arXiv:2303.06864v1 [cs.CC] 13 Mar 2023

无冲突支配集着色：q-子集平方着色

V P Abidha[1], Pradeesha Ashok、Avi Tomar、Dolly Yadav

班加罗尔国际信息技术学院，印度 *dia*

摘要

图G 的平方着色 是指对图的顶点进行着色，使得任意两个距离不超过二的不同顶点都获得不同颜色。本文开始研究一种相关的着色问题，称为图的子集平方着色 。大体而言，图的子集平方着色研究的是用q 种颜色对图的一个支配集进行平方着色，目标是优化所用颜色数。这也推广了广为研究的高效支配集问题。我们证明，对于所有q 值，q-子集平方着色问题即使在二分图和弦图上也是 NP-困难。我们进一步研究以若干结构参数为参数时该问题的参数化复杂性。此外，我们还给出了若干图类进行子集平方着色所需颜色数的界。

关键词：
图着色、平方着色、子集平方着色、参数化算法、支配集

1. 引言

顶点着色是图论领域中的一个重要问题。对于图图 $G(V, E)$，图图 G 的顶点着色是指一个从顶点集V 映射到颜色集合的函数f 。根据对该函数施加的约束，图着色问题有不同类型。一种非常常见的图着色问题是正常着色，即要求任意两个相邻顶点被赋予不同的颜色。此外，还存在多种其他顶点着色变体，例如和谐着色、σ着色、度量着色和无圈着色。除了理论上的研究价值，图着色问题还受到多个领域应用的推动，例如编译器中的寄存器分配、作业调度、交通网络等。参见 [13] ，了解图着色的详细介绍。

许多图着色问题的灵感源自通信领域中的一个问题，即信道分配 问题。在这里，存在发射器 $v_1, v_2, \ldots$、v_n 以及一个发射器

电子邮箱地址： abidha.vp@iiitb.ac.in (V P Abidha), pradeesha@iiitb.ac.in (Pradeesha Ashok),Avi.Tomar@iiitb.ac.in (Avi Tomar), dolly.yadav@iiitb.ac.in (Dolly Yadav)[I]第一作者获得了 TCS研究学者计划的资助。本文的初步版本曾发表于俄罗斯国际计算机科学研讨会（CSR）2022 [2]。

预印本已提交至爱思唯尔 *2023*年*3*月*14*日

可能由于多种原因而受到另一个发射器的干扰。现在的目标是为这些发射器分配频率，以确保信号能够清晰接收。这可以用一个图来表示，其中每个顶点对应一个发射器，而发射器之间的干扰则通过图中相应顶点之间的距离来刻画。分配给发射器的频率对应于分配给相应顶点的颜色。在20世纪90年代，Griggs 和 Yeh [24]提出了一个概念：为顶点分配颜色（等价地，非负整数），使任意两个顶点的颜色赋值取决于它们之间的距离是否至多为2。这称为图的 L(h, k)-染色。图 G 的一个着色 c 是 L(h.k)-染色，当且仅当对于任意两个顶点 u、v ∈V(G)，若 u 和 v 的距离为1，则 |c(u) –c(v)| ≥ h；若 u 和 v 的距离为2，则 |c(u) – c(v)| ≥ k。因此，L(1, 0)-着色就是正常着色本身。基于 h 和 k 的不同取值，这一问题的其他变体也得到了广泛研究 [12]。注意，L(1, 1)-染色是指用非负整数为顶点着色，使相邻顶点的颜色至少相差1，距离为2的顶点的颜色也至少相差1。这种图着色也称为平方着色 [11, 25]，因为它等价于对图的平方进行正常着色。

我们开始研究一种称为子集平方着色的平方着色变体。其定义如下：

定义 1. 设图 $G = (V,E)$为一个无向图。染色函数$c : V(G) \to \{c_0, c_1, \cdots, c_q\}$ 称为q-子集平方着色，当且仅当它满足以下约束：

- 对于每个顶点v 和每种颜色$c_i, 1 \le i \le q$，有 $|c^{-1}(c_i) \cap N[v]| \le 1$。
- 顶点v 在$N[v]$, 中，颜色为c_0 的顶点至多有$deg(v)$个，其中$deg(v)$表示v的度。

顶点v 的开邻域$N(v)$由所有与v相邻的顶点组成，即$N(v) = \{u \in V|(u,v) \in E\}$；而$v$ 的闭邻域为$N[v] = \{v\} \cup N(v)$。

直观而言，将颜色c_0 赋给顶点v，相当于v 未着色。本文中，我们交替使用“顶点未着色”和“顶点着色c_0, ”这两种说法。同理，若顶点使用集合 $\{c_1 \cdots c_q\}$中的某种颜色着色，则称该顶点已着色。因此，该定义意味着每个顶点的闭邻域中至少有一个已着色的顶点，且闭邻域内没有重复的颜色。注意，已着色顶点构成图G的一个支配集。因此，子集平方着色等价于使用q 种颜色对图的支配集进行平方着色。

对于给定的图图 G，令 χ_{ssc}(图 G) 表示满足存在q-子集平方着色图 G的q 的最小值。我们还研究以下算法问题。给定一个图图 G，q-子集平方着色问题定义如下：

输入： 图图 G 和$q \in \mathbb{N}$。**问题：** 图图 G 能否
进行q-子集平方着色?

子集平方着色的概念此前曾在计算几何中的一个经典问题——艺术画廊问题——的背景下得到研究。给定一个多边形 P，以及两个由P内的点构成的集合M 和图 G，艺术画廊问题要求找到图 $G' \subseteq$ 图 G，使得M 中的每个点都能被图 G'中的至少一个点看到。受机器人学应用的启发，Erickson 和 LaValle [17] 提出了彩色艺术画廊问题。该问题旨在找到一个子集图 $G' \subseteq$ 图 G，使得图 G' 可以用q 种颜色着色，并且M 中的每个点$m \in$ 都能被图 G' 中的至少一个点看到；此外，图 G' 中所有能看到m 的点都必须着上不同的颜色。

很容易看出，当M 和图 G 是相同的有限集时，彩色艺术画廊问题可以归约为可见图的子集平方着色。我们认为，信道分配领域还存在许多其他应用场景，在这些场景中，图的子集平方着色会很有用。

研究q-子集平方着色问题的另一个动机是，与平方着色所需的颜色数相比，许多图在进行子集平方着色时所用的颜色数要少得多。例如，完全图、星图、轮图等进行平方着色时需要$O(n)$种颜色，而进行子集平方着色时只需一种颜色。这在许多应用中会很有用，因为颜色数对应着一种需要优化的资源。

现在我们来探讨一些与q-子集平方着色问题相关的问题。

相关问题：和谐着色问题由 Hopcroft 和 Krishnamoorthy [26] 于1983年首次提出，其定义如下：图G的和谐色数记作$h(G)$，是指能够分配给G 的顶点的最少颜色数，并满足每个顶点恰有一种颜色、相邻顶点颜色不同，且任意两条边对应的颜色对均不相同。随后，Yue Li Wang 等人 [31]发展了d– 局部和谐色问题的概念，将和谐色问题加以推广。d-局部和谐（或简称d-和谐）色问题规定：对于任意顶点，只有距离d 以内的边才需要满足颜色对互不相同的要求。因此，1-和谐色问题与平方着色问题相同。

对于给定图的高效支配集问题，在研究子集平方着色问题时也值得关注。高效支配集同时是独立支配集和完美支配集。完美支配集P 是指对于每个顶点，其开邻域中恰有一个邻点属于P；而独立支配集I 满足这样的条件：I 中的顶点构成一个独立集。具体而言，高效支配集是子集平方着色在$q = 1$时的一个特例。

另一个相关问题是无冲突着色问题 [3, 1] ，它指的是对一个支配集进行着色，使得每个顶点的闭邻域中至少有一种颜色仅出现一次。

2. 结果讨论

本节将概述我们的结果。

我们已经提到，高效支配集问题是q-子集平方着色的一个特例。已知高效支配集问题是NP难问题 [5]。因此，当$q = 1$时，q-子集平方着色问题是NP难问题。我们证明，即使在平面二分图上，q-子集平方着色问题在$q = 2$ 时仍是NP难问题；并且对于所有q值，q-子集平方着色问题即使在二分图上也都是NP难问题。我们还证明，q-子集平方着色问题在弦图上是NP难问题。

我们研究 q-子集平方着色问题的参数化复杂性。对于任何问题，解的大小都是一个值得研究的参数。对于 q-子集平方着色问题，这个参数是q，即所用的颜色数。然而，事实证明，这比W-困难问题要难得多，因为该问题不太可能存在运行时间为 $f(q)n^{g(q)}$ 的算法。

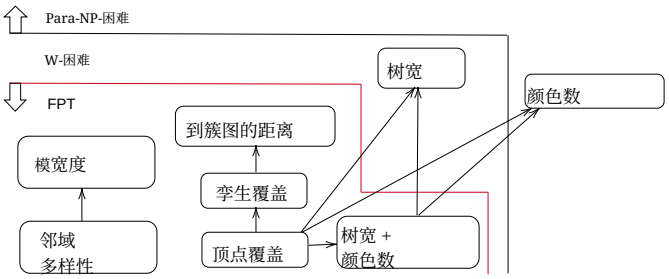


图1：参数化结果总结。

引理1. q-Subset Square Colouring 问题以 q 为参数是 *para-NP*-困难的。

该引理源于这样一个事实：即使在 $q = 1$ 的情况下，q-子集平方着色问题仍是*NP*-困难的。此外，下一个结果表明，即使对于直径为2的图，该问题仍然是*W*-困难的。

定理2. q-Subset Square Colouring 问题以 q 为参数时，在直径为2的图上是 $W[2]$–困难的。

证明。对于图G，$\chi_{ssc}(G)$由最小支配集的大小界定，D 是G的最小支配集。当图的直径为 2 时，任意D 中的所有顶点彼此之间的距离至多为 2。因此，对顶点进行子集平方着色时，同一种颜色不能重复使用超过一次。这意味着 $\chi_{ssc}(G)$ 等于最小支配集的大小。已知最小支配集问题在直径为 2 [27] 的图上是W[2]-困难的。因此结论得证。□

关于定理2，我们指出，q-子集平方着色问题在直径为 2 [28] 的平面图上可在多项式时间内求解。

接下来，我们考察几个结构参数。图问题中一个研究充分的结构参数是图的树宽。已有研究表明，若以树宽为参数，若干困难问题都属于固定参数可处理。然而，若以树宽为参数，q-子集平方着色问题是*W*-困难的。接下来，我们将树宽与颜色数作为一个组合参数来考察，并证明该问题属于固定参数可处理。这一结果是通过一种固定参数可处理算法设计中的标准技术——基于树宽的动态规划——证明的。

接下来，我们考虑可能大于树宽的结构参数。一个研究较为成熟的参数是图的顶点覆盖大小。我们给出了一个固定参数可处理算法，用于解决以顶点覆盖大小为参数的q-子集平方着色问题；此外，我们还证明，除非$NP \subseteq coNP/poly$，否则以顶点覆盖大小为参数的q-子集平方着色问题不存在多项式核。顶点覆盖大小通常是一个较大的参数，尤其是在稠密图中。因此，我们研究一种在稠密图上取值较小的参数，称为邻域多样性。它也是一种可在多项式时间内计算的参数。我们给出了一个固定参数可处理算法，用于解决以邻域多样性为参数的q-子集平方着色问题，并且还证明该问题存在一个

多项式核。进一步地，我们考虑一种邻域多样性的推广，称为模块宽度，并证明以模块宽度为参数的q-子集平方着色问题属于固定参数可处理。我们还考虑一种可证明小于顶点覆盖大小的参数，称为到簇图的距离，并证明以到簇图的距离为参数的q-子集平方着色问题也属于固定参数可处理。此外，我们还考虑一种称为孪生覆盖的参数，其取值通常介于到簇图的距离和顶点覆盖大小之间。由于以到簇图的距离为参数的q-子集平方着色问题属于固定参数可处理，以孪生覆盖为参数的q-子集平方着色问题也属于固定参数可处理。不过，我们给出了一个运行时间更优的算法。

对于这些问题，我们的目标仅仅是判断问题是否属于固定参数可处理，因此不试图优化运行时间。图1总结了q-子集平方着色问题在参数化复杂性方面的结果。

图类	χ_{ssc}**:上界**	χ_{ssc}**:下界**
路径 P_n	1	1
圈 C_{3n}	1	1
圈 C_{3n+1} 或C_{3n+2}	2	2
完全图 K_n	1	1
完全二分图$G_{n,m}$	2	2
二部置换图$G_{n,m}$	4	3
直径为2的平面图	3	2
余图 G_n	2	2
阈值图G_n	1	1
毛毛虫图 $G(n, r_1, r_2, \cdots r_t)$	3	3
树 $T(V, E)$	$O(\Delta)$	$O(\Delta - 1)$
树 $T(V, E)$	$O(n)$	$\Omega(\sqrt{n})$
分裂图 $T(V, E)$	$O(n)$	$\Omega(\sqrt{n})$

表1：不同图类中 χ_{ssc} 的界汇总。

接下来，我们研究当图 G 属于某些图类时，χ_{ssc}(图 G)的界。显然，各种图类（包括许多稀疏和稠密图类），例如完全图、簇图、星图、轮图、路径图、环、网格图等，都是q-子集平方可着色的，其中q 为常数。然而，考虑树时，我们证明存在具有n 个顶点的树，需要$O(\sqrt{n})$种颜色才能进行子集平方着色。由于树是二分图的一个子类，这一结果也适用于二分图类。不过，我们证明了一个定义明确的二分图子类——二分置换图——是4-子集平方可着色的。我们进一步证明，阈值图类是1-子集平方可着色的。注意，阈值图属于分裂图与余图的交集。我们发现，余图和阈值图一样，可以用常数种颜色进行子集平方着色；但也存在需要$O(\sqrt{n})$种颜色才能进行子集平方着色的分裂图。结果汇总见表1。

3. NP–完全性

本节将证明，对于q的所有取值，q-子集平方着色问题都是NP完全的。注意，对于$q = 1$，q-子集平方着色问题是NP-困难的，而且显然该问题属于NP。因此，当$q = 1$时，q-子集平方着色问题是*NP*-完全的。现在，我们考虑$q = 2$。

定理 3。 q-子集 平方 着色 问题（其中 $q = 2$ 为 *NP*完全问题），即使在平面二分图上也是如此。

证明。我们将从平面精确 3 集覆盖（X3C）问题进行归约。

平面 X3C（精确 3 集覆盖）：

输入：一个有限集 X，满足 $|X| = 3n$；以及由 X 的三元子集组成的集合族 S，满足 $|S| = m$。问题：S 是否包含 X 的一个精确覆盖？也就是说，是否存在一个子族 $S' \subseteq S$，使得 X 中的每个元素恰好出现在 S' 的一个成员中?

在平面 X3C 问题中，我们增加了如下约束：二分图 M 满足 $V(M)$ 对应于 $X \cup S$，且 $E(G)$ 为 $\{(x, s)|$，其中 $x \in X$，$s \in S$，$x \in s\}$，且该图为平面图。

设 (U, S) 是平面 X3C 问题的一个实例，其中 $\mathrm{U} = \{\mathrm{u}_1, \mathrm{u}_2, .., \mathrm{u}_{3n}\}$ and $\mathcal{S} = \{\mathrm{S}_1, \mathrm{S}_2, . .., \mathrm{S}_m\}$。我们按如下方式构造一个平面二分图 G：对于每个元素 u_i，我们在图 G 中添加一个顶点 x_i，并将其与图 G 中的一个元素构件 D_i 相连。对于 $1 \le \mathrm{i} \le 3\mathrm{n}$，$\mathrm{D}_i$ 是一棵以顶点 d_i为根的树，如图 3(a) 所示。d_i 的两个子节点分别与三个叶节点相连。对于每个集合 $\mathrm{S\,j}$（$1 \le \mathrm{j} \le \mathrm{m}$），我们添加一个集合构件 T j，其中顶点 t_j与两个叶节点 $\mathrm{v}j$ 和 v'_j相连。我们还添加一个调色板构件 P，其中两个顶点 p_1 和 p_2 彼此相邻，且各自与三个度为一的顶点相连。见图 3(b)。

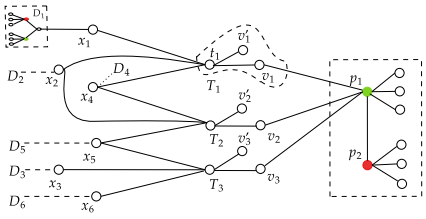


图 2：由精确 3 集覆盖实例构造的图 G_S，其集合为 $S_1 = \{x_1, x_2, x_4\}$、$S_2 = \{x_2, x_4, x_5\}$ 和 $S_3 = \{x_3, x_5, x_6\}$。

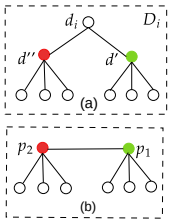


图 3：（a）元素构件，（b）调色板构件

此外，对于 $1 \le i \le 3n$、$1 \le j \le m$，如果 (U, S) 中有 S_j，则在 G 中的 x_i和tj之间添加一条边。对于所有 $1 \le j \le m$，我们还在vj 和$p_1 \in P$之间添加一条边。

我们声称，(U, S) 存在平面精确 3 覆盖，当且仅当$G(V, E)$ 存在使用两种颜色的子集平方着色。

假设图 G 存在一种使用两种颜色的子集平方着色，χ。由于顶点 p_1 和 p_2各有三个度为一的邻点，因此它们将被着成不同的颜色。不失一般性，假设p_1 被着成绿色，而p_2被着成红色。同样，在每个元素构件中，d_i 的两个子节点被着成不同的颜色。因此，顶点d_i 和x_i 无法着色。所以x_i 必须由某个集合构件中的相邻顶点支配。此外，对于 $1 \le j \le m$，由于p_1 被着成绿色，因此与其距离为二的tj 若要着色，就

必须着成红色。因此，对于每个顶点x_i，都存在唯一一个被着成红色且与x_i相邻的t_j。现在不难看出，$\{S_j|t_j$被着成红色的构成$\chi\}$的一个平面精确 3 覆盖（U，S）。见图 2。

反过来，假设 (U, S) 有一个平面精确 3 覆盖 $\mathcal{S}'$。现在我们给出一种使用两种颜色的图 G 的子集平方着色。分别用绿色和红色为顶点 p_1 和 p_2 着色。对于 $1 \le i \le 3n$，在元素构件 D_i中，用不同颜色为 d_i 的两个子节点着色。对于 $1 \le j \le m$，如果 $S_j \in \mathcal{S}'$，则将 t_j 着成红色；否则，用任意一种颜色为顶点$v_{j'}$ 着色。由此得到G的一个子集平方着色。

现在，根据平面 X3C 问题的 NP 困难性 [21]，结论得证。

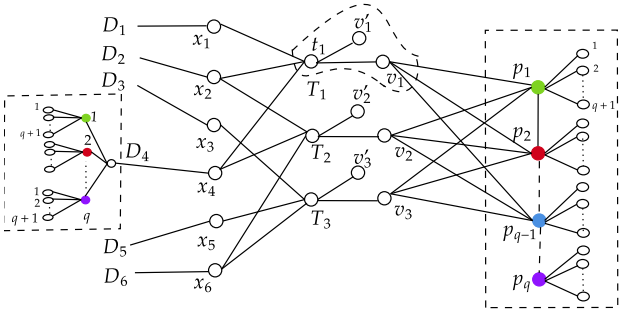
定理 4. q-Subset Square Colouring 问题（其中 $q > 2$）是 *NP* 完全问题，即使限定在二分图上也是如此。

证明。证明与定理 5 的证明非常相似。我们从 X3C 问题进行归约。与前一个证明类似，我们构造一个二分图G，对应于 X3C 问题的一个实例 (U, S)。这里的元素构件D和调色板构件P略有不同。

对于每个元素 u_i，我们在图 G 中添加一个顶点 x_i，并将其与一个元素构件 D_i相连，使D_i 成为以顶点 d_i 为根的树，且d_i 的每个q 个子节点都与$q+1$ 个叶节点相连。对于每个集合 S_j，$1 \le j \le m$，我们按照定理 5 的证明中给出的方法添加一个集合构件 T_j。此外，我们还添加一个调色板构件 P，其中包含一条含q 个顶点p_1, p_2，$\cdots$、p_q的路径，并且路径上的每个顶点都连接到$q+1$ 个度为一的顶点。

此外，对于 $1 \le i \le 3n$、$1 \le j \le m$，如果 $u_i \in S_j$属于 (U, S)，则在图 G中添加连接x_i和t_j的边。对于所有 $1 \le j \le m$，我们还添加连接v_j与顶点p_1, p_2，$\cdots$、$p_{q-1} \in P$的边。参见图 4。

根据定理 5 证明中给出的相同论证，我们现在可以看出，图G 存在使用q种颜色的子集平方着色，当且仅当 (U, S) 存在一个精确 3 集覆盖。



S 1，2， 2，4， 3 =

图 4：由精确 3 集覆盖实例构造的图$G\{3, 5, 6\}$，其实例包含$S_1 = \{x_4\}$、$S_2 = \{x_6\}$和$S.x$。

现在，我们证明，对于所有q值，q-子集平方着色问题在弦图上是 NP完全的。

定理 5. 对于任意 q 值，弦图上的 q-Subset Square Colouring 问题都是 *NP*完全的。

证明。我们从顶点覆盖问题进行归约。设 (G, k) 是顶点覆盖问题的一个实例，其中 |V| = n，|E| = m。我们按如下方式构造一个弦图图 G′ = (V′, E′)。对于图 G 中的每个顶点 $v_i \in V(G)$，我们在 G′ 中引入一个顶点 v'_i；类似地，对于图 G 中的每条边 $e_j \in E(G)$，我们在 G′中引入一个顶点 e'_j。对于每个对应于边 $e_j = (i, h) \in E$ 的顶点 e'_j，我们添加边 (v'_i, e'_j) 和 (v'_h, e'_j)。此外，我们将 G′中的每对顶点 v'_i相连，从而形成一个大小为 |V(G)| 的团 C。

此外，对于每个顶点 e'_j，其中 $1 \le j \le |E(\text{图 } G)|$，我们添加一棵子树 T_j。子树 T_j包含一个根节点 r_j，它有 k 个子节点，并且每个子节点都与 $k+1$ 个叶节点相邻。我们在 T_j 与 e'_j之间添加一条边，将子树与其连接起来；这条边连接 r_j和 e'_j。

除诱导出树的子图外，构造图图 G' 还包含一个团，以及一组与团中顶点相邻的顶点e'_j。因此，图图 G' 中的每个诱导环都恰有三个顶点。由此很容易看出，图 G' 是一个弦图，并且可以在多项式时间内构造。

观察 6. 在每棵 T_j中，任何 k-子集平方着色都会用 k 种不同的颜色为 r_j 的 k 个子节点着色。

该结论源于这样一个事实：根节点 r_j 与 k 个子节点相邻，而每个子节点又与 $k+1$ 个叶节点相邻。因此，为了支配这些叶节点，我们需要用 k 种不同的颜色为 r_j 的 k 个子节点着色。

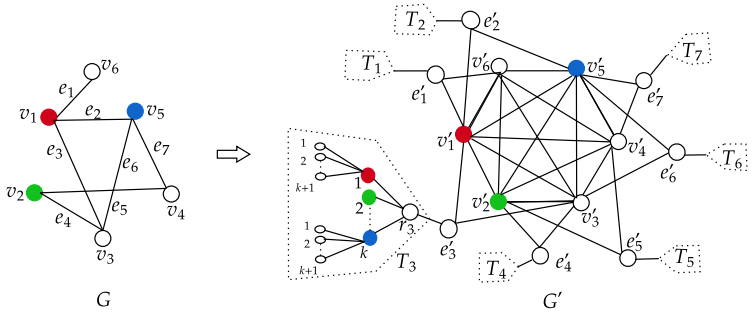


图 5：弦图 $G'(V',E')$ 是由图 $G(V,E)$ 构造而成的。

引理 7. 图 G 有一个大小为 k 的顶点覆盖，当且仅当 G' 存在一种 k-子集平方着色。

证明。假设G' 存在一种k-子集平方着色。由观察 6 可知，在G'的k-子集平方着色中，子树中的所有根节点以及所有e'_j 顶点都不能着色。因此，要支配顶点e'_j，我们需要给与它相邻的团C中的某个顶点着色。我们不能在C 中重复使用任何颜色，因此C 中至多有k 个顶点着色。显然，着色顶点的集合对应于图G的一个顶点覆盖。反方向的证明类似，也很容易看出。见图 5。

现在，该结论可由顶点覆盖问题的 NP 完全性得出。

4. 参数化复杂性

本节中，我们研究q-子集平方着色问题的参数化复杂性。

参数化问题是一个语言 $L \subseteq \Sigma^* \times \mathbb{N}$，其中 Σ 是一个固定的有限字母表。对于实例（x，k）$\in \Sigma^* \times \mathbb{N}$，$k$ 称为参数。复杂性类固定参数可处理包含所有固定参数可处理问题，这些问题都存在一个算法、一个可计算函数 $f: \mathbb{N} \to \mathbb{N}$，以及一个常数 c，使得给定（x，k）$\in \Sigma^* \times \mathbb{N}$，该算法能够正确判定（$x$，$k$）$\in L$ 是否成立，且运行时间有界于 $f(k) \cdot |(x, k)|^c$ [16]。参数化问题的难解性理论根据问题的复杂性将其排列成一个称为 W-层级的层次结构。其组织形式为固定参数可处理 $\subseteq W[1] \subseteq W[2] \cdots$。在标准复杂性理论假设下，$W[i]$-困难问题不允许存在固定参数可处理算法，其中 $i > 0$。如需详细了解参数化复杂性，请参阅 [15]。

如引言所述，当$q = 1$时，高效支配集问题是子集平方着色的一个特例。然而，寻找高效支配集的问题是 NP-困难的 [5]。因此，q-子集平方着色问题也是 NP-困难的。因此，我们探索能否设计固定参数可解的算法来求解图的子集平方着色问题。但对于标准参数 q，该问题是 para-NP-困难的，因为即使$q = 1$ 时，该问题仍是 NP-困难的（见引理1）。因此，我们研究以结构参数为参数的子集平方着色问题。

4.1. 以树宽为参数

我们首先考虑图的一种常见且经过充分研究的结构参数——树宽。下面先定义树宽。

树分解： [15] 图图 G 的一种树分解是指一棵树T，其中每个顶点$i \in T$ 都被分配一个顶点集合$X_i \subseteq V$，称为袋，使得 $\bigcup_{i \in T} X_i = V$，其中存在某个

性质：

- 若$u \in X_i$且$u \in X_k$，则对于T中从i到k的路径上的所有j，都有$u \in X_j$。
- 对于任意边$e(u,v) \in E$(图 G)，存在一个$i \in T$，使得u、$v \in X_i$。

树分解T 的宽度是T 中最大袋的大小减一；图G的树宽（记作 $\tau(G)$）是G所有可能树分解的最小宽度。

树宽用于衡量一个图与类树结构的接近程度。

定理 8. 以树宽为参数的 q-Subset Square Colouring 问题是 $W[1]$-困难的。

我们给出证明概要。与子集平方着色密切相关的问题是 L(1,1)-标记问题。 [18] 的定理 5 证明了，平方着色问题以树宽为参数时是$W[1]$-难的。他们给出了一种参数化归约，将均衡着色问题的任意实例（图 G,r）（以颜色数r 为参数）转换为平方着色问题的一个实例（H, τ）（以树宽 τ 为参数），使得 τ 有一个用r 种颜色的均衡着色，当且仅当H有一个使用r 种颜色的平方着色。

定义 2：以 r 为参数的均衡着色：
输入： 一个图图 G 和一个正整数r。
问题：是否存在一种顶点着色c ，使图G 至多使用r 种颜色，且任意两个颜色类的大小之差至多为 1?

我们按如下方式修改 [18] 中给出的归约，以获得一个q-子集平方着色问题的实例。对于约简图H，我们为每个$V(H)$ 中的$v \in$ 添加 $q+1$ 个度为 1 的顶点，得到图H'。这样一来，在H' 的任何q-子集平方着色中，$V(H)$ 中的所有顶点都必须着色；否则，若$V(H)$ 中的$v \in$ 未着色，我们就必须为图H' 中新添加的$q+1$ 个v 的邻点逐一着色，这显然会超出给定的q 种颜色的预算。此外，注意这一修改不会改变H 的树宽，且其树宽由r的某个函数所界定。现在可以看出，图G 存在一个使用r 种颜色的均衡着色，当且仅当H' 存在子集平方着色。因此，我们给出了一个参数化归约：将以r 为参数的均衡着色问题归约到以 τ 为参数的q-子集平方着色问题。

4.2. 按树宽和颜色数参数化
现在我们考虑以树宽和颜色数为参数的q-子集平方着色问题，并给出一个固定参数可处理 算法。我们采用一种称为树宽动态规划的标准技术，为固定参数可处理性提供构造性证明。我们使用一种经过修改的树分解，称为优美树分解。

优美树分解：[15]带有指定根的树分解称为优美树分解 ，如果满足：

- 所有叶节点和根节点的袋都是空的，即$X_l = X_r = \phi$，其中r 是根节点，l 是叶节点。

- 树分解中的其他每个节点都属于以下三种类型之一：

引入节点： 引入顶点节点t 恰有一个子节点t' ，满足$X_t = X_{t'} \cup \{v\}$ ，其中某个$v \notin X_{t'}$。

遗忘节点： 遗忘节点t 恰好有一个子节点t' ，使得$X_t = X_{t'} \setminus \{w\}$ ，其中$w \in X_{t'}$。

合并节点： 合并节点t 恰好有两个子节点t_1 和t_2, ，使得$X_t = X_{t_1} = X_{t_2}$。

引入边节点： 引入边节点标记有一条边$uv \in E$(图 G)，其中u、$v \in X_t$ ，且恰好有一个子节点t' ，使得$X_t = X_{t'}$。

请注意，我们假设每条边恰好被引入一次。如果某个合并节点同时包含u和v，并且边uv 属于E(图 G)，那么可以注意到，边uv 将在该合并节点上方的子树中被引入。

优美树分解使我们能够逐一添加边和顶点，并相应地执行操作。这种树分解变体仍然有$O(\tau \cdot n)$个节点，其中 τ 是图G的树宽。

以下结果已为人所知。

命题 **9.** 给定一个图 G，在时间 $2^{O(\tau)}n$ 内，我们可以计算出 G 的一个优美树分解 (T, X)，使得 $|V(T)| \in |V(G)|^{O(1)}$ ，且其宽度至多为 5τ，其中 τ 是图 G 的树宽 *[7]*。

对于树分解中的每个节点 t，我们关联一个子图 G_t，它是图G 的子图，定义如下：$G_t = (V_t, E_t = \{e : e$ 在以 t 为根的子树中引入$\})$。这里，V_t 是以t为根的子树中所有袋的并集。

定理 10. 当参数为输入图的树宽 τ 和颜色数 q 时，q-子集 平方 着色 是固定参数可处理的。

证明。我们给出一种基于图 G 的优美树分解 $(T, \mathcal{X})$ 上的动态规划算法，该分解使用命题 9 在 $2^{O(\tau)}n$ 时间内计算，其宽度至多为 5τ，其中 τ 是图 G 的树宽。我们针对图 G_t 中的 $t \in V(T)$ 定义子问题。我们考虑通过映射 $f : X_t \to \{B, W, R\}$ 对袋 X_t 进行划分。为简单起见，我们分别将各划分中的顶点称为黑色、白色和灰色。每个顶点还通过函数 $\hat{c} : X_t \to \{c_0, c_1, ..., c_q\}$ 赋予另一种颜色，并通过函数 $\Gamma: X_t \to \{0, 1, 1\}^q$ 赋予一个 q 长元组。粗略地说，当将“部分”平方着色限制在图 G_t 和 X_t 中的顶点上时，这些函数将决定其表现形式。这里，$c(v)$ 表示赋予 v 的颜色，而 $c(v) = c_0$ 表示 v 未着色。此外，$\Gamma(v)[i]$ 表示 v 的闭邻域中是否存在颜色为 c_i 的顶点（该顶点可以位于当前图中，也可以位于“未来”部分）。此外，$\Gamma(v)[i] = 1$ 表示顶点 v 在图 G_t 中的闭邻域内存在一个颜色为 c_i 的顶点；$\Gamma(v)[i] = 1$ 表示顶点 v 的闭邻域内存在一个颜色为 c_i 的顶点，该顶点尚未出现在图 G_t 中，但将在“未来”出现；而 $\Gamma(v)[i] = 0$ 表示 v 的闭邻域中不存在颜色 c_i。我们稍微滥用一下记号，将 $\Gamma(v)[c_i]$ 和 $\Gamma(v)[i]$ 交替使用。接下来，我们将详细介绍函数 f、c 和 Γ。

黑色，用B表示。每个黑色顶点v 都被赋予一个颜色$c(v) \neq c_0$，属于子集平方着色。

灰色，用R表示。灰色顶点v 未着色且未被支配，即$c(v) = c_0$，并且对每个$i \in [q]$,，它满足 $\Gamma(v)[i] \in \{0, 1\}$。

白色，用W表示。既非黑色也非灰色的顶点v 是白色顶点。注意，对于白色顶点v，$c(v) = c_0$，且存在$i \in [q]$, 使得 $\Gamma(v)[i] = 1$。

若对每个顶点$v \in X_t$，以下条件均成立，则元组(t, c, Γ, f)是有效的：

1. $f(v) = B \implies c(v) \neq c_0$ 且 $\Gamma(v)[c(v)] = 1$，ˆ 2. $f(v) = R \implies c(v) = c_0$ 且 $\Gamma(v)[i] \in \{0, 1\}$，$\forall i \in \{1...q\}$，以及 3. $f(v) = W \implies c(v) = c_0$ 且 $\Gamma(v)[i] = 1$，其中某个 $i \in \{1...q\}$。

对于节点 $t \in V(T)$，对于每个有效元组 (t, c, Γ, f)，都有一个表项，记作 $D[t, c, \Gamma, f]$。当且仅当存在 $col : V_t \to \{c_0, c_1, ..., c_q\}$（其中 c_0 表示不赋予颜色），使得 $D[t, c, \Gamma, f] = true$：

1. $col|_{X_t} = c$；2. 对于每个 $v \in X_t$，以及每个 $i \in \{1, 2, ...q\}$，若 $\Gamma(v)[i] = 1$，则恰有一个顶点 $u \in N_{G_t}[v]$,满足 $col(u) = c_i$，ˆ 3. 对于每个 $v \in X_t$，以及每个 $i \in [q]$，若 $\Gamma(v)[i] \in \{0, 1\}$，则不存在顶点 $u \in N_{G_t}[v]$, 满足 $col(u) = c_i$；4. 对于每个 $v \in V_t \setminus X_t$，至少存在一个顶点 $u \in N_{G_t}[v]$, 满足 $col(u) \neq c_0$,，并且对于所有此类顶点 u，其他每个 $u' \in N_{G_t}[v]$ 都满足 $col(u') \neq col(u)$。

在上述情形中，这种着色col 称为（t，c，Γ，f）-良好着色。（在任何时候查询到无效元组时，其值默认为假。）注意，$D[r, \emptyset, \emptyset, \emptyset] =$

真，当r是树分解的根时，当且仅当图G存在子集平方着色 -最多使用q种颜色。

我们将$f_{v\to\gamma}$定义为函数，其中 $\gamma \in \{B、W、R\}$；当$f_{v\to\gamma}(x) = f(x)$，且$x \neq v$时，函数值为此；否则$f_{v\to\gamma}(x) = \gamma$。类似地，我们定义函数$c_{v\to c_i}$和$\Gamma_{v[i]\to\alpha}$，其中$\alpha \in \{0, \hat{}1, 1\}$。现在我们继续定义$D$各值的递归公式。

叶节点。对于叶节点t，有$X_t = \emptyset$。因此，唯一的条目是$D[t, \emptyset, \emptyset, \emptyset]$。此外，根据定义，有$D[t, \emptyset, \emptyset, \emptyset] =$ 真。

引入顶点节点。设t为引入顶点节点，其子节点t'满足$X_t = X_{t'} \cup \{v\}$，其中$v \notin X_{t'}$。由于顶点v在图 GG_t中是孤立的，因此可得以下递推式。

$$D[t, c, \Gamma, f] = \begin{cases} D[t', c_{|X'}, \Gamma_{|X'}, f_{|X'}] \text{ if} & f(v) = B \text{ and } \Gamma(v)[c_i] \in \{0, \hat{1}\}, \forall\, c_i \neq c(v) \\ D[t', c_{|X'}, \Gamma_{|X'}, f_{|X'}] \text{ if} & f(v) = R \\ \text{False} & \text{otherwise} \end{cases}$$

引入边节点。令 t 为一个标记有边 u^*v^* 的引入边节点，并令t'为它的子节点。因此，图 $G_{t'}$ 没有边 u^*v^*，而图 G_t 有。考虑互不相同的u、$v \in \{u^*, v^*\}$。

1. 若 f (u) = B，f (v) = W 且 Γ(v)[c(u)] = 1，则设 D[t, c, Γ, f] = D[t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, $f_{v\to R}$] ∨ D[t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, $f_{v\to W}$]（如果任何一项无效，则结果为 false）。2. 若 f (u) = f (v) = B 且 Γ(v[c(u)]) = Γ(u[c(v)]) = 1，则设 D[t, c, Γ, f] = D[t′, c, $\Gamma_{v[c(u)]\to\hat{1},u[c(v)]\to\hat{1}}$, f]。3. 若 {f (u), f (v)} ∩ {B} = ∅，则 D[t, c, Γ, f] = D[t′, c, Γ, f]。4. 若以上条件均不成立，则 D[t, c, Γ, f] = false。

引理 11. 引入边节点的递推式是正确的。

证明。注意，X_t 中除u 和v 以外的所有顶点都不受引入边u^*v^*的影响。此外，如果u 和v 都不是黑色，那么边uv 不会影响任何顶点的支配。因此，该元组对应的D 值与子节点中的值相同。

考虑 f (u) = B、f (v) = W 且 Γ(v)[c(u)] = 1 的情形。我们证明，只要 D[t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, f $_{v\to R}$] 和 D[t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, f] 中至少一个为 true，D[t, c, Γ, f] = 就为 true。设 D[t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, f $_{v\to R}$] = 为 true。令 col : $V_{t'} \to \{c_0, \ldots, c_q\}$ 为图 $G_{t'}$ 中的一个 (t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, $f_{v\to R}$)-良好着色。注意，col(v) = c_0，并且不存在顶点 v′ ∈ $N_{G_{t'}}$[v] 使得 col(v′) = c(u)。因此，col 也是一个 (t, c, Γ, f)-良好着色，因为 u 是 N_{G_t}[v] 中唯一满足 col(u) = c(u) 的顶点。于是 D[t, c, Γ, f] = 为 true。类似地，可以证明，当 D[t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, f] = 为 true 时，任意 (t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, f)-良好着色也是一个 (t, c, Γ, f)-着色。反过来，假设 D[t, c, Γ, f] = 为 true，并令 col : $V_t \to \{c_0, \ldots, c_q\}$ 为一个 (t, c, Γ, f)-良好着色。因此，u 是 N_{G_t}[v] 中唯一满足 col(u) = c(u) 的顶点。如果存在一个 $c_i \neq$ c(u)，使得 v 在 N_{G_t}[v] 中有一个邻点 v′，且 col(v′) = c_i，那么 col 是一个 (t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, f)- 着色；否则，col 是一个 (t′, c, $\Gamma_{v[c(u)]\to\hat{1}}$, $f_{v\to R}$)- 着色。其他情形的正确性也可用类似论证证明。

遗忘节点。设t 是一个遗忘节点，其子节点为t'，且$X_t = X_{t'} \setminus \{v\}$，其中$v \in X_{t'}$。由于顶点$v$不会再出现在$t$上方任何节点的袋中，$v$ 必定是黑色或白色（否则，我们将该项设为*false*）。

$$D[t, c, \Gamma, f] = \bigvee_{\substack{1\le i\le q \\ \alpha\in\{0,1\}^q}} (D[t', c_{v\to c_0}, \Gamma_{v\to\alpha}, f_{v\to W}] \vee D[t', c_{v\to c_i}, \Gamma_{v\to\alpha}, f_{v\to B}])$$

合并节点。令 t 表示该合并节点。令 t_1 和 t_2 为 t 的子节点。我们知道，$X_t = X_{t_1} = X_{t_2}$ 和 X_t 在图 G_t、G_{t_1} 和 G_{t_2}中诱导出一个独立集。如果对于每个 $v \in X_t$，以下条件都成立，则称元组对 $[t_1, f_1, c1, \Gamma_1]$ 和 $[t_2, f_2, c2, \Gamma_2]$ 是 $[t, f, c, \Gamma]$-相容的。

- 若$f(v) = B$，则 $(f_1(v), f_2(v)) = (B, B)$，且$c_1(v) = c_2(v) = c$(v</style)。
- 若$f(v) = W$，则 $(f_1(v), f_2(v)) \in \{(W, R), (R, W), (W, W)\}$。
- 若$f(v) = R$，则 $(f_1(v), f_2(v)) = (R, R)$。
- 若 $\Gamma(v)[i] = 0$，则 $(\Gamma_1(v)[i], \Gamma_2(v)[i]) \in \{(0, 0)\}$ 对于 $1 \le i \le q$。
- 若 $\Gamma(v)[i] = 1$，则 $(\Gamma_1(v)[i], \Gamma_2(v)[i]) \in \{(1, \hat{}1), (\hat{}1, 1)\}$ 对于 $1 \le i \le q$。 $\hat{}$ $\hat{}$ $\hat{}$
- 若 $\Gamma(v)[i] = 1$，则 $(\Gamma_1(v)[i], \Gamma_2(v)[i]) \in \{(1, 1)\}$ 对于 $1 \le i \le q$。

容易看出，当顶点 v 在 X_t 中属于黑色划分时，它在 X_{t_1} 和 X_{t_2} 中也必须是黑色，且分配给 v 的颜色 c(v) 在这些袋中相同。类似地，我们也可以理解灰色划分中的顶点。若顶点 v 属于白色划分，则意味着它恰好在袋 X_{t_1} 和 X_{t_2} 中的一个或两个中被支配。然而，如果 v 在这两个袋中都被支配，那么支配它的颜色不同，这由 $\Gamma(v)[i]$ 处理。对于不同的 q 长元组，$\Gamma(v)[i]$ 的值设为 1。因此，顶点要属于白色划分，必须满足以下三种组合之一：{(W, R), (R, W), (W, W)}。现在考虑 $\Gamma(v)[i]$ 在 X_t 中可取的值。如果颜色 i 在子节点 t_1 和 t_2，中的任一个所对应的 N[v] 中都不存在，那么它自然也不会出现在 G_t 中的 N[v] 中，这意味着 $\Gamma(v)[i] = 0$。然而，$\Gamma(v)[i] = 1$ 表示 G_t 中的 N[v] 内存在一个颜色为 i 的顶点。我们进一步注意到，颜色 i 恰好出现在子节点 X_{t_1} 或 X_{t_2} 之一中，而不会同时出现在两者中，因为若颜色 i 同时出现在两个子节点中，就意味着 G_t 中的 N[v] 内存在两个颜色为 i 的顶点。根据优美树分解的性质，X_t 在 G_t 中诱导出一个独立集，由此可得这一结论。$\hat{}$ $\Gamma(v)[i]$ 的下一个可能值是 1。要使其成立，颜色 i 目前尚未出现在 G_t 中，但之后会在树分解中出现。

我们设置 $D[t, c, \Gamma, f] = \bigvee (f_1, f_2) (D[t_1, c1, \Gamma_1, f_1] \wedge D[t_2, c2, \Gamma_2, f_2])$，其中 $[t_1, f_1, c1, \Gamma_1]$ 和 $[t_2, f_2, c2, \Gamma_2]$ 是 $[t, f, c, \Gamma]$-相容的。

我们已描述了 $D[\cdot]$ 各个取值的递归公式。注意，每个条目都可以在不超过 $2^{O(q\tau)} \cdot q^{O(\tau)} n^{O(1)}$ 的时间内计算出来。此外，对于节点 $t \in V(T)$，有效条目的数量不超过 $2^{O(q\tau)} \cdot q^{O(\tau)} n^{O(1)}$，且 $V(T) \in n^{O(1)}$。因此，我们可以得出，该算法的总体运行时间不超过 $2^{O(q\tau)} n^{O(1)}$。

4.3. 以顶点覆盖的大小为参数

本节证明，以顶点覆盖的大小为参数的q-子集平方着色属于固定参数可处理。令X 为图 GG的一个顶点覆盖，$|X| = k$。首先证明以下结果。

引理 12. 对图 G 的顶点进行子集平方着色所需的颜色数，至多为图 G 的顶点覆盖的大小。

证明。我们知道，每条边 $e \in E(G)$ 的至少一个端点属于图G的顶点覆盖。考虑一个图 G，其顶点覆盖为 X，大小为 k。首先，若存在孤立顶点，则给所有孤立顶点分配 k 种颜色中的一种。接下来，使用k 种不同的颜色为 X 中的每个顶点着色。由于我们为 X 中的每个顶点着色，因此每个顶点 $v \in V$ 要么已着色，要么其开邻域中至少有一个已着色的顶点。此外，X 中的每个顶点都使用不同的颜色着色，从而避免颜色重叠。结论得证。

由于子集平方着色所需的树宽和颜色数都受 $|X|$限制，以下定理可由定理 10 得出。

定理 13. 以顶点覆盖大小为参数的 q-子集 平方 着色 问题是固定参数可处理的。

现在，我们讨论以顶点覆盖大小为参数的q-子集平方着色问题是否存在多项式核。当$q = 1$ 时，根据 [4]中给出的结果，q-子集平方着色问题存在多项式核。我们证明，以顶点覆盖大小为参数的q-子集平方着色问题不存在多项式核，除非$NP \subseteq coNP/poly$，其中 $q \geq 2$。这里我们使用或交叉组合技术来证明核化下界。为简化证明，我们先证明$q = 2$ 时的结果，然后将结果推广到$q \geq 3$。

定义 3. 或-交叉组合 [15]。设$L \subseteq \Sigma^*$ 是一个语言，且$Q \subseteq \Sigma^* \times N$ 是一个参数化语言。若存在一个多项式等价关系R 和一个算法 $\mathcal{A}$（称为交叉组合），满足以下条件，则称L 交叉组合到Q 中。算法 $\mathcal{A}$ 以一列字符串$x_1, x_2, \cdots, x_t \in \Sigma^*$ 作为输入，这些字符串关于R等价；算法在关于 $\sum_{i=1}^{t} |x_i|$的多项式时间内运行，并输出一个实例 $(y,k) \in \Sigma^* \times N$，使得：

- $k \leq p(\max_{i=1} |x_i| + \log t)$，其中某个多项式为 $p(.)$，并且
- $(y,k) \in Q$ 当且仅当至少存在一个索引 i 使得 $x_i \in L$。

我们采用 Bodlaender 等人提出的或交叉组合技术[8]。

定义 4. 如果一个 NP-困难问题 L或交叉组合到一个参数化问题 Q，那么除非 $NP \subseteq coNP/poly$ 成立且多项式层次塌缩 [8]，否则 Q 不存在多项式核。

4.3.1. 核化：以顶点覆盖大小为参数的2-S子集平方着色 问题

我们通过从 NP-困难问题团进行或交叉组合，证明以顶点覆盖大小为参数的 2-子集平方着色问题不存在多项式核，除非 $NP \subseteq coNP/poly$ 成立。该框架与 [10, 3] 中使用的框架类似。

在此归约中，我们考虑团问题的t 实例$X_1, X_2, \cdots, X_t$。如果每个X_i 的顶点数均为n，且要求寻找一个大小为k 的团，则称这些实例等价。令$\{v_{l1}, v_{l2}, \cdots, v_{ln}\}$ 为任意实例X_l中的顶点集合，其中 $1 \leq l \leq t$。本节假设c_1 = 为红色，且c_2 =为蓝色。

在创建用于 2-子集平方着色的实例图 G 之前，我们先定义一个特殊顶点 。

特殊顶点： 如果$N(v)$包含三个度为1的顶点，则顶点v 是一个特殊顶点。 .
我们用两个同心、半径不同的圆表示特殊顶点（图6）。



图6：分别以红色和蓝色着色的特殊顶点。

观察 14. 对于 G 的任意2-子集平方着色 C，若 G 中的顶点 v 是特殊顶点，则 $C(v) \neq c_0$。

我们按以下步骤构造一个 2-子集平方着色的实例图 G。我们添加两个特殊顶点 R 和B，并在R 和B之间添加一条边。这确保顶点R 和B在任何 2-子集平方着色中都获得不同的颜色。不失一般性，假设R 获得红色，B获得蓝色。

现在我们添加一组顶点 $\{p_{i,j} | i \in [k], j \in [n]\}$，称为图 G 的网格顶点。直观地说，这些顶点用于选出与某个输入实例中的团相对应的顶点。

为了完成构造，我们需要创建以下构件。我们将这些构件连接到网格顶点，并连接到特殊顶点 R 和 B。

对于每个 $i \in [k],$，我们添加一个路径构件 $\mathcal{P}1_i$，其中包含三个顶点 r'_i、r_i 和 r''_i。在每个 $\mathcal{P}1_i$ 中，都存在边 (r'_i, r_i) 和 (r_i, r''_i)，其中r_i是一个特殊顶点。这些顶点在路径构件中构成一条路径。见图7。此外，通过在r'_i与R之间添加一条边，将 $\mathcal{P}1_i$ 连接到顶点R。另外，对于所有$j \in [n]$，通过r_i 将 $\mathcal{P}1_i$ 连接到网格顶点，方法是添加边 $(r_i, p_{i,j})$。

对于每个 j $\in$ [n],，我们使用一个类似的路径构件，$\mathcal{P}2$j，其中包含三个顶点 w j、w'_j和 w''_j。在每个 $\mathcal{P}2_j$中，都存在边 (w j, w'_j) 和 (w'_j, w''_j)，其中 w'_j是一个特殊顶点。见图8。此外，我们添加一个顶点 w，并通过添加边 (w j, w) 和 (w, R)，将 $\mathcal{P}2_j$连接到顶点 R。还要连接顶点 w''_j和 R。除此之外，我们通过添加边 (w$_i$, p$_i$, j)，将 $\mathcal{P}2_j$ 与网格顶点相连，其中所有 i $\in$ [k]均成立。

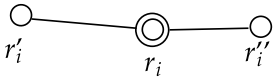


图7：路径构件 $\mathcal{P}1_i$

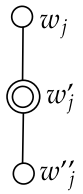


图8：路径构件 $\mathcal{P}2_j$

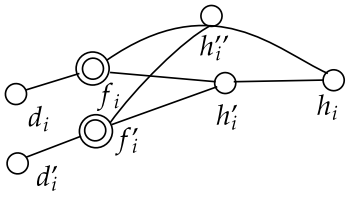


图9：悬挂构件

观察15。 对于 G 的任意 2-子集平方着色 C，所有 $i \in [k], j \in [n], C(r_i) =$ 为蓝色，且 $C(w'_j) =$ 为蓝色；路径构件 $\mathcal{P}1_i$ 和 $\mathcal{P}2_j$ 中的所有其他顶点分别被 r_i 和 w'_j支配。

我们通过为每个 i, i′ ∈ [k]和 j, j′ ∈ [n], 构造连接器构件 S i,i′ j, j′，继续构造图 G，其中 i ＜i′且 j ＜j′。连接器构件 S i,i′ j,j′ 包含七个顶点{$a_1, a2, \cdots, a_6, s^{i}_{,i'}$ j,j′,}，其中 a_5 是一个特殊顶点。我们添加边 (a_1, a_2)、(a_2, a_3)、$(a_2, s^{i}_{,i'}$ j, j′)、(a_3, a_5)、$(a_3, s^{i}_{,i'}$ j,j′)、$(a_4, s^{i}_{,i'}$ j, j′)、(a_4, a_1)、(a_4, a_5) 和 (a_5, a_6)，以完成连接器构件。见图10。此外，通过添加一条连接 a_1 与 B 、 a_6 与 R 的边，将 S i,i′ j, j′ 连接到顶点 R 和 B。该连接器构件通过 a_1 和 a_2 与网格顶点相连。将 S i,i′ j, j′ 中的顶点 a_1 和 a_2 分别与 $p_{i'}$,$d_{'}$ 和 $p_{i,d}$ 相连，其中 d, d′ ∈ {j, j′}。

此外，对于所有 i ∈ [k], ，我们创建一个含七个顶点的悬挂构件 $\mathcal{H}_i$，其顶点为 {h_i、h'_i、h''_i、f_i、f ′i、d_i、d'_i}。两组顶点 {h_i、h'_i、h''_i、f_i} 和 {h_i、h'_i、h''_i、f ′i } 分别构成两个含四个顶点的圈。顶点 f_i 和 f ′i 是 $\mathcal{H}_i$ 中的特殊顶点。分别将 d_i 和 d'_i 与 f_i 和 f'_i 相连。见图9。此外，对于每个 $h_i \in \mathcal{H}_i$、 1 ≤ i ≤ k，将其与网格顶点 $p_{i,j}$相连，其中 j ∈ [n]。

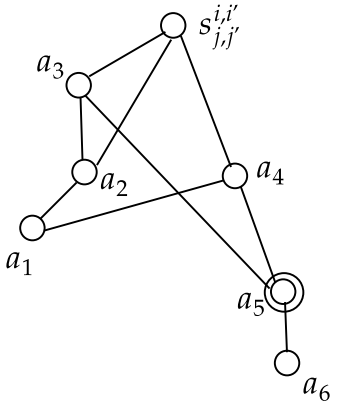


图10：连接器构件

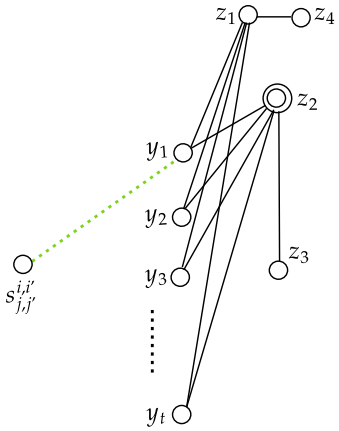


图11：选择器构件

此外，我们添加一个选择器构件 $\mathcal{Y}$，它包含一组顶点$Y = \{y_l | 1 \le l \le t\}$ 以及一组四个顶点 z_1、⋯、z_4，使得z_2 是一个特殊顶点。对于所有$l \in [t]$, ，我们将y_l与z_1 和z_2相连。此外，在z_2 和z_3之间添加一条边。见图11。顶点z_4 与R和B相连。直观地说，这里的选择器构件 $\mathcal{Y}$ 表明哪些实例拥有大小为k的团。

除了这些边之外，对于每个$l \in [t], i$、$i' \in [k]$, 和j、$j' \in [n]$, ，当且仅当$(v_j, v_{j'}) \in E(X_l)$时，添加一条边$(s^{i}_{,i'\,j,j'}, y_l)$。

现在，我们已经完成了图G的构造。以下引理展示了不同构件中顶点的性质，也有助于证明该归约的正确性。

引理 16. 对于 G 的任意2-子集平方着色 C，所有 $i \in [k]$ 的取值下，$C(h_i)$、$C(h'_i)$ 和 $C(h''_i)$ 均为 c_0 。

证明。对于所有$i \in [k]$, ，顶点f_i 和f'_i 都是特殊顶点，且彼此距离为二。因此f_i 和f'_i获得不同的颜色。由于顶点h_i、h'_i和h''_i到f_i 和f'_i的距离至多为 2，我们不能将它们染成红色或蓝色。□

由引理 16 可知，我们无法通过给$N[h_i] \cap \mathcal{H}_i$ 中的某个顶点着色来支配顶点h_i，而 $\mathcal{H}_i$ 中的所有其他顶点都至少被一个特殊顶点f_i'、f_i支配。

观察 17. 对于所有 $i \in [k]$, , 网格顶点 $\{p_{i,j} | 1 \leq j \leq n\}$ 均被顶点 $r_i \in \mathcal{P}1_i$支配。

要支配顶点 $h_i \in \mathcal{H}_i$，我们需要给网格顶点 $\{p_{i,j} | 1 \leq j \leq n\}$中的一个顶点着色。由于 r_i 被着为蓝色，我们可以从集合 $\{p_{i,j} | 1 \leq j \leq n\}$中恰好选一个顶点着为红色。

引理 18. 对于图 G 的任意 2-子集平方着色 C，存在不同的 $j_1, \cdots, j_k$，使得 $C(p_{i,j_i}) =$为红色；并且对于所有其他网格顶点 $\{p_{i,j_t} | 1 \leq t \leq n$ 且 $t \neq i\}$，$C(p_{i,j_t}) =$为 c_0。

证明。我们先证明，对于每个 $i \in [k]$, ，恰好存在一个 $j \in [n]$ ，使得 $C(p_{i,j}) =$为红色。考虑悬挂构件，不难看出，除 h_i 之外的每个顶点都被来自 $\mathcal{H}_i$ 的特殊顶点支配；根据引理 16，$C(h_i) =$为 c_0 。由此可知，$\{p_{i,j} | 1 \leq j \leq n\}$中确实至少包含一个着色顶点。 $\{p_{i,j} | 1 \leq j \leq n\}$中的顶点为红色，因为根据观察 17，$C(r_i) =$为蓝色。设 p_{i,j_i}为 $\{p_{i,j} | 1 \leq j \leq n\}$中那个已着色的顶点。

现在，我们需要证明所有 j_i 都互不相同。可以通过证明对每个 $j \in [n]$, ，恰有一个 $i \in [k]$ 满足 $C(p_{i,j}) =$ 红色来完成。考虑 $\mathcal{P}2_j$ 中的顶点 w_j，注意到 $N(w_j)$ 在 $\mathcal{P}2_j$ 中包含一个蓝色顶点，因此 w_j 有且仅有一个红色邻点，于是恰有一个 $i \in [k]$ 满足 $C(p_{i,j}) =$ 红色。观察 17 还表明，当 $t \neq i$ 时，$C(p_{i,j_t}) = c_0$, 。□

现在，我们可以考虑为每一对 $p_{i'},d'$ 和 $p_{i,d}$（其中 $d, d' \in \{j, j'\}$）所构造的连接器构件 $S_{i,i',j,j'}$。顶点 a_6 与顶点 R 相连，且 a_5 是一个特殊顶点，$C(a_5) =$蓝色。$S_{i,i',j,j'}$ 中的顶点 a_3, a_4, a_6 被顶点 a_5 支配，而顶点 a_1 被 B 支配。

现在，以下观察说明了如何在连接器构件中支配顶点a_2 和$s^i_{,i',j,j'}$。

观察*19*。如果存在一种子集平方着色 C，使得 $C(p_{i,d}) =$红色，且 $C(p_{i',d'}) =$红色，二者均不成立，其中 $d, d' \in \{j, j'\}$，那么以下情况之一成立：*(a)* $C(a_2) =$红色，或 *(b)* $C(a_3) =$红色，或 *(c)* $C(s^i_{,i',j,j'}) =$红色。

证明。如果顶点 $p_{i,d}$ 和 $p_{i',d'}$ 的颜色为 c_0, ，那么要支配 $S_{i,i',j,j'}$中的顶点 $s^i_{,i',j,j'}$ 和 a_2 ，我们可以给顶点 a_3、a_2或 $s^i_{,i',j,j'}$着色。由于 $C(a_5)$ 是蓝色，且 $N[a_1]$包含蓝色，我们不能用蓝色给这些顶点中的任何一个着色（在图12：情形1中，我们给 a_3着了色）。

观察20。 如果存在一种子集平方着色 C，使得 $C(p_{i,d}) = c_0, C(p_{i',d'}) =$ 为红色，那么以下情况之一成立：$(a)\ C(a_3) =$为红色，或 $(b)\ C(s^i_{,i',j,j'}) =$ 为红色。

证明。为了支配顶点 $s^i_{,i',j,j'}$和 a_2, ，我们可以给顶点 a_3或 $s^i_{,i',j,j'}$着色。由于 $e(a_3, a_5)$、$e(s^i_{,i',j,j'}, a_4)$ 和 $e(a_4, a_5)$ 均存在，因此不能将 a_3或 $s^i_{,i',j,j'}$着成蓝色。

观察21。 如果存在一种子集平方着色 C，使得 $C(p_{i,d}) =$为红色，且 $C(p_{i',d'}) =$为 c_0, ，那么以下情况之一成立：$(a)\ C(a_4) =$为红色；或者(b) 存在边$(s^i_{,i',j,j'}, y_l)$ （在图*12*：情况*3*中，我们给 a_4 着色），且 $C(y_l) =$为蓝色。

证明。顶点 a_2 被 $p_{i,d}$ 支配。由于 $C(p_{i,d})$= 为红色，我们不能给 a_3 或 $s^i_{,i'}$ j,j′ 着色来支配顶点 $s^i_{,i'}$j, j′。要支配 $s^i_{,i'}$j,j′，我们可以给 a_4 着色，或者添加一条边 ($s^i_{,i'}$j, j′, y_l)。

观察22. 如果存在一种子集平方着色 C，使得在 C 下 $C(p_{i,d})$ = 为红色，且 $C(p_{i',d'})$ = 为红色，那么边 $(s^i_{,i',j,j'}, y_l)$ 存在。

证明. 如果 $C(p_{i,d})$ = 为红色且 $C(p_{i',d'})$ = 为红色，那么顶点 $s^i_{,i'}$ j,j′ 不能着成红色。由于顶点 $s^i_{,i'}$j, j′ 与 a_5,的距离为二，顶点 $s^i_{,i'}$j, j′ 也不能着成蓝色。因此，为满足 2-子集平方着色，边 ($s^i_{,i'}$j,j′, y_l) 存在。见图12：情形4。

现在可以看出，选择器构件中的一个顶点必须着色，才能支配顶点 z_1。

引理 23。对于图 G 的任意一个2-子集平方着色 C，Y 中恰有一个顶点 $y_l \in \{y_i | i \in [t]\}$ 满足 $C(y_l)$ = 蓝色；对于 Y 中所有其他顶点 y（其中 $y_i \neq y_l \in Y$），$C(y_i) = c_0$。

证明。特殊顶点 $z_2 \in \mathcal{Y}$ 与 B 的距离为二。因此 $C(z_2)$ = 红色。除 z_1 和 z_4 外，$\mathcal{Y}$中的每个顶点都被特殊顶点 z_2支配。由于顶点 z_4 与 B 和 R 相连，我们无法使用 z_1 或 z_4支配 z_1 。只有顶点 $\{y_l | l \in [t]\}$ 能支配顶点 z_1, ，但每个 $\{y_l | 1 \leq l \leq t\}$ 都是 z_2的邻点。因此，$\{y_l | l \in [t]\}$中只有一个顶点能支配顶点 z_1, ，且该顶点的颜色为蓝色。因此，对于 Y 中所有其他顶点 y，$C(y) = c_0$。□

现在，我们提出如下主引理。

引理 24. 图 G 是2-子集平方可着色的，当且仅当存在 $l \in [t]$，使得实例 X_l 含有一个大小为 k 的团。

证明。假设存在一个是实例X_l，其中含有一个大小为k的团。令$v_{l_1}, \cdots, v_{l_k} \in V(X_l)$ 为X_l中构成一个团的顶点。现在，对于所有$i \in [k], j \in [n]$, ，我们按如下方式为图G着色。

1. 将顶点 R 和 B 着色，使得 $C(R)$ = 为红色，$C(B)$ = 为蓝色。 2. 令 $C(f_i)$ = 为红色，$C(f'_i)$ = 为蓝色（根据观察 14，且图 G 中存在边 (B, d_i) 和 (R, d'_i)）。 3. 由于 w''_j与 R 相连，$C(w'_j)$ = 为蓝色。 4. $C(a_5)$ =为蓝色（根据观察 14，且 $N(N(a_5))$ 包含红色）。 5. 由于 z_3 与 B 相连，$C(z_2)$ = 为红色。 6. 由于 r'_i 与 R 相连，$C(r_i)$ = 为蓝色。 7. 令 $C(y_l)$ = 为蓝色（ y_l 支配顶点 z_1）。 8. 对所有 $i \in [k]$，令 $C(p_{i,l_i})$ = 为红色（支配顶点 h_i），其他顶点用 c_0 着色（被顶点 r_i支配）。 9. 若 $C(p_{i,l_i})$ =为 c_0，且 $C(p_{i',l_{i'}})$ =为 c_0，其中 $d, d' \in \{j, j'\}$，则 $C(a_3)$ =为红色。 10. 若 $C(p_{i,l_i})$ =为 c_0，且 $C(p_{i',l_{i'}})$ =为红色，其中 $d, d' \in \{j, j'\}$，则 $C(a_3)$ =为红色。 11. 若 $C(p_{i,l_i})$ =为红色，且 $C(p_{i',l_{i'}})$ =为 c_0，其中 $d, d' \in \{j, j'\}$，则 $C(a_4)$ =为红色。

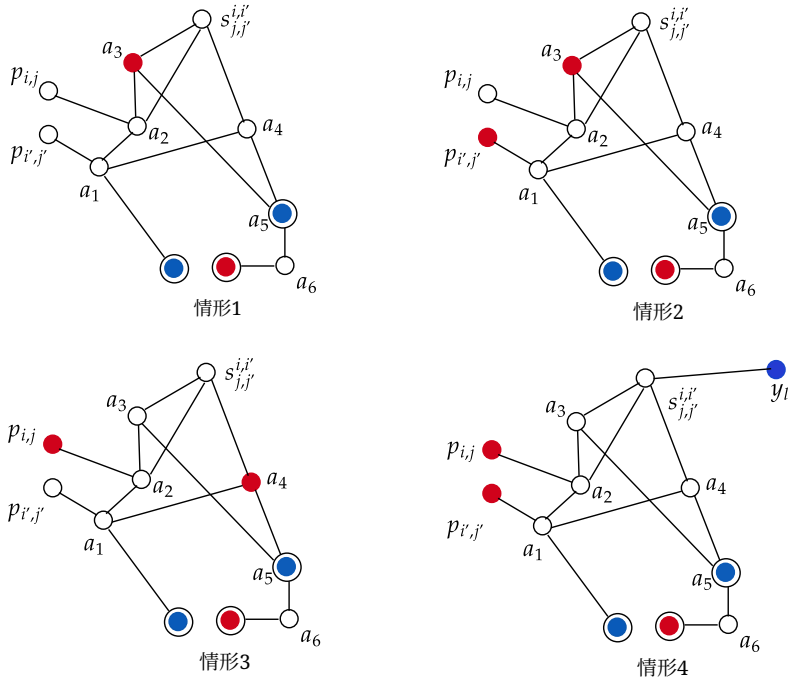


图12：情形分析

容易验证，每个顶点v 和$N[v\{v8>$ in图 G 均满足 2-子集平方着色约束。因此，这种对图 G 的着色是有效的 2-子集平方着色。参见图13。

现在我们可以证明，如果图 G 在C下可以进行2-子集平方着色，那么存在$l \in [t]$，使得实例 X_l 含有一个大小为k的团。

根据引理23，我们知道 Y 中存在一个蓝色顶点。令 y_l 为该顶点。根据引理 18，存在互不相同的 $j1,\cdots、j_k \in [n]$，使得 $C(p_{i,j_i})$ = 为红色。我们将通过证明顶点 $v_{j_1},\cdots、v_{j_k} \in [n]$ 在实例 X_l中构成一个团，来说明实例 X_l 包含一个大小为 k 的团。假设这不成立。那么存在 $i、i' \in [k]$，使得 v_{j_i} 与 $v_{j_{i'}}$ 在 X_l中没有边相连。这会导致矛盾。现在，考虑连接器构件 $S\ i,i'j_i、j_{i'}$，顶点 $p_{i',l_{i'}}$和 p_{i,l_i}被着成红色。由于顶点 $s^{i}_{,i'}j、j_{i'}$与 a_5 和 p_{i,l_i}的距离为二，因此我们不能将顶点 $s^{i}_{,i'}j,j_{i'}$着成红色或蓝色。因此，存在边 $(s^{i}_{,i'}j,j_{i'}, y_l)$，以支配 $s^{i}_{,i'}j、j_{i'}$。□

为证明下界，还需界定图 G 中顶点覆盖的大小。由于 Y 中的顶点在图 G 中构成一个独立集，因此 $V(G)\backslash(Y = \{y_l|1 \le l \le t\})$ 是图 G 的一个顶点覆盖。

图 G 中的网格顶点数量为 nk，且网格中的每对顶点都对应一个连接器构件。因此，连接器构件中的顶点总数为 $7(n^2k^2)$。图 G 中的所有其他顶点数量均由 n 和 k 线性界定。因此 $|V(G)\backslash Y| = O(n^2k^2)$。由此我们证明了以下定理。

定理25。 2-子集平方着色问题以顶点覆盖的大小为参数时，不存在

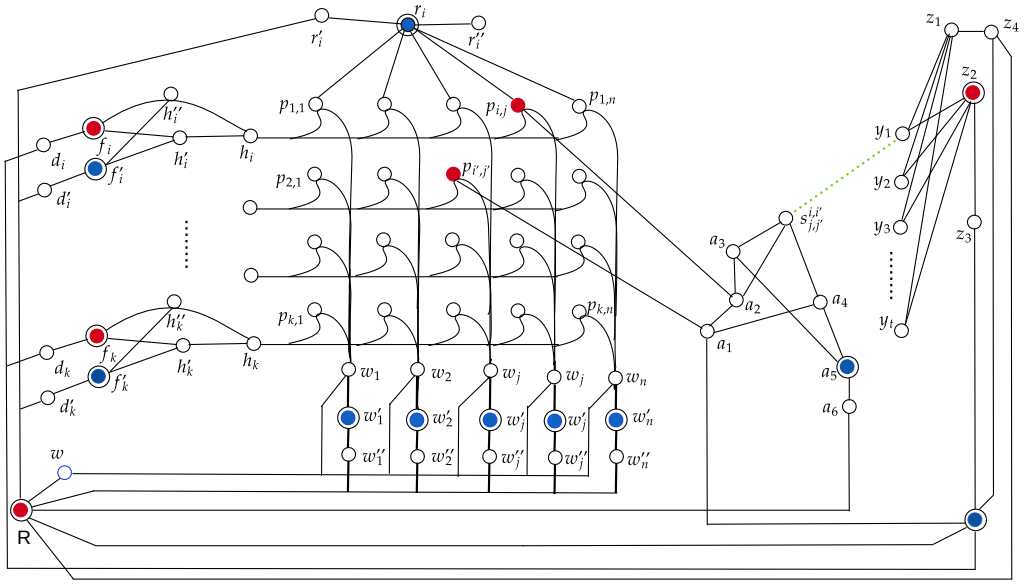


图13：构造图图 G 包含$k = 4$、$n = 5$ 和$t = t_0$。

多项式核。

4.3.2. 核化：q-子集平方着色问题，以顶点覆盖的大小为参数，其中 $q \geq 3$。

本节通过修改图 G 中的构件以及特殊顶点的定义，将结果推广到$q \geq 3$。本节假设$c_1 =$ 为红色，且$c_2 =$ 为蓝色。

特殊顶点：若顶点v 是特殊顶点，则$N(v)$中包含$q + 1$ 个度为 1 的顶点。

观察26。对于G的任意 q-子集平方着色 C，若顶点 v 是特殊顶点，则 $C(v) \neq c_0$。

我们添加一组特殊顶点$Q = \{Q_1, Q_2, \cdots Q_q\}$，使它们在图 G中构成一个团。因此，Q 中的所有顶点都用不同的颜色着色。不失一般性，假设$C(Q_1) = c_1, C(Q_2) = c_2$，依此类推。我们按如下方式修改第 4.3.1 节中使用的构件。

对于每个 i $\in$ [k],，我们添加一个包含三个顶点 r'_i、r_i 和 r''_i 的路径构件 $\mathcal{P}1_i$。在每个 $\mathcal{P}1_i$ 中，都存在边 (r'_i, r_i) 和 (r_i, r''_i)，其中 r_i 是特殊顶点。我们通过在 r_i 与集合 $\{Q_3, Q_4 \cdots Q_q\}$ 中的每个顶点之间添加边，将路径构件 $\mathcal{P}1_i$ 连接到 Q 中的顶点集合。此外，还添加一条边 (r''_i, Q_1)。见图14。此外，$\mathcal{P}1_i$ 通过 r_i 与网格顶点相连，即对所有 j $\in$ [n]，添加边 $(r_i, p_{i,j})$。

对于每个 j $\in$ [n],，我们使用一个路径构件 $\mathcal{P}2$j，其中包含三个顶点 w j、w'_j 和 w''_j。在每个 $\mathcal{P}2_j$ 中，都存在边 (w j, w'_j) 和 (w'_j, w''_j)，其中 w'_j 是一个特殊顶点。我们添加一个顶点 w，并通过在 w_j 与集合 $\{Q_3, Q_4 \cdots Q_q\}$ 中的每个顶点之间添加边，将路径构件 $\mathcal{P}2$j 连接到 Q 中的顶点集合。添加一条边 (w_j, w)，并在 w 与集合 $\{Q_x \in Q | 1 \leq x \leq q$ 和 x $\neq 2\}$ 中的每个顶点之间添加边。见图15。

观察27。 对于图 G 的任意 q-子集平方着色 C，对所有 $i \in [k], j \in [n],$ ，$C(r_i) =$ 为蓝色，且 $C(w'_j) =$ 为蓝色，并且路径构件中的所有其他顶点 $\mathcal{P}1_i$ 和 $\mathcal{P}2_j$ 分别被 r_i 和 w'_j 支配。

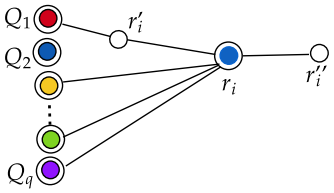


图14：与 Q 中顶点集合相连的路径构件 $\mathcal{P}1_i$

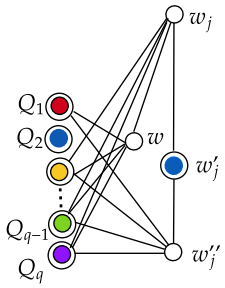


图15：与Q中顶点集合相连的路径构件 $\mathcal{P}2_j$

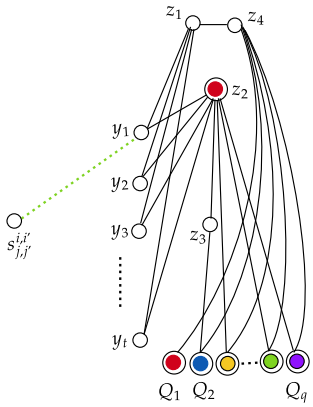


图 16：选择器构件

我们对连接器构件作如下修改。对于每个 i，i′ ∈ [k] 和 j，j′ ∈ [n]，在 S i,i′ j,j′ 中，其中 i <i′ 且 j <j′，构件 S i,i′ j,j′ 包含顶点集合 $\{a_1, a_2, a_3, s^{i,i'}_{j,j'}\}$, $\{a_4^1, a24, \cdots a_4^{q-1}\}$,$\{a_5^1, a2\ 5, \cdots a_5^{q-1}\}$，以及 $\{a_6^1, a2\ 6, \cdots a_6^{q-1}\}$。注意，在 S i,i′ j,j′ 中，所有 a_5^x 都是特殊顶点，其中 $1 \le x \le q-1$。

我们在连接器构件中添加边 (a_1, a_2)、(a_2, a_3)、$(a_2, s^{i,i'}_{j,j'})$ 和 $(a_3, s^{i,i'}_{j,j'})$。对于每个$1 \le x \le q-1$，添加 a_4^x 与顶点 $s^{i,i'}_{j,j'}$、j' 和 a_1 之间的边。除此之外，对于每个$1 \le x \le q-1$，添加 a_4^x 与 a_5^x 之间的边。对于每个 $1 \le x \le q-1$，添加 a_5^x 与 a_3 之间的边。此外，对于每个 $1 \le x \le q-1$，我们在 a_6^x 与 a_5^x 之间添加边。

将 S i,i′ j,j′ 连接到 Q 中的顶点集：在 a_1 与 Q 中除 Q_1以外的所有顶点之间添加一条边。对于每个 $1 \le x \le q-1$，我们将 a_6^x 与 Q 中除顶点 Q_{x+1}以外的所有顶点相连。参见图17。连接器构件通过 a_1 和 a_2与网格顶点相连。分别将 S i,i′ j,j′ 中的顶点 a_1 和 a_2 与 $p_{i',d'}$ 和 $p_{i,d}$ 相连，其中 d, d′ ∈ {j, j′}。

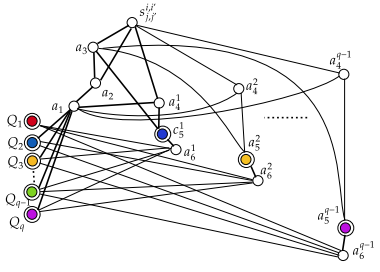


图17：连接器构件与Q中的一组顶点相连

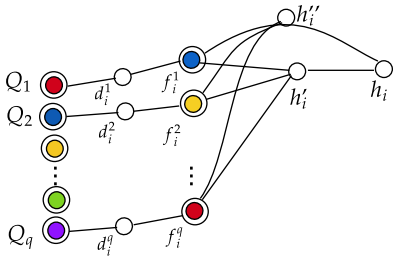


图 18：悬挂构件与Q中的一组顶点相连

此外，对于所有 $i \in [k]$，，我们按如下方式修改悬挂构件。在此证明中，悬挂构件 $\mathcal{H}_i$ 包含顶点集 $\{h_i, h_i', h_i''\}, \{f^1 i, f_i^2 \cdots f^q i\}$ 和 $\{d_i^1, d_i^2 \cdots d^q i\}$，其中所有 $f xi$，$1 \le x \le q$ 都是 $\mathcal{H}_i$中的特殊顶点。对于所有 $i \in [k]$，，这些顶点集 $\{h_i, h_i', h_i'', f xi\}$构成 q 个有四个顶点的圈，其中 $1 \le x \le q$。对于每个 $1 \le x \le q$，d_i^x与 f_i^x之间有一条边。通过在 d_i^x与 $Q_x \in Q$ 之间添加边，将悬挂构件 $\mathcal{H}_i$ 连接到 Q 中的顶点集，对所有 $1 \le x \le q$ 都如此。参见图 18。此外，对于所有 $j \in [n]$，每个 h_i 都与网格顶点 $p_{i,j}$相连。

此外，我们对选择器构件 $\mathcal{Y}$ 作如下修改。选择器构件 $\mathcal{Y}$包含一组顶点 $Y = \{y_l | 1 \le l \le t\}$ 以及四个顶点 $z1, \cdots, z_4$，，其中 z_2 是一个特殊顶点。对于所有 $l \in [t]$，，我们将 y_l 与 z_1 和 z_2 相连。顶点 z_4 与 z_1 相连。此外，在 z_2 和 z_3 之间添加一条边。通过添加一条边 (z_3, Q_2)，并将顶点 z_2 与所有 $Q_x \in Q$ 相连，从而将选择器构件连接到 Q 中的顶点集合，其中 $3 \le x \le q$。除这些边外，还将 z_4 与集合 Q 中的所有顶点相连。见图 16。

引理 28。 对于图 G 的任意 q-子集平方着色 C，$C(h_i)$，$C(h_i')$ 和 $C(h_i'')$ 对于 $i \in [k]$ 的所有取值均为 c_0。

证明。对于所有$i \in [k]$，，顶点f_i^x，其中 $1 \le x \le q$ 是特殊顶点，且它们之间的距离为二。因此，所有f_{xi}都会获得不同的q颜色。由于顶点h_i、h_i'和h_i''与f_i^x的距离至多为 2，因此我们不能给它们着色。□

由引理 28 可知，我们无法通过给$N[h_i] \cap \mathcal{H}_i$中的顶点着色来支配顶点$h_i$，而 $\mathcal{H}_i$中的所有其他顶点都至少被一个特殊顶点f_{xi}支配。不失一般性，假设$C(f_{xi}) = c_{i+1}$，其中 $1 \le x \le q - 1$，且 $C(f_{xq}) = c_1$。

观察 29。 对于所有 $i \in [k]$，，网格顶点 $\{p_{i,j} | 1 \le j \le n\}$ 均被顶点 $r_i \in \mathcal{P}1_i$支配。

为了支配顶点 $h_i \in \mathcal{H}_i$，我们需要给网格顶点 $\{p_{i,j} | 1 \le j \le n\}$中的一个顶点着色。由于 r_i 使用蓝色着色，且r_i与 Q 中的所有顶点相连，除了Q_1，，因此我们可以从集合 $\{p_{i,j} | 1 \le j \le n\}$ 中恰好选一个顶点着红色。

引理 30. 对于图 G 的任意 q-子集平方着色 C，存在互不相同的 $j_1, \cdots j_k$，使得 $C(p_{i,j_i}) =$为红色；并且对于其他所有网格顶点 $\{p_{i,j_t} | 1 \le t \le n$ 和 $t \ne i\}$，$C(p_{i,j_t}) =$为 c_0。

证明。首先证明，对于每个 $i \in [k]$，，恰好存在一个 $j \in [n]$，使得 $C(p_{i,j}) =$ 为红色。考虑悬挂构件，很容易看出，除了 h_i 之外，$\mathcal{H}_i$ 中的每个顶点都被 $\mathcal{H}_i$ 中的特殊顶点支配；根据引理 28，$C(h_i) =$ 为 c_0。由此可知，$\{p_i, j | 1 \le j \le n\}$确实至少包含一个着色顶点。$\{p_{i,j} | 1 \le j \le n\}$中的顶点颜色为红色，因为 $C(r_i) =$ 为蓝色，并且根据观察 29，存在边 (r_i, Q_x)，其中 $3 \le x \le q$。令 p_i, j_i为 $\{p_{i,j} | 1 \le j \le n\}$中的那个着色顶点。

现在，我们需要证明所有 j_i 都互不相同。可以通过证明对于每个 $j \in [n]$，，至多存在一个 $i \in [k]$，使得 $C(p_{i,j_i}) =$ 为红色来说明这一点。考虑 $\mathcal{P}2j$ 中的顶点 $w j$，注意到 $N(w j)$ 包含除红色外所有 q 个着色顶点，因此 w_j 有唯一一个红色邻点，故恰好存在一个 $i \in [k]$，使得 $C(p_{i,j}) =$ 为红色。这也说明 $C(p_{i,j_t}) = c_0$，，其中 $t \ne i$。

现在我们可以考虑连接器构件 S i,i′ j, j

{ ′} 1 ≤ ≤ $^{x}_{6}$ ′ $^{i'}$ ′ $^{i\,d}_{x+1}$ $^{x}_{5}$ ′ ∈，它对应于 p ,d 和 p ，其中 d、d j、j。对于每个 x q，a 与 Q 中除 Q 外的所有顶点相连，并且所有 a 都是特殊顶点，$C(a_5^x) = c_x$。S i,i′ j,j′ 中的顶点 a_3, a_4^x、a_6^x 由顶点 a_5^x 支配，而顶点 a_1 由 B 支配。

现在可以看出，连接器构件、选择器构件和网格顶点可用的颜色是红色和蓝色。因此，可以用第 4.3.1 节中的类似论证证明以下引理。

现在可以看出，选择器构件中的Y 中必须有一个顶点着色，才能支配顶点z_1。

引理31。 对于图 G 的任意 q-子集平方着色 C，Y 中恰好存在一个顶点 $y_l \in \{y_i | i \in [t]\}$，使得 $C(y_l) =$ 为蓝色；对于其他每个顶点 $y_i \in Y$，其中 $y_i \neq y_l \in Y$，$C(y_i) =$ 为 c_0。

证明。特殊顶点 $z_2 \in \mathcal{Y}$ 与 Q 中除 Q_1 之外的所有顶点的距离均为二。因此，$C(z_2) =$ 为红色。$\mathcal{Y}$ 中除 z_1 和 z_4 之外的每个顶点都被特殊顶点 z_2支配。由于顶点 z_4 与 Q 中的所有顶点相连，我们无法使用 z_1 或 z_4支配 z_1。只有顶点$\{y_l | l \in [t]\}$ 能支配顶点 z_1, ，但每个 $\{y_l | 1 \leq l \leq t\}$ 都是 z_2的邻点。因此，只有一个来自 $\{y_l | l \in [t]\}$ 的顶点能支配顶点 z_1, ，且该顶点的颜色为蓝色。因此，对于其他每个顶点 $y \in Y$，$C(y) =$ 为 c_0。□

通过与引理 24 的证明中类似的论证，我们可以得出以下引理。

引理 32。 图 G 可以进行 q-子集平方着色，当且仅当存在 $l \in [t]$，使得实例 X_l 含有一个大小为 k 的团。

现在，存在一个 $|V(\text{图 } G) \backslash Y| = O(qn^2k^2)$。因此，我们证明了以下定理。

定理 33。 以顶点覆盖大小为参数的 q-子集平方着色问题无法拥有多项式核，针对 $q \geq 3$，可利用团问题进行 OR-交叉组合。

4.4. 以邻域多样性为参数

我们首先定义图的邻域多样性。

定义 5. [22] 给定一个图 $G = (V, E)$，两个顶点 $u, v \in V$ 具有相同的类型，当且仅当 $N(v) \backslash \{u\} = N(u) \backslash \{v\}$。若存在一个将 V 划分为至多 t 个集合的划分 $V_1, V_2, \ldots, V_t$，使得对于 $i = 1, 2, \ldots, t$，V_i 中的所有顶点具有相同类型，则图 G 的邻域多样性为 t。集合族 $\nu = \{V_1, V_2, \ldots, V_t\}$ 称为 G 的类型划分。

创建$V(G)$的此类类型划分后，我们注意到，同一划分中的顶点要么构成一个团，要么构成一个独立集。此外，对于 $1 \leq i, j \leq t$，划分 V_i 中的每个顶点要么与另一个划分V_j 中的每个顶点都相邻，要么V_i 与V_j中的顶点之间没有边。

引理 34. 对图 G 进行子集平方着色所需的颜色数至多为其邻域多样性，即 $\chi_{ssc}(G) \leq t$。

证明。对于给定的图G，考虑其邻域多样性为t的类型划分 ν。若有孤立顶点，则用t 种颜色中的一种为所有孤立顶点着色。对于 $1 \le i \le t$，在每个V_i中选取一个顶点，并分别用不同的颜色着色。这样便得到$V(G)$的子集平方着色。理由如下：构成团的划分V_i 中的顶点都被支配，因为V_i中存在一个着色邻点。此外，如果V_i 中的顶点与划分V_j 相邻，其中 $1 \le j \le t$，那么它们还可能与更多着色邻点相邻。现在考虑划分V_i 中的顶点构成独立集的情形；此时，它们会与至少另一个划分V_j 中的所有顶点相邻，从而保证V_i中每个顶点的闭邻域内至少有一个着色邻点。此外，在任何连通分量中，着色时都没有重复使用颜色；我们使用了t 种颜色对图G进行子集平方着色。因此结论成立。

现在，我们通过给出一个多项式核，证明以t 为参数的q-子集平方着色属于固定参数可处理。

定理35。 以邻域多样性为参数的 q-子集 平方 着色 问题存在一个规模为 $O(t^2)$的多项式核。

证明。设图 G 是一个连通图，并具有大小为t、$t > 1$的类型划分。设图 $G'(V',E')$是从图 $G(V,E)$中每个类型划分保留$q + 1$ 个顶点、删除其余顶点后得到的图。我们将证明，图 G' 可进行 q-子集平方着色，当且仅当图 G 可进行 q-子集平方着色。注意，图 G' 也是一个连通图。

设 χ 是图 G′的一个 q-子集平方着色。我们断言，χ 也是图 G 的一个 q-子集平方着色。设 $V'_i = \{v_1, v2, \cdots, v_{q+1}\}$是图 G′的顶点集 V(G′) 中类型划分的一个顶点集。如果 v_i 和 v_j,$1 \le i, j \le q + 1, i \ne j$, 均已着色，那么 $\chi(v_i) \ne \chi(v_j)$。否则，V'_i中的所有顶点至少有一个公共邻点，因为图 G′ 是连通图，这就产生矛盾。由于颜色至多有 q 种，因此至少存在一个未着色顶点，记为 v_i，它属于 V'_i。现在，$V_i \setminus V'_i$中的每个顶点在图 G 中都被与图 G′中支配 v_i 的相同顶点所支配。

反过来，假设图 G 存在一个q-子集平方着色，χ。那么，我们使用V'_i中 χ 所用的颜色（如果有的话），任意为顶点v_1, v_2、$\cdots$、v_{q+1} 着色。现在很容易看出，这对图 G' 而言也是一个有效的q-子集平方着色。

现在，$|V(\text{图 } G')| \le (q + 1)t$。如果$q \ge t$，根据引理 34，该问题显然是一个是实例。因此 $|V(\text{图 } G')| = O(t^2)$。现在结论成立。

4.5. 按到簇图的距离参数化

定义 6。 簇图是若干完全图的不交并。

容易看出，簇图可以进行 1-子集平方着色。

定义 7。 图图 G 的簇图顶点删除集X 是一个顶点集合，使得图 $G\setminus X$ 构成一个簇图。

簇图顶点删除数定义为簇图顶点删除集 X 可能的最小规模。直观而言，它衡量一个图距离成为簇图有多远。

在本节中，令 X 为$V(\text{图 } G)$ 的子集，使得图 $G[V \setminus X]$ 是一个簇图，且 $|X| = k$.
现在我们观察 $\chi_{ssc}(\text{图 } G)$ 与 k 之间的联系。

引理 36。 $\chi_{ssc}(\text{图 } G) \le |X| + 1$。

证明。为了对子集平方着色图图 G进行着色，我们使用k 种不同的颜色为每个顶点 $v \in X$ 着色。现在，对于图 $G\backslash X$中的每个完全子图，若存在未被支配的顶点，则恰好为其中一个顶点着色，所用颜色须与先前使用的k 种颜色不同。注意，位于两个不同完全子图中的两个着色顶点u 和v 在X中没有公共邻点，因为在这种情况下，我们不会选择u 或v 中的任何一个进行着色。我们仅将 $(k+1)^{th}$ 种颜色中的一种分配给X中不与任何着色顶点相邻的顶点。由此得到图G 的子集平方着色，因为X 中的顶点均被分配给它们的颜色所支配。此外，图 $G\backslash X$ 中的一些顶点也可能在X中有邻点。每个$v \in V$(图 G)在$N[v]$ 中至少有一个着色邻点；若$N[v]$, 中有多个着色顶点，则由于我们使用了$k+1$ 种颜色，可以确定这些顶点的颜色彼此不同。因此结论得证。

定理 37. 当参数为到簇图的距离时，q-S子集 平方 着色 是固定参数可处理的。

证明. 如果$q \geq |X|+1$，那么根据引理 36，显然存在子集平方着色。因此，我们假设$q \leq k$。

如果图$G[V \setminus X]$ 中某个团内有两个或更多顶点在X中的邻域相同，则删除除其中一个以外的所有这些顶点。由于这些顶点的闭邻域相同，这不会影响解。因此，图$G[V \setminus X]$ 中的每个团至多有 2^k 个顶点。

现在我们来界定图 G[V \ X]中团的数量。令 $\{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_{2^k}\}$ 为 X 的子集族。对于图 G[V \ X], 中的两个团 C_a 和 C_b，如果对于所有 $1 \leq i \leq 2^k$，C_a 和 C_b 要么都各自有一个顶点，其在 X 中的邻域为 $\mathcal{X}_i$，要么都没有这样的顶点，则称它们具有相同的类型。注意，团的不同类型至多有 2^{2^k} 种。现在我们穷尽地应用以下归约，得到约简图 G′。如果同一类型的团超过 $q2^k+1$ 个，则删除除 $q2^k+1$ 个以外的所有团。因此，G′中至多有 $k+2^{2^k}\cdot(q2^k+1)$ 个顶点。

我们声称，图 G′ 有 q-子集平方着色，当且仅当 G′ 有 q-子集平方着色。假设 G′ 存在一种 q-子集平方着色。令 C_i 为 G\G′中的一个团。因此，G′有 $(q2^k+1)$ 个与 C_i类型相同的团，记为 C'_1、C'_2、. . .、C'_{q2^k+1}。类似地，对于 $v \in V(C_i)$，对所有 $1 \leq j \leq q2^k+1$，存在 $v_j \in V(C'_j)$，使得 $N(v) \cap X = N(v_j) \cap X$。我们将证明，至少存在一个 C'_j 满足其中一个条件。

- C'_j包含一个顶点 v_j，使得 $N(v_j) \cap X = \emptyset$，且 v_j已着色。
- 集合C'_j 中的顶点均未着色。

如果第一个条件成立，那么我们可以通过将对应顶点v 着成与v_j相同的颜色，来支配C_i 中的所有顶点。现在，假设第一个条件不成立。对于所有 $1 \leq j \leq q2^k+1$，考虑顶点$v_j \in V(C'_j)$。由于它们都在X中有公共邻点，因此这类顶点至多可以着色q 个。团中的顶点至多有 2^k 个，因此至多有$q2^k$ 个团含有已着色顶点。团中所有未着色的顶点都由X中的顶点支配。同一组顶点可以支配C'_j 在G中的顶点。反方向显而易见。

由于约简实例的大小由k的函数所界定，因此以距簇图的距离为参数时，q-子集平方着色属于固定参数可处理问题。

4.6. 以孪生覆盖的大小为参数

定义 8。 [20] 对于图图 $G(V,E)$，顶点的一个子集X 是一个孪生覆盖，如果对于每条边$e = uv \in E$(图 G)，要么满足（a）$u \in X$ 或$v \in X$，要么满足（b）u 和v 是真孪生顶点。

如果两个顶点 u 和 v 之间有一条边，且每个其他顶点要么同时与 u 和 v 相邻，要么与它们都不相邻，那么 u 和 v 就是真孪生顶点。根据定义，如果 X 是孪生覆盖，那么$G[V \setminus X]$ 是团的不相交并；并且对于每个顶点 $v \in X$ 以及$G[V \setminus X]$, 中的每个团，v 要么与该团中的每个顶点相邻，要么v 与该团中的任何顶点都不相邻。因此，对于每个图G，图G 到簇图的距离 ≤ 不超过孪生覆盖的大小≤ 不超过顶点覆盖的大小。因此，由定理 37 可知，以孪生覆盖为参数的q-子集平方着色问题是固定参数可处理的。这里，我们给出一个运行时间更优的算法。

定理 38. 以孪生覆盖的大小为参数的 q-子集 正方形 着色 问题是固定参数可处理的。

证明。设$X \subset V$ 是图$G(V,E)$的孪生覆盖，且 $|X| = k$。可以看出，χ_{ssc}(图) ≤$|X|$。若$q \geq k$，则返回 *TRUE*。现在，我们假设$q < k$。

考虑所有可能的 $(q+1)^k$ 着色，其中用 $\{c_0, c_1, \ldots\ldots c_q\}$ 对 X 着色。设 χ 是这样一种对 X 的着色，使得 V 中没有任何顶点在其闭邻域内出现重复颜色。若对 V *extends* χ 的一种着色 χ_G 满足：将 χ_G 限制在 X 上后得到 χ，我们就称该着色可扩展。我们的算法尝试按如下方式将每一种着色扩展为 G 的有效 q-子集平方着色。

在$G[V \setminus X]$, 的每个团中，我们可以保留一个顶点并删除其余顶点，因为团中的所有顶点都与 X 中相同的一组顶点相邻。现在可以观察到，$G[V \setminus X]$ 中的每个顶点都是独立的。

设 $I = V \setminus X$。令 $W \subseteq I$ 为未被 χ 支配的顶点集合。注意，在将 χ 扩展得到G 的有效子集平方着色中，这些顶点均已着色，因为 I 在G 中诱导出一个独立集。注意，任意$v \in X$ 在 W 中至多有 q 个这样的邻点，因为它们中的每一个都需要使用不同的颜色着色。

W 中的顶点数因此有上界 kq，而这至多为 k^2，因为 $q \leq k$。现在我们可以通过暴力搜索，在 $O(k^{k^2})$ 时间内为 W 中的顶点着色。接下来，X 中仍有一些未被支配的顶点。我们将这些顶点在 I 中的邻点集合称为 U。如果 U 中有两个或更多顶点在 X 中具有相同的邻域，那么我们只保留 U 中的一个这样的顶点，并删除其余顶点。这不会影响解，因为被删除的顶点已经被支配，而且任一被删除的顶点都能支配与保留顶点相同的顶点集合。因此，U 至多有 2^k 个顶点，我们可以在 $O(q^{2^k})$ 或 $O(k^{2^k})$ 时间内为它们着色，因为 $q \leq k$。因此，该算法的运行时间为 $O(k^{2^k})$。□

4.7. 按模宽参数化

我们首先说明 χ_{ssc}(图 G) 与图的模宽之间的联系。我们先定义图的模宽。图图 G 的模宽通过四种运算计算，即创建孤立顶点、不交并、完全连接和替换。更确切地说，图 G 的模宽等于任意一次替换运算所使用的操作数的最大数量。图图 G 的模分解涉及的这四种运算见 [19]。为完整起见，我们在此列出这四种运算。

定义 9。 [19] 计算图图 G模宽所涉及的代数运算。

- 创建一个孤立顶点；
- 两个图的不交并，即图不交并G_1 和G_2的不交并，记作$G_1 \otimes G_2$，是一个顶点集为$V(G_1) \cup V(G_2)$、边集为$E(G_1) \cup E(G_2)$的图；
- 两个图的完全连接，即图完全连接G_1 和G_2的完全连接，记作$G_1 \oplus G_2$，是一个顶点集为$V(G_1) \cup V(G_2)$、边集为$E(G_1) \cup E(G_2) \cup \{\{v, u\} : v \in V(G_1)$ 且$u \in V(G_2)\}$。
- 相对于某个图 G′（其顶点为 v1, . . . , vn）的替换运算，即对于图 G1, . . . , Gn，将图 G′ 的顶点分别替换为图 G1, . . . , Gn，记作 G′(G1, . . . , Gn)，所得图的顶点集为 $\bigcup_{1 \le i \le n} V(Gi)$，边集为 $\bigcup_{1 \le i \le n} E(Gi) \cup \{\{u, v\} : u \in V(Gi)$ 且 $v \in V(G\,j)$，$vi, v\,j \in E(G')$ 且 $i \neq j\}$。因此，G′(G1, . . . , Gn) 可由 G′ 得到：将每个顶点 $vi \in V(G')$ 替换为图 Gi，并且每当 $\{vi, v\,j\} \in E(G')$ 时，在图 Gi 的顶点与图 G j 的顶点之间添加所有边。

定义 10。 [19] 设A 为仅使用定义 9 中提到的四种运算的代数表达式。我们将A 的宽度定义为A中任一替换运算所使用的操作数的最大数量。图 G 的模宽记作$w(G)$，是使得G能够由宽度至多为m的此类代数表达式构造得到的最小整数m 。

余图是可通过以下运算构造的图：–创建一个孤立顶点、两个图的不交并，以及两个图的完全连接。根据定义，余图的模宽为二 [14]。

现在，我们通过界定由定义 9 中某种运算生成的每个子图所使用的颜色数，陈述关于最优算法及其界的结果。

引理39。 对图 G 进行子集平方着色所需的最大颜色数等于 G 的模宽，即 $\chi_{ssc}(G) \le w(G)$。

证明。现在我们讨论在执行这些运算时，对图 G 进行子集平方着色所需的最大颜色数。引入一个孤立顶点时，我们用一种颜色为其着色。对于两个图的不交并运算所产生的顶点，我们可以使用为不交并中的每个子图进行子集平方着色时所用颜色数的最大值。现在考虑对子图 G_c进行完全连接运算。设 $G_c = G_{c1} \otimes G_{c2}$。如果顶点 $v \in V(G_{ci})$，$i = \{1, 2\}$ 的度为 $|G_i(V)| - 1$，那么 $\chi_{ssc}(G_c) = 1$。否则，我们可以分别从每个 G_{c1} 和 G_{c2} 中任取一个顶点 v，并用两种不同的颜色为它们着色。因此，χ_{ssc} 的值为 ≤ 2。最后，我们考察替换运算。我们分别从每个 G_i 中任取一个顶点 v，并用不同的颜色为它们着色。现在，对于 G_i 中的每个顶点，它都与 G_j中的另一个已着色顶点相邻，其中 $j \neq i$。因此，χ_{ssc} 的值至多为 w。

定理 40。 q-子集 方着色 是固定参数可处理 ， 当按模宽参数化时。

证明。设图 G 是一个模宽为w的图。我们知道，图 G 可以通过定义 9 中定义的运算以层次化方式构建。我们的算法会为每个运算所生成的中间子图计算最优子集平方着色，从而最终

计算图 G的最优子集平方着色。此外，对于每个子图，算法都会维护该子图的最小支配集大小。

支配一个子图所需的最优颜色数为 1，当且仅当该子图仅包含孤立顶点。该子图的最小支配集包含所有顶点。现在，考虑运算 $G_d = G_{d1} \otimes G_{d2}$。$G_d$ 上所需的最优颜色数为 max{用于(G_{d1})的最优颜色数，用于(G_{d2})的最优颜色数}。G_d 的最小支配集是 G_{d1} 和 G_{d2} 的最小支配集的并集。现在，考虑通过完全连接运算 $G_c = G_{c1} \oplus G_{c2}$ 创建的子图 G_c。如果 $V(G_{ci})$ 中的一个顶点 v（$i = \{1, 2\}$）的度为 $|G_i(V)| - 1$，那么该节点上使用的最优颜色数为 1。否则，我们从每个图中任意选择一个顶点着色。因此，该节点上使用的最优颜色数为 1 或 2。这里，最小支配集的大小与所用的最优颜色数相同。

现在我们考虑图 G′上的替换运算。在此运算中，对于 $1 \le i \le w$，我们将每个顶点 $v_i \in V(G')$ 替换为子图 G_i。因此，G′ 中的顶点与子图 G_i一一对应。考虑图 G 的最优子集平方着色 χ。我们按如下方式猜测与 χ 对应的 G′ 的着色 χ'：若 G_i 中存在在 χ下着色的顶点，则顶点 $v_i \in V(G')$ 着色为 χ'。此外，$\chi'(v_i)$ 是 χ 在 G_i中使用的颜色之一。若 G_i 中的顶点均未在 χ下着色，则 v_i 不着色为 χ'。

我们可以在至多 $(q + 1)^w$ 时间内正确猜测着色 χ'。我们按如下方式将着色χ'扩展到 χ：若顶点 v_i 在 G′中已着色，则在 G_i中任意选择一个顶点 v 着色；否则，G_i 中的顶点均未着色。此时，图 G 中的部分顶点已经被支配。对于子图 G j，若存在边 $(v_i, v_j) \in E(G)$，且 $\chi'(v_i) \neq c_0,$，则由于 G_i中存在着色顶点，G j 中的每个顶点都被支配。此外，若 G_i 中存在着色顶点，则与该着色顶点相邻的 G_i 中的所有顶点都被支配。因此，若 v_i 的邻点（在图 G′中）在 χ'下均未着色，或者等价地，G 中对应的子图没有着色顶点，且 v 与 G_i中的着色顶点不相邻，则顶点 $v \in V(G_i)$ 尚未被支配。在 χ中，通过给 G_i中的更多顶点着色，可以支配 G_i 中的这类顶点。注意，在 G′中，G_i 中的顶点不能使用已在子图 G j 中使用的颜色着色，其中 $v_j \in N(N(v_i))$。这为我们提供了一个可用于 G_i的可用颜色子集。由于 G_i 中的任意一对顶点都有一个公共邻点，因此同一个 G_i中不能重复使用颜色。由此可知，这样一个子图 G_i 中的着色顶点集合对应于 G_i的最小支配集。因此，算法只需检查 $G_i \le$ 的最小支配集的大小是否等于 G_i 的可用颜色数。容易看出，类似地，可以从 G′的最小支配集出发计算图 G 的最小支配集；而后者又可以在 $O(2^w)$ 时间内猜测得到。

由此可知，按模宽参数化时，q-子集平方着色属于固定参数可处理问题。

5. 界χ_{ssc}

本节讨论对某些图类进行子集平方着色所需的最少颜色数的界。

5.1. 树

我们证明，当图 G是一棵树时，$\chi_{ssc}(G)$可由最大度界定。我们还证明，$\chi_{ssc}(G)$存在一个关于n的下界。

引理41。设 Δ 为树 T 的最大度。那么，Δ 种颜色足以对 T 进行子集平方着色。此外，有时需要 Δ − 1 种颜色。

证明。考虑以顶点 r为根的树 T。用颜色 c_i 给根 r 着色，其中 $i \neq 0$。其下一层的节点保持未着色。再下一层的节点（即r的孙节点）用颜色 c_j 着色，其中 $j \notin \{0, i\}$，且与r 的每个子节点相邻的节点都使用不同的颜色。之所以能完成这样的赋值，是因为在用 c_i给 r 着色后，我们还有 $\Delta - 1$ 种颜色。这样做可以保持子集平方着色的特性：为T中每个节点的闭邻域内的顶点赋予至多 Δ 种不同的颜色。换言之，我们对树的交替层进行着色，从而实现子集平方着色。因此，要么顶点v 已着色，要么它的所有邻居都已用至多 Δ 种颜色着色。需要指出的是，就所用颜色数而言，这种着色未必是最优的。不过，它给出了对树进行子集平方着色所需颜色数的上界。

为证明下界，考虑一棵高度为 2 的树 T。令 r 为根，d 为树 T 中除叶节点外每个顶点的子节点数。因此，最大度为 $\Delta = d + 1$。令 v为高度为 1 的顶点。注意，若要对子节点进行子集平方着色，且为 v 的每个子节点使用不同颜色，则需要 d 种颜色。否则，我们必须为v 着色，以支配v 未着色的子节点。将这一论证推广到v 的每个 $d - 1$ 个兄弟节点，可知任何对 T 的子集平方着色都需要d 种颜色。由此可得下界。

推论 **42.** 存在一些树，进行子集平方着色时需要 $\Omega(\sqrt{n})$ 种颜色。

现在我们知道，树是二分图的一个子类。因此，该下界也适用于二分图这一类。下面的结果表明，二分图的一个子类可以使用常数种颜色进行子集平方着色。

5.2. 二部置换图

本节讨论二部置换图中 χ_{ssc} 的界。

如果一个图既是二分图又是置换图，则称其为二部置换图。设 图 $G(A \uplus B, E)$ 是一个连通二部置换图，那么它满足下文定义的强序、邻接性质和包围性质 [30]。

1. 二分图 G(A $\uplus$ B, E) 的顶点集 A 的一个序具有邻接性质，当且仅当对于每个顶点 v $\in$ B，N(v) 中的顶点在 A 的序中连续。2. 二分图 G(A $\uplus$ B, E) 的顶点集 A 的一个序具有包围性质，当且仅当对于任意一对顶点 v, u $\in$ B，若 N(v) 是 N(u) 的子集，则 N(u) – N(v) 中的顶点在 A 的序中连续。3. 二分图 G(A $\uplus$ B, E) 的顶点强序由 A 的一个序和 B 的一个序组成，并且对于所有 (a, b′), (a′, b) $\in$ E，其中 a, a′ 属于 A 且 b, b′ 属于 B，若 a $<$ a′ 且 b $<$ b′，则 (a, b) 和 (a′, b′) 都属于 E。

引理 43. 如果图 G 是连通二部置换图，则 χ_{ssc}(图 G) ≤ 4。

设 $A = \{a_1, a2, \cdots a_n\}$ 和 $B = \{b_1, b2, \cdots b_m\}$ 具有强序性质。对于 $a_j \in A$，令 $s(a_j) = \min\{i | b_i \in N(a_j)\}$ 和 $l(a\ j) = \max\{i | b_i \in N(a\ j)\}$ 分别为与 a_j 相邻的最小和最大顶点。（B 中的定义对称。）

现在，我们为来自 A 的一组顶点着色，使 B 中的所有顶点都被支配。第一步，我们为 A 中的第一个顶点 a_1，着上颜色一。

在第k步中，我们考虑使$b_j \in B$ 尚未被支配的最小j。然后，我们对$a_i \in A$ 进行着色，使i 成为满足$N(a_i)$包含b_j的最大整数；若k 为奇数，则使用颜色一，

否则使用颜色二。重复此过程，直到B中的每个顶点都被支配。现在，若a_i、a_j、$a_k \in A$是在连续步骤中着色的，那么$N(a_i)$与$N(a_k)$不相交。为导出矛盾，假设$N(a_i) \cap N(a_k) \neq \emptyset$。令<style id='j'>′ $= l(a_i)$。那么顶点$b_{j'+1}$同时被a_j和a_k支配。这与a_j由算法着色以支配$b_{j'+1}$相矛盾。因此，B中没有顶点的邻域包含重复的颜色。

同样，我们可以通过给B中的顶点赋予颜色三和颜色四，来支配A中的所有顶点。这就证明了结论。□

此外，我们证明，作为二分置换图子类的毛毛虫图是 3-子集平方可着色的。

定义 11. [29]毛毛虫图是这样一棵树：每个顶点到某条中心路径的距离都至多为一。

引理 44. 如果图 G 是毛毛虫图，那么 $\chi_{ssc}(G) = 3$。

证明。设P是图G的中心路径，其顶点为$v_1, v_2, \ldots\ldots v_{n'}$。现在对$P$中的顶点着色，使$v_i$在 $1 \leq i \leq n'$时获得颜色（i 模 3 + 1），即可得到一种有效的 3-子集平方着色。有时，毛毛虫图进行子集平方着色需要 3 种颜色。考虑这样一类毛毛虫图即可看出这一点：中心路径中的每个顶点都与至少三个度为一的顶点相邻。结论得证。

5.3. 阈值图

定义 12. [23] 如果一个图可以从空图开始，通过反复添加孤立顶点或支配顶点构造出来，那么它就是阈值图。

引理 45. 如果图 G 是阈值图，那么 χ_{ssc}(图 G) = 1。

证明。如果存在孤立顶点，就用相同的颜色为它们着色。通过为最后引入的支配顶点 v着色，我们便满足了子集平方着色图 G 的要求，因为图 G 中的每个顶点在其闭邻域中只有一个已着色的顶点v。

我们知道，阈值图属于分裂图与余图的交集。我们将在后续章节研究这些图类。

5.4. 分裂图

定义 13. [6] 若图图 G 的V(图 G)可划分为两个集合A 和B，使得A 诱导一个团，且B诱导一个独立集，则称图 G 为分裂图。

定理 46. 存在具有 n 个顶点的分裂图，需要$\Omega(\sqrt{n})$ 种颜色才能进行子集平方着色。

证明。我们按如下方式构造一个分裂图 $G = (A \uplus B, E)$。其中 A 诱导一个团，B 诱导一个独立集。令 $A = \{v_1, v2, \cdots v_n\}$，$B = \{v_{i,j} | 1 \leq i, j \leq n\}$。此外，对于所有 $1 \leq i, j \leq n$，我们添加从 $v_{i,j} \in B$ 到 $v_i \in A$ 以及 $v j \in A$ 的边。注意，G 有 $n^2 + n$ 个顶点。

所有顶点 $v_{i,i} \in B$ 的度均为 1。要支配 $v_{i,i} \in B$，我们必须对 v_i 或 $v_{i,i}$ 着色，其中 $1 \leq i \leq n$。若对所有 i 都将 v_i 着色，即 $1 \leq i \leq n$，则需要 n 种颜色，因为这些顶点两两相邻。否则，假设存在某个 i，使得 v_i 未着色，而 $v_{i,i}$ 已着色。现在，n 个 -1 顶点 $v_{i,j} \in B$（其中 $i \neq j$）要么被自身支配，要么被其另一个邻点 $v_j \in A$ 支配。注意，此处每个顶点都由一个不同的顶点支配。因此，至少有一个集合中有 O(n) 个顶点被着色：$\{v_{i,j} | i \neq j\}$ 和$\{v_j | j \neq i\}$。由于这些集合中任意两个顶点的距离至多为 2，因此需要使用 O(n) 种颜色。

5.5. 余图

模宽为二的图是余图 [14]。由引理39可得结论。

引理47。若图 G 是余图，则 $\chi_{ssc}(G) = 2$。

参考文献

[1] Zachary Abel、Victor Alvarez、Erik D Demaine、S´andor P Fekete、Aman Gour、Adam Hesterberg、Phillip Keldenich 和 Christian Scheffer。图的无冲突着色。SIAM离散数学杂志，32(4):2675–2702，2018年。[2] V. P. Abidha、Pradeesha Ashok、Avi Tomar 和 Dolly Yadav。无冲突地为支配集着色：q-子集平方着色。载于《计算机科学——理论与应用》，第17–34页。施普林格国际出版公司，2022年。[3] Pradeesha Ashok、Rathin Bhargava、Naman Gupta、Mohammad Khalid 和 Dolly Yadav。最小无冲突着色的结构参数化。载于《离散应用数学》，爱思唯尔，2022年。[4] Pradeesha Ashok、Rajath Rao 和 Avi Tomar。广义支配问题的多项式核。载于 arXiv 预印本 arXiv:2211.03365，年份：2022年。[5] D. W. Bange。图中的高效支配集。《离散数学应用》，第189–199页，1988年。[6] Alan A Bertossi。分裂图和二分图的支配集。《信息处理快报》，19(1):37–40，1984年。[7] Hans L Bodlaender、Pal Gronass Drange、Markus S Dregi、Fedor V Fomin、Daniel Lokshtanov 和 Michal Pilipczuk。树宽的 cˆkn 5-近似算法。SIAM计算杂志，45(2):317–378，2016年。[8] Hans L Bodlaender、Bart MP Jansen 和 Stefan Kratsch。通过交叉组合得到的核化下界。SIAM离散数学杂志，28(1):277–305，2014年。[9] Hans L Bodlaender 和 Ton Kloks。求图的路径宽和树宽的高效构造算法。算法杂志，21(2):358–402，1996年。[10] Hans L Bodlaender、Sudeshna Kolay 和 Astrid Pieterse。无冲突图着色的参数化复杂性。SIAM离散数学杂志，35(3)，2003–2038，2021年。[11] Yuehua Bu 和 Xubo Zhu。平面图的最优平方着色。《组合优化杂志》，24(4):580–592，2012年。[12] Tiziana Calamoneri。l (h, k)-标号问题：综述与注释书目。《计算机杂志》，49(5):585–608，2006年。[13] Gary Chartrand 和 Ping Zhang。图染色理论。Chap man and Hall/CRC，2008年。[14] David Coudert、Guillaume Ducoffe 和 Alexandru Popa。适用于若干有界团宽图类的完全多项式FPT算法。ACM算法汇刊 (TALG)，15(3):1–57，2019年。[15] Marek Cygan、Fedor V Fomin、Łukasz Kowalik、Daniel Lokshtanov、D´aniel Marx、Marcin Pilipczuk、Michał Pilipczuk 和 Saket Saurabh。参数化算法，第5卷。施普林格，2015年。[16] Rodney G Downey、Michael R Fellows 和 Ulrike Stege。参数化复杂性：系统应对计算难处理性的框架。载于《离散数学当代趋势：从DIMACS和DIMATIA走向未来》，第49卷，第49–99页，1999年。[17] Lawrence H Erickson 和 Steven M LaValle。采用美术馆方法确保地标之间可区分。载于《机器人学：科学与系统》，第7卷，第81–88页，2012年。[18] Ji˘r´ı Fiala、Petr A Golovach 和 Jan Kratochv´ıl。着色问题的参数化复杂性：树宽与顶点覆盖。理论计算机科学，412(23):2513–2523，2011年。[19] Jakub Gajarsk´y、Michael Lampis 和 Sebastian Ordyniak。模块宽度的参数化算法。载于 Gregory Gutin 和 Stefan Szeider（编），《参数化与精确计算》，第163–176页，施普林格·查姆，2013年。施普林格国际出版公司。[20] Robert Ganian。将顶点覆盖改进为图参数。《离散数学与理论计算机科学》，17，2015年。[21] M. R. Garey 和 D. S. Johnson。《计算机与难解性：NP完全性理论指南》（数学科学丛书）。W. H. Freeman，第一版，1979年。[22] Luisa Gargano 和 Adele A. Rescigno。图的无冲突着色的复杂性。理论计算机科学，566:39–49，2015年。[23] Martin Charles Golumbic。算法图论与完美图。爱思唯尔，2004年。[24] Jerrold Griggs 和 Roger Yeh。满足距离2条件的图标号 595, 11 1992 5 586。《SIAM离散数学杂志》，:–。[25] Jan van den Heuvel 和 Sean McGuinness。平面图的平方着色。《图论期刊》，42(2):110–124，2003年。

[26] John E Hopcroft 和 Mukkai S Krishnamoorthy。图的和谐染色。*SIAM*《代数离散方法期刊》，4(3):306–311，1983年。[27] Daniel Lokshtanov、Neeldhara Misra、Geevarghese Philip、MS Ramanujan 和 Saket Saurabh。直径为 (r+ ı) 的图上的 r-支配集问题的困难性。载于 国际参数化与精确计算研讨会，第255–267页。施普林格，2013年。[28] Gary MacGillivray 和 Karen Seyffarth。平面图的支配数。《图论期刊》，22(3):213–229，1996年。[29] Carmen Ortiz 和 M´onica Villanueva。毛毛虫图中的极大独立集。《离散应用数学》，160(3):259–266，2012年。[30] Jeremy Spinrad、Andreas Brandst¨adt 和 Lorna Stewart。二部置换图。《离散应用数学》，18(3):279–292，1987年。[31] Yue-Li Wang、Tsong-Wuu Lin 和 Lin-Yuan Wang。局部和谐色数问题。载于 第*27*届组合数学与计算理论研讨会论文集，台中，台湾，第30卷。Citeseer，2010年。