两个有界度问题的结构参数化再探*

Michael Lampis 巴黎多菲纳大学、PSL大学、法国国家科学研究中心 UMR7243、LAMSADE，法国巴黎
ManolisVasilakis 巴黎多菲纳大学、PSL大学、法国国家科学研究中心 UMR7243、LAMSADE，法国巴黎

arXiv:2304.14724v2 [cs.DS] 4 May 2024

摘要

我们重新研究两个广受关注的问题：有界度顶点删除和缺陷着色。给定一个图 G 和一个目标度 Δ，我们要么编辑图，要么对图进行划分，使其最大度不超过 Δ。众所周知，对于树宽等最著名的结构参数，这两个问题在参数化复杂性方面均难以处理。

我们重新考察以树宽为参数的情形，以及若干相关参数，并更细致地刻画这两个问题的复杂性。具体而言：

- 这两个问题都存在直接的动态规划算法，其表大小分别为 $(\Delta+2)^{\mathrm{tw}}$ 和 $(\chi_{\mathrm{d}}(\Delta+1))^{\mathrm{tw}}$，其中 tw 表示输入图的树宽，$\chi_{\mathrm{d}}$ 表示可用颜色的数量。我们证明，在强指数时间假设成立的情况下，对于 Δ，χ_{d} 的任意非平凡固定取值，这两种算法在本质上都是最优的；即使将树宽替换为路径宽度，结论仍然成立。此外，我们还得到了一种用于缺陷着色的算法，其复杂度相对于表大小为准线性，从而确定了这两个问题在树宽和路径宽度参数下的复杂性。
- 鉴于标准动态规划算法对于树宽和路径宽度是最优的，我们随后转而考虑限制更严格的参数树深度。在此情况下，已知的下界表明，在指数时间假设下，有界顶点度删除和缺陷着色分别无法在时间 $n^{o(\sqrt[4]{\text{树深度}})}$ 和 $n^{o(\sqrt{\mathrm{td}})}$ 内求解，因此仍有希望找到一种相较于树宽情形在定性上更快的算法。我们通过证明，在指数时间假设下，这两个问题均无法在时间 $n^{o(\mathrm{td})}$ 内求解，从而填补了这一空白；为此采用了递归低树深度构造，该构造本身也可能具有独立意义。
- 最后，我们考虑一种已知足够严格、能够使这两个问题都成为固定参数可处理（FPT）的结构参数：顶点覆盖。对于这两个问题，在此设定下已知的最佳算法，其运行时间对 vc 的依赖为超指数形式 $\mathrm{vc}^{\mathcal{O}(\mathrm{vc})}$。我们证明这是最优的，因为若存在对 vc 的依赖为 $\mathrm{vc}^{o(\mathrm{vc})}$ 的算法，就会违背指数时间假设。我们的证明依赖于对 Bonamy 等人 [ToCT 2019] 提出的 d-检测族技术的一种新应用。

尽管我们的结果大多是负面的，但它们清晰地描绘了这两个问题在参数化复杂性领域中的复杂性，因为在所有情况下，我们都给出了基本匹配的上界和下界。

2012年 ACM 学科分类计算理论 → 参数化复杂性与精确算法

关键词与短语指数时间假设、参数化复杂性、强指数时间假设

资助本研究部分由法国国家研究署项目 ANR-21-CE48-0022（S-EX-AP-PE-AL）和 ANR-18-CE40-0025-01（ASSK）资助。

* 本研究的扩展摘要曾在第31届欧洲算法年会（ESA 2023 [36]）上发表。

1 引言

参数化复杂性，尤其是对树宽等结构参数的研究，是处理图上 NP 难问题最成熟的方法之一。树宽无疑是这一领域最成功的案例之一：大量困难问题在以树宽为参数时都变得固定参数可解。自然，这一成功促使人们努力探寻树宽算法能力的边界，试图了解哪些问题无法通过基于树宽的技术解决。

树宽工具箱何时会失效？许多问题似乎都存在一种共同情形：它们在以树宽为参数时是 W[1]-难的[1]，但确实存在一种自然的动态规划算法，只是该动态规划必须为树分解中每个袋内的顶点存储一个任意大的整数——例如与该顶点的度相关的数值。本文旨在研究这类情形，并提出一个自然的问题：我们能否做得比“显而易见”的动态规划更好，通过获得运行时间更优的算法来实现这一点，即使这意味着要考虑比树宽限制更严格的参数？

鉴于上述情况，我们聚焦于两个问题，它们可以说是最能代表这一情形的典型问题：有界度顶点删除和缺陷着色。这两个问题的输入都是一个图 G 和一个目标度 Δ 。对于有界度顶点删除，我们要删除最少数量的顶点，使剩余图的度至多为 Δ；对于缺陷着色，我们要将 G 划分为最少数量的颜色类，使每个颜色类诱导出的图的度至多为 Δ。这两个问题都已得到广泛研究，因为它们分别推广了经典问题（顶点覆盖和着色）；我们将在下文回顾部分已有研究。不过，就本文的目的而言，这两个问题最相关的特点是：**(i)** 两者都存在复杂度形式为 $n^{\mathcal{O}(\mathrm{tw})}$ 的动态规划算法；**(ii)** 两者在以树宽为参数时都是 W[1]-难的；事实上，对于缺陷着色，已知在假设指数时间假设成立的情况下，它无法在$n^{o(\mathrm{tw})}$ [5]时间内求解。

由于这些 $n^{\mathcal{O}(\mathrm{tw})}$ 算法可由标准动态规划技术得到，一个自然的问题是：我们能否做得更好？是否存在更优的算法？实际而言，我们可以期待两种可能之一：要么存在一种仍以树宽为参数处理该问题的算法，鉴于上述下界，它只尝试对运行时间进行细粒度改进；要么存在一种速度显著更快、但使用限制更严格的参数的算法。本文的结果为这两个问题都提供了有力的否定证据：若以树宽（甚至路径宽度）为参数，那么即使对其他相关参数（Δ 和颜色数 χ_{d}）的所有固定取值，标准动态规划的运行时间在强指数时间假设下仍是最优的；而若采用树深度和顶点覆盖等限制更严格的参数，我们得到的下界结果（基于指数时间假设）表明，即使在这些限制严格得多的情形下，最佳算法本质上仍是运行某种形式的标准树宽动态规划。因此，我们的结果全面描绘了这两个问题的结构参数化复杂性，并表明在许多受限情形下，标准动态规划都是最优的。

[1] 我们假设读者熟悉标准教材 [14] 中介绍的参数化复杂性理论基础。

我们的贡献详述。按照标准技术，这两个问题分别可以通过动态规划算法求解，其表大小分别为（$\Delta+2$）^{tw} 和（χ_{d}（$\Delta+1$））^{tw}。我们的第一个结果是一组归约，证明在强指数时间假设成立的前提下，即使针对路径宽度，也不存在能够优于这些动态规划的算法。更准确地说，我们证明，对于任意 $\varepsilon>0$ 以及 Δ，χ_{d} 的固定值组合，不存在分别能在（$\Delta+2-\varepsilon$）$^{\mathrm{pw}}n^{\mathcal{O}(1)}$ 和（χ_{d}（$\Delta+1$）$-\varepsilon$）$^{\mathrm{pw}}n^{\mathcal{O}(1)}$ 时间内求解有界度顶点删除和缺陷着色的算法（但 $\Delta=0$ 和 $\chi_{\mathrm{d}}=2$这一组合除外，因为它会使缺陷着色可在多项式时间内求解）。我们的归约遵循 Lokshtanov、Marx 和 Saurabh 开创的总体策略 [38]，并将他们关于顶点覆盖和着色的结果（已涵盖 $\Delta=0$的情形）推广开来。主要难点在于覆盖次级参数的所有取值；出于技术原因，我们不得不分别为两个问题设计不同版本的归约，以覆盖 $\Delta=1$ 的情形。此外，我们指出，尽管已有针对缺陷着色、复杂度为（$\chi_{\mathrm{d}}\Delta$）$^{\mathcal{O}(\mathrm{tw})}n^{\mathcal{O}(1)}$ 的算法 [5],，但当时尚不清楚是否存在复杂度为（χ_{d}（$\Delta+1$））$^{\mathrm{tw}}n^{\mathcal{O}(1)}$ 的算法（即对表大小具有准线性依赖）。为完整起见，我们利用 Cygan 和 Pilipczuk 提出的快速傅里叶变换技术 [17], 给出一个运行时间达到这一复杂度的算法，从而解决了这个问题。再结合 van Rooij 提出的、运行时间为（$\Delta+2$）$^{\mathrm{tw}}n^{\mathcal{O}(1)}$ 的有界度顶点删除算法 [51],，我们便为这两个问题在树宽和路径宽度参数下分别获得了完全匹配的上界和下界。

鉴于上述结果相当明确地表明，对于树宽和路径宽度参数，标准动态规划是最优算法，我们接下来转向一个限制更严格的情形：树深度。我们回顾一下，树深度为 k 的图是路径宽度为 k 的图的真子类，因此有理由期待能针对这一参数获得更优算法。已知下界与标准算法的复杂度并不匹配，这一事实也进一步支持了这种期待。更具体地说，Ganian、Klute 和 Ordyniak [27] 给出的、针对以树深度为参数的有界度顶点删除问题的 W[1]-困难性归约，其规模膨胀为四次方，因此只能推出不存在 $n^{o(\sqrt[4]{\text{树深度}})}$ 算法；而 [5] 中针对缺陷着色的归约，其规模膨胀为二次方，因此只能推出不存在 $n^{o(\sqrt{\mathrm{td}})}$ 算法（两种情形均基于指数时间假设）。我们的贡献在于证明，这两种归约都可以替换为参数规模呈线性的更高效归约；由此我们确立，这两个问题都无法在时间 $n^{o(\mathrm{td})}$ 内求解，这意味着即使在这一限制情形下，基于树宽的算法在定性意义上仍是最优的。我们的归约有一个有趣之处：我们没有像这类归约中常见的做法那样，使用到低树深度图的调节集，而是采用递归构造，充分利用该参数的全部能力；这种方法或许也有助于进一步收紧针对树深度参数的其他下界。

最后，我们转向一个更特殊的情形：以顶点覆盖为参数研究这两个问题。对于这一参数，这两个问题都是固定参数可处理的。由于顶点覆盖作为参数的限制性很强，我们原本希望终于能够找到一种比基于树宽的动态规划更巧妙的算法。令人有些失望的是，这两个问题已知的固定参数可处理算法的复杂度均为 $\mathrm{vc}^{\mathcal{O}(\mathrm{vc})}n^{\mathcal{O}(1)}$ [5]，而参数依赖呈超指数级，是因为这两种算法都采用了简单的双赢论证；其中一种情形下，这种论证只是执行标准的树宽动态规划。我们证明，这种做法是合理的，因为在指数时间假设下，这两个问题都无法在 $\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$ 时间内求解。这意味着，在某些情况下，盲目执行基于树宽的动态规划的算法仍然是（定性意义上）最优的。我们的结果通过应用 d-检测族这一技术得出，该技术由 Bonamy、Kowalik、Pilipczuk、Socala 和 Wrochna [10] 提出。我们的结果表明，以

顶点覆盖为参数是一个领域，在参数化复杂性研究中，这种前景广阔但目前尚未得到充分利用的技术或许能有更多应用。

相关工作。 有界度顶点删除和缺陷着色都是研究充分、文献丰富的问题。有界度顶点删除可应用于众多领域，从计算生物学 [21] 到投票理论中的一些相关问题 [7, 9], ； 其对偶问题称为 s-团检测，在社会网络分析 [4, 40, 42]中有诸多应用。已有多种近似算法 [24, 25, 45]。该问题也在参数化复杂性的框架下得到了广泛研究。当 Δ 的取值无界且参数为 k （最优值） [21], 时，该问题是 W[2]-困难的；而对于任意固定的 $\Delta \geq 0$，以 k [21, 53]为参数时，它存在一个线性规模的核；在后一种设定下，也已有许多固定参数可处理算法 [42, 43, 52]。针对有界度顶点删除，已有固定参数可处理近似算法 [34]和 [39]。至于缺陷着色（文献中也称为非正常着色），它约于 40 年前被提出 [1, 13]。这一问题的主要动机来自电信领域：颜色对应可用频率，目标是将频率分配给通信节点；节点之间少量干扰可以容忍，而这由参数 Δ 建模。关于该问题已有大量研究（参见 [2, 3, 5, 6, 12, 30]及其中的参考文献），尤其是在单位圆盘图和各种网格类图上。

与我们最相关的这两个问题的既有研究，侧重于它们关于树宽等结构参数的参数化复杂性。在这一框架下，有界度顶点删除是最早被发现关于树宽参数为 W[1]-难的问题之一 [8], ，尽管当参数为 $\mathsf{tw} + \Delta$ 或 $\mathsf{tw} + k$ 时，该问题确实属于固定参数可处理。Ganian、Klute 和 Ordyniak [27], 最近改进了这一难度结果，他们证明，有界度顶点删除关于树深度和反馈顶点集参数为 W[1]-难。文献 [5] 证明，缺陷着色关于树深度参数为 W[1]-难（因而关于路径宽度和树宽也如此）。然而， [5] 给出了一个关于路径宽度的难度归约，其规模关于参数呈线性，因此这意味着在指数时间假设下，缺陷着色存在一个 $n^{o(\mathrm{pw})}$ 下界；但其关于树深度的难度归约规模呈二次，因此仅能推出一个 $n^{o(\sqrt{\mathrm{td}})}$ 下界。类似地， [27] 给出的有界度顶点删除关于树深度参数的归约规模呈四次，因为它经过一个中间问题（子集和问题的一个变体），因此只能推出一个 $n^{o(\sqrt[4]{\mathrm{td}})}$ 下界。已知缺陷着色关于顶点覆盖参数属于固定参数可处理，这可由一个简单的两难论证得出：在一种情况下（若 $\Delta > \mathsf{vc}$），应用基于树宽的动态规划，因为此时图总是可 2-着色；否则，标准动态规划算法可在固定参数时间内运行。有界度顶点删除也同样如此（若 $\Delta \leq \mathsf{vc}$，可使用前述针对参数 $\mathsf{tw}+\Delta$ 的固定参数可处理算法；否则，假设 $k < \mathsf{vc}$，因为不然问题就是平凡的；再沿用 [8] 到向量支配集的归约，并注意到度大于 Δ 的顶点至多有 vc 个，因此根据 [46], 可在 $\mathsf{vc}^{\mathcal{O}(\mathrm{vc})}n^{\mathcal{O}(1)}$ 时间内判定向量支配集）。因此，对于这两个问题，关于该参数的最佳算法复杂度均为 $\mathsf{vc}^{\mathcal{O}(\mathrm{vc})}n^{\mathcal{O}(1)}$。

对结构参数化（例如以树宽为参数）的复杂性进行精细分析，是一个活跃的研究领域。Lokshtanov、Marx 和 Saurabh [38]首创了利用强指数时间假设建立紧运行时间下界的技术。此后，许多问题在以树宽及相关参数（如路径宽度和团宽 [15, 16, 19, 20, 23, 26, 28, 29, 31, 44, 50]）进行参数化时，其紧上界和下界已得到确立。我们在此呈现的结果有一个难点

表 1 本研究确立的下界。第一行的结果基于强指数时间假设，其余结果则基于指数时间假设。

参数	有界度顶点删除	缺陷着色
路径宽度 $+\ \Delta$	$\mathcal{O}^\star((\Delta+2-\varepsilon)^{\mathrm{pw}})$	$\mathcal{O}^\star((\chi_{\mathrm{d}}\cdot(\Delta+1)\ -\varepsilon)^{\mathrm{pw}})$
树深度	$n^{o(\mathrm{td})}$	$n^{o(\mathrm{td})}$
顶点覆盖	$\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$	$\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$

在于需要给出一族归约：对 Δ 和 χ_{d}的每个固定取值各给出一个。另有一些问题已知存在紧下界族，例如k-着色，其关于树宽 [38] 的正确依赖关系为 k^{tw}，关于团宽 [35]的正确依赖关系为 $(2^k-2)^{\mathrm{cw}}$，对所有 $k\geq 3$均成立；距离 r-支配集，其正确依赖关系为 $(2r+1)^{\mathrm{tw}}$ [11] 和 $(3r+1)^{\mathrm{cw}}$ [32]，对所有 $r\geq 1$均成立；以及距离 d-独立集，其正确依赖关系为 d^{tw} [33]。在所有这些情形中，最优算法都是“自然的”动态规划，而我们关于有界度顶点删除和缺陷着色的结果也符合这一模式。

尽管上述先前工作可能会让人觉得，我们基于强指数时间假设的下界并不令人意外，但必须强调的是，不能想当然地认为朴素动态规划对于我们的问题就是最优的。特别是，有界度顶点删除属于一般的（σ,ρ）-支配问题范畴；这类问题最近在 [22] 中得到了研究（关于（σ,ρ）-支配的定义，请参阅该文）。该工作的主要成果之一是证明，在某些情况下确实可以对基本动态规划进行显著改进，尤其是当 σ,ρ 中有一个是余有限集时。由于有界度顶点删除对应于 $\sigma=\{0,\ldots,\Delta\}$，且 $\rho=\mathbb{N}$（即 ρ 是余有限的），我们的结果恰好属于 [22], 留下的尚未探索的领域，在这一领域中仍有可能找到更高效的算法（事实上， [22] 确实针对 σ,ρ 的某些取值发现了这样的算法）。

2 预备知识

本文通篇使用标准图论记号 [18]，并假设读者熟悉参数化复杂性的基本概念 [14]。除非另有明确说明，本文考虑的所有图都是无向且不含自环的。对于图 $G=(V,E)$ 和两个整数 $\chi_{\mathrm{d}}\geq 1,\Delta\geq 0$，若可以将 G 划分为 χ_{d} 个集合，使每个集合所诱导的图的最大度至多为 Δ，则称 G 存在一个 $(\chi_{\mathrm{d}},\Delta)$-着色。在这种情况下，缺陷着色问题是指：给定 $G,\chi_{\mathrm{d}},\Delta$，判断 G 是否存在一个 $(\chi_{\mathrm{d}},\Delta)$-着色。令 $\mathbb{N}$ 表示非负整数集合。对于 $x,y\in\mathbb{Z}$，令 $[x,y]=\{z\in\mathbb{Z}\mid x\leq z\leq y\}$，而 $[x]=[1,x]$。我们使用标准的 $\mathrm{O}^\star$ 记号来省略多项式因子。对于路径宽度界，我们采用混合搜索策略 [48], 的概念，其中通过在一条边的两个端点上各放置一名搜索员，或沿该边滑动一名搜索员，即可清除该边。我们依赖指数时间假设（ETH）的一个较弱形式，该形式指出：对于含有 n 个变量和 m个子句的实例，3-SAT 无法在 $2^{o(n+m)}$ 时间内求解。

在 k-多彩团问题中，给定一个图 $G=(V,E)$，以及将 V划分为 k 个独立集 $V_1,\ldots,V_k$，每个独立集的大小均为 n，我们需要判定 G 是否包含一个 k-团。众所周知，除非指数时间假设为假 [14]，否则该问题不存在任何 $f(k)n^{o(k)}$算法，其中 f 是任意可计算函数。

在 q-CSP-B 中，给定一个包含 n 个变量和 m 个约束的约束满足问题（CSP）实例。变量的取值来自大小为 B 的集合 Y，即 $|Y|=B$。每个

约束涉及至多 q 个变量，并以这些变量的满足赋值列表形式给出，其中一个满足赋值是一个 q 元组，其值来自集合 Y，分别赋给这 q 个变量。Lampis [35] 指出，以下结果是强指数时间假设的自然推论，并且过去曾用于证明各种困难性结果 [19, 20, 29]。

▶ **定理 1** ([35])。 对于任意 $B \geq 2$，如果强指数时间假设成立，那么对所有 $\varepsilon > 0$，都存在一个 q，使得 n-变量 q-约束满足问题-B 无法在时间 $O^\star((B-\varepsilon)^n)$ 内求解。

3 树宽与最大度

本节给出关于求解有界度顶点删除和缺陷着色问题的复杂度紧下界，这两个问题均以输入图的树宽加目标度为参数。对于有界度顶点删除，该下界与 [22], 的算法相匹配；而对于缺陷着色，我们将在第3.2.3节中给出一个运行时间相匹配的算法。

两个归约的性质相似：我们从一个 ϕ 的 q-约束满足问题-B实例出发，并构造一个等价实例，其图的路径宽度为 $\mathrm{pw} = n + \mathcal{O}(1)$，其中 n 表示 ϕ中的变量数。然而，一个有趣的观察是，对于这两个问题，我们都必须区分 $\Delta = 1$ 和 $\Delta \geq 2$的情形；后一种情形会使整个构造复杂得多。

3.1 有界度顶点删除

接下来，我们将介绍一个从 q-CSP-B 到有界度顶点删除的归约，其中 $\Delta \geq 1$ 为任意固定值，且 $\Delta = B - 2$。在这种情况下，如果存在一个用于有界度顶点删除的 $\mathcal{O}^\star((\Delta+2-\varepsilon)^{\mathrm{pw}})$ 算法，其中 $\varepsilon > 0$，那么对于任意常数 q，就存在一个用于 q-CSP-B 的 $\mathcal{O}^\star((B-\varepsilon)^n)$ 算法；根据定理 1，这将导致强指数时间假设失效。

我们的归约基于由 块构件 构造“长路径”，这些块构件以类似路径的方式串联连接。每条这样的“路径”对应给定约束满足问题中的一个变量，而构造中的每一列都与该变量的一个约束相关联。直观地说，我们的目标是将 B^n 种可能的变量赋值嵌入到某个最优动态规划的 $(\Delta+2)^{\mathrm{tw}}$ 个状态中，使其能够在我们构造的实例上求解该问题。

下面，我们介绍归约中使用的一系列构件。前文提到的块构件允许解在 $\Delta + 2$ 个合理选项中进行选择，是其中的关键。请注意，这些构件会因 Δ 是否等于 1 而大不相同。我们以类似路径的方式连接这些构件，确保整个构造中的选择始终一致；同时，以一种能够验证所作选择是否代表满足赋值的方式，连接构造出的网格中不同“列”的约束构件，且不会显著增加图的路径宽度。

▶**定理 2**。 对于任意常数 $\varepsilon > 0$，除非强指数时间假设不成立，否则不存在一个 $O^\star((3-\varepsilon)^{\mathrm{pw}})$ 算法，能够判定 $\Delta = 1$ 上的有界度顶点删除 问题；其中 路径宽度 表示输入图的路径宽度。

证明。 固定一个正数 $\varepsilon > 0$，我们要证明该定理对其成立。我们将把q-CSP-3，其中 q 是一个仅取决于 ε的常数，规约为 有界度顶点删除 问题，其中 $\Delta = 1$，并确保如果所得的有界度顶点删除 实例能在 $\mathcal{O}^\star((3-\varepsilon)^{\mathrm{pw}})$时间内求解，那么我们就能得到一个

用于 q-CSP-3 的算法，这将根据定理 1 与强指数时间假设相矛盾。为此，设 ϕ 是一个 q-CSP-3 实例，包含 n 个变量 $X = \{x_i \mid i \in [n]\}$，取值范围为集合 $Y = [3]$，并有 m 个约束 $C = \{c_j \mid j \in [m]\}$。对于每个约束，我们给定一个至多包含 q 个变量的集合，这些变量参与该约束；同时还给出这些变量的满足赋值列表，其大小记为 s：$C \to [3^q]$，即 $s(c_j) \leq 3^q = \mathcal{O}(1)$ 表示约束 c_j 的满足赋值数量。我们将在多项式时间内构造一个等价实例 $\mathcal{I} = (G, k)$，它是针对 $\Delta = 1$ 的有界度顶点删除实例，且 $\mathrm{pw}(G) \leq n + \mathcal{O}(1)$。

块与变量构件。 对于每个变量 x_i 和每个约束 c_j，构造一条包含 3 个顶点的路径 $p^1_{i,j}$, $p^2_{i,j}$ 和 $p^3_{i,j}$，它们组成块构件 $\hat{}B_{i,j}$。直观地说，我们将把删除 $p^y_{i,j}$ 映射为一种赋值，其中 x_i 取值为 y。接下来，对于 $j \in [m-1]$，我们在 $p^3_{i,j}$ 和 $p^1_{i,j+1}$之间添加一条边，从而得到长度为 $3m$的 n 条路径 $P_1, \ldots, P_n$，称为变量构件。

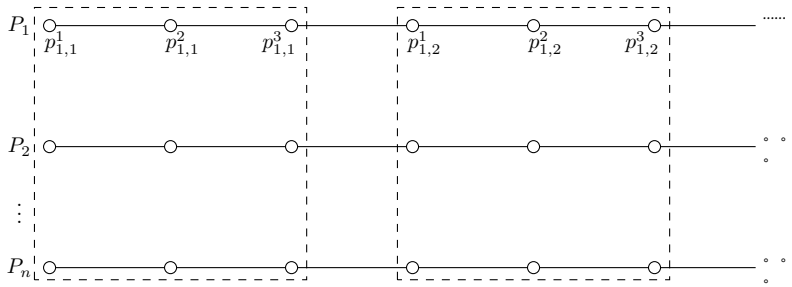


图1 块构件序列构成变量构件。

约束构件。 该构件根据图其余部分中的选择来确定约束是否满足。对于约束 c_j，按如下方式构造约束ˆ构件 C_j：

- 构造一个包含 $s(c_j)$ 个顶点的团 v^j_1，……，$v^j_{s(c_j)}$，并在这些顶点与 c_j 的满足赋值之间固定一个任意的一一映射。对每个顶点 v^j_ℓ，附加一个叶节点 l^j_ℓ。若变量 x_i 参与约束 c_j，且 v^j_ℓ 对应于一种使x_i 取值为 $y \in Y$的赋值，则在 v^j_ℓ 和 p^y_i 之间添加一条边，j。

令图 G_0 为包含所有变量构件 P_i 以及所有约束构件ˆ C_j（对应于 $i \in [n]$ 和 $j \in [m]$）的图。为构造图 G，引入 $\kappa = 2n + 1$ 个 G_0 的副本G_1，……，G_κ，并按如下方式依次将它们连接起来：对于 $i \in [n]$ 和 $j \in[\kappa-1]$，在 p^3_i,$_m(G_j)$ 与 p^1_i,$1(G_{j+1})$ 之间添加一条边，其中 p^y_i ,$_j(G_z)$ 表示ˆ图 G_z 的顶点 p^y_i ,$_j$。我们将 B_i,$_j$ 称为块构件，将 P_i 称为变量构件，并将 G_z 的约束构件ˆ C_j 分别称为ˆ$B^{G_z}_i$,$_j$，$P^{G_z}_i$ 和ˆ$C^{G_z}_j$。令 P^i 表示由$P^{G_1}_i$,……,$P^{G_\kappa}_i$形成的路径。设 $k = \kappa \cdot k'$，其中 $k' = \sum_{j=1}^{m}(s(c_j) - 1 + n) = m \cdot n + \sum_{j=1}^{m}(s(c_j) - 1)$，并令 $\mathrm{I} = (G, k)$ 表示 $\Delta = 1$ 对应的有界度顶点删除实例。

▶ **引理3。** 如果 ϕ 可满足，则存在 $S \subseteq V(G)$ 使得 $G - S$ 的最大度至多为 1 且 $|S| \leq k$。

证明。 令 f: $X \to Y$表示一个满足所有约束 $c_1, \ldots, c_m$的赋值。在这种情况下，令 p^y_i ,$_j(G_z)$ $\in S$，其中 $j \in [m]$ 和 $z \in [\kappa]$，若 $f(x_i) = y$。换言之，从

每条路径 $\mathcal{P}^i$中，根据 $f(x_i)$ 的取值，将每个块构件 $\hat{}B_{i,j}$中的第一个、第二个或第三个顶点纳入 S。此外，由于 f 是一个满足赋值，对于每个约束 c_j，都存在一个满足赋值，它是将 f 限制到 c_j 中涉及的至多 q 个变量后得到的，并且对应于 $\hat{}C_j^{G_z}$的一个顶点 $v_\ell^j(G_z)$。若 $\ell' \neq \ell$，令 $v_{\ell'}^j(G_z) \in S$。成立的是 $|S| = \kappa \cdot (mn + \sum_{i=1}^{m}(s(c_j) - 1)) = k$。

接下来仍需证明 $G - S$ 的最大度至多为 1。考虑约束构件 $\hat{}C_j^{G_z}$。任意叶节点 l_i^j 在 $G - S$中的度均为 0 或 1。令 $v_\ell^j(G_z) \notin S$ 表示约束构件的团中不属于 S的顶点。由于

- $l_\ell^j(G_z) \notin S$、$v_{\ell'}^j(G_z) \in S$，对于 $\ell' \neq \ell$，$v_\ell^j(G_z)$ 与 $p_i^{y\ 相连,\ j}(G_z)$，其中仅当 x_i参与 c_j
- 且 $v_\ell^j(G_z)$对应于一个赋值 g （其中变量 x_i 的值为 y）时， $y \in Y$。然而，由于该赋
- 值是 f的一个限制，因此可得 $g(x_i) = f(x_i) = y$，所以 $p_i^{y\ ,\ j}(G_z) \in S$。

因此，在$G - S$中，约束构件的顶点与块构件的顶点之间没有边。注意，在$\mathcal{P}^i$中，每连续三个顶点中都有一个属于 S。因此，如果$p_i^{y\ ,j}(G_z) \notin S$，那么它的邻居中至多有一个不属于 S，故其在 $G - S$中的度至多为1。◀

▶ **引理 4. </style}如果存在(G) 使得 $G - S$ 的最大度至多为1 且$|S| \leq k$，那么 ϕ 是可满足的。**

证明。 令 $S \subseteq V(G)$满足 $G - S$的最大度至多为1且 $|S| \leq k$。首先我们将证明， S 中包含来自每个块构件$\hat{}B_{i,j}$的单个顶点$p_i^{y\ ,j}$，以及顶点v_l^j，其中 $l \in [s(c_j)] \setminus \{\ell\}$，对每个约束构件$\hat{}\ C_j$，均有某个 $\ell \in [s(c_j)]$。

$\hat{}$ 注意， S 包含来自每个块构件$B_{i,\ j}$的至少 1 个顶点；否则，顶点$p_{i,\ j}^2$ 在 $G - S$中的度至少为 2。此外， S包含来自每个约束构件$\hat{}\ C_j$的至少 $s(c_j)$个 -1 顶点，因为每个此类构件都有一个大小为 $s(c_j)$的团，团中的每个顶点都连接着一个叶节点。因此，对所有 $z \in [\kappa]$，均有 $|S \cap V(G_z)| \geq k' = m \cdot n + \sum_{j=1}^{m}(s(c_j) - 1)$。由于 $|S| \leq \kappa \cdot k'$，可得 $|S \cap V(G_z)| = k'$。因此， S 在每个块构件中恰好包含一个顶点，并在每个约束构件$\hat{}\ C_j$中恰好包含 $s(c_j) - 1$ 个顶点，而这些顶点只能是团中的顶点。

注意，由此对所有 $z \in [\kappa]$，均有 $|S \cap V(G_z)| = k'$。如果变量构件 $P_i^{G_z}$中存在两个连续的块构件$\hat{}B_{i\ ,\ j}^{G_z}$，$\hat{}B_{i\ ,\ j+1}^{G_z}$，使得对于 $y \neq y'$，$p_{i\ ,\ j}^{y}$，$p_{i\ ,\ j+1}^{y'} \in S$成立，我们就称其发生了不一致。此外，如果变量构件 $P_i^{G_z}$（对所有 $i \in [n]$）中均未发生不一致，我们就称 G_z 是一致的。

▷ 命题5. 存在 $\pi \in [\kappa]$，使得 G_π 是一致的。

证明。 注意， S包含 $\kappa \cdot m$ 个来自每条路径 $\mathcal{P}^i$的顶点，因为后者由这么多个块构件组成。我们将证明，每条路径 $\mathcal{P}^i$ 至多会导致2处不一致。在这种情况下，由于有 n 条这样的路径，以及 $\kappa = 2n + 1$ 个 G_0的副本，根据鸽巢原理，存在某个 G_π 不存在不一致。

考虑一条路径 $\mathbf{P}^i$ 以及一个块构件 $\hat{}B_i^{G_z},_j$，其中 $z \in [\kappa]$ 和 $j \in [m]$满足某些条件。令 $N(\hat{}B_i^{G_z},_j)$表示位于 $\hat{}B_i^{G_z},_j$ 右侧的块构件，它由顶点 $n_1(\hat{}B_i^{G_z},_j)$、 $n_2(\hat{}B_i^{G_z},_j)$ 和$n_3(\hat{}B_i^{G_z},_j)$组成，其中 p_i^3、$_j$ 和 $n_1(\hat{}B_i^{G_z},_j)$在 G中通过一条边相连。此外，令 $\hat{}B_i^{G_{z'}},_{j'}$ 表示某个出现在 $\hat{}B_i^{G_z},_j$右侧的块构件，其中满足以下任一条件：a) $z' = z$ 和 $j' > j$，或 b) $z' > z$ 和 $j' \in [m]$。在这种情况下：

- 如果 $p^1_{i,j}(G_z) \in S$，那么 $p^1_{i,j'}(G_{z'}) \in S$。这对 $N(\hat{B}^{G_z}_{i,j})$ 成立，因为否则 $p^3_{i,j}(G_z)$ 在 $G - S$ 中的度为 2。假设这对 $\hat{B}^{G_z}_{i,j}$ 之后直到某个块构件 $\hat{B}^{G_{z^*}}_{i,j^*}$ 之前的所有块构件都成立。那么，这对 $N(\hat{B}^{G_{z^*}}_{i,j^*})$ 也成立，因为否则 $p^3_{i,j^*}(G_{z^*})$ 在 $G - S$ 中的度为 2。如果 $p^2_{i,j}(G_z) \in S$，那么要么 $p^1_{i,j'}(G_{z'}) \in S$，要么 $p^2_{i,j'}(G_{z'}) \in S$。这对 $N(\hat{B}^{G_z}_{i,j})$ 成立，因为否则 $n_3(\hat{B}^{G_z}_{i,j}) \in S$，由此可知 $n_1(\hat{B}^{G_z}_{i,j})$ 在 $G - S$ 中的度为 2。假设这对 $\hat{B}^{G_z}_{i,j}$ 之后直到某个块构件 $\hat{B}^{G_{z^*}}_{i,j^*}$ 之前的所有块构件都成立。那么，这对 $N(\hat{B}^{G_{z^*}}_{i,j^*})$ 也成立，因为否则 $n_3(\hat{B}^{G_{z^*}}_{i,j^*}) \in S$，由此可知 $n_1(\hat{B}^{G_{z^*}}_{i,j^*})$ 在 $G - S$ 中的度为 2。最后，如果 $p^3_{i,j}(G_z) \in S$，那么 S 包含后续每个块构件 $\hat{B}^{G_{z'}}_{i,j'}$ 中的一个顶点，因为它在每个块构件中恰好包含一个顶点。

因此，可以得出每条路径至多导致 2 个不一致；由于总共有 n 条路径，因此存在一个一致的副本 G_π。◁

现在，考虑赋值 $f\colon X \to Y$，其中 $f(x_i) = y$ if $p^y_{i,j}(G_\pi) \in S$。这是一个有效的赋值，因为 G_π 是一致的，而且每个块构件中都恰有一个顶点属于 S。我们将证明它满足所有约束。考虑约束 c_j。关于约束构件 $\hat{C}^{G_\pi}_j$，S 恰好包含 $s(c_j) - 1$ 个顶点，其中没有一个是叶节点。设 $v^j_\ell(G_\pi) \notin S$ 是团中唯一不属于 S 的顶点。由于 $l^j_\ell(G_\pi) \notin S$，因此块构件 $\hat{B}^{G_\pi}_{i,j}$ 中 $v^j_\ell(G_\pi)$ 的每个邻居都属于 S。在这种情况下，与 $v^j_\ell(G_\pi)$ 对应的满足赋值是 f 的限制，因此 f 满足 c_j。由于这对任意 j 都成立，ϕ 得到满足。◀

▶ **引理 6.** 成立：路径宽度$(G) \leq n + O(1)$。

证明。 我们将通过提供一种混合搜索策略来清除 G，并且同时使用的搜索者数量不超过这一数值。由于对于混合搜索数 ms，有 pw$(G) \leq$ ms$(G) \leq$ 路径宽度$(G) + 1$，因此我们将证明 ms$(G) \leq n + 2 \cdot 3^q$，由此即可得出结论。

从图 G_1 开始。将 $2s(c_1)$ 个搜索员放置到 $\hat{C}^{G_1}_1$ 的所有顶点上，并在顶点 $p^1_{i,1}(G_1)$ 上放置 n 个搜索员，其中 $i \in [n]$。这样，约束构件的所有边都被清理了。接下来，我们将描述清理 $\hat{B}^{G_1}_{i,1}$ 的过程。沿连接 $p^1_{i,1}(G_1)$ 和 $p^2_{i,1}(G_1)$ 的边移动搜索员，然后沿连接 $p^2_{i,1}(G_1)$ 和 $p^3_{i,1}(G_1)$ 的边移动。对所有 i 重复整个过程，从而清理所有块构件 $\hat{B}^{G_1}_{i,1}$ 以及这些块构件与 $\hat{C}^{G_1}_1$ 之间的边。

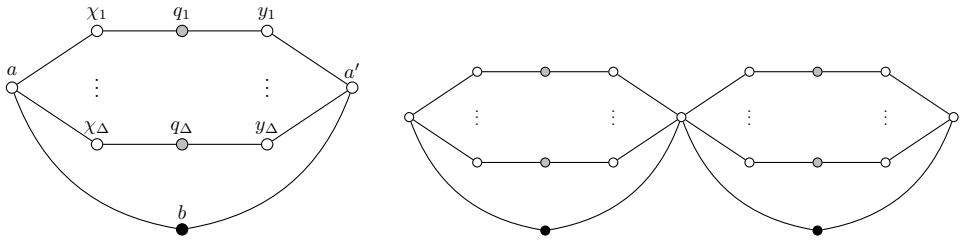
为了清除图中的其余部分，我们首先将搜索者从 $\hat{C}^{G_z}_j$ 移至 $\hat{C}^{G_z}_{j+1}$（若 $j < m$）或移至 $\hat{C}^{G_{z+1}}_1$（作为另一种选择）（必要时可能引入新的搜索者），然后通过清除相应的块构件继续进行：首先将所有搜索者从$p^3_{i,j}(G_z)$ 滑动至 $p^1_{i,j+1}(G_z)$（若 $j < m$），或作为另一种选择滑动至 $p^1_{i,1}(G_{z+1})$。重复这一过程，最终我们至多使用 $n + 2 \cdot 3^q = n + \mathcal{O}(1)$ 个搜索者清除 G 的所有边。◀

因此，根据引理 6，我们可以在多项式时间内构造一个图 G，其路径宽度为 pw$(G) \leq n + \mathcal{O}(1)$；根据引理 3 和引理 4，判定是否存在大小为 $|S| \leq k$ 的 $S \subseteq V(G)$ 且 $G - S$ 的最大度至多为 1，等价于判定 ϕ 是否可满足。在这种情况下，假设存在一个运行时间为 $\mathcal{O}^\star((3-\varepsilon)^{\text{pw}(G)})$ 的算法，用于求解 $\Delta = 1$上的有界度顶点删除问题，则可以在$\mathcal{O}^\star((3-\varepsilon)^{\text{pw}(G)}) = \mathcal{O}^\star((3-\varepsilon)^{n+\mathcal{O}(1)}) = \mathcal{O}^\star((3-\varepsilon)^n)$ 时间内判定 q-CSP-3（对任意常数 q），这与指数时间假设根据定理 1 矛盾。◀

▶ **定理 7**。 对于任意常数 $\varepsilon > 0$，不存在运行时间为 $O^\star((\Delta + 2 - \varepsilon)^{\mathrm{pw}})$ 的算法来判定有界度顶点删除 问题（针对 $\Delta \geq 2$），其中 pw 表示输入图的路径宽度，除非强指数时间假设不成立。

证明。固定任意一个正的 $\varepsilon > 0$，我们要证明该定理。令 $B \geq 4$。对于某个 q，它是仅依赖于 ε 的常数，我们将 q-约束满足问题-B 归约到针对 $\Delta = B - 2$ 的有界度顶点删除，并确保：如果所得的有界度顶点删除实例能在 $\mathsf{O}^\star((\Delta + 2 - \varepsilon)^{\mathrm{pw}})$ 时间内求解，那么我们就能得到一个用于求解 q-约束满足问题-B 的算法，而根据定理 1，这将与强指数时间假设相矛盾。为此，令 ϕ 为一个 q-约束满足问题-B 实例，它有 n 个变量 $X = \{x_i \mid i \in [n]\}$，取值来自集合 $Y = [0, \Delta + 1]$，并有 m 个约束 $C = \{c_j \mid j \in [m]\}$，其中 $\Delta = B - 2$。对于每个约束，我们给定一个至多包含 q 个变量的集合，这些变量参与该约束；还给定这些变量的满足赋值列表，其大小记为 $s : C \to [B^q]$，即 $s(c_j) \leq B^q = \mathsf{O}(1)$ 表示约束 c_j 的满足赋值数量。我们将在多项式时间内构造一个针对 $\Delta = B - 2$ 的有界度顶点删除等价实例 $\mathrm{I} = (G, k)$，其中 $\mathrm{pw}(G) \leq n + \mathsf{O}(1)$。

块与变量构件。对于每个变量 x_i 和每个约束 c_j，构造一个块构件 $B_{i,j}$，如图 2a 所示。为此，我们引入顶点a, a', b, χ_i, y_i和 q_i，其中 $i \in [\Delta]$。然后，我们在 b 上附加 Δ 个叶节点，并在顶点q_i上附加 $\Delta - 1$ 个叶节点。最后，我们添加边 $\{a, b\}$、$\{a', b\}$、$\{a, \chi_i\}$、$\{a', y_i\}$、$\{\chi_i, q_i\}$和 $\{y_i, q_i\}$。直观地说，我们将删除 b 以及从 $\{\chi_i \mid i \in [\Delta]\}$ 中删除 p 个顶点、从 $\{y_i \mid i \in [\Delta]\}$ 中删除 $\Delta - p$ 个顶点，对应于一种赋值：在该赋值下，x_i 取值为 $p \in [0, \Delta]$；而删除 a 则对应于一种赋值，在该赋值下，x_i 取值为 $\Delta + 1$。接下来，对于 $j \in [m - 1]$，我们将块构件 $\hat{}B_{i,j}$ 与 $\hat{}B_{i,j+1}$ 串联连接，使得 $\hat{}B_{i,j}$ 中的顶点 a' 就是 $B_{i,j+1}$中的顶点 $\hat{}a$，从而得到 n 条“路径” P_1，……，P_n，每条路径都由 m 个串联连接的块构件组成，称为变量构件。图 2 展示了示意图。



(a) $\Delta \geq 2$对应的块构件。 **(b)** 串联的块构件。

图 2 灰色顶点连接有 $\Delta - 1$，黑色顶点连接有 Δ 个叶节点。

约束构件。 该构件根据图其余部分中的选择来确定约束满足情况。对于约束 c_j，按如下方式构造约束$\hat{}$构件 C_j：

构造一个包含 $s(c_j)$ 个顶点的团 $v_1^j, \ldots, v_{s(c_j)}^j$，并在这些顶点与 c_j 的满足赋值之间任意固定一个一一映射。为团中的每个顶点附加 Δ 个未命名叶节点。如果变量 x_i 出现在约束 c_j 中，且 v_ℓ^j 对应一个满足赋值，其中 x_i 的值为 $p \in Y$，则考虑两种情况：(i) 如果 $p \in [0, \Delta]$，就在 v_ℓ^j 与 $\hat{}B_i$ 中的顶点 b 之间添加一条边 $_j$。此外，在 v_ℓ^j 与 $\{\chi_i \mid i \in [p]\} \cup \{y_i \mid i \in [p + 1, \Delta]\}$ 之间添加边。

(ii) 另一方面，如果 v_ℓ^j 对应一个满足赋值，其中变量 x_i 的取值为 $\Delta+1$，则在 v_ℓ^j 与 $\hat{}B_{i,j}$ 的顶点 a 之间添加一条边。

令图 G_0 对应于包含所有变量构件 P_i 以及所有约束构件 $\hat{}C_j$的图，其中 $i \in [n]$ 和 $j \in [m]$。我们将块构件 $\hat{}B_{i,j}$、变量构件 P_i以及 G_z 的约束构件 $\hat{}C_j$ 分别称为 $\hat{}B_i^{G_z},_j$，$P_i^{G_z}$ 和 $\hat{}C_j^{G_z}$。为构造图 G，引入 $\kappa = \kappa_1 \cdot \kappa_2$ 个 G_0的副本 $G_1, \ldots, G_\kappa$，其中 $\kappa_1 = n+1$ 和 $\kappa_2 = (2\Delta+1)n+1$，并按如下方式依次连接：对于 $i \in [n]$ 和 $j \in [\kappa-1]$，$\hat{}B_i^{G_j},_m$ 中的顶点 a' 是$\hat{}B_i^{G_{j+1}}{}_1$ 中的顶点 a。令 P^i 表示由 $P_i^{G_1}, \ldots, P_i^{G_\kappa}$得到的"路径"。令 $k = n + \kappa \cdot k_c$，其中 $k_c = mn(\Delta+1) + \sum_{j=1}^{m}(s(c_j)-1)$，并令 $\mathrm{I} = (G,k)$ 表示针对 $\Delta = B-2$的有界度顶点删除实例。

▶ **引理 8.** 如果 ϕ 可满足，则存在 $S \subseteq V(G)$，使得 $G-S$ 的最大度至多为 Δ，且满足 $|S| \leq k$。

证明。令 $f\colon X \to Y$为一个满足所有约束 c_j的赋值。我们将给出一个大小为 $|S| \leq k$ 的集合 $S \subseteq V(G)$，使得 $G-S$ 的最大度至多为 Δ。首先，对于每个约束构件 $\hat{}C_j$，由于 f 是一个满足赋值，因此存在某个顶点 v_ℓ^j，它对应于将 f 限制到涉及 c_j的至多 q 个变量上的限制。将 v_l^j 的每个顶点（对于 $l \in [s(c_j)] \setminus \{\ell\}$）加入 S。因此，S 恰好包含 $\kappa \cdot \sum_{j=1}^{m}(s(c_j)-1)$ 个这样的顶点。然后，对于每个 $i \in [n]$，考虑以下两种情况：

- 如果 $f(x_i) = \Delta+1$，那么对于每个块构件 $\hat{}B_{i,j}$，将顶点 a、a' 和 q_i（对于 $i \in [\Delta]$）加入 S。由于一个块构件的顶点 a' 就是其后续块构件的顶点 a，因此在这种情况下，S 包含 $\kappa \cdot m \cdot (\Delta+1)+1$ 个顶点。另一方面，如果 $f(x_i) = p \in [0,\Delta]$，那么对于每
- 个块构件 $\hat{}B_{i,j}$，将顶点 b、χ_i 和 y_j（对于 $i \in [p]$ 和 $j \in [p+1,\Delta]$）加入 S。在这种情况下，S 包含 $\kappa \cdot m \cdot (\Delta+1)$ 个顶点。

因此，S至多包含 $\kappa \cdot \sum_{j=1}^{m}(s(c_j)-1) + n \cdot (\kappa \cdot m \cdot (\Delta+1)+1) = n + \kappa \cdot k_c = k$个顶点。

接下来要证明 $G-S$ 的最大度至多为 Δ。考虑任意一个约束构件 $\hat{}C_j^{G_z}$。其中存在的任何叶节点在 $G-S$中的度均为 0 或 1。令 $v_\ell^j(G_z) \notin S$ 表示该约束构件的团中唯一不属于 S的顶点。这个顶点对应于 c_j的一个满足赋值，而 c_j是 f的一个限制。由于

- 它的 Δ 个相邻叶节点均不属于 S，$v_{\ell'}^j(G_z) \in S$，对于所有 $\ell' \neq \ell$，$v_\ell^j(G_z)$ 在 $\hat{}B_i^{G_z},_j$
- 中都没有邻居：考虑两种情况，要么 a 是 $v_\ell^j(G_z)$ 的邻居，要么顶点 b, χ_i, y_j 是 $v_\ell^j(G_z)$
- 的邻居，其中 $i \in [p]$ 和 $j \in [p+1,\Delta]$ 对某个 $p \in [0,\Delta]$ 成立。在这两种情况下，由于 $v_\ell^j(G_z)$ 对应于赋值 g，而该赋值是 f的一个限制，因此它在 $\hat{}B_i^{G_z},_j$ 中的所有邻居都属于 S。

因此，$G-S$中约束的顶点与块构件之间没有边。

最后，对于 $i \in [n]$，$,_j$ 在 $\mathcal{P}^i$ 的每个块构件 $\hat{}B_i$ 中，没有顶点的度大于 Δ $G-S$：

- 如果 $f(x_i) = \Delta+1$，那么顶点 b 的度为 Δ，而顶点 χ_i, y_i的度为 0。
- 另一方面，如果 $f(x_i) = p \neq \Delta+1$，则所有顶点 q_i 的度均为 Δ，因为它们的一个邻居属于 S。至于未删除的顶点 χ_i 和 y_i，它们的度等于 $2 \leq \Delta$。最后，任意顶点 a 的度至多为 Δ，因为它有 $\Delta-p$个邻居来自它作为 a 出现的块构件，还有 p 个邻居来自它作为 a'出现的块构件。

证明到此结束。 ◀

▶ **引理 9.** 如果存在 $S \subseteq V(G)$ 使得 $G - S$ 的最大度为 Δ 且 $|S| \leq k$, 那么 ϕ是可满足的。

证明。令 $S \subseteq V(G)$，使得 $G-S$的最大度为 Δ且 $|S| \leq k$。对于 G_z，考虑 $\hat{}B_i^{G_z},_j$ 中属于 S 的顶点子集与某个变量赋值下 x_i 的取值之间的映射。具体而言，包含顶点 b 以及 $\{\chi_i \mid i \in [n]\}$ 中的 p 个顶点和 $\{y_i \mid i \in [n]\}$ 中的 $\Delta - p$ 个顶点的 S，映射到使 x_i取值为 $p \in [0, \Delta]$的赋值；而仅包含顶点 a 的 S，映射到使 x_i 取值为 $\Delta+1$的赋值。任何其他子集都对应于 x_i 的无效赋值。我们称变量构件 $P_i^{G_z}$中出现不一致，当且仅当存在块构件 $\hat{}B_i^{G_z},_j$ 和 $\hat{}B_i^{G_z},_{j+1}$，使得每个块构件中属于 S 的顶点映射到 x_i 的无效赋值或不同赋值。如果 G_z 的块构件 $\hat{}B_i^{G_z},_j$ 中发生的删除映射到 x_i的某个有效赋值，且对于每个 $i \in [n]$，其变量构件 $P_i^{G_z}$中均未出现不一致，则称 G_z 是一致的。

▷ 断言 10. 存在 $\pi \in [\kappa]$，使得 G_π 是一致的。此外， S包含顶点 v_l^j，其中 $l \in [s(c_j)] \setminus \{\ell\}$，对于某个来自 $\hat{}C_j^{G_\pi}$ 的 $\ell \in [s(c_j)]$, 。

证明。在下文中，我们假设块构件 $\hat{}B_i^{G_z},_j$中的顶点 a' （若 $j < m$，则与 $\hat{}B_i^{G_z},_{j+1}$ 中的顶点 a 重合；或者与 $\hat{}B_{i\,1}^{G_{z+1}}$ 中的顶点重合）不属于 $\hat{}B_i^{G_z},_j$的顶点，以免重复计算（块构件 $\hat{}B_i^{G_\kappa},_m$除外）。因此，对于 $z \in [\kappa - 1]$，$\hat{}B_i^{G_z},_m$ 中的顶点 a' 不属于 $V(G_z)$，而属于 $V(G_{z+1})$。我们首先证明，对于所有 $z \in [\kappa]$，都有 $|S \cap V(G_z)| \geq k_c$。注意，每个约束构件 $\hat{}C_j$中至少有 $s(c_j) - 1$ 个顶点属于 S：每个此类构件都包含一个大小为 $s(c_j)$的团，其中每个顶点都连接着 Δ 个叶节点。此外，对于每个块构件 $B_i,_j$，由于 b 以及顶点 q_i 的度都大于 Δ，而任意两个顶点都没有共同邻居，因此需要进行 $\Delta + 1$ 次删除。注意，由于 b 的度为 $\Delta + 2$，即使删除顶点 a'，下一个块构件 b 中的顶点度仍大于Δ。因此，对于每个 G_z，至少要删除 $\sum_{j=1}^{m}(s(c_j) - 1) + nm(\Delta + 1) = k_c$ 个顶点，由此可得 $|S \cap V(G_z)| \geq k_c$。

由于 $\kappa = \kappa_1 \cdot \kappa_2 = (n + 1) \cdot \kappa_2$，而 $|S| \leq n + \kappa \cdot k_c$，因此存在 κ_2个连续的副本 $G_{z'+1}$，…，$G_{z'+\kappa_2}$，使得 $|S \cap V(G_w)| = k_c$，对 $w \in [z' + 1, z' + \kappa_2]$均成立。因此，对于任意这样的 G_w，S 恰好包含来自每个 $\hat{}C_j^{G_w}$的 $s(c_j)$ 个顶点，以及来自 $\hat{}B_i^{G_w},_j$的恰好 $\Delta+1$ 个顶点，其中 $i \in [n]$ 且 $j \in [m]$。在这 $\Delta+1$ 个顶点中，对每个 $i \in [n]$，必有 1 个属于集合 $\{\chi_i, y_i, q_i\}$。最后一个顶点必须是顶点 b或其某个邻居；否则， b 在 $\hat{}B_i^{G_w},_j$ 的顶点中就会有 $\Delta + 1$ 个邻居。假设在 $b \notin S$的情况下，属于 S 的 b 的邻居是 a：若非如此，则可以构造一个基数相同且满足该结论的删除集 S'；并且由于 a 的邻域包含任何连接到 b 的叶节点的邻域， $G - S'$ 的最大度至多为 Δ。

设 $\hat{}B_i^{G_w},_j$ 是其中一个块构件，并用 $N(\hat{}B_i^{G_w},_j)$ 表示其 a顶点为 $\hat{}B_i^{G_w},_j$ 的顶点 a' 所对应的块构件。我们将考虑两种情况：要么是 $a \in S$，要么是 $b \in S$。

如果 $a \in S$，那么对于任意块构件 $\hat{}B_i^{G_{w'}},_{j'}$，其中满足以下条件之一：a) $w' = w$ 且 $j' > j$，或 b) $w' > w$ 且 $j' \in [m]$，均有 $a \in S$：该命题对 $N(\hat{}B_i^{G_w},_j(G_w))$ 成立；否则，$\hat{}B_i^{G_w},_j$ 的顶点 b 在 $G - S$ 中的度为 $\Delta + 1$。假设该命题对某个 $\hat{}B_i^{G_{w^*}},_{j^*}$ 及之前的每个块构件都成立。那么，它对 $N(\hat{}B_i^{G_{w^*}},_{j^*})$ 也成立；否则，$\hat{}B_i^{G_{w^*}},_{j^*}$ 的顶点 b 在 $G - S$ 中的度为 $\Delta + 1$。

另一方面，假设 $b \in S$。我们将证明 P^i 至多会导致 2Δ 个不一致。注意，一旦我们从某个块构件中删除顶点 a，就会同时从所有后续块构件中删除顶点 a。因此，以下假设每个块构件恰好删除 $\Delta+1$ 个顶点，且其中不包括顶点 a。如果对于块构件 $\hat{}B_i^{G_w}$，j、α_p 个顶点分别从集合 $\{\chi_i \mid i \in [n]\}$、$\{q_i \mid i \in [n]\}$ 和 $\{y_i \mid i \in [n]\}$ 中选入 S，其中 p 是表示该块构件的索引，那么 $a'(\hat{}B_i^{G_w},j)$ 在 $G-S$ 中有 $\Delta-\gamma_p$ 条来自 $\{y_i \mid i \in [n]\}$ 的入边，其中 $a'(\hat{}B_i^{G_w},j)$ 表示 $\hat{}B_i^{G_w},j$ 的顶点 a'。注意，顶点 $a'(\hat{}B_i^{G_w},j)$ 与 $a(N(\hat{}B_i^{G_w},j))$ 重合，因此后者从块构件 $N(\hat{}B_i^{G_w},j)$ 最多可以有 γ_p 条入边，即 $\gamma_p+\alpha_{p+1} \geq \Delta$。由于 $\alpha_i+\beta_i+\gamma_i=\Delta$、$\gamma_p \geq \gamma_{p+1}+\beta_{p+1}$ 和 $\alpha_{p+1} \geq \alpha_p+\beta_p$，结论成立。于是，每个不一致都对应于 $\alpha_{p+1} > \alpha_p$ 或 $\gamma_{p+1} < \gamma_p$。考虑到 $0 \leq \alpha_i$、$\gamma_i \leq \Delta$，可以推断可能出现的不一致数至多为 2Δ；其中一半是由于 α_p 增加造成的，另一半则是由于 γ_p 减少造成的（可以假设在最坏情况下，这两种情况彼此独立地发生）。

因此，对于每条 n 路径 $\mathcal{P}_i$，在每个块构件中删除不超过 $\Delta+1$ 个顶点时，不一致的最大数量为 $2\Delta+1$（$+1$ 是由于 $a \in S$的情况）。

ˆ 现在，假设对于某个索引为 p的块构件 $B_{i,j}$，有 $\beta_p > 0$。接下来我们将证明 $\alpha_p \neq \alpha_{p+1}$。假设情况并非如此。则有 $\gamma_p+\alpha_{p+1} \geq \Delta \Longleftrightarrow \gamma_p+\alpha_p \geq \Delta$，这意味着 $\beta_p \leq 0$，矛盾。

由于整个构造重复了 $\kappa_2=(2\Delta+1)n+1$ 次，根据鸽巢原理，存在一个副本 G_π，其中 $P_i^{G_\pi}$的块构件中发生的所有删除都映射到同一个有效赋值（这确实是一个有效赋值，因为 $\beta_p=0$）。此外，由于对于每个约束构件 $\hat{}C_j^{G_\pi}$，恰有 $s(c_j)-1$个顶点被包含在 S中，而每个此类构件都有一个大小为 $s(c_j)$的团，其中每个顶点都连接着 Δ 个叶节点，因此 S 不能包含来自 $\hat{}C_j^{G_\pi}$的任何叶节点。◁

考虑如下赋值 f：$X \to Y$在 ϕ 的变量上：若 $a \in S$，则令 $f(x_i)=\Delta+1$；或者令 $f(x_i)=y$，其中 $y \in [0,\Delta]$ 等于$\{\chi_i \mid i \in [\Delta]\}$ 中属于 S 的顶点数，

其中 a 和 χ_i 指的是块构件 $\hat{}B_i^{G_\pi},j$ 的顶点，其中 $j \in [m]$ 是某个值（因为 G_π 是一致的，选择 j 并不重要）。

我们将证明 f 满足所有约束。考虑约束 c_j。对于约束构件 $\hat{}C_j^{G_\pi}$，S 恰好包含 $s(c_j)-1$ 个顶点，其中没有一个是叶节点。令 $v_\ell^j \notin S$ 为 $\hat{}C_j^{G_\pi}$ 中唯一一个不属于S的非叶节点。由于它的叶节点均不属于 S，因此 v_ℓ^j 在块构件 $\hat{}B_i^{G_\pi},j$ 中的每个邻居都属于 S。在这种情况下，请注意，删除块构件 $\hat{}B_i^{G_\pi},j$ 中v_ℓ^j 的所有邻居可映射为 x_i的一个赋值：如果唯一的邻居是 a，那么 $f(x_i)=\Delta+1$；或者，其邻域还包含 b 以及顶点 $\{\chi_i \mid i \in [p]\} \cup \{y_i \mid i \in [p+1,\Delta]\}$，这意味着 $\{\chi_i \mid i \in [\Delta]\}$ 中恰好有 p 个顶点属于 S。因此，与 v_ℓ^j 对应的满足赋值是f的一个限制，故 f 满足 c_j。由于这对任意 j都成立，因此 ϕ 得到满足。◀

▶ **引理 11.** 成立：路径宽度$(G)=n+O(1)$。

证明。 我们将通过给出一种混合搜索策略来清理 G，并同时最多使用这么多名搜索者。由于混合搜索数 ms 满足

$\mathsf{pw}(G) \leq \mathsf{ms}(G) \leq \mathsf{pw}(G)+1$，我们将证明 $\mathsf{ms}(G) \leq n+3+2\cdot B^q$，从而得出结论。

从图 G_1 开始。将 $2s(c_1)$ 个搜索者放置在 $\hat{C}_1^{G_1}$ 的团顶点以及每个团顶点对应的一个叶节点上，并将 n 个搜索者放置在块构件 $\hat{B}_{i\,1,}^{G_1}$ 的顶点 a 上，其中 $i \in [n]$。通过在约束构件的叶节点之间移动放置在叶节点上的搜索者，可以清除约束构件的所有边。接下来我们将描述清除 $\hat{B}_{i\,1,}^{G_1}$ 的过程。将三个额外的搜索者移至块构件的顶点 b 和 a'，以及 b 的一个叶节点上。在 b 的所有不同叶节点之间移动后者。最后，放置在 b 及其叶节点上的搜索者可以清除该构件剩余的边：将一个放置在 χ_i 上，另一个放置在 q_i 的一个叶节点上。沿着连接 χ_i 与 q_i 的边滑动第一个搜索者。在 q_i 的所有叶节点之间移动后一个搜索者，然后将其放置到 y_i。对所有 i 重复整个过程，最后移除除放置在 a' 上的搜索者之外的所有搜索者。按照相同的过程，最终可以清除所有块构件 $\hat{B}_{i\,,1}^{G_1}$。

为了清理图的其余部分，我们首先将搜索者从 $\hat{C}_j^{G_z}$ 移到 $\hat{C}_{j+1}^{G_z}$（如果 $j<m$），或者移到 $\hat{C}_1^{G_{z+1}}$（也可以按需引入新的搜索者），清理后者，然后继续清理相应的块构件。重复这一过程，最终我们至多使用 $n+3+2\cdot B^q = n+\mathcal{O}(1)$ 个搜索者清理 G 的所有边。◀

因此，我们可以在多项式时间内构造一个图 G，其路径宽度为 $\mathsf{pw}(G) \leq n+\mathcal{O}(1)$（根据引理 11），使得根据引理 8 和 9，判定是否存在大小为 $|S| \leq k$ 的 $S \subseteq V(G)$，且 $G-S$ 的最大度至多为 Δ，等价于判定 ϕ 是否可满足。在这种情况下，假设存在一个用于有界度顶点删除的 $\mathcal{O}^\star((\Delta+2-\varepsilon)^{\mathrm{pw}(G)})$ 算法，就可以在$\mathcal{O}^\star((\Delta+2-\varepsilon)^{\mathrm{pw}(G)}) = \mathcal{O}^\star((B-\varepsilon)^{n+\mathcal{O}(1)}) = \mathcal{O}^\star((B-\varepsilon)^n)$ 时间内判定 q-CSP-B，这与定理 1 所述的 SETH 矛盾。◀

3.2 缺陷着色

本小节中，我们将给出以输入图的树宽加目标度为参数的缺陷着色问题的紧下界。我们首先在第 3.2.1 节介绍构造中用到的多种实用构件，随后在第 3.2.2 节给出下界，并在第 3.2.3节给出一个具有匹配运行时间的算法。

3.2.1 着色构件

这里我们设计了各种构件，主要用于第 3.2.2 节。我们大量借鉴了 [5], 中提出的构造，其中一些内容为求完整起见，在此简要介绍。

直观而言，我们的目标是扩充 [5], 的工具箱，并构造一些构件，使得对于图 G 的任意 (χ_d,Δ)-着色 c（其中 $\chi_\mathrm{d} \geq 2$ 且 $\Delta \geq 1$），都能推出顶点 $v_1, v_2, c_1, c_2 \in V$（G）之间的如下关系（其中 c_1, c_2 可能重合）：

- $D(v_1, v_2)$：顶点 v_1 和 v_2 接收不同颜色，即 $c(v_1) \neq c(v_2)$，
- $E(v_1, v_2, c_1, c_2)$：如果顶点 v_1 接收 c_1的颜色，则顶点 v_2 不接收 c_2的颜色，即 $c(v_1) = c(c_1) \implies c(v_2) \neq c(c_2)$，
- $I(v_1, v_2, c_1, c_2)$：如果顶点 v_1 接收 c_1的颜色，则顶点 v_2 接收 c_2的颜色，即 $c(v_1) = c(c_1) \implies c(v_2) = c(c_2)$。

我们以 [5], 提出的结果为基础，其中介绍了相等构件 $Q(v_1, v_2, \chi_\mathrm{d}, \Delta)$以及调色板构件 $P(v_1, v_2, v_3, \chi_\mathrm{d}, \Delta)$。

▶ 引理*12*（ [5]的引理 *3.5*）。设 $G = (V, E)$是一个具有 $v_1, v_2 \in V$的图，并设 G'是由 G添加 $Q(u_1, u_2, \chi_d, \Delta)$的一个副本，并将 u_1与 v_1、 u_2与 v_2分别识别后得到的图。那么， G'的任意(χ_d, Δ)-着色都必须为 v_1, v_2着上相同的颜色。此外，如果存在一种(χ_d, Δ)-着色，使 G中的 v_1, v_2颜色相同，则该着色可以扩展为 G'的一个(χ_d, Δ)-着色。

▶ 引理*13*（ [5]中的引理 *3.8*）。设 $G = (V, E)$为一个具有 v_1、 v_2、 $v_3 \in V$的图，并设 G'为通过向 G中添加一个 $P(u_1, u_2, u_3, \chi_d, \Delta)$的副本，并将$u_i$ 与 v_i 识别（对 $i \in [3]$）而得到的图。那么，在 G'的任意(χ_d, Δ)-着色中，$\{v_1, v_2, v_3\}$中的至少两个顶点必须共享同一种颜色。此外，如果存在一个 G的(χ_d, Δ)-着色，使得$\{v_1, v_2, v_3\}$中的两个顶点具有相同的颜色，则该着色可以扩展为 G'的(χ_d, Δ)-着色。

▶ **定义14**（差异构件）。对于 $\chi_d \geq 2$、 $\Delta \geq 0$、 $k \geq 0$，令 D（u_1、 u_2、 χ_d、 Δ、 k） 为如下定义的图： D 包含顶点 u_1 和 u_2，后者具有 k 个叶节点l_1、.. .、 l_k，此外还包含 k个相等构件 Q（x、 y、 χ_d、 Δ）的副本；在每个副本中，我们将 x 与 u_1 、 y 与 l_i识别（对 $i \in [k]$而言）。

▶ 引理*15.* 设 $G = (V, E)$是一个具有 $v_1, v_2 \in V$的图，并令 G'为从 G出发，添加 $D(u_1, u_2, \chi_d, \Delta, \Delta+1)$的一个副本，并将 u_1与 v_1、 u_2与v_2分别识别后得到的图。那么， G' 的任意(χ_d, Δ)-着色都必须为 v_1 和 v_2赋予不同的颜色。此外，如果存在一种(χ_d, Δ)-着色，使得 v_1 和 v_2颜色不同，则该着色可以扩展为 G'的一个(χ_d, Δ)-着色。

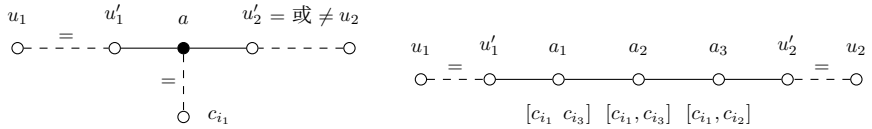
证明. 对于第一个结论，考虑 G'的一个(χ_d, Δ)-着色 c': $V(G') \to [\chi_d]$。注意，由于相等构件 [5,的性质（引理3）， $c'(v_1) = c'(l_i)$对所有 $i \in [\Delta + 1]$都成立，其中 l_i 表示由于差异构件而连接到 v_2 的叶节点。在这种情况下， v_2 至少有 $\Delta + 1$ 个颜色为 $c'(v_1)$的邻居，因此 $c'(v_2) \neq c'(v_1)$。

对于第二个陈述，令 c: $V(G) \to [\chi_d]$ 为 G 的一个 (χ_d, Δ)-着色，其中$c(v_1) \neq c(v_2)$。为了将其扩展为 G' 的一个 (χ_d, Δ)-着色，将顶点 l_i 着成与 v_1相同的颜色，并利用 [5,引理 35]对相等性构件中的顶点进行适当着色。◀

▶ **定义 16** (排斥构件)。对于 $\chi_d \geq 2$, $\Delta \geq 1$, $C = \{c_i \mid i \in [\chi_d]\}$，以及（不一定互不相同的） $i_1, i_2 \in [\chi_d]$，令 $E(u_1, u_2, c_{i_1}, c_{i_2}, C, \chi_d, \Delta)$ 为如下定义的图：

- 若 ***a)*** $i_1 = i_2$，或 ***b)*** $i_1 \neq i_2$ 和 $\chi_d = 2$，则令 E 包含顶点 $u_1, u_2, u'_1, u'_2, c_{i_1}$ 和 a。在 a 与u'_1, u'_2之间添加边，并添加构件 $Q(u_1, u'_1, \chi_d, \Delta)$、 $Q(c_{i_1}, a, \chi_d, \Delta)$ 和 $D(c_{i_1}, a, \chi_d, \Delta, \Delta - 1)$。若 $i_1 = i_2$，则添加构件 $Q(u_2, u'_2, \chi_d, \Delta)$；否则添加构件$D(u_2, u'_2, \chi_d, \Delta, \Delta + 1)$。
- 否则，也就是说，当 $i_1 \neq i_2$ 和 $\chi_d > 2$时，令 E 包含顶点 $u_1, u_2, u'_1, u'_2, c_{i_1}, c_{i_2}, c_{i_3}$和 a_1, a_2, a_3，其中 $i_3 \in [\chi_d] \setminus \{i_1, i_2\}$。令顶点 $u'_1, a_1, a_2, a_3, u'_2$构成一条路径，并添加以下构件：$Q(u_1, u'_1, \chi_d, \Delta)$,$Q(u_2, u'_2, \chi_d, \Delta)$,$P(c_{i_1}, c_{i_3}, a_1, \chi_d, \Delta)$, $D(c_{i_1}, a_1, \chi_d, \Delta, \Delta)$, $D(c_{i_3}, a_1, \chi_d, \Delta, \Delta)$, $P(c_{i_1}, c_{i_3}, a_2, \chi_d, \Delta)$, $D(c_{i_1}, a_2, \chi_d, \Delta, \Delta)$, $D(c_{i_3}, a_2, \chi_d, \Delta, \Delta)$,$P(c_{i_1}, c_{i_2}, a_3, \chi_d, \Delta)$, $D(c_{i_1}, a_3, \chi_d, \Delta, \Delta)$, $D(c_{i_2}, a_3, \chi_d, \Delta, \Delta)$。

▶ **引理 17.** 设 $G = (V, E)$ 是一个满足 $v_1, v_2 \in V$和 $P = \{p^i \mid i \in [\chi_d]\} \subseteq V$的图。对于某些（不一定互不相同的） $i_1, i_2 \in [\chi_d]$, G'，设该图由 G 通过以下方式得到：



(a) 或者是 $i_1 = i_2$，或者是 $i_1 \neq i_2$ 和 $\chi_d = 2$。**(b)** $i_1 \neq i_2$ 和 $\chi_d > 2$的情形。

图3 排斥构件 $E(u_1, u_2, c_{i_1}, c_{i_2}, C, \chi_d, \Delta)$。黑色顶点 a 连接有 $\Delta - 1$ 个叶节点 l，每个叶节点都参与相等构件 $Q(c_{i_1}, l, \chi_d, \Delta)$。顶点 v 下方的方括号 $[x, y]$ 表示存在调色板构件 $P(x, y, v, \chi_d, \Delta)$，并且 v 连接有 2Δ 个叶节点 l，其中一半参与构件 $Q(x, l, \chi_d, \Delta)$，另一半参与 $Q(y, l, \chi_d, \Delta)$。

向其中添加一个 $E(u_1, u_2, c_{i_1}, c_{i_2}, C, \chi_d, \Delta)$ 的副本，并将 u_1 与 v_1、u_2 与 v_2以及c_i 与 p^i分别识别。则在任意 (χ_d, Δ)-着色 $c' : V(G') \to [\chi_d]$ 中，若 G' 满足 *a)* 对不同的 $i, j \in [\chi_d]$，$c'(p^i) \neq c'(p^j)$，且 *b)* $c'(v_1) = c'(p^{i_1})$，则有 $c'(v_2) \neq c'(p^{i_2})$。此外，若存在一个 (χ_d, Δ)-着色 $c : V(G) \to [\chi_d]$，其中 G 满足 *a)* 对不同的 $i, j \in [\chi_d]$，$c(p^i) \neq c(p^j)$，且 *b)* $c(v_1) \neq c(p^{i_1})$ 或 $c(v_2) \neq c(p^{i_2})$，则可将其扩展为 G'的 (χ_d, Δ)-着色。

证明。对于第一个陈述，考虑一个 (χ_d, Δ)-着色 c'： $V(G') \to [\chi_d]$ 的 G'，其中每个顶点 p^i 都接受一个不同的颜色。假设 $c'(v_1) = c'(p^{i_1})$，并考虑以下情形：

- 如果满足以下任一条件：a) $i_1 = i_2$，或 b) $i_1 \neq i_2$ 和 $\chi_d = 2$，则 E 的顶点 a 有 $\Delta - 1$ 个颜色为 $c'(p^{i_1})$ 的叶节点。此外，由于构件 $Q(u_1, u_1', \chi_d, \Delta)$，$c'(u_1') = c'(p^{i_1})$ 成立。因此，$c'(v_2) \neq c'(p^{i_2})$ 成立，因为否则，由于构件 $Q(u_2, u_2', \chi_d, \Delta)$ 或 $D(u_2, u_2', \chi_d, \Delta, \Delta+1)$，$c'(u_2') = c'(p^{i_1})$ 成立，从而 a 具有 $\Delta+1$ 个同色邻居，这就产生了矛盾。否则，即如果 $i_1 \neq i_2$ 和 $\chi_d > 2$，则 a_1 同样有 Δ 个颜色为 $c'(p^{i_1})$ 的叶节点，以及一个通过相等构件与 v_1相连的邻居。因此，a_1 有 $\Delta+1$ 个颜色为 $c'(p^{i_1})$ 的邻居，并且由于调色板构件，$c'(a_1) = c'(p^{i_3})$ 成立。类似地，$c'(a_2) = c'(p^{i_1})$ 成立，并且 $c'(a_3) = c'(p^{i_2})$ 成立。假设 $c'(v_2) = c'(p^{i_2})$。那么，由于其 Δ 个叶节点以及顶点 u_2'（该顶点通过相等构件与 v_2相连），a_3 有 $\Delta + 1$ 个颜色为 $c(p^{i_2})$ 的邻居，这导致矛盾。

对于第二个陈述，考虑 χ_d, Δ-着色 $c : V(G) \to [\chi_d]$ 的 G，其中每个顶点 p^i 都获得不同的颜色，且满足 $c(v_1) \neq c(p^{i_1})$ 或 $c(v_2) \neq c(p^{i_2})$。考虑以下情形：

- 如果满足以下任一条件：a) $i_1 = i_2$，或 b) $i_1 \neq i_2$ 和 $\chi_d = 2$，那么令 E的顶点 a 及后者的$\Delta - 1$ 叶节点接收颜色 $c(p^{i_1})$。接着，用 $c(u_1)$为 u_1' 着色。如果 $i_1 = i_2$，则令 u_2' 接收颜色 $c(v_2)$；否则，它接收不同于 $c(v_2)$的颜色。在这两种情况下，a至多有 Δ 个同色邻居，并且利用引理 12 和 15，我们可以为相等关系构件和差异构件的所有内部顶点着色。否则，也就是说，如果 $i_1 \neq i_2$ 和 $\chi_d > 2$，则考虑以下两种情况：如果 $c(v_1) \neq c(p^{i_1})$，那么令 a_1 接收颜色 $c(p^{i_1})$，令 a_2 接收颜色 $c(p^{i_3})$，并令 a_3 接收颜色 $c(p^{i_1})$。或者，若 $c(v_2) \neq c(p^{i_2})$，则令 a_1 接收颜色 $c(p^{i_3})$，令 a_2 接收颜色 $c(p^{i_1})$，并令 a_3 接收颜色 $c(p^{i_2})$。在这两种情况下，所有顶点 a_1、a_2、a_3都恰有 Δ个同色邻居，接下来只需利用引理 12 和 13 为相等关系构件和调色板构件的内部顶点着色。

证明至此完成。 ◀

▶ **定义 18** (蕴含构件)。 对于 $\chi_d \geq 2$、 $\Delta \geq 1$、 $C = \{c_i \mid i \in [\chi_d]\}$以及（不一定互不相同的） i_1、 $i_2 \in [\chi_d]$，令 $I(u_1$、 u_2、 c_{i_1}、 c_{i_2}、 C、 χ_d、 $\Delta)$ 为如下定义的图： I 包含顶点 u_1 和 u_2，以及排除构造块 $E(u_1$、 u_2、 c_{i_1}、 c_k、 C、 χ_d、 $\Delta)$，对所有 $k \in [\chi_d] \setminus \{i_2\}$均如此。

▶ 引理 *19.* 设 $G = (V, E)$ 是一个图，满足 $v_1, v_2 \in V$和 $P = \{p^i \mid i \in [\chi_d]\} \subseteq V$；对于某些（不一定互不相同的） $i_1, i_2 \in [\chi_d]$，G'，设该图由 G 添加一个 $I(u_1, u_2, c_{i_1}, c_{i_2}, C, \chi_d, \Delta)$ 的副本，并将 u_1 与 v_1、 u_2 与 v_2 以及 c_i 与 p^i 识别得到。那么，在任意 (χ_d, Δ)-着色 $c' : V(G') \to [\chi_d]$ 中，若 G' 满足 *a)* 对不同的 $i, j \in [\chi_d]$，有 $c'(p^i) \neq c'(p^j)$；且 *b)* $c'(v_1) = c'(p^{i_1})$，则有 $c'(v_2) = c'(p^{i_2})$。此外，若存在一个 (χ_d, Δ)-着色 $c : V(G) \to [\chi_d]$，其中 G 满足 *a)* 对不同的 $i, j \in [\chi_d]$，有 $c(p^i) \neq c(p^j)$；且 *b)* $c(v_1) \neq c(p^{i_1})$ 或 $c(v_2) = c(p^{i_2})$，则可将其扩展为 G' 的一个 (χ_d, Δ)-着色。

证明。对于第一个陈述，考虑对 G' 的 (χ_d, Δ)-着色 $c' : V(G') \to [\chi_d]$，其中每个顶点 p^i 都获得一个不同的颜色。假设 $c'(v_1) = c'(p^{i_1})$。那么，由引理 17 可知，对于所有 $k \in [\chi_d] \setminus \{i_2\}$，都有 $c'(v_2) \neq c'(p^k)$，因此可得 $c'(v_2) = c'(p^{i_2})$。

对于第二个陈述，考虑对 G 的 (χ_d, Δ)-着色 $c : V(G) \to [\chi_d]$，其中每个顶点 p^i 都获得一个不同的颜色，此外还满足 $c(v_1) \neq c(p^{i_1})$ 或 $c(v_2) = c(p^{i_2})$。因此，对于所有 $k \in [\chi_d] \setminus \{i_2\}$，都有 $c(v_1) \neq c(p^{i_1})$ 或 $c(v_2) \neq c(p^k)$，由引理 17 可得该陈述。◀

下面的引理证明，使用前文所述的构件不会使图的路径宽度增加太多。请注意，其中一部分已在 [5] 中证明。

▶ **引理 20**。 设 $G = (V, E)$ 为一个图，令 G' 表示通过重复执行以下操作得到的图：找出下列构件之一的一个副本

- $Q(u_1, u_2, \chi_d, \Delta), P(u_1, u_2, u_3,$
- $\chi_d, \Delta), D(u_1, u_2, \chi_d, \Delta, k), E$
- $(u_1, u_2, c_{i_1}, c_{i_2}, C, \chi_d, \Delta), I($
- $u_1, u_2, c_{i_1}, c_{i_2}, C, \chi_d, \Delta),$

从图中移除其所有内部顶点，并在其各个端点之间添加尚未连接的所有边。然后，$\mathrm{tw}(G) \leq$ 最大$\{\mathrm{tw}(G'), O(\chi_d)\}$，且 $\mathrm{pw}(G) \leq \mathrm{pw}(G') + O(\chi_d)$。

证明。 相等关系构件和调色板构件的结果已在 [5, 引理 42]中给出。具体而言，文中证明了可以为相等关系构件或调色板构件构造一个路径分解，分别记作 T_Q 和 T_P，其宽度为 χ_d，且每个袋都包含顶点u_1， u_2 （对于调色板构件，还包含 u_3 ）。

差异构件。请记住，差异构件包含 k 个叶节点 l_i，连接到顶点 u_2。首先，正如 [5, 引理 42], 的证明中所观察到的， $Q(u_1, l_i, \chi_d, \Delta)$存在一个宽度为 χ_d的路径分解，其中每个袋都包含顶点 u_1 和 l_i。在这种情况下， $D(u_1, u_2, \chi_d, \Delta)$存在一个宽度为 $\chi_d + 1$的路径分解：依次连接 $Q(u_1, l_i, \chi_d, \Delta)$的路径分解，其中 $i \in [k]$，并将顶点 u_2加入所有袋中。将此路径分解称为 T_D，并注意其所有袋都同时包含顶点 u_1 和u_2。

排斥构件。首先考虑 $i_1 = i_2$ 或 $\chi_{\mathrm{d}} = 2$的情况。此时，该构件由顶点 u_1、 u_2、 u'_1、 u'_2、 c_{i_1}和 a组成。顶点 a与 u'_1、 u'_2均有一条边，同时还存在构件 $Q(u_1, u'_1, \chi_{\mathrm{d}}, \Delta)$、 $Q(c_{i_1}, a, \chi_{\mathrm{d}}, \Delta)$和 $D(c_{i_1}, a, \chi_{\mathrm{d}}, \Delta, \Delta - 1)$。如果$i_1 = i_2$，还存在相等构件 $Q(u_2, u'_2, \chi_{\mathrm{d}}, \Delta)$；否则，存在差异构件$D(u_2, u'_2, \chi_{\mathrm{d}}, \Delta, \Delta + 1)$。构造一个宽度为 $O(\chi_{\mathrm{d}})$ 的路径分解 T_E，由以下路径分解组成：

- χ_{d} 宽的 $Q(u_1, u'_1, \chi_{\mathrm{d}}, \Delta)$ 的路径分解，其中每个袋都包含 u_1, u'_1；
- χ_{d} 宽的 $Q(c_{i_1}, a, \chi_{\mathrm{d}}, \Delta)$ 的路径分解，其中每个袋都包含 c_{i_1}, a；
- $\chi_{\mathrm{d}} + 1$ 宽的 $D(c_{i_1}, a, \chi_{\mathrm{d}}, \Delta, \Delta - 1)$ 的路径分解，其中每个袋都包含 c_{i_1}, a；视具体情形而定，χ_{d} 宽的 $Q(u_2, u'_2, \chi_{\mathrm{d}}, \Delta)$ 的路径分解，或 $\chi_{\mathrm{d}} + 1$ 宽的 $D(u_2, u'_2, \chi_{\mathrm{d}}, \Delta, \Delta + 1)$ 的路径分解，其中每个袋都包含顶点 u_2, u'_2。

为了完成 T_E 的构造，在此路径分解的每个袋中加入必要的顶点，使得对于所有 $i \in [\chi_{\mathrm{d}}]$，$u_1$， u_2， u'_1， u'_2， c_i 都包含在每个袋中。

对于剩余情形，排斥构件由相等关系、调色板构件和差异构件组成。以类似的方式连接各个构件对应的路径分解，并随后添加顶点，以确保对于所有 $i \in [\chi_{\mathrm{d}}]$，每个袋都包含$u_1$、$u_2$、$u'_1$、$u'_2$、 a_1、 a_2、 a_3、 c_i，由此可得宽度为$\mathcal{O}(\chi_{\mathrm{d}})$ 的路径分解 T_E。

蕴含构件。 最后，对于蕴含构件的情形，注意它由排斥构件组成 $E(u_1$、 u_2、 c_{i_1}、 c_k、 C、 χ_{d}、 $\Delta)$，对于所有 $k \in [\chi_{\mathrm{d}}] \setminus \{i_2\}$。同样，考虑这些 $|C| - 1 = \chi_{\mathrm{d}} - 1$ 排斥构件各自的路径分解，由此得到宽度为 $\mathcal{O}(\chi_{\mathrm{d}})$ (注意，排斥构件的分解中每个袋都包含顶点 $C \cup \{u_1$、 $u_2\}$)。

现在，我们取 G' 的一个最优树分解或路径分解，记为 T'，并以此构造 G 的一个分解。考虑 G 中出现的一个小工具 $H \in \{Q, P, D, E, I\}$，其端点为 u_1, u_2（也可能还有 u_3 和 $C = \{c_1, \ldots, c_{\chi_{\mathrm{d}}}\}$）。由于在 G' 中，这些端点构成一个团，因此 T' 中存在一个袋包含它们。令 B 为满足条件的最小袋，即包含顶点数最少的袋。现在，如果 T' 是树分解，我们就取 T_H 并将其连接到 B。如果 T' 是路径分解，我们就在 B 之后立即插入分解 T_H，并在其中的所有袋中加入 B 的所有顶点。不难看出，在这两种情况下，所得分解仍然有效；我们可以对每个 H 重复此过程，直到得到 G 的一个分解。◀

3.2.2 下界

下面我们将介绍从 q-CSP-B 到缺陷着色的归约，其中 $\Delta \geq 1$ 和 $\chi_{\mathrm{d}} \geq 2$ 是任意固定值，且 $B = \chi_{\mathrm{d}} \cdot (\Delta + 1)$。在这种情况下，如果存在一个用于缺陷着色的$\mathcal{O}^\star((\chi_{\mathrm{d}} \cdot (\Delta + 1) - \varepsilon)^{\mathrm{pw}})$ 算法，其中 $\varepsilon > 0$，那么对于任意常数 q，就存在一个用于 q-CSP-B 的 $\mathcal{O}^\star((B - \varepsilon)^n)$ 算法；根据定理 1，这将导致强指数时间假设失效。

该归约与第 3.1 节中介绍的归约类似，由“长路径”组成，这些路径由串联的块构件构成；每个块构件对应于给定约束满足问题中的一个变量，而该构造的每一列则与其中一个约束相关联。

在整个章节中，我们将使用第 3.2.1 节介绍的着色构件。与有界度顶点删除问题的情况一样，我们首先从 $\Delta = 1$ 的情形开始

然后考虑 $\Delta \geq 2$ 的情形。

▶ **定理 21**。 对于任意常数 $\varepsilon > 0$ 和任意固定的 $\chi_d \geq 2$，除非强指数时间假设不成立，否则不存在一个 $O^*((2\chi_d - \varepsilon)^{pw})$ 算法来判定 G 是否存在$(\chi_d,1)$-着色，其中路径宽度 表示图的路径宽度。

证明。固定一个正数 $\varepsilon > 0$，我们希望证明该定理对这个数成立。令 $B = 2\chi_d$。对于某个 q，它是仅取决于 ε 的常数，我们将把 q-CSP-B 归约为适用于 $\Delta = 1$ 和 χ_d 种颜色的缺陷着色问题，记作 q-CSP-B。该归约确保：如果所得的缺陷着色实例能在 $O^*((2\chi_d - \varepsilon)^{pw})$ 时间内求解，那么我们就能得到一个用于求解 q-CSP-B 的算法，而根据定理 1，这将与强指数时间假设相矛盾。为此，令 ϕ 是一个 q-CSP-B 实例，其中有 n 个变量 $X = \{x_i \mid i \in [n]\}$，其取值来自集合 $Y = [B]$ ；此外还有 m 个约束 $C = \{c_j \mid j \in [m]\}$。对于每个约束，我们给定一个至多包含 q 个变量的集合，这些变量参与该约束；同时还给出这些变量的满足赋值列表，其大小记为 s： $C \to [B^q]$，也就是说， $s(c_j) \leq B^q = O(1)$ 表示约束 c_j 的满足赋值数量。我们将在多项式时间内构造一个等价实例 G，将其转化为适用于 $\Delta = 1$ 和 χ_d 种颜色的缺陷着色问题，且 $pw(G) \leq n + O(1)$。

由于我们将反复使用相等关系构件、差异构件和调色板构件（参见第 3.2.1 节），因此采用以下约定：每当 v_1、v_2、v_3 是我们已向 G 引入的顶点时，如果我们说添加一个相等构件 $Q(v_1, v_2)$、一个差异构件 $D(v_1, v_2)$ 或一个调色板构件 $P(v_1, v_2, v_3)$，这表示我们向 G 添加一个 $Q(u_1, u_2, \chi_d, \Delta)$ 、 $D(u_1, u_2, \chi_d, \Delta, \Delta + 1)$ 或 $P(u_1, u_2, u_3, \chi_d, \Delta)$ 的副本，然后分别将 u_1、u_2(u_3) 与 v_1、v_2(v_3) 识别为同一对象。

调色板顶点。构造一个由 χ_d 个顶点 $P = \{p^i \mid i \in [\chi_d]\}$。在顶点 p^i上连接一个叶节点 p^i_l，并添加相等性构件 Q（p^i， p^i_l），其中 $i \in [\chi_d]$。

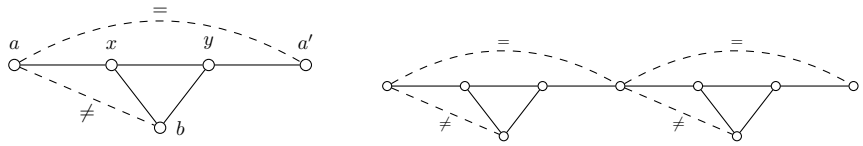
每当 v_1, v_2 是我们已经引入到 G 中的顶点时，如果我们说添加一个排斥构件 $E(v_1, v_2, v'_1, v'_2)$ 或蕴含构件 $I(v_1, v_2, v'_1, v'_2)$，这意味着我们向 G 中添加 $E(u_1, u_2, c_1, c_2, C, \chi_d, \Delta)$ 或 $I(u_1, u_2, c_1, c_2, C, \chi_d, \Delta)$ 的一个副本，然后分别将 u_1, u_2, c_i 与 v_1, v_2, p^i 识别为同一顶点，其中对所有 $i \in [\chi_d]$ 均如此。

块与变量构件。对于每个变量 x_i 和每个约束 c_j，构造一个块构件 $B_{i,j}$，如图 4a 所示。 u_1 与 u_2 之间的虚线表示相等关系 $Q(u_1, u_2)$ 或差异构件 $D(u_1, u_2)$，而不是一条边。如果 $\chi_d \geq 3$，则添加调色板构件 $P(a, b, x)$ 和 $P(a', b, y)$。接下来，对于 $j \in [m-1]$，我们将块构件 $\hat{}B_{i,j}$ 与 $\hat{}B_{i,j+1}$ 串联起来，使得 $\hat{}B_{i,j}$ 中的顶点 a' 就是 $\hat{}B_{i,j+1}$ 中的顶点 a，从而得到 n 条“路径” $P_1, \ldots, P_n$，每条路径由 m 个串联的块构件组成，称为变量构件。直观而言，变量构件旨在表示一个变量 x_i，因此需要具有 $2\chi_d$ 种不同的可行配置。这些配置通过先为 a 选择一种颜色（有 χ_d 种选择），再决定 x, y 中哪一个与 a 取相同的颜色（有两种选择）来构成。我们将证明，该构件的设置方式恰好使 x, y 中只有一个与 a 取相同的颜色（且 a'）。

约束构件。 此构件根据图中其余部分所做的选择，负责确定约束是否满足。对于约束 c_j，构造约束$\hat{}$构件 C_j，如图 5 所示：

- 引入顶点 r_w，其中 $w \in [s(c_j)]$，并添加 $Q(p^2, r_1)$，以及在$\chi_d \geq 3$时添加 $P(p^1, p^2, r_w)$，对于 $w \in [2, s(j)]$，

20 重新审视两个有界度问题的结构参数化



(a) 当 $\Delta = 1$ 时的块构件。 **(b)** 串联的块构件。

图 4 变量构件由串联的块构件组成。

对于 $w \in [s(j)]$,，引入顶点 v_w^j，以及在$\chi_{\mathrm{d}} \geq 3$时引入调色板构件 $P(p^1, p^2, v_w^j)$，并在这些顶点与 c_j 的满足赋值之间固定一个任意的一一映射；引入顶点 k_w，其中 $w \in [s(c_j)]$, 并添加 $Q(p^2, k_{s(c_j)})$，以及在 $\chi_{\mathrm{d}} \geq 3$ 时为 $w \in [s(j) - 1]$添加 $P(p^1, p^2, k_w)$，在顶点 k_w 与顶点 r_w、v_w^j之间添加边，以及添加 $D(k_w, r_{w+1})$；如果变量 x_i 涉及约束 c_j，且 v_ℓ^j 对应于一个赋值，其中x_i 的值为 $s \in Y$，那么：(i) 如果 $s \leq \chi_{\mathrm{d}}$，则添加蕴含构件 $I(v_\ell^j, a, p^1, p^s)$ 和 $I(v_\ell^j, x, p^1, p^s)$；(ii) 另一方面，如果 $\chi_{\mathrm{d}} < s \leq 2\chi_{\mathrm{d}}$，则添加蕴含构件 $I(v_\ell^j, a, p^1, p^{s'})$ 和排斥构件 $E(v_\ell^j, x, p^1, p^{s'})$，对于 $s' = s - \chi_{\mathrm{d}}$，ˆ 其中顶点 a 和 x 属于 B_i、$_j$。

直观地说，约束构件的设置方式会迫使某个 $\ell \in [s(c_j)]$, 顶点v_ℓ^j 获得颜色 1，从而“激活”我们添加到该顶点上的蕴含构件和排斥构件。这确保变量构件编码的赋值与由v_ℓ^j 表示的 c_j 的满足赋值一致。

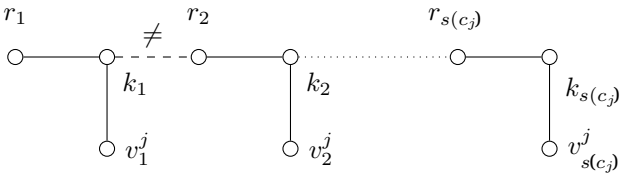


图 5 约束构件 ˆ C_j 当 $\Delta = 1$。

令图 G_0 表示包含所有变量构件 P_i 以及所有约束构件 ˆ C_j 的图，其中 $i \in [n]$ 和 $j \in [m]$。我们分别将 G_z 的块构件 ˆ$B_{i,j}$、变量构件 P_i 和约束构件 ˆ C_j 称为 ˆ$B_i^{G_z}{}_{,j}$、$P_i^{G_z}$ 和 ˆ$C_j^{G_z}$。为构造图 G，引入 G_0 的 $\kappa = n + 1$ 个副本 $G_1, \ldots, G_\kappa$，使它们按如下方式依次相连：对于 $i \in [n]$ 和 $j \in [\kappa - 1]$，ˆ$B_i^{G_j}{}_{,m}$ 的顶点 a' 就是 ˆ$B_i^{G_{j+1}}{}_{1}$, 的顶点 a。令 $\mathcal{P}^i$ 表示由 $P_i^{G_1}, \ldots, P_i^{G_\kappa}$ 形成的“路径”。

▶ **引理 22.** 对于任意 $\chi_{\mathrm{d}} \geq 2$，如果 ϕ 可满足，那么 G 存在一个 $(\chi_{\mathrm{d}}, 1)$-着色。

证明。 令 f： $X \to Y$表示一个满足所有约束 $c_1, \ldots, c_m$的赋值。我们将描述一个 $(\chi_{\mathrm{d}}, 1)$-着色 c： $V(G) \to [\chi_{\mathrm{d}}]$，其对应于 G。

令 $c(p_i^j)$ 其中 $p_i^j \in [\kappa]$，其中 $i \in [n]$ 和 $j \in [m]$，接下来点以下惯构件的顶点 ˆ $B_i^{G_z}$ (p_i^j) 其中 $p_i^j \in [\kappa]$，其中 $i \in [n]$ 和 $j \in [m]$，考虑以下惯构

1. 若 $f(x_i) = k$，对于 $k \in [\chi_{\mathrm{d}}]$，则令 $k' \in [\chi_{\mathrm{d}}] \setminus \{k\}$ 为任意一种颜色，并令$c(a) = c(x) = k$，而 $c(b) = c(y) = k'$，

2. 若 $f(x_i) = \chi_d + k$，对于 $k \in [\chi_d]$，则令 $k' \in [\chi_d] \setminus \{k\}$ 为任意一种颜色，并令$c(a) = c(y) = k$，而 $c(b) = c(x) = k'$。
关于约束小工具，令 c_j 是 ϕ的一个约束。由于 f 是一个满足赋值，对于某个 $z \in [\kappa]$，在 $\hat{} C_j^{G_z}$ 中，$v_1^j, \ldots, v_{s(j)}^j$ 中至少存在一个顶点映射到 f 在 c_j中出现的变量上的限制。令 v_ℓ^j 是其中索引最小的一个顶点。然后，将 $c(v_\ell^j) = 1$设为，而任何其他顶点 $v_{\ell'}^j$（具有 $\ell' \neq \ell$）都赋予颜色 2。此外，令 k_w 在 $w < \ell$ 时取颜色 1，在 $\ell \leq w \leq s(c_j)$时取颜色 2。另一方面，令 r_w 在 $w \leq \ell$ 时取颜色 2，在 $\ell < w \leq s(c_j)$时取颜色 1。

最后，使用引理12、13、15、17和19对相等关系/调色板/差异/排除/蕴- 含构件的内部顶点进行正确着色。要看出所有构件都使用这些引理进行了正确着色，请注意，迄今已着色的任何顶点至多有1个同色邻居，而以下几点成立：

- $P = \{p^i \mid i \in [\chi_d]\}$由 χ_d 个顶点组成，每个顶点均被赋予不同的颜色；对于所有$i \in [\chi_d]$，在所有块构件中，$c(p^i) = c(p_l^i)$，$c(a) = c(a')$，$c(a) \neq c(b)$，且顶点 x、y 的颜色为 $c(a)$或$c(b)$。对于 $z \in [\kappa]$ 和 $j \in [m]$，在约束构件$\hat{} C_j^{G_z}$中，有 $c(r_1) = c(p^2)$，$c(k_{s(c_j)}) = c(p^2)$，$c(k_w) \neq c(r_{w+1})$，对于 $w \in [s(c_j) - 1]$，顶点 k_w、r_w、v_w^j 对于 $w \in [s(c_j)]$ 的颜色为 $c(p^1)$或 $c(p^2)$。最后，如果 $c(v_\ell^j) = c(p^1)$，且 x_i 表示出现在c_j 中的一个变量，使得顶点 v_ℓ^j 对应于一种赋值，其中 x_i 取值为 $s \in [2\chi_d]$（这是赋值 f的限制），那么当 $s \leq \chi_d$时，$c(a) = s$ 且 $c(x) = s$；否则，$c(a) = s'$ 且 $c(x) \neq s'$，对于 $s' = s - \chi_d$，其中顶点 a、x 属于$\hat{} B_i^{G_z}$、j。
-
-

证明至此结束。 ◀

▶ **引理 23.** 对于任意 $\chi_d \geq 2$，如果 G 存在一个$(\chi_d, 1)$-着色，那么 ϕ 是可满足的。

证明。设 $c : V(G) \to [\chi_d]$ 是 G的一个（χ_d, 1）-着色。根据相等性构件的性质，对于所有 $i \in [\chi_d]$，有 $c(p^i) = c(p_l^i)$。由于 c 是一个（χ_d, 1）- 着色，因此对于不同的 i、$j \in [\chi_d]$，有 $c(p^i) \neq c(p^j)$。不失一般性，假设对于 $i \in [\chi_d]$，$c(p^i) = i$。

对于 G_z，考虑 $\hat{} B_i^{G_z}{}_{,j}$ 中顶点的着色与 x_i的取值之间的映射，其中变量 ϕ采用某个赋值。具体而言，当顶点 a 和 x都着色为颜色 $k \in [\chi_d]$ 时，该情况映射到满足 x_i 取值为 k 的赋值；而当只有a 着色为颜色 k时，则映射到满足 x_i 取值为 $\chi_d + k$的赋值。若变量构件 $P_i^{G_z}$中存在块构件 $\hat{} B_i^{G_z}{}_{,j}$ 和 $\hat{} B_i^{G_z}{}_{,j+1}$，使得每个块构件中顶点的着色映射到 x_i的不同赋值，我们就称其发生了不一致。对于每个 $i \in [n]$，如果 G_z 的变量构件 $P_i^{G_z}$中均未发生不一致，我们就称 G_z 是一致的。

▷ 断言 24. 存在 $\pi \in [\kappa]$，使得 G_π 是一致的。

证明。 我们将证明，每条路径 $\mathcal{P}^i$ 至多会导致 1 个不一致。在这种情况下，由于有 n 条这样的路径，以及 $\kappa = n + 1$ 个 G_0的副本，根据鸽巢原理，存在某个 G_π 没有任何不一致。

考虑一条路径 $\mathcal{P}^i$ 以及一个块构件 $\hat{} B_i^{G_z}{}_{,j}$，其中 $z \in [\kappa]$ 和 $j \in [m]$。令$N(\hat{} B_i^{G_z}{}_{,j})$ 表示 $\hat{} B_i^{G_z}{}_{,j}$ 右侧的块构件，即 $\hat{} B_i^{G_z}{}_{,j}$ 的顶点 a' 与 $N(\hat{} B_i^{G_z}{}_{,j})$ 的顶点 a重合。此外，令 $\hat{} B_i^{G_{z'}}{}_{,j'}$，其中要么 a) $z' = z$ 和 $j' > j$，要么 b) $z' > z$ 和 $j' \in [m]$，

表示出现在 $\hat{}B_i^{G_z},_j$右侧的某个块构件。对于每个块构件，根据相等构件的性质，有 $c(a) = c(a')$，因此，属于同一路径$\mathcal{P}^i$的所有块构件中，顶点a 的颜色都相同。关于 $\hat{}B_i^{G_z},_j$的顶点，考虑以下两种情况：

- 如果 $c(a) \neq c(x)$，则可推出 $c(a) = c(y)$，因为否则 b 将会有 2 个同色邻居。于是，可知 a'，即 $N(\hat{}B_i^{G_z},_j)$ 的顶点 a，在 $\hat{}B_i^{G_z},_j$ 中有一个同色邻居，因此对于 $N(\hat{}B_i^{G_z},_j)$ 的顶点 x，可推出 $c(x) \neq c(a)$；以此类推，对于所有块构件 $\hat{}B_i^{G_{z'}},_{j'}$，均可推出 $c(a) \neq c(x)$。如果 $c(a) = c(x)$，则可推出 $c(a) \neq c(y)$，因为 $c(x)$ 不可能有两个同色邻居。于是，可知 a'，即 $N(\hat{}B_i^{G_z},_j)$ 的顶点 a，在 $\hat{}B_i^{G_z},_j$ 中没有同色邻居。因此，对于 $N(\hat{}B_i^{G_z},_j)$ 的顶点 a 和 x，可推出以下两种情况之一：$c(a) = c(x)$，或 $c(a) \neq c(x)$。对于所有块构件 $\hat{}B_i^{G_{z'}},_{j'}$，情况同样如此。

因此可知，每条路径至多能导致 1 个不一致；由于总共有 n 条路径，因此存在一个一致的副本 G_π。◁

考虑如下赋值 $f : X \to Y$。令 a 和 x 表示块构件 $\hat{}B_i^{G_\pi},_j$ 的顶点，其中 $c(a) = k \in [\chi_{\mathrm{d}}]$。接着，若 $c(x) = k$，则令 $f(x_i) = k$；否则令 $f(x_i) = \chi_{\mathrm{d}} + k$。注意，对于每个块构件，上述两种情况必有一种成立，因此所有变量x_i 都已赋值，且 f 定义良好。

还需证明该赋值满足所有约束。考虑约束构件 $\hat{}C_j^{G_\pi}$，其中 $j \in [m]$。我们先证明，对于某个 $\ell \in [s(j)]$，有 $c(v_\ell^j) = 1$。假设并非如此。则可知，每个顶点 k_w 都有两个颜色为 2 的邻居，因此对于所有 $w \in [s(j)]$，有 $c(k_w) = 1$。然而，由于 $Q(p^2, k_{s(j)})$，可知 $c(k_{s(j)}) = 2$，这就产生了矛盾。令 v_ℓ^j 满足 $c(v_\ell^j) = 1$。在这种情况下，由于涉及 v_ℓ^j 的蕴含/排斥构件，可知若变量 x_i 参与约束 c_j，且 v_ℓ^j 对应于一种赋值，其中 x_i 的取值为 $s \in Y$，那么

(i) 若 $s = k$，其中 $k \in [\chi_{\mathrm{d}}]$，则 $c(a) = k = c(x)$，
(ii) 若 $s = \chi_{\mathrm{d}} + k$，其中 $k \in [\chi_{\mathrm{d}}]$，则 $c(a) = k \neq c(x)$，
其中，顶点 a 和 x 属于 $\hat{}B_i^{G_\pi},_j$。然而，在这种情况下，与 v_ℓ^j对应的赋值是 f的限制，因此 f满足约束 c_j。由于 $j = 1, \ldots, m$ 是任意的，这就证明了 f 是 ϕ的满足赋值。◀

▶ **引理 25**。 成立：路径宽度$(G) \leq n + O(1)$。

证明。 根据引理 20，有 $\mathsf{pw}(G) = \mathrm{pw}(G' - P) + 3\chi_{\mathrm{d}}$，其中 G'是通过从 G 中移除所有相等关系/颜色集/差异/排除/蕴含构件，并在它们的端点之间添加尚未连接的所有边而得到的图。因此，只需证明 $\mathsf{pw}(G' - P) = n + \mathcal{O}(1)$。

我们将通过提供一种混合搜索策略来清理 $G' - P$，同时至多使用这么多名搜索者。由于混合搜索数 ms 满足 $\mathsf{pw}(G' - P) \leq \mathsf{ms}(G' - P) \leq \mathsf{pw}(G' - P) + 1$，我们将证明 $\mathsf{ms}(G' - P) \leq n + 5 + B^q$，从而得到该结论。

从图 G_1开始。将 $s(c_1) + 1$ 个搜索者放置在 $\hat{}C_1^{G_1}$的顶点 r_1 和 v_w^1 上，并将 $w \in [s(c_1)]$，个搜索者以及 n 个搜索者分别放置在块构件 $\hat{}B_{i\,1}^{G_1}$ 的顶点 a 上，其中 $i \in [n]$。沿由 k_1、r_2、k_2、.....形成的路径移动放置在 r_1 上的搜索者，即可清除约束构件的所有边。接下来我们将描述清除 $\hat{}B_{i\,1}^{G_1}$的过程。将四个额外的搜索者移动到 $\hat{}B_i^{G_1}$、1 的所有其他顶点上，即 x、y、b、a'。之后，从顶点 a、x、y、b 上移除搜索者。对所有 $i \in [n]$ 重复整个过程。

为了清除图的其余部分，我们首先将搜索者从 $\hat{}C_j^{G_z}$ 移动到 $\hat{}C_{j+1}^{G_z}$（如果 $j < m$ ），或者移动到 $\hat{}C_1^{G_{z+1}}$（作为替代方案）（必要时可能引入新的搜索者），清除后者，然后继续清除相应的块构件。重复此过程，最终我们使用至多 $n+5+B^q = n+\mathcal{O}(1)$ 个搜索者清除 $G'-P$ 的所有边。◀

因此，根据引理 25，我们可以在多项式时间内构造一个图 G，其路径宽度为 $\mathrm{pw}(G) \leq n+\mathcal{O}(1)$；根据引理 22 和 23，判断 G 是否存在 $(\chi_{\mathrm{d}},1)$-缺陷着色，等价于判断 ϕ 是否可满足。在这种情况下，假设存在一个用于 χ_{d} 种颜色和 $\Delta=1$ 的缺陷着色问题的 $\mathcal{O}^*((2\chi_{\mathrm{d}}-\varepsilon)^{\mathrm{pw}(G)})$ 算法，那么对于 $B=2\chi_{\mathrm{d}}$，我们可以在 $\mathcal{O}^*((2\chi_{\mathrm{d}}-\varepsilon)^{\mathrm{pw}(G)}) = \mathcal{O}^*((B-\varepsilon)^{n+\mathcal{O}(1)}) = \mathcal{O}^*((B-\varepsilon)^n)$ 时间内判定 q-CSP-B，这与定理 1 中的强指数时间假设相矛盾。◀

▶**定理 26.** 对于任意常数 $\varepsilon > 0$ 以及任意固定的 $\chi_{\mathrm{d}} \geq 2$, $\Delta \geq 2$，除非强指数时间假设不成立，否则不存在一个 $O^*((\chi_{\mathrm{d}} \cdot (\Delta+1) - \varepsilon)^{\mathrm{pw}})$ 算法，能够判定 G 是否存在 $(\chi_{\mathrm{d}}, \Delta)$-着色，其中 pw 表示图的路径宽度。

证明。固定一个正的 $\varepsilon > 0$，我们希望证明该定理。令 $B = \chi_{\mathrm{d}} \cdot (\Delta+1)$。我们将把 q-约束满足问题-B归约为最大度为 $\Delta \geq 2$ 且使用 χ_{d} 种颜色的缺陷着色，其中 q 是一个仅取决于 ε的常数，并确保：如果所得的缺陷着色实例可以在 $\mathrm{O}^*((\chi_{\mathrm{d}} \cdot (\Delta+1) - \varepsilon)^{\mathrm{pw}})$ 时间内求解，那么我们将得到一个 q-约束满足问题-B 算法，而根据定理 1，该算法将与强指数时间假设相矛盾。为此，令 ϕ 为一个 q-约束满足问题-B 实例，它包含 n 个变量 $X = \{x_i \mid i \in [n]\}$，这些变量取值于集合 $Y = [0, B-1]$，并具有 m 个约束 $C = \{c_j \mid j \in [m]\}$。对于每个约束，我们给出一个至多包含 q 个变量的集合，这些变量参与该约束；同时给出这些变量的满足赋值列表，其大小记为 $s : C \to [B^q]$，即 $s(c_j) \leq B^q = \mathrm{O}(1)$表示约束 c_j的满足赋值数量。我们将在多项式时间内构造一个等价实例 G，它是一个使用$\Delta \geq 2$ 和 χ_{d} 种颜色的缺陷着色实例，并且其路径宽度$(G) \leq n + \mathrm{O}(1)$。

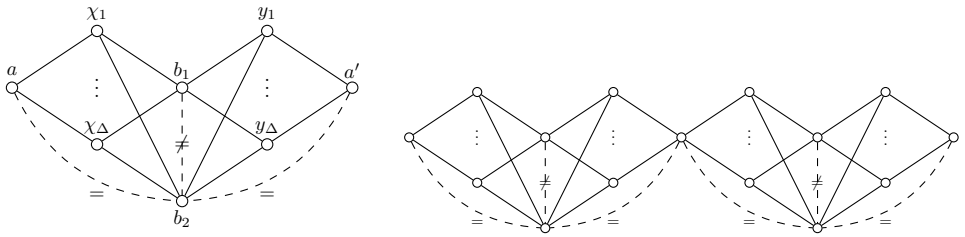
由于我们将反复使用相等关系构件、差异构件和调色板构件（见第 3.2.1 节），因此采用以下约定：对于我们已向 G 引入的顶点 v_1、v_2、v_3，当我们说添加一个相等构件 $Q(v_1, v_2)$、一个差异构件 $D(v_1, v_2, \delta)$ 或一个调色板构件 $P(v_1, v_2, v_3)$ 时，意指向 G 添加一个 $Q(u_1, u_2, \chi_{\mathrm{d}}, \Delta)$、$D(u_1, u_2, \chi_{\mathrm{d}}, \Delta, \delta)$ 或 $P(u_1, u_2, u_3, \chi_{\mathrm{d}}, \Delta)$ 的副本，并分别将 u_1、u_2(u_3) 与 v_1、v_2(v_3) 识别为同一对象。

调色板顶点。构造一个由 χ_{d} 个顶点组成的团 $P = \{p^i \mid i \in [\chi_{\mathrm{d}}]\}$。对于 $i \in [\chi_{\mathrm{d}}]$，在顶点 p^i 上连接 p_l^i 个叶节点；对于 $l \in [\Delta]$，添加相等性构件 $Q(p^i, p_l^i)$。

每当 v_1, v_2 是我们已经为 G 引入的顶点时，当我们说添加一个排斥构件 $E(v_1, v_2, v_1', v_2')$ 或蕴含构件 $I(v_1, v_2, v_1', v_2')$，这意味着我们向 G 添加一个 $E(u_1, u_2, c_1, c_2, C, \chi_{\mathrm{d}}, \Delta)$ 或 $I(u_1, u_2, c_1, c_2, C, \chi_{\mathrm{d}}, \Delta)$ 的副本，然后分别将 u_1, u_2, c_i 与 v_1, v_2, p^i 识别，对于所有 $i \in [\chi_{\mathrm{d}}]$。

块与变量构件。对于每个变量 x_i 和每个约束 c_j，如图6a所示，构造一个块构件 $\hat{}B_{i,j}$。为此，我们引入顶点 a, a', b_1, b_2, χ_i和 y_i，其中 $i \in [\Delta]$。然后，添加构件 $Q(a, b_2)$、$Q(b_2, a')$ 和 $D(b_1, b_2, \Delta+1)$。最后，添加边 $\{a, \chi_i\}$、$\{a', y_i\}$，以及连接顶点 b_1, b_2 与每个顶点 χ_i, y_i的边。此外，如果 $\chi_{\mathrm{d}} \geq 3$，则对所有 $i \in [\Delta]$添加调色板构件 $P(b_1, b_2, \chi_i)$和 $P(b_1, b_2, y_i)$。接下来，对于$j \in [m-1]$，我们将块构件 $\hat{}B_{i,j}$ 和 $\hat{}B_{i,j+1}$ 串联起来，使得顶点 a' 属于

ˆ ˆ$B_{i,\ j}$是$B_{i,\ j+1}$的顶点 a，从而得到 n条“路径” P_1，.....，P_n，每条路径由 m个串联的块构件组成，称为变量构件。直观而言，变量构件旨在表示一个变量 x_i，因此需要具有 χ_{d}（$\Delta+1$）种不同的可行配置。这些配置通过先为 a 选择一种颜色（有χ_{d} 种选择），再决定 $\{\chi_i \mid i \in [\Delta]\}$ 中有多少个顶点将获得与 a 相同的颜色（有$\Delta+1$ 种选择）来构成。



(a) $\Delta \geq 2$ 时的块构件。 **(b)** 串联的块构件。

图6 变量构件由串联的块构件组成。

约束构件。 该构件负责根据图中其他部分的选择来确定约束满足情况。对于约束 c_j，构造约束 ˆ 构件 C_j，如图 7 所示：

- 引入顶点 r_w，其中 $w \in [s(c_j)]$,并添加 $Q(p^2, r_1)$，以及在$\chi_{\mathrm{d}} \geq 3$时添加 $P(p^1, p^2, r_w)$，
- 对于 $w \in [2, s(j)]$对于 $w \in [s(j)]$,引入顶点 v_w^j，以及调色板构件 $P(p^1, p^2, v_w^j)$，当
- $\chi_{\mathrm{d}} \geq 3$时，并在这些顶点与 c_j 的满足赋值之间固定一个任意的一一映射；引入顶点 k_w，
- 其中 $w \in [s(c_j)]$,并添加构件 $D(p^2, k_w, \Delta-1)$、 $Q(p^2, k_{s(c_j)})$，以及当 $\chi_{\mathrm{d}} \geq 3$ 时添加 P
- (p^1, p^2, k_w)。对于 $w \in [s(j)\ -1]$，在顶点 k_w 与顶点 r_w、v_w^j 之间添加边，以及添加 D
- $(k_w, r_{w+1}, \Delta+1)$；如果变量 x_i 涉及约束 c_j，并且 v_ℓ^j 对应于一种赋值，其中 x_i 的
- 值为 $s \in Y$，其中 $s = (\Delta+1)\cdot(k-1)\ +\delta$，对于 $k \in [\chi_{\mathrm{d}}]$ 和 $\delta \in[0, \Delta]$，则添加
- 蕴含构件 $I(v_\ell^j, a, p^1, p^k)$、 $I(v_\ell^j, \chi_{i_1}, p^1, p^k)$ 和排除构造块$E(v_\ell^j, \chi_{i_2}, p^1, p^k)$；对于 $i_1 \in [\delta]$
- 和 $i_2 \in [\delta+1, \Delta]$，其中顶点 a 和 χ 属于 ˆ$B_{i,j}$。

直观地说，约束构件的构造方式确保，对于某个 $\ell \in [s(c_j)]$, 顶点v_ℓ^j，其颜色为 **1**，进而“激活”我们添加到该顶点上的蕴含构件和排斥构件。这确保变量构件所编码的赋值与 c_j 由 v_ℓ^j 表示的满足赋值一致。

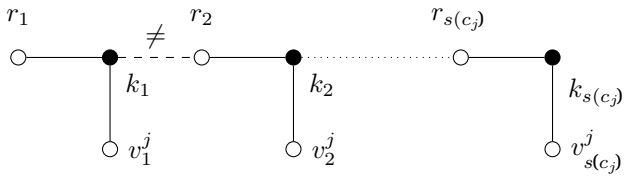


图7 约束构件。对于每个黑色顶点 k_w，都存在一个构件 $D(p^2, k_w, \Delta-1)$。

令图 G_0 对应于包含所有变量构件 P_i 以及所有约束构件 ˆ C_j（对于 $i \in [n]$ 和 $j \in [m]$）的图。我们分别将块构件 ˆ$B_{i,j}$、变量构件 P_i以及 G_z 的约束构件 ˆ C_j 称为 ˆ$B_i^{G_z}{}_{,j}$、 $P_i^{G_z}$ 和 ˆ$C_j^{G_z}$。

为构造图 G，引入 $\kappa = n \cdot \Delta + 1$ 个 G_0的副本 $G_1, \ldots, G_\kappa$，并按如下方式依次连接它们：对于 $i \in [n]$ 和 $j \in [\kappa - 1]$，$\hat{}B_i^{G_j}{}_{,m}$ 的顶点 a' 就是 $\hat{}B_{i\,1}^{G_{j+1}}{}_{,}$的顶点 a。令 $\mathcal{P}^i$ 表示由 $P_i^{G_1}$, $\ldots, P_i^{G_\kappa}$ 产生的“路径”。

▶ **引理 27.** 对于任意 $\chi_\mathrm{d} \geq 2$，如果 ϕ 可满足，那么 G 存在一种$(\chi_\mathrm{d}, \Delta)$着色。

证明。 令 $f\colon X \to Y$表示一个满足所有约束 $c_1, \ldots, c_m$的赋值。我们将描述一种 $(\chi_\mathrm{d}, \Delta)$着色 $c\colon V(G) \to [\chi_\mathrm{d}]$ 在 G上。

令 $c(p^i) = c(p_l^i) = i$，其中 $i \in [\chi_\mathrm{d}]$ 和 $l \in [\Delta]$。接下来，对于块构件 $\hat{}B_i^{G_z}{}_{,j}$ 的顶点，其中 $z \in [\kappa], i \in [n]$ 和 $j \in [m]$，若 $f(x_i) = (\Delta + 1) \cdot (k - 1) + \delta$，其中 $k \in [\chi_\mathrm{d}]$ 和 $\delta \in [0, \Delta]$，则令

- $c(a) = c(\chi_{i_1}) = c(y_{i_2}) = k$，其中 $i_1 \in [\delta]$ 和 $i_2 \in [\delta + 1, \Delta]$,$c(b) = c(\chi_{i_1}) = c(y_{i_2}) = k'$，
- 对于某个任意的 $k' \in [\chi_\mathrm{d}] \setminus \{k\}$，其中 $i_1 \in [\delta + 1, \Delta]$ 和$i_2 \in [\delta]$。

关于约束构件，令 c_j 为 ϕ 的一个约束。令连接到顶点 k_w 的 $\Delta - 1$ 个叶节点着色为 2，对于 $w \in [s(c_j)]$。由于 f 是一个满足赋值，在 $\hat{}C_j^{G_z}$ 中的 v_1^j、.....、$v_{s(j)}^j$ 中，至少存在一个顶点 $z \in [\kappa]$ 被映射到 f 的一个限制。令 v_ℓ^j 为其中索引最小的一个顶点。接着，令 $c(v_\ell^j) = 1$，而其他任何满足 $\ell' \neq \ell$ 的顶点 $v_{\ell'}^j$ 均着色为 2。此外，对于 $w < \ell$，令 k_w 着色为 1；对于 $\ell \leq w \leq s(c_j)$，令其着色为 2。另一方面，对于 $w \leq \ell$，令 r_w 着色为 2；对于 $\ell < w \leq s(c_j)$，令其着色为 1。

最后，使用引理12、13、15、17和19，对相等关系/调色板/差异/排斥/蕴含构件的内部顶点进行正确着色。为说明使用这些引理可以使所有构件都得到正确着色，请注意，到目前为止着色的任何顶点至多有 Δ 个同色邻居，而以下条件成立：

- $P = \{p^i \mid i \in [\chi_\mathrm{d}]\}$由 χ_d 个顶点组成，每个顶点接收一种不同的颜色，并且对于所有$i \in [\chi_\mathrm{d}]$, $l \in [\Delta]$，$c(p^i) = c(p_l^i)$，在所有块构件中，$c(a) = c(b_2)$，$c(b_2) = c(a')$，$c(b_1) \neq c(b_2)$，而顶点 χ_i、y_i 对于$i \in [\Delta]$ 的颜色为 $c(b_1)$ 或 $c(b_2)$ 中的一种；对于 $z \in [\kappa]$ 和 $j \in [m]$，在约束构件$\hat{}C_j^{G_z}$中，有 $c(r_1) = c(p^2)$，$c(k_{s(c_j)}) = c(p^2)$，$c(k_w) \neq c(r_{w+1})$ 对于 $w \in [s(c_j) - 1]$，顶点 k_w、r_w、v_w^j 对于 $w \in [s(c_j)]$ 的颜色为 $c(p^1)$ 或 $c(p^2)$ 中的一种，而连接到 k_w 的叶节点均接收颜色 $c(p^2)$；最后，如果$c(v_\ell^j) = c(p^1)$，且 x_i 表示出现在 c_j 中的一个变量，使得顶点 v_ℓ^j 对应于一种赋值，其中 x_i 取值为 $s = (\Delta + 1) \cdot (k - 1) + \delta$，对于 $k \in [\chi_\mathrm{d}]$ 和 $\delta \in [0, \Delta]$（这是赋值 f的一个限制），则对于 $i_1 \in [\delta]$和 $i_2 \in [\delta + 1, \Delta]$，$c(a) = c(\chi_{i_1}) = k$ 且 $c(\chi_{i_2}) \neq k$，其中顶点 a、χ_{i_1}、χ_{i_2}属于$\hat{}B_i^{G_z}$、j。

证明到此结束。 ◀

▶ **引理 28.** 对于任意 $\chi_\mathrm{d} \geq 2$，如果 G 存在一个$(\chi_\mathrm{d}, \Delta)$-着色，那么 ϕ 是可满足的。

证明。令 $c\colon V(G) \to [\chi_\mathrm{d}]$ 为 G的一个$(\chi_\mathrm{d}, \Delta)$-着色。根据相等性构件的性质，对于所有 $i \in [\chi_\mathrm{d}], l \in [\Delta]$，有 $c(p^i) = c(p_l^i)$。由于 c是一个$(\chi_\mathrm{d}, \Delta)$-着色，因此对于不同的 $i, j \in [\chi_\mathrm{d}]$，有 $c(p^i) \neq c(p^j)$。不失一般性，假设对于 $i \in [\chi_\mathrm{d}]$，$c(p^i) = i$。

对于 G_z，考虑$\hat{}B_i^{G_z}{}_{,j}$ 的顶点着色与x_i 的值之间的映射，其中涉及对 ϕ的变量进行某种赋值。特别地，顶点 a 以及$\{\chi_i \mid i \in [\Delta]\}$ 中颜色为 $k \in [\chi_\mathrm{d}]$ 的 $\delta \in [0, \Delta]$ 个顶点的着色，被映射到一种赋值，其中 x_i 取值为 $(\Delta + 1) \cdot (k - 1) + \delta$。如果变量构件 $P_i^{G_z}$中存在块构件$\hat{}B_i^{G_z}{}_{,j}$ 和$\hat{}B_i^{G_z}{}_{,j+1}$，使得每个块构件的顶点着色

26 重访两个有界度问题的结构参数化

分别映射到 x_i的不同赋值，我们称其中发生了不一致。对于每个 $i \in [n]$，如果其变量构件 $P_i^{G_z}$ 中均未发生不一致，则称 G_z 是一致的。

注意，对于块构件 $\hat{}B_i^{G_z}{}_{,j}$ 中的顶点，顶点 χ_{i_1}，y_{i_2}，对于i_1，$i_2 \in [\Delta]$，其颜色为 $c(a)$ 或 $c(b_1)$，并且由于这类顶点总共有 2Δ 个，因此两种颜色各占一半；否则，b_1 或 b_2 的同色邻居就会多于 Δ 。现在我们来证明以下断言。

▷ 断言 29. 存在 $\pi \in [\kappa]$，使得 G_π 是一致的。

证明。 我们将证明，每条路径 $\mathcal{P}^i$ 至多会导致 Δ 个不一致。在这种情况下，由于共有 n 条这样的路径，且有 $\kappa = n \cdot \Delta + 1$ 个 G_0副本，根据鸽巢原理，存在某个 G_π 不存在任何不一致。

考虑一条路径 P^i 以及一个块构件 $\hat{}B_i^{G_z}{}_{,j}$，其中 $z \in [\kappa]$ 和 $j \in [m]$。令$N(\hat{}B_i^{G_z}{}_{,j})$ 表示位于 $\hat{}B_i^{G_z}{}_{,j}$ 右侧的块构件，即 $\hat{}B_i^{G_z}{}_{,j}$ 的顶点 a' 与 $N(\hat{}B_i^{G_z}{}_{,j})$的顶点 a重合。此外，令 $\hat{}B_i^{G_{z'}}{}_{,j'}$ 表示位于 $\hat{}B_i^{G_z}{}_{,j}$右侧的某个块构件，其中或者 a) $z' = z$ 和 $j' > j$，或者 b) $z' > z$ 和 $j' \in [m]$。对于每个块构件，根据相等构件的性质，有 $c(a) = c(a')$，因此，属于同一路径 P^i的所有块构件中，顶点a 的颜色都相同。对于 $\hat{}B_i^{G_z}{}_{,j}$中的顶点，假设恰有 δ 个顶点 χ_{i_1} 被染成颜色 $c(a)$。我们将证明，在这种情况下，每个构件 $\hat{}B_i^{G_{z'}}{}_{,j'}$中，至多有 δ 个顶点 χ_{i_1} 被染成颜色 $c(a)$。作为归纳基础，注意在 $\hat{}B_i^{G_z}{}_{,j}$ 中，恰有 $\Delta - \delta$ 个顶点 y_{i_2} 被染成颜色 $c(a)$。因此，在 $N(\hat{}B_i^{G_z}{}_{,j})$中，至多有 δ 个顶点 χ_{i_1} 获得颜色 $c(a)$，因为 $\hat{}B_i^{G_z}{}_{,j}$ 的顶点 a' 与 $N(\hat{}B_i^{G_z}{}_{,j})$的顶点 a 重合。假设对于 $\hat{}B_i^{G_z}{}_{,j}$ 右侧的某个构件 $\hat{}B_i^{G_{z'}}{}_{,j'}$，这一结论成立。那么，由于 $\hat{}B_i^{G_z}{}_{,j}$ 中至少有 $\Delta - \delta$ 个顶点 y_{i_2} 获得颜色 $c(a)$，可知 $N(\hat{}B_i^{G_{z'}}{}_{,j'})$中至多有 δ 个顶点 χ_{i_1} 获得颜色 $c(a)$。因此，每条路径至多会导致 Δ 个不一致；又由于路径总数为 n，故存在一个一致的副本 G_π 。◁

令 $f\colon X \to Y$为一个赋值，使得 $f(x_i) = (\Delta + 1) \cdot (k - 1) + \delta$；其中，对于块构件 $\hat{}B_i^{G_\pi}{}_{,j}$ 中的顶点，有 $c(a) = k \in [\chi_{\mathrm{d}}]$，且 $\{\chi_i \mid i \in [\Delta]\}$ 中恰有 $\delta \in [0, \Delta]$ 个顶点的颜色为 k。

还需证明这是一个满足所有约束的赋值。考虑约束构件 $\hat{}C_j^{G_\pi}$，其中 $j \in [m]$。我们首先证明，对于某个 $\ell \in [s(j)]$，有 $c(v_\ell^j) = 1$。假设情况并非如此。由此可知，每个顶点 k_w 都有 $\Delta + 1$ 个颜色为 2 的邻居（记住，由于 $D(p^2, k_i, \Delta - 1)$，k_w 有 $\Delta - 1$ 个颜色为$c(p^2)$ 的相邻叶节点），因此，对于每个 $w \in [s(j)]$，都有 $c(k_w) \neq 2$。然而，由于 $Q(p^2, k_{s(j)})$，可知 $c(k_{s(j)}) = 2$，这就产生了矛盾。令 v_ℓ^j满足 $c(v_\ell^j) = 1$。在这种情况下，由于涉及 v_ℓ^j 的蕴含构件和排斥构件可知，若变量 x_i参与约束 c_j，且 v_ℓ^j 对应于一个赋值，在该赋值下，x_i 对于 $k \in [\chi_{\mathrm{d}}]$ 和 $\delta \in [0, \Delta]$取值 $(\Delta + 1) \cdot (k - 1) + \delta$，那么在 $\hat{}B_i^{G_\pi}{}_{,j}$ 中，顶点 a 以及 $\{\chi_i \mid i \in [\Delta]\}$中恰好 δ 个顶点的颜色为 k。在这种情况下，对应于 v_ℓ^j 的赋值是 f的一个限制，因此 f满足约束 c_j。由于 $j = 1, \ldots, m$ 是任意的，这就证明了 f 是 ϕ的一个满足赋值。◀

▶ **引理 30.** 成立：路径宽度$(G) \leq n + O(1)$。

证明。 根据引理 20，有 $\mathrm{pw}(G) = \mathrm{pw}(G' - P) + 3\chi_{\mathrm{d}}$，其中 G'是通过移除 G 中的所有相等关系/调色板/差异/排除/蕴含构件

并在它们的端点之间添加所有尚未连接的边。因此，只需证明 $\mathrm{pw}(G'-P) = n + \mathcal{O}(1)$。

我们将通过提供一种混合搜索策略来清理 $G'-P$，并同时使用至多这么多名搜索者。由于混合搜索数 ms 满足 $\mathrm{pw}(G'-P) \leq \mathrm{ms}(G'-P) \leq \mathrm{pw}(G'-P)+1$，我们将证明 $\mathrm{ms}(G'-P) \leq n+5+B^q$，由此即可得出结论。

从图 G_1 开始。将 $s(c_1)+1$ 名搜索者放置在 $\hat{C}_1^{G_1}$ 的顶点 r_1 和 v_w^1 上；对于 $w \in [s(c_1)]$, ，也将 n 名搜索者放置在块构件 $\hat{B}_{i\,1}^{G_1}$ 的顶点 a 上，其中 $i \in [n]$。沿着由 k_1、r_2、k_2、.....构成的路径移动放置在 r_1 上的搜索者，即可清理约束构件的所有边。接下来，我们将描述清理 $\hat{B}_{i\,1}^{G_1}$ 的步骤。将四名额外的搜索者移至 $\hat{B}_i^{G_1}$,1 的顶点 a'、b_1、b_2、χ_1。将后面那名搜索者移至所有其他顶点 χ_p 和 y_p，从而成功清理 $\hat{B}_{i\,1}^{G_1}$。最后，从 $\hat{B}_i^{G_1}$，1 上移走所有搜索者，只留下位于顶点 a' 上的一名搜索者。对所有 $i \in [n]$ 重复整个步骤。

为了清理图的其余部分，我们先将搜索者从 $\hat{C}_j^{G_z}$ 移至 $\hat{C}_{j+1}^{G_z}$（如果是 $j<m$），或者改为移至 $\hat{C}_1^{G_{z+1}}$（必要时可能需要引入新的搜索者），清理后者，然后继续清理相应的块构件。重复这一过程，最终我们至多使用 $n+5+B^q = n+\mathcal{O}(1)$ 个搜索者，清理完 $G'-P$ 的所有边。◀

因此，我们可以在多项式时间内构造一个图 G，其路径宽度为 $\mathrm{pw}(G) \leq n+\mathcal{O}(1)$（由引理 30 得出）；根据引理27和引理28，判定 G 是否存在一个 $(\chi_\mathrm{d},\ \Delta)$-着色，等价于判定 ϕ 是否可满足。在这种情况下，假设存在一个用于缺陷着色的$\mathcal{O}^\star((\chi_\mathrm{d}\cdot(\Delta+1)-\varepsilon)^{\mathrm{pw}(G)})$ 算法，那么对于 $B=\chi_\mathrm{d}\cdot(\Delta+1)$，我们可以在$\mathcal{O}^\star((\chi_\mathrm{d}\cdot(\Delta+1)-\varepsilon)^{\mathrm{pw}(G)}) = \mathcal{O}^\star((B-\varepsilon)^{n+\mathcal{O}(1)}) = \mathcal{O}^\star((B-\varepsilon)^n)$ 时间内判定 q-约束满足问题-B，这与强指数时间假设相矛盾（由定理 1 得出）。◀

3.2.3 算法

这里我们提出一种用于缺陷着色的算法，其参数由输入图的树宽与目标度 Δ 共同确定。该算法采用标准技术，并严格遵循此前在 [5, 6]的（$\chi_\mathrm{d}\Delta$）$^{\mathcal{O}(\mathrm{tw})}n^{\mathcal{O}(1)}$ 算法中概述的方法。其创新之处在于采用了 [17] 中提出的卷积技术，以加快合并节点情况下的计算速度。

▶ **定理 31**。 给定缺陷着色的一个实例 $I=(G,\chi_\mathrm{d},\Delta)$，以及一个宽度为tw的 G 良好树分解，存在一种算法，可在 $O^\star((\chi_\mathrm{d}\cdot(\Delta+1))^{\mathrm{tw}})$时间内判定 I。

证明。 为避免混淆，我们将良好树分解的顶点称为节点，将 G 的顶点称为顶点。令 $\mathcal{T}=(T,\{X_t\}_{t\in V(T)})$ 表示 G 的良好树分解，其中 r 表示根节点。对于 T 的一个节点 t，令 $X_t^\downarrow$ 表示以 t 为根的子树中所有袋的并集，包括 X_t。此外，令 $s\colon V(T)\to[\mathrm{tw}+1]$，使得 $s(t)=|X_t|$，即 $s(t)$ 表示袋 X_t 的大小，并假设 $X_t=\{v_1^t,\ldots,v_{s(t)}^t\}$。

假设存在 G 的 $(\chi_\mathrm{d},\ \Delta)$-着色，那么对于每个顶点 $v\in V(G)$，都有 $\chi_\mathrm{d}\cdot(\Delta+1)$ 种不同的可能性，因为它属于相应子图中某个度为 $\delta\in[0,\Delta]$ 的 χ_d 颜色类。因此，对于 T 的每个节点 t，我们考虑元组 $z_i^t=(w^{t,\ 1}{}^i$、.....、$w^{t,\ |X_t|}{}^i)$，其中每个 $w^{t,\ j}{}^i$ 都是一对 $w^{t,\ j}{}^i=(c^{t,\ j}{}^i,\ \delta^{t,\ j}{}^i)$，满足 $c^{t,\ j}{}^i\in[\chi_\mathrm{d}]$，并且

$\delta^{t,\ j}{}^{\ i} \in [0, \Delta]$，用于编码顶点 v_j^t 的这项信息；因此每个节点 t 对应 $(\chi_{\mathrm{d}} \cdot (\Delta+1))^{s(t)}$ 个元组。

在这种情况下，对于节点 t，令 $S[t, z_i^t]$，其中 $i \in [(\chi_{\mathrm{d}} \cdot (\Delta+1))^{s(t)}]$，表示图 $G[X_t^{\downarrow}]$ 的 $(\chi_{\mathrm{d}}, \Delta)$-着色数，其中顶点 v_j^t 被赋予颜色 $c^{t,}{}_{j}{}^{i}$，且在 $X_t^{\downarrow} \setminus X_t$ 中恰有 $\delta^{t,}{}_{j}{}^{i}$ 个同色邻居。如果 $s(t) = 0$，那么令 $S[t, \emptyset]$ 等于图 $G[X_t^{\downarrow}]$ 的 $(\chi_{\mathrm{d}}, \Delta)$-着色数。于是，为了求出 G 的 $(\chi_{\mathrm{d}}, \Delta)$-着色数，只需计算 $S[r, \emptyset]$。

注意，每个这样的元组 z_i^t 都会诱导出 χ_{d} 个同色顶点集合 $V^{t,}{}_{c}{}^{i} = \{v_j^t \in X_t \mid c^{t,}{}_{j}{}^{i} = c\}$，其中 $c \in [\chi_{\mathrm{d}}]$。要考虑元组 z_i^t，必须满足：

$$\forall v_j^t \in X_t, v_j^t \in V_c^{t,i} \implies |N(v_j^t) \cap V_c^{t,i}| + \delta_j^{t,i} \le \Delta, \tag{1}$$

换句话说，如果某个顶点在子图中有 δ 个同色邻接顶点不属于该袋，那么它在袋内最多应有 $\Delta - \delta$ 个同色邻居。

叶节点。如果节点 t 是叶节点，那么 $X_t = \{v_1^t\}$ 并且

$$S[t, (c_1^{t,i}, \delta_1^{t,i})] = \begin{cases} 1, & \text{if } \delta_1^{t,i} = 0, \\ 0, & \text{otherwise} \end{cases}$$

由于无论顶点 v_1^t 被赋予什么颜色，$X_t^{\downarrow} \setminus X_t$ 都是空的，因此它不可能有任何同色邻居。

引入节点。设 t 是一个引入节点，其子节点为 t_1，满足 $X_t = X_{t_1} \cup \{v_{s(t)}^t\}$，对于 $v_{s(t)}^t \notin X_{t_1}$，其中对于 $j \in [s(t_1)]$，，顶点 v_j^t 和 $v_j^{t_1}$ 重合。考虑节点 t 的一个元组 $z_i^t = (w^{t,}{}_{1}{}^{i}, \ldots, w^{t,}{}_{s\ t()}{}^{i})$。我们将计算其值 $S[t, z_i^t]$。首先，我们验证方程 (1) 是否成立（若不成立，可将答案设为 0）。然后，

$$S[t, z_i^t] = \begin{cases} S[t_1, z_{i_1}^{t_1}], & \text{if } \delta_{s(t)}^{t,i} = 0, \\ 0, & \text{otherwise,} \end{cases}$$

其中 $z_{i_1}^{t_1} = (w^{t_1,}{}_{1}{}^{i_1}, \ldots\ldots, w^{t_1,}{}_{s\ t_1()}{}^{i_1})$ 是节点 t_1 的元组，其中 $c^{t,}{}_{j}{}^{i} = c^{t_1,}{}_{j}{}^{i_1}$ 和 $\delta^{t,}{}_{j}{}^{i} = \delta^{t_1,}{}_{j}{}^{i_1}$，对所有 $j \in [s(t_1)]$ 均成立。直观地说，顶点 $v_{s(t)}^t$ 在 $X_t^{\downarrow} \setminus X_t$ 中不能有任何邻居，因此 $\delta^{t,}{}_{s\ t()}{}^{i}$ 唯一有效的取值是 0。

遗忘节点。假设 t 是一个遗忘节点，其子节点为 t_1，且对于 $v_{s(t_1)}^{t_1} \in X_{t_1}$，有 $X_t = X_{t_1} \setminus \{v_{s(t_1)}^{t_1}\}$；其中，对于 $j \in [s(t)]$，，顶点 v_j^t 和 $v_j^{t_1}$ 重合。考虑节点 t 的一个元组 $z_i^t = (w^{t,}{}_{1}{}^{i}, \ldots\ldots, w^{t,}{}_{s\ t()}{}^{i})$。我们将计算其值 $S[t, z_i^t]$。首先，我们验证方程 (1) 是否成立（如果不成立，可以将答案设为 0）。然后，

$$S[t, z_i^t] = \sum_{i_1 \in \mathcal{R}_1(i)} S[t_1, z_{i_1}^{t_1}],$$

其中 $i_1 \in \mathcal{R}_1(i)$，当且仅当对所有 $j \in [s(t)]$，

- $c_j^{t,i} = c_j^{t_1,i_1}$ and
- 如果 $v_{s(t_1)}^{t_1} \in N(v_j^{t_1})$ 且 $c^{t_1,}{}_{s\ t_1()}{}^{i_1} = c^{t_1,}{}_{j}{}^{i_1}$，那么 $\delta^{t,}{}_{j}{}^{i} = \delta^{t_1,}{}_{j}{}^{i_1} + 1$；否则 $\delta^{t,}{}_{j}{}^{i} = \delta^{t_1,}{}_{j}{}^{i_1}$。

在这种情况下，我们考虑关于被遗忘顶点的所有可能性，并考虑它对其余顶点的影响；如果它是袋中另一个顶点的同色邻居，那么后者在子树中、袋之外的同色邻居数量就增加了一个。

连接节点。 假设 t 是一个连接节点，其子节点为 t_1, t_2，且满足 $X_t = X_{t_1} = X_{t_2}$，其中 $j \in [s(t)]$, 顶点 $v_j^t, v_j^{t_1}$和 $v_j^{t_2}$重合。考虑节点 t的一个元组 $z_i^t = (w^{t,1}_{\ i}, \ldots, w^{t,s}_{\ i}{}^{t()})$。我们将计算值 $S[t, z_i^t]$。首先，我们验证方程 (1) 是否成立（若不成立，则答案可设为 0）。然后，

$$S[t, z_i^t] = \sum_{(i_1,i_2)\in\mathcal{R}_2(i)} S[t_1, z_{i_1}^{t_1}] \cdot S[t_2, z_{i_2}^{t_2}],$$

其中 $(i_1, i_2) \in \mathcal{R}_2(i)$，当且仅当对所有 $j \in [s(t)]$,

- $c_j^{t,i} = c_j^{t_1,i_1} = c_j^{t_2,i_2}$ and
- $\delta^{t_1,i_1}_{j} + \delta^{t_2,i_2}_{j} = \delta^{t,i}_{j}$, 其中 $0 \le \delta^{t_1,i_1}_{j}, \delta^{t_2,i_2}_{j} \le \delta^{t,i}_{j}$.

直观地，在连接节点中，对于袋中的每个顶点，我们都应考虑其在两个子树中的同色邻居（不包括袋中的顶点），并考虑这些邻居在这些子树中的所有可能划分方式。

请注意，表 S 最多有 $(\chi_{\mathrm{d}} \cdot (\Delta + 1))^{\mathrm{tw}+1}$ 个单元格。此外，通过采用动态规划，我们可以使用自底向上方法填充所有单元格。为了检查方程 (1)，需要 $n^{\mathcal{O}(1)}$ 时间。随后，叶节点或引入节点的每个元组都可以在常数时间内计算，而遗忘节点的每个元组需要 $\chi_{\mathrm{d}} \cdot (\Delta + 1)$ 时间。不过，对于合并节点，每个元组所需的时间为 $\mathcal{O}((\Delta + 1)^{2\mathrm{tw}})$，因为我们需要考虑每个顶点在每个子节点中可能具有的所有度值。为了解决这一问题，我们采用了 [17] 中介绍的一种基于快速傅里叶变换的技术。对于给定的合并节点，该技术使我们能够在 $\mathcal{O}^\star((\chi_{\mathrm{d}} \cdot (\Delta + 1))^{\mathrm{tw}})$ 时间内，计算其$\mathcal{O}((\chi_{\mathrm{d}} \cdot (\Delta + 1))^{\mathrm{tw}})$ 个元组对应的所有表值。

更快的连接计算。 令 t 为一个连接节点，其子节点为 t_1 和 t_2。首先，我们为 X_t的顶点固定一种着色，因此从 $\chi^{s(t)}{}_{\mathrm{d}} = \mathcal{O}(\chi_{\mathrm{d}}^{\mathrm{tw}})$种不同选择中选定一种。我们将说明如何计算任意满足该着色的元组 $S[t, z_i^t]$ z_i^t。在下文中，对于所考虑的每个 z_i^t，我们都假设其满足该着色。

令元组 z_i^t, $\Sigma(z_i^t) = \sum_{j=1}^{s(t)} \delta^{t,i}_{\ j}$的和为 $\delta^{t,i}_{\ j} \in [0, \Delta]$。由于对所有 $j \in [s(t)],$, 均有 $\Sigma(z_i^t) \in [0, s(t) \cdot \Delta]$，因此结论成立。对于 t_1和 t_2中的每个元组 $z_i^{t_p}$，其中 $p \in \{1, 2\}$，我们将按如下方式构造标识符 $i(z_i^{t_p})$:

$$i(z_i^{t_p}) = \sum_{j=1}^{s(t)} (\Delta+1)^{j-1} \cdot \delta_j^{t_p,i} \le \Delta \cdot \sum_{j=1}^{s(t)} (\Delta+1)^{j-1} = \Delta \cdot \frac{1-(\Delta+1)^{s(t)}}{1-(\Delta+1)} = (\Delta+1)^{s(t)} - 1.$$

接下来，我们引入多项式 $P^{t_p}_{\Sigma_q}(x)$，其中 $p \in \{1, 2\}$ 且 $\Sigma_q \in [0, s(t) \cdot \Delta]$，使得$P^{t_p}_{\Sigma_q}(x)$由单项式 $S[t_p, z_i^{t_p}] \cdot x^{i(z_i^{t_p})}$ 构成，对每个满足 $\Sigma(z_i^{t_p}) = \Sigma_q$的元组 $z_i^{t_p}$均如此。随后，通过使用快速傅里叶变换，我们计算所有 $\Sigma_1, \Sigma_2 \in [0, s(t) \cdot \Delta]$对应的多项式 $P^{t_1}_{\Sigma_1} \cdot P^{t_2}_{\Sigma_2}$。由于两个度为 n的多项式相乘需要 $O(n \log n)$ 时间 [41], , 且我们对次数至多为 $(\Delta + 1)^{s(t)} - 1$的多项式执行了 $(s(t) \cdot \Delta + 1)^2 = O(n^2)$ 次此类乘法，因此总共需要 $O(n^2 \cdot (\Delta + 1)^{s(t)} \cdot s(t) \log(\Delta + 1)) = O^\star((\Delta + 1)^{s(t)})$ 时间。

▷ 断言 32. 对于 $z_{i_1}^{t_1}, z_{i_2}^{t_2}$，令 $i(z_{i_1}^{t_1}) + i(z_{i_2}^{t_2}) = \sum_{j=1}^{s(t)+1} a_j \cdot (\Delta + 1)^{j-1}$，其中 $0 \le a_j \le \Delta$。那么，以下条件等价:

- $\Sigma(z_{i_1}^{t_1}) + \Sigma(z_{i_2}^{t_2}) = \sum_{j=1}^{s(t)} a_j$，对于所
- 有 $j \in [s(t)]$, $\delta^{t_1,}_{\ j}{}^{\ i_1} + \delta^{t_2,j}{}^{\ i_2} \le \Delta$。

证明。 将 $i(z_{i_1}^{t_1})$、 $i(z_{i_2}^{t_2})$，以及 $\Sigma(z_{i_1}^{t_1}) + \Sigma(z_{i_2}^{t_2})$ 分别表示为 $(\Delta+1)$ 进制中的数 d^1, d^2 和 a，其中它们的每一位数字 d_j^1, d_j^2, a_j 均为介于 0 和 Δ 之间的数。于是， $\delta^{t_1}{}_{,j}{}^{i_1}$、 $\delta^{t_2}{}_{,j}{}^{i_2}$ 和 a_j 分别对应于 d^1、 d^2 和 a 的第 j 位数字。

现在，假设我们将数字 d^1 和 d^2 相加，并令 $s_j = d_j^1 + d_j^2$（若 $d_j^1 + d_j^2 \leq \Delta$），而当 $\Delta > 0$ 时，我们假设 $s_j = 1 + d_j^1 + d_j^2 - (\Delta+1) < d_j^1 + d_j^2$。注意， $\sum_{j=1}^{s(t)} s_j \geq \sum_{j=1}^{s(t)} a_j$ 成立。在这种情况下，假设 $\Sigma(z_{i_1}^{t_1}) + \Sigma(z_{i_2}^{t_2}) = \sum_{j=1}^{s(t)} a_j$，则可推出 $\sum_{j=1}^{s(t)} d_j^1 + \sum_{j=1}^{s(t)} d_j^2 \leq \sum_{j=1}^{s(t)} s_j$，进而可推出对所有 j， $s_j = d_j^1 + d_j^2 \implies d_j^1 + d_j^2 \leq \Delta$。

另一方面，如果对于每个 j 都有 $d_j^1 + d_j^2 \leq \Delta$，则可得 $a_j = d_j^1 + d_j^2$，因此 $\sum_{j=1}^{s(t)} d_j^1 + \sum_{j=1}^{s(t)} d_j^2 = \sum_{j=1}^{s(t)} a_j \implies \Sigma(z_{i_1}^{t_1}) + \Sigma(z_{i_2}^{t_2}) = \sum_{j=1}^{s(t)} a_j$。 ◁

作为断言 32 的推论，我们可以轻松判断 $P^{t_1}_{\Sigma_{q_1}} \cdot P^{t_2}_{\Sigma_{q_2}}$ 的一个单项式是否对应于一个 t 元组，该元组由分别来自和为 Σ_{q_1} 和 Σ_{q_2} 的 t_1 与 t_2 元组相加而产生。此外，任何这样的 t 元组都由某个单项式编码，其系数决定了该元组由和分别为上述值的元组对产生的不同方式的数量。因此，为了确定一个 t 元组可能出现的方式数量，只需将所执行的所有 $\mathcal{O}(n^2)$ 次多项式乘法中对应单项式的系数相加即可。

最后，为了计算 t 的其余元组的 S 值，只需对 X_t 的所有不同着色重复整个过程，从而得到 $\chi^{s(t)}{}_{\mathrm{d}}$ 次迭代。

最终，为了计算 t 的所有 $\mathcal{O}((\chi_{\mathrm{d}}(\Delta+1))^{\mathrm{tw}})$ 个元组的 S 值，我们需要 $\mathcal{O}^\star(\chi_{\mathrm{d}}^{\mathrm{tw}} \cdot (\Delta+1)^{\mathrm{tw}})$ 时间。

复杂度。 关于算法的最终复杂度，注意，对于节点 t，为了计算其所有 S 个 $\mathcal{O}((\chi_{\mathrm{d}} \cdot (\Delta+1)^{\mathrm{tw}})$ 元组对应的值，所需时间为：

- 若 t 是叶节点或引入节点，则每个元组需要$\mathcal{O}(1)$；若 t
- 是遗忘节点，则每个元组需要$\mathcal{O}(\chi_{\mathrm{d}}(\Delta+1))$；若 t 是连接
- 节点，则所有元组需要$\mathcal{O}^\star((\chi_{\mathrm{d}}(\Delta+1))^{\mathrm{tw}})$。

因此，该算法的总运行时间上界为$\mathcal{O}^\star((\chi_{\mathrm{d}}(\Delta+1))^{\mathrm{tw}})$。 ◀

4 树深度下界

本节给出以输入图的树深度为参数时，求解有界度顶点删除和缺陷着色问题的复杂度紧下界。与以往的归约一样，我们从一个 k-多彩团实例 $G = (V, E)$ 出发，其中顶点被划分为 k 个集合 V_i，对所有 $i \in [k]$ 均成立。我们的主要技术贡献是一种递归构造，它使所构造图的树深度相对于 k 保持线性，从而收紧了此前已知的下界。接下来，我们将简要介绍该构造中的新要素。图示请参见图8。

ˆ 对于每个集合 V_i，我们设计一个简单的选择构件 C_i，用于编码 V_i 中一个顶点的选择。我们还设计了一个简单的“副本”构件，它利用每个副本所需的常数个额外顶点，使我们能够生成多个编码相同值的选择构件。在以往的归约中（[5, 27]），我们会针对每个 k 个主要选择构件，至多构造 $k-1$ 个副本；随后，对于每个 i_1， $i_2 \in [k]$，我们会选取 ˆ ˆ C_{i_1}， C_{i_2} 的一对不同副本，并在这些副本上添加一些机制，以验证这些组所编码的选择是否对应某条边的两个端点。这种方法自然会得到一个树深度为

$\mathcal{O}(k^2)$ 的图；事实上，它甚至证明了更严格参数下的困难性：正如 [27] 指出的那样，如果移除用于确保选择构件的副本编码相同值的 $\mathcal{O}(k^2)$ 个顶点，我们就会得到一族树深度为常数的图。也就是说，该参数实际上是到常数树深度的调节集大小，而非树深度本身。

我们方法中的新思路是：注意到，如果允许归约充分利用树深度作为参数，就能避免这一构造中的二次膨胀。考虑一个稍微更一般的问题：我们有两个区间I_1、 $I_2 \subseteq [k]$ ，想构造某种机制，检查对于每个 $i_1 \in I_1$ 和$i_2 \in I_2$ ，我们对 V_{i_1}、 V_{i_2} 的选择是否有效，也就是说，是否编码了一条边的两个端点。从较高的层面看，这就是我们要为 $I_1 = I_2 = [k]$解决的问题；假设对于 $|I_1| = |I_2| = 1$的情形，我们已有一个基础构件。现在我们注意到，解决一般情形的一种方法是递归地处理：将两个区间各自对半切分，得到 $I_1 = I_1^L \cup I_1^H$和 $I_2 = I_2^L \cup I_2^H$，然后针对(I_1^L、 I_2^L)、(I_1^L、 I_2^H)、(I_1^H、 I_2^L)、(I_1^H、 I_2^H)中的每一对，检查相同的条件。为此，我们将每个选择构件复制两份，从而构造出 $\mathsf{O}(k)$ 个新的分隔顶点；同时，将问题归约为四个相同的问题实例，其中所有区间的大小都缩减了一半。因此，要计算这种构造的树深度，我们得到如下递推关系： $T(k) \leq \mathsf{O}(k) + T(k/2)$，最终得到树深度 $\mathsf{O}(k)$。请注意，这项技术之所以能比以往的归约取得更好的结果，恰恰是因为我们充分利用了树深度的全部能力：我们构造的实例中，有 ck 个顶点，移除它们后并不会将图分解成平凡的连通分量，而是得到一些分量；递归地，在这些分量中移除 $ck/2$ 个顶点后，会产生更简单的分量，如此继续，递归深度为 $\log k$。换句话说，与以往的归约不同，我们的做法关键在于依赖树深度的递归定义。

4.1 有界度顶点删除

▶ **定理 33**。 对于任意可计算函数 f，如果存在一种算法，能够在时间 $f(\text{树深度})n^{o(\mathrm{td})}$ 内求解有界度顶点删除，其中 树深度 表示输入图的树深度，那么指数时间假设不成立。

证明。设 (G, k) 是 k-多色团问题的一个实例，且 G中的每个顶点都有一个自环，即对于所有 $v \in V(G)$，都有 $\{v, v\} \in E(G)$。回顾一下，我们假设 G 已划分为 k 个独立集 $V_1, \ldots, V_k$，其中 $V_i = \{v_1^i, \ldots, v_n^i\}$。不失一般性，假设对于某个 $z \in \mathbb{N}$， $k = 2^z$ （可以通过添加与图中所有其他顶点相连的虚拟独立集来做到这一点）。此外，令 $E^{i_1,i_2} \subseteq E(G)$ 表示 G 中一端点位于 V_{i_1} 、另一端点位于 V_{i_2} 的边。令 $\Delta = n^3$。我们将在多项式时间内构造一个树深度为 $\mathsf{td}(H) = \mathsf{O}(k)$、大小为 $|V(H)| = k^{\mathcal{O}(1)} \cdot n^{\mathcal{O}(1)}$ 的图 H，使得当且仅当 G 有一个 k-团时，存在 $S \subseteq V(H)$、 $|S| \leq k'$，且 $H - S$ 的最大度至多为 Δ，其中 k' 为某个值。

选择构件。 对于一个独立集 V_i，我们构造选择构件 C_i ，如图 9a 所示。我们首先构造独立集 $C_i^p = \{v^i_{,1}{}^p, \ldots, v^i_{,n}{}^p\}$，其中 $p \in \{h, l\}$。之后，我们用一个顶点 q_j^i连接 $v^i_{,j}{}^h$ 和 $v^i_{,j}{}^l$ ，并在后者上添加 $\Delta - 1$ 个叶节点。直观地说，我们将考虑 v_j^i 中的顶点与恰好删除 C_i^l 中的 j 个顶点以及 C_i^h中的 $n - j$ 个顶点之间的一一映射，其中该顶点属于 G 的一个假定的 k-团。

复制构件。 给定两个实例 $\mathcal{I}_1, \mathcal{I}_2$ 选择构件 C_i，当我们说用一个 复制构件将它们连接起来时，我们会引入两个顶点 g_1 和 g_2，并分别连接到每个

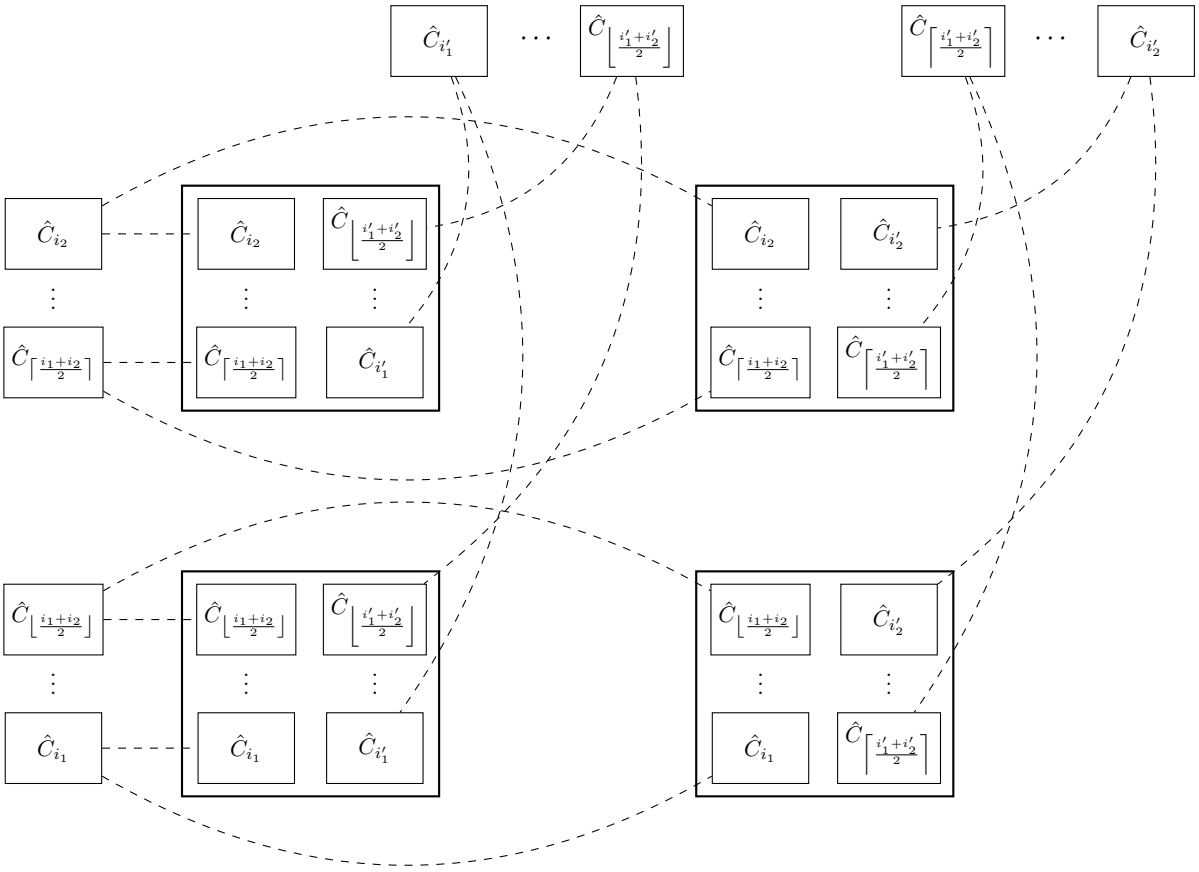


图8 $I_1 = [i_1, i_2]$ 和 $I_2 = [i'_1, i'_2]$的示意图。虚线表示副本，矩形表示约简后的实例。

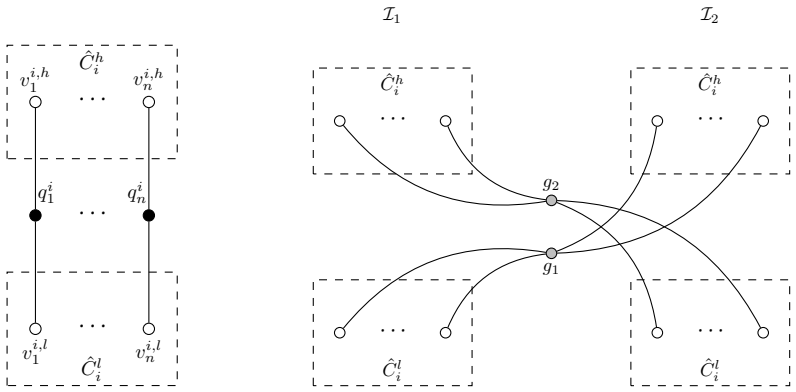
这些 $\Delta - n$ 叶节点，最后在 g_1（分别为 g_2）与实例 $\mathcal{I}_1$ 的 $\hat{C}_i^l$ 顶点之间添加一条边（分别与实例 $\mathcal{I}_2$的顶点相连），并在实例 $\mathcal{I}_2$ 的 $\hat{C}_i^h$顶点与（分别为实例 $\mathcal{I}_1$的顶点）之间添加一条边，如图 9b 所示。

边小工具。 令 $e = \{v_{j_1}^{i_1}, v_{j_2}^{i_2}\} \in E^{i_1,i_2}$ 为 G的一条边。如图 10 所示，构造边小工具 $\hat{E}_e$，其中每个顶点 c_j^i 都附有 Δ 个叶节点。

邻接构件。 对于 $i_1 \le i_2$ 和 $i'_1 \le i'_2$，我们将 邻接构件 $\hat{A}(i_1, i_2, i'_1, i'_2)$定义如下：

- 首先考虑 $i_1 = i_2$ 和 $i'_1 = i'_2$的情况。令邻接构件包含边构件 $\hat{E}_e$的实例，其中 $e \in E^{i_1, i'_1}$，选择构件 $\hat{C}_{i_1}$ 和 $\hat{C}_{i'_1}$，以及顶点 $\ell^l_{i_1,i'_1}$, $\ell^h_{i_1,i'_1}$, $r^l_{i_1,i'_1}$ 和 $r^h_{i_1,i'_1}$。在以下对象之间添加边
  - $\ell^l_{i_1,i'_1}$ and $\hat{C}^l_{i_1}$,
  - $\ell^h_{i_1,i'_1}$ and $\hat{C}^h_{i_1}$,
  - $r^l_{i_1,i'_1}$ and $\hat{C}^l_{i'_1}$,
  - $r^h_{i_1,i'_1}$ and $\hat{C}^h_{i'_1}$.

  若 $e = \{v_{j_1}^{i_1}, v_{j_2}^{i'_1}\} \in E^{i_1, i'_1}$，则添加与 $\hat{E}_e$ 相邻的以下边：



(a) 选择构件 $\hat{C}_i$. **(b)** 复制选择构件 $\hat{C}_i$。

图 9 黑色顶点连接了 $\Delta - 1$ 个叶节点，灰色顶点连接了 $\Delta - n$ 个叶节点。

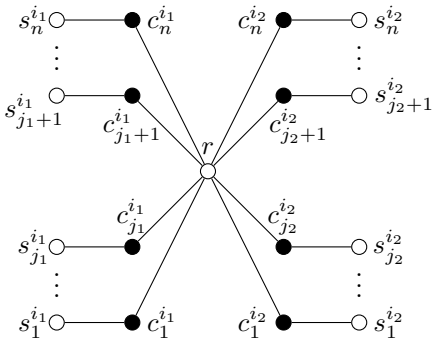


图 10 用于 $e =$ 的边小工具 $\hat{} E_e$ $\{v_{j_1}^{i_1}, v_{j_2}^{i_2}\}$。黑色顶点附有 Δ 个叶节点。

- ℓ_{i_1,i'_1}^l with $s_\kappa^{i_1}$, for $\kappa \in [j_1]$,
- ℓ_{i_1,i'_1}^h with $s_\kappa^{i_1}$, for $\kappa \in [j_1+1, n]$,
- r_{i_1,i'_1}^l with $s_\kappa^{i'_1}$, for $\kappa \in [j_2]$,
- r_{i_1,i'_1}^h with $s_\kappa^{i'_1}$, for $\kappa \in [j_2+1, n]$.

令 $\tau(x)$ 表示属于某个边小工具的 x 邻居数，其中 $x \in \{\ell_{i_1,i'_1}^l, \ell_{i_1,i'_1}^h, r_{i_1,i'_1}^l, r_{i_1,i'_1}^h\}$。在顶点 x 上附加 $\Delta - \tau(x)$ 个叶节点。示例见图 11。

- 现在考虑 $i_1 < i_2$ 和 $i'_1 < i'_2$的情况。然后，令 $\hat{}A(i_1、\ i_2、\ i'_1、\ i'_2)$ 包含选择构件 $\hat{} C_i$ 和 $\hat{} C_{i'}$，其中 $i \in [i_1, i_2]$ 和 $i' \in [i'_1, i'_2]$；我们将它们称为 $\hat{}A(i_1、\ i_2、\ i'_1、\ i'_2)$的原始选择构件，以及邻接构件

  - $A\left(i_1, \left\lfloor \frac{i_1+i_2}{2} \right\rfloor, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$、$\hat{}$
  - $A\left(i_1, \left\lfloor \frac{i_1+i_2}{2} \right\rfloor, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$、
  - $\hat{A}\left(\left\lceil \frac{i_1+i_2}{2} \right\rceil, i_2, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$，
  - $\hat{A}\left(\left\lceil \frac{i_1+i_2}{2} \right\rceil, i_2, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$.

  最后，我们用一个复制构件，将上述 $\hat{}\hat{}$ 邻接构件中出现的任意选择构件 C_i 和 $C_{i'}$，分别与对应的原始选择构件 C_i 和 $C_{i'}$相连。注意，这样一来，每个原始选择构件都参与两个复制构件。高层次示意图见图8。

令图 H为邻接构件 $\hat{}A(1,\ k,\ 1,\ k)$。注意，成立的是 $|V(H)| =(n \cdot k)^{\mathcal{O}(1)}$。令 $\beta = 2k(2k-1)$，并设 $k' = 2(|E(G)| - kn) \cdot 2n + kn \cdot 2n + 2\binom{k}{2} + k + n \cdot \beta$。

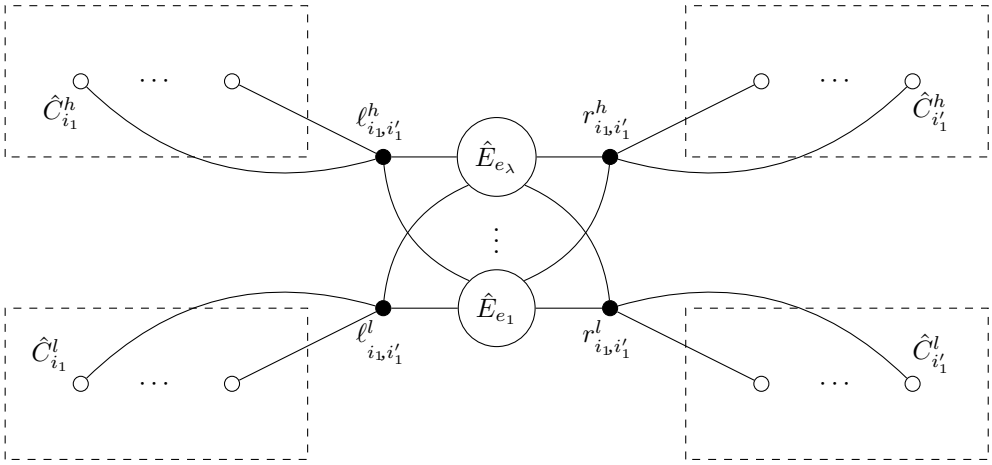


图11 邻接构件 $\hat{A}(i_1, i_1, i'_1, i'_1)$，其中 $E^{i_1,i_1} = \{e_i \mid i \in [\lambda]\}$。黑色顶点上连接着叶节点。

▶**引理 34**。 H 具有以下性质：

- 存在于 H 中的选择构件实例数量为 β；存在于 H中的边小工具$\hat{}$ E_e 的实例数量为：当
- $e = \{v^{i_1}_{j_1}, v^{i_2}_{j_2}\} \in E(G)$时为一个， $i_1 = i_2$时为两个。

证明。 对于第一个结论，注意到选择构件的实例数量由以下递归公式给出： $T(k) = 2k + 4T(k/2)$，其中 $T(1) = 2$。在这种情况下，可以推出 t

$$T(k) = \sum_{i=0}^{\log k} \left(4^i \cdot 2 \cdot \frac{k}{2^i}\right) = 2k \sum_{i=0}^{\log k} 2^i = 2k(2k-1) = \beta.$$

对于第二个结论，首先我们将证明，对于 H中出现的每个邻接构件$\hat{}A(i_1, i_2, i'_1, i'_2)$，都有 $i_2 - i_1 = i'_2 - i'_1 = 2^c - 1$，其中某个 $c \in \mathbb{N}$。对于$\hat{}A(1, k, 1, k)$，该结论成立；当 $i_2 - i_1 = i'_2 - i'_1 = 0$时亦然。假设对于某个$\hat{}A(i_1, i_2, i'_1, i'_2)$，该结论成立，即对于某个 $c \in \mathbb{N}$，有 $i_2 - i_1 = i'_2 - i'_1 = 2^c - 1 > 0$。那么，可以推出 $\left\lfloor \frac{i_1+i_2}{2} \right\rfloor - i_1 = \lfloor i_2 - 2^{c-1} + 0.5 \rfloor - i_1 = i_2 - i_1 - 2^{c-1} = 2^{c-1} - 1$。此外，可以推出 $i_2 - \left\lceil \frac{i_1+i_2}{2} \right\rceil = i_2 - \lceil i_1 + 2^{c-1} - 0.5 \rceil = i_2 - (i_1 + 2^{c-1}) = 2^{c-1} - 1$。因此，所述性质成立。

在这种情况下，对于某个 $\hat{}A(i_1, i_2, i'_1, i'_2)$，在递归的每一步中，区间 $[i_1, i_2]$ 和 $[i'_1, i'_2]$ 都从中间划分，并针对四种组合分别考虑一个邻接构件。在这种情况下，从 $\hat{}A(1, k, 1, k)$ 出发，只有一种方式可以构造出每个邻接构件 $\hat{}A(i_1, i_1, i_2, i_2)$，其中 $i_1, i_2 \in [k]$。考虑一个边小工具 $\hat{}E_e$，其中$e = \{v^{i_1}_{j_1}, v^{i_2}_{j_2}\} \in E(G)$。分两种情况：

- 如果 $i_1 = i_2$，那么这个构件只出现在邻接构件 $\hat{}A(i_1, i_1, i_1, i_1)$ 中；或者，它同时出现在邻接构件 $\hat{}A(i_1, i_1, i_2, i_2)$ 和 $\hat{}A(i_2, i_2, i_1, i_1)$ 中。证明至此完成。

◀

▶ **引理 35.** 成立：$\mathrm{td}(H) = O(k)$。

证明。 令 $T(\kappa)$ 表示在 $i_2 - i_1 = i'_2 - i'_1 = \kappa$ 的情况下，$\hat{}A(i_1, i_2, i'_1, i'_2)$ 的树深度。首先，注意到对于 $i_1, i_2 \in [k]$，$\hat{}A(i_1, i_1, i_2, i_2)$ 的树深度小于 8：删除顶点 $\ell^l_{i_1,i_2}$、$\ell^h_{i_1,i_2}$、$r^l_{i_1,i_2}$ 和 $r^h_{i_1,i_2}$ 后，所有剩余的连通分量都是高度至多为 3 的树。因此， $T(1) \leq 8$。

现在，考虑邻接构件 $\hat{A}(i_1,\ i_2,\ i'_1,\ i'_2)$，其中 $i_2 - i_1 = i'_2 - i'_1 = \kappa$。它由邻接构件组成

- $\hat{A}\left(i_1, \lfloor \frac{i_1+i_2}{2} \rfloor, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$,
- $\hat{A}\left(i_1, \lfloor \frac{i_1+i_2}{2} \rfloor, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$,
- $\hat{A}\left(\lceil \frac{i_1+i_2}{2} \rceil, i_2, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$,
- $\hat{A}\left(\lceil \frac{i_1+i_2}{2} \rceil, i_2, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$.

以及恰好 2κ 个原始选择构件，每个原始选择构件通过两个复制构件与邻接构件中存在的其他选择构件实例相连。删除所有复制构件中的顶点 g_1 和 g_2（见图 9b）后，所有原始选择构件以及邻接构件彼此断开。因此，有 $T(\kappa) \leq 8\kappa + T(\kappa/2)$，由此可得 $T(k) \leq 8\sum_{i=0}^{\log k} \frac{k}{2^i} = \mathcal{O}(k)$。◀

▶ **引理 36**。若 G 包含一个 k-团，则存在 $S \subseteq V(H)$，其中 $|S| \leq k'$，使得$H - S$ 的最大度至多为 Δ。

证明。令 $\mathcal{V} \subseteq V(G)$ 为 k-团，它属于 G，由顶点 $v^i_{s(i)} \in V_i$组成，其中 $i \in [k]$。我们将按如下方式构造删除集 $S \subseteq V(H)$：

- 令 S 包含来自每个选择构件 C_i的实例 $\hat{}$ 的顶点 $v^{i,\ h}_{j_1}$和 $v^{i,\ l}_{j_2}$，其中 $j_1 \in [s(i)]$ 且 $j_2 \in [s(i) + 1,\ n]$。令 $\hat{} E_e$ 为边 e的边小工具，其中 $e = \{v^{i_1}_{j_1},\ v^{i_2}_{j_2}\}$。若 $v^{i_1}_{j_1}$,
- $v^{i_2}_{j_2} \in \mathcal{V}$，则令 S从 $\hat{} E_e$ 中包含顶点 r、$s^{i_1}_j$和 $s^{i_2}_j$，其中 $j \in [n]$。或者，令 S 从 $\hat{} E_e$ 中包含所有顶点 $c^{i_1}_j$和 $c^{i_2}_j$，其中 $j \in [n]$。

第一种情况成立的边恰好是 $\binom{k}{2} + k$。根据引理 34，若 e 是自环，则 $\hat{}E_e$ 在 H 中出现一次；否则出现两次，而选择构件的实例在 H中恰好有 β 个。最终，S 包含由边小工具贡献的 $2(|E(G)| - kn) \cdot 2n + kn \cdot 2n + 2\binom{k}{2} + k$个顶点，另有由选择构件贡献的 $\beta \cdot n$ 个顶点，因此可得 $|S| = k'$。

剩下需要证明的是，$H - S$ 的最大度至多为 Δ。容易验证，复制构件中的每个顶点 g_1 和 g_2 的度恰好为 Δ。此外，选择构件实例中的每个顶点q^i_j 在 S中恰有一个邻居。对于边构件中的每个顶点 c^i_j，它自身或其邻接顶点 r 和 s^i_j 属于 S。最后，对于i_1、$i_2 \in [k]$，令$v^{i_1}_s(i_1)\ v^{i_2}_s(i_2) \in V$，其中 e 表示连接它们的边。那么，对于$\ell^l_{i_1,\ i_2}$，由于其叶节点、在$\hat{}\ C^l_{i_1}$中的邻居以及所有边构件中的邻居，它在 $H - S$中恰有 $\Delta - \tau(\ell^l_{i_1,\ i_2}) + v^{i_1}_s(i_1) + \tau(\ell^l_{i_1,\ i_2}) - v^{i_1}_s(i_1) = \Delta$个邻居。类似地，可以证明$\ell^h_{i_1,\ i_2}$、$r^l_{i_1,\ i_2}$和$r^h_{i_1,\ i_2}$在 $H - S$中的度均为 Δ。◀

▶ **引理 37**。如果存在 $S \subseteq V(H)$，且满足 $|S| \leq k'$，使得 $H - S$ 的最大度不超过 Δ，那么 G 包含一个 k-团。

证明。设 $S \subseteq V(H)$，其中 $|S| \leq k'$，使得 $H - S$ 的最大度至多为 Δ。根据引理 34，可以得出 H 恰好包含 β 个选择构件实例，而边构件 $\hat{}\ E_e$，其中 $e = \{v^{i_1}_{j_1}, v^{i_2}_{j_2}\}$，在 $i_1 = i_2$ 时出现一次，否则出现两次。注意，每个选择构件中，S必须至少包含 n 个顶点，因为每个顶点 q^i_j 的度为 $\Delta + 1$，且任意两个顶点都没有共同的邻居，因此顶点总数至少为 $n \cdot \beta$。此外，每个边构件中，S 至少包含 $2n$ 个顶点，因为顶点 c^i_j 的度为 $\Delta + 2$，且它们仅共享一个邻居。注意，边构件实例共有 $2(|E(G)| - kn) + kn$ 个，因此顶点总数为 $2(|E(G)| - kn) \cdot 2n + kn \cdot 2n$。

▷ 断言 38. 在 $\hat{}A(i_1, i_1, i_2, i_2)$ 中存在一个边小工具，使得 S 恰好包含 $2n+1$ 个顶点，对所有 $i_1, i_2 \in [k]$ 均成立。

证明。 由于 S 在每个选择构件和边构件中分别至少包含 n 个和 $2n$ 个顶点，因此我们剩余的额外预算至多为 $2\binom{k}{2}+k$。根据引理 34，邻接构件的数量$\hat{}A(i_1,\ i_1,\ i_2,\ i_2)$为 $2\binom{k}{2}+k$。因此，我们将证明，对于每个这样的邻接构件，S 包含 $2n+2n\cdot|E^{i_1,\ i_2}|+1$（请记住，每个这样的构件还包含 2 个选择构件以及 $|E^{i_1,\ i_2}|$ 个边构件）。事实上，我们将证明，对于每个这样的邻接构件，都存在一个边构件，使得 S 包含 $2n+1$个顶点。

首先，我们将证明，S 在每个邻接构件 $\hat{}A(i_1, i_1, i_2, i_2)$中恰好包含 $2n+2n\cdot|E^{i_1,\ i_2}|+1$ 个顶点。如果其包含的顶点少于此数，则必然意味着它从两个选择构件中各包含 n 个顶点，并且从该邻接构件的所有边构件中总共包含 $2n|E^{i_1,i_2}|$ 个顶点 c^i_j。在这种情况下，顶点 $\ell^l_{i_1,i_2}$ 和 $\ell^h_{i_1,i_2}$ 分别不能有来自 $\hat{}\,C^l_{i_1}$ 和 $\hat{}\,C^h_{i_1}$ 的邻居，但这是不可能的，因为 $\hat{}\,C_{i_1}$ 中只有 n 个顶点属于 S。另一方面，如果 S 从某个邻接构件 $\hat{}A(i_1, i_1, i_2, i_2)$中包含了多于 $2n+2n\cdot|E^{i_1,i_2}|+1$ 个顶点，那么必然存在另一个邻接构件 $\hat{}A(i'_1, i'_1, i'_2, i'_2)$，使得它从中包含的顶点少于这一数量，这就产生了矛盾。

现在，假设对于某个邻接构件 $\hat{}A(i_1, i_1, i_2, i_2)$，不存在任何边构件使得 S 包含 $2n+1$ 个顶点。在这种情况下，对于每个边构件，其中的所有顶点c^i_j 都属于 S，因此，对于每个顶点$\ell^l_{i_1,i_2}$, $\ell^h_{i_1,i_2}$, $r^l_{i_1,i_2}$ 和$r^h_{i_1,i_2}$，要么其在选择构件中的所有邻居都属于 S，要么该顶点本身属于 S。由于这些顶点中至多只有一个可以属于 S，这就导致矛盾。由此可知，S 在$\hat{}A(i_1, i_1, i_2, i_2)$的每个选择构件中恰好包含 n 个顶点，并且包含除一个边构件之外的所有边构件中的全部顶点 c^i_j。对于额外的边构件，我们可以假设 S 包含顶点 r 以及所有顶点 s^i_j（若非如此，则存在一个基数相同的删除集 S' 使这一点成立，因为只有顶点s^i_j 与边构件外的顶点相邻）。◁

因此，S 每个选择构件恰好包含 n 个顶点，以及来自邻接构件 $\hat{}A(i_1, i_1, i_2, i_2)$ 的边构件的 $2n\cdot|E^{i_1,i_2}|+1$个顶点。注意，复制构件中的顶点g_1 和 g_2 均不属于 S，并且二者在对应选择构件的部分中至多各有 n 个邻居属于 $H-S$（见图 9b）。在这种情况下，可以推出，只有顶点 $v^{i,\ l}_j$ 和 $v^{i,\ h}_j$ 属于来自选择构件的 S。此外，可以推出，在图 9b 中，$\hat{}C^j_i$ 中属于实例 $\mathbf{I}_1$ 的 S 的顶点数量，与对于 $j \in \{l, h\}$ 的实例 $\mathbf{I}_2$, 中的相应数量相同。因此，我们得出结论：在选择构件 $\hat{}C_i$的所有实例中，属于 $\hat{}\,C^l_i$（分别为 $\hat{}\,C^h_i$）的 S 的顶点数量均相同。

令$\mathcal{V} \subseteq V(G)$为一个基数为 k的集合，其中当选择构件$\hat{}C_i$满足 $|S \cap \hat{}C^h_i| = s(i)$且 $|S \cap \hat{}\,C^l_i| = n - s(i)$时，包含顶点 $v^i_{s(i)} \in V_i$。注意，对于所有$i \in [k]$，都有$\mathcal{V} \cap V_i \neq \emptyset$。我们将证明 $\mathcal{V}$ 是一个团。

令$v^{i_1}_s(i_1)$ $v^{i_2}_s(i_2)$属于V。考虑邻接构件$\hat{}A(i_1, i_1, i_2, i_2)$。我们将证明它包含一个边小工具$\hat{}$ E_e，其中 $e = \{v^{i_1}_s(i_1)\ v^{i_2}_s(i_2)\}$。考虑顶点$\ell^h_{i_1,i_2}$,$\ell^l_{i_1,i_2}$, $r^h_{i_1,i_2}$和$r^l_{i_1,i_2}$。可知 x有 $\Delta - \tau(x)$个叶节点，并且在边构件中有 $\tau(x)$个邻居，其中 $x \in \{\ell^h_{i_1,i_2}, \ell^l_{i_1,i_2}, r^h_{i_1,i_2}, r^l_{i_1,i_2}\}$。此外，

- $\ell^h_{i_1,i_2}$ 由于 $\hat{}\,C^h_{i_1}$ 属于 $H-S$，有 $n-s\ (i_1)$ 个邻居
- $\ell^l_{i_1,i_2}$ 由于 $\hat{}\,C^l_{i_1}$ 属于 $H-S$，有 $s\ (i_1)$ 个邻居$r^h_{i_1,i_2}$
- 由于 $\hat{}\,C^h_{i_2}$ 属于 $H-S$，有 $n-s\ (i_2)$ 个邻居，

$r^l_{i_1,i_2}$ 有 $s(i_1)$ 个邻居，这是因为 $H - S$ 中的 $C^l_{i_2}$。注意，除一个边构件外，S 包含所有 c^i_j 顶点。由于所有 x 在 $H - S$ 中的度都至多为 Δ，因此存在一个边小工具 $E_{e'}$，其中

$\ell^h_{i_1,i_2}$ 在 $n - s(i_1)$ 中至少有 $n - s(i_1)$ 个邻居，
$\ell^l_{i_1,i_2}$ 在 $s(i_1)$ 中至少有 $s(i_1)$ 个邻居，$r^h_{i_1,i_2}$
在 $n - s(i_2)$ 中至少有 $n - s(i_2)$ 个邻居，$r^l_{i_1,}$
$_{i_2}$ 在 $s(i_1)$ 中至少有 $s(i_1)$ 个邻居，

and，且所有顶点 s^i_j 都属于 S。这种情况唯一可能发生在 $e\prime = e$，因此 G 中存在这样一条边。

由于这对属于 $\mathcal{V}$ 的任意两个顶点都成立，因此 G 中存在一个 k-团。◀

因此，根据引理 35，我们可以在多项式时间内构造一个图 H，其树深度 td 为 $=\mathcal{O}(k)$；根据引理 36 和 37，判定是否存在大小为 $|S| \leq k'$ 的 $S \subseteq V$（H），且 $H - S$ 的最大度至多为 $\Delta = n^3$，等价于判定 G 中是否存在一个 k-团。在这种情况下，假设存在一个用于有界度顶点删除的 $f(\mathrm{td})|V(H)|^{o(\mathrm{td})}$算法，其中 f 是任意可计算函数，那么便可以在 $f(\mathrm{td})|V(H)|^{o(\mathrm{td})} = g(k) \cdot n^{o(k)}$ 时间内判定 k-多色团问题，其中 g 是某个可计算函数，这与指数时间假设相矛盾。◀

4.2 缺陷着色

证明将紧密沿用 [5] 中给出的困难性证明。由于我们会反复使用相等构件和调色板构件（见第 3.2.1 节），因此采用以下约定：对于我们已经引入到所构造图 H 中的两个顶点 v_1, v_2，当我们说添加相等构件 $Q(v_1, v_2)$ 时，意思是向 H 中添加 $Q(u_1, u_2, \chi_\mathrm{d}, \Delta)$ 的一个副本，然后分别将 u_1, u_2 与 v_1, v_2 识别为同一顶点（调色板构件也同理）。

▶ **定理 39.** 对于任意固定的 $\chi_\mathrm{d} \geq 2$，如果存在一个算法，能够在时间 $f(\mathrm{td})n^{o(\mathrm{td})}$ 内解决 缺陷着色 问题，其中 f 是任意可计算函数，且 td 表示输入图的树深度，那么指数时间假设不成立。

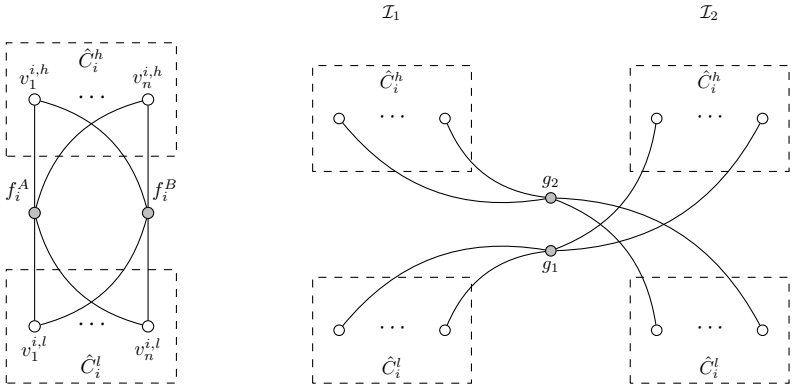
证明。设 (G, k) 是 k-多彩团的一个实例，且 G 的每个顶点都有一个自环，即对于所有 $v \in V(G)$，都有 $\{v, v\} \in E(G)$。回顾一下，我们假设输入图 G 已被划分为 k 个独立集 $V_1, \ldots, V_k$，其中 $V_i = \{v^i_1, \ldots, v^i_n\}$，且每个独立集的大小恰为 n。不失一般性，假设 $k = 2^z$ 对某个 $z \in \mathbb{N}$ 成立（可通过添加与图中所有其他顶点相连的虚拟独立集来实现）。此外，令 $E^{i_1,i_2} \subseteq E(G)$ 表示 G 中一端点位于 V_{i_1}、另一端点位于 V_{i_2} 的边。令 $\Delta = 2(|E(G)| - kn) + kn - (2\binom{k}{2} + k)$。我们将在多项式时间内构造一个树深度 $\mathrm{td}(H) = \mathrm{O}(k)$ 的图 H，使得 H 存在一个 $(\chi_\mathrm{d}, \Delta)$-着色，当且仅当 G 有一个 k-团。

调色板顶点。构造两个顶点 p^A 和 p^B，称它们为主调色板顶点，并添加一条连接它们的边。接下来，构造顶点 p^A_i和 p^B_i，对于 $i \in [\Delta]$，添加相等性构件 $Q(p^j, p^j_i)$ 以及 p^j 与 p^j_i 之间的边，其中 $j \in \{A, B\}$ 和 $i \in [\Delta]$。

选择构件。对于独立集 V_i，我们构造选择构件 C_i，如图12a所示。首先构造独立集 $C^p_i = \{v^{i,}_{j}{}^{p} \mid j \in [n]\}$，其中 $p \in \{h, l\}$。我们将这些顶点称为选择顶点。接着，引入顶点 f^A_i和f^B_i，并将它们与所有选择顶点相连，再添加相等性构件 $Q(p^A, f^A_i)$和 $Q(p^B, f^B_i)$。

最后，我们将 $\Delta-n$ 个叶节点 $l_j^{f_i^A}$（分别为 $l_j^{f_i^B}$ 个）连接到 f_i^A（分别连接到 f_i^B），其中 $j\in[\Delta-n]$，并添加相等性构件 $Q(p^A, l_j^{f_i^A})$（分别为 $Q(p^B, l_j^{f_i^B})$）。若 $\chi_{\mathrm{d}}\geq 3$，则对所有选择顶点 $v^{i,q}_j$ 添加一个调色板构件 $P(p^A, p^B, v^{i,q}_j)$。直观地说，我们考虑这样一种一一映射：将属于 G 中一个假定的 k-团的 V_i 的顶点 v_j^i，对应到对ˆ C_i^l 中的恰好 j 个顶点以及ˆ C_i^h中的 $n-j$ 个顶点进行着色，且它们的颜色与用于给 p^A着色的颜色相同。

ˆ复制构件。给定选择构件 C_i 的两个实例 $\mathrm{I}_1, \mathrm{I}_2$，当我们说用复制构件将它们连接起来时，我们引入两个顶点 g_1 和 g_2，并添加相等性构件 $Q(p^A, g_1)$ 和 $Q(p^A, g_2)$。此外，我们在 g_1（分别为 g_2）与实例 I_1（分别为 I_2）的ˆ C_i^l 顶点之间添加一条边，同时也在实例 I_2（分别为 I_1）的ˆ C_i^h 顶点之间添加一条边，如图 12b 所示。最后，我们分别在 g_1 和 g_2 $\Delta-n$ 上连接 $l_j^{g_1}$ 和 $l_j^{g_b}$ 个叶节点，其中 $j\in[\Delta-n]$，并添加相等性构件 $Q(p^A, l_j^{g_1})$ 和 $Q(p^A, l_j^{g_2})$。



(a) 选择构件ˆ C_i。　　**(b)** 复制一个选择构件ˆ C_i。

图12 灰色顶点附有 $\Delta-n$ 个叶节点。

边小工具。令 $e=\{v_{j_1}^{i_1}, v_{j_2}^{i_2}\}\in E^{i_1,i_2}$ 为 G的一条边。按照图 10 所示构造边小工具ˆE_e，其中顶点 c_e 连接了 Δ 个叶节点 $l_j^e, j\in[\Delta]$，并为每个这样的叶节点添加相等性构件 $Q(p^B, l_j^e)$。此外，若 $\chi_{\mathrm{d}}\geq 3$，则对每个顶点 $v\in\{c_e, s_j^{i_1}, s_j^{i_2}\}$，添加一个调色板构件$P(p^A, p^B, v)$。

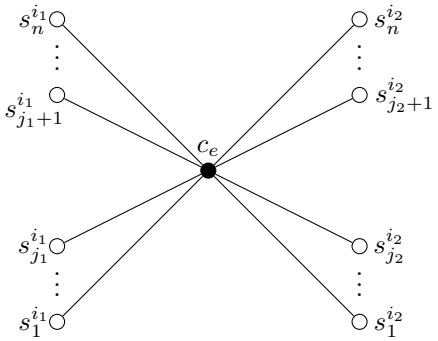


图 13 边小工具ˆ E_e 用于 $e=\{v_{j1}^{i_1}, v_{j2}^{i_2}\}$。黑色顶点连接着 Δ 个叶节点。

邻接构件。 对于 $i_1 \leq i_2$ 和 $j_1 \leq j_2$，我们定义邻接构件$\hat{}A(i_1, i_2, j_1, j_2)$如下：

- 首先考虑 $i_1 = i_2$ 和 $i'_1 = i'_2$ 的情况。令邻接构件包含边构件 $\hat{}\, E_e$ 的实例，其中 $e \in E^{i_1,\, i'_1}$；选择构件 $\hat{}\, C_{i_1}$ 和 $\hat{}\, C_{i'_1}$；以及顶点 $\ell^l_{i_1,\, i'_1}$、 $\ell^h_{i_1,\, i'_1}$、 $r^l_{i_1,\, i'_1}$ 和 $r^h_{i_1,\, i'_1}$。在它们之间添加边
  - $\ell^l_{i_1,\, i'_1}$ and $\hat{C}^l_{i_1}$,
  - $\ell^h_{i_1,\, i'_1}$ and $\hat{C}^h_{i_1}$,
  - $r^l_{i_1,i'_1}$ and $\hat{C}^l_{i'_1}$,
  - $r^h_{i_1,i'_1}$ and $\hat{C}^h_{i'_1}$.

  如果 $e = \{v^{i_1}_{j_1},\ v^{i'_1}_{j_2}\} \in E^{i_1,\, i'_1}$，则在 $\hat{}\, E_e$ 邻接的位置添加以下边：
  - $\ell^l_{i_1,i'_1}$ 对于 $\kappa \in [j_1]$，使用 $s^{i_1}_\kappa$,
  - $\ell^h_{i_1,i'_1}$对于 $s^{i_1}_\kappa$，对于 $\kappa \in [j_1 + 1, n]$,
  - $r^l_{i_1,\, i'_1}$ 对于 $s^{i'_1}_\kappa$，对于 $\kappa \in [j_2]$,
  - $r^h_{i_1,i_1}$对于 $s^{i'_1}_\kappa$，对于 $\kappa \in [j_2 + 1, n]$。

  对于每个顶点 $x \in \{\ell^l_{i_1,\, i'_1},\ \ell^h_{i_1,\, i'_1},\ r^l_{i_1,\, i'_1},\ r^h_{i_1,\, i'_1}\}$，添加一个相等构件 $Q(p^A,\ x)$，并连接 $\Delta - n$ 个叶节点。最后，对于每个叶节点 l，添加一个相等构件 $Q(p^A,\ l)$。图示见图14。

- 现在考虑 $i_1 < i_2$ 和 $i'_1 < i'_2$ 的情形。于是，令 $\hat{}A(i_1,\ i_2,\ i'_1,\ i'_2)$ 包含选择构件 $\hat{}\, C_i$ 和 $\hat{}\, C_{i'}$，其中 $i \in [i_1, i_2]$ 和 $i' \in [i'_1, i'_2]$，我们将其称为 $\hat{}A(i_1,\ i_2,\ i'_1,\ i'_2)$ 的**原始选择构件**，以及邻接构件
  - $\hat{}A\left(i_1, \left\lfloor \frac{i_1+i_2}{2} \right\rfloor, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$, $\hat{}A\left(\left\lceil \frac{i_1+i_2}{2} \right\rceil, i_2, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$, $\hat{}A\left(i_1, \left\lfloor \frac{i_1+i_2}{2} \right\rfloor, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$, $\hat{}A\left(\left\lceil \frac{i_1+i_2}{2} \right\rceil, i_2, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$.

  $\hat{}\,\hat{}$ 最后，我们用一个复制构件将上述邻接构件中出现的任意选择构件 C_i 和 $C_{i'}$，与相应的原始选择构件 C_i 和 $C_{i'}$ 相连。注意，此时每个原始选择构件都参与两个复制构件。高层次示意图见图8。

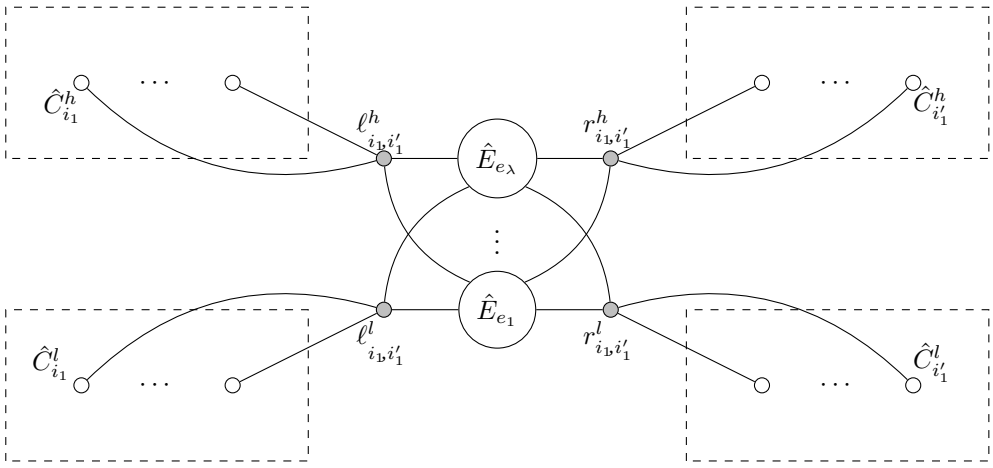


图14 邻接构件 $\hat{}A(i_1, i_1, i'_1, i'_1)$，其中 $E^{i_1,i_1} = \{e_i \mid i \in [\lambda]\}$。灰色顶点附有 $\Delta - n$ 叶节点。

要构造图 H，首先构造邻接构件 $\hat{}A(1,\ k,\ 1,\ k)$。然后引入顶点 u；对于所有 $e \in E(G)$，该顶点都与 $\hat{}\, E_e$ 中的顶点 c_e 相连。最后，添加相等构件 $Q(p^A,\ u)$。

40 两个有界度问题的结构参数化再探

▶ **引理 40**。在 H 中出现的边小工具 $\hat{} E_e$ 的实例数量为：当 $i_1 = i_2$ 时为一个，否则为两个，其中 $e = \{v_{j_1}^{i_1}, v_{j_2}^{i_2}\} \in E(G)$。

证明。首先，我们将证明，对于 H 中出现的每个邻接构件 $\hat{} A(i_1, i_2, i_1', i_2')$，都有 $i_2 - i_1 = i_2' - i_1' = 2^c - 1$，其中某个 $c \in \mathbb{N}$。对于 $\hat{} A(1, k, 1, k)$，以及当 $i_2 - i_1 = i_2' - i_1' = 0$ 时，该命题成立。假设该命题对某个 $\hat{} A(i_1, i_2, i_1', i_2')$ 成立，即 $i_2 - i_1 = i_2' - i_1' = 2^c - 1 > 0$，其中某个 $c \in \mathbb{N}$。那么可得 $\left\lfloor \frac{i_1+i_2}{2} \right\rfloor - i_1 = \lfloor i_2 - 2^{c-1} + 0.5 \rfloor - i_1 = i_2 - i_1 - 2^{c-1} = 2^{c-1} - 1$。此外，可得 $i_2 - \left\lceil \frac{i_1+i_2}{2} \right\rceil = i_2 - \lceil i_1 + 2^{c-1} - 0.5 \rceil = i_2 - (i_1 + 2^{c-1}) = 2^{c-1} - 1$。因此，所述性质成立。——

在这种情况下，对于某个 $\hat{} A(i_1, i_2, i_1', i_2')$，在递归的每一步中，区间 $[i_1, i_2]$ 和 $[i_1', i_2']$ 都在中间划分，并针对四种组合分别考虑一个邻接构件。在这种情况下，从 $\hat{} A(1, k, 1, k)$ 出发，生成每个邻接构件 $\hat{} A(i_1, i_1, i_2, i_2)$ 的方式只有一种，其中 $i_1, i_2 \in [k]$。考虑一个边小工具 $\hat{} E_e$，其中 $e = \{v_{j_1}^{i_1}, v_{j_2}^{i_2}\} \in E(G)$。分两种情况：

- 如果 $i_1 = i_2$，那么这个构件只会出现在邻接构件 $\hat{} A(i_1, i_1, i_1, i_1)$ 中；或者，它会同时出现在邻接构件 $\hat{} A(i_1, i_1, i_2, i_2)$ 和 $\hat{} A(i_2, i_2, i_1, i_1)$ 中。证明完毕。

◀

$\hat{}$▶ **引理 41**。设 I_1, I_2 是由一个复制构件连接的两个 C_i 实例。则对于 χ_d，Δ（c）-着色 V：H（$\to [\chi_d]$）*of* H，有

- I_1 中颜色为 p^A（c）的顶点数量等于 I_2，中颜色为 p^A（c）的顶点数量，其中前者分别对应 $\hat{} C_i^l$，后者分别对应 $\hat{} C_i^l$；I_1 中颜色为 p^B（c）的顶点数量等于 I_2 中颜色为 p^B（c）的顶点数量，其中前者分别对应 $\hat{} C_i^l$，后者分别对应 $\hat{} C_i^l$。

证明。由于调色板构件，$\hat{} C_i$ 的任意选择顶点所着颜色必为 $c(p^A)$ 或 $c(p^B)$。根据相等性构件的性质可知，顶点 f_i^A（分别为 f_i^B）的颜色为 $c(p^A)$（分别为 $c(p^B)$），并且具有 $\Delta - n$ 个同色相邻叶节点。因此，$\hat{} C_i$ 中恰有 n 个选择顶点获得颜色 $c(p^A)$，另有恰好 n 个获得颜色 $c(p^B)$。

设 g_1, g_2 为连接实例 $\mathcal{I}_1, \mathcal{I}_2$ 的复制构件中存在的顶点。因此，除叶节点外，它们每个至多有 n 个颜色为 c（p^A）的邻居。在这种情况下，对于 $j \in \{l, h\}$，如果实例 $\mathcal{I}_1$ 的 $\hat{} C_i^j$ 中颜色为 c（p^A）的顶点数与实例 $\mathcal{I}_2$，中相应的顶点数不同，那么 g_1 或 g_2 的同色邻居数必然超过 Δ，矛盾。◀

▶ **引理 42.** 有 树深$(H) = O(k)$。

证明。我们首先注意到，添加到图中的所有相等关系构件和调色板构件在 $\{p^A, p^B\}$ 之外至多有一个端点。因此，根据 [5, 引理 3.6 和 3.9], ，我们可以得出 $\mathsf{td}(H) = \mathsf{td}(H' \setminus \{p^A, p^B\}) + \chi_d + 1$，其中 H' 是从 H 中移除所有相等关系构件和调色板构件后得到的图。因此，只需证明 $\mathsf{td}(H') = O(k)$。我们首先移除顶点 u。

现在，令 $T(\kappa)$ 表示当 $i_2 - i_1 = i_2' - i_1' = \kappa$ 时，$\hat{} A(i_1, i_2, i_1', i_2')$ 的树深度。首先注意，对于 $i_1, i_2 \in [k]$，$\hat{} A(i_1, i_1, i_2, i_2)$ 的树深度至多为 8：移除顶点 ℓ_{i_1,i_2}^l，ℓ_{i_1,i_2}^h，r_{i_1,i_2}^l 和 r_{i_1,i_2}^h，使选择构件与边构件断开。然后，只需从选择构件中移除顶点 f_i^A, f_i^B, f_j^A 和 f_j^B，而边构件是高度为 2 的树。因此，$T(1) \leq 8$。

现在，考虑邻接构件 $\hat{} A(i_1, i_2, i_1', i_2')$，其中 $i_2 - i_1 = i_2' - i_1' = \kappa$。它由邻接构件组成

- $\hat{A}\left(i_1, \lfloor \frac{i_1+i_2}{2} \rfloor, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$,
- $\hat{A}\left(i_1, \lfloor \frac{i_1+i_2}{2} \rfloor, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$,
- $\hat{A}\left(\lceil \frac{i_1+i_2}{2} \rceil, i_2, i'_1, \left\lfloor \frac{i'_1+i'_2}{2} \right\rfloor\right)$,
- $\hat{A}\left(\lceil \frac{i_1+i_2}{2} \rceil, i_2, \left\lceil \frac{i'_1+i'_2}{2} \right\rceil, i'_2\right)$.

以及恰好 2κ 个原始选择构件组成，其中每个构件都通过两个复制构件，与邻接构件中存在的其他选择构件实例相连。移除复制构件的所有顶点 g_1 和 g_2（见图 12b）后，所有原始选择构件以及邻接构件彼此断开。因此，有 $T(\kappa) \leq 8\kappa + T(\kappa/2)$，进而可得 $T(k) \leq 8\sum_{i=0}^{\log k} \frac{k}{2^i} = \mathcal{O}(k)$，即 $\mathsf{td}(H') = \mathcal{O}(k)$。◀

▶ **引理 43.** 对于任意 $\chi_d \geq 2$，如果 G 包含一个 k-团，那么 H 可进行 (χ_d, Δ)-着色 .

证明。 令 $\mathcal{V} \subseteq V(G)$ 为 G中由顶点 $v^i_{s(i)} \in V_i$（$i \in [k]$）组成的一个 k-团。我们将按如下方式构造 (χ_d, Δ)-着色： H

- 顶点 p^A 着色为1，顶点 p^B 着色为2。对于所有添加了相等构件且其中一个端点与p^A
- （分别与 p^B）重合的顶点，均着色为1（分别为2）。我们使用 [5, 引理3.5] 对相等构
- 件的内部顶点进行正确着色。对于选择构件 $\hat{} C_i$，将 $\{v^i_{j,h} \in \hat{} C^h_i \mid j \in [s(i)]\} \cup \{v^i_{j,l} \in$
- $\hat{} C^l_i \mid j \in [s(i)+1, n]\}$ 中的顶点着色为1，同时将 $\{v^i_{j,l} \in \hat{} C^l_i \mid j \in [s(i)]\} \cup \{v^i_{j,h} \in \hat{}$
- $C^h_i \mid j \in [s(i)+1, n]\}$ 中的顶点着色为2。对于团中包含的每条 $e = \{v^{i_1}_s(i_1)\ v^{i_2}_s(i_2)\}$，将 $\hat{}$ E_e 中所有 $s^{i_1}_j, s^{i_2}_j$（$j \in [n]$）着色为1，并将顶点 c_e 着色为2。对于所有其他不属于该团的边 e'，采用相反的着色方式：将顶点 $s^{i_1}_j, s^{i_2}_j$着色为2，将顶点 $c_{e'}$ 着色为1。我们使
- 用 [5, 引理3.8] 对调色板构件的内部顶点进行正确着色，因为我们添加的所有调色板构件，其端点上要么两次使用颜色1，要么两次使用颜色2。

-

着色至此完成。

还需证明这确实是一个（χ_d, Δ）-着色。首先注意，根据 [5], 的引理 3.5 和 3.8，相等关系构件和调色板构件中的内部顶点都已正确着色。顶点 p^A, p^B 恰有 Δ 个同色邻居。顶点 f^A_i 和 f^B_i 在 $\hat{} C^l_i \cup \hat{} C^h_i$ 的顶点中恰有 n 个同色邻居，因此总共恰有 Δ 个同色邻居。复制构件中的每个顶点 g_1 和 g_2 也同样如此。选择顶点的同色邻居数量为常数（由顶点 $f^A_i, f^B_i, g_1, g_2, \ell^l_{i_1,i_2}, \ell^h_{i_1,i_2}, r^l_{i_1,i_2}, r^h_{i_1,i_2}$导致）。顶点 u 恰有 $\deg_H(u) - (2\binom{k}{2} + k) = \Delta$ 个颜色为 1 的邻居，因为由引理 40 可知，对于邻接构件 $\hat{A}(i, i, j, j)$，若 $i \neq j$，则它在 H 中出现两次，否则出现一次，而团包含 $\binom{k}{2}$ 条边 $e = \{v^{i_1}_s(i_1)\ v^{i_2}_s(i_2)\}$，其中 $i_1 \neq i_2$，且 k，其中 $i_1 = i_2$。最后，对于顶点$\ell^l_{i_1,i_2}, \ell^h_{i_1,i_2}, r^l_{i_1,i_2}, r^h_{i_1,i_2}$，注意到 $\ell^h_{i_1,i_2}$（分别为 $\ell^l_{i_1,i_2}$）在选择顶点中恰有 $s(i_1)$（分别为$n - s(i_1)$）个颜色为 1 的邻居。此外，$\ell^h_{i_1,i_2}$（分别为$\ell^l_{i_1,i_2}$）在边构件中恰有 $n - s(i_1)$（分别为 $s(i_1)$）个颜色为 1 的邻居，这些邻居属于 $\hat{} E_e$，其中 $e = \{v^{i_1}_s(i_1)\ v^{i_2}_s(i_2)\}$。类似地，可以证明 $r^l_{i_1,i_2}$ 和 $r^h_{i_1,i_2}$ 恰有 Δ 个同色邻居。◀

▶ **引理 44.** 对于任意 $\chi_d \geq 2$，如果 H 存在一个(χ_d, Δ)-着色，那么 G 包含一个 k-团。

证明。假设 $c\colon V(H) \to [\chi_d]$ 是 H的（χ_d, Δ）着色。由于相等构件的性质，每个顶点都有 Δ 个同色相邻叶节点，因此成立 $c(p^A) \neq c(p^B)$。不失一般性，假设 $c(p^A) = 1$ 且 $c(p^B) = 2$。对于每个

ˆ 选择构件 C_i 的实例中，一半的选择顶点被染成颜色 1，另一半被染成颜色 2：由于调色板构件，每个顶点都被染成这两种颜色之一；如果超过 n 个顶点被染成相同颜色，那么 f_i^A，f_i^B 中就会有一个顶点拥有超过 Δ 个同色邻居。此外，由于引理 41，可知对于 H 中出现的每个选择构件 C_i 实例，ˆ C_i^l （分别为 ˆ C_i^h）中被染成颜色 1 的顶点数量相同。我们将首先证明以下断言。

▷ 断言 45。对于每个 i_1，$i_2 \in [k]$，邻接构件 ˆ$A(i_1, i_1, i_2, i_2)$ 中都存在一个单独的边构件 ˆ E_e，其中 $c(c_e) = 2$。

证明。首先，注意到根据引理 40，u 总共有 $2(E(G) - kn) + kn$ 个邻居。因此，其邻居中至少有 $2(E(G) - kn) + kn - \Delta = 2\binom{k}{2} + k$ 个被染成颜色 2，因为根据相等构件 $Q(p^A, u)$，u 被染成颜色 1。注意，这也是邻接构件 ˆ$A(i_1, i_1, i_2, i_2)$ 的数量，因为每个这样的构件在 $i_1 \neq i_2$ 时出现两次，否则出现一次。假设对于某个邻接构件 ˆ$A(i_1, i_1, i_2, i_2)$，存在一个边构件 ˆ E_e，使得 $c(c_e) = 2$。在这种情况下，c_e 有 Δ 个颜色相同的相邻叶节点，因此 ˆ E_e 的所有顶点 $s_j^{i_1}$,$s_j^{i_2}$ 都被染成颜色 1 （若 $\chi_{\mathrm{d}} \geq 3$，这是附着于这些顶点的调色板构件所导致的结果）。如果 ˆ$A(i_1, i_1, i_2, i_2)$ 中还存在第二个边构件 ˆ $E_{e'}$，使得 $c(c_{e'}) = 2$，那么可知 $\ell_{i_1,\ i_2}^l$ 和 ℓ_{i_1,i_2}^h 总共有 $2n$ 个邻居被染成颜色 1，且属于边构件。因此，由于两者都已经有 $\Delta - n$ 个相邻的同色叶节点，可知它们各自恰有 n 个来自边构件且被染成颜色 1 的邻居，并且在选择构件 C_{i_1} 中没有这样的邻居；这就导致矛盾，因为 C_{i_1} 的选择顶点中恰有 n 个被染成颜色 1。因此，为了使 u 至少有 $2\binom{k}{2} + k$ 个颜色为 2 的邻居，可知每个邻接构件 ˆ $A(i_1, i_1, i_2, i_2)$ 都有一个单独的边构件 ˆ E_e，使得$c(c_e) = 2$。◁

令 $\mathcal{V} \subseteq V(G)$ 为一个基数为 k 的集合，其中包含顶点 $v_{s(i)}^i \in V_i$；如果对于选择构件 ˆC_i，恰好有 $s(i)$ 个顶点属于 ˆ C_i^h，以及 $n - s(i)$ 个顶点属于 ˆ C_i^l，且它们被着色为颜色 1，则上述条件成立。注意，对于每个 i，$\mathcal{V} \cap V_i \neq \emptyset$。我们将证明 $\mathcal{V}$ 是一个团。

令 $v_s^{i_1}(i_1)$ $v_s^{i_2}(i_2)$ 属于 V。考虑邻接构件 ˆ$A(i_1, i_1, i_2, i_2)$。我们将证明它包含一个边小工具 ˆ E_e，其中 $e = \{v_s^{i_1}(i_1)\ v_s^{i_2}(i_2)\}$。考虑顶点 $\ell_{i_1}^h$、$_{i_2}$、$\ell_{i_1}^l$、$_{i_2}$、$r_{i_1}^h$、$_{i_2}$ 以及 $r_{i_1}^l$、$_{i_2}$。它们中的每一个都附有 $\Delta - n$ 个颜色为 1 的相邻叶节点。此外，有

- $\ell_{i_1,\ i_2}^h$ 有 $s(i_1)$ 个来自 ˆ $C_{i_1}^h$ 的同色邻居，$\ell_{i_1,\ i_2}^l$ 有
- $n - s(i_1)$ 个来自 ˆ $C_{i_1}^l$ 的同色邻居，$r_{i_1,\ i_2}^h$ 有 $s(i_2)$
- 个来自 ˆ $C_{i_2}^h$ 的同色邻居，$r_{i_1,\ i_2}^l$ 有 $n - s(i_2)$ 个来自
- ˆ $C_{i_2}^l$ 的同色邻居。

注意，由于命题 45，存在一个边小工具 ˆ $E_{e'}$，其中 $e' = \{v_{j_1}^{i_1}, v_{j_2}^{i_2}\}$，使得 $c(c_{e'}) = 2$ 成立，因此所有顶点 $s_j^{i_1}, s_j^{i_2}$ 的颜色均为 1。要使 c 成为 $(\chi_{\mathrm{d}}, \Delta)$-着色，$\ell_{i_1,i_2}^l$ （分别地，ℓ_{i_1,i_2}^h）来自集合 $\{s_j^{i_1} \mid j \in [n]\}$ 的邻居至多有 $s(i_1)$ 个（分别地，$n - s(i_1)$ 个）。因此可知，ℓ_{i_1,i_2}^l （分别地，ℓ_{i_1,i_2}^h）来自该集合的邻居恰有 $s(i_1)$ 个（分别地，$n - s(i_1)$ 个）。同理可知，r_{i_1,i_2}^l （分别地，r_{i_1,i_2}^h）来自集合 $\{s_j^{i_2} \mid j \in [n]\}$ 的邻居恰有 $s(i_2)$ 个（分别地，$n - s(i_2)$ 个）。因此，$e' = \{v_{j_1}^{i_1}, v_{j_2}^{i_2}\}$，故在 G 中存在这样一条边。

由于这对属于 $\mathcal{V}$ 的任意两个顶点都成立，因此 G 有一个 k-团。◀

因此，我们可以在多项式时间内构造一个图 H，其树深度为 $\mathsf{td} = \mathcal{O}(k)$（依据引理 42），并且根据引理 43 和 44，判定 H 是否存在 $(\chi_{\mathrm{d}}, \Delta)$-

着色等价于判定 G 是否包含一个 k-团。在这种情况下，假设存在一个 $f(\mathrm{td})|V(H)|^{o(\mathrm{td})}$ 算法用于缺陷着色，其中 f 是任意可计算函数，则可以在时间 $f(\mathrm{td})|V(H)|^{o(\mathrm{td})} = g(k)$ $n^{o(k)}$ 内判定 k-多色团问题，其中 g 为某个可计算函数，这与指数时间假设相矛盾。◀

5 顶点覆盖下界

本节给出以输入图的顶点覆盖数为参数时，求解有界度顶点删除和缺陷着色问题的复杂度下界。对于这两种情况，我们都从一个含有 n 个变量的 3-SAT 实例出发，并构造一个等价实例，使其输入图的顶点覆盖数为 $\mathcal{O}(n/\log n)$；因此，任何能在 $\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$ 时间内求解后者问题的算法都会推翻指数时间假设。由此可知，这两个问题已有的算法基本上都是最优的。我们先介绍两个归约都会用到的一些必要工具，然后证明上述结果。

5.1 预备工具

我们首先定义 3-SAT 的一种受限版本，称为 (3,4)-XSAT。该变体与 (3,4)-SAT 问题密切相关，[49] 该问题询问给定公式 ϕ 是否可满足，其中 ϕ 是一个 3-SAT 公式，每个子句恰好包含 3 个不同的变量，且每个变量最多出现在 4 个子句中。正如 Bonamy 等人所指出的，[10], Tovey 的工作 [49] 的一个推论是：除非指数时间假设不成立，否则不存在用于 (3,4)-SAT 的 $2^{o(n)}$ 算法，其中 n 表示公式中的变量数。这里我们证明 (3,4)-XSAT 也有类似的下界。随后，我们紧随 [10], 中的引理 3.2，介绍一种将公式的变量和子句划分为若干组的方法，使得同一子句组中的子句所包含的变量分别属于不同的变量组。

(3,4)-XSAT

实例：	一个 3-SAT 公式 ϕ ，其中每个子句恰好包含 3 个不同的变量，且每个变量最多出现在 4 个子句中。
目标：	确定是否存在对变量的赋值，使得 ϕ 每个子句恰好有一个真文字。

▶**定理 46.** $(3,4)$-XSAT 除非指数时间假设不成立，否则无法在时间 $2^{o(n)}$ 内判定，其中 n 表示输入公式中的变量数。

证明。我们将紧密遵循一个已知的从 3-SAT 到 1-IN-3-SAT 的归约，该归约是对 Schaefer 工作的简化 [47]。设 ϕ 是一个含有 n 个变量和$m \leq \frac{4}{3} \cdot n$ 个子句的 (3,4)-SAT 公式。令 V 表示其变量集合，C 表示其子句集合。我们将按如下方式构造一个 (3,4)-XSAT 的等价实例 ϕ'，其中 表示其变量集合，V' 表示其子句集合：C'

- 对于每个变量 $x \in V$，引入一个变量 $x \in V'$；对于每个子句 $c_i \in C$，引入变量 α_i、β_i、
- γ_i、$\delta_i \in V\prime$；对于 ϕ 中的每个子句 $c_i = x \vee y \vee z$，在 ϕ' 中引入子句
- $c_i^1 = \neg x \vee \alpha_i \vee \beta_i$、$c_i^2 = y \vee \beta_i \vee \gamma_i$ 和 $c_i^3 = \neg z \vee \gamma_i \vee \delta_i$。

请注意，ϕ' 是一个有效的 (3,4)-XSAT 实例，因为它的每个 $3m$ 子句都恰好包含 3 个不同的变量，而它的所有 $n + 4m$ 变量都至多出现在 4 个子句中。接下来我们将证明，当且仅当存在一个对 ϕ' 中变量的赋值，使得其每个子句都恰好有一个真文字时，ϕ 才是可满足的。

44 两个有界度问题的结构参数化再探讨

对于正向方向，令 $f\colon V \to \{T, F\}$ 为 ϕ 的满足赋值。考虑一个赋值 $f'\colon V' \to \{T, F\}$，使得对于所有 $x \in V$， $f'(x) = f(x)$。剩下的任务是确定属于 $V' \setminus V$ 的任意变量 f' 的值。令 $c_i = x \vee y \vee z$ 为 ϕ 的一个子句。由于 f 是一个满足赋值，因此 x、y、z 中至少有一个取真值赋值。若 $f(y) = T$，则令 $f'(\beta_i) = f'(\gamma_i) = F$，同时令 $f'(\alpha_i) = f(x)$ 且 $f'(\delta_i) = f(z)$。另一方面，若 $f(y) = F$，则以下情况之一成立：

(i) $f(x) = f(z) = T$。然后，令 $f'(\alpha_i) = T, f'(\beta_i) = F, f'(\gamma_i) = T$，以及 $f'(\delta_i) = F$。(ii) $f(x) = T$和 $f(z) = F$。然后，令 $f'(\alpha_i) = F, f'(\beta_i) = T, f'(\gamma_i) = F$，以及 $f'(\delta_i) = F$。(iii) $f(x) = F$和 $f(z) = T$。然后，令 $f'(\alpha_i) = F, f'(\beta_i) = F, f'(\gamma_i) = T$，以及 $f'(\delta_i) = F$。注意， f' 是 V' 上的一个赋值，使得 ϕ' 的所有子句恰好包含一个真文字。

反向证明中，设 $f'\colon V' \to \{T, F\}$ 是一个赋值，使得 ϕ' 的每个子句恰好有一个真文字；再设 $f\colon V \to \{T, F\}$ 是 f' 在 V 上的限制，即对于所有 $x \in V$， $f(x) = f'(x)$。我们将证明 f 是 ϕ 的一个满足赋值。事实上，假设存在一个子句 $c_i = x \vee y \vee z$，它在 ϕ 中且未被 f 满足。在这种情况下，$f(x) = f(y) = f(z) = F$；又因为 f 是 f' 的限制，所以

- $f'(\alpha_i) = f'(\beta_i) = F$，这是因为 c_i^1；
- $f'(\beta_i) \neq f'(\gamma_i)$，这是因为 c_i^2；并且
- $f'(\gamma_i) = f'(\delta_i) = F$，这是因为 c_i^3，

这就产生了矛盾。

最后，假设存在一个 $2^{o(|V'|)}$ 算法用于 (3,4)-XSAT。那么，由于 $|V'| = n+4m$，可以在 $2^{o(n)}$ 内判定 (3,4)-SAT，因此指数时间假设不成立。◀

接下来证明：给定一个 (3,4)-XSAT 实例，我们可以将公式中的变量和子句划分为若干组，使得出现在同一子句组的子句中的变量分别属于不同的变量组。

▶**引理 47**。设 ϕ 是一个 $(3,4)$-XSAT 实例，其中 V 表示其 n 个变量构成的集合， C 表示其子句构成的集合。此外，设 $b \leq \sqrt{n}$。可以在 $n^{\mathcal{O}(1)}$ 时间内，将 ϕ 的变量划分为 n_V 个两两不相交的集合 V_1、.....、V_{n_V}，每个集合的大小至多为 b；同时，将其子句划分为 n_C 个两两不相交的集合 C_1、.....、C_{n_C}，每个集合的大小至多为 $\sqrt{n}$，其中整数 $n_V =$ 为 $O(n/b)$， $n_C =$ 为 $O(\sqrt{n})$；并且对于任意 $i \in [n_C]$，出现在 C_i 中的任意两个变量都属于不同的变量子集，而任意变量在 C_i 中至多出现于1 个子句中。

证明。我们首先将变量集合 V 划分为互不相交的子集 V_1、.....、V_{n_V}，其中 $n_V = O(n/b)$，使得 $|V_i| \leq b$，并且出现在同一子句中的变量属于不同的 V_i。考虑图 G_1：它为 ϕ 的每个变量设有一个顶点；如果存在 ϕ 中的某个子句同时包含对应的两个变量，则这两个顶点之间有一条边，即 G_1 是 ϕ 的原始图。由于没有子句包含重复变量， G_1 不含任何自环。由于 ϕ 的每个变量至多出现在四个子句中，且这些子句各包含另外两个变量，因此 G_1 的最大度至多为 8。因此，可以使用 9 种颜色对 G_1 进行贪心着色，从而将其划分为颜色各不相同的组，其大小分别为 n_1、.....、n_9，其中 $n_1 + \ldots + n_9 = n$。随后，我们细化上述划分，使每个组的大小至多为 b，从而得到至多

$$n_V = \sum_{i=1}^{9} \left\lceil \frac{n_i}{b} \right\rceil \leq 9 + \sum_{i=1}^{9} \frac{n_i}{b} = 9 + \frac{n}{b}$$

组 $V_1, \ldots, V_{n_V}$。注意，同一子句中出现的变量属于不同的 V_i，因为任意两个这样的变量都相邻于 G_1 ，因此会获得不同的颜色。

接下来，我们将子句集 C 划分为互不相交的子集 $C_1, \ldots, C_{n_C}$，其中 $n_C = \mathcal{O}(\sqrt{n})$，使得 $|C_i| \leq \sqrt{n}$，并且同一组 C_i 中各子句里出现的任意两个变量都属于不同的变量组。为此，考虑图 G_2，其顶点为子句；若两个子句包含来自同一变量组的变量，则在它们之间连一条边。G_2 没有自环，因为同一子句中出现的变量属于不同的 V_i。由于每个子句恰好包含 3 个变量，每个变量组的大小至多为 b，且每个变量最多出现在 4 个子句中，因此 G_2 的最大度至多为 $12b$。因此，我们可以用 $12b+1$ 种颜色对 G_2 进行贪心着色。与之前类似，我们将上述划分细化为 $n_C \leq 12b+1+|C|/\sqrt{n}$ 个子集 $C_1, \ldots, C_{n_C}$，每个子集的大小至多为 $\sqrt{n}$。根据着色的构造可知，若变量 $v \in V_i$ 出现在某个子句 $c \in C_j$ 中，则属于 V_i 的变量不会出现在 $C_j \setminus \{c\}$ 中的任何子句里，而 V_i 中的其他变量不会出现在 c 中。此外，由于 $|C| \leq \frac{4}{3} \cdot n$ 且 $b \leq \sqrt{n}$，可知 $n_C = \mathcal{O}(\sqrt{n})$。◀

▶ **定义 48.** 有限集 U 上的 d-检测族是一个由 U 的子集组成的族 $F \subseteq 2^U$，满足：对于任意两个函数 $f, g: U \to \{0, \ldots, d-1\}$，其中 $f \neq g$，都存在 $S \in F$，使得 $\sum_{x \in S} f(x) \neq \sum_{x \in S} g(x)$。

Lindström [37] 给出了次线性 d-检测族的确定性构造，而 Bonamy 等人 [10] 最早将其用于计算复杂性的研究，并证明了多重着色问题在指数时间假设下的紧下界。以下定理对于证明上述下界至关重要。

▶ **定理 49** ([37])。对于任意常数 $d \in \mathbb{N}$ 和有限集 U，都存在一个定义在 U 上的 d-检测族 F，其大小为 $2^{|U|/\log_d |U|} \cdot (1+o(1))$。此外，$F$ 可以在关于 $|U|$ 的多项式时间内构造出来。

5.2 有界度顶点删除

▶**定理 50**。除非指数时间假设不成立，否则不存在针对有界度顶点删除的 $\mathrm{vc}^{o(\mathrm{vc})} n^{\mathcal{O}(1)}$ 时间算法，其中 vc 表示输入图的最小顶点覆盖的大小。

证明。设 ϕ 是一个 (3,4)-XSAT实例，包含 n 个变量。不失一般性地假设 n 是 4 的幂（如有需要，可向实例中添加虚变量来实现这一点）。利用引理 47，可以在 $n^{\mathcal{O}(1)}$ 时间内得到以下结果：

- 将 ϕ 的变量划分为子集 V_1、……、V_{n_V}，其中 $|V_i| \leq \log n$，且 $n_V = \mathcal{O}(n/\log n)$，
- 将 ϕ 的子句划分为子集 C_1、……、C_{n_C}，其中 $|C_i| \leq \sqrt{n}$，且 $n_C = \mathcal{O}(\sqrt{n})$；同一子句子集中的子句所包含的任意两个变量，均属于不同的变量子集。对于 $i \in [n_C]$，令 $\{C_{i,1}, \ldots, C_{i,n^i_{\mathcal{F}}}\}$ 为 C_i 的一个 4-检测族，其中 $n^i_{\mathcal{F}} = \mathcal{O}(\sqrt{n}/\log n)$，该检测族可依据定理 49 在 $n^{\mathcal{O}(1)}$ 时间内构造得到。此外，令 $n_{\mathcal{F}} = \max_{i \in [n_C]} n^i_{\mathcal{F}}$。定义 $\Delta = n^3$ 和 $k = n_V$。我们将构造一个图 $G = (V, E)$，使得存在大小为 $|S| \leq k$ 的 $S \subseteq V(G)$，且 $G - S$ 的最大度至多为 Δ，当且仅当存在一种赋值，使得 ϕ 的每个子句都恰有一个真文字。

选择构件。对于每个变量子集 V_i，我们定义选择构件图 G_i 如下：引入顶点 κ_i、λ_i 和 v^i_j，其中 $j \in [n]$，添加边 $\{\kappa_i, v^i_j\}$ 以及 $\{\lambda_i, v^i_j\}$ 对所有 $j \in [n]$，为 κ_i 和 λ_i 连接足够多的叶节点，使其度为 $\Delta + 1$。

令 $\mathcal{V}_i = \{v_j^i \mid j \in [n]\}$，对于 $i = 1$、.....、n_V。我们固定一个任意的一一映射，使$\mathcal{V}_i$ 中的每个顶点都对应于变量 V_i 的一个不同赋值。由于$2^{|V_i|} \leq n$，顶点数量足够多，可以唯一编码 V_i 的所有不同赋值。令 $\mathcal{V} = \mathcal{V}_1 \cup \ldots \cup \mathcal{V}_{n_V}$表示所有此类顶点的集合。

子句构件。对于 $i \in [n_C]$，令 C_i 为一个子句子集，$\{C_{i,1}, \ldots, C_{i,n^i}\}$ 为其 4-检测族。对于 4-检测族的每个子集 $C_{i,j}$，引入顶点 $c_{i,j}$ 和 $c'_{i,j}$。若存在变量 $x \in V_p$ 使得 x 出现在某个子句$c \in C_{i,j}$中，且 v_q^p 对应于满足 c的 V_p 的一个赋值，则在 $c_{i,j}$ 与 v_q^p 之间添加一条边。根据引理 47，$c_{i,j}$ 恰有$|C_{i,j}| \cdot \binom{3n}{2}$ 条这样的边：恰有 $3|C_{i,j}|$ 个不同的变量出现在 $C_{i,j}$的子句中，且它们分别属于不同的变量子集；对于每个这样的变量，相应变量子集中的一半赋值会使 $C_{i,j}$中对应的子句得到满足。为$c_{i,j}$ 连接足够数量的叶节点，使其总度为 $\Delta + |C_{i,j}|$。此外，对于 $v \in V$，若 $v \notin N(c_{i,j})$，则令 $v \in N(c'_{i,j})$。注意，此时对 $N(c_{i,j})$，有 $\cup N(c'_{i,j}) \supseteq V$；而对 $N(c_{i,j})$，有 $\cap N(c'_{i,j}) = \emptyset$。最后，为 $c'_{i,j}$连接足够数量的叶节点，使其总度为 $\Delta + (k - |C_{i,j}|)$。

令 $\mathcal{I} = (G, \Delta, k)$ 为有界度顶点删除的一个实例。

▶ **引理 51.** 有 $\mathrm{vc}(G) = O(n/\log n)$。

证明。 注意，删除所有顶点 κ_i、 λ_i、 $c_{i,j}$ 以及 $c'_{i,j}$ 会导出一个独立集。因此，

$$\mathrm{vc}(G) \leq 2n_V + 2\sum_{i=1}^{n_C} n_{\mathcal{F}}^i \leq 2n_V + 2n_C \cdot n_{\mathcal{F}} = \mathcal{O}\left(\frac{n}{\log n} + \sqrt{n} \cdot \frac{\sqrt{n}}{\log n}\right) = \mathcal{O}\left(\frac{n}{\log n}\right),$$

由此可得结论。 ◀

▶**引理 52.** 若 ϕ 是 (3,4)-XSAT 的是实例，则 I 是有界度顶点删除的一个是实例。

证明。设 $f: V_1 \cup \ldots \cup V_{n_V} \to \{T, F\}$ 是一个赋值，使得 ϕ 中的每个子句恰有 1 个真文字。令 S 从每个 V_i 中取出与该赋值在 V_i上的限制相对应的顶点。于是有 $|S| = n_V = k$。我们将证明 $G - S$ 的最大度至多为 Δ。首先注意，在 $G - S$中，任意顶点 v_p^q 至多有 $2 + n_C \cdot n_{\mathcal{F}} \leq \Delta$ 个邻居，而任意顶点 κ_i 和 λ_i 的度为 Δ，因为对所有 $i \in [n_V]$，$|S \cap V_i| = 1$。最后，对于每个顶点 c_i，$_j$ 对应于子句集 C_i、$_j$，有如下结论：由于 f 是一个赋值，且 ϕ 中的每个子句恰有 1 个真文字，并根据引理 47，S 恰好包含 $|C_i$ 个 c_i 的邻居$_j|$、$_j$。在这种情况下，属于 S 的其余 $k - |C_i$、$_j|$ 个顶点是c'_i、$_j$的邻居。◀

▶**引理53。 </style}如果 I 是有界度顶点删除**的一个是实例，那么 ϕ 是(3,4)-XSAT的一个是实例。

证明。 令 $S \subseteq V(G)$，其中 $|S| \leq k$ 且 $G - S$ 的最大度至多为 Δ。我们首先证明 S 在每个集合中都恰好包含一个顶点 $\mathcal{V}_i$。

▷ 命题54。$|S \cap \mathcal{V}_i| = 1$，对所有 $i \in [n_V]$。

证明。 我们首先证明 $|S \cap V(G_i)| = 1$，对所有 $i \in [n_V]$ 均成立。假设此结论不成立。那么，由于 $|S| \leq n_V$，可知存在某个 i 使得 $|S \cap V(G_i)| = 0$。在这种情况下，度$_{G-S}(\kappa_i) = \Delta + 1$，矛盾。最后，假设存在 i 使得， $v \in S$

且 $v \in V(G_i) \setminus \mathcal{V}_i$。由于度$_G(\kappa_i)$ =度$_G(\lambda_i)$ $= \Delta + 1$，且 $v \notin N(\kappa_i) \cap N(\lambda_i)$，因此 κ_i，λ_i 中必有一个不属于 S，且在 $G - S$ 中的度为 $\Delta + 1$，矛盾。◁

现在考虑变量 h 的赋值：$V_1 \cup \ldots \cup V_{n_V} \to \{T, F\}$，具体取决于 $\mathcal{V}_i$ 中的哪个顶点属于 S。

设 C_i，其中 $i \in [n_C]$，是由引理 47 所得划分产生的一个子句子集。设 $f\colon C_i \to \{0, 1, 2, 3\}$ 是一个函数，它为每个子句 $c \in C_i$ 指定满足以下条件的顶点数：$v \in S$ 使得 $v \in V_j$ 对应于满足 c 的赋值 $V_j \to \{T, F\}$。注意，由于每个子句都包含 3 个文字，且 $|V_i \cap S| = 1$，$f(c) \leq 3$ 成立。因此，对于任意 $j \in [n^i_{\mathcal{F}}]$：$|S \cap N(c_{i,j})| \geq |C_{i,j}|$，$|S \cap N(c'_{i,j})| \geq k - |C_{i,j}|$，都有 $\sum_{c \in C_{i,j}} f(c) = |C_{i,j}|$；而 $N(c_{i,j}) \cap N(c'_{i,j}) = \emptyset$ 和 $|S| = k$。

现在将 $g\colon C_i \to \{0, 1, 2, 3\}$ 视为常值函数 $g \equiv 1$。注意，对于所有 $j \in [n^i_{\mathcal{F}}]$，$\sum_{c \in C_{i,j}} f(c) = \sum_{c \in C_{i,j}} g(c)$。由于 $\{C_{i,1}, \ldots, C_{i,n^i\mathcal{F}}\}$ 是一个 4-检测族，因此可推出 $f = g$。于是，对每个子句 $c \in C_i$，都有 $f(c) = 1$，这意味着 S 中存在唯一一个顶点，它对应的部分赋值满足 c。由于 $i = 1, \ldots, n_C$ 是任意的，由此可证 h 是 ϕ 的一个满足赋值。◀

因此，我们可以在多项式时间内构造一个图 G，其顶点覆盖数 vc 为 $= \mathcal{O}(n/\log n)$，这由引理 51 得出；根据引理 52 和 53，判定是否存在大小为 $|S| \leq k$ 的 $S \subseteq V(G)$ 且其最大度至多为 Δ 的 $G - S$，等价于判定是否存在一种赋值，使得 ϕ 的每个子句都恰有一个真文字。在这种情况下，假设存在一个用于有界度顶点删除的 $\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$ 算法，则可以在以下时间内判定 (3,4)-XSAT：

$$\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)} = \left(\frac{n}{\log n}\right)^{o(n/\log n)} n^{\mathcal{O}(1)} = 2^{(\log n - \log\log n)o(n/\log n)} = 2^{o(n)},$$

这与定理 46 所蕴含的指数时间假设相矛盾。◀

5.3 缺陷着色

▶ **定理 55.** 对于任意固定的 $\chi_d \geq 2$，如果存在一种算法，能在时间 $\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$ 内求解缺陷着色，其中 vc 表示输入图的最小顶点覆盖的大小，那么指数时间假设不成立。

证明. 令 ϕ 为一个 (3,4)-XSAT实例，包含 n 个变量。不失一般性，假设 n 是 16 的幂（如有需要，可通过向实例中添加虚变量来实现）。利用引理 47，可以在时间 $n^{\mathcal{O}(1)}$ 内得到以下结果：

- 将 ϕ 的变量划分为子集 V_1、...、V_{n_V}，其中 $|V_i| \leq \log \sqrt[4]{n}$，且 $n_V = \mathcal{O}(n/\log n)$，
- 将 ϕ 的子句划分为子集 C_1、.....、C_{n_C}，其中 $|C_i| \leq \sqrt{n}$ 和 $n_C = \mathcal{O}(\sqrt{n})$，且同一子句子集中的子句里出现的任意两个变量都属于不同的变量子集。对于 $i \in [n_C]$，令 $\{C_{i,1}, \ldots, C_{i,n^i\mathcal{F}}\}$ 为 C_i 的一个 4-检测子集族，其中某个 $n^i_{\mathcal{F}} = \mathcal{O}(\sqrt{n}/\log n)$，由定理 49 在 $n^{\mathcal{O}(1)}$ 时间内生成。此外，令$n_{\mathcal{F}} = \max_{i \in [n_C]} n^i_{\mathcal{F}}$。定义 $\Delta = n_V + n_C \cdot n_{\mathcal{F}}$。我们将构造一个图 $G = (V, E)$，使得 G 存在一个 (χ_d, Δ)-着色，当且仅当存在一种赋值，使得 ϕ 的每个子句都恰好有一个真文字。

主调色板顶点。引入顶点 b_i 和 r_j，其中 $i \in [\Delta+1]$ 和 $j \in [2\Delta+1]$，使它们构成一个完全二部图（即所有边 $\{b_i,\ r_j\}$ 均存在）。我们将其称为主调色板顶点。

选择构件。对于每个变量子集 V_i，我们定义选择构件图 G_i 如下：引入顶点 κ_i、λ_i 和 v^i_j，其中 $j \in [\sqrt[4]{n}]$；添加边 $\{\kappa_i, v^i_j\}$ 和 $\{\lambda_i, v^i_j\}$，对所有 $j \in [\sqrt[4]{n}]$；添加边 $\{\kappa_i, r_p\}$ 和 $\{\kappa_i, b_q\}$，对于 $p \in [\Delta+1]$ 和 $q \in [\Delta-1]$；添加边 $\{\lambda_i, b_p\}$ 和 $\{\lambda_i, r_q\}$，对于 $p \in [\Delta+1]$ 和 $q \in [\Delta-(\sqrt[4]{n}-1)]$。

令 $\mathcal{V}_i = \{v^i_j \mid j \in [\sqrt[4]{n}]\}$，对于 $i = 1, \ldots, n_V$。我们固定一个任意的一一映射，使得 $\mathcal{V}_i$ 中的每个顶点都对应于 V_i 中变量的不同赋值。由于 $2^{|V_i|} \leq \sqrt[4]{n}$，顶点数量足够多，可以唯一编码 V_i 的所有不同赋值。令 $\mathcal{V} = \mathcal{V}_1 \cup \ldots \cup \mathcal{V}_{n_V}$ 表示所有此类顶点的集合。

子句构件。对于 $i \in [n_C]$，令 C_i 为一个子句子集，$\{C_{i,1}, \ldots, C_{i,n^i_{\mathcal{F}}}\}$ 为其 4-检测族。对于 4-检测族中的每个子集 $C_{i,j}$，引入顶点 $c_{i,j}$ 和 $c'_{i,j}$。若存在变量 $x \in V_p$，使得 x 出现在某个子句 $c \in C_{i,j}$ 中，且 v^p_q 对应于满足 c 的 V_p 的一个赋值，则在 $c_{i,j}$ 与 v^p_q 之间添加一条边。根据引理 47，$c_{i,j}$ 恰有 $|C_{i,j}| \cdot \binom{\sqrt[4]{n}}{2}$ 条这样的边：出现在 $C_{i,j}$ 中且各自属于不同变量子集的变量恰有 $3|C_{i,j}|$ 个；对于每个这样的变量，对应变量子集中的赋值有一半会使 $C_{i,j}$ 中对应的子句得到满足。此外，添加边 $\{c_{i,j}, r_p\}$ 和 $\{c_{i,j}, b_q\}$，其中 $p \in [\Delta+1]$ 和 $q \in [\Delta - |C_{i,j}|]$。对于 $c'_{i,j}$，对所有满足 $v \in N(c_{i,j})$ 的 $v \in V$，添加一条边 $\{c'_{i,j}, v\}$。最后，添加边 $\{c'_{i,j}, b_p\}$ 和 $\{c'_{i,j}, r_q\}$，其中 $p \in [\Delta+1]$ 和 $q \in [\Delta - (|C_{i,j}| \cdot \binom{\sqrt[4]{n}}{2} - |C_{i,j}|)]$。

次级调色板顶点。如果 $\chi_{\mathrm{d}} \geq 3$，则引入独立集 $P_i = \{p^i_j \mid j \in [i \cdot \Delta + 1]\}$，对于 $i \in [3, \chi_{\mathrm{d}}]$。我们将 $P = P_3 \cup \ldots \cup P_{\chi_{\mathrm{d}}}$ 中的顶点称为 次级调色板顶点。从每个次级调色板顶点 $p^i_j \in P_i$ 向除属于 P_i 的顶点之外的所有新引入顶点添加一条边。

▶ **引理 56**。有 $\mathrm{vc}(G) = O(n/\log n)$。

证明。注意，删除所有主调色板顶点和次级调色板顶点，以及所有顶点 κ_i、λ_i、$c_{i,\ j}$ 和 $c'_{i,\ j}$ 后，得到一个独立集。因此，

$$\mathrm{vc}(G) \leq \sum_{i=1}^{\chi_{\mathrm{d}}} (i \cdot \Delta + 1) + 2n_V + 2\sum_{i=1}^{n_C} n^i_{\mathcal{F}} \leq \chi^2_{\mathrm{d}} \cdot \Delta + \chi_{\mathrm{d}} + 2n_V + 2n_C \cdot n_{\mathcal{F}} = \mathcal{O}\left(\frac{n}{\log n}\right),$$

结论由此得出。 ◀

▶**引理 57**。对于任意 $\chi_{\mathrm{d}} \geq 2$，如果 ϕ 是 (3,4)-XSAT 的是实例，那么 G 存在一个 $(\chi_{\mathrm{d}}, \Delta)$-着色。

证明。设 $f\colon\ V_1 \cup \ldots \cup V_{n_V} \to \{T,\ F\}$ 是一个赋值，使得 ϕ 的每个子句都恰有 1 个真文字。如果 $\chi_{\mathrm{d}} \geq 3$，则将 P_i 的每个次级调色板顶点着为 i，其中 $i \in [3, \chi_{\mathrm{d}}]$，从而得到那么多个颜色各不相同的独立集。将剩下的两种颜色称为蓝色和红色，并考虑如下着色：

- 顶点 b_i、κ_i 和 $c_{i,\ j}$ 被染成蓝色，顶点
- r_i、λ_i 和 $c'_{i,\ j}$ 被染成红色，

- 对于每个 $\mathcal{V}_i$，与限制在 V_i 上的赋值 f 相对应的单个顶点被染成蓝色，其余顶点被染成红色。

设 B 和 R 分别是 $V(G)$ 中着蓝色和红色的子集。我们将证明，$G[B]$ 和 $G[R]$ 的最大度至多为 Δ。对于顶点 b_i 和 r_i，注意到每个顶点至多有 $n_V + \sum_{i=1}^{n_C} n^i_{\mathcal{F}} \leq \Delta$ 个同色邻居。对于顶点 κ_i 和 λ_i，有 $|\mathbf{V}_i \cap B| = 1$ 且 $|\mathbf{V}_i \cap R| = \sqrt[4]{n} - 1$，因此它们在 $G[B]$和 $G[R]$ 中的度分别等于 Δ。对于每个 $v \in \mathbf{V}_i$，由于它至多有 $1 + \sum_{i=1}^{n_C} n^i_{\mathcal{F}} \leq \Delta$ 个同色邻居：若其着蓝色，则这些邻居来自 κ_i 以及顶点 $c_{i,j}$；否则则来自 λ_i 以及顶点 $c'_{i,j}$。最后，对于每个顶点 $c_{i,j}$，它们对应于子句集 $C_{i,j}$。由于 f 是一个赋值，其中 ϕ 的每个子句都恰有 1 个真文字，且根据引理 47，$\mathbf{V}$ 中 $c_{i,j}$ 恰有 $|C_{i,j}|$ 个邻居着蓝色。在这种情况下，其余位于 $\mathbf{V}$ 中的邻居着红色。因此可知，每个 $c_{i,j}$ 以及 $c'_{i,j}$ 恰有 Δ 个同色邻居。◀

▶ **引理 58**。对于任意 $\chi_{\mathrm{d}} \geq 2$，如果 G 具有一个$(\chi_{\mathrm{d}}, \Delta)$-着色，那么 ϕ 是$(3,4)$-XSAT的一个是实例。

证明。令 $q^*: V(G) \to [\chi_{\mathrm{d}}]$ 为$(\chi_{\mathrm{d}}, \Delta)$-着色 G。我们先给出以下断言。

▷断言 59。将 q^*限制在 $V(G-P)$ 上，得到的是 $G-P$ 的一个 $(2,\Delta)$-着色。

证明。如果 $\chi_{\mathrm{d}} = 2$，则这是成立的，因为 $P = \emptyset$。假设 $\chi_{\mathrm{d}} \geq 3$，并令 $i \in [3, \chi_{\mathrm{d}}]$，$G_i = G - \bigcup_{k=i+1}^{\chi_{\mathrm{d}}} P_k$，而 $G_{\chi_{\mathrm{d}}} = G$。我们将证明：若 G_i 存在一个 (i,Δ)-着色，则将同一着色限制在 G_{i-1} 上后，对于 $i \in [3, \chi_{\mathrm{d}}]$，它是 G_{i-1}的一个 $(i-1,\Delta)$-着色。该结论对 χ_{d} 成立：注意，由于 $P_{\chi_{\mathrm{d}}}$ 包含 $\chi_{\mathrm{d}} \cdot \Delta + 1$ 个顶点，至少有 $\Delta + 1$ 个顶点具有相同的颜色。在这种情况下，由于这些顶点都与 G 中不属于 $P_{\chi_{\mathrm{d}}}$ 的任意其他顶点相连，因此图的其余部分至多使用 $\chi_{\mathrm{d}} - 1$ 种颜色；于是将 q^* 限制在 $V(G - P_{\chi_{\mathrm{d}}})$ 上后，得到的是 $G - P_{\chi_{\mathrm{d}}}$ 的一个 $(i-1,\Delta)$-着色。现在假设 G_i 存在一个 (i,Δ)-着色，其中$i \in [3, \chi_{\mathrm{d}} - 1]$。那么，注意，由于 P_i 包含 $i \cdot \Delta + 1$ 个顶点，至少有$\Delta + 1$ 个顶点具有相同的颜色。在这种情况下，由于这些顶点都与 G_i 中不属于 P_i 的任意其他顶点相连，因此图的其余部分至多使用 $i - 1$ 种颜色；于是将 q^* 限制在 $V(G_{i-1})$ 上后，得到的是 G_{i-1} 的一个 $(i-1,\Delta)$-着色。◁

令 $q: V(G-P) \to \{b, r\}$ 为 q^* 在 $G-P$上的限制，其中 $q(v)$表示 $G-P$中每个顶点 v 的（蓝色或红色）颜色。不失一般性，假设 $q(b_1) = b$。

- 我们首先证明，对于所有 $i \in [\Delta + 1]$，$q(b_i) = b$。假设情况并非如此，即存在 i使得 $q(b_i) = r$。对于所有 $j \in [2\Delta + 1]$，均有 $r_j \in N(b_1) \cap N(b_i)$；又因为每个顶点 r_j 的颜色不是蓝色就是红色，所以至少存在$\Delta + 1$ 个颜色相同的此类顶点。在这种情况下，b_1 或 b_i 有 $\Delta + 1$ 个同色邻居，这与假设矛盾。由于每个顶点 r_i、λ_i和 $c'_{i,j}$ 都有 $\Delta + 1$ 个蓝色邻居，因此可得$q(r_i) = q(\lambda_i) = q(c'_{i,j}) = r$。由于每个顶点 κ_i 和 $c_{i,j}$ 都有 $\Delta + 1$ 个红色邻居，因此可得 $q(\kappa_i) = q(c_{i,j}) = b$。在这种情况下，任何 r_i 至多有 $n_V + \sum_{i=1}^{n_C} n^i_{\mathcal{F}} < \Delta$ 个同色邻居，这是因为顶点 λ_i 和 $c'_{i,j}$。对称地，任何 b_i 至多有 $n_V + \sum_{i=1}^{n_C} n^i_{\mathcal{F}} < \Delta$ 个同色邻居。
-

对于每个子集 $\mathcal{V}_i$，恰有一个顶点被染成蓝色：如果所有顶点都被染成红色，那么 λ_i 总共会有 $\Delta-(\sqrt[4]{n}-1)+\sqrt[4]{n}>\Delta$ 个红色邻居；而如果有多个顶点（例如 $p>1$）被染成蓝色，那么 κ_i 就会有 $\Delta-1+p>\Delta$ 个蓝色邻居。

现在，根据 V_i 中哪个顶点被染成蓝色，考虑 ϕ 的变量赋值 $h\colon V_1\cup\ldots\cup V_{nV}\to\{T,F\}$。设 C_i，其中 $i\in[n_C]$ 是由引理 47 所述划分得到的子句子集。设 $f\colon C_i\to\{0,1,2,3\}$ 为一个函数，它将每个子句 $c\in C_i$ 映射为满足以下条件的蓝色顶点数：$v\in V$ 满足 $v\in V_j$ 对应于一个满足 c 的赋值 $V_j\to\{T,F\}$。注意，每个子句都包含 3 个文字，而在每个子集 V_i 中恰有一个顶点被染成蓝色，因此可得 $f(c)\leq 3$。对于任意 $j\in[n^i_{\mathcal{F}}]$: c_i，均有 $\sum_{c\in C_{i,j}}f(c)=|C_{i,j}|$；并且 $_j$ 和 $c'_{i,j}$ 在 V 中具有相同的邻居集合，其大小为 $|C_{i,j}|\cdot\binom{3\sqrt[4]{n}}{2}$，其中至多 $|C_{i,j}|$ 个顶点可以是蓝色，而至多 $|C_{i,j}|\cdot\binom{3\sqrt[4]{n}}{2}-|C_{i,j}|$ 个顶点可以是红色。

现在考虑 $g\colon C_i\to\{0,1,2,3\}$ 为常函数 $g\equiv 1$。注意，对于所有 $j\in[n^i_{\mathcal{F}}]$，$\sum_{c\in C_{i,}}{}_jf(c)=\sum_{c\in C_{i,j}}g(c)$。由于 $\{C_{i,1},\ldots,C_{i,n^i\mathcal{F}}\}$ 是一个 4-检测族，这意味着 $f=g$。因此，对于每个子句 $c\in C_i$，都有 $f(c)=1$，也就是说，$\mathcal{V}$ 中恰有一个蓝色顶点对应于一个满足 c 的部分赋值。由于 $i=1,\ldots,n_C$ 是任意的，这就完成了证明：h 是 ϕ 的满足赋值。◀

因此，我们可以在多项式时间内构造一个图 G，其顶点覆盖数 $\mathrm{vc}=\mathcal{O}(n/\log n)$，这由引理 56 得出；并且根据引理57和58，判定 G 是否存在一个 $(\chi_{\mathrm{d}},\Delta)$-着色，等价于判定是否存在一个赋值，使得 ϕ 的每个子句都恰有一个真文字。在这种情况下，假设存在一个用于缺陷着色的 $\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}$ 算法，就可以在以下时间内判定 (3,4)-XSAT

$$\mathrm{vc}^{o(\mathrm{vc})}n^{\mathcal{O}(1)}=\left(\frac{n}{\log n}\right)^{o(n/\log n)}n^{\mathcal{O}(1)}=2^{(\log n-\log\log n)o(n/\log n)}=2^{o(n)},$$

这与定理 46 所得出的指数时间假设相矛盾。◀

6 结论

在本文中，我们从参数化复杂性的角度，深入研究了有界度顶点删除和缺陷着色的复杂性。特别是，我们精确确定了以一些最常用的结构参数参数化时这两个问题的复杂性。作为未来研究的一个方向，我们考虑这样一个问题：当 $\chi_{\mathrm{d}}=2$ 时，是否可以为有界度顶点删除以及缺陷着色获得 $n^{o(\mathrm{fvs})}$ 下界，其中 fvs 表示输入图的最小反馈顶点集的大小。

参考文献

1 James A. Andrews 和 Michael S. Jacobson。关于色数的一种推广。数论大会论文集，47:33–48，1985年。 doi:10.7155/jgaa.00418。2 Dan Archdeacon。关于曲面上图的缺陷着色的一则注记。图论杂志，11(4):517–519，1987。 doi:10.1002/jgt.3190110408。

4 Balabhaskar Balasundaram、Sergiy Butenko 和 Illya V. Hicks。社会网络分析中的团松弛：最大 k-团问题。运筹学，59(1):133–142，2011年。 doi:10.1287/opre.1100.0851。 5 Rémy Belmonte、Michael Lampis 和 Valia Mitsou。参数化（近似）缺陷着色。SIAM离散数学杂志，34(2):1084–1106，2020年。 doi:10.1137/18M1223666。 6 Rémy Belmonte、Michael Lampis 和 Valia Mitsou。完美图类上的缺陷着色。离散数学理论与计算机科学，24，2022 年。 doi:10.46298/dmtcs.4926。 7 Nadja Betzler、Hans L. Bodlaender、Robert Bredereck、Rolf Niedermeier 和 Johannes Uhlmann。通过顶点删除使指定顶点达到最小度。算法学，68(3):715–738，2014年。 doi:10.1007/s00453-012-9695-6。 8 Nadja Betzler、Robert Bredereck、Rolf Niedermeier 和 Johannes Uhlmann。以树宽为参数的有界度顶点删除。离散应用数学，160(1-2):53–60，2012年。doi:10.1016/j.dam.2011.08.013。 9 Nadja Betzler 和 Johannes Uhlmann。选举中的候选人控制及相关有向图问题的参数化复杂性。理论计算机科学，410(52):5425–5442，2009年。doi:10.1016/j.tcs.2009.05.029。 10 Marthe Bonamy、Lukasz Kowalik、Michal Pilipczuk、Arkadiusz Socala 和 Marcin Wrochna。多重着色复杂度的紧下界。ACM计算理论汇刊，11(3):13:1–13:19，2019年。 doi:10.1145/3313906。 11 Glencora Borradaile 和 Hung Le。针对树分解上 r-支配问题的最优动态规划。载于：第11届参数化与精确计算国际研讨会（IPEC 2016），LIPIcs第63卷，第8:1–8:23页。Schloss Dagstuhl - Leibniz信息学中心，2016年。 doi:10.4230/LIPIcs.IPEC.2016.8。 12 Ilkyoo Choi 和 Louis Esperet。图在曲面上的非正常着色。图论杂志，91(1):16–34，2019年。 doi:10.1002/jgt.22418。 13 Lenore J. Cowen、Robert Cowen 和 Douglas R. Woodall。曲面上图的缺陷着色：划分为有界度子图。图论杂志，10(2):187–195，1986年。doi:10.1002/jgt.3190100207。 14 Marek Cygan、Fedor V. Fomin、Lukasz Kowalik、Daniel Lokshtanov、Dániel Marx、Marcin Pilipczuk、Michal Pilipczuk 和 Saket Saurabh。参数化算法。Springer，2015年。 doi:10.1007/978-3-319-21275-3。 15 Marek Cygan、Stefan Kratsch 和 Jesper Nederlof。通过完美匹配基快速检验哈密顿性。ACM杂志，65(3):12:1–12:46，2018年。 doi:10.1145/3148227。 16 Marek Cygan、Jesper Nederlof、Marcin Pilipczuk、Michal Pilipczuk、Johan M. M. van Rooij 和 Jakub Onufry Wojtaszczyk。以树宽为参数的连通性问题的单指数时间求解。ACM算法汇刊，18(2):17:1–17:31，2022年。 doi:10.1145/3506707。 17 Marek Cygan 和 Marcin Pilipczuk。精确与近似带宽。理论计算机科学，411(40-42):3701–3713，2010年。 doi:10.1016/j.tcs.2010.06.018。 18 Reinhard Diestel。图论，数学研究生教材第173卷。Springer，2017年。 doi:10.1007/978-3-662-53622-3。 19 Louis Dublois、Michael Lampis 和 Vangelis Th. Paschos。混合支配集的新算法。离散数学理论与计算机科学，23(1)，2021 年。 doi:10.46298/dmtcs.6824。 20 Louis Dublois、Michael Lampis 和 Vangelis Th. Paschos。上支配集：针对路径宽度的紧算法与亚指数近似。理论计算机科学，923:271–291，2022年。 doi:10.1016/j.tcs.2022.05.013。 21 Michael R. Fellows、Jiong Guo、Hannes Moser 和 Rolf Niedermeier。Nemhauser–Trotter 局部优化定理的一种推广。计算机与系统科学期刊，77(6):1141–1158，2011年。 doi:10.1016/j.jcss.2010.12.001。 22 Jacob Focke、Dániel Marx、Fionn Mc Inerney、Daniel Neuen、Govind S. Sankar、Philipp Schepper 和 Philip Wellnitz。针对有界树宽图上广义支配集计数的紧复杂度界。载于：2023年ACM-SIAM离散算法研讨会论文集（SODA 2023），第3664–3683页。工业与应用数学学会，2023年。 doi:10.1137/1.9781611977554.ch140。

52 两个有界度问题的结构参数化再探

23 Jacob Focke、Dániel Marx 和 Pawel Rzazewski。对有界树宽图上的列表同态进行计数：紧确的复杂度界。ACM算法汇刊，20(2)，2024年。doi:10.1145/3640814. 24 Toshihiro Fujito。用于节点删除问题的统一近似算法。离散应用数学，86(2-3):213–231，1998年。 doi:10.1016/S0166-218X(98)00035-3. 25 Toshihiro Fujito。通过拟阵匹配近似有界度删除。载于《算法与复杂性——第10届国际会议》，CIAC 2017，计算机科学讲义第10236卷，第234–246页，2017年。 doi:10.1007/978-3-319-57586-5_20. 26 Robert Ganian、Thekla Hamm、Viktoriia Korchemna、Karolina Okrasa 和 Kirill Simonov。以团宽为参数的图同态的细粒度复杂性。ACM算法汇刊，2024年。 doi:10.1145/3652514. 27 Robert Ganian、Fabian Klute 和 Sebastian Ordyniak。关于有界度顶点删除问题的结构参数化。Algorithmica，83(1):297–336，2021年。 doi:10.1007/s00453-020-00758-8. 28 Carla Groenland、Isja Mannens、Jesper Nederlof 和 Krisztina Szilágyi。以割宽为参数，对着色和连通边集计数的紧确界。载于第39届理论计算机科学国际研讨会 STACS 2022，LIPIcs第219卷，第36:1–36:20页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2022年。 doi:10.4230/LIPIcs.STACS.2022.36. 29 Tesshu Hanaka、Ioannis Katsikarelis、Michael Lampis、Yota Otachi 和 Florian Sikora。参数化可定向删除。Algorithmica，82(7):1909–1938，2020年。 doi:10.1007/s00453-020-00679-6. 30 Frédéric Havet、Ross J. Kang 和 Jean-Sébastien Sereni。单位圆盘图的非正常染色。Networks，54(3):150–164，2009年。 doi:10.1002/net.20318. 31 Lars Jaffke 和 Bart M. P. Jansen。图着色问题的细粒度参数化复杂性分析。离散应用数学，327:33–46，2023年。 doi:10.1016/j.dam.2022.11.011. 32 Ioannis Katsikarelis、Michael Lampis 和 Vangelis Th. Paschos。结构参数、紧确界及 (k, r)-中心的近似算法。离散应用数学，264:90–117，2019年。 doi:10.1016/j.dam.2018.11.002. 33 Ioannis Katsikarelis、Michael Lampis 和 Vangelis Th. Paschos。结构参数化的 d-散布集。离散应用数学，308:168–186，2022年。 doi:10.1016/j.dam.2020.03.052. 34 Michael Lampis。利用图宽度的参数化近似方案。载于《自动机、语言与编程——第41届国际研讨会》，ICALP 2014，计算机科学讲义第8572卷，第775–786页。Springer，2014年。 doi:10.1007/978-3-662-43948-7_64. 35 Michael Lampis。团宽上的着色的更精细紧确界。SIAM离散数学杂志，34(3):1538–1558，2020年。 doi:10.1137/19M1280326. 36 Michael Lampis 和 Manolis Vasilakis。两个有界度问题的结构参数化再探。载于第 31 届欧洲算法年会 ESA 2023，LIPIcs第274卷，第77:1–77:16页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2023年。doi:10.4230/LIPIcs.ESA.2023.77. 37 Bernt Lindström。数论中的一个组合问题。加拿大数学通报，8(4):477–490，1965年。 doi:10.4153/CMB-1965-034-2. 38 Daniel Lokshtanov、Dániel Marx 和 Saket Saurabh。有界树宽图上的已知算法很可能已是最优的。ACM算法汇刊，14(2):13:1–13:30，2018年。doi:10.1145/3170442. 39 Daniel Lokshtanov、Pranabendu Misra、M. S. Ramanujan、Saket Saurabh 和 Meirav Zehavi。FPT问题的固定参数近似。载于《2021年ACM-SIAM离散算法研讨会论文集》，SODA 2021，第199–218页。工业与应用数学学会，2021年。 doi:10.1137/1.9781611976465.14. 40 Benjamin McClosky 和 Illya V. Hicks。最大 k-团问题的组合算法。组合优化期刊，23(1):29–49，2012年。 doi:10.1007/s10878-010-9338-2.

41 Robert T. Moenck。实用快速多项式乘法。载于第三届ACM符号与代数操作研讨会论文集，SYMSAC 1976，第136–148页。ACM，1976年。 doi:10.1145/800205.806332. 42 Hannes Moser、Rolf Niedermeier 和 Manuel Sorge。寻找最大 k-团簇的精确组合算法与实验。组合优化期刊，24(3):347–373，2012年。 doi:10.1007/s10878-011-9391-5. 43 Naomi Nishimura、Prabhakar Ragde 和 Dimitrios M. Thilikos。用于顶点覆盖非平凡推广问题的快速固定参数可处理算法。离散应用数学，152(1-3):229–245，2005年。 doi:10.1016/j.dam.2005.02.029. 44 Karolina Okrasa 和 Pawel Rzazewski。有界树宽图的图同态问题的细粒度复杂性。SIAM 计算机学报，50(2):487–508，2021年。 doi:10.1137/20M1320146. 45 Michael Okun 和 Amnon Barak。一种近似节点删除问题的新方法。信息处理快报，88(5):231–236，2003年。 doi:10.1016/j.ipl.2003.08.005. 46 Venkatesh Raman、Saket Saurabh 和 Sriganesh Srihari。广义支配的参数化算法。载于《组合优化与应用》，第二届国际会议，COCOA 2008，计算机科学讲义第5165卷，第116–126页。Springer，2008年。 doi:10.1007/978-3-540-85097-7_11. 47 Thomas J. Schaefer。可满足性问题的复杂性。载于第 10 届 ACM 计算理论年会论文集，第 216–226页。ACM，1978年。 doi:10.1145/800133.804350. 48 Atsushi Takahashi、Shuichi Ueno 和 Yoji Kajitani。混合搜索与适当路径宽度。理论计算机科学，137(2):253–268，1995年。 doi:10.1016/0304-3975(94)00160-K. 49 Craig A. Tovey。一个简化的 NP 完全可满足性问题。离散应用数学，8(1):85–89，1984年。 doi:10.1016/0166-218X(84)90081-7. 50 Bas A. M. van Geffen、Bart M. P. Jansen、Arnoud A. W. M. de Kroon 和 Rolf Morel。有界割宽平面图上动态规划的下界。图算法与应用期刊，24(3):461–482，2020年。 doi:10.7155/jgaa.00542. 51 Johan M. M. van Rooij。用于树分解连接操作的通用卷积算法。载于《计算机科学——理论与应用——第16届俄罗斯国际计算机科学研讨会》，CSR 2021，计算机科学讲义第12730卷，第435–459页。Springer，2021年。 doi:10.1007/978-3-030-79416-3_27. 52 Mingyu Xiao。一种用于有界度顶点删除的参数化算法。载于《计算与组合——第22届国际会议》，COCOON 2016，计算机科学讲义第9797卷，第79–91页。Springer，2016年。 doi:10.1007/978-3-319-42634-1_7. 53 Mingyu Xiao。关于 Nemhauser–Trotter 局部优化定理的一种推广。计算机与系统科学期刊，84:97–106，2017年。 doi:10.1016/j.jcss.2016.08.003.