arXiv:2603.02400v1 [cs.DM] 2 Mar 2026

双向树中请求集的干扰有向图着色

Hugo Boulier[1], David Coudert[2], Frédéric Havet[2], 和 François Pirot[3]

[1]法国雷恩高等师范学校[2]蔚蓝海岸大学、法国国家科学研究中心、法国国家信息与自动化研究所、I3S，索菲亚-安提波利斯，法国[3]LISN，巴黎-萨克雷大学，伊维特河畔吉夫，法国。

2026年3月4日

摘要

本文研究无滤波光网络中广播效应对波长分配问题计算复杂性的影响。我们使用恰当的干扰有向图对冲突进行建模，其正常着色对应于可行的波长分配。因此，最小化所需波长数等价于确定该干扰有向图的色数。在此框架下，我们首先提出一种多项式时间2-近似算法，以最小化波长数。随后，我们证明，当可用波长数 k 作为参数时，该问题是固定参数可解的。我们还推导出多项式时间算法，用于计算该干扰有向图的独立数和团数。

1 引言

本文研究无滤波光网络中波长分配问题的计算复杂性。这类网络采用波分复用（WDM）技术共享光纤的光谱，并利用无源光学器件（分路器和合路器）在路由节点之间建立连接。回顾一下，光网络由一组通过有向光纤连接的路由节点构成，因此，这类网络的网络拓扑可以自然地建模为有向图。在WDM网络中，每个业务请求都必须分配一条从源节点到目的节点的光路，即网络中的一条路径及一个波长。波长分配必须满足波分复用约束：使用同一根光纤的两条光路必须分配不同的波长。光路由节点通常配备可重构光分插复用器（ROADMs），用于将每条光路从输入端口引导至输出端口。在WDM网络中使用ROADMs可以非常高效地利用光谱

但网络的总体成本较高。为在网络成本与频谱效率之间寻求更好的折中，人们提出使用廉价的无源光学器件 [22]。

在无滤波光网络中， [1,16], 进入路由器节点的光信号会被简单地广播到该节点所有的出向光纤，但不包括与输入光纤方向相反的那一条。这是通过廉价的无源光器件（分路器和合路器）实现的。此外，某些节点配备了阻断器，用于在特定节点处阻止光信号继续传播。为简化此类网络的管理，通常会放置阻断器，将网络划分为若干棵树，并将每个业务请求分配给唯一的一棵树。由于广播效应，与配备可重构光分插复用器的WDM网络相比，无滤波光网络的频谱效率较低。具体而言，光路所用的波长会被广播至整棵树，从而增加波长分配中的冲突数量。不过，网络建设的资本支出以及控制网络运行的运营支出都会降低。

无滤波光网络中的一般优化问题是：给定网络拓扑和一组业务请求，将网络拓扑划分为若干棵树，并为每个业务请求分配一条位于某棵树中的光路，使网络中使用的不同波长总数最少。已有多种精确方法和启发式方法用于求解这一优化问题， [6, 11–15, 18–20]。特别是， [11, 12] 提出了该问题的数学模型，该模型将具有树形拓扑的子网络、业务请求和光路作为变量。由于树中的路由显而易见，这一模型引出了本文所研究的波长分配问题的计算复杂度问题。

注意，当为每个业务请求给定一条路径作为输入时，配备可重构光分插复用器的 WDM 网络中的波长分配问题，等价于在有向图中对一组路径进行着色，使共享同一条弧的两条路径获得不同颜色 [3]。即使有向图是双向有向环 [21] 或双向树 [8,17]，这一问题仍然是 NP完全问题 [2,5] 。事实上，路径着色问题对应于路径冲突图中的顶点着色问题。该冲突图中，每条路径对应一个顶点；若两条路径共享一条弧，则在对应的两个顶点之间连一条边。

1.1 建模

在无滤波光网络中，广播效应会增加波长分配时业务请求之间的冲突数量。我们使用一个干扰有向图 来建模这些冲突，其中每个业务请求对应一个顶点；若请求 r 的广播树与请求 r'的光路相交，则从顶点 r 到顶点 r' 连一条弧。于是，为请求分配波长等价于对干扰有向图的顶点进行适当着色。

更确切地说，广播树可以用一棵双向树 T建模，也就是说，它是由无向树将每条边替换为两个方向相反的弧而得到的有向图。 T 中的请求是 T 的一条长度至少为 1 的有向子路径。令 r 为一个

请求，或更一般地说，一条有向路径。我们用 s_r 表示其第一个顶点，用 t_r 表示其终极顶点，用 s_r^+表示其第二个顶点，用 t_r^-表示其倒数第二个顶点。 r的发射弧 是弧 $e_r^+ = (s_r,\ s_r^+)$，而 r的接收弧 是弧 $e_r^- = (t_r^-,\ t_r)$。

设 R 是双向树 T 中的请求集合。请求 r 干扰请求 r'，当且仅当 $T[s_r, t_{r'}]$：从 s_r 到 $t_{r'}$ 在 T 中的唯一有向路径，其第一条弧为 e_r^+，末端弧为 $e_{r'}^-$。 R 在 T 中的干扰有向图记为 $\vec{\mathrm{I}}(R,T)$，它是一个顶点集为 R 的有向图，其中 (r,r') 是一条弧，当且仅当 r 干扰 r'。如果两个请求中的一个干扰另一个，则称这两个请求相互干扰。 R 在 T 中的干扰图记为 $\mathrm{I}(R,T)$，是 $\vec{\mathrm{I}}(R,T)$ 的底层图。换言之，它是一个顶点集为 R 的图，其中 $\{r,r'\}$ 是一条边，当且仅当 r 和 r' 相互干扰。

1.2 我们的结果

在本文中，给定一棵双向树 T 以及其中的请求集合 R，我们关注 $\omega(R,T)$：两两干扰的请求的最大子集的大小；以及 $\alpha(R,T)$：两两不干扰（或独立）的请求的最大子集的大小。等价地，$\omega(R,T)$ 是 $\mathrm{I}(R,T)$ 的团数，即 $\omega(R,T) = \omega(\mathrm{I}(R,T))$；而 $\alpha(R,T)$ 是 $\mathrm{I}(R,T)$ 的独立数，即 $\alpha(R,T) = \alpha(\mathrm{I}(R,T))$。我们还关注最小数量 $\chi(R,T)$ 的颜色（波长或频率），将其分配给请求，使任意两个相互干扰的请求获得不同的颜色。这就是 $\mathrm{I}(R,T)$ 的色数，即 $\chi(R,T) = \chi(\mathrm{I}(R,T))$。

我们在第3节中证明，给定 R和 T，可以在多项式时间内计算 $\alpha(R,T)$和 $\omega(R,T)$。随后，在第4节中，我们证明 $\chi(R,T) \leq 2\omega(R,T)$，并描述了一个用于 $\chi(R,T)$的简单2-近似多项式时间算法。

随后，我们在第5节中研究计算 $\chi(R,T)$的复杂性以及如下相关判定问题。干扰可着色性 输入：双向树 T中的请求集合 R，以及整数 k。问题： $\chi(R,T) \leq k$？

特别地，我们考虑当 k 固定时该问题的受限情形。干扰 k-可着色性 输入：双向树 T中的请求集合 R。问题：$\chi(R,T) \leq k$？

我们在第5.1节中证明，干扰3-可着色性可以在$\mathcal{O}(|R|^2)$时间内求解。随后，在第5.2节中，我们证明带有 $k \geq 4$ 的干扰 k-可着色性在以 k为参数时是固定参数可解（固定参数可解）的，或等价地，以$\mathcal{I}(R,T)$中最大团的大小为参数时亦然。我们提出了一个时间复杂性为$\mathcal{O}(k36^k|R|^3)$的固定参数算法来求解该问题。

我们在第6节以一些开放问题作为本文的结尾。

2 预备知识

2.1 记号与定义

出树是指所有弧均背离根定向的定向有根树；入树则是指所有弧均朝向根定向的定向有根树。给定一棵双向树 T 和一个顶点 $z \in V(T)$，我们用 $T[z]$ 表示与 T 同构且根为 z 的有根树。双向树中的枝是连接两个叶子的双向路径，而有根双向树中的分支是从根到任意叶子的双向路径。有根双向树中的入枝是从叶子到根的有向路径，出分支则是从根到叶子的有向路径。

给定一条有向路径 $P = (v_0, \ldots, v_\ell)$，对于任意 $0 \le i \le j \le \ell$，我们用 $P[v_i, v_j]$ 表示有向子路径 $(v_i, \ldots, v_j)$。更一般地，如果 T 是双向树、入树或出树，我们用 $T[x, y]$ 表示 T 中从 x 到 y 的唯一有向路径（如果存在）。如果 T 是双向树，我们用 $T\langle x, y\rangle$ 表示 T 中端点为 x 和 y 的双向路径。

有向图 D 的逆向图记为 $\overleftarrow{D}$，它是通过反转所有弧的方向从 D 得到的有向图：$V(\overleftarrow{D}) = V(D)$ 和 $A(\overleftarrow{D}) = \{(u, v) : (v, u) \in A(D)\}$。

给定图 G 以及其顶点集合 S，记 S 为集合 $V(G) \setminus S$，记 $G\langle S\rangle$ 为由 S 诱导的 G 子图。顶点 v 的邻域记为 $N(v)$，其闭邻域 记为 $N[v] = N(v) \cup \{v\}$。

2.2 化简问题

缩小树的规模。设 R 是双向树 T 中的一个请求集。注意，若 T 中有一对反向弧既不是发射弧也不是接收弧，则我们可以在 T 和请求中收缩这两条弧（即删除它们并合并其端点），从而得到一个新的双向树 T' 和请求集 R'，其干扰有向图相同：$\vec{\mathcal{I}}(R', T') = \vec{\mathcal{I}}(R, T)$。依次收缩每一对既不包含发射弧也不包含接收弧的反向弧后，我们可以在 $\mathcal{O}(|T|)$ 时间内将问题化简为一个实例 (T^*, R^*)，使得 $\vec{\mathcal{I}}(R^*, T^*) = \vec{\mathcal{I}}(R, T)$ 且 $|V(T^*)| \le 2|R^*| + 2$。

因此，不妨先执行上述$\mathcal{O}(|T|)$时间归约，**我们假设每一数对反向弧都包含一条发射弧或接收弧**，因此

$$|V(T)| \le 2|R| + 2.$$

计算干扰有向图。对于每一个 $|R|$ 请求 $r \in R^*$，可以在$\mathcal{O}(|T|) = \mathcal{O}(|R|)$时间内构建出树 $T^+(r)$，其顶点是从 s_r^+在 $T \setminus \{s_r s_r^+, s_r^+ s_r\}$ 中可达的，从而找出 r在 $\vec{\mathcal{I}}(R, T)$中的出邻点。因此，在$\mathcal{O}(|R|^2)$时间内可以计算出 $\vec{\mathcal{I}}(R, T)$。注意，该时间在一般情况下已是最快的，因为$\vec{\mathcal{I}}(R, T)$可以有$\Omega(|R|^2)$条弧。

在我们提供的算法中，有时需要构造树上请求子集的干扰有向图。每当需要这样做时，相关成本

都不会超过在该有向图上执行操作的成本。因此，为了简化证明，我们将这些构造隐含处理，从不提及其成本。

2.3 结构性质

设 T 是一棵以 z为根的双向树。它是入树 T^-与出树 T^+的弧并，前者的弧是指向根的弧，后者的弧是背离根的弧。

在 T中，一条有向路径（特别是一个请求）若指向 z （即包含于 T^-中），则为汇聚；若背离 z （即包含于 T^+中），则为发散；否则为单峰（即 T^-的一条子路径与 T^+的一条子路径的拼接）。给定 T上的请求集合 R，我们用 R^-（相应地， R^+、$R^\vee$）表示 R中汇聚（相应地，发散、单峰）请求的集合。根据定义，（$R^-, R^+, R^\vee$）是 R的一个划分。

若 r 是 T的一条有向子路径，则 r的中点（记为 m_r）是其中最接近根的顶点。若 r 是汇聚（相应地，发散）路径，则 $m_r = t_r$ （相应地，$m_r = s_r$）。若 r 是单峰的，则它是 r 中作为一条汇聚弧的头端且作为一条发散弧的尾端的顶点。沿 r 的 m_r 的前驱记为 m_r^-，其后继记为 m_r^+（若它们存在）。

设 x， y为 T的两个顶点。若 $T[x, y]$是发散的，我们称 x为 y的祖先，称 y为x的后代。只含一个顶点的路径是发散的，因此每个顶点都是自身的祖先。若 x 是 y 的祖先，或 y 是 x的祖先，我们称 x 和 y 是相关的。换言之，当且仅当 $T[x, y]$是汇聚的或发散的， x和 y才相关。注意，同一顶点的两个祖先彼此相关。

x 和 y 的最近公共祖先 是距离根 z最远、且同时是 x和 y祖先的顶点。这对应于 $T[x, y]$的中点。

现在我们可以陈述并证明一个引理，用以刻画两个请求何时会发生干扰。有时我们会直接使用这个引理，而不明确提及它。

引理1。 设 T 为一棵以 z为根的双向树。

(i) 两个汇聚请求 r, r' 干扰，当且仅当 t_r^- 和 $t_{r'}^-$ 相关。(ii) 两个发散请求 r, r' 干扰，当且仅当 s_r^+ 和 $s_{r'}^+$ 相关。(iii) 一个汇聚请求 r 和一个发散请求 r' 干扰，当且仅当 s_r 和 $t_{r'}$ 不相关。(iv) 两个单峰请求 r, r' 不干扰，当且仅当 $m_r = m_{r'}$、s_r 和 $t_{r'}$ 相关，且 $s_{r'}$ 和 t_r 相关。(v) 一个单峰请求 r 和一个汇聚请求 r' 不干扰，当且仅当 m_r 是 $t_{r'}$ 的祖先，且 t_r 和 $s_{r'}$ 相关。

(vi) 一个单峰请求 r 和一个发散请求 r' 不干扰，当且仅当 m_r 是 $s_{r'}$ 的祖先，且 s_r 和 $t_{r'}$ 相关。

证明。(i) 令 r 和 r' 为两个汇聚请求。

假设 r 在 r' 上发生干扰。则 $T[s_r, t_{r'}]$ 的第一条弧为 e_r^+，末端弧为 $e_{r'}^-$。由于 r 和 r' 是汇聚的，这两条弧 e_r^+ 和 $e_{r'}^-$ 都指向根，因此 $T[s_r, t_{r'}]$ 是一条汇聚路径。由此，t_r^- 和 $t_{r'}^-$ 都是 s_r 的祖先，因此二者存在祖先关系。

反过来，假设 t_r^- 和 $t_{r'}^-$ 存在祖先关系。不失一般性，可假设 $t_{r'}^-$ 是 t_r^- 的祖先。但此时 $T[s_r, t_{r'}]$ 是一条汇聚路径，其第一条弧为 e_r^+，末端弧为 $e_{r'}^-$。由此，r 在 r' 上发生干扰。

(ii) 根据方向对偶，证明与 (i) 类似。

(iii) 假设 r 和 r' 发生干扰。由于 T 中的有向路径要么是汇聚的、发散的，要么是单峰的，因此不可能第一条弧发散而末端弧汇聚。特别地，T 中不存在第一条弧为 $e_{r'}^+$ 且末端弧为$e_{r'}^-$的有向路径。因此，r' 不可能在 r 上发生干扰。于是，r 在 r' 上发生干扰，且 $T[s_r, t_{r'}]$ 是一条第一条弧为 e_r^+、末端弧为 $e_{r'}^-$ 的有向路径。由于 e_r^+ 是汇聚的而 $e_{r'}^-$ 是发散的，$T[s_r, t_{r'}]$ 是单峰的，并且 s_r 和 $t_{r'}$ 不存在祖先关系。

反之，假设 s_r和 $t_{r'}$没有关联。那么 $T[s_r, t_{r'}]$是单峰的。因此，这条路径的第一条弧是汇聚的，故必为 e_r^+；其末端弧是发散的，故必为 $e_{r'}^-$。因此，r 在 r'上干扰。

(iv) 设 r 和 r' 是两个单峰请求。

首先假设 s_r和 $t_{r'}$没有关联，则 $T[s_r, t'_r]$是一条经过它们最近公共祖先的有向路径，其第一条弧为 e_r^+，末端弧为 $e_{r'}^-$。因此，r 在 r'上干扰。

类似地，如果 $s_{r'}$ 和 t_r 没有关联，则 r' 在 r上干扰。

现在假设 s_r 和 $t_{r'}$ 有关联，且 $s_{r'}$ 和 t_r 有关联。那么必有$m_r = m_{r'}$。此外，很容易看出 r 和 r' 不会相互干扰。

(v) 设 r 是一个单峰请求，r' 是一个汇聚请求。如果 m_r 和 $t_{r'}$ 没有关联，那么根据 (iii)，$T[m_r, t_r]$ 和 r'会相互干扰，因此 r和 r'也会相互干扰。

此后，我们可以假设 s_r 和$t_{r'}$ 有关联。那么$T[s_{r'}, t_r]$ 要么是汇聚路径，要么是发散路径，因此不包含汇聚弧$e_{r'}^+$ 或发散弧 e_r^-。因此，r' 不会干扰 r。

如果 $t_{r'}$ 是 m_r 和 $t_{r'} \neq m_r$的祖先，那么 $T[s_r, t_{r'}]$ 是一条有向路径，其首弧为 e_r^+，末端弧为 e_r^-。因此，r 会干扰 r'。如果 m_r 是 $t_{r'}$的祖先，那么$T[s_r, t_{r'}]$ 是一条单峰路径，因此不包含汇聚弧 $e_{r'}^-$。因此，r和 r' 不会相互干扰。

(vi) 根据方向对偶，与 (v) 类似。 □

由于树中任意一组两两相关的节点必定处于同一分支上，我们立即由引理1(i)和引理1(ii)得出以下结论。

推论2。设 T 是以 z为根的双向树。

(i) 如果 W 是汇聚请求团，那么 T 中存在一条入枝，包含 W中所有请求的接收弧。(ii) 如果 W 是发散请求团，那么 T 中存在一条出分支，包含 W中所有请求的发射弧。

可比图是一种图，存在一个偏序，使得两个顶点之间有边，当且仅当它们在该偏序中可比。共二部图是二部图的补图。换言之，共二部图的顶点集可以划分为两个团。

可比图和共二部图都是完美图 [7]。因此，对于任意可比图或共二部图 G， $\omega(G) = \chi(G)$。此外，可比图和共二部图的团数、独立数和色数均可在多项式时间内计算。

设 $G_i, i \in I$ 为图。 $G_i, i \in I$ 的不交并记作 $\biguplus_{i\in I} G_i$，它是每个 G_i 的顶点不交副本的并。 $G_i, i \in I$ 的联接记作 $\bigoplus_{i\in I} G_i$，它是在 $\biguplus_{i\in I} G_i$ 中，对于所有 i、$j \in I$、$i \neq j$，在（ G_i 的副本）与（ G_j 的副本）之间添加所有可能的边后得到的图。

下面的命题简单且广为人知。

命题3。 (i) 可比图的不交并和联接仍是可比图。

(ii) 共二部图的联接是共二部图。

推论 4。 设 R 是有根双向树 T 上的请求集。我们有：

(i) $I(R^-, T)$ 是可比图，因此 $\omega(R^-, T) = \chi(R^-, T)$。*(ii)* $I(R^+, T)$ 是可比图，因此 $\omega(R^+, T) = \chi(R^+, T)$。*(iii)* $I(R^\vee, T)$ 是共二部图，因此 $\omega(R^\vee, T) = \chi(R^\vee, T)$。

证明。 (i) 直接由引理1(i)得出。

(ii) 直接由引理1(ii)得出。

(iii) 令 z 为 T的根。

令 x 为 T的一个顶点。我们将 $R_x^\vee$ 记为 R 中以 x为中点的单峰请求的集合。令 $S = \{s_1, \ldots, s_p\}$ 为 x的子节点 集合，它们是 x的邻居且不属于 $T[z, x]$。注意，对于 $R_x^\vee$中的每个请求， m_r^-和 m_r^+都是 S中的不同顶点。因此，令 i_r^- 为满足 $m_r^- = s_{i_r^-}$ 的指标，并令 i_r^+ 为满足 $m_r^+ = s_{i_r^+}$的指标。

令 A_x （分别地， B_x）为 $R_x^\vee$ 中满足 $i_r^- < i_r^+$（分别地， $i_r^- > i_r^+$）的请求集合。显然，(A_x, B_x) 是 $R_x^\vee$的一个划分。此外，根据引理1(iv)， A_x 和 B_x 是$\mathcal{I}(R_x^\vee, T)$中的团。因此，$\mathcal{I}(R_x^\vee, T)$是一个共二部图。

现在，由引理1(iv)，$\mathcal{I}(R^\vee, T)$ 是 $\mathcal{I}(R_x^\vee, T)$ 与 $x \in V(T)$ 的联接。因此，由命题3，$\mathcal{I}(R^\vee, T)$ 是一个共二部图。 □

引理 5。 设 r^-、 r^+ 和 $r^\vee$ 分别是有根双向树 T 上的一个汇聚请求、一个发散请求和一个单峰请求。那么，$\{r^-、r^+、r^\vee\}$ 中有两个请求会相互干扰。

证明。 如果 $r^\vee$ 不与 r^-, r^+中的任何一个干扰，那么根据引理1 (v)和(vi)， $m_{r^\vee}$ 是 t_{r^-} 和 s_{r^+}的共同祖先， $t_{r^\vee}$ 和 s_{r^-} 相关，且 $s_{r^\vee}$ 和 t_{r^+} 相关。

那么 $T[s_{r^-}, t_{r^+}]$ 是一条中点为 m^r、首弧为 $e^+_{r^-}$、末端弧为$e^-_{r^+}$的单峰路径。因此，r^- 在 r^+上发生干扰。 □

引理 6。 设 R 是有根双向树 T上的请求集， $I \subseteq R$ 是一个独立请求集。那么，要么 I 恰好包含两个单峰请求，要么$|I \cap (R^+ \cup R^\vee)| \leq 1$，要么 $|I \cap (R^- \cup R^\vee)| \leq 1$。

证明。 如果 I 包含两个单峰请求 r, r'，那么根据引理1(iv)， r 和 r' 的弧方向相反（例如(m_r, m_r^+)和($m_{r'}^-, m_{r'}$)，因为 $m_r = m_{r'}$ 和 $m_r^+ = m_{r'}^-$ ）。因此，$\{r, r'\}$与所有其他请求都发生干扰。因此，$\{r, r'\}$。

如果 I 恰好包含一个单峰请求，则结论由引理 5 得出。

下文中，我们可以假设 $I \cap R^\vee = \varnothing$。

令 $I^- := I \cap R^-$ 且 $I^+ := I \cap R^+$；注意 $I^- \cap I^+ = \varnothing$。为导出矛盾，假设 I^- 包含两个请求 r_1 和 r_2，且 I^+ 包含两个请求 r_3和 r_4。根据引理1(i)， t_{r_1} 和 t_{r_2} 在 T 中不具有关联；根据引理1(ii)， s_{r_3} 和s_{r_4} 在 T中不具有关联。此外，根据引理1(iii)， s_{r_3} 与 t_{r_1} 和 t_{r_2}均具有关联，因此它是二者的公共祖先。s_{r_4}也同样如此，这与s_{r_3} 和 s_{r_4} 不具有关联这一事实相矛盾。 □

3 干扰图的独立数和团数

在本节中，我们证明，任何干扰图的独立数（第3.1节）和团数（第3.2节）都可以在多项式时间内计算。

3.1 计算 $\alpha(R, T)$

定理 7。 给定双向树 T 中的请求集合 R，存在一种算法，可在 $O(|R| \log |R| + |T|)$ 时间内返回 $I(R, T)$ 的最大独立集。

证明。 将 T 以某个（任意）顶点 z 为根，并令 $\leq_T$ 表示 T 中的祖先关系，即，对于 $x, y \in V(T)$，当且仅当 x 是 T 中 y 的祖先时， $x \leq_T y$。众所周知，该关系具有自反性、反对称性和传递性。

令 $R^+ = r_1$、.....、 r_n 为 R 中按其发射弧进行后序深度优先搜索（DFS）排序的发散请求（因此，如果 $s^+_{r_i}$ 是 T 中 $s^+_{r_j}$ 的祖先，则必须有 $i \geq j$）。令 $S^+ \subseteq V(T)$ 表示节点 $x \in V(T)$ 的集合，其中存在一个发散请求 $r \in R^+$，使得 $x = s^+_r$。

命题 7.1。 设 $I \subseteq R^+$ 是由发散请求组成的独立集。若存在一个请求 $r \in I$ 和一个请求 $r' \in R^+ \backslash I$，使得 s^+_r 是 T 中 $s^+_{r'}$ 的祖先，则 $(I \backslash \{r\}) \cup \{r'\}$ 也是由发散请求组成的独立集。

命题证明。根据 $\leq_T$ 的传递性和定理 1(ii)，如果存在一个与 r' 干扰的请求 $r'' \in R^+$，则 r'' 也会与 r 干扰。因此 $r'' \notin I \setminus \{r\}$，由此我们推断 $(I \setminus \{r\}) \cup \{r'\}$ 确实是一个独立集。◊

根据命题 7.1，通过对每个极小元（相对于 $\leq_T$）在 S^+ 中选取一个请求 $r \in R^+$，使得 $s_r^+ = x_0$，自然可以得到一个由发散请求组成的最大独立集 $I^+ \subseteq R^+$。

我们采用对称的方法构造由汇聚请求组成的最大独立集 $I^- \subseteq R^-$，即令 $R^- = q_1, \ldots, q_m$ 为 R 中按其接收弧进行后序深度优先搜索（DFS）排序的汇聚请求（因此，如果 $t_{r_i}^-$ 是 T 中 $t_{r_j}^-$ 的祖先，则必须有 $i \geq j$），并令 $S^- \subseteq V(T)$ 为满足以下条件的节点 $x \in V(T)$ 集合：存在一个汇聚请求 $r \in R^-$，使得 $x = t_r^-$。

到目前为止，我们有 $|I^+| = \alpha(R^+, T)$ 和 $|I^-| = \alpha(R^-, T)$。

命题 7.2。设 $I \subseteq R^+$ 是一个由发散请求组成的独立集，其大小至少为 2，并设 $S := \{t_r \mid r \in I\}$。设 $x_0 := \mathrm{LCA}(S)$ 为 S 的最近公共祖先，并设$r \in R^- \cup R^\vee$。那么，当且仅当 s_r 是 T 中 x_0 的祖先时，$I \cup \{r\}$ 是一个独立集。

命题证明。如果 s_r 是 T 中 x_0 的祖先，那么，根据定理 1(iii)（当 $r \in R^-$ 时）或定理 1(vi)（当 $r \in R^\vee$ 时），可知 $I \cup \{r\}$ 是一个独立集。

反之，如果 s_r 不是 T 中 x_0 的祖先，那么存在一个请求 $r' \in I$，使得s_r与 $t_{r'}$ 在 T 中不相关，因此 r 与 r' 会发生干扰，所以 $I \cup \{r\}$ 不是一个独立集。◊

如果存在请求 $r^- \in R^- \cup R^\vee$，使得 s_{r^-} 是 T 中 LCA($\{t_r \mid r \in I^+\}$) 的祖先，那么我们就将其加入 I^+。对于 I^-，我们以对称的方式处理；如果存在请求$r^+ \in R^+ \cup R^\vee$，使得 t_{r^+} 是 T 中 LCA($\{s_r \mid r \in I^-\}$) 的祖先，那么我们就将其加入 I^-。

令 I 为 I^-、I^+ 中较大的那个。如果 $|I| \geq 2$，则返回 I；否则，我们寻找一对请求 r、$r' \in R$，使得 r 和 r' 不会相互干扰；若找到这样的数对，则返回 $I := \{r$、$r'\}$。如果到目前为止仍未返回任何结果，我们就得出结论：$\alpha(R) = 1$（假设$R \neq \emptyset$），并且可以为任意请求 $r \in R$ 返回 $I := \{r\}$。

命题 7.3。返回的独立集 I的大小为 $\alpha(R, T)$。

命题证明。若 $\alpha(R,T) \leq 2$或 $\alpha(R,T) = \max\{\alpha(R^+,T)\ \alpha(R^-,T)\}$，则显然上述算法返回的是 I$(R,T)$ 的最大独立集。现在假设~ $\alpha(R,T) > \max\{2, \alpha(R^+,T)\ \alpha(R^-,T)\}$，并令 I为 I(R,T) 的最大独立集~ ~。由定理 6 可知，I要么包含 $|I|-1$个发散请求，要么包含 ~ ~ ~$|I|-1$个汇聚请求。根据对称性，不妨假设 I包含 $|I|-1$个发散请求，这意味着

$$\alpha(R,T) = 1 + \alpha(R^+,T),$$

因此，特别是 $\alpha(R^+, T) \geq 2$。

如果 r^- 不存在，那么根据命题 7.2，$\mathcal{I}(R^+,\ T)$ 中不存在能够扩展为$\mathcal{I}(R,\ T)$ 中更大独立集的最大独立集，这就产生了矛盾~（$I \cap R^+$ 是这样的独立集）。由此可知，r^- 存在，因此在算法结束时，$|I^+| = 1 + \alpha(R^+,\ T) = \alpha(R,\ T)$。因此，返回的独立集是$\mathcal{I}(R,\ T)$ 中的~~最大独立集。情形 $|I \cap R^-| = |I| - 1$ 可由对称性得到。◊

命题 7.4。 上述算法可以实现为在 $O(|R| \log |R|+|T|)$时间内运行。

命题证明。可以通过对 T 执行深度优先搜索，在 $\mathcal{O}(|R|+|T|)$ 时间内构造 S^+ 和 S^-，然后在 $\mathcal{O}(|R|)$ 时间内初始化 I^+ 和 I^-。众所周知，经过 $\mathcal{O}(|T|)$ 时间的预处理后，可以在 T 中以常数时间完成最近公共祖先查询，因此可以在 $\mathcal{O}(|R|+|T|)$ 时间内计算 $\mathsf{LCA}(\{t_r \mid r \in I^+\})$。

除非算法结束时出现 $|I^+| = |I^-| = 1$，否则我们已经完成；若出现这种情况，就需要高效地寻找一对由独立请求组成的数对 $\{r, r'\}$。显然，可以通过测试所有请求对，在 $\mathsf{O}(|R|^2)$ 时间内完成这项工作。下面我们将说明如何在 $\mathsf{O}(|R| \log |R| + |T|)$ 时间内完成此任务。为此，我们需要在 T 中执行大量祖先查询；为了高效地完成这些查询，我们将其转换为范围查询，方法如下。令 $E = (v_1, \ldots, v_m)$ 为双向树 T 的一次欧拉游历，从根 z 出发（并最终返回该根）。由 E 可以构造两个数组 $\mathtt{tin}$和 $\mathtt{tout}$，使得对于每个 $x \in V(T)$，$\mathtt{tin}[x] = \min\{i \mid v_i = x\}$ 且 $\mathtt{tout}[x] = \max\{j \mid v_j = x\}$。注意，$(v_{\mathtt{tin}[x]}, \ldots, v_{\mathtt{tout}[x]})$ 是双向子树 $T[x]$ 的一次欧拉游历，该子树属于以 x 为根的 T。特别地，在 T 中，x 是 y 的祖先，当且仅当 $\mathtt{tin}[y] \in [\mathtt{tin}[x], \mathtt{tout}[x]]$（这一点可以在常数时间内检查）。

请求 r 和 r' 的类型在单模态、汇聚或发散之间有几种可能的组合，每种都需要单独处理，因为它们各自有不同的干扰条件（见定理 1）。

单模态/单模态对 为了检测这样的独立对，我们首先根据由 $r \sim r'$ 定义的关系 $\sim$，将 $R^\vee$ 划分为等价类，其中 $m_r = m_{r'}$。借助哈希表（其键为 $\{m_r \mid r \in R^\vee\}$），可以在（期望）时间$\mathcal{O}(|R|)$内完成此操作；若使用基于平衡二叉搜索树的字典，则可在确定性时间$\mathcal{O}(|R| \log |R|)$内完成。现在，在每个等价类 X中，我们要寻找一对 $r, r' \in X$，使得 s_r和 $t_{r'}$，并且 t_r和 $s_{r'}$在 T中相关。除对称情况以及交换 r 和 r'的角色外，需要考虑两种情况。

(i) 我们首先寻找一对 $r,\ r' \in X$，满足 $s_r \leq_T t_{r'}$（s_r 是 T 中 $t_{r'}$ 的祖先）且 $t_r \leq_T s_{r'}$。为此，对于每个 $r \in X$，我们寻找一个请求 $r' \in X \setminus \{r\}$，使得

$$\begin{cases} \mathtt{tin}[t_{r'}] \in [\mathtt{tin}[s_r], \mathtt{tout}[s_r]]; \text{and} \\ \mathtt{tin}[s_{r'}] \in [\mathtt{tin}[t_r], \mathtt{tout}[t_r]]. \end{cases}$$

因此，我们执行 $|X|$ 2维范围查询。利用二维范围树子结构，可以在$\mathcal{O}(|X| \log |X|)$时间内完成（参见 $[4, \text{定理 } 5.11]$）。

请注意，通过交换 r 和 r'的角色，这也涵盖了满足$s_{r'} \leq_T t_r$ 和 $t_{r'} \leq_T s_r$的数对（r， r'）。

(ii) 接下来，我们要寻找一对 $r, r' \in X$，使得 $s_r \leq_T t_{r'}$ 和 $s_{r'} \leq_T t_r$。利用定理 **4(iii)**，$\mathcal{I}(X, T)$ 是一个共二部图，其两个部分 A, B 可在线性时间内计算。因此，我们只需考虑数对 $(r, r') \in A \times B$ 和 $(r, r') \in B \times A$（后者根据对称性以类似方式处理）。我们分别按照相对于 s_r 和 $t_{r'}$ 的前序深度优先搜索排序，对 T 中的请求 $r \in A$ 和 $r' \in B$ 进行排序（等价地，按照 $\texttt{tin}[s_r]$ 和 $\texttt{tin}[t_{r'}]$ 递增的顺序排序）。为了找到一个独立对 $(r, r') \in A \times B$，我们按如下方式进行。

- 令 D 为一棵空的一维范围树（即一棵平衡二叉搜索树）。在算法的每一步，D 都将包含一个请求子集$r \in A$，其中所有请求的节点 s_r 都位于 T 的同一条分支上，并且各自关联着键 $\texttt{tin}[t_r]$。
- 对于每个 $i \in \{1, \ldots, |E|\}$（回顾一下，E是 T的欧拉游历）：– 将所有满足 $\texttt{tin}[s_r] = i$ 且键为 $\texttt{tin}[t_r]$的请求 $r \in A$ 添加到 D 中；– 从 D 中移除所有满足 $\texttt{tout}[s_r] = i - 1$的请求 $r \in A$；– 对于每个满足 $\texttt{tin}[t_{r'}] = i$的请求 $r' \in B$，在 $O(\log |D|)$ 时间内测试是否存在一个满足 $\texttt{tin}[t_r] \in [\texttt{tin}[s_{r'}], \texttt{tout}[s_{r'}]]$的请求 $r \in D$。如果存在，这意味着 $s_{r'} \leq_T t_r$，且由于 $r \in D$，我们有$\texttt{tin}[t_{r'}] = i \in [\texttt{tin}[s_r], \texttt{tout}[s_r]]$，即 $s_r \leq_T t_{r'}$。因此，我们可以停止搜索并返回数对 (r, r')。

上述过程的运行时间为 $\mathcal{O}(|B| \log |A| + |T|)$，其对称过程（交换 A和 B的角色）的运行时间为 $\mathcal{O}(|A| \log |B| + |T|)$。
注意，寻找满足 $t_r \leq_T s_{r'}$ 和 $t_{r'} \leq_T s_r$ 的数对 (r, r') 是对称的（只需交换 s_r 和 t_r 的角色），因此可以用类似的方法完成。

单峰／发散对 对于每个单峰请求 $r \in R^\vee$，我们寻找一个请求 $r' \in R^+$，使得 $m_r \leq_T s_{r'}$，并且满足 $s_r \leq_T t_{r'}$ 或 $t_{r'} \leq_T s_r$。第一个条件可转化为

$$\begin{cases} \texttt{tin}[s_{r'}] \in [\texttt{tin}[m_r], \texttt{tout}[m_r]]; \text{and} \\ \texttt{tin}[t_{r'}] \in [\texttt{tin}[s_r], \texttt{tout}[s_r]], \end{cases}$$

其处理方式与单模态/单模态对的情形 **(i)** 类似。第二个条件可转化为

$$\begin{cases} \texttt{tin}[s_{r'}] \in [\texttt{tin}[m_r], \texttt{tout}[m_r]]; \text{and} \\ \texttt{tin}[s_r] \in [\texttt{tin}[t_{r'}], \texttt{tout}[t_{r'}]], \end{cases}$$

其处理方式与单模态/单模态对的情形 **(ii)** 类似。

单模态/收敛对 这与前一种情况对称，因此采用类似的方法处理。

发散／汇聚对 对于每个 $r \in R^+$，我们要找一个请求 $r' \in R^-$，使得$t_r \le_T s_{r'}$（即 $\texttt{tin}[s_{r'}] \in [\texttt{tin}[t_r], \texttt{tout}[t_r]]$）；对于每个 $r \in R^-$，我们要找一个请求$r' \in R^+$，使得 $s_r \le_T t_{r'}$（即 $\texttt{tin}[t_{r'}] \in [\texttt{tin}[s_r], \texttt{tout}[s_r]]$）。这可以通过在一维范围树上进行范围查询来完成，总成本为 $\mathcal{O}(|R| \log |R|)$。

The overall cost of all these tests is in $\mathcal{O}(|R| \log |R| + |T|)$ time. ◊

结合上述论断，可以实现一个$\mathcal{O}(|R| \log |R| + |T|)$时间的算法：当输入为 (R, T) 时，该算法返回$\mathcal{I}(R, T)$ 的最大独立集，正如所需。

□

3.2 计算 $\omega(R, T)$

定理8。给定双向树 T 上的请求集合 R，可以在 $O(|R|^{4.5})$ 时间内计算 $I(R, T)$ 的团数 $\omega(R, T)$。

为了证明定理8，我们需要对干扰图中的团建立一个结构性结果。我们先介绍以下事实。

引理9。设 T 是一棵有根双向树，Q 是 T 的一个分支。设 r_0 是与 Q 相交的请求，r 是与 Q 弧不相交的单峰请求弧。那么 r 和 r_0 会相互干扰。

证明。考虑 r 和 r_0 的逆向图并不失一般性；它们相互干扰当且仅当 r 和 r_0 相互干扰，因此我们可以假设 r_0 与 Q 在一条汇聚弧 (u, v) 上相交。

反设 r 与 r_0 互不干扰。则由引理1 (v)，m_r 是 v的祖先，且 t_r 与 u 相关。因此（m_r, m_r^+）是 Q中的一条弧，矛盾。

□

引理10. 设 R 是双向树 T上的一个请求集，并设 W 是干扰图$I(R, T)$中的一个团。存在 T的一个枝 Q_W 以及一个顶点 $x_W \in V(Q_W)$，使得 W中所有与 Q_W弧不相交的请求在 $T[x_W]$中都是单峰的。

此外，数对集合(Q_W, x_W) 在$I(R, T)$ 的所有团 W上的规模仅为二次的（关于 $|R|$）。

证明. 任意选取 $x_0 \in V(T)$，并设 $T_0 := T[x_0]$。设 W^+、W^-和 $W^\vee$分别为 W 在 T_0中的发散请求、汇聚请求和单峰请求的集合。由推论2，存在 T_0 的一个分支 Q^+，使得W^+ 中每个请求的发射弧都包含于 Q^+，并且存在 T_0 的一个分支 Q^-，使得 W^+中每个请求的接收弧都包含于 Q^-。因此，当 $W^- = \varnothing$时，取$x_W = x_0$ 以及任意包含 Q^+的枝 Q_W，结论成立；当 $W^+ = \varnothing$ 时，取 $x_W = x_0$以及任意包含 Q^-的枝 Q_W，结论亦成立。

现在可以假设 W^+ 和 W^- 都非空。令 x_W 为 $Q^- \cap Q^+$ 的终极顶点，也就是满足 $T\langle x_0, x_W\rangle = Q^- \cap Q^+$ 的顶点 x_W。令 y^-（分别为 y^-）为 Q^-（分别为 Q^+）的叶子，并令 $Q_W = T\langle y^-, y^+\rangle$。

如果所有发散请求的发射弧都在 $Q^+ \cap Q^-$ 中，我们可以取任意包含 Q^- 的枝作为 $x_W = x_0$ 和 Q_W ；结论成立，因为 W^+ 和 W^- 中的所有请求都与 Q_W 相交，而其余请求都在 $W^\vee$ 中，因此是单峰的。同理，如果所有汇聚请求的接收弧都在 $Q^- \cap Q^+$ 中，结论也成立。此后我们可以假设，存在一个发散请求 r^+，其发射弧位于 $T\langle x_W, y^+\rangle$ 中；并存在一个汇聚请求 r^-，其接收弧位于 $T\langle x_W, y^-\rangle$ 中。

设 r 是 W 中与 Q_W弧不相交的一个请求。

假设 r 在 T_0中是汇聚的（相应地，发散的）。则其接收（相应地，发射）弧必须位于 $Q^+ \cap Q^-$ 中，否则它不会与 Q_W弧不相交。此外， r 在 $T[x_W]$中不能是发散的（相应地，汇聚的），否则它不会干扰 r^+（相应地， r^-），从而不属于 W。因此 r在 $T[x_W]$中是单峰的。

现假设 r 在 T_0中是单峰的。为得出矛盾，假设 r 在 $T[x_W]$中是汇聚的。则 r的末端弧必须属于 T（x_0, x_W）。因此 r不干扰 r^-，矛盾。类似地，若 r 在$T[x_W]$中是发散的，我们也会得到矛盾。因此 r在 $T[x_W]$中是单峰的。

我们已证明， W中所有与 Q_W弧不相交的请求在 $T[x_W]$中都是单峰的 .

每个枝都完全由其两个端点确定；在$\mathcal{I}(R,T)$中，可能的 Q_W至多与叶子对的数量相同。此外，如果 T始终以同一个顶点 x_0为根，那么给定 Q_W 后， x_W 只有唯一的选取方式（它是 Q_W中距离 x_0最近的顶点），因此数对（Q_W, x_W）的数量仅为二次级（关于 $|R|$）。 □

现在我们可以证明定理8了。

定理8的证明。任意选取 $x_0 \in V(T)$，并令 $T_0 := T[x_0]$。

对于每个叶子 $y_i \in V(T_0)$，令 Q_i 为从 x_0 到 y_i的分支。对于每个数对（i， j），若 $i \neq j$，则令 $Q_{ij} := T_{y_i, y_j}$，并令 $Q_{ii} := Q_i$。令x_{ij}为 T_0中 $V(Q_i)$和 $\cap V(Q_j)$之间最深的顶点。令 R_{ij} 为 R 中与 Q_{ij}相交的请求集，并令 R'_{ij} 为 $T[x_{ij}]$ 中来自 R 且与Q_{ij}弧不相交的单峰请求集。注意，图$\mathrm{I}(R_{ij}, T)$和$\mathrm{I}(R'_{ij}, T)$都是共二部图。因此，根据柯尼希定理，计算其团数归结为在补图中求解最大匹配问题。使用Hopcroft 和 Karp 的算法 [10]，可以在$O(|R|^{5/2})$时间内完成。

设 W是$\mathcal{I}(R,T)$的一个最大团。根据引理10，存在一对$\{i,j\}$使得 $W \subseteq R_{ij} \cup R'_{ij}$。因此有 $\omega(R,T) \leq \omega(R_{ij},T) + \omega(R'_{ij},T)$。另一方面，根据引理9，每一对请求$(r,r')$都 $\in R_{ij} \times R'_{ij}$ 干扰，因此$\mathcal{I}(R_{ij},T)$中的一个团与$\mathcal{I}(R'_{ij},T)$中的一个团的并集构成$\mathcal{I}(R,T)$中的一个团。综上，

$$\omega(R,T) = \max_{i,j} \Big(\omega(R_{ij},T) + \omega(R'_{ij},T)\Big).$$

由于 T是一个可能的约化树，因此 $|V(T)| \leq 2|R| + 2$（见第 2.2 小节），所以 $\mathcal{O}(|R|^2)$ 是 $\{i,j\}$ 的可能选取数，因此可以在 $\mathcal{O}(|R|^{4.5})$ 时间内计算出这个最大值。 □

4 近似干扰图的色数和 χ-有界性

根据先前的结果，我们可以推导出 $\mathcal{I}(R,T)$ 的色数$\chi(R,T)$ 的以下界。

引理 11. 设 R 是双向树 T 上的一个请求集。我们有

$$\chi(R,T) \le \omega(R^+,T) + \omega(R^-,T) + \omega(R^\vee,T) \le 2\,\chi(R,T).$$

证明。由于 $R = R^- \cup R^+ \cup R^\vee$，我们有 $\chi(R,T) \le \chi(R^-,T) + \chi(R^+,T) + \chi(R^\vee,T)$。由推论 4，$\chi(R^-,T) = \omega(R^-,T)$，$\chi(R^+,T) = \omega(R^+,T)$，且 $\chi(R^\vee,T) = \omega(R^\vee,T)$。因此，

$$\chi(R,T) \quad \le \quad \omega(R^+,T) + \omega(R^-,T) + \omega(R^\vee,T). \tag{1}$$

现在，设 c 是请求集 R 在 T 上的一个 $\chi(R,\ T)$-着色。设 k^+、k^- 和 $k^\vee$ 分别为 c 中出现在 R^+、R^- 和 $R^\vee$ 上的颜色数。由引理 5，某个给定颜色不可能同时出现在 R^+、R^- 和 $R^\vee$ 上，因此 $k^+ + k^- + k^\vee \le 2\chi(R,\ T)$。另一方面，$k^- \ge \chi(R^-,\ T) = \omega(R^-,\ T)$，类似地 $k^+ \ge \omega(R^+,\ T)$ 以及 $k^\vee \ge \omega(R^\vee,\ T)$。因此

$$\omega(R^+,T) + \omega(R^-,T) + \omega(R^\vee,T) \quad \le \quad 2\,\chi(R,T). \tag{2}$$

□

推论 12. 对于给定的干扰图 $I(R,T)$的色数，存在一个 $O(|R|^{5/2})$ 时间的2-近似算法。

证明. 为 T 选取一个根 x。用 $\omega(R^-,T)$ 种颜色为 R^- 着色，用另外 $\omega(R^+,T)$ 种颜色为 R^+ 着色，并用 $\omega(R^\vee,T)$ 种颜色为 $R^\vee$ 着色。这可以在 $\mathcal{O}(|R|^{5/2})$ 时间内完成：利用 Hopcroft 和 Karp [10] 的算法为 $\mathcal{I}(R^\vee,T)$（它是一个共二部图）着色，并利用 Hoàng [9] 的算法为 $\mathcal{I}(R^-,T)$ 和 $\mathcal{I}(R^+,T)$（它们是可比图）着色。

于是我们得到 $\mathcal{I}(R,T)$ 的一种着色，所用颜色数为 $\omega(R^+,T) + \omega(R^-,T) + \omega(R^\vee,T) \le 2\chi(R,T)$。 □

显然，$\omega(R^+,T)$、$\omega(R^-,T)$ 和 $\omega(R^\vee,T)$ 中的每一个都至多为 $\omega(R,T)$。因此引理 11 给出 $\chi(R,T) \le 3\,\omega(R,T)$。我们现在证明关于 $\chi(R,T)$ 的一个用 $\omega(R,T)$ 表示的更好上界。

定理 13. 设 R 是双向树 T 上的一个请求集合。那么

$$\chi(R,T) \le 2\,\omega(R,T).$$

良好对 良好对 (R,T) 是树中的一个请求集合，其中每个请求的长度至少为 2。通过细分每对反向弧（即将弧 (u,v) 和 (v,u) 替换为弧 (u,w)、(w,v)、(v,w)、(w,u)，其中 $w = w(\{u,v\})$ 是一个新顶点），即可轻易得到如下结果。

引理 14。设 R 是树 T 中的一个请求集合。存在一个良好对 (R,T) 使得 $I(R,T) = I(R,T)$。

引理 15。设（R, T）是一组良好对。那么存在一个节点 $x \in V$（T），使得

$$\omega(R^+, T[x]) + \omega(R^-, T[x]) \le \omega(R,T).$$

证明。如果一条有向路径与某个请求相交，即包含其发射弧或接收弧。设 $Q = (q_0, q_1, \ldots, q_\ell)$ 是一条有向路径 T，它与 R 中数量最多的请求相交。不失一般性，我们可以假设 q_0 和 q_ℓ 是 T 的叶。注意，与 Q 相交的请求构成一个团，因此 $m \le \omega(R,T)$。令 $T_0 := T[q_0]$。设 $Q' = (q'_{\ell'}, q'_{\ell'-1}, \ldots, q'_0)$ 是 T_0 中一条以 $q'_0 = q_0$ 为终端顶点、且与最多数量 p 的请求相交的有向路径。根据定义，$p \le m$。

令 $i \ge 0$ 为满足 $q'_i = q_i$ 的最大指标。

对于每个 $j \le i$，令 M_j^-（分别地，M_j^+、$M_j^\pm$）表示由$Q[q_0, q_j]$满足但不由 $Q[q_j, q_\ell]$ 满足的请求集（分别地，由 $Q[q_j, q_\ell]$满足但不由 $Q[q_0, q_j]$满足的请求集，以及同时由 $Q[q_0, q_j]$和 $Q[q_j, q_\ell]$满足的请求集）。令 $m_j^- = |M_j^-|$、$m_j^+ = |M_j^+|$和 $m_j^\pm = |M_j^\pm|$。根据定义，$m_j^- + m_j^+ + m_j^\pm = m$。

类似地，对于每个 $j \le i$，令 P_j^-（分别地，P_j^+、$P_j^\pm$）表示由 $Q'[q'_\ell, q_j]$ 满足但不由 $Q'[q_j, q_0]$ 满足的请求集（分别地，由 $Q'[q_j, q_0]$ 满足但不由 $Q'[q'_\ell, q_j]$满足的请求集，以及同时由 $Q'[q'_\ell, q_j]$ 和 $Q'[q_j, q_0]$满足的请求集）。令 $p_j^- = |P_j^-|$、$p_j^+ = |P_j^+|$和 $p_j^\pm = |P_j^\pm|$。我们有 $p_j^- + p_j^+ + p_j^\pm = p$。

注意到 $m_0^+ + p_0^- = m + p$。此外，$m_i^+ + p_i^-$小于与 $Q'[q'_\ell, q'_i] \cup Q[q_i, q_\ell] = T[q'_\ell, q_\ell]$ 相交且构成一个团的请求数。因此，$m_i^+ + p_i^- \le \omega(R, T)$。令 s 为满足 $m_s^+ + p_s^- \le \omega(R, T)$ 的最小整数。

断言 15.1. $m_s^- + p_s^+ \le \omega(R, T)$。

命题证明。若 $s = 0$，则根据定义，$m_0^- + p_0^+ = 0$，结论成立。现在假设 $s > 0$。由于 (R, T) 是一个良好对，每个请求在 T 中的长度至少为 2。因此，一个请求不可能既属于 M_{s-1}^+，又属于 M_s^-。所以 $m_s^- \le m_{s-1}^- + m_{s-1}^\pm = m - m_{s-1}^+$。同理，$p_s^+ \le p - p_{s-1}^-$。因此 $m_s^- + p_s^+ \le m + p - m_{s-1}^+ - p_{s-1}^-$。但是 $p \le m \le \omega(R, T)$，并且根据 s 的定义，$m_{s-1}^+ + p_{s-1}^- \ge \omega(R, T)$。因此 $m_s^- + p_s^+ \le \omega(R, T)$。◊

设 $T_s := T[q_s]$。下面证明

$$\omega(R^-, T_s) + \omega(R^+, T_s) \le \omega(R,T).$$

设 W^- 是 T_s 中构成大小为 $\omega(R^-, T_s)$ 的最大团的汇聚请求集合。由引理 2 (i)，W^- 中每个请求的接收弧都包含在 T_s 的同一个入枝 P^- 中。注意，要么 P^- 与 $Q[q_0, q_s]$ 弧不相交，要么 P^- 与 $Q[q_\ell, q_s]$ 弧不相交。在第一种情况下，P^- 与 $Q[q_s, q_0]$ 的并构成一条有向路径，该路径与 $\omega(R^-, T_s)$ 个请求相交。根据定义，$p_s^- + p_s^+ + p_s^\pm = p$，因此 $\omega(R^-, T_s) \le p_s^-$。在第二种情况下，$P^-$ 与 $Q[q_s, q_\ell]$ 的并构成一条有向路径，该路径与 $\omega(R^-, T_s) + m_s^+ + m^\pm$ 个请求相交。根据定义，$m_s^- + m_s^+ + m^\pm = m$，因此 $\omega(R^-, T_s) \le m_s^-$。两种情况下都有 $\omega(R^-, T_s) \le \max(m_s^-, p_s^-)$。由对称性，同样有 $\omega(R^+, T_s) \le \max(m_s^+, p_s^+)$。因此

$$\omega(R^-, T_s) + \omega(R^+, T_s) \le \max\{m_s^- + m_s^+,\ m_s^- + p_s^+,\ p_s^- + m_s^+,\ p_s^- + p_s^+\} \le \omega(R, T)。\square$$

引理 14、15 和不等式 (1) 直接推出定理 13。

5 在多项式时间内求解干扰 k-可着色性

5.1 干扰3-可着色性

在本小节中，我们展示一种求解干扰 3-可着色问题的多项式时间算法。我们需要一些预备知识。

图 G 的一个 k-列表赋值 L 是一个映射，它为每个顶点 v 指定一个由 k 个非负整数构成的集合，这些整数称为颜色。 G 的一个 L-着色 是一个映射 c： $V \to \mathbb{N}$，使得对每个 $v \in V$都有 $c(v) \in L(v)$。k-列表可着色性是指：给定图G 和一个 k-列表赋值 L，判定是否存在 G的一个 L-着色。

以下结果是众所周知的，可以直接证明，也可以通过归约到2-SAT来证明。

命题 16. 2-列表可着色性可在$O(|V(G)| + |E(G)|)$时间内求解。

引理 17. 设 R是双向树 T上的一个请求集。则要么$I(R,T)$包含一个大小至多为3的支配集，要么它是一个可比图。

证明。 由于我们考虑的是第 2.2 小节所述的约化树， T 的所有叶子都包含在某个请求中。

设 s_1 是一片叶子， r_1 是包含它的一个请求。根据方向对偶性，我们不妨假设 s_1 是 r_1的第一个顶点。

设 A_1^+ 是 $T^+[s_1]$ 中包含在 R的某个请求中的弧的集合。它非空，因为其中包含 r_1的弧。设 $a_2 = t_2^- t_2$ 是 A_1^+ 中距离 s_1 在 T中最远的一条弧。那么， A_1^+ 中不存在头端是 t_2 在$T[s_1]$中的后代的弧。设 r_2 是包含 a_2的一个请求。（可能 $r_2 = r_1$。）

设 T_2 是 T 中由 t_2 在 $T[s_1]$中的后代所诱导的双向子树。设R' 是与 r_1， r_2 中任何请求都不干扰的请求集。注意， R' 的所有请求都包含在 $T[t_2, s_1] \cup T_2^-[t_2]$中。

如果请求 r包含 $T[t_2, s_1]$中的一条弧，那么它会干扰 R'中的所有请求，因此$\{r_1$， r_2，$r\}$是$\mathcal{I}(R,\ T)$的一个支配集。此后，我们可以假设 R' 中的所有请求都包含在 $T_2^-[t_2]$中。

如果请求 r 包含 $T[t_2]$中的一条发散弧，那么根据 t_2的定义，这条弧不在 T_2中。因此， R' 中的每个请求都会在 r 上发生干扰，并且$\{r_1,\ r_2,\ r\}$ 是$\mathcal{I}(R,\ T)$的一个支配集。此后，我们可以假设 $T[t_2]$中的所有请求都是汇聚的。但根据推论 4(i)，$\mathcal{I}(R,\ T)$是一个可比图。

□

注18。 上述证明很容易转化为一个运行时间为 $\mathcal{O}(|R|)$ 的算法，该算法要么找到一个大小至多为 3 的支配集，要么判定 $\mathcal{I}(R,\ T)$ 是一个可比图。

引理 17 可用于在平方时间内求解干扰3-可着色性 e.

推论19。 干扰3-可着色性可在 $O(|R|^2)$ 时间内解决。

证明。设 R 是双向树 T 中的一个请求集，并令 $G = \mathcal{I}(R, T)$。根据注18，我们可以在线性时间内，要么找到 G 中一个大小至多为 3 的支配集 S，要么得出 G 是一个可比图。

在第二种情况下，可以使用 Hoàng [9] 的算法，在 $= \mathcal{O}(|R|^2)$ 时间内计算 $\chi(G)$，其中 $\mathcal{O}(|V(G)|^2)$。

在第一种情况下，我们枚举 S 的 $3^{|S|} \leq 27$ 可能着色，并对每一种着色检查该部分着色是否可以扩展为 G。使用 2-列表可着色性（命题 16），可以在$\mathcal{O}(|V(G)| + |E(G)|)$ $= \mathcal{O}(|R|^2)$ 时间内完成。事实上，$V(G)$ 中每个顶点的可用颜色集合 $\setminus S$ 的大小至多为 2，因为 S 中的某个邻居至少会禁止一种颜色。 □

我们刚刚证明了干扰3-可着色性可以在多项式时间内求解，并希望将这一结果扩展到较大 k 值下的干扰 k-可着色性。不幸的是，上述方法无法用于后一种情形。事实上，我们利用了 2-列表可着色性可以在多项式时间内求解这一事实。但正如我们现在将要看到的，即使在干扰图上，3-列表可着色性也是 NP完全的。

下面的定理显然广为人知，但为完整起见，我们仍给出简短的证明。

定理20。3-列表着色在完全二分图上是*NP*完全的。

证明。从3-SAT归约。

设 $C = (C_1, \ldots, C_m)$ 是一个定义在变量集合 X 上的 3-SAT 实例，其中有 n 个变量 $x_1, \ldots, x_n$。记 $C_j = \alpha_j \vee \beta_j \vee \gamma_j$。设 (U, V) 是 $K_{m,n}$的二部划分，其中 $U = \{u_1, \ldots, u_m\}$ 且 $V = \{v_1, \ldots, v_n\}$。设 L 是如下定义的列表赋值：对所有 $j \in [m]$，$L(u_j) = \{\alpha_j, \beta_j, \gamma_j\}$；对所有 $i \in [n]$，$L(v_i) = \{x_i, x_i\}$。

—

如果 C 是可满足的，设 $\phi\colon X \to \{\text{true}, \text{false}\}$ 是一个满足 C 的真值赋值。我们令ψ 为一个 L-着色，使得 $\psi(u_j)$ 是 C_j 中任意一个为真的文字，而 $\psi(v_i)$ 是 x_i 中为假的文字。于是 ψ 是 G 的一个合法 L- 着色。

逆向图显而易见。

若要使 L 的所有列表长度均为3，则可以在每个列表 $L(v_i)$ 中添加一个额外的通用颜色 δ，在 U 中添加一个包含6个顶点的额外集合 U_0，其列表为$\{\delta\} \cup S$（对每个 $S \in \binom{[4]}{2}$），并在 V中添加一个包含3个顶点的额外集合 V_0，其列表包括$\binom{[4]}{3}$中的所有可能列表。注意，在对 $G\langle V_0\rangle$进行任何合法的L-着色时，至少必须出现两个不同的颜色 $x, y \in [4]$，这迫使 U_0 中列表为$\{x, y, \delta\}$ 的顶点着成 δ。因此，颜色 δ 不能出现在 V 的任何列表中，于是我们又回到了上文所述的情形。 □

接下来我们将看到，存在一个请求集，其干扰图是完全二分图 $K_{m,n}$（对每个 $m, n \geq 1$）。因此，由定理20可推出以下结论。

推论21. 3-列表着色在干扰图类 $\mathcal{I}(R, T)$上是 *NP*完全的，即使树 T是星形图且 R 的所有请求长度均为1。

证明。设$T = K_{1,\ m+n}$ 以其唯一内部节点 x_0为根。设(L_1, L_2)为 T 的叶的一个划分，且 $L_1 = m$ 与 $L_2 = n$。设 R 由 m组成

对每个 $y \in L_1$形如 $y \to x_0$ 的汇聚请求，以及对每个 $y \in L_2$的 n 个形如 $x_0 \to y$的发散请求。则$\mathcal{I}(R,\ T)$是完全二分图$K_{m,\ n}$，其二部划分为$(L_1,\ L_2)$。

□

5.2 干扰 k-可着色性当 $k \geq 4$

定理 22. 干扰 k-可着色性可在时间 $O(k36^k|R|$ 内求解 $^3)$.

证明。设 R是双向树 T上的请求集，该树以 z为根。令 $G = \mathcal{I}(R,T)$。

设 ϕ 是 G的一个正常 k-着色。根据引理 6，每种颜色 $i \in [k]$ 可分为以下三种类型:

(i) 若来自 $R^+ \cup R^\vee$ 且被着为 i的请求至多有一个，则称 i 为主要汇聚;

(ii) 若来自 $R^- \cup R^\vee$ 且被着为 i的请求至多有一个，则称 i 为主要发散;

(iii) i 是单峰 的，当且仅当其颜色类恰由来自 $R^\vee$ 的两个请求组成。

如果一个非发散（分别地，非汇聚）请求被赋予主要发散（分别地，汇聚）颜色，我们称其为例外 请求。注意，某个颜色类可能既主要汇聚又主要发散，此时它的形式为$\{r, r'\}$，其中 r 是汇聚请求， r' 是发散请求；如果t_r 比根 z 更近于 $s_{r'}$，那么 r 是该颜色类中的例外请求，否则 r' 是。

我们首先证明，可以假设 ϕ 中的例外请求能够由两个团覆盖。

断言 22.1. 存在 G 的一个正常 k-着色，使得所有非发散例外请求构成一个团，所有非汇聚例外请求也构成一个团。

命题证明。考虑一个使例外请求数量最少的正常着色 ϕ。反设存在两个独立的非发散例外请求 r, r'。令 $\gamma = \phi(r)$ 和 $\gamma' = \phi(r')$。注意，根据例外请求的定义， $\gamma \neq \gamma'$。着色为 γ 的发散请求都包含在 $T^+_{s_r} \cup T[m_r, s_r]$中（回想一下，如果 r 是汇聚请求，则 $m_r = t_r$），其中 $T^+_{s_r}$ 是以 s_r 为根的 T^+ 子树；特别地，对于每个着色为伽马的发散请求 r_0，$s^+_{r_0}$ 是T 中 m^-_r 的后代。同样，对于每个着色为 γ'的发散请求 r'_0， $s^+_{r'_0}$ 是 T 中 $m^-_{r'}$ 的后代。

由于 r 和 r' 不相邻，因此 m^-_r 和 $m^-_{r'}$ 在 T中不可比；特别地，对于 m^-_r的每个后代 x 以及 $m^-_{r'}$ 的每个后代 x'， x 和 x' 也不可比。因此，着色为 γ 或 γ' 的发散请求共同构成一个独立集。由此，我们可以将它们全部重新着色为颜色 γ，并将 r（以及 r'）重新着色为颜色 γ'。这样，我们就得到了 G的一个适当的 k-着色，其中例外请求比 ϕ少两个，这与假设矛盾。

根据方向对偶，我们可以得到关于非汇聚例外请求的结果。 ♢

设 ϕ 是 G 的一个适当的 k-着色，其中非汇聚例外请求的集合 Q^+ 是一个团，非发散例外请求的集合 Q^- 也如此，并设

$Q := Q^+ \cup Q^-$ 是 ϕ中的例外请求集合。我们用 ϕ_0 表示由 ϕ 在子图 $G\langle Q\rangle$上诱导的部分着色 $\phi_{|Q}$ 。设 $\mathscr{C}_0$ 为 G 的所有适当的 k-着色 ϕ' 构成的集合，使得 $\phi'_{|Q} = \phi_0$，Q^+ 分别是 ϕ'的非汇聚例外请求集合和非发散例外请求集合；注意，该集合非空，因为 $\phi \in \mathscr{C}_0$。我们将证明，若给定 ϕ_0，则可以在多项式时间内求得 G 的一个适当的 k-着色。

设 $Q^- = q_1$、.....、q_t为 ϕ的非发散例外请求，按照其源节点的后序深度优先搜索（DFS）顺序排列（因此，若 s_{q_j} 是 T中 s_{q_i} 的祖先，则必须有 $i \le j$）。由于可以置换颜色，不妨假设对每个 $i \in [t]$，均有$\phi_0(q_i) = i$。设 $D = r_1$、.....、r_n为发散的非例外请求，按照其发射弧的前序 DFS 顺序排列（因此，若 $s^+_{r_i}$ 是 T中 $s^+_{r_j}$ 的祖先，则必须有 $i \le j$）。令 ϕ_1 为从 ϕ_0 得到的着色：按此顺序逐个为每个 $r_i \in D$贪心地扩展着色至 D，并为其使用最小的可用颜色。

命题 22.2。存在 G 的一个适当的 k-着色 $\phi' \in \mathscr{C}_0$，使得对于每个请求 $r \in D$，都有 $\phi_1(r) \in [t] \implies \phi'(r) = \phi_1(r)$。

命题证明。给定一个着色 $\phi' \in \mathscr{C}_0$，我们称请求 $r \in D$ 在ϕ' 中是坏的，如果 $\phi_1(r) \in [t]$ 且 $\phi'(r) \ne \phi_1(r)$。为了得到矛盾，我们假设 $\mathscr{C}_0$ 中的每个着色都至少包含一个坏请求，并令 $\phi' \in \mathscr{C}_0$ 是使坏请求的最小下标 i_0 最大化的那个着色。因此，设 $D_0 := \{r_i \in D : i < i_0\}$，则 D_0 中没有请求是坏的。这意味着，对于每种颜色 $\gamma \in [t]$，$\phi_1^{-1}(\gamma) \cap D_0 = \phi'^{-1}(\gamma) \cap D_0$。我们记 $r := r_{i_0}$，$\alpha := \phi_1(r) \in [t]$，以及 $\beta := \phi'(r)$。由于 $\alpha \in [t]$，存在一个例外请求 $q \in Q^-$，满足 $\phi_1(q) = \phi_0(q) = \alpha$。由构造，$\alpha = \min\Big([k] - \phi_0(N(r) \cap Q) - \phi_1(N(r) \cap D_0)\Big)$。令 ϕ'' 为 ϕ' 的一个副本，我们在其中执行以下重新着色步骤:

- $\phi''(r) \leftarrow \alpha$;
- 对于每个 $r\prime \in D$，如果 $\phi\prime(r\prime) = \alpha$，且 $s^+_{r\prime}$ 是 s^+_r 在 T 中的后代，我们执行 $\phi\prime\prime(r\prime) \leftarrow \beta$。

我们断言，ϕ'' 是 G 的一个正常 k-着色，其中没有满足 $i \le i_0$ 的请求 r_i 是不良的，这与 ϕ'的选取相矛盾。为此，我们先证明 $\alpha \notin \phi''(N(r))$。事实上，$\alpha$ 在 ϕ'中是主要发散的，因此除 q 之外（它不在 $N(r)$中，因为 $\phi_1(r) = \phi_1(q) = \alpha$），$\phi'$ 中所有着色为 α 的请求都在 D中。我们有 $\alpha \notin \phi_1(N(r) \cap D_0)$，并且由于 D_0 中没有请求是不良的，特别有 $\alpha \notin \phi'(N(r) \cap D_0)$。由此可得$N(r) \cap \phi'^{-1}(\alpha) \subseteq \{r_i \in D : i > i_0\}$。因此，$\phi'$ 中每个着色为 α、且与 r 相邻的邻居 r'，都满足 $s^+_{r'}$ 是 T中 s^+_r的后代，这意味着 $\phi''(r') = \beta$。其余所有请求在 ϕ' 和 ϕ''中的颜色相同，因此我们得出 $\alpha \notin \phi''(N(r))$。接下来，若存在$r' \in D$ 使得 $\phi'(r') = \alpha$，且 $s^+_{r'}$ 是 T中 s^+_r的后代，我们证明 r' 中唯一着色为 β、且与 ϕ'相邻的是 r。注意 $\beta > \alpha$；事实上，要么 $\beta \notin [t]$，这显然成立；要么 $\phi'^{-1}(\beta) \cap D_0 = \phi_1^{-1}(\beta) \cap D_0$，于是 $\beta \notin \phi_1(N(r) \cap D_0)$，因此根据构造，$\phi_1(r) \le \beta$。由于 r 不是例外的，着色为 β的非发散请求至多有一个，即 q'，此时 $q' \in Q^-$。由于 $\beta > \alpha$，我们可推知$s_{q'}$ 不是

T中 s_q 的后代，并且由于 $\phi'(r) = \phi'(q')$，我们可推知 r 和 q' 不会相互干扰，因此$s_{q'}$ 是 T中 t_r 的祖先；由此可推知（因为 r 和 q 不会相互干扰），$s_{q'}$ 是 T中 s_q 的祖先。特别地，q' 和 r' 不会相互干扰。所有其他着色为 β 的请求r'' 都是发散的，且不会干扰 r，因此 $s^+_{r''}$ 和 s^+_r在T中彼此无关，这意味着 $s_{r''}$和 $s_{r'}$在 T中也彼此无关。由此可知，ϕ''是正常的，正如所需。为结束证明，还需证明 ϕ''中没有满足 $i \le i_0$ 的请求 r_i 是不良的。 ϕ'中没有来自 D_0 的请求是不良的，并且 $\alpha \notin \phi_1(N(r) \cap D_0)$，所以 ϕ''中没有来自 D_0的请求被重新着色，即 $\phi'_{|D_0} = \phi''_{|D_0}$。因此，$\phi''$中没有来自 D_0 的请求是不良的；$r = r_{i_0}$也不是不良的，证明至此结束。◊

根据方向对偶，对于汇聚请求也有类似的结论。

算法首先进行以下预处理，这可以在$\mathcal{O}(|R|^2)$时间内轻松完成。它首先检查一些显然会导致无法进行 k着色的条件：首先，检查是否存在一条有向分支与超过 k 个请求相交，从而这些请求构成一个大小大于 k的团；其次，检查是否 $|R^\vee| > 2k$，此时$\mathcal{I}(R^\vee, T)$包含一个大小大于 k的团，因为根据推论 4 (iii)，它是一个共二部图。若答案为是，则图 G 无法进行 k着色，算法返回“否”。否则，即答案为否时，没有有向分支与超过 k 个请求相交，并且 $|R^\vee| \le 2k$。随后，算法根据节点 s^+_r，按照前序深度优先搜索排序对发散请求 $r \in R^+$ 进行排序，并为每个请求记录在该顺序中位于 r之前的发散请求 $r' \in N(r) \cap R^+$列表，并将其存入绑定祖先[r]（该列表的大小小于 k）。对于汇聚请求集合 R^-，算法也以类似方式处理。

接下来，算法搜索命题 22.2 所保证的 k对 G 的着色。它按如下方式枚举 (Q^-, Q^+) 的所有可能选择。首先考虑所有单峰请求，并逐一决定将其归入 Q^-、Q^+，还是两者都不归入。由于 $|R^\vee| \le 2k$，选择至多有 3^{2k} 种。对于每种选择，首先检查 $GR^\vee \setminus (Q^- \cup Q^+)$ 的补图是否存在完美匹配 M；使用 Hopcroft 和 Karp 的算法 [10] 可在 $O(k^{2.5})$ 时间内完成此检查。若不存在，则推断这一特定的 $(Q^- \cap R^\vee, Q^+ \cap R^\vee)$ 选择无效，并考虑下一种选择。否则，Q^- 的汇聚请求至多有 $2^k|R|$种可能性（分支至多有 $|R|$种选择，且该分支上的汇聚请求子集有 2^k 种选择），而 Q^+ 的发散请求至多有 $2^k|R|$种可能性（由对称性可得）。对于 (Q^-, Q^+) 的每种可能选择，按照源节点对 $Q^- = q_1$、……、q_t 进行后序深度优先搜索排序，并按照接收节点对 $Q^+ = q_{t+1}$、……、$q_{t+t'}$ 进行后序深度优先搜索排序。然后，对每个 $i \in [t + t']$ 固定 $\phi_0(q_i) := i$，并为每一对匹配的单峰非例外请求分配一个不同的颜色，颜色取自 $\{t + t' + 1, \ldots, t + t' + |M|\}$。根据命题 22.2 及其方向对偶，我们可以通过贪心方式将 ϕ_0 扩展为 ϕ_1：对非例外发散请求集合 $R^+ \setminus Q^+$，仅使用 $[t]$ 中的颜色（如果某个请求 $r \in R^+ \setminus Q^+$ 没有可用的 $[t]$ 中的颜色，则将其留为未着色）；对非例外汇聚请求集合 R^-，仅使用 $\{t + 1, \ldots t + t'\}$ 中的颜色（必要时同样将其中一些请求留为未着色）。R^+ 中所有其余未着色请求，按照 R^+ 上预处理后的顺序，使用一组新的 k^+ 种颜色贪心着色。这样便能在

时间$\mathcal{O}(k|R|)$ 内返回这些请求的最优着色（对于每个请求 r，只需在$\mathcal{O}(k)$ 时间内计算分配给其祖先[r] 的颜色集合，即可确定分配给 r 的颜色）。类似地， R^- 中其余未着色的请求使用一组新的 k^- 种颜色贪心着色。所得着色使用 $t+t'+|M|+k^+ +k^-$ 种颜色。若此数量小于或等于 k，算法便停止并返回该着色；否则，继续考虑 (Q^-, Q^+) 的下一种可能选择。

由于该算法执行了$\mathcal{O}(36^k|R|^2)$次扩展测试，每次耗时$\mathcal{O}(k|R|)$，因此算法的复杂性为$\mathcal{O}(k36^k|R|^3)$。 □

6 结论与进一步研究

在定理8中，我们证明了一个$\mathcal{O}(|R|^{4.5})$时间算法，可以求出干扰图的团数。很可能还能找到更快的算法。

问题23。 设计一种算法，以少于$\mathcal{O}(|R|^{4.5})$的时间计算给定的、具有 $|R|$ 个顶点的干扰图的团数。

关于干扰图的色数，我们针对 $\chi(R,T)$给出了一个$\mathcal{O}(|R|^{5/2})$时间的2-近似算法（推论 12）。接着，我们证明了：判定 $\chi(R,T) \leq k$是否成立的干扰可着色性问题，在以 k为参数时是固定参数可解的。为此，我们提出了一种算法，当 $k \geq 4$时，可分别在$\mathcal{O}(|R|^2)$时间内求解干扰3-着色问题，并在$\mathcal{O}(k36^k|R|^3)$时间内求解干扰 k-着色问题。然而，干扰可着色性的复杂性仍然未知。

问题24. 干扰可着色性是否可以在多项式时间内求解？它是否是 NP 完全问题？

另一个问题是，能否在多项式时间内对其进行良好的近似。在推论 12 中，我们证明它可以达到 2-近似，但也可能存在更好的近似。

问题25. 针对某个 $r < 2$，求一个多项式时间的 r-近似算法，或者证明对于某个 $r \geq 1$，这样的算法不存在（在标准假设下）。

我们的 2-近似利用了这样一个事实：干扰图类存在一个 χ-界函数，也就是说，存在一个函数 f，使得对于每个干扰图 G， $\chi(G) \leq f(\omega(G))$。定理 13 表明， $f\colon t \mapsto 2t$ 是干扰图的一个 χ-界函数。

问题26. 干扰图的最小 χ-界函数是什么？也就是说，对于双向树 T 中的每个请求集合 R，满足 $\chi(R,T) \leq f^*(\omega(R,T))$ 的最小函数 f^* 是什么？

我们注意到 $f^*(2t) \geq \lceil 5t/2 \rceil$。事实上，考虑一个顶点集为 $\{a,b,c,d\}$、边集为 $\{ab, ba, bc, cb, bd, db\}$ 的双向树。令 R_1（分别为 R_2、 R_3、 R_4、 R_5）为一组t 请求，等于 (a,b)（分别为 (b,c)、(d,b)、(b,a)、(c,b,d)），并令 $R = R_1 \cup R_2 \cup R_3 \cup$

$R_4 \cup R_5$。容易验证，$\mathcal{I}(R,\ T)$ 是 $C_5[K_t]$，即 5-圈 C_5 与完全图 K_t 的字典序积。因此 $\omega(R,\ T) = 2t$，且 $\chi(R,\ T) = \lceil 5t/2 \rceil$。

寻找图的正常着色的一种自然方法是依次选择一个最大独立集。对于干扰图而言，这是一个多项式时间算法，因为根据定理 7，可以在多项式时间内计算最大独立集。

问题 27。 在由未着色顶点诱导的子图中依次寻找最大独立集所构成的算法，对于干扰图是否具有近似保证？也就是说，是否存在一个常数 C，使得对于每个干扰图 G，该算法生成的着色所使用的颜色数至多为 $C \cdot \chi(G)$?

请注意，图 1 所示的双向树 T 中的请求集合 R 表明，这样的常数 C 至少为 $3/2$。

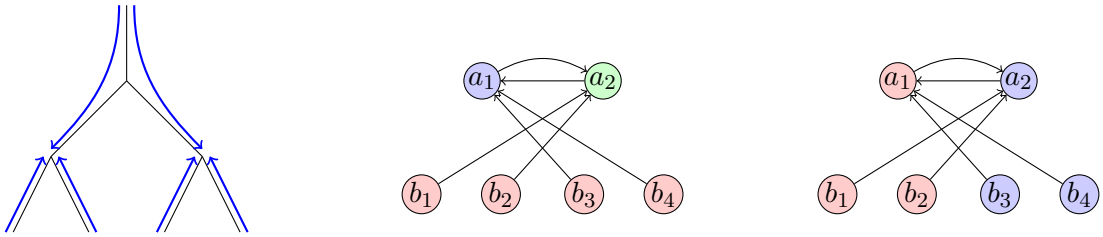

图 1：双向树 T 中以蓝色表示的请求集合 R 和以黑色表示的树（左）；由算法着色的干扰有向图 $\mathcal{I}(R,\ T)$（中）以及最优着色（右）。

参考文献

[1] Omran Ayoub、Oleg Karandin、Memedhe Ibrahimi、Andrea Castoldi、Francesco Musumeci 和 MassimoTornatore。《无滤波光网络》教程（特邀）。*IEEE/Optica*《光通信与网络期刊》，14(3)：1–15，2022 年 3 月。

[2]Bruno Beauquier、Jean-Claude Bermond、Luisa Gargano、Pavol Hell、Stephane Perennes 和 Ugo Vaccaro。全光网络中由波长路由产生的图问题。载于计算机科学光学研讨会（*WOCS*）论文集，1997 年。[3] Imrich Chlamtac、Aura Ganz 和 Gadi Karmi。光路通信：一种实现高带宽光学广域网的方法。*IEEE*《通信汇刊》，40(7)：1171–1182，1992 年。[4] Mark De Berg、Otfried Cheong、Marc Van Kreveld 和 Mark Overmars。计算几何：算法与应用。施普林格，2008 年。

[5] Thomas Erlebach 和 Klaus Jansen。路径着色与呼叫调度的复杂性。理论计算机科学，255(1)：33–50，2001 年。[6] Ehsan Etezadi、Carlos Natalino、Christine Tremblay、Lena Wosinska 和 Marija Furdek。可编程无滤波光网络：架构、设计与资源分配。*IEEE/ACM*网络汇刊，第 1–14 页，2024 年。[7] Martin Charles Golumbic。第 5 章——可比图。载于算法图论与完美图，第 105–148 页。学术出版社，1980 年。[8]Martin Charles Golumbic 和 Robert Edward Jamison。树中路径的边交图。组合理论杂志*B*辑，38(1)：8–22，1985 年。[9] Chính T. Hoàng。若干类完美图的最小加权着色高效算法。离散应用数学，55(2)：133–143，1994 年。[10] John E. Hopcroft 和 Richard M. Karp。二分图最大匹配算法。*SIAM*计算杂志，2(4)：225–231，1973 年。[11] Brigitte Jaumard 和 Yan Wang。无滤波光网络最优设计的双子问题分解，2021 年。arXiv:2101.01022。[12] Brigitte Jaumard、Yan Wang 和 David Coudert。用于无滤波光网络设计的丹齐格-沃尔夫分解。*IEEE/OSA*《光通信与网络期刊》，13(12)：312–321，2021 年。[13] Brigitte Jaumard、Yan Wang 和 Nicolas Huin。无滤波光网络的最优设计。国际透明光网络会议（*ICTON*），第 1–5 页，2018 年。[14] Oleg Karandin、Omran Ayoub、Memedhe Ibrahimi、Andrea Castoldi、Rosanna Pastorelli、Francesco Musumeci 和 Massimo Tornatore。面向城域无滤波光网络的低成本设备设计。*IEEE/Optica*《光通信与网络期刊》，17(3)：198–208，2025 年 3 月。[15] Qian Lv 和 Zuqing Zhu。采用 OTN 加密的无滤波光网络多层规划。*IEEE/ACM*网络汇刊，31(6)：2529–2544，2023 年。[16] Jean-Philippe Savoie、Christine Tremblay、David V. Plant 和 Michel P. Bélanger。无滤波光网络的物理层验证。载于欧洲光通信会议（*ECOC*），第 1–3 页，意大利都灵，2010 年 9 月。[17] Robert Endre Tarjan。基于团分隔子的分解。离散数学，55(2)：221–232，1985 年。

[18] Christine Tremblay、Emile Archambault、Michel P. Bélanger、Jean-Philippe Savoie、François Gagnon与David V. Plant。基于先进调制和电补偿技术的无滤波无源核心网络。《电信系统》，52(4):167–181，2013。[19] Christine Tremblay、François Gagnon、Benoit Châtelain、Eric Bernier与Michel P. Bélanger。无滤波光网络：一种独特而新颖的无源广域网解决方案。载于第*12*届光电子与通信会议暨第*16*届国际集成光学与光纤通信会议（*OECC–IOOC*），第466–467页，日本神奈川，2007。[20] Christine Tremblay、Paul Littlewood、Michel P. Bélanger、Lena Wosinska与Jiajia Chen。灵活无滤波光网络。载于 国际光网络设计与建模会议（*ONDM*），第1–4页，2017。[21] Alan Tucker。圆弧族着色。《*SIAM* 应用数学杂志》，29(3):493–502，1975。[22] Anna Tzanakaki与Mike J. O’Mahony。采用半导体光放大器中交叉增益调制的无滤波波长转换器分析。载于 激光与电光会议（*CLEO*），第433–434页，美国马里兰州巴尔的摩，1999。