可以表述如下。这基本上是将引理 24], 表述为一种算法的 [22, 引理。

定理 3.5 (引理 24 的 [22], 算法版本)。对于每个常数 p，都存在一个常数 c和一种多项式时间算法；给定一个无顶点的连通 p-近似可嵌入图 G，该算法会计算出 G的一个连通划分 $\mathcal{P}$、一个 G 的分层 $\mathcal{L}$，使得$(\mathcal{P}, \mathcal{L})$ 的宽度至多为 c，并且得到 $G/\mathcal{P}$ 的一个宽度小于 c的树分解。

为了从树分解中提取平衡分隔集，我们需要以下标准引理。

引理 3.6。设 G 为一个图，(T, β) 为其树分解，且 $\mu\colon V(G) \to \mathbb{R}_{\geqslant 0}$ 为一个权重函数，其支撑包含于 $U \subseteq V(G)$中。

- 给定 G、(T, β)和 μ，可以在多项式时间内计算出 $t \in V(T)$，使得t 的袋是一个平衡分隔集，具体而言：$T - \{t\}$的每个连通分量 S 都满足 $\mu(\bigcup_{r \in S} \beta(r) - \beta(t)) \leqslant \mu(V(G))/2$。

- 给定 G、(T, β) 和 U，可以在多项式时间内计算出 T 的一个节点集合 Y，其大小至多为 $2|U| - 1$，使得对于每个支撑包含于U的权重函数 $\mu\colon V(G) \to \mathbb{R}_{\geqslant 0}$，都存在一个 $t \in Y$，它是如上一条所述的平衡分隔集。

证明。在任意节点中选取 T 作为根。

对于每个 $u \in U$，令 t_u 为 T 中袋包含 u 的最顶部节点。令 $X = \{t_u \mid u \in U\}$，并令 Y 由 X 以及 X 中任意一对节点的最近公共祖先组成。显然，$|X| \leqslant |U|$和 $|Y| \leqslant 2|X| - 1 \leqslant 2|U| - 1$。注意，$Y$ 的定义取决于 U，而非权重函数 μ 的具体形式。

如果对于 $T-\{t\}$ 的每个连通分量 S，都有$\mu(\bigcup_{r\in S}\beta(r)-\beta(t)) \leqslant \mu(V(G))/2$，则称节点 $t\in V$（T）是良好的 。

对于每条边 $st\in E(T)$，其中 t 是 s 的父节点，考察 $T-\{st\}$ 的两个连通分量 T_s 和 T_t，比较 $\mu(\bigcup_{r\in V(T_s))}\beta(r))$ 与 $\mu(\bigcup_{r\in V(T_t)}\beta(r))$ 的权重，并将边 st 从权重较小的一端指向权重较大的一端；若权重相等，则将边向上定向（从 s 指向 t）。令 t_0 为出度为 0 的节点。由于与 t_0 相邻的每条边都指向 t_0，因此 t_0 是好的。这证明了存在一个好的节点（我们可以在第一条中返回这样一个节点）。

对于第二条，只需证明以下结论：如果 t 是好的，但 $t\notin Y$，那么 t 不是 T 的根，而且 t 的父节点也很好。这证明最顶端的好节点位于 Y 中，第二条的证明也就完成了。

为此，假设 t 是好的，而 $t\notin Y$ 不是。这意味着，对于 $T-\{t\}$ 中至少一个作为 t 的子树的连通分量 S，有 $U\cap(\bigcup_{r\in S}\beta(r)-\beta(t))\neq\emptyset$。然而，由于 $\mu(\bigcup_{r\in S}\beta(r)-\beta(t)) \leqslant \mu(V(G))/2$，对于任意 $t\notin X$，必定存在 $T-\{t\}$ 中的一个连通分量 S'，它不是 t 的子树；特别地，t 不是 T 的根。令 s 为 t 的父节点。由于 $t\notin X$，我们有 $\beta(s)\cap U\supseteq\beta(t)\cap U$。这意味着 s 也是好的：$T-\{s\}$ 的每个连通分量要么包含于 S' 中，要么其袋与 S 的袋包含 U 中相同的顶点。 □

推论 *3.7*。固定一个常数 p，并令 c 为定理 *3 53 5.*. 给出的关于 p 的常数。于是，存在一个多项式时间算法：给定一个直径为 Δ、不含顶点的 p-近似可嵌入图 G，该算法可计算出 G 的树分解 (T,β)，并且对每个 $t\in V$（T），计算出 $G[\beta$（t）的生成森林 $F_t]$,，使得对每个 $t\in V$（T）均满足以下条件：

1. $|\beta$（t）$|\leqslant c^2\cdot(\Delta+1)$，
2. F_t 至多有 c 个连通分量，
3. F_t 的最大度至多为 $3c^2-1$，并且
4. 对每个整数 $d\geqslant 0$ 以及每个 $v\in V(G)$，$\beta(t)$ 中与 v 的距离至多为 d 的顶点不超过 $(2d+1)c^2$ 个。

特别地，我们有以下几个程序。

（显式权重函数的用法。）此外，给定一个权重函数 $\mu\colon V(G)\to\mathbb{R}_{\geqslant 0}$，我们可以在多项式时间内计算出 G 的一个分离(A,B)，使得 $|A\cap B|\leqslant c^2(\Delta+1)$、$\mu(B-A)\ \mu(A-B)\leqslant\frac{2}{3}\mu(V(G))$，以及一个 $G[A\cap B]$ 的生成森林 F 满足上述性质，其中 $A\cap B$ 扮演 $\beta(t)$ 的角色；

（显式支撑的用法。）此外，给定一个集合 $U\subseteq V(G)$，我们可以在多项式时间内计算出一个大小至多为 $2|U|-1$ 的元组族 $\mathcal{X}$，其中的元组 $((A,B)\ F)$ 如前一条所述；对于每个支撑包含于 U 的权重函数 $\mu\colon V(G)\to\mathbb{R}_{\geqslant 0}$，至少有一个元素 $((A,B)\ F)\in\mathcal{X}$ 满足前一条中针对 μ 的条件。

证明。 将定理 3 53..5 应用于图 G，取常数 p，得到 $\mathcal{P}$、 $\mathcal{L}$，以及 $G':=G/\mathcal{P}$ 的树分解 (T,β')。对于每个 $t\in V(T)$，令 $\beta(t):=\bigcup_{P\in\beta'(t)}P$。显然，$(T,\beta)$ 是 G 的树分解。

固定 $t\in V(T)$。由于 $\beta(t)$ 由 $\mathcal{P}$ 的 $|\beta'(t)|\leqslant c$ 个部分组成，且每个部分 $P\in\mathcal{P}$ 在 $\mathcal{L}$ 的每一层至多包含 c 个顶点，因此 $\beta(t)$ 的单层顶点数至多为 c^2。由于 G 的直径至多为 Δ，$\mathcal{L}$ 中至多有 $\Delta+1$ 层非空。这意味着 $|\beta(t)|\leqslant c^2(\Delta+1)$。

由于 $\beta(t)$ 是至多由 c 个连通部分构成的并集，因此 $G[\beta(t)]$ 至多有 c 个连通分量。此外，由于 G 的每个顶点仅与至多

$\mathcal{L}$ 的三层中的顶点相邻，因此 $G[\beta(t)]$ 的最大度至多为 $3\cdot|\beta'(t)|\cdot c-1\leqslant 3c^2-1$。因此， F 的 $G[\beta(t)]$ 的任意生成森林至多有 c 个连通分量，且最大度至多为 $3c^2-1$。

为证明最后一个性质，注意，若顶点 $v\in V(G)$ 位于 $\mathcal{L}$ 的第 i 层，而顶点 $u\in\beta(t)$ 到 v 的距离至多为 d，则 u 位于第 $(i-d)$ 层与第 $(i+d)$ 层之间。由于 $\beta(t)$ 在 $\mathcal{L}$ 的每一层中至多包含 c^2 个顶点，因此 $\beta(t)$ 中到 v 的距离至多为 d 的顶点至多有 $(2d+1)c^2$ 个，证毕。

关于两个项目符号中的论断，首先不失一般性地假设 T 的最大度至多为 3（通过复制一些节点即可轻松进行简单修改来实现这一点）。接着，在树分解 (T,β) 和 μ 或 U 上应用引理 3 63..6 中相应的项目符号。最后，注意，对于满足引理 3 63..6 意义下的、作为 μ 的平衡分隔集的节点 t，存在一个对 $T-\{t\}$ 的连通分量的划分 $(\mathcal{A},\mathcal{B})$，使得分离 $(A:=\beta(t)\cup\bigcup_{r\in\mathcal{A}}\beta(r)\ B:=\beta(t)\cup\bigcup_{r\in\mathcal{B}}\beta(r))$ 满足要求，其中森林 F_t 作为 $A\cap B=\beta(t)$。由于 T 的最大度至多为 3，因此需要检查（若给定 μ）或枚举（若给定 U）的划分 $(\mathcal{A},\mathcal{B})$ 只有常数个。□

3.3 簇与簇族

在定理 131..3的证明中，我们将在某个阶段方便地将叙述从球转为仅讨论具有有界（强）直径的子图。下面介绍一些记号和基本事实。

固定一个整数 d，令 G 为一个图。d-簇 在 G 中是一个非空集合 $A\subseteq V(G)$，满足 $G[A]$ 连通且直径至多为 d。注意，0-簇是单顶点集。我们注意到如下事实。

引理 3.8。 令 $d\geqslant 0$ 为一个整数，G 为一个图，且 $\mathcal{Z}$ 为 d-簇的一个族。则存在一个族 $\mathcal{Z}'$，其中的簇在 G 中两两不交，且为$(2d)$-簇，并满足 $|\mathcal{Z}'|\leqslant|\mathcal{Z}|$ 和 $\bigcup\mathcal{Z}=\bigcup\mathcal{Z}'$。

证明。令 $U=\bigcup\mathcal{Z}$。令 $A\subseteq U$ 为 U 的顶点中一个包含意义下极大的集合，使得其中任意两个顶点在图 $G[U]$ 中的距离都大于 d。由于每个 $K\in\mathcal{Z}$ 至多包含 A 中的一个顶点，因此有 $|A|\leqslant|\mathcal{Z}|$。如下构造 U 的一个划分 $(K_u)_{u\in A}$：将每个 $v\in U$ 分配给 K_u，其中 u 是图 $G[U]$ 中距 v 最近的 A 顶点，并任意打破平局（即，K_u 是图 $G[U]$ 中以 A 为中心的沃罗诺伊胞）。

根据构造，对于每个 $u\in A$ 和 $v\in K_u$，$G[U]$ 中从 v 到 u 的一条最短路径包含在 K_u中。这表明 $G[K_u]$ 是连通的；并且根据 A的极大性，这条路径的长度至多为 d。因此，$G[K_u]$ 的直径至多为 $2d$。由此可知，$\mathcal{Z}'=\{K_u\mid u\in A\}$ 满足所需的性质。□

引理 383..8 使我们能够将以 d 为半径、以 Z 中顶点为中心的球组成的族，转化为至多 k 个两两不交的 $(4d)$-簇。

4 流-割对偶

本节将发展定理 2 12..1 的一个变体，该变体处理模式的距离d 邻域，但仅适用于$K_{3,\ t}$-子式自由图。证明中，我们需要深入探究定理 2 12..1 的证明，以推导出一些额外性质。以下结论可从 [27] 中的证明提取出来；为完整起见，我们在附录 A 中重现该证明A

定义 4.1 ((*p*， *q*)-结构)。 设 G 为无向图， s， $t \in V(G)$ 为两个不同的顶点，且 p， $q \geqslant 0$ 为整数。(*p*, *q*)-结构在 (*G*， *s*， *t*) 中由一个 (C_1，……， C_p) 的链（由 (*s*， *t*)-分隔集构成）和一个 (P_1，……， P_q) 的序列（由 (*s*， *t*)-路径构成）组成，并满足以下条件：

1. 对于每个 $1 \leqslant i \leqslant q$ 和 $1 \leqslant j \leqslant p$，我们有 $|V(P_i) \cap C_j| = 1$。注意，这意味着对于一条路径 P_i，位于 $V(P_i)$ 与分隔集 C_1、C_2、……、C_p 交集中的唯一顶点，按照我们从 s 到 t 遍历 P_i 的顺序，恰好依次排列。2. 对于每个 $1 \leqslant i \neq i' \leqslant q$， $V(P_i) \cap V(P_{i'}) - \{s,t\}$ 中的每个顶点都位于分隔集 C_j 之一上。注意，这意味着 $|V(P_i) \cap \bigcup_{i' \in [q]-\{i\}} V(P_{i'})| \leqslant p$。3. 对于每个 $1 \leqslant j \leqslant p$， C_j 的每个顶点都位于某条路径 P_i 上。注意，这意味着对于每个 $1 \leqslant j \leqslant p$, $|C_j| \leqslant q$。

定理 4.2。 存在一个多项式时间算法，给定一个图 G、不同的顶点$s, t \in V(G)$以及一个整数 $q \geqslant 0$, 输出一个整数 $p \geqslant 0$ 和一个 (p, q)-结构于 (G, s, t)中。

注意，定理 424。.2 可以输出 $p = 0$；在这种情况下，链为空，但第二个性质意味着路径 P_i 除端点外彼此顶点不相交。

4.1 距离版本

定理 2 52..5 的证明首先调用定理 4 24..2，以找到一个带有$q = q_0$的 (p_0, q_0)-结构。然后我们考虑两种情况。如果 p_0 较小，那么我们修改 (p_0, q_0)-结构中的路径，以得到所需的路径集。如果 p_0 较大，那么我们修改 (p_0, q_0)-结构中的分隔集，以得到所需的分隔集。本文提出的引理 4 64..6 和 4 124..12 分别处理这两种情况。

在这两种情况下，我们都需要以下民间结果：最大出度较小的有向图包含一个较大的独立集。

引理 4.3. 若有向图 H 的出度至多为 d，则它包含一个大小至少为 $\lceil |V(H)|/(2d+1) \rceil$ 的独立集。

证明. 设 H' 是 H的底层无向图。我们尝试通过贪心地选择一个度最低的顶点 v，将 V 加入独立集并删除其邻域，从而找到一个大小为H' 的独立集。 H 的每个诱导子图的出度都至多为 d，因此 H' 的每个诱导子图的平均度都至多为 $2d$。这意味着贪心过程总能找到一个度至多为 $2d$ 的顶点，因此每次将新顶点 v 加入独立集时，至多会额外舍弃 $2d$ 个顶点。由此，算法找到的独立集大小为 $\lceil |V(H)/(2d+1) \rceil$。

□

处理几乎不相交的路径。我们首先处理这样的情形：存在一个具有较小 p 的 (p, q)-结构，也就是说，有一个很大的几乎不相交路径集合，其中每条路径都只有 p 个公共顶点。主要论证在于证明：如果某个 Z 顶点的 d-邻域能够影响 (p, q)-结构中某条路径上的过多顶点，那么我们就能在该图中找到一个 $K_{h,3}$-子式。但这只有在修改路径之后（使用捷径）才成立：如果这类顶点靠近路径上的某个公共顶点，那么一条路径上可能有很多这样的顶点。

在给出证明之前，我们先确立一个基本的组合结果。首先，我们需要 Erdős 和 Rado 的向日葵引理。

引理 4.4 (向日葵引理 [44])。 设 $\mathcal{F}$ 是某个全集中由大小为 r 的子集组成的族。若 $|\mathcal{F}| > r! \cdot (k-1)^r$, 则存在 k 个互异集合 $S_1, S_2, \ldots, S_k \in \mathcal{F}$ 以及全集的一个（可能为空的）子集 C，使得对所有 $1 \leqslant i < j \leqslant k$，都有 $S_i \cap S_j = C$。

我们可以使用向日葵引理来界定交叉相交对的大小。

引理 4.5。设 $A_1, \ldots, A_n, B_1, \ldots, B_n$ 是全集的子集，使得对每个 $i \in [n]$，都有 $|A_i| \leqslant a, |B_i| \leqslant b$，$A_i \cap B_i = \emptyset$，且当 $i, j \in [n], i \neq j$ 时，有 $B_i \neq B_j$。若 $n \geqslant b!(t(1+2a) - 1)^b$，则存在一个大小为 t 的子集 $I \subseteq [n]$，使得对每个 i，都有 $A_i \cap B_j = \emptyset, j \in I$。

证明。让我们将引理 4 44..4 应用于互异集合 B_i 和 $k = t(1+2a)$；令 $I_1 \subseteq [n]$ 对应于所得的 k 集合 $B_i, i \in I_1$。注意，对于每个 $i \in I_1$，至多有 a 个值 $j \in I_1$ 使得 $A_i \cap B_j \neq \emptyset$：公共交集 $C = \bigcap_{j \in I_1} B_j$ 中的顶点不能属于 A_i，而其他每个顶点对于 $j \in I_1$ 至多属于一个 B_j。让我们在 I_1 上构造一个有向图 H，其中当且仅当 $A_i \cap B_j \neq \emptyset$ 时存在一条边 $\overrightarrow{ij}$。由于 H 的每个顶点的出度至多为 a，引理 4 34..3 表明存在一个大小至少为 $|I_1|/(1+2a) = t$ 的独立集 I_2。 □

下面的引理对一个 (p, q)-结构进行修改。该结构的 q 很大（相较于 p 和 k），修改后，对于某个大小至多为 k 的未知集合 Z，其中一条路径与 Z 的交集较小，而且与其 d-邻域的交集也较小。

引理 *4.6.* 存在一个多项式时间算法：给定一个连通的 $K_{3,h}$-子式自由图 G、满足 $r \geqslant p$ 和 $q \geqslant (p + k_0/r + k_1)(10d^2h)^{d+3}\lambda$ 的正整数 $p, q, r, d, \lambda, k_0, k_1$、一对不同的顶点 $s, t \in V(G)$、一个大小至多为 k_0 的集合 $Z_0 \subseteq V(G)$、一个集合 $X \subseteq V(G) - \{s, t\}$（使得 $G[X]$ 至多有 λ 个连通分量），以及一个由 $(G - X, s, t)$ 构成的 (p, q)-结构；对于某个 $q' \leqslant q$，输出一条由 $G - X$ 中的 (s, t)-路径组成的序列 $(P'_1, \ldots, P'_{q'})$，并为每个满足以下条件的 $i \in [q']$ 输出一个集合 $Q_i \subseteq V(P'_i)$：

- $|Q_i| \leqslant 10dr$ 对每个 $i \in [q']$。
- 对于每个大小至多为 k_1 的 $Z_1 \subseteq V(G)$，都存在一个 $i \in [q']$，使得 $V(P'_i) - Q_i$ 与 $N^d_G[Z_0 \cup Z_1]$ 不相交。

证明。设 $(C_1, \ldots, C_p)$、$(P_1, \ldots, P_q)$ 是给定的 (p, q)-结构，属于 (G, s, t)。令 Pub_i 为 P_i 的公共顶点，即 $P_i - \{s, t\}$ 中也位于某条其他路径 $P_j, i \neq j$ 上的那些顶点。令 $\mathsf{Priv}_i = V(P_i) - \mathsf{Pub}_i - \{s, t\}$。我们以自然的方式为 $G - X$ 中的顶点定义分层编号，如下所示。对于 $1 \leqslant i \leqslant p$，$(p, q)$-结构中的分隔集 C_i 的每个顶点都位于第 i 层，分隔集 C_i 与 C_{i+1} 之间的顶点也如此。顶点 s 以及 s 与 C_1 之间的所有顶点都位于第 0 层，而顶点 t 以及 C_p 与 t 之间的所有顶点都位于第 p 层。注意，相邻顶点的层号之差至多为 1，并且每条路径 P_i 在每一层 $\ell \in [p]$ 中恰有一个公共顶点。

以下构造有助于处理集合 X。令 G' 为从 G 收缩 $G[X]$ 中每个顶点所得的图，其中每个顶点都收缩为一个单独的顶点；令 X' 为这些收缩顶点构成的集合。注意，如果 $v \notin X'$，那么 $G' - (X' - \{v\})$ 就是 $G' - X'$；如果 $v \in X'$，那么 v 就是 $G' - (X' - \{v\})$ 中唯一位于 $G' - X'$ 之外的顶点。

对于 $G - X$ 中的任意顶点 v，我们希望界定路径 P_i 的数量，使得沿着 $G - X$ 中长度至多为 d 且不经过 P_i 中公共顶点的路径，可以从 v 到达 P_i 的某个私有顶点。类似地，我们也希望界定这类路径 P_i 的数量，使得沿着 G 中长度至多为 d 且不经过 P_i 中公共顶点的路径，可以从 X 到达某个私有顶点。以下命题在所到达的私有顶点位于同一层这一假设下，为这两种情形给出了统一的界。

引理 4.7. 设 v 是 $G' - \{s, t\}$ 的一个顶点，$I \subseteq [q]$ 是一个指标子集。对于每个 $i \in I$，令 $u_i \in \mathrm{Priv}_i$ 为一个顶点，并令 W_i 为 $G' - (X - \{v\})$ 中的一条长度至多为 d 的 $v - u_i$ 路径。进一步假设

- 对于每个 $i \in I$，顶点 u_i 都位于同一层 ℓ，并且

- 对于每个 $i \in I$，都有 $P_i \cap W_i = \{u_i\}$。

那么 $|I| < (10d^2h)^{d+2}$。

断言的证明。假设 $|I| \geqslant (10d^2h)^{d+2}$。我们构造一个以 I 为顶点集的有向图 H，其中当 $W_{i_1} \cap \mathbf{Priv}_{i_2} \neq \emptyset$ 时，$i_1, i_2 \in I$ 之间存在一条边 $\overrightarrow{i_1 i_2}$。由于 W_{i_1} 至多有 $d+1$ 个顶点，且对于不同的 i_2 值，集合 $\mathbf{Priv}_{i_2}$ 两两不相交，因此 H 的每个顶点的出度至多为 $d+1$。于是，根据引理 4 34..3，在 H 中存在一个大小至少为 $|I|/(2(d+1)+1) \geqslant (10d^2h)^{d+1}$ 的独立集 I'。特别地，I' 是一个独立集，这意味着对于每个 $i \in I'$，都有 $v \notin \mathbf{Priv}_i$；否则，对于任意 $j \in I', j \neq i$，W_j 都会与 $\mathbf{Priv}_i$ 相交（此处我们用到了 $|I'| \geqslant 2$）。因此，我们可以假设对于每个 $i \in I'$，都有 $v \neq u_i$。此外，$s, t \notin W_i$，因为否则该顶点就会是 $P_i \cap W_i$ 中一个不同于 u_i 的顶点。

考虑某些 i、$j \in I'$、$i \neq j$，使得 P_i 和 W_j 相交。注意，W_j 中至多只有一个不属于 $G' - X'$ 的顶点：其端点 v。由于 u_j 位于第 ℓ 层，这意味着 $W_j - X'$ 的每个顶点都位于第 $\ell - d$ 层与第 $\ell + d$ 层之间。因此，如果 P_i 和 W_j 在顶点 w 处相交，那么根据 I' 的定义，我们有 $w \notin \mathbf{Priv}_i$，从而 w 必须是位于第 $\ell - d$ 层与第 $\ell + d$ 层之间的 P_i 中至多 $2d+1$ 个公共顶点之一。令 A_i 为 $\mathbf{Pub}_i$ 的这一子集，并令 $B_i = W_i$；注意，由于 W_i 与 $\mathbf{Pub}_i$ 不相交，$A_i \cap B_i = \emptyset$。让我们将引理 4 54..5 应用于 $a = 2d+1$、$b = d+1$ 和 $k = h$：注意，$|I'| \geqslant (10d^2h)^{d+1} \geqslant (d+1)!(h(2d+2)-1)^{d+1} = b!(h(1+2a)-1)^b$ 成立。因此，存在一个大小为 h 的 $I'' \subseteq I'$ 子集，使得对于所有 i、$j \in I''$，$A_i \cap B_j = \emptyset$。

由于对于任意 $i, j \in I''$，路径 P_i 和 W_j 互不相交，且每个 u_i 都位于同一层，对每个 $i \in I'$ 执行以下收缩操作，会生成一个包含 $\{s, t, v\}$ 于一侧、$\{u_i \mid i \in I''\}$ 于另一侧的 $K_{3,h}$-子图：

- 将 $W_i - \{u_i\}$ 收缩到顶点 v。
- 将 P_i 中从 s 到（但不包括）u_i 的子路径收缩到顶点 s。
- 将 P_i 中从紧接在 u_i 之后的顶点到 t 的子路径收缩到顶点 t。

我们可以验证，被收缩到不同顶点（s、t 或 v）的顶点彼此不相交。为此，我们观察以下事实。

- 对于任意 $i, j \in I'$，路径 $W_i - \{u_i\}$ 与每个 P_j 都不相交。
- 收缩为 s 的 P_i 的顶点，只有在 P_i 和 P_j 的公共顶点处，才会与收缩为 t 的 P_j 的顶点重合。然而，对于任意 i，$j \in I'$，从 s 到 u_i 的 P_i 子路径上的公共顶点，其层号至多为 ℓ；而从紧接 u_j 之后的顶点到 t 的 P_j 子路径上的公共顶点，其层号大于 ℓ。

因此，这一系列收缩在 $G' - (X - \{v\})$ 中构成了一个 $K_{3,\ h}$-子式。这与以下假设矛盾：G（因此其子式 $G' - (X - \{v\})$ 也是）不含 $K_{3,\ h}$-子式。」

以下两个断言利用一个简单的鸽巢原理论证，去除断言 47 中的层要求47。。。

断言 4.8。 对于每个顶点 $v \in V(G) - X$，至多有 $(10d^2h)^{d+3}$ 条路径 P_i 满足距离 $_{G-(X \cup \mathrm{Pub}_i \cup \{s,t\})}(\mathrm{Priv}_i, v) \leqslant d$。

断言的证明。假设这样的路径多于 $(10d^2h)^{d+3}$ 条；令 I 包含这些路径的索引。对于每个 $i \in I$，我们选取一条最短路径 W_i，从 v 到 $\mathbf{Priv}_i$，位于$G - (X \cup \mathbf{Pub}_i \cup \{s, t\})$中；令 u_i 为其在 $\mathbf{Priv}_i$中的端点。显然，W_i 与 $\mathbf{Priv}_i$ 仅在 u_i处相交。

由于在 $G - X$ 中，每个 u_i 到 v 的 d 至多为 d，因此 u_i 的层与 v 的层相差至多 d。根据鸽巢原理，存在一个大小大于 $|I|/(2d+1) \geqslant (10d^2h)^{d+2}$ 的子集 $I' \subseteq I$，使得对每个 $i \in I'$，u_i 都处于同一层。然而，断言 4 74 7.. 表明 $|I'| < (10d^2h)^{d+2}$，矛盾。」

断言 4。 9. </style}至多有 ($10d^2h)^{d+2}\lambda$ 条路径 P_i 满足 距离$G-\mathrm{Pub}_i\cup\{s,t\}(\mathrm{Priv}_i, X) \leqslant d$.

断言的证明。假设这样的路径多于 $p\cdot(10d^2h)^{d+3}\lambda$ 条；令 I 包含这些路径的索引。注意，dist $G-\mathrm{Pub}_i\cup\{s,t\}$(Priv$_i$, X) $\leqslant d$ 也可以写成 dist$_{G'-\mathrm{Pub}_i\cup\{s,t\}}$(Priv$_i$, X') $\leqslant d$。对于每个 $i\in I$，我们选取一条从 X 到 Priv$_i$、位于 $G-(X\cup\mathrm{Pub}_i\cup\{s,t\})$ 中的最短路径 W_i；令 u_i 为其在 Priv$_i$中的端点。显然， W_i 与 Priv$_i$ 的交集仅为u_i。根据鸽巢原理，存在一个大小大于 $|I|/p \geqslant(10d^2h)^{d+2}\lambda$的子集 $I'\subseteq I$，使得其中每个 u_i 都处于同一层。再次根据鸽巢原理，存在一个大小大于 $|I'|/\lambda \geqslant (10d^2h)^{d+1}$ 的子集$I''\subseteq I'$，使得对于每个 $i\in I'$， W_i 都从同一顶点 $x'\in X'$出发。然而，断言 4 74..7 表明 $|I'|<(10d^2h)^{d+2}$，矛盾。」

对于 $i\in[q]$，令 $Z_0^i\subseteq Z_0-X$ 为由所有满足 dist$_{G-(X\cup\mathrm{Pub}_i\cup\{s,t\})}(z,$ Priv$_i)\leqslant d$ 的顶点 $z\in Z_0-X$ 组成的集合（由于 Z_0 在输入中已给出，我们可以计算此集合）。考虑由顶点集诱导的图 G_i

$$\mathrm{Priv}_i\cup\bigcup_{v\in\mathrm{Pub}_i\cup Z_0^i\cup\{s,t\}}N_{G-X}^d[v].$$

令 P_i' 为 G_i 中的一条最短 $s-t$ 路径。

断言 4.10. $|V(P_i')\cap N_{G-X}^d[v]|\leqslant 2d+1$ 对每个 $v\in\mathrm{Pub}_i\cup Z_0^i\cup\{s,t\}$。

断言的证明。 对于某个 $v\in\mathrm{Pub}_i\cup Z_0^i\cup\{s,t\}$，令 u_1（分别为 u_2）表示 $N_{G-X}^d[v]$ 中 P_i'的第一个（分别为最后一个）顶点。若 P_i' 在 $N_{G-X}^d[v]$ 中至少有 $2d+2$ 个顶点，则 P_i' 的 u_1-u_2子路径包含所有这些顶点，因此其长度至少为 $2d+1$。然而，在图 G_i 中，存在一条经过 v、长度至多为 $2d$ 的 u_1-u_2路径。这与路径 P_i' 的极小性矛盾。」

定义 $Q_i=V(P_i')\cap\bigcup_{v\in\mathrm{P\_u\,b}_i\cup Z_0^i\cup\{s,t\}}N_{G-X}^d[v]$。由断言 4 104。.10可知，显然$|Q_i|\leqslant(2d+1)(p+|Z_0^i|+2)$。此外，以下命题表明，对于过多的路径 $|Z_0^i|$ (以及 Q_i) 而言， P_i'不会太大。

命题 4.11。 至多有 $(10d^2h)^{d+3}k_0/r$ 个 $i\in[q]$ 的取值满足 $|Q_i|\geqslant 10dr$。

断言的证明。 根据断言 4 84..8，每个顶点 $Z\in Z_0-X$ 至多可属于 Z_0^i 中对应于 $(10d^2h)^{d+3}$ 个 i 的情形。因此，根据平均论证，至多有 $(10d^2h)^{d+3}k_0/r$ 个 i 的取值满足 $|Z_0^i|>r$。假设对于某个 $i\in[q]$，有 $|Z_0^i|\leqslant r$。对于这样的 i，由于 $|\mathrm{P\_u\,b}_i\cup Z_0^i\cup\{s,t\}|\leqslant p+r+2$ 4 10 $Q_i=|V\ P_i'\ \cap\bigcup_{v\in\ \ _i\cup Z_0^i\cup\{s,t\}}N_{G-X}^d[v]|\leqslant\ p+r+2\ \ 2d+1\ \leqslant$，断言 .。推出 $10dr$ 」 ( ) P_u b( )( ) 。

现在假设 $Z_1\subseteq V(G)$ 的大小至多为 k_1，并设 $v\in V(P_i')-Q_i\subseteq\mathrm{Priv}_i$ 是 $N_G^d[Z_0\cup Z_1]$中的一个顶点。设 W 是 G中一条长度至多为 d 的 $v-(Z_0\cup Z_1)$ 路径。从 v 沿着 W 前进，令 w 为 W 在 $(\mathrm{Pub}_i\cup\{s,t\})\cup X\cup Z_0\cup Z_1$ 中的第一个顶点。我们考虑四种可能的情况，并证明每种情况都只会发生在数量有界的不同路径 P_i' 上，因此存在一个 $i\in[q]$，使得 $V(P_i')-Q_i$ 与 $N_G^d[Z_0\cup Z_1]$不相交。

1. $w\in\mathrm{Pub}_i\cup\{s,t\}$。由于 W 中没有更早的顶点属于 X，因此我们有 $v\in N_{G-X}^d[w]\subseteq Q_i$，矛盾。2. $w\in Z_0-X$。 $w\in\mathrm{Pub}_i\cup\{s,t\}$ 的情况已在上文处理；由于 W 中没有更早的顶点属于 X 或 Pub$_i\cup\{s,t\}$，因此我们有 dist$_{G-(X\cup\mathrm{Pub}_i\cup\{s,t\})}(v,\ X)\leqslant d$。这意味着 $w\in Z_0^i\subseteq Q_i$，矛盾。

3. $w \in X$。由于 W 中没有更早的顶点属于 X 或 $\mathsf{Pub}_i \cup \{s,t\}$，我们得出 $\mathsf{dist}_{G-(X\cup\mathrm{Pub}_i\cup\{s,t\})}(v, X) \leqslant d$。根据断言 4 4。.9，这种情况至多可以发生于 $p(10d^2h)^{d+2}\lambda$ 个 i 的取值中。

4. $w \in Z_1$。W 的情形 $w \in X \cup \mathsf{Pub}_i \cup \{s,t\}$ 已在上文处理，且 W 的更早顶点均不在 $X \cup \mathsf{Pub}_i \cup \{s,t\}$ 中，因此有 $\mathsf{dist}_{G-(X\cup\mathrm{Pub}_i\cup\{s,t\})}(v,w) \leqslant d$。对于任意 $w \in Z_1$，断言 4 84..8 表明，这种情况最多只会发生于 $(10d^2h)^{d+3}$ 条不同的路径 P_i 上，因此总共最多会有 $(10d^2h)^{d+1}k_1$ 个不同的 i 出现这种情况。

如果路径 P_i' 满足以下任一条件，我们就称其为坏路径：$|Q_i| > 10dr$，或者 $V(P_i') - Q_i$ 与 $N_G^d[Z_0 \cup Z_1]$ 不相交。利用断言 4 114..11 中的界以及上文最后两种情形中的界，可以对坏路径的数量给出上界。由于 q 大于这三个界之和 $(10d^2h)^{d+3}k_0/r + p(10d^2h)^{d+2}\lambda + (10d^2h)^{d+3}k_1$，因此存在一个 i，使得 $V(P_i') - Q_i$ 与 $N_G^d[Z_0 \cup Z_1]$ 和 $|Q_i| \leqslant 10dr$ 不相交。换言之，若令 $I \subseteq [q]$ 在 $|Q_i| \leqslant 10dr$ 时包含 i，那么返回路径 $q' = |I|$、P_i'、$i \in I$ 以及集合 Q_i，便满足引理的要求。 □

处理分隔集序列。接下来，我们将展示如何修改分隔集链，使得每个 Z 顶点的距离 d 邻域只会影响有界数量的顶点。我们还将再次利用该图是 $K_{3,h}$-子式自由图这一假设，限制链中每个顶点对分隔集的影响。

引理 *4.12.* 存在一个多项式时间算法：给定一个连通的 $K_{3,h}$-子式自由图 G、满足 $q \geqslant 2$ 且 $p \geqslant 100d\log_2 q$ 的正整数 p, q, d, λ、一对不同顶点 $s, t \in V(G) - \{s,t\}$，使得 $G[X]$ 至多有 λ 个连通分量，以及图 $G - X$ 中一条由大小至多为 $2q$ 的 (s,t)- 分隔集组成的链 $(C_1, \ldots, C_p)$，则对某个 $p' \leqslant p$，输出图 $G - X$ 中一条由大小至多为 $2q$ 的 (s,t)-分隔集组成的序列 $(C_1', \ldots, C_{p'}')$，满足以下条件：

1. 对于每个 $j \in [p']$ 和 $v \in V(G)$，

$$\left|V(C_j') \cap N_G^d[v]\right| \leqslant (10d^dh)^{2d+1}(1 + \lambda\log_2(4q)); \text{ and}$$

2. 对于大小至多为 k的每个集合 $Z \subseteq V(G)$，存在一个指标 $j \in [p']$ 使得

$$\left|V(C_j') \cap N_G^d[Z]\right| \leqslant (k/p) \cdot \log_2 q \cdot 50d(10d^dh)^{2d+1} \cdot \lambda.$$

证明。令

$$z := 2\log_2(2q+1) + 2d.$$

注意，由于 $d \geqslant 1$ 和 $q \geqslant 2$，因此 $z \leqslant 10d\log_2 q \leqslant p/10$。我们按以下方式根据分隔集 C_{z+d+1}, C_{2z+d+1}, C_{3z+d+1}, … 定义分隔集 C_1', C_2', C_3', …。令

$$J = [\lfloor (p - 2(d+1))/z \rfloor - 2] \qquad \text{and} \qquad \iota(j) = jz + d + 1.$$

注意，对于 $j \in J$，我们有 $d + 1 + z \leqslant \iota(j) \leqslant p - (z + d + 1)$。此外，根据 p,q, d和 z的界，我们有

$$\begin{aligned} |J| &\geqslant (p - 2(d+1))/z - 3 \geqslant (96/100)(p/z) - 3 \geqslant (96/100)(3/10)(p/z) \\ &\geqslant (96/100)(3/10)(p/(10d\log_2 q)) > p/(50d\log_2 q). \end{aligned}$$

我们按如下方式定义基于 $C_{\iota(j)}$的分隔集 C_j'。考虑定义在 $G - X$上的权重函数：

$$w_j(v) := 2^{\mathrm{dist}_{G-X}(v, C_{\iota(j)})}.$$

令 C'_j 为该权重函数下的最小权重（s，t）-分隔集。注意，$w(C_{\iota(j)}) = |C_{\iota(j)}| = 2q$，因此 $w(C'_j) \leqslant 2q$ 也成立；特别地，$|C'_j| \leqslant 2q$。我们指出，根据定义，s，$t \notin C'_j$，因此在选择 C'_j 时，它们的权重无关紧要。然而，最短路径可能经过 $\{s,t\}$，因此顶点 s 和 t 会影响其他顶点的权重。

首先我们说明，C'_j 的定义确保这些集合两两不交。事实上，$G-X$ 中的任何顶点都不可能同时靠近两个这样的集合。

断言 4.13。对于 $G-X$ 中的每个顶点 v，至多存在一个指标 $j \in J$ 使得距离 $\mathrm{dist}_{G-X}(v, C'_j) \leqslant d$。此外，对于 $v \in \{s,t\}$，不存在这样的 $j \in J$。

断言的证明。假设存在两个这样的值 $j_1 \neq j_2$，并且 $u_1 \in C'_{j_1}$ 和 $u_2 \in C'_{j_2}$ 在 $G-X$ 中与 v 的距离至多为 d。由于 $(C_1, \ldots\ldots, C_p)$ 构成 $G-X$ 中的一条 (s,t)-分隔集链，集合 $C_{\iota(j_1)}$ 和 $C_{\iota(j_2)}$ 在 $G-X$ 中彼此距离至少为 z。因此，不失一般性，我们可以假设距离 $\mathrm{dist}_{G-X}(v, C_{\iota(j_1)}) \geqslant z/2$。这意味着距离 $\mathrm{dist}_{G-X}(u_1, C_{\iota(j_1)}) \geqslant z/2 - d \geqslant \log_2(2q+1)$。但此时 $w(u_1) \geqslant 2q+1$，这与 $w(C'_{j_1}) \leqslant 2q$ 这一事实矛盾。

对于第二个陈述，假设对于某个 $u \in C'_j$，$\mathrm{dist}_{G-X}(s, u) \leqslant d$。由于 $\iota(j) \geqslant z+d+1$，我们有 $\mathrm{dist}_{G-X}(s, C_{\iota(j)}) \geqslant z+d+1$，因此 $\mathrm{dist}_{G-X}(u, C_{\iota(j)}) \geqslant z+1$。所以，我们有 $w_j(v) \geqslant 2^{z+1} > 2q$，这与事实 $w(C'_j) \leqslant 2q$ 矛盾。」

令 G' 为通过将 $G[X]$ 的每个连通分量收缩为一个顶点而从 G 得到的图；令 X' 为这些收缩顶点的集合。若顶点 $u \in C'_j$ 属于 $N^d_G[Z]$，则要么 v 与 $G-X$ 中 $Z-X$ 的某个顶点之间的距离至多为 d，要么 u 与 $G[X]$ 的至多 λ 个连通分量之一之间的距离至多为 d。换言之，要么 u 与 $G-X$ 中某个 $v \in Z-X$ 之间的距离至多为 d，要么 u 与 $G-(X-\{x'\})$ 中某个 $x' \in X'$ 之间的距离至多为 d（即不经过任何其他收缩顶点）。下述断言将用于给出一个统一论证，以界定 C'_j 中这两类顶点的数量；然而，该形式的断言只能界定在权重函数 w_j 下权重相同的顶点数量。注意，若 $v \notin X'$，则 $G'-(X'-\{v\})$ 就是 $G'-X'$；若 $v \in X'$，则 v 是 $G'-(X'-\{v\})$ 中 $G'-X'$ 之外的唯一顶点。

命题 4.14. 任取一个 $j \in J$。设 v 是 $V(G')$ 中的一个顶点，并设 $U \subseteq C'_j$ 是一个子集，使得对每个 $u \in U$，$w_j(u)$ 都相同，并且对每个 $u \in U$，$\mathrm{dist}_{G'-(X'-\{v\})}(v,u) \leqslant d$。那么我们有 $|U| \leqslant (10d^d h)^{2d}$。

断言的证明。假设 $|U| > (10d^d h)^{2d}$。若 $v \in U$，则从 U 中删去这个顶点；以下我们假设 $v \notin U$ 且 $|U| \geqslant (10d^d h)^{2d}$。

考虑 $G'-(X'-\{v\})$ 中从 v 到 U 的一棵最短路径树；这棵树的深度至多为 d。因此，如果每个节点的子节点都少于 $10dh^2 2^d$ 个，那么这棵树的节点数将少于 $d(10dh^2 2^d)^d < |U|$；这与假设矛盾。这意味着我们可以假设存在一个节点 v'，它至少有 $10dh^2 2^d$ 个子节点。令 $U' \subseteq U$ 由以下方式得到：对 v' 的每个子节点 v''，选取 U 中一个可从 v' 经由 v'' 在最短路径树中到达的顶点。对每个 $u' \in U$，令 W'_u 为最短路径树中的 $v'-u'$ 路径。注意，v' 是这些路径共有的唯一顶点。由于 $v \notin C'_j$，路径 W'_u 的长度至少为 1。

我们注意到两个技术细节。首先，我们声称，W'_u 中可能属于 X' 的顶点只有一个：其端点 v'，且只有在 $v = v'$ 时才如此。事实上，如果 $v \notin X'$，那么 W'_u 完全包含在 $G' - (X' - \{v\}) = G' - X'$ 中。如果 $v \in X'$，那么要么 $v' = v$，因此 $v \in X'$ 是 W'_u 的端点；要么 $v \neq v'$，但此时 v 不包含在路径 W'_u 中的任何一条路径里。特别是，这意味着 $W'_u - X'$ 是连通的。其次，W'_u 不能包含 s 或 t：根据断言 4 134..13，这些顶点到 C'_j 的距离在 $G'-X'$ 中大于 d（且只有 W'_u 的端点可以位于 $G'-X'$ 之外）。特别是，v' 不在 $\{s,t\}$ 中。

考虑图 $G^* = G - (X \cup (C'_j - U'))$。注意，$U'$是 G^*中的一个（s，t）-分隔集。令 s和 t 的容量为无穷大，并令每个其他顶点的容量为$w_j(v)$。如果 G^*中存在一个权重小于$w_j(U')$ 的（s，t）-分隔集 S，那么$(C'_j - U') \cup S$就是 $G-X$ 中的一个（s，t）-分隔集，其权重严格小于 $w_j(C'_j)$，这与C'_j的选取相矛盾。因此，根据流-割对偶，我们可以假设 G^* 中存在一个由 $s-t$ 条路径组成的集合 $\mathcal{P}$，使得$|\mathcal{P}| = w_j(U')$，且 G^*中的每个顶点 v至多属于 $\mathcal{P}$中的$w_j(v)$ 条路径。由于 U'是 G^*中权重为$w_j(U')$的一个（s，t）-分隔集，因此还可知，$\mathcal{P}$中的每条路径恰好包含 U' 中的一个顶点，且每个顶点 $u \in U'$恰好属于$w_j(u)$ 条这样的路径；令 $\mathcal{P}_u \subseteq \mathcal{P}$为这些路径构成的集合。注意，$\{\mathcal{P}_u \mid u \in U'\}$ 是 $\mathcal{P}$的一个划分。

我们按以下方式在顶点集 U' 上构造有向图 H。令 m 为每个 $u \in U'$ 对应的 $w_j(u)$ 的公共值。若 W'_{u_1} 与 $\mathcal{P}_{u_2}$ 中至少 $m/(h-1)$ 条路径相交，则引入一条边 $\overrightarrow{u_1u_2}$。我们观察到 $W'_{u_1} - X'$ 是连通的，因此 $W'_{u_1} - X'$ 的每个顶点到 u_1 的距离在 $G' - X' = G - X$ 中至多为 d。这意味着，对于每个 $v \in W'_{u_1} - X'$，$w_j(v) \leqslant 2^d \cdot m$。由此可知，$W'_{u_1}$ 至多与 $2^d \cdot dm$ 条路径相交。特别地，至多有 $2^d \cdot d(h-1)$ 个顶点 $u_2 \in U$，使得 W'_{u_1} 至少与 $\mathcal{P}_{u_2}$ 中的 $m/(h-1)$ 条路径相交。因此，u_1 的出度至多为 $2^d \cdot d(h-1)$。

根据引理 4 34。.3，有向图 H 有一个独立集 U''，其大小有如下界：

$$|U''| \geqslant |U'|/(1 + 2^d \cdot 2d(h-1)) \geqslant h.$$

对于每个 $u \in U''$，$t-1$ 中的每个集合 $W'_{u'}$，与 $u' \in U'' - \{u\}$ 中少于 $m/(h-1)$ 条路径相交。因此，我们可以选取一条路径 $P_u \in \mathcal{P}_u$，使其与每个 $W'_{u'}$ 不相交（对每个$u' \in U'' - \{u\}$均成立）。注意，对于任意 $u \in U''$，都有 $v' \notin P_u$；否则，$W'_{u'} \cap P_u$ 将包含任意 $u' \in U'' - \{u\}$（这里我们使用了 $|U''| \geqslant 2$）。

直观而言，我们希望通过以下一系列收缩得到一个 $K_{3,\ h}$-子式：(1) 将 P_u 的 $s-u$ 部分（不包括 u）收缩到 s；(2) 将 P_u的 $u-t$ 部分（不包括 u）收缩到 t；(3) 将每个 W'_u（不包括 u）收缩到 v'。然而，由于 W'_u 可能与 P_u 相交，这些收缩必须谨慎进行（见图 33）。我们声称，对每个 $u \in U''$，在 $G' - (X' - v)$ 中进行以下收缩，就能得到一个 $K_{3,\ h}$-子图，其中一侧为 $\{s, t, v\}$，另一侧为 U'' 的顶点：

- 首先，从 v'开始，令顶点 u'为 W'_u中属于 P_u的第一个顶点（注意，W'_u包含 $u \in P_u$）。将 W'_u 的 $u'-u$ 子路径收缩为 u。注意，$v' \notin P_u$意味着 $u' \neq v'$。将 W'_u 中从 v' 到 u'的前一个顶点之间的子路径收缩为 v'。

- 收缩后，$P_u - u$ 至少由两个连通分量组成（如果 W'_u 的收缩子路径与 P_u 相交多次，则连通分量可能更多）。将子图 $P_u - u$ 中包含 s 的连通分量收缩为顶点 s，并将 $P_u - u$ 中包含 t的子路径收缩为顶点 t。

我们来验证收缩到不同顶点（即某个 $u \in U''$、s、t 或 v'）的集合两两不相交。收缩到不同顶点 $u \in U''$的集合两两不相交，因为对于不同的 $u \in U''$，路径 $W'_u - v'$彼此不相交。由于当 $u_1 \neq u_2$时，P_{u_1}与 W'_{u_2}不相交，因此 W'_{u_2}中没有顶点被收缩到 s或 t。由于 U'是 G^*中的一个（s，t）-分隔集，且每条路径 P_u都与 U'恰好相交于一个顶点，因此对于任意 u_1，$u_2 \in U''$，P_{u_1} 的 $s-u_1$ 子路径都不可能与 P_{u_2} 的 $t-u_2$ 子路径相交。由此可见，这些收缩确实在 $G' - (X' - v)$中得到一个 $K_{3,\ h}$ 子图。由于 $X' - v$的 G' –是 G的子式，这与假设 G 是 $K_{3,\ h}$-子式自由矛盾。」

利用断言 4 144..14，很容易界定分隔集 C'_j中与某个固定顶点 v 在 $G - X$.

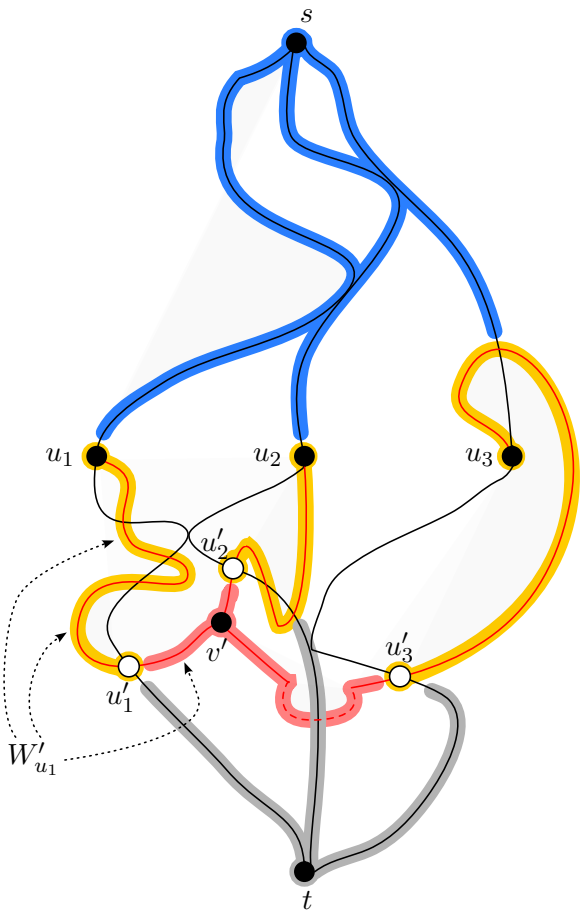

图 3：通过收缩得到一个$K_{3,\ 3}$-子式。黄色标出的路径收缩为 u_1、 u_2 和 u_3。蓝色、灰色和红色标出的路径分别收缩为 s、 t 和 v'。

命题 4.15. 对于 $G - X$ 中的每个顶点 v 以及每个 $j \in J$，至多有 $(10d^d h)^{2d+1}$ 个顶点 $u \in C'_j$ 满足 距离$_{G-X}(v, u) \leqslant d$。

断言的证明。令 $U_1 \subseteq C'_j$ 为这些顶点的集合，并假设这样的顶点多于 $(10d^d h)^{2d+1}$ 个。由于 U_1 中的每个顶点在 $G - X$ 中到 v 的距离至多为 d，因此它们在 $G - X$ 中彼此之间的距离至多为 $2d$。这意味着 U_1 上 w_j 的不同取值至多有 $2d + 1$ 个，因此存在一个大小大于 $|U_1|/(2d + 1) \geqslant (10d^d h)^{2d}$ 的子集 $U_2 \subseteq U_1$，使得对于每个 $u \in U_2$，$w_j(u)$ 都相同。然而，断言 4.14 和 4.14. 将 U_2 的大小限制为 $(10d^d h)^{2d}$，矛盾。」

同样，断言 4.14 和 4.14. 使我们能够界定 C'_j 中靠近 X 的顶点数量 G.

命题 4.16。 对于每个 $j \in J$，满足$\mathrm{dist}_G(u, X) \leqslant d$的顶点 $u \in C'_j$ 至多有 $\lambda(10d^d h)^{2d} \log_2(4q)$ 个。

断言的证明。令 $U_1 \subseteq C'_j$ 为这些顶点的集合，并设 $|U_1| > \lambda(10d^d h)^{2d} \log_2(4q)$。回顾一下，对于每个 $u \in C'_j$，$w_j(u) \leqslant 2q$，因此 $u \in U_1$ 的 $w_j(u)$ 至多有 $\log_2(4q)$ 个不同取值。因此，存在一个大小至少为 $|U_1|/\log_2(4q) \geqslant \lambda(10d^d h)^{2d}$ 的子集 $U_2 \subseteq U_1$，使得对于每个 $u \in U_1$，$w_j(u)$ 都相同。

对于任意 $u \in C'_j$，条件 $\mathrm{dist}_G(u, X) \leqslant d$ 也可以写作 $\mathrm{dist}_{G'}(u, X') \leqslant d$：在 G 中，靠近 X 意味着靠近 G 中的一个收缩顶点。由于 X' 中至多有λ 个顶点，这意味着存在一个 $x' \in X'$ 以及一个大小大于$|U_2|/\lambda = (10d^d h)^{2d}$ 的子集 $U_3 \subseteq U_2$，使得对每个 $u \in U_3$，都有 $\mathrm{dist}_{G'-(X-\{x'\})}(u, x') \leqslant d$。然而，引理 4.14 和 4.14. 将 U_3 的大小限制为 $(10d^d h)^{2d}$，矛盾。」

命题 4.15 和 4.16 使我们能够界定 $|C'_j \cap N^d_G[v]$ 对于任意 $v \in V(G)$。

断言 4.17. 对于每个 $j \in J$和 $v \in V(G)$，满足$\mathrm{dist}_G(v, u) \leqslant d$的顶点 $u \in C'_j$ 至多有$(10d^d h)^{2d+1}(1 + \lambda \log_2(4q))$个。

断言的证明。设 $u \in C'_j$ 是这样一个顶点，Q 是从 u 到 v的一条最短路径。若$V(Q) \cap X = \emptyset$，则 $\mathrm{dist}_{G-X}(u, v) \leqslant d$，根据断言 4.15.，这样的顶点 u 至多有 $(10d^d h)^{2d+1}$ 个。否则，$\mathrm{dist}_G(u, X) \leqslant d$，根据断言 4.16.，这样的顶点$u$ 至多有 $\lambda(10d^d h)^{2d} \log_2(4q)$个。断言得证。」

断言 4.17 确立了引理陈述的第一点。对于第二点，令 $C^* := \bigcup_{j \in J} C'_j$ (注意，根据断言 4.13)，这是一个不交并)，并令 $Z \subseteq V(G)$ 为一个至多包含 k 个顶点的集合。我们要证明 $|C^* \cap N^d_G[Z]|$的一个上界。令 $u \in C^* \cap N^d_G[Z]$，并假设 $u \in C'_j$。以下两种情况至少有一种适用于 u:

1. $\mathrm{dist}_{G-X}(u, z) \leqslant d$对某个 $z \in Z$成立。根据断言 4.13.，顶点 z在 $G - X$ 中与 C'_j的距离至多为 d的情况，对应的 $j \in J$至多只有一个。对于这个 j，根据断言 4.15.，顶点 z与 C'_j中至多 $(10d^d h)^{2d+1}$ 个顶点的距离至多为 d。因此，总共至多有 $k \cdot (10d^d h)^{2d+1}$ 个顶点 $u \in C^*$ 满足这种情况。
2. $\mathrm{dist}_G(u, X) \leqslant d$. 根据断言 4.16.，最多有 $\lambda(10d^d h)^{2d} \log_2(4q)$ 个这样的顶点对应每个 $j \in J$，因此总共最多有 $|J| \cdot \lambda(10d^d h)^{2d} \log_2(4q)$ 个这样的顶点。

因此，我们可以将 $|C^* \cap N_G^d[Z]|$ 界定为 $k\cdot(10d^dh)^{2d+1}+|J|\cdot\lambda(10d^dh)^{2d}\log_2(4q)$。由于集合 C'_j 两两不相交，且此前已得到界 $|J| \geqslant p/(50d\log_2 q)$，因此存在一个 $j \in J$ 使得

$$\begin{aligned}|C'_j \cap N_G^d[Z]| &\leqslant k(10d^dh)^{2d+1}/|J| + \lambda(10d^dh)^{2d}\log_2(4q)\\ &\leqslant (k/p)\cdot 50d(10d^dh)^{2d+1}\log_2 q + \lambda(10d^dh)^{2d}\log_2 4q\\ &\leqslant (k/p)\cdot\log_2 q\cdot 50d(10d^dh)^{2d+1}\cdot\lambda.\end{aligned}$$

这就完成了证明。 □

定理 2 5 的证明25。。。现在我们可以完成定理 2 5 的证明25。。。简而言之，若我们有一个 $(p,\ q)$-结构，且其 p 很大，那么引理 4 124。。12 会给出所需的分隔集；而当 p 很小时，引理 4 64。。6 会给出所需的近乎不相交路径集合。

定理 2 52 5 的证明。令 p 和 q 为给定对象。使用定理 4 24..2 的算法，在 $(G-X,s,t)$ 中为 $q_0=q$ 找到一个 (p_0,q_0)-结构。若 $p_0 \geqslant p$，则我们得到一条由 (s,t)-分隔集组成的链 $(C_1,\ldots,C_p)$，这些分隔集位于 $G-X$ 中，且每个分隔集的大小至多为 q。对这条链应用引理 4 124..12，可得到一个分隔集序列，其中一个分隔集与 $N_G^d[Z]$ 的交集满足所需的界。

如果 $p_0<p$，那么我们在4..6 上调用引理 4 6$(p_0,\ q_0)$-结构，其中 $r=p>p_0$。注意$q_0=q\geqslant(p+k_0/p+k_1)(10d^2h)^{d+3}\lambda>(p_0+k_0/r+k_1)(10d^2h)^{d+3}\lambda$ 成立。因此，我们得到了所需的路径集合 P_i 和集合 Q_i，其中 $|Q_i|\leqslant 10dr=10dp$ 对每个 i都成立。 □

5 分隔集改进过程

本节我们将 Nederlof 的“分隔集改进”过程推广到排除固定 minor 的图 [43]。（回顾记号 $[d\geqslant 1]$，其值等于1，当 $d\geqslant 1$ 时；否则为0。）

定理 5.1。对于任意固定整数 c、$d\geqslant 0$ 和 $h\geqslant 4$，存在一个满足以下规范的算法。

记 $H=K_h$ *if* $d=0$ *and* $H=K_{3,h-3}$ *if* $d\geqslant 1$。给定输入后，算法接收一个H-子式自由图 G、一个整数 $k\geqslant 2$、一个集合 $Z_0\subseteq V(G)$、G 中的一个分离(A°,B°)，满足对每个 $v\in V(G)$都有 $|N_G^d[v]\cap A^\circ\cap B^\circ|\leqslant c$；此外还接收 $G[A^\circ\cap B^\circ]$的一个生成森林 F°，其连通分量少于 h 个，且最大度至多为 h。记 $k_0:=|Z_0|$，以及$\ell:=|A^\circ\cap B^\circ|$。

~该算法计算出一个元组族 $\mathcal{S}$，其中的元组(A,B,C,C) 由 $V(G)$ 的子集构成，并满足以下性质：

~*1.* 对于每个$(A,B,C,C)\in\mathcal{S}$，都有

- (A,B) 是 G,~ 中的分离
- $C\subseteq C\subseteq A\cap B$,
- $|C|\leqslant \ell+\mathcal{O}_{d,h}((k+k_0/\sqrt{k})$ 对数 $k)$,
- $A\cap B\cap N^d[Z_0]\subseteq C$,
- $|\widetilde{C}|\leqslant\mathcal{O}_{d,\ h}((\sqrt{k}+\log^{[d\geqslant 1]}(k+k_0))\log\ k)$，以及
- 对于每个 $v\in V(G)$，$|(C-\widetilde{C})\cap N_G^d[v]|=\mathcal{O}_{d,\ h}(c\log(k+k_0)\log\ k)$。[3]

[3]请注意，对于 $d=0$，这一条件显然成立。

2. 对于每个正整数 θ，设 $W \subseteq V(G)$，以及由两两不交的 d-簇组成的族 $\mathcal{Z}$，其中 G 满足

- $|W| \geqslant 4\theta$,
- $|A^\circ \cap W| \geqslant \theta$, $|B^\circ \cap W| \geqslant \theta$，以及
- $|\mathcal{Z}| \leqslant k$,

~存在一个元组 $(A, B, C, C) \in \mathcal{S}$，使得

- $|A \cap W| \geqslant \theta$, $|B \cap W| \geqslant \theta$,
- $A \cap B \cap \bigcup \mathcal{Z} \subseteq C$,
- $|A \cap B \cap \bigcup \mathcal{Z}| = |C \cap \bigcup \mathcal{Z}|$ 有界于

$$\mathcal{O}_{d,h}\left(c\left(\sqrt{k} + k_0/\sqrt{k}\right)\log^{[d\geqslant 1]}(k+k_0)\log k\right)$$

- 此外，如果

$$|\{K \in \mathcal{Z} \mid K \cap A^\circ \cap B^\circ \neq \emptyset\}| > h(h+1)\sqrt{k}$$

then

$$|\{K \in \mathcal{Z} \mid K \cap C \neq \emptyset\}| \geqslant \sqrt{k}/c.$$

3. 族 $\mathcal{S}$ 的大小有界于

$$(\mathcal{O}_{d,h}((k+k_0)\ell))^{\mathcal{O}_h(\log k)}.$$

4. 运行时间的上界是关于 G 的大小以及上一点中 $\mathcal{S}$ 大小上界的多项式。

此外，该算法可以修改为多项式时间随机算法：~给定相同的输入，采样一个元组 $(A, B, C, C) \in \mathcal{S}$，使得每个元组被采样的概率至少为

$$(\mathcal{O}_{d,h}((k+k_0)\ell))^{-\mathcal{O}_h(\log k)}.$$

如果在输入中额外给定集合 W 和上述的 θ，则该算法只会从满足 $|A \cap W| \geqslant \theta$, $|B \cap W| \geqslant \theta$（即这些不等式以概率 1成立）的~个元组 (A, B, C, C) 中进行采样。

~注意，对于情形 $d = 0$，我们总可以取 $c = 1$ 并使用 $C = \emptyset$。换言之，~常数 c 和元素 C 仅与定理的距离 $d \geqslant 1$ 版本有关。

我们需要以下工具箱来操作 $G[A^\circ \cap B^\circ]$ 的生成森林。

固定一个整数 $\delta \geqslant 1$。假设给定一个图 G，并且其中存在某个未知族 $\mathcal{Z}$，它包含至多 k 个两两不交的 d- 簇。对于一对 (F, ζ)，其中 F 是最大度至多为 δ 的森林，而 ζ 为 F 的每个连通分量 J 指定一个猜测值 $\zeta(J) \in \{0, 1, \ldots, ck\}$，作为 $|J \cap \bigcup \mathcal{Z}|$ 实际值的猜测。若 $\zeta(J) = |J \cap \bigcup \mathcal{Z}|$ 对 F 的每个连通分量 J 都成立，则称带标注森林与 $\mathcal{Z}$ 一致。我们有时将 $\zeta(J)$ 称为连通分量 J 的权重，而 (F, ζ) 的权重是 F 所有连通分量权重之和。

我们需要以下观察结果。

引理 5.2。若 T 是一棵最大度不超过 δ 的树，且 $Z \subseteq V(T)$ 的大小至少为 2，则存在一条边 $e \in E(T)$，使得 $T - \{e\}$ 的两个连通分量都至少包含 $\frac{|Z|}{\delta+1}$ 个 Z 的顶点。

——

证明。 首先，将 T 的每条边 e 从 $T-\{e\}$ 中包含较少 Z 顶点的连通分量指向包含较多顶点的连通分量（若数量相同，则任意选择方向）。由于 $|E(T)| = |V(T)| - 1$，存在一个出度为 0 的顶点 v。因此， $T-\{z\}$ 的每个连通分量最多包含 $|Z|/2$ 个 Z 的顶点。于是，连接 z 与 $T-\{z\}$ 中包含最多 Z 顶点的连通分量的边 e 满足该结论的要求。这里，我们使用假设 $|Z| \geqslant 2$ 来确保该连通分量至少包含一个 Z 的顶点，即 $Z \not\subseteq \{v\}$。

□

给定带标注森林 (F, ζ)，我们将 $\mathcal{Z}$-分裂的（分支）操作定义如下：选取 J 作为 F 中具有最大 $\zeta(J)$ 的连通分量，随机选择一条边 e，该边属于 J；并猜测 $|J_1 \cap \bigcup \mathcal{Z}|$ 和 $|J_2 \cap \bigcup \mathcal{Z}|$，分别对应 $J-\{e\}$ 的两个连通分量 J_1, J_2，且仅考虑不小于 $\zeta(J)_{\delta+1}$ 的值。更准确地说，我们随机采样一个生成森林 (F', ζ')，其中 $\overline{F'} = F - \{e\}$，而 e 是从 J 中随机选取的一条边；对于与 J 不同的连通分量 J'， $\zeta'(J') = \zeta(J')$；对于 $J-\{e\}$ 的两个连通分量 J_1, J_2，$\zeta'(J_1)$ 的值从 $\{\lceil \zeta(J)_{\delta+1} \rceil, \ldots, \zeta(J) - \lceil \zeta(J)_{\delta+1} \rceil\}$ 中均匀随机选取，而 $\zeta'(J_2)$ 则设为 $\zeta(J) - \zeta'(J_1)$。仅当 $\zeta(J) > 1$ 时，才定义上述操作；否则，可能不存在满足上述条件的 $\zeta'(J_1)$ 取值。关于 $Z := J \cap \bigcup \mathcal{Z}$ 的引理 5 25..2 保证，若 (F, ζ) 与 $\mathcal{Z}$ 一致，则总能选择 e 和 $\zeta'(J_1)$，使得 (F', ζ') 与 $\mathcal{Z}$ 一致。

注意， $\mathcal{Z}$-分裂最多有 $|E(J)| \cdot (1 + \zeta(J)) \leqslant (ck+1)|E(F)|$ 种针对 (F', ζ') 的选择。我们只会使用作为输入森林 F° 的子图的森林 F，因此在这种情况下 $|E(F)| < \ell$，最多有 $(ck+1)(\ell-1)$ 种选择。

我们用下面这个显然的引理概括分裂操作。

引理 5.3。 假设 (F, ζ) 与 $\mathcal{Z}$ 一致。

如果 F 中存在一个连通分量 J，满足 $\zeta(J) > 1$，那么对 (F, ζ) 执行 $\mathcal{Z}$-分裂操作所得的输出 (F', ζ') 也与 $\mathcal{Z}$ 一致，且概率至少为

$$((ck+1)(\ell-1))^{-1}.$$

现在我们继续证明引理 5151。。。

引理 *5 1* 的证明*51*。。。 下面描述该随机多项式时间算法。枚举算法只需做简单修改，将每次随机猜测替换为分支即可。

令 $\delta \leqslant h$ 为 F°的最大度。回顾一下， $\ell = |A^\circ \cap B^\circ| = |V(F^\circ)|$。

该算法将描述为一种递归算法：每一步都在多项式时间内运行，并从多项式数量（关于 $k+\ell$）的选项中随机猜测一个。我们将确保递归深度有界于 $3h(\delta+1)\log k$。这将给出所承诺的运行时间和成功概率界限。为便于描述，我们按照定理陈述固定 θ、 W 和 $\mathcal{Z}$。所有猜测步骤实际上都是关于 W 和 $\mathcal{Z}$ 的某些性质进行猜测，并且我们确保至少有一个选项中的猜测是正确的。如果算法已知 W 和 θ，那么在涉及 W 和 θ 性质的猜测步骤中，我们就知道应选择哪种情况，并确定性地作出选择。

初始猜测。 回顾一下， F° 是输入给定的 $G[A^\circ \cap B^\circ]$ 的生成森林， F°有 $\lambda^\circ < h$ 个连通分量。在开始递归之前，对于F°的每个连通分量，我们猜测该分量中包含多少个 $\bigcup \mathcal{Z}$的顶点。根据我们的假设，每个 $K \in \mathcal{Z}$在 $A^\circ \cap B^\circ$中至多有 c个顶点；因此， $|A^\circ \cap B^\circ \cap \bigcup \mathcal{Z}| \leqslant ck$。这一步将 F°转化为一个带标注森林（F°， ζ°），它以至少为（$ck+1$）$^{-\lambda^\circ}$的概率与 $\mathcal{Z}$一致。下文中，我们假设情况确实如此。

如果带标注森林（F，ζ）的一个连通分量 J满足 $\zeta(J) > (\delta+1)\sqrt{k}$，我们称其为 $\mathcal{Z}$-重的；如果 $\zeta(J) > \sqrt{k}$，则称其为$\mathcal{Z}$-显著的。如果一个带标注森林包含一个 $\mathcal{Z}$-重的连通分量，则称其为 $\mathcal{Z}$-重的。

现在执行以下自环：当 (F°, ζ°) 包含少于 h 个连通分量且至少一个 $\mathcal{Z}$-重连通分量时，执行分裂操作。注意，此操作会严格增加 F° 的连通分量数量，而分裂操作产生的两个新连通分量都是 $\mathcal{Z}$-显著的。因此，该自环最多运行 $h - \lambda^\circ$ 次迭代。此外，由于 $\lambda^\circ < h$，除非 (F°, ζ°) 一开始就不是 $\mathcal{Z}$-重的，否则至少会运行一次迭代。

令 (F^1, ζ^1) 为最终输出。注意，(F^1, ζ^1) 至多有 h 个连通分量；如果它是 $\mathcal{Z}$-重的，那么它恰好包含 h 个连通分量，且至少有两个 $\mathcal{Z}$-显著连通分量（因为自环至少运行了一次）。最后，注意到 (F^1, ζ^1) 与 $\mathcal{Z}$ 一致的概率至少为

$$(ck+1)^{-\lambda^\circ} \cdot ((ck+1)(\ell-1))^{-(h-\lambda^\circ)} \geqslant ((ck+1)(\ell-1))^{-h}.$$

如果 (F^1, ζ^1) 不是 $\mathcal{Z}$-重的，那么我们输出 $(A^\circ, B^\circ, C := A^\circ \cap B^\circ, \widetilde{C} := \emptyset)$ 并终止。也就是说，我们输出初始分离，其中 C 是整个割集 $A^\circ \cap B^\circ$，且 $C = \emptyset$。请注意，如果 (F^1, ζ^1) 与 $\mathcal{Z}$ 一致，那么

$$\left|A^\circ \cap B^\circ \cap \bigcup \mathcal{Z}\right| = \left|V(F^1) \cap \bigcup \mathcal{Z}\right| \leqslant h(\delta+1)\sqrt{k} = \mathcal{O}_h(\sqrt{k}).$$

因此，输出满足所有要求的性质。

我们注意到，如果满足关于 $A^\circ \cap B^\circ \cap \bigcup \mathcal{Z}$ 的附加条件，该循环就不会过早终止。

断言 5.4。如果

$$|\{K \in \mathcal{Z} \mid K \cap A^\circ \cap B^\circ \neq \emptyset\}| > h(h+1)\sqrt{k},$$

那么 (F^1, ζ^1) 是 $\mathcal{Z}$-重的。

断言的证明。上述条件意味着 $|A^\circ \cap B^\circ \cap \bigcup \mathcal{Z}| > h(\delta+1)\sqrt{k}$，进而意味着在上述循环的每一步，当前森林的至多 h 个连通分量 J 中，有一个的权重大于 $(\delta+1)\sqrt{k}$。」

我们假设 (F^1, ζ^1) 是 $\mathcal{Z}$-重的。特别地，它恰有h 个连通分量，且至少有两个 $\mathcal{Z}$-显著连通分量。

主过程。现在我们可以开始递归；控制流示意图见图 44。一次递归调用的输入将包括 G 的一个分离 (A, B)、$A \cap B$ 的一个子集元组 $(C, D, \widetilde{C})$，其中 $\widetilde{C} \subseteq C$, $C \cap D = \emptyset$, $D \cap N_G^d[Z_0] = \emptyset$，以及一个带标注森林 (F, ζ)，其中 F 是 $(A \cap B) - (C \cup D)$ 的生成森林，并满足：

- F 恰有 h 个连通分量；
- 中至少有两个 F 的连通分量是 $\mathcal{Z}$-显著的；
- F 是 F^1 的子图；
- 的每个连通分量要么是 F^1 的一个连通分量，要么是 $\mathcal{Z}$-显著的。

在根调用中，我们令 $(A, B) = (A^\circ, B^\circ)$、$C = D = \widetilde{C} = \emptyset$ 和 (F, ζ) 为 (F^1, ζ^1)。如果递归调用的深度大于 $\xi := 3h(\delta+1)\log k$，则无需执行任何操作即可终止。我们将确保深度为 j 的调用满足

- $|C| \leqslant \ell + j \cdot \mathcal{O}_{h,d}(k + k_0/\sqrt{k})$;

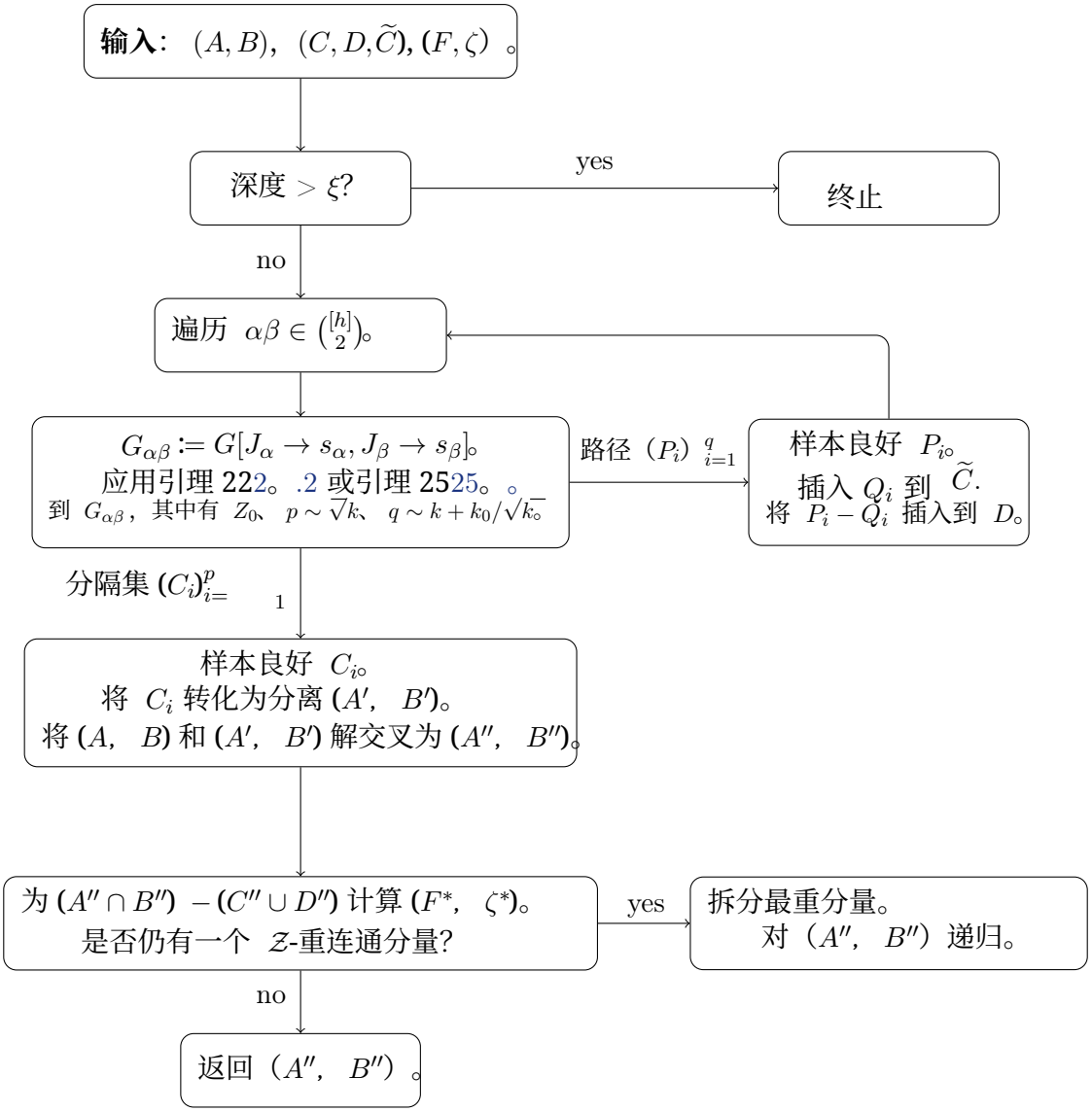

图 4：主过程的控制流示意图。

- $|\widetilde{C}|\leqslant j\cdot\mathcal{O}_{h,d}(\sqrt{k}$对数 $k)$;
- 对于每个 $v\in V(G)$，我们都有 $|N_G^d[v]\cap(C-\widetilde{C})|\leqslant j\cdot\mathcal{O}_{h,\ d}(\log(k+k_0))$。

这显然对根调用成立。

我们称深度为 j 的调用是一致的，当且仅当

- $|A\cap W|,|B\cap W|\geqslant\theta$;
- $|C\cap\bigcup\mathcal{Z}|$ 受以下值限制

$$\begin{cases}j\cdot\mathcal{O}_{h,d}(\sqrt{k}+k_0/\sqrt{k}), & \text{if } d=0,\\ j\cdot\mathcal{O}_{h,d}((\sqrt{k}+k_0/\sqrt{k})\log(k+k_0)), & \text{if } d\geqslant 1,\end{cases}$$

- $D\cap\bigcup\mathcal{Z}=\emptyset$；以及

- $(F,\ \zeta)$与 $\mathcal{Z}$一致。

注意，由递归之前的猜测正确这一假设可知，根调用是一致的。

现在考虑一个以 $((A,\ B),\ (C,\ D,\ C),\ (F,\ \zeta))$ 为输入的递归调用，并假设该调用是一致的。首先，我们猜测 $|A\cap W|$ 和 $|B\cap W|$ 中哪一个更大。（如果已知 θ 和 W，这一步便是确定性的。）在下面的描述中，我们假设 $|A\cap W|\geqslant|B\cap W|$，因此特别有 $|A\cap W|\geqslant|W|/2\geqslant 2\theta$；另一种情况是对称的。

内部自环。令 J_1、……、J_h 为 F 的连通分量。我们初始化 $C':=\emptyset$ 和 $D':=\emptyset$；我们按任意选定的顺序遍历所有对 $\alpha\beta\in\binom{[h]}{2}$。我们将维持一个不变量：$C'$ 和 D' 是 $V(G)-V(F)$ 的不相交子集，并且 $G[C'\cup D']$ 至多有 ι 个连通分量，其中 ι 是目前已执行的迭代次数。对于固定的一对 $\alpha\beta$，按如下步骤进行。

令 $G_{\alpha\beta}$ 为图 G，将 J_α 和 J_β 收缩为单个顶点后得到的图，我们将其记为s_α和s_β。注意，$G_{\alpha\beta}$ 是 G的一个子式。令 $X=C'\cup D'\cup(V(F)-(J_\alpha\cup J_\beta))$。对于 $d=0$，我们将引理 2 22..2 应用于 $G_{\alpha\beta}-X$，其中 $(s,t)=(s_\alpha,s_\beta)$，Z_0、$p=\lceil\sqrt{k}\rceil$ 和 $q=k+\lceil k_0/\sqrt{k}\rceil+1$。对于 $d\geqslant 1$，我们将定理 2 52..5 应用于 $G_{\alpha\beta}$，其中 $(s,t)=(s_\alpha,s_\beta)$，X、Z_0，

$$p=\lceil\sqrt{k}\rceil+\Theta_{d,h}(\log(k+k_0)),$$
$$q=\Theta_{d,h}(k+k_0/\sqrt{k})$$

等于定理允许的 Z_0的最小值、大小至多为 k 的集合 Z 以及 $\lambda:=\binom{h+1}{2}$。（注意，$G[X]$ 至多有 $\iota+h<\binom{h+1}{2}$ 个连通分量。）

首先，假设返回了由集合 $(Q_1,\dots,Q_q)$ 给出的 (s_α,s_β) 路径序列 $(P_1,\dots,P_q)$。根据引理 2 22..2 和引理 2 52..5 的保证，存在 $i\in[q]$，使得 $V(P_i)-Q_i$ 中没有顶点属于 $\bigcup\mathcal{Z}$ 或 $N_G^d[Z_0]$。在所有满足 $V(P_i)-Q_i$ 与 $N_G^d[Z_0]$ 不相交的索引 i 中，随机猜测索引 $i\in[q]$。记 $P^{\alpha\beta}:=P_i$。将 $Q_i-\{s_\alpha,s_\beta\}$ 插入 $\widetilde{C'}$，并将 $V(P_i)-(Q_i\cup\{s_\alpha,s_\beta\})$ 插入 D'。这里，端点 s_α 和 s_β 既不放入 C'，也不放入 D'——它们在 G 中并不存在——因此我们维持不变量：C' 和 D' 是 $V(G)-V(F)$ 的不相交子集。此外，插入 $C'\cup D'$ 的新顶点构成路径 $P_i-\{s_\alpha,s_\beta\}$，因此 $G[C'\cup D']$ 的连通分量数至多增加一。注意，此步骤中我们将 $\mathcal{O}_{d,h}(\sqrt{k}+\log^{[d\geqslant 1]}(k+k_0))$ 个顶点放入 $\widetilde{C'}$。我们维持 $D'\cap N_G^d[Z_0]=\emptyset$；此外，若对 i 的猜测正确，我们还维持 $D'\cap\bigcup\mathcal{Z}=\emptyset$，并且将 $\bigcup\mathcal{Z}$ 中的 $\mathcal{O}_{d,h}(\sqrt{k}+\log^{[d\geqslant 1]}(k+k_0))$ 个顶点插入 $\widetilde{C'}$。

在继续描述第二种情形——即返回分隔集链时——我们先注意到，这种情形最终必然会发生。

命题 5.5. 对于某个 $\alpha\beta\in\binom{[h]}{2}$，可以找到一条链$(C_1,\dots,C_p)$。

断言的证明。反证，假设对于所有 $\alpha\beta\in\binom{[h]}{2}$，迭代都持续进行，并且每一步都找到一个路径序列，并定义路径 $P^{\alpha\beta}$。由于在 $G^{\alpha\beta}$ 中，我们找到排除 $X=$ 的 条路径（$C'\cup D'$）∪（V（F）−（$J_\alpha\cup J_\beta$））$,$ 并且 J_α、J_β 分别收缩到 s_α 和 s_β 上，而 $P^{\alpha\beta}$ 的内部顶点被插入 $\widetilde{C'}\cup D'$ 中，因此，$1\leqslant i\leqslant h$ 对应的集合 K_i 以及 $\alpha\beta\in\binom{[h]}{2}$ 对应的路径 $P^{\alpha\beta}$ 构成了 G 中 K_h 的一个子式模型。这就产生了矛盾。」

让我们分析这个自环，直到找到一条分隔集链。对于每个 $\alpha\beta$，我们在出现路径序列（$P_1,\dots,P_q$）时向 $\widetilde{C'}$ 添加 $\mathcal{O}_{d,\ h}(\sqrt{k}+\log^{[d\geqslant 1]}(k+k_0))$ 个顶点，

找到后，我们有

$$|\widetilde{C}'| \leqslant \left(\binom{h}{2}-1\right)\mathcal{O}_{d,h}\left(\sqrt{k}+\log^{[d\geqslant 1]}(k+k_0)\right) \leqslant \mathcal{O}_{d,h}\left(\sqrt{k}+\log^{[d\geqslant 1]}(k+k_0)\right). \tag{1}$$

此外，$D' \cap N_G^d[Z_0] = \emptyset$ 并且，如果猜测正确， $D' \cap \bigcup \mathcal{Z} = \emptyset$。

汇总 本次递归调用中迄今为止所有猜测的结果，猜测正确的概率为 a-
概率至少为east

$$\frac{1}{2}\cdot(\mathcal{O}_{d,h}(k+k_0))^{-\left(\binom{h}{2}-1\right)} \geqslant (\mathcal{O}_{d,h}(k+k_0))^{-\binom{h}{2}+1}. \tag{2}$$

从分隔集链中提取一个良好的分离。现在，我们继续描述返回一个由 (s_α, s_β)-分隔集构成的链 $(C_1,\ldots,C_p)$ 时的算法。我们终止对这些对 $\alpha\beta \in \binom{[h]}{2}$ 的迭代，并按如下方式继续。

根据 由引理 2 22..2 和引理 2 52..5，存在 $i \in [p]$ 满足 $|C_i \cap \bigcup \mathcal{Z}|$ 界 d
by

$$\mathcal{O}_{d,h}\left(\frac{k+k_0}{p}\log^{[d\geqslant 1]}(k+k_0)\right) \leqslant \mathcal{O}_{d,h}\left((\sqrt{k}+k_0/\sqrt{k})\log^{[d\geqslant 1]}(k+k_0)\right).$$

我们随机猜测索引 $i \in [p]$ ，并在下文中假设该猜测是正确的 .

可想 $D' \cup (V(\widetilde{F}) - (J_\alpha \cup J_\beta))$。按照如下方式构造 G 的分离 (A', B')，对应于 C_i ：令 $A' \cap B' = C_i \cup X$，$A' - B'$ 为 J_α 以及从 s_α 可达的 $G_{\alpha\beta} - (C_i \cup X)$ 的所有顶点，并令 $B' - A'$ 为 J_β 以及 $G_{\alpha\beta} - (C_i \cup X)$ 中其余的顶点。由于$|A \cap W| \geqslant 2\theta$，我们有

$$|W \cap (A' \cap A)| \geqslant \theta \quad \text{or} \quad |W \cap (B' \cap A)| \geqslant \theta. \tag{3}$$

我们随机猜测 (33) 中哪一个不等式成立。（如果算法已知 W 和 θ ，则这一步是确定性的。）在下述描述中，不失一般性地假设$|W \cap (A' \cap A)| \geqslant \theta$；另一种情况是对称的。

令 $Y = V(F) - J_\beta$。我们将 (A', B') 和 (A, B) 解交叉为 G 的一个分离 (A'', B'')，如下所示：

$$A'' := Y \cup (A' \cap A), \text{ and}$$
$$B'' := Y \cup (B' \cup B).$$

也就是说，我们以标准方式将 (A', B') 和 (A, B) 解交叉为 $(A' \cap A, B' \cup B)$，但人为地将 Y 保留在该分离的两侧。

显然 ，(A'', B'') 是 G 的一个分离。此外，由于 $J_\beta \subseteq B' - A'$，我们有 $J_\beta \subseteq B'' - A''$ and

$$A'' \cap B'' \subseteq ((A \cap B) \cup (A' \cap B')) - J_\beta.$$

根据 (33) 和 $|A'' \cap W| \geqslant \theta$，我们有 $|B'' \cap W| \geqslant \theta$，因为 $|B \cap W| \geqslant \theta$ 以及 $B \subseteq B''$。

Set

$$\widetilde{C}'' := (A'' \cap B'') \cap (\widetilde{C} \cup \widetilde{C}'),$$
$$C'' := (A'' \cap B'') \cap (C \cup \widetilde{C}' \cup C_i), \text{ and}$$
$$D'' := (A'' \cap B'') \cap (D \cup D').$$

现在需要注意的是，

$$(A'' \cap B'') - (C'' \cup D'') = Y = V(F) - J_\beta. \tag{4}$$

~ ~ ~现在我们来分析元组 $((A'', B''), (C'', D'', C''))$。由 $C \subseteq C$，可得 $C'' \subseteq C''$。

根据 (11)，并且由于 $|C_i| = \mathcal{O}_{d,\ h}(k + k_0/\sqrt{k})$，我们有

$$|C''| \leqslant |C| + \mathcal{O}_{d,h}(k + k_0/\sqrt{k}) \tag{5}$$

and

$$|\widetilde{C}''| \leqslant |\widetilde{C}| + \mathcal{O}_{d,h}(\sqrt{k} + \log^{[d\geqslant 1]}(k + k_0)). \tag{6}$$

~由于 $C'' - (C \cup C'') \subseteq C_i$，对于每个 $v \in V(G)$，都有

$$\left|N_G^d[v] \cap (C'' - (C \cup \widetilde{C}''))\right| \leqslant \left|N_G^d[v] \cap C_i\right| = \mathcal{O}_{h,d}(\log(k + k_0)). \tag{7}$$

我们有 $D'' \cap N_G^d[Z_0] = \emptyset$，并且如果每次猜测都正确，我们就有 $D'' \cap \bigcup \mathcal{Z} = \emptyset$，并且

$$|(C'' - C) \cap \bigcup \mathcal{Z}| = \mathcal{O}_{d,h}\left((\sqrt{k} + k_0/\sqrt{k}) \log^{[d\geqslant 1]}(k + k_0)\right) \tag{8}$$

汇总本次递归调用中迄今为止的所有猜测，根据 (22)，可知猜测正确的概率至少为

$$(\mathcal{O}_{d,h}(k + k_0))^{-\binom{h}{2}+1} \cdot \left(\mathcal{O}_{d,h}\left(\sqrt{k} + \log^{[d\geqslant 1]}(k + k_0)\right)\right)^{-1} \cdot \frac{1}{2} \geqslant (\mathcal{O}_{d,h}(k + k_0))^{-\binom{h}{2}}. \tag{9}$$

使用分离 (A'', B'')：递归情形。 令 (F^*, ζ^*) 为移除连通分量 J_β 后得到的带标注森林 (F, ζ)。注意，F^* 是 $(A'' \cap B'') - (C'' \cup D'')$ 的生成森林，且是 F^1 的子图。

我们首先考虑这样一种情况：F^* 的某个连通分量 J^* 是 $\mathcal{Z}$-重的，即其权重大于 $(\delta + 1)\sqrt{k}$。我们对 (F^*, ζ^*) 执行分裂操作，得到 (F'', ζ'')，并且 ~对元组 $((A'', B''), (C'', D'', C''), (F'', \zeta''))$ 进行递归。如果迄今为止的所有猜测都正确，则 (F^*, ζ^*) 与 $\mathcal{Z}$ 一致；而分裂操作以至少 $((ck + 1)(\ell - 1))^{-1}$ 的概率输出与 $\mathcal{Z}$ 一致的 (F'', ζ'')。此外，F'' 恰有 h 个连通分量，并且 F'' 的两个新连通分量是 $\mathcal{Z}$-显著的，因为 (F^*, ζ^*) 是 $\mathcal{Z}$-重的。

让我们检查子调用的假设。关于 $|C|$ 的界由式（55）得出。关于 $|N_G^d[v] \cap (C - \widetilde{C})|$ 的界由式（77）得出。如果所有猜测都正确，那么根据式（88），子调用是一致的。

最后，根据式（99）以及上一段中的分析，对这次递归调用中的所有猜测进行汇总后，子递归调用再次以至少

$$(\mathcal{O}_{d,h}(k + k_0))^{-\binom{h}{2}} \cdot \frac{1}{(ck + 1)(\ell - 1)} \geqslant (\mathcal{O}_{d,h}(k + k_0))^{-h^2} \cdot \ell^{-1}.$$

利用分离 (A'', B'')：叶情形。 剩下的情况是 (F^*, ζ^*) 不是 $\mathcal{Z}$-heavy，也就是说，F^* 的每个连通分量 J^* 都满足 $\zeta^*(J^*) \leqslant (\delta + 1)\sqrt{k}$。这是递归的一片叶节点：我们输出 $(A'', B'', C'' \cup Y, C'')$ 并终止。（回想一下 $Y = V(F) - J_{\beta^\circ}$）

我们来分析这个输出。我们声称它满足要求。回想一下，递归深度被人为截断为 $\xi = 3h(\delta + 1)\log k = \mathcal{O}_h(\log k)$。因此，由 (55) 可知

$$|C'' \cup Y| \leqslant \ell + \mathcal{O}_{d,h}((k + k_0/\sqrt{k}) \log k).$$

同样，由（66）可得

$$|\widetilde{C}''| \leqslant \mathcal{O}_{d,h}\left(\left(\sqrt{k} + \log^{[d\geqslant 1]}(k + k_0)\right) \log k\right).$$

由（77）可知，对于每个 $v \in V$（G），都有

$$\left|N_G^d[v] \cap (C'' - \widetilde{C}'')\right| = \mathcal{O}_{h,d}(\log(k+k_0) \cdot \log k).$$

回顾一下，此处 $|N_G^d[v] \cap Y| \leqslant c$ as $Y \subseteq V(F^\circ)$。

由于 (F, ζ) 至少有两个连通分量是 $\mathcal{Z}$-显著的，因此我们有 $\zeta(Y) > \sqrt{k}$。由于对于每个 $K \in \mathcal{Z}$，$Y \subseteq V(F^\circ)$ 和 $|K \cap V(F^\circ)| \leqslant c$，我们得到

$$|\{K \in \mathcal{Z} \mid K \cap Y \neq \emptyset\}| \geqslant \sqrt{k}/c.$$

此外，由于 (F^*, ζ^*) 不是 $\mathcal{Z}$-重的，如果 (F^*, ζ^*) 与 $\mathcal{Z}$ 一致，那么

$$\left|Y \cap \bigcup \mathcal{Z}\right| \leqslant h(\delta+1)\sqrt{k} = \mathcal{O}_h(\sqrt{k}).$$

如果到目前为止的每个猜测都正确，则由 (88) 可知 $A'' \cap B'' \cap \bigcup \mathcal{Z} \subseteq C'' \cup Y$ 且

$$\left|C'' \cap \bigcup \mathcal{Z}\right| \leqslant \mathcal{O}_{d,h}\left((\sqrt{k}+k_0/\sqrt{k}) \log^{[d \geqslant 1]}(k+k_0) \log k\right).$$

至此，证明了输出满足保证的性质；并且，如果调用是一致的且所有猜测均正确，那么它还满足关于 $\mathcal{Z}$ 的保证的性质。

我们还注意到，得益于 (99)，直到深度 $\xi = \mathcal{O}_h(\log k)$ 所有猜测均正确的概率至少为

$$(\mathcal{O}_{d,h}((k+k_0)\ell))^{-\mathcal{O}_h(\log k)}.$$

递归深度的界。接下来需要说明，如果所有猜测都正确，那么递归最迟会在深度 $\xi = 3h(\delta+1)\log k$ 时到达叶节点。也就是说，超过这一深度的所有递归调用，其历史中都曾有过错误猜测，因此可以终止。

我们先给出以下观察。

断言 5.6。在一次递归调用 $((A,B),(C,D,C),(F,\zeta))$ 中，F 的每个连通分量 J 要么是 F^1 的一个连通分量，要么其权重至少占 F 总权重的 $h^{-1}(\delta+1)$。

断言的证明。如果 J 不是递归树根部带标注森林中的一个连通分量，那么它就是在所考察调用的某个祖先处，对带标注森林 (F^*,ζ^*) 中的一个连通分量 J^* 进行分裂后产生的。设 (F,ζ) 是该祖先调用中的带标注森林；回忆一下，F^* 等于删去一个连通分量后的 F。由于在分裂操作中，我们取 ζ^* 值最大的连通分量，而 F 的所有连通分量都是 F^* 中连通分量的子集，$\zeta^*(J^*) \geqslant \zeta(J'')$ 对 F 的每个连通分量 J'' 都成立。因此，对 F 的每个连通分量 J''，$\zeta(J) \geqslant \frac{1}{\delta+1}\zeta(J'')$。由于 F 有 h 个连通分量，J 的权重至少占 F 权重的 $h^{-1}(\delta+1)$。」

考虑一个递归调用 $((A,B),(C,D,C),(F,\zeta))$ 及其子调用 $((A'',B''),(C'',D'',C''),(F'',\zeta''))$.$F''$ 由 F 构造而成：首先丢弃一个连通分量（在上文的描述中记为 J_β），然后对剩余分量中 ζ 值最大的一个执行分裂操作。如果被丢弃分量的权重至少为 F 权重的 $h^{-1}(\delta+1)$，则称该子调用为重子调用；否则称为轻子调用。引理 5 65..6 保证，当所有猜测均正确且所有调用均一致时，轻调用至多有 h 次，重调用至多有 $\log k / \log\left(1 - h^{-1}(\delta+1)\right)$ 次。因此，这类调用的深度有界于

$$1 + h + \left\lceil \left(\log\left(1 - \frac{1}{h(\delta+1)}\right)\right)^{-1} \log k \right\rceil \leqslant 3h(\delta+1)\log k.$$

至此，完成了引理 51 的证明。。。 □

6 算法

现在我们可以证明引理 1 21。.2 和 1 313。。。本节的主要部分用于证明以下情形下的陈述： G是无顶点的近似可嵌入图，且其直径相对于 k呈线性有界；这部分内容见第 6 1 节61。。。将这一情形推广到引理 1 21。.2 的完整陈述将在第 6 2 节6。.2 中完成；推广到引理 1 31。.3 的完整陈述则见第 6 3 节63。。。

6.1 无顶点的近可嵌入图

本节重点证明以下命题，该命题讨论图在无顶点的情况下近似可嵌入，且我们可以在界中使用直径。

我们指出，对于定理 1 21..2，下述定理只需在 $d = 0$ 的情况下使用；不关心定理 1 31..3 的读者只需关注这一情况。此外，此时可以放心忽略集合 Y_0，因为对于 $d = 0$，关于集合 Y_0 的性质很容易通过后处理满足：只需将 Y_0 加入 $V(G')$，并加入所构造树分解中的每个袋即可。我们还回顾一下记号 $[d \geqslant 1]$，它在 1 时等于 $d \geqslant 1$，在 0 时等于 $d = 0$。

定理 6.1。对于任意非负整数 p, d，存在一个常数 $c_{d,p} > 0$ 以及一个随机化多项式时间算法，该算法给定一个整数 $k \geqslant 2$、一个图 G，它是

1. 直径为 $\Delta \geqslant 1$，*2.* 不含顶点的 p-近似可嵌入，且 *3.* 此外，若 $d \geqslant 1$, $K_{3,\ p}$-子式自由，

以及一个大小至多为 p 的集合 $Y_0 \subseteq V$（G），则输出 G 的一个诱导子图 G'，以及 G' 的一个树分解 （T, β），其最大袋大小有上界

$$c_{d,p}(k \log k + \Delta)$$

使得 $N^d[Y_0] \subseteq V$（G'），T 的 β 中的每个袋都至多包含

$$c_{d,p}\left(\sqrt{k} + \log(k + \Delta)\right) \log(k + \Delta)$$

$N^d[Y_0]$ 的顶点，并且对于任意一个由两两不交的 d-簇组成的族 $\mathcal{Z}$，若 $\sqrt{k} \leqslant |\mathcal{Z}| \leqslant k$，则其概率至少为

$$\left((k\Delta)^{\frac{c_{d,p}|\mathcal{Z}| \log k}{\sqrt{k}}} |V(G)|^{\frac{c_{d,p}|\mathcal{Z}|}{k}}\right)^{-1}$$

我们有 $\bigcup \mathcal{Z} \subseteq V$（$G'$），且 T 的 β 中的每个袋都至多包含

$$d_{d,p}\left(\sqrt{k} \log k + \frac{\Delta}{\sqrt{k}}\right) \log^{1+3[d \geqslant 1]} k \cdot \log^{[d \geqslant 1]}(k + \Delta).$$

$\bigcup \mathcal{Z}$ 的顶点。

证明。设输入中给定了 G、Y_0 和 k，并设 G 的直径为 Δ。为便于分析，固定按陈述所述取 $\mathcal{Z}$。

我们假设$k > \kappa_{d,\ p}$ 对于某个常数 $\kappa_{d,\ p}$ 取决于 d 和 p，其值稍后确定。事实上，如果 $k \leqslant \kappa_{d,\ p}$，则只需返回 $G' = G$ 以及引理 37 中 G 的一个树分解37。。。

递归过程的定义。我们给出一个递归过程，其~规范如下。每次递归调用 $\mathfrak{C}$ 都会获得集合 M，R，R，R_{old} 满足 R，$R_{\text{old}} \subseteq R \subseteq M \subseteq V(G)$ 和 $Y_0 \cap M \subseteq R$，其中

$$|R| = \mathcal{O}_{d,p}\left(k \log k + \Delta\right),$$
$$|\widetilde{R}| = \mathcal{O}_{d,p}\left(\left(\sqrt{k} + \log^{[d\geqslant 1]}(k+\Delta)\right)\log k\right).$$

我们将确保以下性质：对于每个 $v \in V(G)$，都有

$$\left|N_G^d[v] \cap (R - \widetilde{R})\right| = \mathcal{O}_{d,p}(\log^2 k \log(k+\Delta)). \tag{10}$$

~我们指出，对于 $d = 0$关于 R 的性质（包括上述性质），显然都将~在 $R = \emptyset$ 下满足；集合 R 仅在 $d \geqslant 1$ 的情形中起重要作用。此外，集合R_{old} 仅用作维持 (10) 的工具，因此只在 $d \geqslant 1$ 中起重要作用。

一次递归调用返回一个子集 $M \subseteq \overline{M}$，以及 $G[M]$ 的一个有根树分解 (T, β)，并具有以下性质：

1. $M \cap N_G^d[R] \subseteq \overline{M} \subseteq M$；（注意，特别是，第一个包含关系意味着 $R \subseteq M$，而对于$\overline{d} = 0$，它实际上等价于 $R \subseteq M$。）
2. R 包含在 (T, β) 的根袋中；3. (T, β) 的每个袋的大小均为

$$\mathcal{O}_{d,p}(k \log k + \Delta);$$

4. 对于每个 $t \in V(T)$，都存在一个大小为 $\mathcal{O}_{d,p}((\sqrt{k} + \log^{[d\geqslant 1]}(k+\Delta))\,\overline{\log}\ k)$ 的集合 $\tilde{\beta}(t) \subseteq \beta(t)$，使得对于每个 $v \in V(G)$，

$$\left|N_G^d[v] \cap (\beta(t) - \tilde{\beta}(t))\right| = \mathcal{O}_{d,p}(\log^2 k \log(k+\Delta)).$$

~ ~(与 R 类似，对于 $d = 0$ with $\beta(t) = \emptyset$，这一点显然成立。)

~若递归调用的输入为 $\overline{M}$, R, R, R_{old}，则称其输出 $(M, (T, \beta))$ 与 $\mathcal{Z}$ 一致*consistent*，如果满足以下条件：

1. $M \cap \bigcup \mathcal{Z} \subseteq \overline{M}$, and
2. (T, β) 的每个袋都包含

$$\mathcal{O}_{d,p}\left(\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log^{1+3[d\geqslant 1]} k \cdot \log^{[d\geqslant 1]}(k+\Delta)\right).$$

$M \cap \bigcup \mathcal{Z}$ 的顶点。

~讨论单次递归调用时，我们将使用简洁的记号 M, R, R, R_{old}。在考虑多次递归调用的推理中，我们将使用记号 $M(\mathfrak{C})$、$R(\mathfrak{C})$ 等，表示调用 $\mathfrak{C}$ 中对应的集合。

~初始调用和最终输出。 </style}初始调用针对 ，且 ()。输出 and (, β) 成为整个算法的输出，其中包含 $G' = G[M]$ 及其树分解 (T, β)。不难验证，输出 $(M, (T, \beta))$ 的保证性质蕴含算法最终输出的保证性质。特别是，关于 $N_G^d[Y_0]$ 的保证由关于 $\tilde{\beta}(t)$ 的保证性质确保：对于每个 $t \in V(T)$，我们有

$$\begin{aligned}\left|N_G^d[Y_0] \cap \beta(t)\right| &\leqslant |\tilde{\beta}(t)| + \bigcup_{v \in Y_0} \left|N_G^d[v] \cap (\beta(t) - \tilde{\beta}(t))\right| \\ &= \mathcal{O}_{d,p}\left(\left(\sqrt{k} + \log^{[d \geqslant 1]}(k+\Delta)\right)\log k\right) + p \cdot \mathcal{O}_{d,p}(\log^2 k \log(k+\Delta)) \\ &= \mathcal{O}_{d,p}\left(\left(\sqrt{k} + \log^{[d \geqslant 1]}(k+\Delta)\right)\log(k+\Delta)\right).\end{aligned}$$

此外，注意，如果输出 $(M, (T, \beta))$ 与 $\mathcal{Z}$ 一致，那么算法的最终输出就满足关于集合 $\mathcal{Z}$ 的保证性质。我们将确保递归的根调用所产生的输出以足够高的概率与 $\mathcal{Z}$ 一致。

递归叶节点。在递归的叶节点处，如果 $|M - R| \leqslant 4\sqrt{k}$，我们只需返回 $M = M$ 以及一个只有单个袋 M 的平凡树分解。保证的性质显然成立。（对于此调用中创建的单个袋，我们可以使用 $\beta(t) = R \cup (M - R)$。）

递归调用的记号。在描述非叶递归调用之前，我们先引入一些记号。

对于调用 $\mathfrak{C}$ 和固定的 $\mathcal{Z}$，我们记作

$$\begin{aligned}\mathcal{Z}_{\text{border}}(\mathfrak{C}) &= \{K \in \mathcal{Z} \mid K \cap R(\mathfrak{C}) \neq \emptyset\}, \\ \mathcal{Z}_{\text{inside}}(\mathfrak{C}) &= \{K \in \mathcal{Z} \mid K \subseteq M(\mathfrak{C}) - R(\mathfrak{C})\}, \\ \mathcal{Z}_{\text{all}}(\mathfrak{C}) &= \{K \in \mathcal{Z} \mid K \cap M(\mathfrak{C}) \neq \emptyset\}.\end{aligned}$$

注意， $\mathcal{Z}_{\text{all}}(\mathfrak{C}) = \mathcal{Z}_{\text{border}}(\mathfrak{C}) \uplus \mathcal{Z}_{\text{inside}}(\mathfrak{C})$ 是一个划分。如果从上下文中可以清楚地看出调用对象，我们将省略参数 $\mathfrak{C}$。

设 a_F 和 δ_F 分别为引理 3 73..7 给出的森林 F 的连通分量数上界和最大度上界，并设 c_F 满足 $|A \cap B| \leqslant c_F(\Delta + 1)$，且对于引理 3 73 7.. . 给出的分离 (A, B)，每个顶点 $v \in V(G)$ 对每个 $d' \geqslant 0$ 均满足 $|N_G^{d'}[v] \cap A \cap B| \leqslant c_F(2d' + 1)$。注意， a_F、 c_F 和 δ_F 仅取决于 p，且都是常数。

将引理 3 23。.2 应用于以下图类： p-近似可嵌入且不含顶点的图，从而得到一个图 H ，使得每个 p-近似可嵌入且不含顶点的图都是 H-子式自由的。令 $h := \max(p+4,\ a_F + 1,\ \delta_F,\ |V(H)|)$。注意， h 是一个仅取决于 p 的常数。

非叶递归调用。递归以轮流方式交替执行两种类型的递归步骤：常规模式或模式模式。此外，常规模式分为四种，具体取决于变量 W 的初始化方式。递归调用的模式取决于其深度模 5 的余数：如果深度模 5 的余数为 0、1、2 或 3，则进入常规模式的四种模式之一；如果深度模 5 的余数为 4，则进入模式模式。根调用的深度为 0。

在某些情况下，我们什么也不做，只将相同的参数传递给一个深度大一的递归调用（因此它具有不同的模式）。在所有其他情况下，我们将确定一个集合 $W \subseteq M$、集合 $\theta := \lfloor |W|/4 \rfloor$，并找到 G 的一个分离 (A°, B°)，该分离相对于 W 是平衡的，具体如下： $|A^\circ \cap W| \geqslant \theta$ 和 $|B^\circ \cap W| \geqslant \theta$。我们将把 (A°, B°) 细化为一个分离

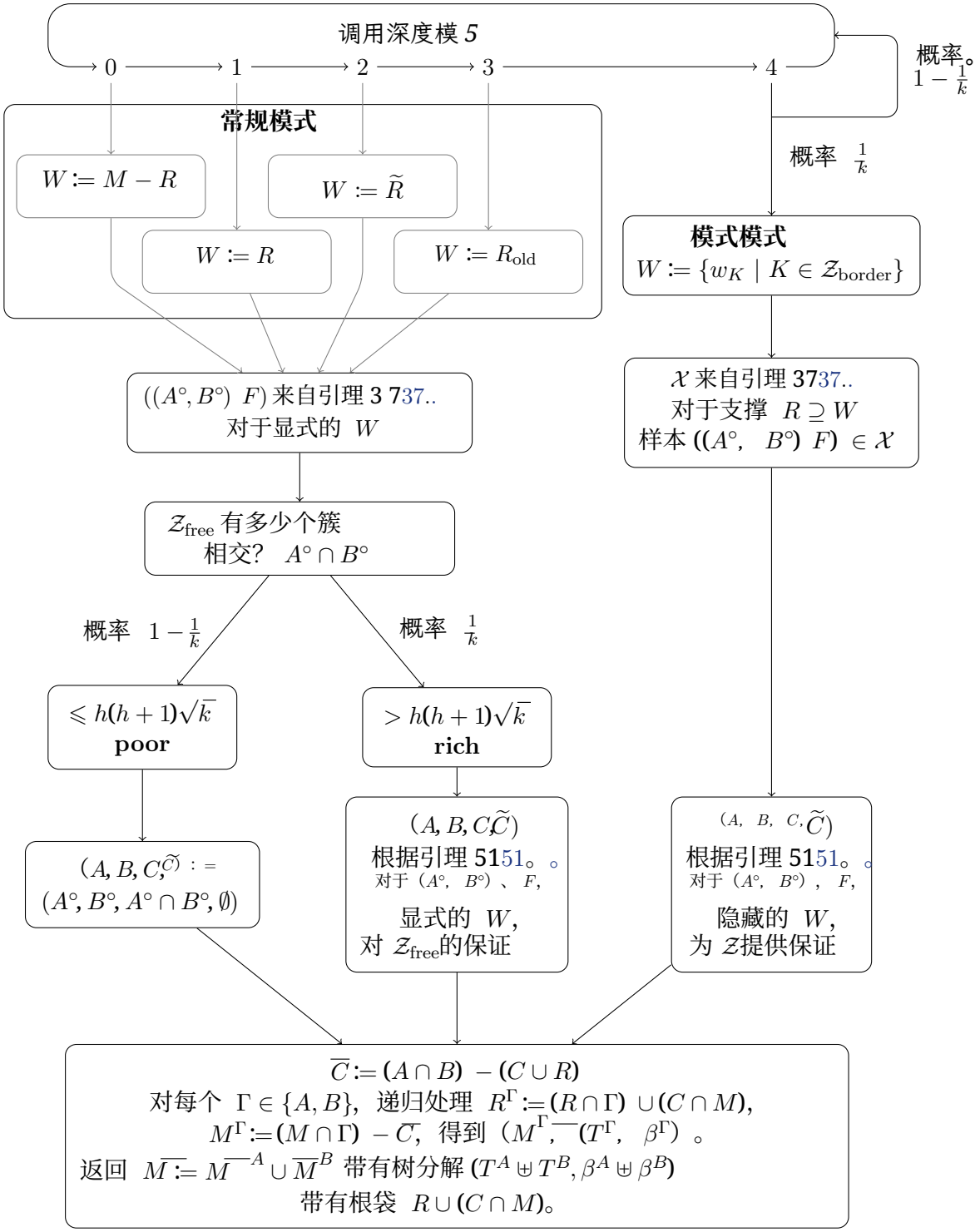

图 5：非叶递归调用的控制流示意图。

(A, B) （通常使用定理 5 151。。）以控制 $\mathcal{Z}$ 与 $A \cap B$的交集。基于 (A, B)，我们将调用两个递归子调用，并根据递归子调用的输出计算输出 $(M, (T, \beta))$。

现在我们将精确描述各种模式。示意图见图 55 。

常规模式。 首先，根据深度除以 5 的余数定义 $W \subseteq M$： 对于~余数 0、1、2、3，我们分别设定 $W := M - R$、 $W := R$、 $W := R$ 和 $W := R_{\mathrm{old}}$。此外，在最后一种情况（即余数为 3）下，如果 $|R_{\mathrm{old}}| \leqslant 3$，那么我们会先重置 $R_{\mathrm{old}} := R$，再设定 $W := R_{\mathrm{old}}$。

给定 W，使用显式权重函数的用法应用引理 3 73。.7 于 G，并在 W上使用均匀测度，得到 $((A^\circ, B^\circ)\ F)$。（重要的是，我们在整个图 G中进行这一步；不能仅对 3。.7 应用引理 3 7 $G[M]$，因为它的直径可能非常大。）令 $\theta = \lfloor |W|/4 \rfloor \geqslant 1$。因此我们有 $|W| \geqslant 4\theta$, $|A^\circ \cap W| \geqslant \theta$，以及 $|B^\circ \cap W| \geqslant \theta$。
我们称 $A^\circ \cap B^\circ$ 为丰富，如果 $A^\circ \cap B^\circ$ 与 $\mathcal{Z}_{\mathrm{inside}}$ 的集合相交超过 $h(h \bar{+} 1)\sqrt{k}$ 个；否则称其为贫。我们以有偏概率随机猜测 $A^\circ \cap B^\circ$ 是丰富还是贫。更确切地说，我们以概率 $\frac{1}{k}$ 猜测 $A^\circ \cap B^\circ$ 是丰富，并以剩余概率 $1 - \frac{1}{k}$ 假设 $A^\circ \cap B^\circ$ 是贫。

- 如果我们猜测“丰富”，那么就针对常数 $c := c_F(2d+1)$、 d和 h，将 G、 $Z_0 := R$、 k、 W、 θ5..1 调用引理 5.1 的随机化多项式时间版本 ~$(A^\circ$、 $B^\circ)$，以及森林 F。我们将得到的元组记为（A、 B、 C、 C）。~
- 如果我们猜测“贫”，那么就令（A、 B、 C、 C）：$=(A^\circ$、 B°、 $A^\circ \cap B^\circ$、 $\emptyset)$。

模式模式。 这种模式与前一种略有不同，因为这里的集合 W 是基于算法未知的 $\mathcal{Z}$定义的。

我们先抛一枚偏置硬币：以概率 $1 - \frac{1}{k}$ 什么也不做（即，将相同的输入传递给唯一一个子节点递归调用，并将深度增加一，因此它将进入常规模式），以概率 $\frac{1}{k}$ 则继续进行。

我们按以下方式定义一个集合 W （算法未知）。对于每个 $K \in \mathcal{Z}_{\mathrm{border}}$，选取一个 $w_K \in K \cap R$，并令 $W = \{w_K \mid K \in \mathcal{Z}_{\mathrm{border}}\}$ 和 $\theta = \lfloor |W|/4 \rfloor$。注意，尽管我们有 $W \subseteq R$，但算法并不知道 W 和 θ。我们将引理 3 73..7 与显式支撑用法应用于 G 和 $U := R$，得到一个族 $\mathcal{X}$，并从中均匀随机选取 $((A^\circ, B^\circ)\ F) \in \mathcal{X}$。以至少 $(2|R|)^{-1}$ 的概率，选出的 $((A^\circ, B^\circ)\ F)$ 满足 $|A^\circ \cap W| \geqslant \theta$，且 $|B^\circ \cap W| \geqslant \theta$。

接着，我们对常数 $c := c_F(2d+1)$、 d 和 h，调用引理 5 15..1 的随机化多项式时间版本，以处理 G, $Z_0 := R$, k、(A°, B°) 以及森林 F。所得元组 ~ 仍记作 (A, B, C, C)。

Let

$$\overline{C} := (A \cap B) - (C \cup R).$$

对于每个 $\Gamma \in \{A, B\}$，递归处理

$$\begin{aligned} R^\Gamma &:= (R \cap \Gamma) \cup (C \cap M), \\ \widetilde{R}^\Gamma &:= (\widetilde{R} \cap (\Gamma - C)) \cup (\widetilde{C} \cap M), \\ R^\Gamma_{\mathrm{old}} &:= R_{\mathrm{old}} \cap (\Gamma - C), \\ M^\Gamma &:= (M \cap \Gamma) - \overline{C}, \end{aligned}$$

获取 M^Γ 和 $(T^\Gamma,\ \beta^\Gamma)$。（另见图 6。）将子调用记为 $\mathfrak{C}^\Gamma$ 对 $\Gamma \in \{A, B\}$。我们设 $M := M^A \cup M^B$ 并通过对 $(T,\ \beta)$ 和 $(T^A,\ \beta^A)$ 的不交并集构造树分解 $(T,\ \beta)$，并创建一个根袋

$$R \cup (C \cap M)$$

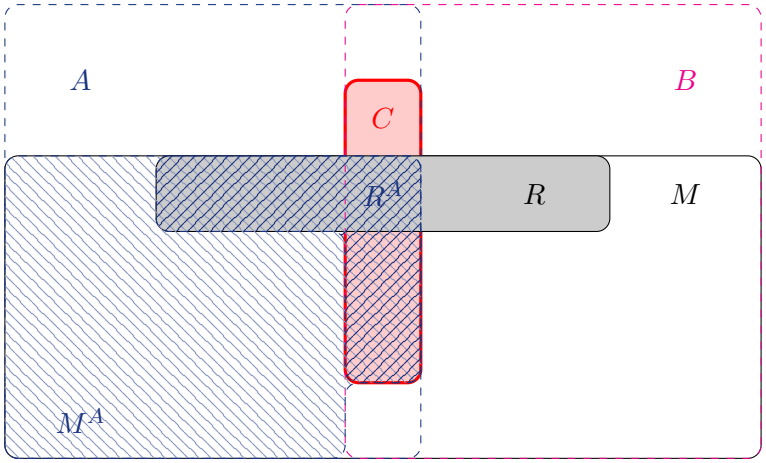

图 6：子节点递归调用中使用的集合

.

并将 $(T^\Gamma,\ \beta^\Gamma)$ 的根作为 $\Gamma \in \{A, B\}$ 的子节点连接起来。显然可以验证，$(T,\ \beta)$ 确实是 $G[M]$的树分解。（注意，C与 $M^A \cup M^B$不相交，因此 $C \cap M = \emptyset$。）至此，递归算法的描述完成。

分析：基本观察。现在我们继续进行分析。首先，根据 R^Γ、$\widetilde{R}^\Gamma$、R^Γ_{old} 和 M^Γ 的定义，得出一些直接的观察。固定 $\Gamma \in \{A, B\}$。首先，注意到 $R^\Gamma_{\text{old}} \subseteq R_{\text{old}}$。其次，观察到 $R^\Gamma - R \subseteq C$ 和 $\widetilde{R}^\Gamma - \widetilde{R} \subseteq \widetilde{C}$，因此特别有 $|R^\Gamma - R| \leqslant |C|$ 和 $|\widetilde{R}^\Gamma - \widetilde{R}| \leqslant |\widetilde{C}|$。第三，注意到 $M^\Gamma - R^\Gamma \subseteq M - R$ 和 $(M^A - R^A) \cap (M^B - R^B) = \emptyset$。这意味着 $\mathcal{Z}_{\text{inside}}(\mathfrak{C}^\Gamma) \subseteq \mathcal{Z}_{\text{inside}}(\mathfrak{C})$。

Let

$$\mathcal{Z}_{\text{pinned}}(\mathfrak{C}) \coloneqq \{K \in \mathcal{Z}_{\text{inside}}(\mathfrak{C}) \mid K \cap A \cap B \neq \emptyset\}.$$

我们可以立即得到如下观察。

断言 6.2。

$$\mathcal{Z}_{\text{inside}}(\mathfrak{C}) = \mathcal{Z}_{\text{inside}}(\mathfrak{C}^A) \uplus \mathcal{Z}_{\text{inside}}(\mathfrak{C}^B) \uplus \mathcal{Z}_{\text{pinned}}(\mathfrak{C})$$

是一个划分。

We now analyse the output $(A, B, C, \widetilde{C})$ regardless of the mode.

断言 6.3。以下不等式成立：

$$|C| \leqslant c_F(\Delta + 1) + \mathcal{O}_{d,p}(k \log k) + \mathcal{O}_{d,p}\left(\frac{|R| \log k}{\sqrt{k}}\right), \tag{11}$$

$$|\widetilde{C}| \leqslant \mathcal{O}_{d,p}\left(\left(\sqrt{k} + \log^{[d\geqslant 1]}(k + \Delta)\right) \log k\right), \tag{12}$$

$$C \supseteq N^d_G[R] \cap A \cap B, \tag{13}$$

$$\forall_{v \in V(G)} \quad \left|N^d_G[v] \cap (C - \widetilde{C})\right| = \mathcal{O}_{d,p}(\log^2 k). \tag{14}$$

~断言的证明。如果元组（A，B，C，C）来自定理 5 15..1 的一次应用，则该~断言可直接由定理 5 15 1.. 的保证得出。否则，我们有（A，B，C，C）=（A°，B°，$A^\circ \cap B^\circ$，$\emptyset$）；随后，（1111）和（1414）由引理 3 73..7 的保证得出，而（1212）和（1313）则显然成立。」

我们还注意到，在常规模式下，我们总有 $|A \cap W| \geqslant \theta$ 且 $|B \cap W| \geqslant \theta$，正如引理 5 15..1 对显式的 W 和 θ所保证的。在模式模式下，由于 W 和 θ 对算法而言是未知的，因此若引理 5 15..1 关于 W 和 θ的应用成功，则这些不等式成立。

^若调用引理 515。.1 ，则我们将其保证应用于集合 $\mathcal{Z}$ ，该集合定义如下。

$$\widehat{\mathcal{Z}} := \begin{cases} \mathcal{Z}_{\text{inside}} & \text{in the regular mode,} \\ \mathcal{Z} & \text{in the pattern mode.} \end{cases}$$

也就是说，以至少如下概率

$$(\mathcal{O}_{d,p}(k\Delta))^{-\mathcal{O}_p(\log k)},$$

我们有

$$A \cap B \cap \bigcup \widehat{\mathcal{Z}} \subseteq C \tag{15}$$

and

$$|C \cap \bigcup \widehat{\mathcal{Z}}| = \mathcal{O}_{d,p}\left(\left(\sqrt{k} + \frac{|R|}{\sqrt{k}}\right) \log^{1+[d\geqslant 1]} k\right). \tag{16}$$

此外，在相同的概率界内，

- 在模式模式下，我们还有 $|A \cap W| \geqslant \theta$ 以及 $|B \cap W| \geqslant \theta$(假设 $|A^\circ \cap W| \geqslant \theta$且 $|B^\circ \cap W| \geqslant \theta$，这件事发生的概率至少为 $(2|R|)^{-1}$，且独立于引理 5 1 的应用51。。);

- 在常规模式下，若 X 确实丰富，则此外我们有

$$|\mathcal{Z}_{\text{pinned}}| \geqslant \begin{cases} \frac{\sqrt{k}}{c_F(2d+1)} & \text{if } d \geqslant 1, \\ \sqrt{k} & \text{if } d = 0. \end{cases} \tag{17}$$

最后，请注意，在常规模式下，如果 $\mathcal{Z}_{\text{inside}} = \emptyset$ ，则 (1515) 和 (1616) 显然成立；如果处于贫乏情形且我们正确地猜中了它们，结论同样成立。

分析：控制集合的大小和递归深度。 现在，我们分析常规模式如何控制各种集合的大小，以及它如何界定递归深度。

断言 6.4。 在常规模式下，对于每个 $\Gamma \in \{A, B\}$，均有如下结论（取决于递归深度的模 *5* 的余数）。

余数 0： $|M^\Gamma - R^\Gamma| \leqslant \frac{6}{7}|M - R|$；**余数 1：** $|R^\Gamma - C| \leqslant \max(3, \frac{6}{7}|R|)$；**余数 2：** $|\widetilde{R}^\Gamma - C| \leqslant \max(3, \frac{6}{7}|\widetilde{R}|)$。**余数 3：** $|R^\Gamma_{\text{old}}| \leqslant \max(3, \frac{6}{7}|R_{\text{old}}|)$。

因此，递归深度有如下界

$$5 \log_{7/6} n \leqslant 23 \log n.$$

断言的证明。 在“余数 0”的情形下，由于我们不处于叶情形，$|M-R|>4\sqrt{k}\geqslant 4$。~在“余数 1–3”的情形下，子情形 $|R|\leqslant 3$、$|R|\leqslant 3$ 和 $|R_{\mathrm{old}}|\leqslant 3$ 分别可直接得到，因为 $R^{\Gamma}-C\subseteq R$、$\widetilde{R}^{\Gamma}-C\subseteq\widetilde{R}$ 和 $R^{\Gamma}_{\mathrm{old}}\subseteq R_{\mathrm{old}}$，因此我们可以假设所有情形下都有$|W|\geqslant 4$。于是，$\theta=\lfloor|W|/4\rfloor\geqslant|W|/7$。回想一下 $|A\cap W|$，$|B\cap W|\geqslant\theta$。这意味着，对于 $\Gamma\in\{A,B\}$，在 $M^{\Gamma}-R^{\Gamma}$、$R^{\Gamma}-C$、$\widetilde{R}^{\Gamma}-C$或 $R^{\Gamma}_{\mathrm{old}}$之外，分别至少有 θ 个 W的顶点。断言的第一部分得证。

第二部分的结论可由第一部分中“余数 0”的情形直接推出；此外，每第五次调用（从根节点开始计数，因为根的深度为 0）都会处于常规模式，并适用“余数 0”的情形。」

命题 646..4 的第二部分特别意味着，整个算法都能在随机多项式时间内运行。

令 $c_{d,p}$ 为式（1111）中 $\mathcal{O}_{d,p}$ 记号所隐含的两个常数中较大的一个。我们假设$\kappa_{d,p}$ 足够大，使得 $k>\kappa_{d,p}$ 蕴含

$$c_{d,p}\cdot\frac{\log k}{\sqrt{k}}<0.01$$

对于整数 $x\geqslant 0$，令

$$f(x)=\left\lfloor c_F(\Delta+1)+c_{d,p}k\log k+x\cdot\left(1+c_{d,p}\frac{\log k}{\sqrt{k}}\right)\right\rfloor.$$

S由于 $R^{\Gamma}\subseteq R\cup C$，根据式（1111）可得 $|R^{\Gamma}|\leqslant f(|R|)$。注意，$k>\kappa_{d,p}$ 蕴含

$$f(x)<c_F(\Delta+1)+c_{d,p}k\log k+1.01x.$$

这意味着 R 的大小满足以下界。

断言 6.5。 在每次调用 $\mathfrak{C}$中，

$$|R(\mathfrak{C})|\leqslant 100c_F(\Delta+1)+100c_{d,p}k\log k=:\Lambda_R. \tag{18}$$

断言的证明。 若 $\mathfrak{C}'$ 是 $\mathfrak{C}$ 的距离为 a的祖先，则 $|R(\mathfrak{C}')|\leqslant f^{(a)}(|R(\mathfrak{C}|))$，其中$f^{(a)}$是函数 a 连续应用 f次的复合。断言 6 46。.4对于余数 1 和 (1111) 意味着，若 $\mathfrak{C}$ 是 $\mathfrak{C}'$ 的子节点，且 $\mathfrak{C}'$ 处于常规模式，且 $W=R$ (即模 5 的余数为 1)，则

$$|R(\mathfrak{C})|\leqslant\max\left(3,\frac{6}{7}|R(\mathfrak{C}')|\right)+c_F(\Delta+1)+c_{d,p}k\log k+0.01|R(\mathfrak{C}')|.$$

通过对调用的深度作直接归纳，即可得出该断言。 」

利用 (1111)，可得

$$|C|\leqslant 2c_F(\Delta+1)+2c_{d,p}k\log k=:\Lambda_C. \tag{19}$$

由于每个新建的袋都是集合 R 与集合 C的一个子集的并集，因此 (1818) 和 (1919) 表明，所构造的树分解的袋大小至多为

$$\Lambda_R+\Lambda_C=102c_F(\Delta+1)+102c_{d,p}k\log k=\mathcal{O}_{d,p}(\Delta+k\log k).$$

由式（1212）和引理 646。.4 可推得

$$|\widetilde{R}|=\mathcal{O}_{d,p}\left(\left(\sqrt{k}+\log^{[d\geqslant 1]}(k+\Delta)\right)\log k\right). \tag{20}$$

我们还注意到，式（1818）使我们能够将式（1616）简化为

$$|C \cap \bigcup \widehat{\mathcal{Z}}| = \mathcal{O}_{d,p}\left(\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log^{1+[d\geqslant 1]} k\right). \tag{21}$$

现在我们使用 R_{old} 来论证性质（1010）。

断言 6.6。 在递归树中，每条长度至少为

$$5\log_{7/6}\Lambda_R = \mathcal{O}_{d,p}(\log(\Delta + k)),$$

该算法至少重置一次 $R_{\text{old}} := R$。

因此，对于每次调用，集合 R 都包含在由 $\mathcal{O}_{d,\ p}(\log(\Delta+k))$ 个最近的祖先调用中计算出的集合 C 的并集，以及至多 $\max(3$ 个其他顶点之中， $p)$。

断言的证明。 第一个断言可由界（1818）和引理 6464的“余数 3”情形直接得出。。。

对于第二个断言，考虑调用 $\mathfrak{C}_0$ ，并按如下方式定义关于 $i \geqslant 1$ 的调用 $\mathfrak{C}_i$ ： $\mathfrak{C}_i$ 是 $\mathfrak{C}_{i-1}$ 的最近的真祖先，其中发生了重置 $R_{\text{old}} := R$ ；若不存在这样的调用，则取根调用。由第一个断言可知，在 $\mathfrak{C}_{i-1}$ 和 $\mathfrak{C}_i$ 之间，对每个 $i \geqslant 1$，都有 $\mathcal{O}_{d,\ p}(\log(\Delta + k))$ 次调用。因此，对于 $i \geqslant 0$，集合 $R(\mathfrak{C}_i) - R(\mathfrak{C}_i)$ 包含在 $\mathfrak{C}_i$ 与 $\mathfrak{C}_{i+1}$ 之间各次调用中计算出的 $\mathcal{O}_{d,\ p}(\log(\Delta + k))$ 个集合的并集中。

我们关注调用 $\mathfrak{C}_0$ 和 $\mathfrak{C}_1$。我们有

$$\begin{aligned} R(\mathfrak{C}_0) &= (R(\mathfrak{C}_0) - \widetilde{R}(\mathfrak{C}_0)) \cup \widetilde{R}(\mathfrak{C}_0) \\ &\subseteq (R(\mathfrak{C}_0) - \widetilde{R}(\mathfrak{C}_0)) \cup R(\mathfrak{C}_1) \\ &= (R(\mathfrak{C}_0) - \widetilde{R}(\mathfrak{C}_0)) \cup (R(\mathfrak{C}_1) - \widetilde{R}(\mathfrak{C}_1)) \cup \widetilde{R}(\mathfrak{C}_1). \end{aligned}$$

上面并集中的前两个集合分别包含于调用 $\mathfrak{C}_2$ 和 $\mathfrak{C}_0$之间计算出的 $\mathcal{O}_{d,\ p}(\log(k + \Delta))$ 个集合 C的并集中，而最后一个集合的大小至多为 $\max(3,\ p)$。」

性质 (1010) 现在直接由引理 6666..6 和界 (1414) 得出。现在我们分析关于 $N_G^d[R]$ 的承诺性质。

命题 6.7。我们有 $M \cap N_G^d[R] \subseteq M$。

断言的证明。 我们通过自底向上归纳证明该断言，因此假设 $M^\Gamma \cap N_G^d[R^\Gamma] \subseteq M^\Gamma \subseteq M$对于 $\Gamma \in \{A, B\}$。在定理 5 15..1 (其应用发生在粘合模式或模式模式下) 中，已将 $Z_0 := R$ 作为参数传入。因此， $N_G^d[R] \cap \bar{C} = \emptyset$；注意，即使没有使用定理 5 15..1 （即处于常规模式、“贫”猜测）时，这一点也成立，因为此时$\bar{C} = \emptyset$。因此， $M \cap N_G^d[R] \cap A \cap B \subseteq M$ ，断言得证。 」

分析：成功运行。我们称调用 $\mathfrak{C}$为

主要的，当 $|\mathcal{Z}_{\text{inside}}(\mathfrak{C})| > \sqrt{k}$,

子式的，当 $1 \leqslant |\mathcal{Z}_{\text{inside}}(\mathfrak{C})| \leqslant \sqrt{k}$,

无效的，当 $\mathcal{Z}_{\text{inside}}(\mathfrak{C}) = \emptyset$。

我们称调用 $\mathfrak{C}$ 为局部成功 ， 当

- $\mathfrak{C}$ 是无效的，或者

- $\mathfrak{C}$ 是子式，并且：

  – $\mathfrak{C}$ 处于常规模式，且它猜测为“贫”；或者– $\mathfrak{C}$ 处于模式模式，并决定什么也不做，

  or

- $\mathfrak{C}$是主要的，并且：

  – $\mathfrak{C}$ 处于常规模式，它是贫的，而且我们正确地猜中了；或者 – $\mathfrak{C}$ 处于常规模式，它是丰富的，我们正确地猜中了，并且后续对定理 5 15..1 的$\widehat{}$调用返回了一个元组，该元组满足其相对于 $\mathcal{Z} = \mathcal{Z}_{\text{inside}}(\mathfrak{C})$ 的承诺；或者 – $\mathfrak{C}$处于模式模式，它正确地猜中了元组 $((A^\circ, B^\circ)\ F) \in \mathcal{X}$，该元组以平衡的方式分隔了 W，并且后续对定理 5 15..1 的调用返回了一个$\widehat{}$元组，该元组满足其相对于 $\mathcal{Z} = \mathcal{Z}$和 $W(\mathfrak{C})\ \theta(\mathfrak{C})$ 的承诺。

对于局部成功调用，我们可以如下界定 $\mathcal{Z}_{\text{pinned}}$ 的大小。

断言 6.8. 设 $\mathfrak{C}$是一个非叶局部成功调用。那么，我们要么什么也不做，要么

$$|\mathcal{Z}_{\text{pinned}}(\mathfrak{C})| = \mathcal{O}_{d,p}\left(\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log^{1+[d\geqslant 1]} k\right).$$

断言的证明。 如果该调用无效，那么 $\mathcal{Z}_{\text{inside}} = \emptyset$，因此 $\mathcal{Z}_{\text{pinned}} = \emptyset$，断言显然成立。如果该调用是次要调用，那么由于该调用局部成功，我们要么什么也不做，要么处于常规模式并猜测为“贫”。根据次要调用的定义，断言由 $|\mathcal{Z}_{\text{pinned}}| \leqslant |\mathcal{Z}_{\text{inside}}| \leqslant \sqrt{k}$ 得出。

如果该调用是主要调用，我们要么处于常规模式，要么处于模式模式。在常规模式下，我们正确猜测 $A^\circ \cap B^\circ$ 是丰富还是贫乏。在“丰富”的情形下，我们调用定理 5 151。。$\widehat{}$并期望在 $\mathcal{Z} = \mathcal{Z}_{\text{inside}}$ 方面成功，因此断言由 (2121) 得出。在“贫乏”的情形下，我们有 $(A,\ B) = (A^\circ,\ B^\circ)$，断言由贫乏情形的定义得出。最后，在模式模式下，我们调用定理 5 15。.1 并期望在$\sim\mathcal{Z} = \mathcal{Z}$ 方面成功，因此断言由 (2121) 得出。 」

令 Λ_{pinned} 为以下各项的最大值： $3 + p$、 $\sqrt{k}$、式 (2121) 的上界，以及命题 6 8 中 $\mathcal{Z}_{\text{pinned}}$ 的上界 68。。。

我们称运行对 $\mathcal{Z}$ 成功，如果每次调用都局部成功，并且此外，对于模式模式中的每个主要调用 $\mathfrak{C}$，当且仅当以下条件成立时，我们什么也不做

$$|\mathcal{Z}_{\text{border}}(\mathfrak{C})| \leqslant 30\Lambda_{\text{pinned}}.$$

这里我们强调以下重要技巧。在成功运行的定义中，我们不对无效调用提出任何要求。在这次调用中，我们有 $M \cap \bigcup \mathcal{Z} \subseteq N_G^d[R]$，而我们需要的一切都由关于 $N_G^d[R]$ 的性质保证（这些性质以概率 1 成立）。

命题 6.9。在一次成功运行中，若 G' 是根调用返回的诱导子图，则 $\bigcup \mathcal{Z} \subseteq V(G')$。

~断言的证明。我们通过自底向上归纳证明：对于输入为 M、 R、 R 和 R_{old} 的一次调用，若该调用返回 M 和 $(T,\ \beta)$，并且若 $\bigcup \mathcal{Z}$ 与 $\overline{N}$ 不相交（M） $- R$，则 $M \cap \bigcup \mathcal{Z} \subseteq M$。将此断言应用于根调用即可得到结论。——

对于 $M = M$ 的叶调用，该断言显然成立。—如果该调用无效，则 $M \cap \bigcup \mathcal{Z} \subseteq N_G^d[R]$ 且 $N_G^d[R] \cap M \subseteq M$由断言 6 767 得出。。

否则，该调用是子式或主要调用。特别是，定理 5 15..1所述潜在运行成功，意味着 $\bar{C}\cap M\cap\bigcup\mathcal{Z}=\emptyset$。因此，子调用满足 $\bigcup\mathcal{Z}\cap(N(M^{\Gamma})-R^{\Gamma})=\emptyset$。根据归纳假设，该断言成立，因为 $M^{A}\cup M^{B}=M$且 $M^{A}\cup M^{B}=M$。」

分析：与 $\mathcal{Z}$ 的交集上界。我们现在可以证明单个袋与 $\bigcup\mathcal{Z}$ 的交集具有所承诺的上界。

断言 6.10。假设该运行是成功运行。那么，对于每个调用 $\mathfrak{C}$，都有

$$|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|\leqslant 37\Lambda_{\mathrm{pinned}}.$$

此外，如果 $\mathfrak{C}$ 处于模式模式，且成功运行的定义要求执行一次操作（即，$\mathfrak{C}$ 是主要调用，且 $|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|>30\Lambda_{\mathrm{pinned}}$），那么设 $\mathfrak{C}_1$ 和 $\mathfrak{C}_2$ 为 $\mathfrak{C}$的两个子节点调用，对于 $i=1$，有2

$$|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C}_i)-\mathcal{Z}_{\mathrm{all}}(\mathfrak{C}_{3-i})|\geqslant 6\Lambda_{\mathrm{pinned}}.$$

断言的证明。首先注意，该断言在根调用处显然成立，因为在那里我们有$R=Y_0$ 和 $|Y_0|\leqslant p$，所以 $|\mathcal{Z}_{\mathrm{border}}|\leqslant p<\Lambda_{\mathrm{pinned}}$。此外，根调用处处于常规模式，因此第二个断言空泛地成立。

现在，我们按调用的深度对该断言进行归纳证明。我们将稍微加强归纳假设：对于每个调用 $\mathfrak{C}$，若它是根调用或某个主要调用的子节点，我们证明一个更强的上界，即 $|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|\leqslant(31+\rho)\Lambda_{\mathrm{pinned}}$，其中 ρ 是调用 $\mathfrak{C}$ 的深度除以 5 所得的余数。（稍后我们将证明其余调用 $\mathfrak{C}$也满足所承诺的结论。）注意，如果一个调用是主要调用，那么它的所有祖先也都是主要调用。我们已经验证了根调用处的断言。

于是考虑调用 $\mathfrak{C}$，其子调用为 $\mathfrak{C}_1$、$\mathfrak{C}_2$，且 $\mathfrak{C}$ 是主要调用或根调用。令 ρ 为 $\mathfrak{C}$ 的深度模 5 的余数。我们假设 $|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|\leqslant(31+\rho)\Lambda_{\mathrm{pinned}}$；目标是证明对 $\mathfrak{C}_i$、$i=1$、2 的类似界，以及在适用时证明该断言关于 $\mathfrak{C}$ 的第二部分。注意，如果 $\rho\neq 4$ 或 $|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|\leqslant 30\Lambda_{\mathrm{pinned}}$，则由引理 6 86..8 可立即得到结论；因为对于$i=1$、2，我们有 $\mathcal{Z}_{\mathrm{border}}(\mathfrak{C}_i)\subseteq\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})+\mathcal{Z}_{\mathrm{pinned}}(\mathfrak{C})$。

于是，假设 $\rho=4$（因此 $\mathfrak{C}$ 是主要调用，且处于模式模式下），并且 $|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|>30\Lambda_{\mathrm{pinned}}$。由归纳假设，$|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|\leqslant 35\Lambda_{\mathrm{pinned}}$。成功运行的定义要求在 $\mathfrak{C}$ 处执行一个操作。在调用 $\mathfrak{C}$ 中，我们定义 $W(\mathfrak{C})=\{w_K\mid K\in\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})\}$。由于$\mathcal{Z}$ 的簇两两不交，顶点 w_K 两两不同，且 $|W(\mathfrak{C})|=|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})|$。因此，$\theta(\mathfrak{C})\geqslant 7\Lambda_{\mathrm{pinned}}$。根据定理 5 15..1 的成功应用所作出的保证，

$$|A(\mathfrak{C})\cap W(\mathfrak{C})|,|B(\mathfrak{C})\cap W(\mathfrak{C})|\geqslant\theta(\mathfrak{C}).$$

注意，对于 $i=1$，2，每个 $K\in\mathcal{Z}_{\mathrm{border}}(\mathfrak{C}_i)$ 要么与 $A(\mathfrak{C})\cap B(\mathfrak{C})$ 相交，要么与 $K\notin\mathcal{Z}_{\mathrm{all}}(\mathfrak{C}_{3-i})$ 相交。因此，结合 (2121)，可得：

$$|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C}_i)-\mathcal{Z}_{\mathrm{all}}(\mathfrak{C}_{3-i})|\geqslant\theta(\mathfrak{C})-\Lambda_{\mathrm{pinned}}\geqslant 6\Lambda_{\mathrm{pinned}}.$$

因此，我们有

$$|\mathcal{Z}_{\mathrm{border}}(\mathfrak{C}_i)|\leqslant 35\Lambda_{\mathrm{border}}-6\Lambda_{\mathrm{pinned}}+\Lambda_{\mathrm{border}}=30\Lambda_{\mathrm{pinned}}.$$

这就完成了对父节点为主要节点的调用以及根调用的归纳证明。

现在考虑一个非根调用 $\mathfrak{C}$，且其父节点并非主要节点。令 $\mathfrak{C}'$ 为 $\mathfrak{C}$ 的最低祖先，该祖先是主要节点或为根调用。假设 $\mathfrak{C}'$ 产生了一个分离 (A',B')。令 $\mathfrak{C}''$ 为 $\mathfrak{C}'$ 的一个子调用，它是 $\mathfrak{C}$ 的祖先（也可能是 $\mathfrak{C}''=\mathfrak{C}$）。那么，$\mathcal{Z}_{\mathrm{border}}(\mathfrak{C})$ 的每个元素都属于以下三种类型之一：

- $\mathcal{Z}_{\text{边界}}(\mathfrak{C}')$ 中的一个元素；根据归纳假设，这类元素至多有 $35\Lambda_{\text{固定}}$ 个；

与 $A' \cap B'$ 相交；此类元素至多有 $\Lambda_{\text{固定}}$ 个；

位于 $\mathcal{Z}_{\text{内部}}(\mathfrak{C}'')$；此类元素至多有 $\sqrt{k} \leqslant \Lambda_{\text{固定}}$ 个，因为 $\mathfrak{C}''$ 并非主要。

断言的证明至此完成。 」

断言 6.11. 如果运行成功，那么输出树分解的每个袋至多包含

$$\mathcal{O}_{d,p}\left(\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log^{1+3[d\geqslant 1]} k \cdot \log^{[d\geqslant 1]}(k+\Delta)\right).$$

$\bigcup \mathcal{Z}$ 的顶点。

断言的证明。 由于输出分解的每个袋都是集合 R 与分隔集 $A \cap B$ (非叶节点)的某一部分，或至多 $4\sqrt{k} \leqslant 4\Lambda_{\text{pinned}}$ 个顶点(叶节点)的并集，因此，根据断言 6.8 和 6.10，每个输出分解的袋与 $41\Lambda_{\text{pinned}}$ 的交集至多包含 $\mathcal{Z}$ 个元素。若 $\bigcup \mathcal{Z}$，则 $d = 0$ 的顶点数也满足相同的上界；而 $d \geqslant 1$ 的上界由性质 (10) 和上界 (20) 得出。 」

分析：成功概率。接下来需要界定此次运行成功的概率。随后的两个断言给出了两个关键观察：需要在模式模式下采取操作，或需要在常规模式下作出“丰富”选择的调用次数，至多为 $\mathcal{O}_{d,\ p}(|\mathcal{Z}|/\sqrt{k})$。

断言 6.12。在一次成功运行中，处于常规模式且 $A^\circ \cap B^\circ$ 丰富的调用次数至多为 $\mathcal{O}_{d,\ p}(|\mathcal{Z}|/\sqrt{k})$。当 $d = 0$ 时，该上界仅为 $|\mathcal{Z}|/\sqrt{k}$。

证明。首先注意，簇 $K \in \mathcal{Z}$ 在递归树中沿着从根出发的一条向下路径（当 $K \cap Y_0 \neq \emptyset$ 时，该路径可能为空）属于集合 $\mathcal{Z}_{\text{inside}}$，直至遇到叶调用，或遇到 K 被计算所得集合 C 相交的调用。将最后这个调用记为 $\mathfrak{C}(K)$。如果 $\mathfrak{C}(K)$ 是非叶调用，处于常规模式且丰富，那么由 (1717) 可知，该调用等于 $\Omega_{d,\ p}(\sqrt{k})$ 中除 $K \in \mathcal{Z}$ 之外的其他集合 $\mathfrak{C}(K)$。断言得证。

该界变为 $|\mathcal{Z}|/\sqrt{k}$，因为 $d = 0$，此时 $\Omega_{d,\ p}(\sqrt{k})$ 中的（1717）实际上只是 $\sqrt{k}$. □

断言 6.13。 在一次成功运行中，模式模式下存在 $\mathcal{O}(|\mathcal{Z}|/\sqrt{k})$ 个调用，其中成功运行的定义要求执行操作。（即，$|\mathcal{Z}_{\text{border}}| > 30\Lambda_{pinned}$）。

断言的证明。 如果某个调用是主要调用、处于模式模式下，并且成功运行的定义要求执行操作，则称其为有趣的 调用。

递归树是一棵二叉树——每个非叶节点恰好有两个递归子调用。对于一个带有子节点 $\mathfrak{C}_1$ 和 $\mathfrak{C}_2$ 的非叶调用 $\mathfrak{C}$，我们将具有较大 $\mathcal{Z}_{\text{inside}}(\mathfrak{C}_i)$ 的子节点指定为重 子节点；若两者相等，则任意打破平局；另一个子节点称为轻 子节点。移除节点与其轻子节点之间的边后，递归树分解为重路径。重路径的首节点 是其最顶部节点。

现在考虑一条重路径 P 及其首节点 $\mathfrak{C}_P$。若 $\mathfrak{C}_P$ 不是主要的，则在 P 上没有有趣的调用，因此假设 $\mathfrak{C}_P$ 是主要的。注意，若 $\mathfrak{C}$ 是对 P的一次有趣调用，且 $\mathfrak{C}_h$ 和 $\mathfrak{C}_l$分别是 $\mathfrak{C}$的重子节点和轻子节点，则引理 6 106。.10 蕴含

$$|\mathcal{Z}_{\text{border}}(\mathfrak{C}_l) - \mathcal{Z}_{\text{all}}(\mathfrak{C}_h)| \geqslant 6\Lambda_{\text{pinned}}.$$

因此，

$$|\mathcal{Z}_{\text{all}}(\mathfrak{C}_h)| \leqslant |\mathcal{Z}_{\text{all}}(\mathfrak{C})| - 6\Lambda_{\text{pinned}}.$$

这意味着， P 上的有趣调用数量有上界

$$\frac{|\mathcal{Z}_{\text{all}}(\mathfrak{C}_P)|}{6\Lambda_{\text{pinned}}} \leqslant \frac{|\mathcal{Z}_{\text{free}}(\mathfrak{C}_P)| + |\mathcal{Z}_{\text{border}}(\mathfrak{C}_P)|}{6\Lambda_{\text{pinned}}} \leqslant \frac{|\mathcal{Z}_{\text{free}}(\mathfrak{C}_P)|}{6\Lambda_{\text{pinned}}} + \frac{37}{6}.$$

这里，最后一个不等式由断言 6.10 得出6.10。。。

定义重路径 P 的层级为 $\lfloor \log_2 |\mathcal{Z}_{\text{free}}(\mathfrak{C}_P)| \rfloor$。注意，若 $\mathfrak{C}_P$ 不是根，且 P' 是包含 $\mathfrak{C}_P$ 的父节点的重路径，则 P' 的层级严格大于 P 的层级。由此可知，对于固定的层级 i，递归树中沿层级为i 的路径 P 进行的调用 $\mathfrak{C}_P$ 之间不存在祖先/后代关系，因此集合$\mathcal{Z}_{\text{free}}(\mathfrak{C}_P)$ 两两不交。因此，我们可以对这些调用所在路径的层级求和，从而界定有趣调用的总数，如下所示：

$$\begin{aligned}
&\sum_{i=\lfloor \log_2 \sqrt{k} \rfloor}^{\lfloor \log_2 |\mathcal{Z}| \rfloor} \frac{|\mathcal{Z}|}{2^i} \cdot \left(\frac{2^{i+1}}{6\Lambda_{\text{pinned}}} + \frac{37}{6} \right) \\
&\qquad \leqslant \sum_{i=\lfloor \log_2 \sqrt{k} \rfloor}^{\lfloor \log_2 k \rfloor} \frac{2|\mathcal{Z}|}{6\Lambda_{\text{pinned}}} + \frac{37|\mathcal{Z}|}{6 \cdot 2^i} \\
&\qquad = \mathcal{O}\left(\frac{|\mathcal{Z}|}{\sqrt{k}} \right).
\end{aligned}$$

这就完成了证明。 ⌟

分析中的另一个要素是关于次要调用和主要调用数量的以下简单界限。

断言 6.14。 在一次成功运行中，主要调用和次要调用的数量至多为

$$|\mathcal{Z}| \cdot 23 \log |V(G)|.$$

断言的证明。设 $\mathfrak{C}$ 是一次次要调用或主要调用。也就是说， $\mathcal{Z}_{\text{free}}(\mathfrak{C}) \neq \emptyset$。将 $\mathfrak{C}$ 计入 $\mathcal{Z}_{\text{free}}(\mathfrak{C})$ 中的一个元素。

$\mathcal{Z}$ 中的一个元素最多只能属于递归树中的一条向上路径。由于递归深度不超过 23 log $|V(G)|$，因此每个 $K \in \mathcal{Z}$ 被计入的次数不超过 23 log $|V(G)|$。由此可得该结论。⌟

命题 6126。.12、6.136.13和 6146。.14 蕴含了以下关于运行成功概率的下界。

$$\overbrace{\left(k \cdot \left(\mathcal{O}_{d,p}(k\Delta)^{\mathcal{O}_{d,p}(\log k)}\right)\right)^{-\mathcal{O}_{d,p}\left(\frac{|\mathcal{Z}|}{\sqrt{k}}\right)}}^{\text{regular and rich calls}} \cdot \overbrace{\left(k \cdot 2\Lambda_R \cdot \left(\mathcal{O}_{d,p}(k\Delta)^{\mathcal{O}_{d,p}(\log k)}\right)\right)^{-\mathcal{O}_{d,p}\left(\frac{|\mathcal{Z}|}{\sqrt{k}}\right)}}^{\text{pattern and mandate an action calls}}$$

$$\cdot \overbrace{\left(1 - \frac{1}{k}\right)^{|\mathcal{Z}| \cdot 23 \log |V(G)|}}^{\text{regular and poor, pattern and minor calls}} \geqslant (k\Delta)^{-\mathcal{O}_{d,p}\left(\frac{|\mathcal{Z}| \log k}{\sqrt{k}}\right)} \cdot |V(G)|^{-\mathcal{O}\left(\frac{|\mathcal{Z}|}{k}\right)}.$$

在上述内容中：

- 在第一项中， k 因子对应于选择“丰富”情形的抛硬币步骤。
- 在第二项中， k 因子对应于执行某项操作时抛硬币的概率， $2\Lambda_R$是选择满足 $|A^\circ \cap W|$, $|B^\circ \cap W| \geqslant \theta$的（$2|R|$）$^{-1}$ 概率下界（A°， B°） $W \in \mathcal{X}$。

- 在第三项中，$1-\frac{1}{k}$ 因子对应于抛硬币决定是选择常规模式下的贫乏情形，还是在模式模式下什么也不做。

这就完成了引理 6161的证明。。 □

6.2 将结果推广到 H-子式自由图，适用于 $d=0$

本节将引理 616的结果.1 推广到排除某个固定子式的图这一一般情形，从而证明引理 1212。。本质上，这需要在引理 316的树分解.1 的每个躯干中应用引理 613。.1，然后将这些应用的结果粘合起来。

设 $(T,\ \beta)$ 是图 G 的一个有根树分解。对于非根节点 t，令 $\pi(t)$ 为 t 的父节点。回忆一下记号 $\sigma(t)=\beta(t)\cap\beta(\pi(t))$，其中根节点对应 $\sigma(r)=\emptyset$。

下面这个简单的引理有助于在树分解上执行粘合过程。

引理 6.15。 设给定一个图 G、一个整数 $k\geqslant 2$、一个有根树分解(T,β)，其分解对象为 G，最大粘合度大小至多为 q；并且对于每个 $t\in V(T)$，给定 t 的躯干的一个诱导子图G_t，以及 G_t 的一个树分解 (T_t,β_t)。令

$$\ell:=\max_{t\in V(T)}\max_{s\in V(T_t)}|\beta(s)|.$$

然后，可以在多项式时间内计算出 G 的一个诱导子图 G'，以及 G' 的一个最大袋大小至多为 $\ell+q$ 的树分解 (T',β')，并保证：对于每个 $Z\subseteq V(G)$，如果对于每个 $t\in V(T)$ 都有 $Z\cap(\beta(t)-\sigma(t))\subseteq V(G_t)$，那么 $Z\subseteq V(G')$。此外，如果对于每个 $t\in V(T)$，(T_t,β_t) 的每个袋至多包含 Z 的 ℓ_Z 个顶点，那么 (T',β') 的每个袋至多包含 Z 的 ℓ_Z+q 个顶点。

证明。令

$$G'=G\left[\bigcup_{t\in V(T)}V(G_t)-\sigma(t)\right].$$

固定 $Z\subseteq V(G)$，并令对于每个 $t\in V(T)$ 都有 $Z_t=Z\cap(\beta(t)-\sigma(t))$。显然，如果 $Z_t\subseteq V(G_t)$，那么 $Z_t\subseteq V(G')$。注意，$(Z_t)_{t\in V(T)}$是 Z 的一个划分，因为 $(\beta(t)-\sigma(t))_{t\in V(T)}$是 $V(G)$ 的一个划分。因此，$Z\subseteq V(G')$。

对于每个 $t\in V(T)$，我们按从上到下的顺序定义躯干的一个诱导子图 G'_t，具体如下。对于根，设 $G'_t=G_t$。否则，令 G'_t 为由 $V(G_t)-\sigma(t)$ 与 $\sigma(t)\cap V(G'_{\pi(t)})$ 的并集诱导出的 t 的躯干诱导子图。按如下方式将 (T_t,β_t) 修改为 G'_t的树分解 (T'_t,β'_t)：若 t是 T的根，则设 $(T'_t,\beta'_t)=(T_t,\beta_t)$；否则，从 (T_t,β_t) 的袋中移除 $\sigma(t)\cap V(G_t)$ 的顶点，并将 $\sigma(t)\cap V(G'_{\pi(t)})$ 的顶点插入每个袋中。这样，每个袋的大小至多增加 q 个顶点，且 (T'_t,β'_t) 的每个袋至多包含 Z的 ℓ_Z+q 个顶点。

将树分解 (T'_t,β'_t) 合并为 G' 的树分解 (T',β')，具体如下：对于每个非根节点 t，选取 $T'_{\pi(t)}$ 中一个袋包含 $\sigma(t)\cap V(G'_{\pi(t)})$ 的节点（这样的节点存在，因为粘合集在躯干中是团），并将 T'_t 中任意一个节点与这个选定的节点相连。元组 $(G',(T',\beta'))$ 即为所需的输出。 □

现在，我们使用引理 6156。.15 将引理 616。.1 推广到小直径的无顶点子式图。

引理 6.16。 对于每个顶点图 H，都存在一个常数 $d_H>0$ 以及一个随机多项式时间算法。给定一个直径为 $\Delta\geqslant 1$ 的 H-子式自由图 G 和一个整数 $k\geqslant 2$,

该算法输出 G 的一个诱导子图 G'，以及 G' 的一个树分解(T,β)，其最大袋大小至多为

$$d_H(k\log k+\Delta)$$

使得对于每个大小至多为 k 的 $Z\subseteq V(G)$，其概率至少为

$$\left((k\Delta)^{\frac{d_H|Z|\log k}{\sqrt{k}}}|V(G)|^{\frac{d_H|Z|}{k}}\right)^{-1}$$

我们有 $Z\subseteq V(G')$，且 (T,β) 的每个袋都至多包含 $d_H\left(\sqrt{k}\log k+\frac{\Delta}{\sqrt{k}}\right)\log k$ 个 Z 的顶点。

证明。为了进行分析，固定陈述中的 $Z\subseteq V(G)$。

应用引理 3 13。.1 到 G，得到一个树分解 (T,β)，其最大粘合度大小为 p_H，且每个袋的躯干都是 p_H-近似可嵌入且不含顶点，其中常数p_H 仅取决于 H 。

将 T 根在任意一个袋上。对于 $t\in V(T)$，令 $Z_t=Z\cap(\beta(t)-\sigma(t))$。

对于每个 $t\in V(T)$，抛一枚硬币：以 $1/k$ 的概率称 t 为充足，以剩余概率 $1-1/k$ 称 t 为稀缺。

若 t 稀缺，我们令 G_t 为 t 的（完整）躯干，并令 (T_t,β_t) 为引理 3 43 4.. 给出的 G_t 的宽度为 $\mathcal{O}_H(\Delta)$ 的树分解。若 t 充足，则将引理 6 16..1 应用于 t 和 k 的躯干，得到一个诱导子图 G_t 以及 G_t 的树分解 (T_t,β_t)。回想一下，t 的躯干是不含顶点的 p_H-近似可嵌入；令 c_H 为引理 6 16 1.. 得出的常数。

将引理 6 156..15 应用于 G、(T,β) 和 $(G_t,(T_t,\beta_t))$，其中 $t\in V(T)$，得到一个诱导子图 G' 及其树分解 (T',β')。由于每个分解 (T_t,β_t) 的每个袋大小均为 $\mathcal{O}_H(k\log k+\Delta)$，且 (T,β) 的每个粘合大小均为 $\mathcal{O}_H(1)$，所以 (T',β') 的每个袋大小均为 $\mathcal{O}_H(k\log k+\Delta)$。此外，引理 6 156 15 .. 指出：若对于每个 $t\in V(T)$，(T_t,β_t) 的每个袋都包含 Z 的 $\mathcal{O}_H(\left(\sqrt{k}\log k+\frac{\Delta}{\sqrt{k}}\right)\log k)$ 个顶点，则 (T',β') 也满足相同的界（但隐藏在 $\mathcal{O}_H$ 记号中的常数更大）。

现在，我们给出这样的概率下界：对每个 $t\in V(T)$，都有 $Z_t\subseteq V(G_t)$，并且 $(T_t,\ \beta_t)$ 的每个袋都包含 $\mathcal{O}_H(\left(\sqrt{k}\log k+\frac{\Delta}{\sqrt{k}}\right)\log k)$ 个 Z_t 的顶点。为此，我们注意到以下断言。

断言 6.17. 存在一个仅依赖于 H 的常数 $c'_H>0$，使得以下结论成立。对于每个 $t\in V(T)$ 以及每个大小至多为 k 的 $Z'\subseteq\beta(t)$，满足 $Z'\subseteq V(G_t)$ 且 (T_t,β_t) 的每个袋都至多包含 $c'_H\left(\sqrt{k}\log k+\frac{\Delta}{\sqrt{k}}\right)\log k$ 个 Z' 的顶点的概率至少为

$$\left((k\Delta)^{\frac{c'_H|Z'|\log k}{\sqrt{k}}}|V(G)|^{\frac{c'_H|Z'|}{k}}\right)^{-1}.$$

断言的证明。对于 $Z'=\emptyset$，所承诺的保证以概率 1 成立（无论我们选择“稀缺”还是“充足”），并且该表达式对任意 $c'_H>0$ 的值均为 1，因此断言成立。现在假设 $Z'\neq\emptyset$。

如果 $1\leqslant|Z'|\leqslant\sqrt{k}$，我们希望猜测 t 为“稀缺”。这种情况发生的概率为

$$1-\frac{1}{k}\geqslant 2^{-2/k}\geqslant 2^{-2\frac{|Z'|}{\sqrt{k}}}\geqslant k^{-2\frac{|Z'|}{\sqrt{k}}}.$$

（我们在上文用过 $k\geqslant 2$。）那么，由于 $V(G_t)=\beta(t)$，显然有 $Z'\subseteq V(G_t)$；而且，显然 (T_t,β_t) 的每个袋至多包含 $\sqrt{k}$ 个 Z' 的顶点，因为 $|Z'|\leqslant\sqrt{k}$。

如果 $|Z'| \geqslant \sqrt{k}$，我们想要猜测 t 是“充足”的。这种情况发生的概率为

$$\frac{1}{k} = k^{-1} \geqslant k^{-\frac{|Z'|}{\sqrt{k}}}.$$

接下来注意，若引理 6 1[6。.1 对 Z'给出的保证成立，那么只要满足 $c'_H \geqslant c_H$ (回想一下， c_H 是定理 6 1[61。所得到的常数)。结合之前的猜测，该结论以至少.....的概率成立

$$\left((k\Delta)^{\frac{(c_H+2)|Z|\log k}{\sqrt{k}}}|V(G)|^{\frac{c_H|Z|}{k}}\right)^{-1}$$

因此，该结论以 $c'_H = c_H + 2$ 成立。 ⌟

回想一下，$(Z_t)_{t\in V(T)}$ 是 Z 的一个划分。由于每个 $t \in V(T)$ 的随机选择彼此独立，对于每个 $t \in V(T)$ 都有 $Z_t \subseteq V(G_t)$，且 (T_t, β_t) 的每个袋中至多包含 $c'_H\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log k$ Z_t 的顶点的概率至少为：

$$\prod_{t\in V(T)}\left((k\Delta)^{\frac{c'_H|Z_t|\log k}{\sqrt{k}}}|V(G)|^{\frac{c'_H|Z_t|}{k}}\right)^{-1} = \left((k\Delta)^{\frac{c'_H|Z|\log k}{\sqrt{k}}}|V(G)|^{\frac{c'_H|Z|}{k}}\right)^{-1}.$$

证明至此完成。 □

现在，我们通过经典的 Baker 分层技术，去掉引理 6166。.16 中的直径假设。

引理 6.18. 对于每个顶点图 H，都存在一个常数 $d'_H > 0$以及一个随机化多项式时间算法。给定一个排除 H 子式的图 G 和一个整数 $k \geqslant 2$，该算法会输出 G 的一个诱导子图 G'，以及 G' 的一个树分解 (T, β)，其最大袋大小至多为 $d'_H k\log k$，使得对于每个大小至多为 k 的 $Z \subseteq V(G)$，其概率至少为

$$\left(k^{\frac{d'_H|Z|\log k}{\sqrt{k}}}|V(G)|^{\frac{d'_H|Z|}{k}}\right)^{-1}$$

我们有 $Z \subseteq V(G')$，且 (T, β) 的每个袋都至多包含 Z 中的 $d'_H\sqrt{k}\log^2 k$ 个顶点。

证明. 固定一个排除的顶点图 H。通过可能向 H 添加一些边，我们假设H 是二连通的；注意，这样做仍可使 H 保持为顶点图。给定作为输入的 G 和 k，其中 G 是排除 H 子式的图。为了分析，固定一个大小至多为k的 $Z \subseteq V(G)$。

对于 G 的每个连通分量 C，新建一个顶点 v_C，使其与 C 中任意选取的一个顶点相邻，并从 v_C 在 C 中进行广度优先搜索。由于 H 是二连通的，添加顶点 v_C 不会破坏 G 是 H-子式无关这一假设。对于 $j \geqslant 0$，令 L_j 为与其对应顶点 v_C 的距离恰好为 j 的顶点集合；$(L_j)_{j\geqslant 0}$ 是 $V(G)$ 的一个划分。我们为 $j < 0$ 定义 $L_j = \emptyset$。记 $L_{\leqslant i} := \bigcup_{j\leqslant i} L_j$。

均匀随机地猜测一个余数 $a \in \{0, 1, \ldots, k\}$，使得 $\bigcup_{i\geqslant 0} L_{a+(k+1)i}$ 不包含 Z 的任何顶点。该猜测成功的概率至少为

$$1 - \frac{|Z|}{k+1} \geqslant 2^{-2\frac{|Z|}{k}\log k} \geqslant 2^{-2\frac{|Z|}{\sqrt{k}}\log k}. \tag{22}$$

（在第一个不等式中，我们使用了 $k \geqslant 2$ 和 $|Z| \leqslant k$。）

对于 G 和 $i \geqslant 0$ 的一个连通分量 C，我们将图 $G_{C,\,i}$ 定义为将图 $G[C \cap L_{\leqslant a+(k+1)(i-1)}]$ 收缩到 v_C 上所得的图，其中 $i > 0$；此次收缩得到的顶点记为 $v_{C,\,i}$，并令 $v_{C,\,0} = v_C$ 属于 $G_{C,\,0}$。注意，

$$V(G_{C,i}) = \{v_{C,i}\} \cup \left(\bigcup_{j=a+1+(k+1)(i-1)}^{a-1+(k+1)i} C \cap L_j \right).$$

此外，$V(G_{C,\,i})$ 的半径至多为 k，因此直径至多为 $2k$。如果对余数 a 的猜测正确，则集合 Z 包含于顶点集 $V(G_{C,\,i} - v_{C,\,i})$的并集中。[4] 令 $\mathcal{D}$为由满足 $V(G_{C,\,i}) - \{v_{C,i}\} \neq \emptyset$的数对 $(C,\ i)$ 构成的族。

对于每个 $(C,\ i) \in \mathcal{D}$，我们对 $G_{C,\,i}$应用引理 6 166..16，得到一个诱导子图 $G'_{C,\,i}$ 及其树分解 $(T'_{C,\,i},\ \beta'_{C,\,i})$。我们从每个 $G'_{C,\,i}$ 及其对应的树分解中移除 $v_{C,\,i}$，并将这些树分解任意连接成一个树分解 $(T',\ \beta')$，它是图的不交并 $(G'_{C,\,i} - \{v_{C,i}\})_{(C,i)\in\mathcal{D}}$的树分解，而该不交并是 G 的诱导子图。

对于每个 $(C,\ i) \in \mathcal{D}$，顶点 $v_{C,\,i}$ 不属于 Z，根据引理 6 166..16，我们有 $Z \cap V(G_{C,\,i}) \subseteq V(G'_{C,\,i})$，并且 $(T'_{C,\,i},\ \beta'_{C,\,i})$ 的每个袋至多包含 $Z \cap V(G_{C,\,i})$ 的 $d_H\sqrt{k}\log$ 个顶点，其概率至少为

$$\left((k \cdot 2k)^{\frac{d_H |Z \cap V(G_{C,i})| \log k}{\sqrt{k}}} |V(G)|^{\frac{d_H |Z \cap V(G_{C,i})|}{k}} \right)^{-1}.$$

对每个 $(C,\ i) \in \mathcal{D}$ 将上述结果相乘，可得 $Z \subseteq V(G')$，并且 $(T',\ \beta')$ 的每个袋至多包含 Z 的 $d_H\sqrt{k}\log$ 个顶点，其概率至少为

$$\left(k^{\frac{3d_H |Z| \log k}{\sqrt{k}}} |V(G)|^{\frac{d_H |Z|}{k}} \right)^{-1}.$$

将上述结果与对于正确选择 a 的估计式（2222）相乘，我们便得到关于 $d'_H = 3d_H + 2$ 的所需结论。 □

现在我们可以证明引理 1212。。。

引理 *1 2* 的证明*12*。。。固定一个图 H。令 G 和 $k \geqslant 2$ 为输入中给定的对象。将引理 3 13。.1 应用于G；令 p_H 为该常数，并令 $(T,\ \beta)$ 为所得的分解。

将根 T 设在任意节点处。固定大小至多为 k 的 $Z \subseteq V(G)$，并令每个 $t \in V(T)$ 都满足 $Z_t = Z \cap (\beta(t) - \sigma(t))$。

设 H' 为引理 3 23..2 针对以下图给出的顶点图：这些图是 p_H-近似可嵌入且不含顶点。

对每个 $t \in V(T)$，按如下方式计算一个诱导子图 G_t 及其树分解 (T_t, β_t)。令 G_t^1 为 t 的躯干，并令 A_t 为 G_t^1 中至多含 p_H 个顶点的集合，使得 $G_t^2 := G_t^1 - A_t$ 是不含顶点的 p_H-近似可嵌入图。将引理 6 186 18 .. 应用于 H'-子式自由图 G_t^2，得到一个常数 $f_H := d'_{H'}$ 及一个元组 $(G'_t, (T'_t, \beta'_t))$。通过将 A_t 的所有顶点添加到 G_t 以及该分解的每个袋中，将 $(G'_t, (T'_t, \beta'_t))$ 修改为 $(G_t, (T_t, \beta_t))$。

根据引理 6 186..18，$Z_t - A_t \subseteq V(G'_t)$ 的概率，以及 (T'_t, β'_t) 中每个袋至多包含 $f_H\sqrt{k}\log^2 k$ 个 $Z_t - A_t$ 顶点的概率至少为

$$\left(k^{\frac{f_H |Z_t - A_t| \log k}{\sqrt{k}}} |V(G)|^{\frac{f_H |Z_t - A_t|}{k}} \right)^{-1}.$$

[4]我们添加顶点 v_C，而不是直接在 C 中任取一个顶点作为 v_C 的唯一原因，是为了确保 $v_C \notin Z$，这样表述会更加简洁。

因此，这也是 $Z_t \subseteq V(G_t)$ 的概率下界，因为 $A_t \subseteq V(G_t)$。此外，由于 $|A_t| \leqslant p_H$，以以上概率，(T_t, β_t) 中的每个袋至多包含$f_H\sqrt{k}\log^2 k + p_H \leqslant (f_H + p_H)\sqrt{k}\log^2 k$ 个 Z_t（回想一下 $k \geqslant 2$）。

对所有 $t \in V(T)$ 相乘，由于 $(Z_t)_{t\in V(T)}$ 是 Z 的一个划分，因此对于每个 $t \in V(T)$，上述事件发生的概率至少为

$$\left(k^{\frac{f_H|Z|\log k}{\sqrt{k}}}|V(G)|^{\frac{f_H|Z|}{k}}\right)^{-1} \geqslant \left(k^{f_H\sqrt{k}\log k}|V(G)|^{f_H}\right)^{-1}.$$

若此情况发生，元组 $(G_t, (T_t, \beta_t))$ 满足引理 6 156..15 的前提。引理 6 156 15.. 给出一个元组 $(G', (T', \beta'))$，使得 $Z \subseteq V(G')$，并且 (T', β') 的每个袋至多包含 $(f_H+p_H)\sqrt{k}\log^2 k+p_H \leqslant (f_H+2p_H)\sqrt{k}\log^2 k$ 个 Z 的顶点。此外，(T', β') 的每个袋的大小均为 $\mathcal{O}_H(k\log k)$。由此证明完成。

□

6.3 提升至 $K_{3,h}$-子式自由图，其中 $d \geqslant 1$

在本节中，我们证明定理 1 313。。证明遵循上一节的总体思路：我们首先应用相对标准的 Baker 技巧，将直径限制为关于 k的线性量，计算定理 3 13。.1中的分解，并将定理 6 16。.1 应用于各袋的躯干。

不过，我们需要处理两个复杂之处。首先，袋的躯干可能不再是$K_{3,\ h}$-子式-自由的，因此我们需要更加谨慎，将定理 6 16。.1 应用于躯干的一个子图，该子图是原图的子式。其次，对于 $d \geqslant 1$，d-簇可能跨越定理 6 161。。. 所构造的树分解中的许多袋。我们通过指定与某个簇相交的最顶层袋对该簇“负责”来解决这个问题；其他所有袋则通过将 Y_0 设为其顶层粘合，以便应用定理 6 161。。。

我们首先将定理 6116..1 包装成定理 3131..的一个分解。这是引理 6 16616..的一个类似结果。

引理 6.19。 对于任意整数 $d \geqslant 1$ 和 $h \geqslant 3$，存在一个常数 $c_{d,\ H} > 0$ 以及一种随机化多项式时间算法，给定一个 $K_{3,\ h}$-子式自由图 G，其直径为 $\Delta \geqslant 1$，以及一个整数 $k \geqslant 2$，输出 G 的一个诱导子图 G' 和一个树分解(T, β)，其最大袋大小至多为

$$c_{d,H}(k\log k + \Delta)$$

使得对于任意由 d-簇组成且满足 $|\mathcal{Z}| \leqslant k$的族 $\mathcal{Z}$，其概率至少为

$$\left((k\Delta)^{\frac{c_{d,H}|\mathcal{Z}|\log k}{\sqrt{k}}}|V(G)|^{\frac{c_{d,H}|\mathcal{Z}|}{k}}\right)^{-1}$$

我们有 $\bigcup\mathcal{Z} \subseteq V(G')$，并且 (T, β) 的每个袋至多包含 $c_{d,\ H}\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log^6 k$个 $\bigcup\mathcal{Z}$的顶点。

证明。 为便于分析，固定如陈述中所示的 $\mathcal{Z}$。

将定理 3 13..1 应用于 $K_{3,\ h}$-子式自由图 G，得到一个最大粘合度大小为 p_h 的树分解 $(T,\ \beta)$，使得每个袋的躯干都是不含顶点的 p_h-近似可嵌入图，其中常数 p_h 仅取决于 h。（注意，$K_{3,\ h}$ 是一个顶点图。）

将 T 置于任意一个袋中。不失一般性，我们可以假设 $(T,\ \beta)$ 满足以下连通性：对于每个 $t \in V(T)$，如果 $T_{\downarrow t}$ 是以 t为根的 T 子树，那么 $D_{\downarrow t} := \bigcup_{t'\in V\ (T_{\downarrow t})} \beta(t') - \sigma(t)$ 在 G 中连通，且 $N_G(D_{\downarrow t}) = \sigma(t)$。

对于每个 $t \in V(T)$，按如下步骤进行。首先，抛一枚硬币：以 $1/k$ 的概率称 t 为充足，以剩余概率 $1-1/k$ 称 t 为稀缺。

若 t 是稀缺的，我们令 G_t 为 t 的整个躯干，且令(T_t, β_t) 为由推论 3.7 得到的 G_t 的树分解37。。。

若 t 充足，我们按如下方式进行。令 $G_t^\bullet$ 为 t的躯干。由于 $G_t^\bullet$ 是无顶点的 p_h-近似可嵌入图，根据定理 3.23.2，它不包含某个顶点图 H 作为子式；特别地， $G_t^\bullet$ 是 δ_h-退化的，其中 δ_h 仅取决于 h。令 $\preceq$ 为 $G_t^\bullet$的一个退化序：每个 $v \in \beta(t)$在 $G_t^\bullet$ 中至多有 δ_h 个比 v $\preceq$的邻点。

令 $\mathcal{D}_t$ 为 $G-\beta(t)$的连通分量集合。注意， $\mathcal{D}_t$ 包含所有属于 t的子节点 t' 对应的连通分量$D_{\downarrow t'}$，以及包含在 $T-\{t\}$ 的连通分量的袋中的一些（可能很多）连通分量；该连通分量包含 t 的父节点（除非 t 是 T的根）。

令 G_t^* 按如下方式构造：我们从 G 开始，并对每个 $D \in \mathcal{D}_t$，将 D收缩到 $N_G(D)$中按 $\preceq$取最大值的顶点 v_D 上。这是可行的，因为 G是连通的。显然， G_t^*既是 G 的子式，也是 t 的躯干的子图（因为 D 包含在 $T-\{t\}$ 的一个连通分量的袋中，而 $N_G(D)$包含在该连通分量与 t之间那条边的粘合中）。因此， G_t^* 既是$K_{3,\ h}$-子式自由的，也是无顶点的 p_h-近似可嵌入图。 G_t^* 的直径至多为 Δ，因为它仅由收缩 G 构造而成。此外，对每个 $v \in \beta(t)$，我们有如下界：

$$|\{N_G(D) \mid D \in \mathcal{D}_t \wedge v_D = v\}| \leqslant 2^{\delta_h}. \tag{23}$$

事实上，如果 $v = v_D$，那么 v 是 $\preceq$-最大顶点，于 $N_G(D)$，因此 $N_G(D)-\{v\}$ 是 $G_t^\bullet$ 中 v 的邻点中 $\preceq$-小于 v的邻点，而这样的邻点至多有 δ_h 个。这证明了 (2323)。

我们对6..1 应用定理 G_t^*，其常数为 $4(d+4)$、 $p := \max(h,\ p_h)$ 以及 $Y_0 := \sigma(t)$。我们将结果记为 G_t 和 $(T_t,\ \beta_t)$。

对每个 $t \in V(T)$，初始化 $X_t := \emptyset$。对于每条边 $st \in E(T)$，若 s 是 t的子节点，

- 若 $v_{D_{\downarrow s}} \in V(G_t)$，则通过向每个包含 $v_{D\downarrow}$的袋中添加 $\sigma(s) \cap V(G_t)$，修改 β_t $_s$.
- 否则，将 $\sigma(s) \cap V(G_t)$ 添加到 X_t中。

上述第一种方案使$(T_t、\ \beta_t)$ 的最大袋大小至多增加 $p_h 2^{\delta_h}$ 倍，这归功于（2323），同时创建一个节点 $t_{s\nearrow t}$，其袋大小为 β_t，且包含 $\sigma(s) \cap V(G_t)$。在第二种情况下，我们将其记为 $t_{s\nearrow t}$ 以及 $(T_t、\ \beta_t)$的任意节点。

我们定义

$$G' := G\left[\left(\bigcup_{t \in V(T)} (V(G_t) - \sigma(t))\right) - \left(\bigcup_{t \in V(T)} X_t\right)\right].$$

对于 $t \in V(T)$，令 β_t' 等于将 β_t 限制到 $V(G') \cap V(G_t)$。我们从所有分解 $(T_t,\ \beta_t')$ 的不交并构造 G' 的树分解 $(T',\ \beta')$：对于每个 $st \in E(T)$，若 s 是 t 的子节点，则在 T_s 的根与 T_t 的节点 $t_{s\nearrow t}$ 之间添加一条边。

由 $t_{s\nearrow t}$ 的定义可知，$(T',\ \beta')$ 是 G' 的树分解：要么 $\beta_t(t_{s\nearrow t})$ 包含 $\sigma(s) \cap V(G_t)$ 的所有顶点，因此 $\beta_t'(t_{s\nearrow t})$ 包含 G' 中剩余的 $\sigma(s)$ 的所有顶点；要么 $\sigma(s) \cap V(G_t) \subseteq X_t$，且 $\beta(st)$ 中没有顶点属于 G'。此外，$(T',\ \beta')$ 的宽度至多为一个乘法常数（取决于 h）乘以各分解 $(T_t,\ \beta_t)$ 的最大宽度。接下来还需分析关于 $\mathcal{Z}$ 的承诺成立的概率。

对于 $K \in \mathcal{Z}$，令 $t_K \in V(T)$ 为满足 $\beta(t) \cap K \neq \emptyset$ 的最顶部节点。对于 $t \in V(T)$，令

$$\mathrm{load}(t) = |\{K \in \mathcal{Z} \mid t_K = t\}|.$$

我们按如下方式构造 G_t^* 中由 $4(d+4)$-簇组成的族 $\mathcal{Z}_t$。首先，从 $\mathcal{Z}_t^1 := \{K \in \mathcal{Z} \mid t_K = t\}$ 开始。其次，对于每个 $K \in \mathcal{Z}_t^1$，令 K^2 为 G_t^* 中的一个 d-簇，它由 K 构造而成；每当对于某个 $D \in \mathcal{D}_t$ 有 $K \cap D \neq \emptyset$ 时，就用 v_D 替换 $K \cap D$。令 $\mathcal{Z}_t^2 = \{K^2 \mid K \in \mathcal{Z}_t^1\}$。第三，对于每个 $K^2 \in \mathcal{Z}_t^2$，按如下方式在 G_t^* 中创建一个 $(d+4)$-簇 K^3：从 $K^3 := K^2$ 开始；对于 T 中 t 的每个子节点 s，如果 $K^2 \cap N_G(D_{\downarrow s}) \neq \emptyset$，则将 $N_G(D_{\downarrow s})$ 的所有顶点加入 K^3。这使直径至多增加 4，因此 $\mathcal{Z}_t^3 = \{K^3 \mid K^2 \in \mathcal{Z}_t^2\}$ 是 G_t^* 中由 $(d+4)$-簇组成的族。最后，将引理 3 83..8 应用于 $\mathcal{Z}_t^3$，得到由两两不交的 $4(d+4)$-簇组成的族 $\mathcal{Z}_t^4$。我们令 $\mathcal{Z}_t := \mathcal{Z}_t^4$。注意，

$$|\mathcal{Z}_t| = |\mathcal{Z}_t^4| \leqslant |\mathcal{Z}_t^3| = |\mathcal{Z}_t^2| = |\mathcal{Z}_t^1| = \mathrm{load}(t). \tag{24}$$

我们称 $t \in V(T)$ 为

高负载的当 $|\mathcal{Z}_t| > \sqrt{k}$；

低负载的当 $1 \leqslant |\mathcal{Z}_t| \leqslant \sqrt{k}$；

零负载的当 $\mathcal{Z}_t = \emptyset$。

根据 (2424)，高负载节点至多有 $|\mathcal{Z}|/\sqrt{k}$ 个，低负载节点至多有 $|\mathcal{Z}|$ 个。我们希望对所有高负载节点猜测为 *充足*，对所有低负载节点猜测为 *稀缺*（我们不关心零负载节点的猜测）。这种情况发生的概率至少为：

$$\left(\frac{1}{k}\right)^{-\frac{|\mathcal{Z}|}{\sqrt{k}}} \cdot \left(1-\frac{1}{k}\right)^{-|\mathcal{Z}|} \geqslant 2^{-\mathcal{O}\left(\frac{|\mathcal{Z}|\log k}{\sqrt{k}}\right)}.$$

此外，对于每个 t 高负载节点，我们希望应用定理 6 16。.1 时，集合 $\mathcal{Z}_t$满足该承诺。对所有这类节点相乘后，这种情况发生的概率至少为

$$\begin{aligned}
&\prod_{t \text{ of high load}} \left((k\Delta)^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}_t|\log k}{\sqrt{k}}\right)} |V(G)|^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}_t|}{k}\right)}\right)^{-1} \\
&\geqslant \prod_{t \text{ of high load}} \left((k\Delta)^{\mathcal{O}_{d,h}\left(\frac{\mathrm{load}(t)\log k}{\sqrt{k}}\right)} |V(G)|^{\mathcal{O}_{d,h}\left(\frac{\mathrm{load}(t)}{k}\right)}\right)^{-1} \\
&\geqslant \left((k\Delta)^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}|\log k}{\sqrt{k}}\right)} |V(G)|^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}|}{k}\right)}\right)^{-1}
\end{aligned}$$

接下来只需说明，这一事件保证 G' 和 $(T',\ \beta')$ 具有所需的性质。

断言 6.20。对于每个 $t, t' \in V(T)$，若 t' 是 t 的后代，则 $X_{t'}$ 与 $N_G^{d+2}[\sigma(t)]$ 不相交。

断言的证明。 注意，$X_{t'} \subseteq \beta(t')$。由于 $N_G^{d+2}[\sigma(t)] \cap \beta(t') \subseteq N_G^{d+2}[\sigma(t')],$，只需证明该断言对 $t' = t$ 成立。

注意，无论我们将 t 称为充足还是稀缺，都有 $N_{G_t^*}^{4(d+4)}[\sigma(t)] \subseteq V(G_t)$。因此，对于 t 的每个子节点 s，如果 $N_G(D_{\downarrow s}) \cap N_G^{d+2}[\sigma(t)] \neq \emptyset$，那么 $N_G(D_{\downarrow s}) \subseteq N_{G_t^0}^{d+4}[\sigma(t)] \subseteq V(G_t)$。因此，$v_{D_{\downarrow s}} \in V(G_t)$，也就是说，在处理边 st 时，我们不会将 $\sigma(s)$ 的顶点插入 X_t。由于 st 的选择在 $N_G(D_{\downarrow s}) \cap N_G^{d+2}[\sigma(t)] \neq \emptyset$ 的条件下是任意的，证明至此完成。」

考虑 $K \in \mathcal{Z}$。如果节点 t_K 的负载较低，那么 $G_t = \beta(t_K)$。否则，根据 $K^3 \in \mathcal{Z}^3_{t_K}$ 的构造，对于 t_K 的每个子节点 s，如果 $N_G(D_{\downarrow s}) \cap K \neq \emptyset$，那么 $N_G(D_{\downarrow s}) \subseteq K^3 \subseteq V(G_{t_K})$。由于 s 的选择是任意的，我们有 $X_{t_K} \cap K = \emptyset$。结合断言 6 206..20，以及对 t_K 的每个真后代 t，都有 $\beta(t) \cap K \subseteq N^d_{G^*_t}[\sigma(t)]$，这一事实，可推出$K \subseteq V(G')$。

现在考虑一个由（T'，β'）构成的袋。根据构造，它等于 $\beta'_t(x)$，其中 $t \in V(T)$ 且$x \in V(T_t)$。我们注意到

$$\beta'_t(x) \cap \bigcup(\mathcal{Z} - \mathcal{Z}^1_t) \subseteq N^d_G[\sigma(t)] \cap \beta(t).$$

因此，上述左侧大小的界为$\underline{\mathcal{O}}_{d,\ h}\left(\left(\sqrt{k}\log\ k + \frac{\Delta}{\sqrt{k}}\right)\log^5 k\right)$可由推论 3 73..7 中分解的性质（当 t 稀缺时）或定理 6 16..1 （当 t 充$\overline{\text{足}}$时）所保证。（注意，这里即使 t 是零负载时结论也成立。）

因此，接下来只需给 $\beta'_t(x) \cap \bigcup \mathcal{Z}^1_t$ 的大小一个上界。

断言 6.21。

$$|\beta'_t(x) \cap \bigcup \mathcal{Z}^1_t| = \mathcal{O}_{d,h}\left(\left(\sqrt{k}\log k + \frac{\Delta}{\sqrt{k}}\right)\log^5 k\right).$$

断言的证明。如果 t 是零负载，则 $\mathcal{Z}^1_t = \emptyset$，无需证明。否则，考虑 $K \in \mathcal{Z}^1_t$，并令 $v \in K \cap \beta'_t(x)$，其中 $x \in V(T_t)$。我们有 $v \in K \cap \beta_t(x)$，或者存在 t 的一个子节点 s，使得 $v \in N_G(D_{\downarrow s})$ 且 $v_{D_{\downarrow s}} \in \beta_t(x)$。于是，$v_{D_{\downarrow s}} \in K^3 \cap \beta_t(x)$，并且根据 (2323)，每个顶点 $u \in \beta(t)$ 至多可以像上文那样充当 $v_{D_{\downarrow s}}$，对应 $p_h \cdot 2^{\delta_h} = \mathcal{O}_h(1)$ 个顶点 v。

因此，只需界定 $\beta'_t(x) \cap \bigcup \mathcal{Z}^3_t$ 的大小即可。如果 t 的负载较低，我们猜测它是稀缺的，并由推论 3 73 7.. . 得到 (T_t, β_t)。这意味着每个 $K^3 \in \mathcal{Z}^3_t$ 对 $K^3 \cap \beta_t(x)$ 的贡献最多为 $\mathcal{O}_{d,h}(1)$ 个顶点。由于$|\mathcal{Z}^3_t| = |\mathcal{Z}^1_t| \leqslant \sqrt{k}$，我们有 $|\beta'_t(x) \cap \bigcup \mathcal{Z}^1_t| = \mathcal{O}_{d,h}(\sqrt{k})$。最后，如果 t 的负载较高，那么根据定理 6 16 1.. . 的承诺，$\beta_t(x) \cap \bigcup \mathcal{Z}^3_t = \beta_t(x) \cap \bigcup \mathcal{Z}^4_t$ 的大小被 $\mathcal{O}_{d,h}\left(\left(\sqrt{k}\log\ k + \frac{\Delta}{\sqrt{k}}\right)\log^5 k\right)$ 所界定。这就完成了断言的证明。」

这就完成了引理 619 的证明619。。。 □

下一个引理是引理 618 的一个类似结果6。.18 并立即完成了定理 1 3 的证明13。。。

引理 6.22。对于每个 $d \geqslant 1$ 和 $h \geqslant 3$，都存在一个常数 $d'_H > 0$ 和一个随机多项式时间算法，该算法给定一个 $K_{3,\ h}$-子式自由图 G 和一个整数 $k \geqslant 2$，输出 G 的一个诱导子图 G' 以及 G' 的一个树分解 (T, β)，其最大袋大小至多为 $d'_H k$ 对数 k；对于具有 $|\mathcal{Z}| \leqslant k$ 的 d- 簇族 $\mathcal{Z}$，其概率至少为

$$\left(k^{\frac{d'_H|\mathcal{Z}|\log k}{\sqrt{k}}}|V(G)|^{\frac{d'_H|\mathcal{Z}|}{k}}\right)^{-1}$$

具有 $\bigcup \mathcal{Z} \subseteq V(G')$，并且 (T, β) 的每个袋至多包含 $d'_H\sqrt{k}$ 对数^6~~k~~个 $\mathcal{Z}$的顶点 we .

证明。对于 C 的每个连通分量，创建一个新顶点 v_C，将其与 C 中任意选定的一个顶点相邻（这不会破坏其为 $K_{3,\ h}$-子式无关图的性质），并从 v_C 在 C 中执行广度优先搜索。对于 $j \geqslant 0$，令 L_j 表示距其对应顶点 v_C 恰好为 j 的顶点集合；$(L_j)_{j\geqslant 0}$ 是 V（G）的一个划分。我们对 $j < 0$ 定义 $L_j = \emptyset$。记作$L_{\leqslant i} : = \bigcup_{j\leqslant i} L_j$。

令 $k' := (d+1)k$。均匀随机地猜测一个余数 $a \in \{0,1,\dots,k'\}$，使得$\bigcup_{i\geqslant 0} L_{a+(k'+1)i}$ 不包含 $\bigcup \mathcal{Z}$ 中的顶点。成功的概率至少为

$$1 - \frac{|\mathcal{Z}|}{k'+1} \geqslant 2^{-2\frac{|\mathcal{Z}|}{k'}\log k'} \geqslant 2^{-\mathcal{O}_d\left(\frac{|\mathcal{Z}|}{\sqrt{k}}\log k\right)}. \tag{25}$$

（在第一个不等式中，我们使用了 $k \geqslant 2$ 和 $|\mathcal{Z}| \leqslant k$。）

对于图 G 的连通分量 C 和 $i \geqslant 0$，我们将图 $G_{C,i}$ 定义为 $G[C \cap L_{\leqslant a-1+(k'+1)i}]$，其中对于 $i > 0$，将图 $G[C \cap L_{\leqslant a+(k'+1)(i-1)}]$ 收缩到 v_C；该收缩所得的顶点记作 $v_{C,i}$，并令 $v_{C,0} = v_C$ 属于 $G_{C,0}$。注意，

$$V(G_{C,i}) = \{v_{C,i}\} \cup \left(\bigcup_{j=a+1+(k'+1)(i-1)}^{a-1+(k'+1)i} C \cap L_j \right).$$

此外，$V(G_{C,i})$ 的半径至多为 k'，因此直径至多为 $2k' = 2(d+1)k$。如果对余数 a 的猜测正确，那么 $\mathcal{Z}$ 的每个簇都包含在顶点集的并集$V(G_{C,i} - v_{C,i})$中。[5] 设 $\mathcal{Z}_{C,i} = \{K \in \mathcal{Z} \mid K \subseteq V(G_{C,i})\}$。设 $\mathcal{D}$ 是由满足 $V(G_{C,i}) - \{v_{C,i}\} \neq \emptyset$ 的数对 (C,i) 构成的族。

对于每个 $(C,i) \in \mathcal{D}$，我们对 $G_{C,i}$应用引理 6 19[6..19]，得到一个诱导子图 $G'_{C,i}$ 及其树分解 $(T'_{C,i}, \beta'_{C,i})$。我们从每个 $G'_{C,i}$ 及其对应的树分解中移除 $v_{C,i}$，并任意连接这些树分解，形成一个单一的树分解 (T', β')，它是图的不交并 $(G'_{C,i} - \{v_{C,i}\})_{(C,i)\in\mathcal{D}}$的树分解，而该不交并是$G$的诱导子图。

对于每个 $(C,i) \in \mathcal{D}$，顶点 $v_{C,i}$不在 $\bigcup \mathcal{Z}$ 中，并且根据引理 6 19[619..]，我们有 $\bigcup \mathcal{Z}_{C,i} \subseteq V(G'_{C,i})$，且 $(T'_{C,i}, \beta'_{C,i})$ 的每个袋都至多包含 $\bigcup \mathcal{Z}_{C,i}$ 中$\mathcal{O}_{d,h}(\sqrt{k}\log^6 k)$ 个顶点，概率至少为

$$\left(\left(k \cdot 2k'\right)^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}_{C,i}|\log k}{\sqrt{k}}\right)} |V(G)|^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}_{C,i}|}{k}\right)} \right)^{-1}.$$

将上述结果对每个 $(C,i) \in \mathcal{D}$ 相乘，可得 $\bigcup \mathcal{Z} \subseteq V(G')$，且 (T', β') 的每个袋都包含 $\mathcal{O}_{d,h}(\sqrt{k}\log^6 k)$ 个 $\bigcup \mathcal{Z}$顶点的概率至少为

$$\left(k^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}|\log k}{\sqrt{k}}\right)} |V(G)|^{\mathcal{O}_{d,h}\left(\frac{|\mathcal{Z}|}{k}\right)} \right)^{-1}.$$

将上述结果与正确选择 a所对应的估计式 (2525) 相乘，即得证明 . □

参考文献

[1] Shyan Akmal、Virginia Vassilevska Williams 和 Nicole Wein。《线性时间检测不相交最短路径及其他》。载于 Karl Bringmann、Martin Grohe、Gabriele Puppis 和 Ola Svensson 编，第 *51* 届国际自动机、语言与编程研讨会，*ICALP 2024*，*2024*年*7*月*8*日至*12*日，爱沙尼亚塔林，第 297 卷，*LIPIcs*，第 9:1–9:17 页。Schloss Dagstuhl - Leibniz信息学中心，2024年。

[2] Omid Amini、Fedor V. Fomin 和 Saket Saurabh。隐式分支与参数化部分覆盖问题。《计算机与系统科学杂志》，77(6):1159–1171，2011年。

[5]我们添加顶点 v_C 而不是直接在 C 中任取一个顶点作为 v_C，唯一的原因是要知道 $v_C \notin \bigcup \mathcal{Z}$，这样表述会更简洁。

[3] Sayan Bandyapadhyay、William Lochet、Daniel Lokshtanov、Saket Saurabh 和 Jie Xue。排除 H 子式的图上的割与环命中问题的亚指数参数化算法。载于 Joseph (Seffi) Naor 和 Niv Buchbinder 编，*2022*年*ACM-SIAM*离散算法研讨会论文集，*SODA 2022*，线上会议 / 美国弗吉尼亚州亚历山德里亚，*2022*年*1*月*9*日至*12*日，第2063–2084页。SIAM，2022年。[4] Matthias Bentert、Fedor V. Fomin 和 Petr A. Golovach。顶点不相交最短路径的紧确近似与核化界。载于 Olaf Beyersdorff、Michal Pilipczuk、Elaine Pimentel 和 Kim Thang Nguyen 编，第*42*届计算机科学理论方面国际研讨会，*STACS 2025*，*2025*年*3*月*4*日至*7*日，德国耶拿，第327卷，*LIPIcs*，第17:1–17:17页。Schloss Dagstuhl - Leibniz信息学中心，2025年。[5] Matthias Bentert、André Nichterlein、Malte Renken 和 Philipp Zschoche。利用几何透镜寻找 k 不相交最短路径。《*SIAM*离散数学杂志》，37(3):1674–1703，2023年。[6] Kristóf Bérczi 和 Yusuke Kobayashi。有向不相交最短路径问题。载于 Kirk Pruhs 和 Christian Sohler 编，第*25*届欧洲算法年度研讨会，*ESA 2017*，*2017*年*9*月*4*日至*6*日，奥地利维也纳，第87卷，*LIPIcs*，第13:1–13:13页。Schloss Dagstuhl - Leibniz信息学中心，2017年。[7] Hans L. Bodlaender、Jesper Nederlof 和 Tom C. van der Zanden。用于嵌入 H子式自由图的亚指数时间算法。载于 Ioannis Chatzigiannakis、Michael Mitzenmacher、Yuval Rabani 和 Davide Sangiorgi 编，第*43*届国际自动机、语言与编程学术讨论会，*ICALP 2016*，*2016*年*7*月*11*日至*15*日，意大利罗马，第55卷，*LIPIcs*，第9:1–9:14页。Schloss Dagstuhl - Leibniz信息学中心，2016年。[8] Nicolas Bourgeois、Aristotelis Giannakos、Giorgio Lucarelli、Ioannis Milis 和 Vangelis Th. Paschos。稠密 k 子图问题的精确与超多项式近似算法。《欧洲运筹学杂志》，262(3):894–903，2017年。[9] Jaroslaw Byrka、Mateusz Lewandowski、Syed Mohammad Meesum、Joachim Spoerhase 和 Sumedha Uniyal。地图图上的斯坦纳树多项式时间近似方案。载于 Yoshiharu Kohayakawa 和 Flávio Keidi Miyazawa 编，*LATIN 2020*：理论信息学——第*14*届拉丁美洲研讨会，巴西圣保罗，*2021*年*1*月*5*日至*8*日，论文集，第12118卷，《计算机科学讲义》，第3–14页。施普林格，2020年。[10] Zhi-Zhong Chen。地图图中独立集的近似算法。《算法杂志》，41(1):20–40，2001年。[11] Zhi-Zhong Chen、Michelangelo Grigni 和 Christos H. Papadimitriou。地图图。《*ACM*杂志》，49(2):127–138，2002年。[12] Rajesh Hemant Chitnis、Andreas Emil Feldmann、Mohammad Taghi Hajiaghayi 和 Dániel Marx。固定数量终端的平面强连通斯坦纳子图（及其扩展）的紧确界。《*SIAM*计算杂志》，49(2):318–364，2020年。[13] Vincent Cohen-Addad、Hung Le、Marcin Pilipczuk 和 Michal Pilipczuk。平面度量与排除子图度量可嵌入树宽为多对数级的度量中，且其期望乘法失真可任意接近1。*CoRR*，abs/2304.07268，2023年。

[14] Marek Cygan、Fedor V. Fomin、Łukasz Kowalik、Daniel Lokshtanov、Dániel Marx、Marcin Pilipczuk、Michał Pilipczuk 和 Saket Saurabh。《参数化算法》。施普林格，2015年。[15] Mark de Berg、Hans L. Bodlaender、Sándor Kisfaludi-Bak、Dániel Marx 和 Tom C. van der Zanden。几何交图中指数时间假设紧致算法与下界的框架。《SIAM计算杂志》，49(6):1291–1331，2020年。[16] Éric Colin de Verdière。有界终端点数的平面图和有界亏格图中的多重割。《Algorithmica》，78(4):1206–1224，2017。[17] Erik D. Demaine、Fedor V. Fomin、Mohammad Taghi Hajiaghayi 和 Dimitrios M. Thilikos。双维参数与局部树宽。《SIAM离散数学杂志》，18(3):501–511，2004年。[18] Erik D. Demaine、Fedor V. Fomin、Mohammad Taghi Hajiaghayi 和 Dimitrios M. Thilikos。平面图和地图图中（k，r）中心问题的固定参数算法。《ACM算法汇刊》，1(1):33–47，2005年。[19] Erik D. Demaine 和 Mohammad Taghi Hajiaghayi。具有有界亏格的图及不含H子式的图上的亚指数参数化算法。《ACM杂志》，52(6):866–893，2005年。[20] Erik D. Demaine 和 Mohammad Taghi Hajiaghayi。局部树宽与线性局部树宽的等价性及其算法应用。载于第*15*届*ACM-SIAM*离散算法年度研讨会（*SODA 2004*），第840–849页。SIAM，2004年。[21] Erik D. Demaine 和 MohammadTaghi Hajiaghayi。双维性理论及其算法应用。《计算机杂志》，51(3):292–302，2008年。[22] Vida Dujmovic、Gwenaël Joret、Piotr Micek、Pat Morin、Torsten Ueckerdt 和 David R. Wood。平面图具有有界队列数。《ACM杂志》，67(4):22:1–22:38，2020年。[23]Zdenek Dvorák 和 Robin Thomas。不含顶点子式的图的列表着色。《CoRR》，abs/1401.1399，2014年。[24] Kord Eickmeyer 和 Ken-ichi Kawarabayashi。地图图上的一阶逻辑模型检测。载于 Ralf Klasing 和 Marc Zeitoun（编），计算理论基础——第*21*届国际研讨会（*FCT 2017*），法国波尔多，*2017*年*9*月*11*日至*13*日，会议论文集，《计算机科学讲义》第10472卷，第204–216页。施普林格，2017。[25] Grzegorz Fabianski、Michal Pilipczuk、Sebastian Siebertz 和 Szymon Torunczyk。用于支配与独立性的渐进算法。载于 Rolf Niedermeier 和 Christophe Paul（编），第*36*届计算机科学理论方面国际研讨会（*STACS 2019*），*2019*年*3*月*13*日至*16*日，德国柏林，《LIPIcs》第126卷，第27:1–27:16页。Schloss Dagstuhl - Leibniz信息学中心，2019年。[26] Fedor V. Fomin、Daniel Lokshtanov、Dániel Marx、Marcin Pilipczuk、Michal Pilipczuk 和 Saket Saurabh。通过低树宽模式覆盖，求解平面图和无顶点子式图上的亚指数参数化算法。载于 Irit Dinur（编），*IEEE*第*57*届计算机科学基础年度研讨会（*FOCS 2016*），*2016*年*10*月*9*日至*11*日，美国新泽西州新不伦瑞克，凯悦酒店，第515–524页。IEEE计算机学会，2016年。[27] Fedor V. Fomin、Daniel Lokshtanov、Dániel Marx、Marcin Pilipczuk、Michal Pilipczuk 和 Saket Saurabh。通过低树宽模式覆盖，求解平面图和无顶点子式图上的亚指数参数化算法。《SIAM计算杂志》，51(6):1866–1930，2022年。

[28] Fedor V. Fomin、Daniel Lokshtanov、Fahad Panolan、Saket Saurabh 和 Meirav Zehavi。地图图的分解及其应用。载于 Christel Baier、Ioannis Chatzigiannakis、Paola Flocchini 和 Stefano Leonardi（编），第*46*届自动机、语言与程序设计国际研讨会（*ICALP 2019*），*2019*年*7*月*9*日至*12*日，希腊帕特雷，《LIPIcs》第132卷，第60:1–60:15页。Schloss Dagstuhl - Leibniz信息学中心，2019年。[29] Fedor V. Fomin、Daniel Lokshtanov、Venkatesh Raman 和 Saket Saurabh。部分覆盖问题的亚指数算法。《信息处理快报》，111(16):814–818，2011年。[30] Russell Impagliazzo、Ramamohan Paturi 和 Francis Zane。哪些问题具有强指数复杂度? 《计算机与系统科学杂志》，63(4):512–530，2001年。[31] Gwenaël Joret 和 David R. Wood。完全图子式与图小图结构定理。《组合理论杂志B辑》，103(1):61–74，2013年。[32] Ken-ichi Kawarabayashi、Yusuke Kobayashi 和 Bruce A. Reed。二次时间内求解不相交路径问题。《组合理论杂志B辑》，102(2):424–435，2012年。[33] Philip N. Klein 和 Dániel Marx。在 $O(n^{c\sqrt{k}})$时间内求解平面 k终端点割问题。载于 Artur Czumaj、Kurt Mehlhorn、Andrew M. Pitts 和 Roger Wattenhofer（编），自动机、语言与程序设计——第*39*届国际研讨会（*ICALP 2012*），英国华威，*2012*年*7*月*9*日至*13*日，会议论文集，第一部分，《计算机科学讲义》第7391卷，第569–580页。施普林格，2012年。[34] Philip N. Klein 和 Dániel Marx。平面图上子集旅行商问题的亚指数参数化算法。载于 Chandra Chekuri（编），第二十五届*ACM-SIAM*离散算法年度研讨会论文集（*SODA 2014*），美国俄勒冈州波特兰，*2014*年*1*月*5*日至*7*日，第1812–1830页。SIAM，2014年。[35] Tuukka Korhonen、Michal Pilipczuk 和 Giannos Stamoulis。近线性时间内解决子式包含与不相交路径问题。载于*IEEE*第*65*届计算机科学基础年度研讨会（*FOCS 2024*），美国伊利诺伊州芝加哥，*2024*年*10*月*27*日至*30*日，第53–61页。IEEE，2024。[36] William Lochet。求解k-不相交最短路径问题的多项式时间算法。载于 Dániel Marx（编），*2021*年*ACM-SIAM*离散算法研讨会论文集（*SODA 2021*），线上会议，*2021*年*1*月*10*日至*13*日，第169–178页。SIAM，2021年。[37] Daniel Lokshtanov、Pranabendu Misra、Michal Pilipczuk、Saket Saurabh 和 Meirav Zehavi。平面不相交路径问题的指数时间参数化算法。《SIAM计算杂志》，54(2):321–418，2025年。[38] Daniel Lokshtanov、Saket Saurabh 和 Magnus Wahlström。平面图上的奇环横截集亚指数参数化算法。载于 Deepak D'Souza、Telikepalli Kavitha 和 Jaikumar Radhakrishnan（编），*IARCS* 软件技术与理论计算机科学基础年度会议（*FSTTCS 2012*），*2012*年*12*月*15*日至*17*日，印度海得拉巴，《LIPIcs》第18卷，第424–434页。Schloss Dagstuhl - Leibniz信息学中心，2012年。[39] Dániel Marx。终端点数固定时平面多路割问题的紧下界。载于 Artur Czumaj、Kurt Mehlhorn、Andrew M. Pitts 和 Roger Wattenhofer（编），自动机、语言与程序设计——第*39*届国际研讨会（*ICALP 2012*），

英国华威，*2012*年*7*月*9*日至*13*日，会议论文集，第一部分，《计算机科学讲义》第7391卷，第677–688页。施普林格，2012年。[40] Dániel Marx、Pranabendu Misra、Daniel Neuen和Prafullkumar Tale。平面图上广义环命中问题的参数化亚指数算法框架。载于Joseph (Seffi) Naor和Niv Buchbinder编，*2022*年*ACM-SIAM*离散算法研讨会论文集（*SODA 2022*），线上会议/美国弗吉尼亚州亚历山大，*2022*年*1*月*9*日至*12*日，第2085–2127页。SIAM，2022年。[41] Dániel Marx、Marcin Pilipczuk和Michal Pilipczuk。平面图上斯坦纳树和有向子集旅行商问题的亚指数参数化算法。载于Mikkel Thorup编，第*59*届*IEEE*计算机科学基础年度研讨会（*FOCS 2018*），法国巴黎，*2018*年*10*月*7*日至*9*日，第474–484页。IEEE计算机学会，2018年。[42] Dániel Marx和Michal Pilipczuk。利用沃罗诺伊图求解平面设施选址问题的最优参数化算法。《*ACM*算法汇刊》，18(2):13:1–13:64，2022年。[43] Jesper Nederlof。在亚指数参数化时间内检测和计数平面图中的小型模式。载于Konstantin Makarychev、Yury Makarychev、Madhur Tulsiani、Gau- tam Kamath和Julia Chuzhoy编，第*52*届*ACM SIGACT*计算理论研讨会论文集（*STOC 2020*），美国伊利诺伊州芝加哥，*2020*年*6*月*22*日至*26*日，第1293–1306页。美国计算机协会，2020年。[44] Paul Erdős和Richard Rado。集合系统的交定理。伦敦数学会杂志，1(1):85–90，1960年。[45] Marcin Pilipczuk、Michal Pilipczuk、Piotr Sankowski和Erik Jan van Leeuwen。平面图和有界亏格图上斯坦纳问题的网络稀疏化。《*ACM*算法汇刊》，14(4):53:1–53:73，2018年。[46] Michal Pilipczuk、Giannos Stamoulis和Michal Wlodarczyk。平面图上的不相交最短路径问题是固定参数可处理的。*CoRR*，abs/2505.03353，2025年。

[47] Neil Robertson 和 Paul D. Seymour。图小 XIII：不相交路径问题。*J.Comb. Theory B*，63(1)：65–110，1995年。

[48] Ildikó Schlotter。保守图中的最短两条不相交路径。载于 Olaf Beyersdorff、Mamadou Moustapha Kanté、Orna Kupferman 和 Daniel Lokshtanov 编，第*41*届理论计算机科学方面国际研讨会，*STACS 2024*，*2024*年*3*月*12*日至*14*日，法国克莱蒙费朗，*LIPIcs*第289卷，第57:1–57:17页。Schloss Dagstuhl - Leibniz信息学中心，2024年。

[49] Ondrej Suchý。将平面斯坦纳树的核扩展到斯坦纳顶点数。*Algorithmica*，79(1)：189–210，2017年。

定理 42 的证明42..

定理 *4 2* 的证明*42...*我们将寻找序列 $(P_1, \ldots, P_q)$ 表述为一个最小成本最大流问题；割 C_j 将从对偶线性规划的最优解中读出。

为此，构造一个带顶点权重的流网络 G'，如下所示。对于每个 $v \in V(G) - \{s, t\}$，引入 v 的两个副本：一个 v_0，成本为 0、容量为 1；另一个 v_1，成本为 1，

容量为 $+\infty$。顶点 s 成为容量为 q、成本为 0 的源点，顶点 t 成为容量为 q、成本为 0 的汇点。G' 的边按自然方式定义：G 的每条边在 G' 中至多对应四条边，连接相应顶点的每一对副本。

我们要求在 G' 中求一个值为 q 的最小费用流。由于每个容量要么是整数，要么是 $+\infty$，因此我们可以找到一个最小费用流，将其分解为 q 条流路径 $P'_1, \ldots, P'_q$，每条路径承载单位流。每条流路径 P'_i 都投影为从 s 到 t 的 G 中的一条游走；我们将这条游走缩短为一条路径，并将该路径记为 P_i。

为了定义链 $(C_1, \ldots, C_p)$，我们将流问题表述为一个线性规划，并考察其对偶。这是原问题：

$$
\begin{aligned}
\min \quad & \sum_{v \in V(G) - \{s,t\}} \sum_{a \in N_{G'}(v_1)} f(v_1, a) & \\
\text{s.t.} \quad & \sum_{b \in N_{G'}(a)} f(a,b) - f(b,a) = 0 & \forall a \in V(G') - \{s,t\} \\
& \sum_{a \in N_{G'}(s)} f(s,a) - f(a,s) = q & \\
& \sum_{a \in N_{G'}(t)} f(t,a) - f(a,t) = -q & \\
& \sum_{a \in N_{G'}(v_0)} f(v_0, a) \leqslant 1 & \forall v \in V(G) - \{s,t\} \\
& f(a,b) \geqslant 0 & \forall ab \in E(G').
\end{aligned}
$$

这是对偶问题：

$$
\begin{aligned}
\max \quad & q(y_t - y_s) - \sum_{v \in V(G) - \{s,t\}} z_v & \\
\text{s.t.} \quad & y_{v_0} \leqslant y_a + z_v & \forall v \in V(G) - \{s,t\}, a \in N_{G'}(v_0) \\
& y_{v_1} \leqslant y_a + 1 & \forall v \in V(G) - \{s,t\}, a \in N_{G'}(v_1) \\
& y_s \leqslant y_a & \forall a \in N_{G'}(s) \\
& y_t \leqslant y_a & \forall a \in N_{G'}(t) \\
& z_v \geqslant 0 & \forall v \in V(G) - \{s,t\}
\end{aligned}
$$

对偶问题可以解释为一个距离问题：我们最大化从 s 到 t、权重为 q 的距离；每个顶点 $v \in V(G) - \{s,t\}$ 都会产生 $\min(1, z_v)$ 的行进成本，并且每个 z_v 都会受到惩罚。由于对偶问题在对所有变量 y_a 加上同一偏移量后保持不变，我们可以添加约束 $y_s = 0$。

由于原问题是一个流问题，可以用全幺模矩阵表示，因此我们可以找到一个对偶问题的整数最优解。注意，这个最优解满足 $z_v \in \{0, 1\}$ 对于每个 $v \in V(G) - \{s,t\}$，因为将变量 z_v 的值设得高于 1 并无意义。

我们设定 $p = y_t$，并且对于 $1 \leqslant j \leqslant p$，取

$$C_j = \{v \in V(G) - \{s,t\} \mid y_{v_0} = j \text{ and } z_v = 1\}.$$

我们首先证明 $(C_1, \ldots, C_p)$ 是 (s,t)-分隔集的链。记 $s_0 := s$ 和 $t_0 := t$。令 P 为 G 中一条从某个 u 到某个 w 的路径；并设 $j_u := y_{u_0}$, $j_w := y_{w_0}$ 满足

$0 \leqslant j_u < j_w \leqslant p$。令 P' 为 G' 中从 u_0 到 w_0 的对应路径；当且仅当 P 经过 v 时，该路径经过 v_0。令 $j_u < j \leqslant j_w$。由于对于每个 $ab \in E(G')$，都有 $|y_a - y_b| \leqslant 1$，因此 P' 上存在一个顶点 v_0 满足 $y_{v_0} = j$；令 v_0 为第一个这样的顶点。由于 $j > j_u \geqslant 0$,$v_0 \neq u_0$，且 v_0 在 P' 上的前驱 b 满足 $y_b = j - 1$。这意味着 $v \in V(G) - \{s,t\}$ 且 $z_v = 1$。因此，$v \in C_j$，也就是说，C_j 与 P 相交。这证明，对于每个 $1 \leqslant j \leqslant p$，$C_j$ 都是一个 (s,t)-分隔集，并且每个 $1 \leqslant j < j' \leqslant p$, C_j 都与从 s 到 $C_{j'}$ 的每条路径相交，且与从 C_j 到 t 的每条路径相交。

根据互补松弛条件，对于每个 $1 \leqslant i \leqslant q$，$P_i'$ 的每条边相对于对偶中的对应不等式都是紧的。这意味着，对于 $v \in V(G) - \{s,t\}$，P_i' 是从 s 到 t、在 v_0 上赋予权重 z_v 且在 v_1 上赋予权重 1 时的一条最短路径。因此，P_i' 不会同时经过 v_0 和 v_1（即，P_i 是 P_i' 在没有任何捷径时的投影），并且对于每个 $1 \leqslant j \leqslant p$，$|V(P_i) \cap C_j| \leqslant 1$。由于 C_j 是一个 $(s,\ t)$-分隔集，我们实际上有 $|V(P_i) \cap C_j| = 1$。这证明了 $(p,\ q)$-结构的第一个性质。

现在考虑某个 $i \neq i'$ 的 $v \in V(P_i) \cap V(P_{i'})$。由于 v_0 的容量为 1，P_i' 或 $P_{i'}'$ 必有一个经过 v_1；不失一般性，假设 P_i' 经过 v_1。根据互补松弛条件，对于 v_1 在 P_i' 上的前驱 a，相应的不等式 $y_{v_1} \leqslant y_a + 1$ 是紧的。由于 P_i' 是从 s 到 t 的一条最短路径，经过 v_0 而非 v_1 并不会缩短距离，因此 $z_v = 1$。此外，$0 < y_{v_0} = y_{v_1} \leqslant p$。这意味着 $v \in C_{y_{v_0}}$，从而证明了 $(p,\ q)$-结构的第二个性质。

再次根据互补松弛条件，如果 $z_v = 1$ 对于某个 $v \in V(G) - \{s,t\}$，那么表示 v_0 容量的不等式就是紧的。因此，某条流路径 P_i' 经过 v_0。由于 P_i 是 P_i' 在不经过任何捷径的情况下的投影，P_i 经过 v。这证明了 (p,q)-结构的第三个性质，并完成了定理 4 2 的证明42。。。 □