arXiv:2301.10503v1 [cs.DS] 25 Jan 2023

在哪些图结构中可以高效地寻找时间不相交路径和游走?

Pascal Kunz*、Hendrik Molter† 和 Meirav Zehavi

柏林洪堡大学，算法工程，德国柏林；柏林工业大学，算法学与计算复杂性，德国柏林；
p.kunz.1@tu-berlin.de

内盖夫本-古里安大学计算机科学系，以色列贝尔谢巴；molterh@post.bgu.ac.il,
meiravze@bgu.ac.il

摘要

时序图的边集可能随离散时间步而变化，而时序路径（或游走）必须沿着在递增时间步出现的边行进。因此，如果两条时序路径（或游走）不会在同一时间访问任何顶点，那么它们就是时间不相交的。Klobas 等人近期启动了对时序图中寻找时间不相交路径或游走的计算复杂性研究。 [IJCAI '21]这项研究受到多智能体路径规划（MAPF）应用的启发，包括机器人学、仓库管理、航空器管理和交通路径规划。

我们扩展了 Klobas 等人的研究，针对限制极为严格的情形给出了参数化困难性结果，重点关注所谓底层图的结构参数。另一方面，我们也找到了足够简单、能够高效求解该问题的情形。我们的结果揭示了该问题的"路径版本"与"游走版本"（其中顶点可以被多次访问）之间一些出人意料的差异，并回答了 Klobas 等人提出的若干开放问题。

1 引言

判断图中的一组顶点对（称为源汇对）能否通过两两顶点不相交的路径连接起来，是算法图论中一个具有根本意义的问题。它是最早被证明为 NP完全的问题之一 [15] ，而对该问题的进一步研究与近代离散数学中最具开创性的成就之一——图小理论 [24, 25]——密切相关。已知当需要连接的顶点对数量为常数时，不相交路径问题可在二次时间内求解；也就是说，对于所求路径的数量而言，该问题是固定参数可解的 [16]。对于有向图，寻找两条不相交路径已经是 NP难问题 [10]，；但在有向无环图中，如果路径数量为常数，该问题便可在多项式时间内求解 [28]。

*由德国研究联合会研究生培训组2434"复杂性的侧面"资助。
†由以色列科学基金会（资助编号1456/18）和欧洲研究委员会（资助编号949707）资助。

Klobas 等人 [18]最近提出并研究了不相交路径问题的两个自然的时序版本，分别称为 Temporally Disjoint Paths 和 Temporally Disjoint Walks。简单来说，时序图的边集可能会随离散时间步而变化。因此，时序路径必须沿着出现时间递增的边前进，且每个顶点至多访问一次；而时序游走可以多次访问同一顶点。此外，如果两条时序路径（或游走）在同一时间没有占据任何相同顶点，则称它们时间不相交。与非时序情形类似，我们的目标是从给定的多重集中，为各顶点对寻找时序路径（或游走），使这些路径（或游走）两两时间不相交。我们将在第2节给出形式化定义。

由于时序图中的连通性具有非对称性和非传递性，与路径搜索相关的问题在时序情形下的表现与静态情形大不相同。事实上，许多自然的时序路径搜索问题在静态情形下都没有直接对应的问题 [5, 11]。对于 Temporally Disjoint Paths和 TemporallyDisjointWalks，情况也类似。Klobas 等人 [18], 的研究还得出了其他结果，例如，他们证明，即使底层图[1] 是一条路径，这两个问题仍然都是 NP难问题；而在这种情况下，静态不相交路径问题是平凡的。此外，他们揭示了 Temporally Disjoint Paths和 Temporally Disjoint Walks在计算复杂性方面一些颇令人意外的差异。我们以 Klobas 等人 [18] 的研究为基础，继续系统研究 Temporally Disjoint Paths和 Temporally Disjoint Walks的（参数化）计算复杂性。我们给出了若干新的困难性结果和算法结果，揭示了这两个问题变体之间更多有趣的差异，并解决了 Klobas 等人 [18]提出的一些开放问题。

时间不相交路径问题的主要应用领域之一是多智能体路径规划（MAPF）。近年来，这一领域吸引了人工智能和机器人学界的大量研究 [26, 30, 31]。其目标是为多个智能体寻找路径，使所有智能体都能同时沿各自的路径移动而不发生碰撞。经典不相交路径问题与多智能体路径规划问题的基本设定之间，主要区别在于：在后者中，我们假设智能体沿路径每次移动一步，且只有当它们在同一时间移动到同一个顶点时才会发生碰撞。这意味着，多智能体路径规划问题的解中，各条路径不一定是顶点不相交的；但如果两条路径有共同顶点，该顶点在两条路径中的序数位置不能相同。经典多智能体路径规划设定与本文所研究问题之间的一个关键区别，在于我们假设智能体移动期间网络或图结构可能随时间变化。多智能体路径规划在现实世界中的应用包括自动驾驶车辆、机器人学、自动化仓库和机场拖曳 [26, 30, 31],；在许多现实场景中，网络拓扑随时间发生可预测的变化有充分的实际依据 [13, 20, 23]。

相关工作。非时序的不相交路径问题是算法图论中最核心的问题之一，因此在过去几十年间得到了学术界的广泛研究。概述可参见 Korte 等人 [19]。

近年来，人工智能和机器人学领域也对多种变体的（非时序）多智能体路径规划问题（MAPF）进行了大量研究 [26, 30, 31]。因此，MAPF 可视为图上所谓棋子移动问题的推广，且已知其为 NP难问题 [12, 34]。据我们所知，目前最先进的（最优）

[1]时序图的 *underlying graph* 是指包含时序图中至少出现过一次的所有边的静态图。

算法 用于 MAPF 问题的方法采用所谓的基于冲突的搜索方法 [27].

在多智能体路径规划领域，人们研究了各种不同的设置和问题变体，包括协作设置 [29], 鲁棒性要求 [3] 或存在延迟的情况 [22], 在线设置 [32], 连续时间设置 [2], 可解释性要求 [1], 涉及大型智能体（可能占据多个顶点）的设置 [21], 以及其他许多情况。更全面的概述请参阅 Stern [30] 以及 Stern 等人 [31]。

Klobas 等人 [18] 开创了对 Temporally Disjoint Paths 和 Temporally Disjoint Walks的研究；这类问题可以理解为多智能体路径规划设置，其中图中边的可用性可能会在智能体移动期间发生变化。按照 Stern 等人[31], 使用的多智能体路径规划术语，该问题设置同时考虑智能体之间所谓的“顶点冲突”和“边冲突”，也就是说，两个智能体不能在同一时间占据同一个顶点，也不能在同一时间经过同一条边。此外，该问题采用“到达目标后消失”的假设，即智能体到达目标顶点后，其他智能体可以再次占据该顶点。我们指出，Klobas 等人 [18] 考虑的是所谓的非严格时间路径，这类路径经过的边出现时间步非递减。而在本文中，我们考虑所谓的严格时间路径，如前所述，这类路径必须按出现时间步严格递增的顺序经过边。后者模拟了多智能体路径规划中的常见假设，即智能体每个时间步至多沿一条边移动 [30]。仔细审视 Klobas 等人 [18] 的证明后可以发现，下面提到的结果也适用于严格情形。 Temporally Disjoint Paths即使只有两个源汇对，也是 NP难的；而 Temporally Disjoint Walks对于源汇对的数量而言是 W[1] 难的，但当源汇对的数量为常数时，可以在多项式时间内求解 [18]。此外，即使底层图是一条路径，这两个问题变体也都是 NP难的 [18]。尽管如此，如果底层图是森林，
Temporally Disjoint Paths 关于源汇对数量是固定参数可解的 [18]。

最后，我们指出，Kempe 等人 [17]研究了时序图中不相交路径的另一种定义。他们认为，如果两条时序路径没有经过同一个顶点，那么它们就是不相交的，即使两条时序路径并非在同一时刻占据该顶点。他们证明，寻找这样两条路径是 NP难的。

我们的贡献。我们的工作旨在进一步了解可以利用底层图的哪些结构，高效地解决
Temporally Disjoint Paths 和 TemporallyDisjoint Walks （就参数化计算复杂性而言）。我们的首个主要计算复杂性结果表明，即使在图的顶点数较少且底层图为星形图（一个中心顶点与若干叶节点相连）的极其受限情形下，我们也很可能无法高效地解决
Temporally DisjointPaths 和 Temporally Disjoint Walks。

- Temporally Disjoint Paths 和 Temporally Disjoint Walks关于顶点数是 NP难的且为 W[1]-难的，即使底层图是星形图也是如此。

请注意，我们有一个源汇对的多重集，也就是说，输入中的源汇对数量可能远多于顶点数量。因此，我们着重研究将源汇对数量作为（部分）参数的情形。如前所述，Klobas 等人 [18] 证明，当底层图为森林时， Temporally Disjoint Paths关于源汇对数量是固定参数可解的。他们留下了一个开放问题：能否将这一算法推广为固定参数算法，使参数为源汇对数量与底层图的某种距森林距离度量

之和。他们还留下了另一个开放问题：能否为 Temporally Disjoint Walks 找到类似的算法。我们解决了这两个开放问题。

对于 Temporally Disjoint Paths ，我们证明，即使将源汇对的数量与底层图的点覆盖数结合起来，想必也无法得到固定参数算法。

- Temporally Disjoint Paths 关于源汇对的数量与底层图的点覆盖数的组合是 W[1]-难的。

这一结果排除了若干常见的"到森林的距离"度量作为潜在参数，例如树宽或反馈顶点数；这些度量都小于点覆盖数。从积极的一面来看，我们可以证明，将反馈边数（这是一种与点覆盖数不可比较的"到森林的距离"度量）作为额外参数，可以实现可处理性。

- Temporally Disjoint Paths关于源汇对的数量与底层图的反馈边数组合是固定参数可解的。

Temporally Disjoint Walks 的参数化复杂性则截然不同。对于这个问题，我们可以证明，与路径版本相反，即使底层图是星形图，想必也无法仅以源汇对的数量为参数得到固定参数算法。

- Temporally Disjoint Walks 即使底层图是星形图，对于源汇对的数量而言仍是 W[1]-难的。

考虑到 Klobas 等人 [18] 证明了：当源汇对的数量作为参数且底层图不受限制时，
Temporally Disjoint Walks比 Temporally Disjoint Paths 更容易求解，这一点相当令人意外。如前所述，他们证明了一般情况下，当源汇对的数量为常数时，
Temporally Disjoint Walks可以在多项式时间内求解；而 Temporally Disjoint Paths即使只有两个源汇对，也是 NP难的。另一方面，如果将底层图限制为路径，则我们可以使
Temporally Disjoint Walks达到固定参数可处理性。

若底层图是路径，则• Temporally Disjoint Walks关于源汇对数量这一参数是固定参数可解的。

我们的结果较为全面地展示了 TemporallyDisjoint Paths 和 Temporally Disjoint Walks 的参数化复杂性，其中考察了源汇对的数量以及底层图的结构参数（尤其是衡量其与森林相似程度的参数）。我们在第 5 节指出了尚未解决的问题和未来的研究方向。

2 预备知识与问题定义

我们分别用 $\mathbb{N}$ 和 $\mathbb{N}_0$ 表示不包括和包括 0 的自然数。从 a 到 b 的 $\mathbb{N}_0$ 上的区间记作 $[a,b] := \{i \in \mathbb{N}_0 \mid a \leq i \leq b\}$ 和 $[a] := [1,a]$。静态无向图 $G = (V,E)$ 由顶点集 V 和边集 $E \subseteq \binom{V}{2}$ 组成。在 G 中，从顶点 $s = v_0$ 到顶点 $z = v_k$ 、长度为 k 的 (s,z)-游走（或从 s 到 z 的游走）是一个由静态转移构成的序列 $P = (v_{i-1}, v_i)_{i=1}^{k}$ ，

使得对于所有 $i \in [k]$，都有 $\{v_{i-1}, v_i\} \in E$。如果对于任意 $i \neq j$ 都有 $v_i \neq v_j$，则称 (s,z)-游走 P 为 (s,z)- 路径（或从 s 到 z 的路径）。

时序图 $\mathcal{G} = (V, E_1, \ldots, E_T)$ 或 $\mathcal{G} = (V, (E_t)^{t\in[T]})$ 由顶点集 V 和 T 个边集 $E_1, \ldots, E_T \subseteq \binom{V}{2}$ 组成。$\mathcal{G}$ 的底层图是静态图 $G_U = (V, E_U)$，其中 $E_U := \bigcup_{i=1}^{T} E_i$。索引 $1, \ldots, T$ 是 $\mathcal{G}$ 的时间步。如果时序图 $\mathcal{G}$ 的底层图分别是树、星形图或路径，则称其为时序树、星形图或线图。如果 $e \in E_i$，则称顶点对 (e,i) 是 $\mathcal{G}$ 的时间边。图 (V, E_i) 是 $\mathcal{G}$ 的第 i 层。如果顶点 $v \in V$ 不与层 $i \in [T]$ 中的任何边关联，则称其在该层中是孤立的。如果存在（不一定互不相同的）层 j, $j' \in [T]$，使得 $j \leq i \leq j'$ 且 v 在 E_j 或 $E_{j'}$ 中不孤立，则称其在时间步 $i \in [T]$ 活跃。

在 $\mathcal{G}$ 中，从顶点 $s = v_0$ 到顶点 $z = v_k$、长度为 k 的时序 (s,z) 游走（或从 s 到 z 的时序游走）是由转移组成的序列 $P = ((v_{i-1}, v_i, t_i))_{i=1}^{k}$，满足对所有 $i \in [k]$，都有 $\{v_{i-1}, v_i\} \in E_{t_i}$，且对所有 $i \in [k-1]$，都有 $t_i < t_{i+1}$。若对于任意 $i \neq j$，都有 $v_i \neq v_j$，则称时序 (s,z) 游走 P 为时序 (s,z) 路径（或从 s 到 z 的时序路径）。P 的到达时间为 t_k。我们称 P 按照顺序 $v_0, v_1, \ldots, v_k$ 访问顶点 $V(P) := \{v_i \mid i \in [0,k]\}$。对于所有 $i \in [k-1]$，我们称 P 在时间区间 $[t_i, t_{i+1}]$ 内占据顶点 v_i。此外，我们称 P 在时间区间 $[t_1, t_1]$ 内占据 v_0，而 P 在时间区间 $[t_k, t_k]$ 内占据 v_k。我们称 P 遵循 $\mathcal{G}$ 的底层图中的（静态）路径或游走 $P' = (v_{i-1}, v_i)_{i=1}^{k}$。如果 P 的到达时间在 $\mathcal{G}$ 中所有时序 (s,z) 游走的到达时间中最小，则称 P 为最早到达。若对于所有满足 $i \in [k]$ 的 v_i，都有 t_i 是 $\mathcal{G}$ 中最早到达的 (s, v_i) 路径的到达时间，则称 P 为前缀最早到达。如果 $\mathcal{G}$ 中存在时序 (s,z) 路径，那么总是存在一条前缀优先的 (s,z) 路径，且可在多项式时间内计算出这样一条路径 [33]。给定两个时序游走 P_1、P_2，若存在一个顶点 v 和两个时间区间 $[a_1, b_1]$、$[a_2, b_2]$，其中 $[a_1, b_1] \cap [a_2, b_2] \neq \emptyset$，使得 v 在 $[a_1, b_1]$ 期间被 P_1 占据，且在 $[a_2, b_2]$ 期间被 P_2 占据，我们称 P_1 与 P_2 时间相交。若 P_1 与 P_2 不发生时序相交，则称它们时间不相交。问题 Temporally Disjoint Paths 的正式定义如下。

Temporally Disjoint Paths输入：一个时序图 $\mathcal{G} = (V, (E_t)^{t\in[T]})$，以及一个包含来自 $V \times V$ 的源汇对的多重集 S。问题：对于所有 $(s_i, z_i) \in S$，是否存在两两时间不相交的时序 (s_i, z_i)- 路径?

问题 Temporally Disjoint Walks 的输入相同，但询问对于所有 $(s_i, z_i) \in S$，是否存在两两时间不相交的时序 (s_i, z_i)-游走。给定一个 ˆTemporally Disjoint (Paths/Walks) 的实例，我们用 S 表示 V 中作为 S 的源点或汇点出现的顶点集合，也就是说，对于某个 $z \in V\} \cup \{z \in V \mid (s,z) \in S$，若存在某个 $s \in V\}$，则 $S = \{s \in V \mid (s,z) \in S$。

我们研究上述两个问题的（参数化）计算复杂性。我们使用《参数化复杂性理论》中的以下标准概念 [6, 7, 9]。参数化问题 $L \subseteq \{(x,k) \in \Sigma^* \times \mathbb{N}\}$ 是来自 $\Sigma^* \times \mathbb{N}$ 的所有实例 (x,k) 的一个子集，其中 k 表示参数。如果存在一个算法，能够在 f 仅取决于参数的某个可计算函数所构成的 $f(k) \cdot |x|^{O(1)}$ 时间内，判定每个实例 (x,k)（其中 L），则参数化问题 L 属于 FPT 类（即固定参数可解）。如果参数化问题 L 是 W[1]-困难的，那么它很可能不是固定参数可解的 [6, 7, 9]。

3 时间不相交（路径／游走）的参数化困难性

本节中，我们分析 Temporally Disjoint (Paths/Walks) 关于底层图的结构参数（与源汇对的数量结合）的参数化困难性。我们首先考虑输入时序图中的顶点数 $|V|$，并可将其视为底层图最大的结构参数。在第 3.1 节中，我们证明，当参数取底层图的点覆盖数与源汇对数量的组合时，Temporally Disjoint Paths 是 W[1]-困难的。在第 3.2 节中，我们证明，即使底层图是星形图，当参数取源汇对的数量时，Temporally Disjoint Walks 也是 W[1]-困难的。

我们首先证明，Temporally Disjoint (Paths/Walks)是 NP 难问题，且在以 $|V|$ 为参数时是 W[1] 难问题；即使在时序星形图上，这种困难性仍然成立。

定理 1. 在时序星形图上，Temporally Disjoint (Paths/Walks)是 NP 难问题；以顶点数 $|V|$ 为参数时，它是 W[1]难问题。

我们将通过从 W[1]难问题 Unary Bin Packing 进行参数化归约，证明关于 Temporally Disjoint Walks的定理 1，其中参数为箱子数 [14]。在我们的归约中，任一解中的所有时序游走都是路径，因此该结果对Temporally Disjoint Paths也成立。

在 Unary Bin Packing中，输入包括 m 个物品，其大小分别为 x_1、.....、$x_m \in \mathbb{N}$，箱子数 $b \in \mathbb{N}$，以及箱子容量 $B \in \mathbb{N}$；所有整数均以一元编码。问题是判定是否存在一种将物品分配到箱子的分配 $f\colon [m] \to [b]$，使得 $\sum_{i \in f^{-1}(j)} x_i \leq B$ 对所有 $j \in [b]$均成立；也就是说，分配到任一箱子的物品总大小至多为 B。参数为 b。不失一般性，我们可以假设 $S := \sum_{i=1}^{m} x_i = bB$。如果 $S > bB$，那么该实例显然是否实例。如果 $S < bB$，那么可以添加 $bB - S$ 个大小为 1 的物品，而不改变该实例是否为是实例。直观而言，这一限制的好处是，此问题可以被视为试图将每个箱子都装满。

归约背后的思路如下。对于每个箱子 $j \in [b]$，都有 B 个代表该箱子的终端对副本（此外还有一个额外的“虚拟”终端对）。归约输出一个时序图，其中每个物品 $i \in [m]$都对应一个由连续层构成的序列，用来表示该物品。表示物品 i 的构件按如下方式构造：为了能够在每个终端对之间发送足够多条两两时间不相交的时间游走，就必须在物品 i对应的构件中，同一个终端对之间恰好发送 x_i 条时间游走。

现在，我们将给出从 Unary BinPacking 到 Temporally Disjoint Walks 的参数化归约的形式化构造，其中前者以 b 为参数，后者以 $|V|$为参数。

构造1。设 $(x_1, \ldots, x_m, b, B)$ 是满足上述限制的 Unary Bin Packing 的一个实例。我们将定义 Temporally Disjoint Walks 的一个实例 $(\mathcal{G}, S)$。$\mathcal{G}$ 的顶点集为 $V := \{s_1, \ldots, s_b, z_1, \ldots, z_b, \tilde{s}, \tilde{z}, c\}$。因此，$|V| = 2b + 3$。我们令多重集 S 对每个 $j \in [b]$ 包含 B 个数对 (s_j, z_j)，并对 $b - 1$ 包含 $(\tilde{s}, \tilde{z})$ 个数对。现在我们来定义其层

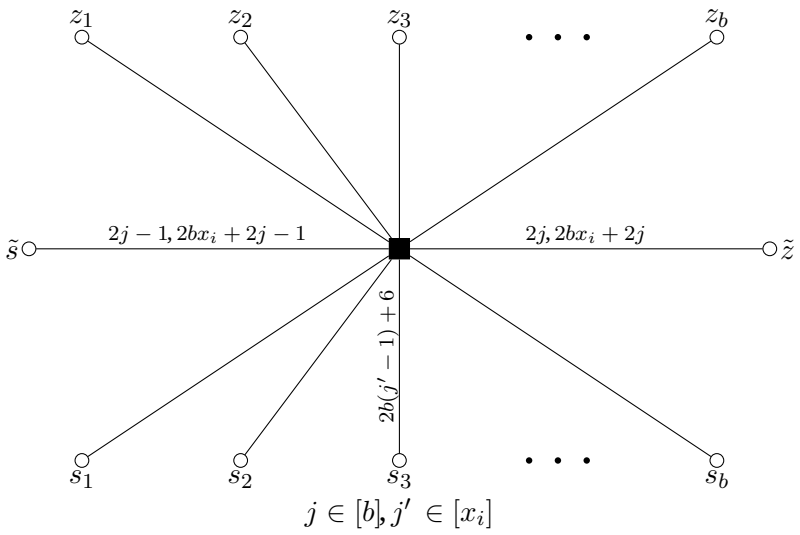

图1：构造1的示意图：边上的标签表示索引 j，使得该边包含在 E^i_j 中，其中 j 和 j' 取底部标示的值。中心顶点 c 用正方形表示。

。对于每个物品 $i \in [m]$，我们添加一个层序列 $E^i_1, \ldots, E^i_{2bx_i}$。我们令

$$
\begin{aligned}
E^i_{2j-1} &\coloneqq \{\{\tilde{s}, c\}\} \text{ for all } j \in [b],\\
E^i_{2j} &\coloneqq \{\{c, \tilde{z}\}, \{s_j, c\}\} \text{ for all } j \in [b],\\
E^i_{2bj'+j} &\coloneqq \{\{c, z_j\}\} \text{ for all } j \in [b], j' \in [x_i - 1],\\
E^i_{2bj'+j+1} &\coloneqq \{\{s_j, c\}\} \text{ for all } j \in [b], j' \in [x_i - 1],\\
E^i_{2bx_i+2j-1} &\coloneqq \{\{c, z_j\}, \{\tilde{s}, c\}\} \text{ for all } j \in [b], \text{ and}\\
E^i_{2bx_i+2j} &\coloneqq \{\{c, \tilde{z}\}\} \text{ for all } j \in [b].
\end{aligned}
$$

物品对应的层如图1所示。我们先放置第一个物品的层，然后放置第二个物品的层，依此类推。$\diamond$

至此，我们现在证明定理 1。

定理 1 的证明。证明基于构造1中描述的归约。显然，构造1可在多项式时间内完成，且输出实例中的顶点数受参数 b 的某个函数所界定。

剩下要证明的是：对于 Unary Bin Packing，输入实例 $(x_1, \ldots, x_m, b, B)$ 是是实例，当且仅当对于 Temporally Disjoint Walks，输出实例 $(\mathcal{G}, S)$ 是是实例。

($\Rightarrow$) 假设 $f\colon [m] \to [b]$ 是一个满足以下条件的分配：对于所有 $j \in [b]$，$\sum_{i \in f^{-1}(j)} x_i \leq B$。由于 $S \colon = \sum_{i=1}^{m} x_i = bB$，并且对于所有 $j \in [b]$，$\sum_{i \in f^{-1}(j)} x_i \leq B$，因此对于所有 $j \in [b]$，$\sum_{i \in f^{-1}(j)} x_i = B$。我们按如下方式构造集合 $\mathcal{S}$，其中包含 $\mathcal{G}$ 中时间不相交的 S 条游走。对于每个物品 $i \in [m]$，我们添加从 $s_{f(i)}$ 到 $z_{f(i)}$ 的 x_i 条游走，以及从 $\tilde{s}$ 到 $\tilde{z}$ 的 $b-1$ 条游走。对于 $j \in [x_i]$，第 j 条 $(s_{f(i)}, z_{f(i)})$- 游走在层 $E^i_{2b(j-1)+2f(i)}$ 中从 $s_{f(i)}$ 移动到 c，并在层 $E^i_{2bj+2f(i)-1}$ 中从 c 移动到 $z_{f(i)}$。

至于从 $\tilde{s}$ 到 $\tilde{z}$ 的 $b-1$ 条游走，其中前 f 条 $i(-1$ 条)游走在层 E^i_1、E^i_3、.....、$E^i_{2f(i)-1}$ 中从 $\tilde{s}$ 移动到 c，并在 E^i_2、E^i_4、.....、$E^i_{2f(i)}$ 中从 c 移动到 $\tilde{z}$。其余从 $\tilde{s}$ 到 $\tilde{z}$ 的 $b-f(i)$ 条游走发生在层 $E^i_{2bx_i+2f\ (i)\ +1}$、.....、$E^i_{2x_ib}$ 中。总而言之，这意味着 (s_j, z_j)-游走的数量为 $\sum_{\substack{i\in[m]\\ f(i)=j}} x_i = B$ 且（$\tilde{s}, \tilde{z}$）-游走的数量为 $m(b-1)$。

（$\Leftarrow$）假设 $\mathcal{S}$ 是图 $\mathcal{G}$ 中针对 S 的一组时间不相交游走。首先，考虑任意物品 $i \in [m]$ 及其对应的层 $E^i_1, \ldots, E^i_{2bx_i}$。在这些层中，至多有 x_i 条路径可以从任意 s_j 移动到 z_j，因为边 $\{s_j, c\}$ 和 $\{c, z_j\}$ 各自只出现 x_i 次。由于总共有 $bB = \sum_{i=1}^m x_i$ 条游走必须从某个 s_j 移动到 z_j，因此，对于与每个物品对应的层，恰好有 x_i 条路径必须从任意 s_j 移动到 z_j。接着，对于每件物品，至多有 $b-1$ 条游走可以从 $\tilde{s}$ 移动到 $\tilde{z}$，因此恰好有 $b-1$ 条游走必须这样移动。要实现这一点，唯一的方法是：与某个特定物品对应的层中，只包含某个单一索引 j 对应的（s_j, z_j）-路径。令 $f(i)$ 表示物品 i 对应的该索引。由于从任意特定的 s_j 到 z_j 只需要 B 条路径，因此有 $\sum_{i\in f^{-1}(j)} x_j \leq B$。因此，$f$ 是一种满足所需条件的物品到箱子的分配。 □

3.1 时间不相交路径的参数化困难性

本节证明，当参数取源汇对的数量与底层图的点覆盖数的组合时，
Temporally Disjoint Paths 是 W[1]-困难的。这表明，对于 Klobas 等人 [18] 提出的、适用于时序森林且以源汇对数量为参数的 Temporally DisjointPaths 固定参数算法，我们可以在多大程度上对其进行推广。我们的结果排除了这样的固定参数算法：以源汇对数量与底层图的树宽或反馈顶点数等参数的组合作为参数，求解 Temporally Disjoint Paths 。这是因为这两个参数都小于点覆盖数。若要实现可处理性，就必须使用大于点覆盖数或与其不可比的参数，例如反馈边数。我们将在第 4.1 节证明这确实可行。

定理 2. 当参数取 $|S|$ 个源汇对的数量与底层图的点覆盖数的组合时， Temporally Disjoint Paths 是 W*[1]*-困难的。

我们将通过从参数由颜色数 [8] 控制的 W[1]-困难问题 Mul-ticolored Clique 进行参数化归约，来证明定理 2。这里，给定一个 k-部图 $G = (V_1 \uplus V_2 \uplus \ldots \uplus V_k, E)$，我们要判断 G 是否包含一个大小为 k 的团。参数为 k。若 $v \in V_i$，则称 v 的颜色为 i。不失一般性，假设 $|V_1| = |V_2| = \ldots = |V_k| = n$，并且每个顶点至少有一个各颜色的邻居。令 $E_{i,j}$ 表示 V_i 与 V_j 中顶点之间所有边的集合。不失一般性，我们假设对于所有 $i \neq j$，$|E_{i,j}| = m$。

在完整描述如何构造一个 TemporallyDisjoint Paths 实例之前，我们先定义以下构件 $H(p, q, t)$。我们将把这个构件用于“顶点选择”和“边选择”。每个构件都关联一个源汇对，并且每个构件有三个参数： p、 q 和 t。直观地说， p 决定了从构件的源点到汇点的时序路径有多少种选择。参数 q 决定了有多少条其他时序路径可以穿过该构件，同时彼此之间以及与对应于该构件源汇对的时序路径都不发生时间相交。最后， t 是一个时间偏移。

非正式地说，当我们将 $H(p, q, t)$ 用作某种颜色 i 的顶点选择构件时，我们会将 p 设置为颜色 i 的顶点数量。因此，从源点

到该构件汇点的时序路径对应于选择一个颜色为 i 的顶点。我们会将 q 设置为 $k-1$，也就是说，对于每一种不同于 i 的颜色，都有一条时序路径可以穿过该构件。直观地说，这些时序路径中的每一条都会验证：我们为颜色 i 选定的顶点与为其他颜色选定的每个顶点之间都有一条边。

当我们将 $H(p,q,t)$ 用作颜色组合 i,j 的边选择构件时，我们将 p 设为颜色为 i 的顶点与颜色为 j的顶点之间的边数。因此，从构件的源点到汇点选择一条时序路径，就对应于选择一条颜色组合为i,j的边。我们将 q 设为 1，也就是说，至多只有另一条时序路径可以经过该构件。这条时序路径将用于验证：我们为颜色 i 选定的顶点与我们为颜色 j选定的顶点之间存在一条边（即在该构件中选定的边）。

形式化地说，该构件定义如下。

构造 2。我们构造一个时序图 $H(p,q,t)$，其中包含一个源点和一个汇点顶点，具体如下。

- 该构件的“骨架”由四个顶点 s,z,c,c' 和两条边 $\{s,c\},\{c',z\}$组成。
- 该构件的“主体”由 p 个不同的 $(q+1)$ 元组顶点组成，即 $(w_{(\ell)\,0}, w_{(\ell)\,1}, w_{(\ell)\,2}, \ldots, w_{(\ell)q})$，其中 $\ell \in [p]$。此外，它还包含 $2p(q+1)$ 条边，即 $\{c, w_{(\ell)r}\}, \{w_{(\ell)r}, c'\}$，其中 $\ell \in [p]$ 和 $0 \le r \le q$。

构件的标签如下所示。

- 边 $\{s,c\}$ 被标记为 1，而 $\{c',z\}$ 被标记为 $4n^3$（这将是所构造时序图中最大的时间标签）。
- 对于（$q+1$）元组（$w_{(\ell)}\,0$，$w_{(\ell)}\,1$，$w_{(\ell)}\,2$，...，$w_{(\ell)q}$），我们添加以下标签。– 我们将 $\{c, w_{(\ell)\,0}\}$ 标记为 $t+(2\ell-1)q-1$，并将 $\{w_{(\ell)}\,0$标记为 $t+2\ell q+2$。– 对于 $r \in [q]$，我们将 $\{c, w_{(\ell)r}\}$ 标记为 $t+(2\ell-1)q+2r-1$，并将 $\{w_{(\ell)r}$、$c'\}$ 标记为$t+(2\ell-1)q+2r$。

这样就完成了构件 $H(p,q,t)$ 的构造，其示意图见图 2。我们将 (s,z) 称为该构件的源汇对，将顶点 c 称为该构件的入口，将顶点 c' 称为该构件的出口。$\diamond$

我们首先证 明构件 $H(p,q,t)$ 的一些性质，以体现我们在开头给出的直观说明 g.

引理 3. 设 P 是构件 $H(p,q,t)$ 中从 s 到 z 的一条时序路径。如果 P 在某个时刻 $\ell \in [p]$ 访问了 $w_r^{(\ell)}$，并且 $r \in [q]$（即 $r \neq 0$），那么从该构件的入口顶点到出口顶点的每条时序路径 P' 都与 P 在时间上相交。

证明。设 P 是构件 $H(p,q,t)$ 中一条从 s 到 z 的时序路径，它在某个 $\ell \in [p]$ 和 $r \in [q]$ 的情况下经过 $w_r^{(\ell)}$。注意，边 $\{s,c\}$ 的标签为 1，而边 $\{c',z\}$ 的标签为 $4n^3$。对于 $r \neq 0$，边 $\{c, w_r^{(\ell)}\}$ 和 $\{w_r^{(\ell)}, c'\}$ 的标签连续，分别为 x 和 $x+1$，且 $1 < x < 4n^3 - 1$。因此，顶点 c 从时间 1 到 x 被 P 占据，而顶点 c' 从时间 $x+1$ 到 $4n^3$ 被 P 占据。现在，任何从 c 到 c' 的路径 P' 都需要在某个时间 $t > x$ 从 c 出发。然而，它最早也要到 $x+1$ 才能到达 c'。因此，P' 与 P 时间相交。

□

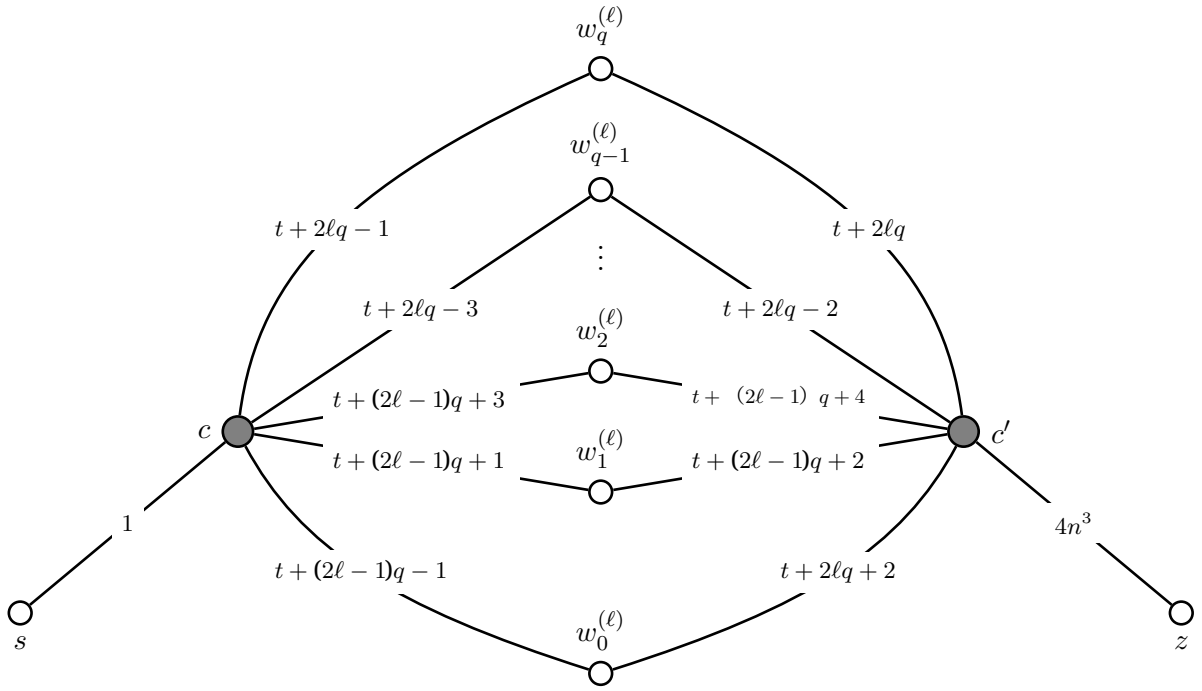

图 2：构件 $H(p,q,t)$ 的示意图。图中仅展示了一个 $(q+1)$ 元组 $(w_0^{(\ell)}, w_1^{(\ell)}, w_2^{(\ell)}, \ldots, w_q^{(\ell)})$。灰色顶点 c, c' 构成该构件的一个顶点覆盖。

引理 4。考虑构件 $H(p,q,t)$ 中一条从 s 到 z 的时序路径 P，其中 P 在某个 $\ell \in [p]$ 的情况下经过 $w_0^{(\ell)}$。那么，恰有 q 条时序路径 P'_1、……、P'_q 从该构件的入口顶点到出口顶点；这些路径两两不发生时序相交，且均不与 P 时间相交。此外，每条时序路径 P'_1、……、P'_q 都经过顶点 $w_1^{(\ell)}$、……、$w_q^{(\ell)}$ 中的一个。

证明。若 P 在某个 $\ell \in [p]$ 访问 $w_0^{(\ell)}$，则它会在时间 1 到 $t+(2\ell-1)q-1$ 期间占据入口顶点 c，而 P 会在时间 $t+2\ell q+2$ 到 $4n^3$ 期间占据出口顶点 c'。现在设 P'_r 是一条时序路径，它从 c 出发，接着经过 $w^{(\ell')}{}_r$，然后到达 $c'2$。假设 P 与 P'_r 不在时间上相交。注意，若 $\ell \neq \ell'$，则 P'_r 会在 c 或 c' 中与 P 时间相交。因此，我们可以得出 $\ell = \ell'$。此外，若 $r=0$，则 P'_r 和 P 会在 $w_0^{(\ell)}$ 中时间相交。由此可得 $r \neq 0$。我们有，P'_r 最迟在时间 $t+(2\ell-1)q+2r-1$ 从 c 出发，并在时间 $t+(2\ell-1)q+2r$ 到达 c'。不难验证，P'_r 和 $P'_{r'}$ 不在时间上相交，当且仅当 $r \neq r'$。引理得证。 □

现在继续构造时序图 $\mathcal{G}$ 和源汇对集合 S。

构造3。设 $G=(V_1 \uplus V_2 \uplus \ldots \uplus V_k, E)$ 是 Multicolored Clique 的一个实例。对于每种颜色 i，我们创建一个构件 $H(n, k-1, 2(i-1)(kn+m)+2k)$，并将其源汇对添加到$S$。对于每种颜色组合 i,j，满足 $i<j$，我们创建一个构件 $H(m, 1, 2(i-1)(kn+m)+2kn)$，并将其源汇对添加到 S。对于每种颜色 i，将 V_i 中的顶点按一种固定但任意的方式排序，即 $V_i=\{v_1^{(i)}, v_2^{(i)}, \ldots, v_n^{(i)}\}$。对于每种颜色组合 i,j，满足 $i<j$，将

$E_{i,\ j}$ 中的边按一种固定但任意的方式排序，即 $E_{i,\ j} = \{e_1^{(i,j)}, e_2^{(i,j)}, \ldots, e_m^{(i,j)}\}$。此外，我们还添加：两个顶点 $s_{i,\ j}$, $z_{i,\ j}$，并将源汇对 $(s_{i,\ j}, z_{i,\ j})$ 添加到 S； $2m$ 个顶点$\{v_e, v'_e \mid e \in E_{i,j}\}$；以及 $2n$ 个顶点 $\{u_\ell^{(i,j)} \mid v_\ell^{(i)} \in V_i\} \cup \{v_\ell^{(i,j)} \mid v_\ell^{(j)} \in V_j\}$。我们添加以下边。

- 我们在 $s_{i,\ j}$ 与每个顶点 $u_{(i,\ j)\ \ell}$（其 $v_{(i)\ \ell} \in V_i$）之间添加一条边。
- 我们在每个顶点 $u_{(i,\ j)\ \ell}$（其 $v_{(i)\ \ell} \in V_i$）与对应于颜色 i 的构件的入口之间添加一条边。
- 我们在 $z_{i,\ j}$ 与每个顶点 $v_{(i,\ j)\ \ell}$（其 $v_{(j)\ \ell} \in V_j$）之间添加一条边。
- 我们在每个顶点 $v_{(i,\ j)\ \ell}$（其 $v_{(j)\ \ell} \in V_j$）与对应于颜色 j 的构件的出口之间添加一条边。
- 对于每个 $e \in E_{i,\ j}$，我们添加以下边。
  - 我们在 v_e 与对应于颜色 i 的构件的出口之间添加一条边。– 我们在 v_e 与对应于颜色组合 i, j 的构件的入口之间添加一条边。– 我们在 v'_e 与对应于颜色组合 i, j 的构件的出口之间添加一条边。– 我们在 v'_e 与对应于颜色 j 的构件的入口之间添加一条边。

For 对于每条边 $e_\ell^{(i,j)} = \{v_{\ell'}^{(i)}, v_{\ell''}^{(j)}\} \in E_{i,\ j}$，我们添加以下时间标签。

- 我们在 $s_{i,\ j}$ 和 $u_{(i,j)\ \ell'}$ 之间的边上添加标签 1。
- 我们在 $u_{(i,j)\ \ell'}$与对应于颜色 i的构件入口之间的边上添加标签 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) + 2(j-1) - 2$。
- 我们在 v_e与对应于颜色 i的构件出口之间的边上添加标签 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) + 2(j-1) + 1$。
- 我们在 v_e 与对应于颜色组合 i, j的构件入口之间的边上添加标签 $2(i-1)(kn+m) + 2kn + 2\ell - 1$。
- 我们将标签 $2(i-1)(kn+m) + 2kn + 2\ell + 2$ 添加到 v'_e 与对应颜色组合 i, j 的构件出口之间的边上。
- 我们将标签 $2(j-1)(kn+m) + 2k + (2\ell''-1)(k-1) + 2i - 2$ 添加到 v'_e 与对应颜色 i 的构件入口之间的边上。
- 我们将标签 $4n^3$ 添加到 $z_{i,\ j}$ 与 $v_{(i,\ j)\ \ell''}$ 之间的边上。
- 我们将标签 $2(j-1)(kn+m) + 2k + (2\ell''-1)(k-1) + 2i + 1$ 添加到 $v_{(i,j)\ \ell''}$ 与颜色 j对应的构件的出口之间的边上。

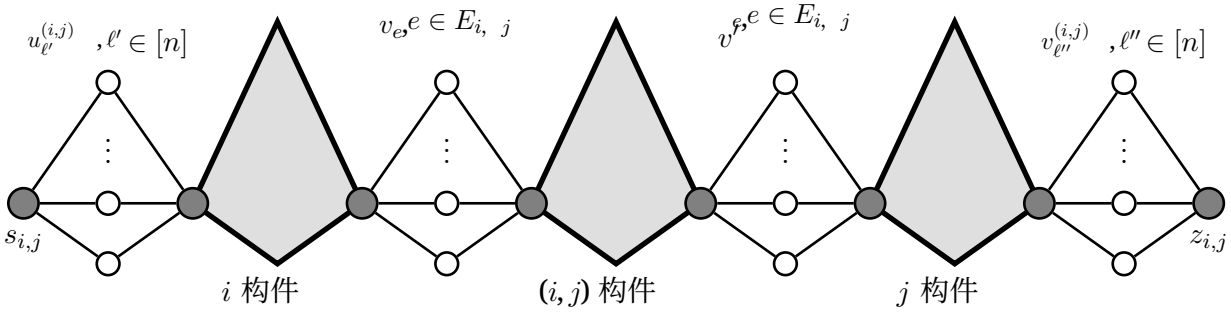

图3：颜色组合 i, j所添加顶点和边的示意图。注意，灰色顶点构成一个顶点覆盖。

至此， Temporally Disjoint Paths 实例 $(\mathcal{G} = (V, (E_t)^{t\in[T]})\ S)$ 的构造完成。图3给出了示意图。$\diamond$

构造3中描述的时序图 $\mathcal{G}$ 显然可以在多项式时间内计算。注意， $|S| \in O(k^2)$。此外，令 X 为所有作为某个构件入口或出口的顶点集合。注意， $X \cup S$ 构成了 $\mathcal{G}$ 的底层图的一个顶点覆盖，并且我们有$|X \cup S| \in O(k^2)$。

在证明归约正确性之前，我们先展示所构造的时序图 $\mathcal{G}$的以下两个性质。直观而言，我们证明，一旦时序路径进入一个构件，就必须穿过该构件的主体，然后离开该构件。

引理 5. 设 P是 $\mathcal{G}$中的一条时序路径，它访问了某个构件 $H(p, q, t)$的入口顶点 c，该构件存在于 $\mathcal{G}$中，且此后至少还访问两个顶点。那么在 c之后，路径 P 会访问主体中某个 $\ell \in \{0, 1, \ldots, p\}$ 和 $r \in [q]$ 对应的顶点 $w_r^{(\ell)}$，然后访问该构件的出口顶点 c'。

证明。设 P 是 $\mathcal{G}$中的一条时序路径。我们证明该引理，假设 P 经过对应于颜色 i 的构件的入口顶点 c，随后至少还经过两个顶点。通过类似的论证，我们也可以证明：如果 P 经过对应于颜色组合 i, j 的构件的入口顶点，则该结论同样成立。

反设 P 不访问组件主体中的顶点。根据构造，它要么是某个 $j > i$对应的顶点 $v_\ell^{(i,j)}$，要么是某个 $e \in E_j$对应的 $v'_{e,\ i}$，其中有 $j < i$。注意， $v_\ell^{(i,j)}$ 和 v'_e 在底层图中的度均为 2，因此路径只有一种可能的延伸方式。在第一种情况下，当 P 访问某个 $j > i$对应的 $v_\ell^{(i,j)}$ 时，路径需要继续前往 $s_{i,\ j}$。然而，边 $\{v_\ell^{(i,j)}, s_{i,j}\}$ 的标签小于边 $\{c, v_\ell^{(i,j)}\}$的标签。因此，我们可以排除 $P_{s,\ z}$ 访问 $v_\ell^{(i,j)}$ 的可能性。在第二种情况下，当 P 访问某个 $e \in E_j$对应的 $v'_{e,\ i}$，其中有 $j < i$时，路径需要继续前往与颜色组合 j、i对应的构件出口。同样，路径必须使用的第二条边的标签小于第一条边的标签。因此，我们也可以排除 P 访问 v'_e的可能性。由此可知， P 在 c 之后直接访问的顶点是组件主体中的某个顶点。组件主体中的所有顶点的度均为 2，因此 P 接下来访问的顶点是出口顶点 c'。

□

引理 6。设 P是 $\mathcal{G}$中的一条时序路径，它访问了 $\mathcal{G}$ 中存在的某个构件 $H(p,q,t)$的出口顶点 c'，并且其后至少还有两个顶点。在 c'之前，路径 P 访问了构件主体中某个 $\ell \in \{0,1,\ldots,p\}$ 和 $r \in [q]$ 对应的顶点 $w_r^{(\ell)}$，而在此之前，它访问了该构件的入口顶点 c 。

证明。该引理可用与引理 5 类似的方法证明。 □

假设 $(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Paths 的是实例，我们利用引理 5 和引理 6，证明解 $\mathcal{P}$ to $(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 中连接源汇对 $(s_{i,j},z_{i,j})$ 且具有 $1 \le i < j \le k$ 的时序路径具有一项重要性质。直观地说，这里我们展示这些时序路径会按图 3 所示穿过各个构件。

引理 7。假设 $(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Paths 的是实例，且令 $\mathcal{P}$ 为一个解。令 $1 \le i < j \le k$，并令 $P \in \mathcal{P}$ 为 $\mathcal{G}$ 中一条从 $s_{i,j}$ 到 $z_{i,j}$ 的时序路径。则 P 会经过以下每个构件的主体中的一个顶点。

- 对应颜色 i 的构件。
- 对应颜色 j 的构件。
- 对应颜色组合 i,j 的构件。

证明。根据 $\mathcal{G}$ 的构造，时序路径 P 首先访问某个顶点 $v_{\ell''}^{(i,j)}$，然后访问对应颜色 i 的构件的入口顶点 c_i （见图3）。由于 $z_{i,\ j}$ 未与 c_i直接连接，因此 P 至少还会访问另外两个顶点；根据引理 5，它会访问对应颜色 i 的构件主体中的某个顶点，然后访问该构件的出口顶点 c_i' 。

类似地，路径 P 会先访问某个顶点 $v_{\ell''}^{(i,j)}$，然后到达顶点 $z_{i,\ j}$。在访问 $v_{\ell''}^{(i,j)}$ 之前，路径 P 已访问对应颜色 j 的构件的出口顶点 c_j'。由于 $s_{i,\ j}$ 未与 c_j' 直接连接，因此 P 之前至少还访问了另外两个顶点；根据引理 6，它访问了对应颜色 j 的构件主体中的某个顶点，而在此之前访问了该构件的入口顶点 c_j 。

颜色 i 对应构件的出口顶点 c_i' 与某些顶点 v_e 相连，其中 $e \in E_{i,i'}$ 满足 $i < i'$；还与某些顶点 $v^{(i',i)}{}_\ell$ 相连，其中 $i' < i$满足 P。我们可以排除 P访问后一类顶点的情况，因为这些顶点之后只与顶点 $z_{i',i}$ 相连，而根据 $\mathcal{G}$的构造， P 无法从这些顶点继续前进。因此，我们可以假设，在经过颜色 i对应构件的出口顶点 c_i' 之后， P 会访问某个顶点 v_e ，其中 $e \in E_{i,i'}$ 满足 $i < i'$。从那里开始，路径P必须继续前往颜色组合 i对应构件的入口顶点 $c_{i,j}$。由于 $z_{i,i'}$ 与 $c_{i,i'}$不相连，因此 P至少还会访问另外两个顶点；根据引理 5，它会访问颜色组合i对应构件主体中的某个顶点i'，然后访问该构件的出口顶点。

现在考虑对应于颜色组合 i,i' 的构件。根据引理 4， $\mathcal{P}$ 中至多有两条路径可以穿过该构件，其中一条是对应于该构件源汇对的路径。上文已证明，每条从 $s_{i,\ j}$ 到 $z_{i,\ j}$ 的路径 $P \in \mathcal{P}$ 都会经过至少一个对应于某种颜色组合的构件。如果路径 P 经过两个或更多不同的颜色组合构件，那么必有一个颜色组合构件总共被至少三条路径经过。根据引理 4，这与 $\mathcal{P}$ 中所有路径两两时间不相交相矛盾。

□

现在我们可以证明定理 2 了。

定理 2 的证明。令 $(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 为构造3针对实例 Multicolored Clique $(V_1 \uplus \ldots \uplus V_k, E)$ 描述的实例 Temporally Disjoint Paths。回顾一下，$(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 可在多项式时间内计算，且 $\mathcal{G}$ 的底层图的点覆盖数为 k^2。我们证明，$(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Paths的一个是实例，当且仅当 G 包含一个大小为 k 的团。

$(\Rightarrow)$：假设 $(\mathcal{G}=(V,(E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Paths 的一个是实例，并令 $\mathcal{P}$ 为一个解。注意，在构造的实例 $(\mathcal{G},S)$ 中，多重集 S 不包含同一源汇对的多个副本；也就是说，它实际上是一个集合。因此，对于每个 $(s,z)\in S$，我们用 $P_{s,\ z}\in\mathcal{P}$ 表示 $\mathcal{P}$中从 s 到 z 的时序路径。

考虑颜色为 i的构件。令 (s_i,z_i) 为该构件的源汇对。由于 s_i 仅与该构件的入口顶点 c_i 相连，且 z_i 未与 c_i相连，因此 $P_{s_i,\ z_i}$会经过该构件的入口顶点 c_i，随后至少再经过两个顶点。根据引理 5，$P_{s_i,\ z_i}$ 会经过该构件主体的一个顶点，然后到达出口顶点 c_i'。之后，$P_{s_i,\ z_i}$需要继续前往 z_i，否则该路径就必须再次经过 c_i'。

我们可以得出，$P_{s_i,\ z_i}$ 具有以下形式：从 s_i 出发，它首先访问构件的入口 c_i，然后恰好访问构件主体中的一个顶点，接着访问构件的出口 c_i'，最后到达 z_i。

设 $w_r^{(\ell)}$为构件主体中被 $P_{s_i,\ z_i}$访问的顶点，其中 $\ell\in[n]$ 且 $r\in\{0,1,\ldots,k-1\}$。我们称，在对应于颜色 i的构件中，$w_r^{(\ell)}$被选中。

设在对应于颜色 i 的构件中，$X=\{v_\ell\in V_i \mid i\in[k]\wedge w_r^{(\ell)}$被选中，其中某个 $r\in\{0,1,\ldots,k-1\}\}$。我们证明 X是 G中的一个团。反设 X不是 G中的一个团。那么存在顶点 $v_\ell\in X\cap V_i$和 $v_{\ell'}\in X\cap V_j$，以及某个 i、$j\in[k]$，使得 $i\neq j$，且 $\{v_{\ell'},v_{\ell''}\}\notin E_{i,\ j}$。

考虑源汇对 $(s_{i,\ j},z_{i,\ j})$，并令 $P_{i,\ j}$表示 $\mathcal{P}$中从 $s_{i,\ j}$到 $z_{i,\ j}$的时序路径。根据引理 7，我们知道 $P_{i,\ j}$ 首先经过对应颜色 i的构件，然后经过对应颜色组合 i,j的构件，最后经过对应颜色 j的构件。

根据 $\mathcal{G}$的构造，存在一个 $e\in E_{i,\ j}$，使得 $P_{i,\ j}$ 在进入对应颜色组合 i,j 的构件之前恰好访问 v_e，而 $P_{i,\ j}$ 在离开对应颜色组合 i,j的构件之后恰好访问 v_e'。否则，根据引理 4，$P_{i,\ j}$ 会与 $\mathcal{P}$ 中对应于该构件源汇对的时序路径时间相交。

现在，根据 $\mathcal{G}$ 的构造和引理 4，我们知道 $v_{\ell'}$ 和 $v_{\ell''}$ 是边 e的两个端点；否则，$P_{i,\ j}$ 会与 $\mathcal{P}$ 中对应于对应颜色 i 和 j之一或两者的构件源汇对的时序路径时间相交。因此，我们有 $\{v_{\ell'},v_{\ell''}\}\in E_{i,\ j}$，这就产生了矛盾。

（$\Leftarrow$）：假设 G 是 Multicolored Clique 的一个是实例，并令 $X\subseteq\bigcup_{i=1}^{k}V_i$ 满足对所有 $i\in[k]$ 均有 $|X\cap V_i|=1$，且为 G中的一个多色团。

我们使用以下时序路径连接与不同颜色对应的构件的源汇对。令 $i\in[k]$，并令 (s_i,z_i) 为与颜色 i 对应的构件的源汇对。令$\{v_\ell^{(i)}\}=X\cap V_i$。令 c_i 表示与颜色 i 对应的构件的入口顶点，c_i' 表示其出口顶点。我们使用一条从 s_i 出发，接着经过 c_i，然后到达 $w_0^{(\ell)}$，再到达 c_i'，最后到达 z_i 的路径。注意，由于 $\mathcal{G}$ 中的每条边恰好有一个时间标签，这唯一确定了该时序路径。根据构件的构造，该路径使用的时间边具有

严格递增的时间标签。显然，连接与不同颜色对应的构件的源汇对的时序路径不会发生时间相交，因为它们不会经过公共顶点。

我们使用以下时序路径连接对应于颜色组合的构件的源汇对。令 $1 \le i < j \le k$，并令 $(s'_{i,\ j}, z'_{i,\ j})$ 为对应于颜色组合 i, j 的构件的源汇对。令 $\{v_{\ell'}^{(i)}\} = X \cap V_i$，令 $\{v_{\ell''}^{(j)}\} = X \cap V_j$，并令 $e_{\ell}^{(i,j)} = \{v_{\ell'}^{(i)}, v_{\ell''}^{(j)}\} \in E_{i,\ j}$。令 $c_{i,\ j}$ 表示对应于颜色组合 i, j 的构件的入口顶点，$c'_{i,\ j}$ 表示其出口顶点。我们使用这条路径：从 $s'_{i,\ j}$ 出发，接着到达 $c_{i,\ j}$，然后到达 $w_0^{(\ell)}$，再到达 $c'_{i,\ j}$，最后到达 $z'_{i,\ j}$。注意，由于 $\mathcal{G}$ 中的每条边都恰有一个时间标签，这唯一确定了该时序路径。根据构件的构造，这条路径使用的时间边具有严格递增的时间标签。容易看出，连接对应于不同颜色组合的构件的源汇对的时序路径不会发生时间相交，因为它们不会经过公共顶点。

我们还可以进一步观察到，连接与颜色对应的构件的源汇对的时序路径，与连接与颜色组合对应的构件的源汇对的时序路径不会时间相交，因为它们不会经过公共顶点。

接下来，我们描述如何用一条时序路径 $P_{i,j}$连接源汇对 $(s_{i,j}, z_{i,j})$ 和 $1 \le i < j \le k$。同样，设 $\{v_{\ell'}^{(i)}\} = X \cap V_i$，设 $\{v_{\ell''}^{(j)}\} = X \cap V_j$，并设 $e_{\ell}^{(i,j)} = \{v_{\ell'}^{(i)}, v_{\ell''}^{(j)}\} \in E_{i,j}$。构件以及 $P_{i,j}$经过和访问的顶点如图3所示。从 $s_{i,j}$出发，路径 $P_{i,j}$ 继续到达 $v_{\ell'}^{(i,j)}$，然后按以下方式经过与颜色 i 对应的构件：继续到构件的入口顶点，然后到构件主体中的顶点 $w^{(\ell')}{}_{j-1}$，再到构件的出口顶点。经过与颜色 i对应的构件后，路径继续到顶点 v_e。然后按以下方式经过与颜色组合 i, j 对应的构件：继续到构件的入口顶点，然后到构件主体中的顶点 $w_1^{(\ell)}$，再到构件的出口顶点。经过与颜色组合 i, j对应的构件后，路径继续到顶点 v'_e。然后按以下方式经过与颜色 j 对应的构件：继续到构件的入口顶点，然后到构件主体中的顶点 $w^{(\ell'')}{}_i$，再到构件的出口顶点。最后，路径继续到 $v_{\ell''}^{(i,j)}$，然后到达 $z_{i,j}$。

首先，我们论证 $P_{i,\ j}$ 确实是一条时序路径，即该路径所使用的时间边上的标签严格递增。该路径经过以下时间边：

1. $\{s_{i,j}, v_{\ell'}^{(i,j)}\}$ 颜色为一的标签。2. $\{v_{\ell'}^{(i,j)}, c_i\}$，其中 c_i 表示颜色为 i的构件的入口顶点，标签为 $2(i-1)(kn+m)+2k+(2\ell'-1)(k-1)+2(j-1)-2$。3. $\{c_i, w^{(\ell')}{}_{j-1}\}$ i $2\ i-1\ \ kn+m\ +$位 $2k+\ 2\ell'-1\ \ k-1\ +2\ j-1\ -1$

于颜色为 的构件主体中，标签为 ( )( ) ( )( ) ( )。4. $\{w^{(\ell')}{}_j\ c'_i\}$ i $2\ i-1\ \ kn+m\ +$, $2k+\ 2\ell'-1\ \ k-1\ +2\ j-1$

位于颜色为 的构件主体中，标签为 ( )( ) ( )( ) ( )。5. $\{c'_i, v_e\}$，其中c'_i 表示颜色为 i的构件的出口顶点，标签为 $2(i-1)(kn+m)+2k+(2\ell'-1)(k-1)+2(j-1)+1$。

6. $\{v_e, c_{i,j}\}$，其中$c_{i,j}$表示对应于颜色组合i, j的构件的入口顶点，标签为 $2(i-1)(kn+m) + 2kn + 2\ell - 1$。7. $\{c_{i,j}, w_1^{(\ell)}\}$位于对应于颜色组合 i, j的构件主体中，标签为 $2(i-1)(kn+m) + 2kn + 2\ell$。8. $\{w_1^{(\ell)}, c_{i,j}\}$位于对应于颜色组合 i, j的构件主体中，标签为 $2(i-1)(kn+m) + 2kn + 2\ell + 1$。9. $\{c'_{i,j}, v'_e\}$，其中$c'_{i,j}$表示对应于颜色组合i, j的构件的出口顶点，标签为 $2(i-1)(kn+m) + 2kn + 2\ell + 2$。10. $\{v'_e, c_j\}$，其中c_j表示颜色为 j的构件的入口顶点，标签为 $2(j-1)(kn+m) + 2k + (2\ell''-1)(k-1) + 2i - 2$。11. $\{c_j, w^{(\ell'')}{}_i\}$位于颜色为 j的构件主体中，标签为 $2(j-1)(kn+m) + 2k + (2\ell''-1)(k-1) + 2i - 1$。12. $\{w_i^{(\ell'')}, c'_j\}$位于颜色为 j的构件主体中，标签为 $2(j-1)(kn+m) + 2k + (2\ell''-1)(k-1) + 2i$。13. $\{c'_j, v_{\ell''}^{(i,j)}\}$，其中$c'_j$表示颜色为 j的构件的出口顶点，标签为 $2(j-1)(kn+m) + 2k + (2\ell''-1)(k-1) + 2i + 1$。14. $\{v_{\ell''}^{(i,j)}, z_{i,j}\}$标签为 $4n^3$。

在大多数情况下，标签严格递增这一点显而易见。对于第 5 个和第 6 个标签，请注意我们假设 $k < n$。对于第 9 个和第 10 个标签，请注意我们有 $i < j$。

接下来，我们证明 $P_{i,\ j}$ 不会与我们构造的其他时序路径发生时间相交。请注意，$P_{i,\ j}$ 只与以下时序路径共享顶点：同样经过对应颜色 i 的构件、对应颜色 j 的构件，以及对应颜色组合 i、j 的构件的时序路径。经过对应颜色 i 的构件的时序路径包括：对于某个 $i < j' \neq j$ 的时序路径 $P_{i,\ j'}$，对于某个 $i' < i$ 的时序路径 $P_{i',\ i}$，以及连接对应颜色 i 的构件的源汇对的时序路径。类似地，经过对应颜色 i 的构件的时序路径包括：对于某个 $i \neq i' < j$ 的时序路径$P_{i',\ j}$，对于某个 $j < j'$ 的时序路径 $P_{j,\ j'}$，以及连接对应颜色 j 的构件的源汇对的时序路径。最后，经过对应颜色组合 i、j 的构件的唯一其他时序路径，是连接该构件源汇对的时序路径。

考虑对应颜色 i 的构件。构造出的时序路径唯一共享的顶点是构件的入口顶点 c_i 和出口顶点 c'_i。连接该构件源汇对的时序路径 P_i 从时间 1 到时间 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) - 1$ 访问 c_i。具有 $i < j'$ 的时序路径 $P_{i,j'}$ 从时间 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) + 2(j'-1) - 2$ 到时间 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) + 2(j'-1) - 1$ 访问 c_i。因此，它们不会在 c_i 中发生时间相交 P_i（因为 $j' > 1$），并且它们在 c_i 中也不会彼此发生时间相交。类似地，具有 $i' < i$ 的时序路径 $P_{i',\ i}$ 从时间 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) + 2i' - 2$ 到时间 $2(i-1)(kn+m) + 2k + (2\ell'-1)(k-1) + 2i' - 1$ 访问 c_i。因此，它们也不会在 c_i 中发生时间相交 P_i，并且它们也不会彼此发生时间相交

c_i。此外，形式为 $P_{i,\ j'}$ 的时序路径不会与形式为 $P_{i',\ i}$ 的时序路径在 c_i 中发生时间相交，因为 $j' > i$ 和 $i' < i$。通过检查构造出的时序路径在构件出口顶点 c'_i 处的到达时间和离开时间，我们可以用类似的方法得出结论：这些路径在 c'_i 中不会发生时间相交。

通过类似的论证，我们可以证明，在对应于颜色 j 的构件中，所有构造出的时序路径都不会在时间上相交于 $P_{i,\ j}$。

最后，考虑对应于颜色组合 i, j 的构件。与该构件共享顶点的唯一时序路径是连接该构件源汇对的时序路径 P，其中 $P_{i,\ j}$。同样，这两条时序路径都会经过该构件的入口顶点 c 和出口顶点 c'。路径 P 在时间 1 到时间 $2(i-1)(kn+m) + 2kn + 2\ell - 2$ 期间经过 c。时序路径 $P_{i,\ j}$ 在时间 $2(i-1)(kn+m) + 2kn + 2\ell - 1$ 到时间 $2(i-1)(kn+m) + 2kn + 2\ell$ 期间经过 c。因此，这两条时序路径在 c 中不会时间相交。通过检查构造出的时序路径在该构件出口顶点 c' 处的到达时间和离开时间，我们可以用类似的方式得出结论：这些路径在 c' 中不会时间相交。

因此，所构造的时序路径两两之间不会时间相交。证明至此完成。

□

3.2 时间不相交游走的参数化困难性

Klobas 等人 [18] 将时序树上 Temporally Disjoint Walks关于源汇对数量 $|S|$ 的参数化复杂性留作一个开放问题。本节将证明，对于这一参数化，该问题是 W[1]-难的，即使在时序星形图上也是如此。考虑到 Klobas 等人 [18] 证明了 Temporally Disjoint Paths 在树上是固定参数可解的，这一结果或许有些出人意料。这意味着，尽管在任意图上，以 $|S|$ 为参数时，Temporally Disjoint Paths比 Temporally Disjoint Walks 更难（前者对于 $|S| = 2$是 NP 难的，而后者在 $|S|$为常数时可在多项式时间内求解），但在时序树上，Temporally Disjoint Walks比 Temporally DisjointPaths 更难。

定理 8. 当参数为源汇对的数量时，时序星上的 Temporally Disjoint Walks 是 W[1]-困难的。

我们将给出一个从 W[1]-困难问题 [8] Multicolored Clique出发的参数化归约。该问题的输入由一个整数 k 和一个经过适当 k着色的图 $G = (V_1 \uplus V_2 \uplus \ldots \uplus V_k, E)$ 组成，问题要求判断 G 是否包含一个大小为 k 的团。当然，这样的团必须恰好包含来自每个颜色类的一个顶点。

构造 4. 设 $(G = (V_1 \uplus V_2 \uplus \ldots \uplus V_k, E)\ k)$ 是 Multicolored Clique 的一个实例。我们可以不失一般性地假设 $|V_1| = |V_2| = \ldots |V_k| =: n$。假设 $V_i = \{v_1^i, \ldots, v_n^i\}$。现在我们将构造一个 Temporally Disjoint Walks 的实例 $(\mathcal{G} = (V, (E_t)_{t\in[T]})\ S)$。

我们首先描述 S。对于每个 $i \in [k]$，都有两个终端对 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$。连接这两个终端对的时序游走将编码在 V_i 中对一个顶点的选择。此外，对于每个满足 $i < j$ 的 $i, j \in [k]$，都有一个终端对 $(s_{i,\ j}, z_{i,\ j})$。这个终端对对应的时序游走用于验证在 V_i 和 V_j 中分别至少选出了一个顶点，并且这两个顶点相邻。令 S 表示这一组终端对。

接下来我们将定义 $\mathcal{G} = (V, E_1, \ldots, E_T)$。我们首先给出 V。对于每个 $i \in [k]$，都有两组顶点，第一组 W_i 用于连接 (s_i, z_i) 的时序游走，

另一组 $\tilde{W}_i$ 则用于 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走。令 $W_i := \{w_1^i, \ldots, w_{kn}^i, x_1^i, \ldots, x_{kn}^i, y_1^i, \ldots, y_n^i\}$，且 $\tilde{W}_i := \{\tilde{w}_1^i, \ldots, \tilde{w}_{kn}^i, \tilde{x}_1^i, \ldots, \tilde{x}_{kn}^i, \tilde{y}_1^i, \ldots, \tilde{y}_n^i\}$。然后，对于每条满足 $u \in V_i$ 和 $v \in V_j$（其中某个 $i < j$）的边 $e = \{u, v\} \in E$，都有顶点 $W_e := \{\alpha_u^j, \beta_e, \gamma_v^i\}$。最后，还有一个中心顶点 c，每条边都与它关联。令 $V := \{c\} \cup \left(\bigcup_{(s,\ z)\in S} \{s, z\}\right) \cup \left(\bigcup_{i\in[k]} W_i \cup \tilde{W}_i\right) \cup \left(\bigcup_{e\in E} W_e\right)$。

还需定义 $E_1, \ldots, E_T$。对于每个 $i \in [k]$，都有一个边集序列 $E_0^i, \ldots, E_{4kn+6}^i$。简单来说，在这个序列中，连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走会选定 V_i 中的一个顶点。对于 $j \neq i$，连接 $(s_{i,j}, z_{i,j})$ 的时序游走验证了所选顶点与在 V_j 中选定的顶点相邻。首先，对于 $\ell \in [kn]$，顶点 w_ℓ^i 在层 $E_{4\ell-3}$ 和 $E_{4\ell}^i$ 中与 c 相邻，而顶点 $\tilde{w}_\ell^i$ 在 $E_{4\ell-1}^i$ 和 $E_{4\ell+2}^i$ 中与 c 有边相连。接着，对于 $\ell \in [kn]$，顶点 x_ℓ^i 在 $E_{4\ell-2}^i$ 和 $E_{4\ell+1}^i$ 中与 c 相邻，且 $\tilde{x}_\ell^i$ 在 $E_{4\ell}^i$ 和 $E_{4\ell+3}^i$ 中与 c 有边相连。最后，对于 $\ell \in [n]$，顶点 y_ℓ^i 在层 $E_{4k(\ell-1)+1}^i$ 和 $E_{4k\ell+1}^i$ 中与 c 相邻；而对于 $\tilde{y}_\ell^i$，这些层则为 $E_{4k(\ell-1)+3}^i$ 和 $E_{4k\ell+3}^i$。此外，起始顶点 s_i 在层 E_0^i 中与 c 相邻，而 z_i 在 E_{4kn+4}^i 中与 c 有边相连。同样，对于 $\tilde{s}_i$ 和 $\tilde{z}_i$，相应的层分别为 E_2^i 和 E_{4kn+6}^i。

对于任意满足 $i < j$ 的 $i, j \in [k]$，存在一层 $E^{i,j}_1$，其中包含边 $\{s_{i,j}, c\}$；还有后续的第二层 $E^{i,j}_2$，其中包含所有在 V_j 中有邻居的 $v \in V_i$ 对应的 $\{c, \alpha_v^j\}$。此外，还存在一层 $E^{i,j}_{f-1}$，对于所有在 V_i 中有邻居的 $v \in V_j$，该层将 c 连接到 γ_v^i；最后，还有 $E^{i,j}_f$ 将 $z_{i,j}$ 连接到 c。接下来，考虑边 $e \in E$。假设该边的一个端点是 $v_a^i \in V_i$，另一个端点是 $v_b^j \in V_j$，且满足 $i < j$。那么，在层 $E^i_{4k(a-1)+4(j-1)}$ 中存在一条从 $\alpha_{v_a^i}^j$ 到 c 的边，在层 $E^i_{4k(a-1)+4(j-1)+2}$ 中存在一条从 c 到 β_e 的边，在 $E^j_{4k(b-1)+4i}$ 中存在一条从 β_e 到 c 的边，并在 $E^j_{4k(b-1)+4i+2}$ 中存在一条从 c 到 $\gamma_{v_b^j}^i$ 的边。

$\mathcal{G}$ 中各层的顺序如下。对于每个 $i, j \in k$，层 $E^{i,j}_1$ 和 $E^{i,j}_2$ 彼此相邻，且所有这类层都位于时序图的最开头。接下来是层 $E_1^1, \ldots, E_{4kn+6}^1$，然后是层 $E_1^2, \ldots, E_{4kn+6}^2$，依此类推。时序图以层 $E^{i,j}_{f-1}$ 和 $E^{i,j}_f$ 收尾；对于每个 $i, j \in k$，这两层彼此相邻。我们在图4中展示了该构造的示意图。$\diamond$

在正式证明这一结论之前，我们先简要介绍该构造为何正确的直观解释。首先，对于任意连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的 $i \in [k]$ 条时序游走 P_i 和 P_i，暂且假设顶点y_ℓ^i 和 $\tilde{y}_\ell^i$ 不存在。第一条游走 P_i 必须在层 E_0^i 从 s_i 移动到 c，因为 s_i 随后会变成孤立顶点。类似地，第二条游走 $\tilde{P}_i$ 必须在 E_2^i 到达 c。因此，P_i 必须在 E_1^i 离开c，否则它会与 $\tilde{P}_i$ 时间相交。它必须移动到 w_1^i。该顶点在层 E_4^i 又与 c 相邻，此后便成为孤立顶点。因此，P_i 必须在该层返回到 c。所以，$\tilde{P}_i$ 必须在层 E_3^i 离开 c，否则它会与 P_i 时间相交。它必须移动到 $\tilde{w}_1^i$。我们可以看到，这两条游走 P_i、$\tilde{P}_i$ 被“锁定”，只能交替地从中心 c 移动到顶点 w_ℓ^i 和 $\tilde{w}_\ell^i$。示意图见图4。

现在考虑顶点 y_ℓ^i 和 $\tilde{y}_\ell^i$ 存在的情形。如果 P_i 或 $\tilde{P}_i$ 分别移动到 y_ℓ^i 或 $\tilde{y}_\ell^i$，那么 P_i 和 $\tilde{P}_i$ 的移动模式就会被“打破”。现在我们可以得出两个观察。

- 如果两条游走几乎同时这样做，就会产生一个大小为 $O(k)$ 的区间，在此期间，两条游走都不会占据中心顶点 c。该区间对应于 V_i 中的一个顶点。因此，分别选择何时移动到 y_ℓ^i 和 $\tilde{y}_\ell^i$，就能从 V_i 中“选出”一个顶点。

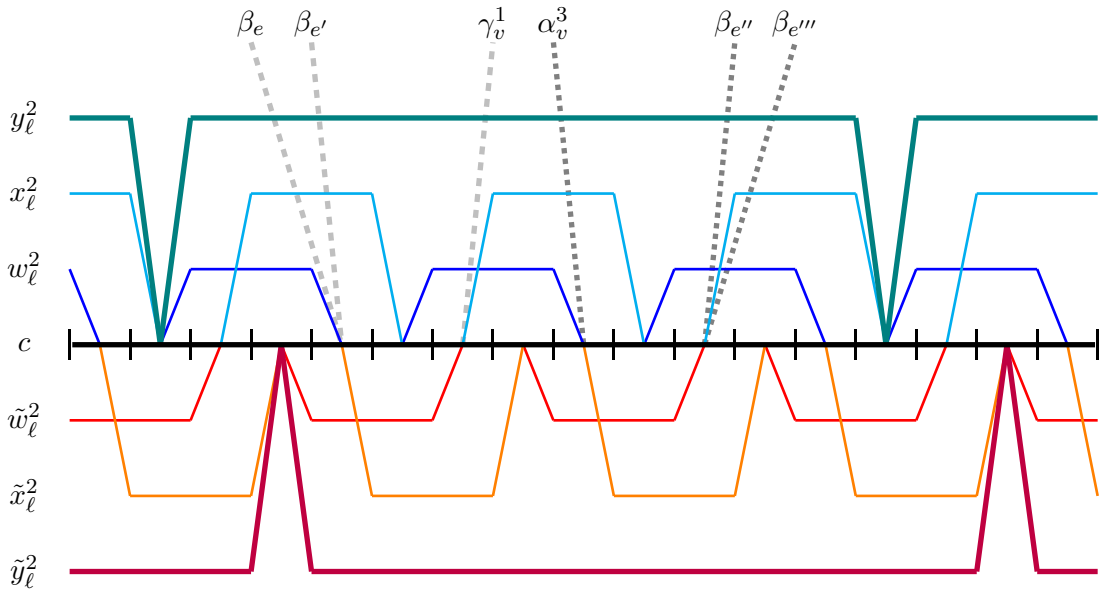

图4：构造 4 所定义的时序图 $\mathcal{G}$ 中与 $k = 3$ 种颜色下的颜色 $i = 2$ 对应部分的示意图。顶点用水平线表示。水平线的位置表示“顶点类型”（w_ℓ^i、x_ℓ^i、y_ℓ^i、$\tilde{w}_\ell^i$、$\tilde{x}_\ell^i$、$\tilde{y}_\ell^i$ 分别对应 $\ell \in [n]$ 和 c），如左侧所述。非水平线表示时间边，其标签对应于它们与中心顶点 c （中间的黑色水平线）连接的位置。越靠左的位置对应越早的时间标签。边 e、e' 将颜色1的顶点连接到颜色2的顶点 v。边 e''、e''' 将颜色3的顶点连接到颜色2的顶点 v。

- 两条游走移回 c 后，便会陷入类似的模式：它们交替地从中心移动到顶点 x_ℓ^i 和 $\tilde{y}_\ell^i$。（即使只有一条游走移到 y_ℓ^i 或 $\tilde{y}_\ell^i$，也会出现这种情况。）在这种模式下，它们无法移动到顶点 y_ℓ^i 或 $\tilde{y}_\ell^i$，因此每种颜色最多只能选出一个顶点。

如果没有选中任何顶点，也就是说，没有访问顶点 y_ℓ^i 或 $\tilde{y}_\ell^i$，那么对于任意 $i < j$，从 $s_{i,j}$ 到 $z_{i,j}$ 的时序游走 $P_{i,j}$ 都会与 P_i 或 P_i 时间相交。（对于 $i = k$，我们可以作出类似的观察，此时 i 和 j 互换角色。）因此，我们可以假设每种颜色都选中了一个顶点。可以观察到，$P_{i,j}$ 必须先从 $s_{i,j}$ 移动到 c，然后再移动到 α_e；其中 e 是一条连接 V_i 中某个顶点与 V_j 中某个顶点的边。通俗地说，只有当 e 的两个端点分别被选为颜色 i 和 j 对应的顶点时，才有可能做到这一点而不与任何时序游走 P_i、P_i、P_j、P_j 时间相交。由此，我们验证了所选顶点确实在 G 中构成一个团。

现在我们证明该归约的正确性。首先，我们将形式化证明上述关于连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时间游走 P_i、P_i 的性质成立。

引理 9。设 P_i、P_i 是分别连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走，且 P_i 与 P_i 时间不相交。则有以下结论。

- 若 P_i 未访问任何满足 $\ell \in [n]$ 的顶点 y_ℓ^i，且 $\tilde{P}_i$ 未访问任何满足 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$，则 P_i 会访问所有顶点 $w_1^i, \ldots, w_{kn}^i$，而 $\tilde{P}_i$ 会访问所有顶点 $\tilde{w}_1^i, \ldots, \tilde{w}_{kn}^i$。

- P_i 至多访问一个满足 $\ell \in [n]$ 的顶点 y_ℓ^i，且不访问任何满足 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$。
- 如果 P_i 经过 y_ℓ^i，那么 P_i 会经过所有顶点 $w_1^i, \ldots, w_{k(\ell-1)}^i$ 以及所有顶点 $x_{k\ell+1}^i, \ldots, x_{kn}^i$。此外，$\tilde{P}_i$ 会经过所有顶点 $\tilde{w}_1^i, \ldots, \tilde{w}_{k(\ell-1)}^i$ 以及所有顶点 $\tilde{x}_{k\ell+1}^i, \ldots, \tilde{x}_{kn}^i$。
- $\tilde{P}_i$ 最多经过一个满足 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$，且不会经过任何满足 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$。
- 如果 $\tilde{P}_i$ 经过 $\tilde{y}_{\ell'}^i$，那么 $\tilde{P}_i$ 会经过所有顶点 $\tilde{w}_1^i, \ldots, \tilde{w}_{k(\ell'-1)}^i$ 以及所有顶点 $\tilde{x}_{k\ell'+1}^i, \ldots, \tilde{x}_{kn}^i$。此外，$P_i$ 会经过所有顶点 $w_1^i, \ldots, w_{k(\ell'-1)}^i$ 以及所有顶点 $x_{k\ell'+1}^i, \ldots, x_{kn}^i$。
- 如果 P_i 经过 y_ℓ^i 且 $\tilde{P}_i$ 经过 $\tilde{y}_{\ell'}^i$，那么 $\ell = \ell'$。

证明。开始之前，注意到构造图 $\mathcal{G}$ 中的每条边最多有两个标签。这意味着，如果一条时序游走经过一个叶顶点（该顶点既不是源点也不是汇点），那么该游走何时到达该顶点、何时离开该顶点便已确定。检查边 $\{c, \alpha_v^j\}$、$\{c, \beta_e\}$ 和 $\{c, \gamma_v^j\}$ 上的标签可以发现，P_i 和 $\tilde{P}_i$ 不会经过这些边。

假设 P_i 不经过任何满足 $\ell \in [n]$ 的顶点 y_ℓ^i，且 $\tilde{P}_i$ 不经过任何满足 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$。根据构造，P_i 必须在第 E_0^i 层从 s_i 移动到 c，因为之后 s_i 会变成孤立顶点。同理，$\tilde{P}_i$ 必须在 E_2^i 中移动到 c。因此，P_i 必须在第 E_1^i 层离开 c，否则它会与 $\tilde{P}_i$ 时间相交。它必须移动到 w_1^i，因为我们假设它不会移动到 y_1^i。该顶点在第 E_4^i 层又与 c 相邻，此后则变为孤立顶点。因此，P_i 必须在该层返回 c。所以，$\tilde{P}_i$ 必须在第 E_3^i 层离开 c，否则它会与 P_i 时间相交。它必须移动到 $\tilde{w}_1^i$，因为我们假设它不会移动到 y_1^i。重复这一论证可知，两条游走 P_i 和 $\tilde{P}_i$ 被“锁定”，只能交替地从中心 c 移动到顶点 w_ℓ^i 和 $\tilde{w}_\ell^i$。图4对此作了示意。由此可知，P_i 经过所有顶点 w_1^i、……、w_{kn}^i，而 $\tilde{P}_i$ 经过所有顶点 $\tilde{w}_1^i$、……、$\tilde{w}_{kn}^i$。

时序游走 P_i 和 $\tilde{P}_i$ 只有在以下情况下才会偏离上述模式：P_i 访问一个带有 $\ell \in [n]$ 的顶点 y_ℓ^i，或 $\tilde{P}_i$ 访问一个带有 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$。假设 P_i 是第一个访问带有 $\ell \in [n]$ 的顶点 y_ℓ^i 的游走。若 $\tilde{P}_i$ 先访问带有 $\ell \in [n]$ 的顶点 $\tilde{y}_\ell^i$，情况类似。根据上述论证，我们可以立即得出：P_i 访问所有顶点 w_1^i、……、$w_{k(\ell-1)}^i$，而 $\tilde{P}_i$ 访问所有顶点 $\tilde{w}_1^i$、……、$\tilde{w}_{k(\ell-1)}^i$。现在我们仔细考察 P_i 何时从 y_ℓ^i 返回 c，以及这会带来哪些后果，并考察 P_i 和 P_i 接下来如何行动。令 t 表示 P_i 从 y_ℓ^i 返回 c 的时间步，也就是说，令 t 为边 $\{y_\ell^i, c\}$ 上的两个时间标签中较大的那个。根据构造，t 也是边 $\{y_{\ell+1}^i, c\}$ 和 $\{w_{k\ell+1}^i, c\}$ 上较小的时间标签，示意见图4。因此，P_i 既不能访问 $y_{\ell+1}^i$，也不能访问 $w_{k\ell+1}^i$。接下来我们考察 $\tilde{P}_i$ 的行为。显然，$\tilde{P}_i$ 在时间 t 不能占据中心顶点 c，否则它就会与 P_i 时间相交。因此，在时间 t，游走 $\tilde{P}_i$ 占据 $\tilde{w}_{k\ell}^i$、$\tilde{x}_{k\ell}^i$ 或 $\tilde{y}_\ell^i$，示意见图4。如果 $\tilde{P}_i$ 在时间 t 占据 $\tilde{w}_{k\ell}^i$，那么 P_i 会在时间 $t+1$ 移动到 c，并与 P_i 时间相交。由此可知，$\tilde{P}_i$ 在时间 t 占据 $\tilde{x}_{k\ell}^i$ 或 $\tilde{y}_\ell^i$。无论哪种情况，$\tilde{P}_i$ 都会在时间 $t+2$ 返回 c。因此，P_i 不能在时间 $t+2$ 占据 c。要实现这一点，唯一的办法是让 P_i 在时间 $t+1$ 移动到 $x_{k\ell+1}^i$。随后，P_i 在时间 $t+4$ 返回 c，这意味着 P_i 必须在时间 $t+3$ 离开 c。要实现这一点，唯一的办法是让 P_i 移动到 $\tilde{x}_{k\ell+1}^i$，然后在时间 $t+6$ 返回 c。重复这一论证可知，此时这两条游走 P_i 和 P_i 被“锁定”，只能交替地从中心 c 移动到顶点 $x_{\ell'}^i$ 和 $\tilde{x}_{\ell'}^i$，分别如此，持续 $\ell' \geq k\ell + 1$。由此可知，P_i 访问所有顶点 $x_{k\ell+1}^i$、……、x_{kn}^i，而 $\tilde{P}_i$ 访问所有顶点 $\tilde{x}_{k\ell+1}^i$、……、$\tilde{x}_{kn}^i$。此外，我们还可以得出，P_i 访问

没有带有 $\ell' \neq \ell$ 的 $y^i_{\ell'}$，也没有带有 $\ell' \in [n]$的顶点 $\tilde{y}^i_{\ell'}$；并且，如果 $\tilde{P}_i$ 因某个 $\ell' \in [n]$而访问顶点 $\tilde{y}^i_{\ell'}$，那么该顶点必须是 $\tilde{y}^i_\ell$ P_i。通过类似的一系列论证，我们可以证明，当 P_i 互换角色时，也会出现类似的行为。 □

引理 10. 假设 $(\mathcal{G} = (V, (E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Walks 的是实例，且 $\mathcal{P}$ 是一个解。设 P_i、P_i 分别为连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走，使得 P_i 和 P_i 时间不相交。设 $P_{i',j'} \in \mathcal{P}$ 是一条连接 $(s_{i',j'}, z_{i',j'})$ 的时序游走，其中 $i' = i$ 或 $j' = i$。如果 P_i 和 $\tilde{P}_i$ 中有一个既不经过满足 $\ell \in [n]$ 的顶点 y^i_ℓ，也不经过满足 $\ell \in [n]$ 的顶点 $\tilde{y}^i_\ell$，那么 $P_{i',j'}$ 与 P_i 或 P_i 在时间上相交。

证明。考虑 $i' = i$ 的情形；$j' = i$ 的情形可以用类似方法证明。假设 P_i 不经过满足 $\ell \in [n]$ 的顶点 y^i_ℓ，也不经过满足 $\ell \in [n]$ 的顶点 $\tilde{y}^i_\ell$。$\tilde{P}_i$ 不经过任何此类顶点的情形类似。

时序游走 $P_{i,j'}$（带有 $i < j'$）从 $s_{i,j'}$ 出发，然后移动到中心顶点 c。在下一个时间步，$P_{i,j'}$ 必须移动到某个顶点 $\alpha^{j'}_v$，该顶点带有 $v \in V_i$；否则，$P_{i,j'}$ 会与连接 $(s_{i'',j''}, z_{i'',j''})$ 的另一条时序游走 $P_{i'',j''}$ 或连接 (s_1, z_1) 的时序游走 P_1 在时间上相交。根据构造，$P_{i,j'}$ 会在某个大于 $\{s_i, c\}$ 上时间标签且小于 $\{c, z_i\}$ 上时间标签的时间步 t 从 $\alpha^{j'}_v$ 返回到 c。更准确地说，根据构造，存在一个顶点 $w^i_{\ell^\star}$，对应某个 $\ell^\star \in [kn]$，使得 $\{w^i_{\ell^\star}, c\}$ 上两个标签中较大的一个也是 t。现在考虑这样的情况：$\tilde{P}_i$ 既不访问任何带有 $\ell \in [n]$ 的顶点 y^i_ℓ，也不访问任何带有 $\ell \in [n]$ 的顶点 $\tilde{y}^i_\ell$。那么根据引理 9，P_i 会访问所有顶点 $w^i_1, \ldots, w^i_{kn}$，因此 P_i 与 $P_{i,j'}$ 在时间上相交。

剩下需要考虑的情况是 $\tilde{P}_i$ 其带有 $\ell \in [n]$，或者访问某个顶点 $\tilde{y}^i_\ell$，其带有 $\ell \in [n]$。由引理 9 可知，$\tilde{P}_i$ 恰好访问一个带有 $\ell^{\star\star} \in [n]$ 的顶点 $y^i_{\ell^{\star\star}}$，且不访问任何带有 $\ell \in [n]$ 的顶点 $\tilde{y}^i_\ell$。此外，引理 9 还告诉我们，在这种情况下，P_i 会访问所有顶点 $w^i_1, \ldots, w^i_{k(\ell^{\star\star}-1)}$ 以及所有顶点 $x^i_{k\ell^{\star\star}+1}, \ldots, x^i_{kn}$。我们立即得到，若 $\ell^\star \leq k(\ell^{\star\star} - 1)$，则 P_i 与 $P_{i,j'}$ 时间相交。根据构造，我们还知道，t 恰好比边 $\{x^i_{\ell^\star}, c\}$ 上较大的标签小 1。因此，若 $\ell^\star \geq k\ell^{\star\star} + 1$，则 P_i 与 $P_{i,j'}$ 时间相交。

我们可以得出结论，$k(\ell^{\star\star} - 1) < \ell^\star < k\ell^{\star\star} + 1$。对于剩下的情形，我们需要考虑所有连接 $(s_{i',j'}, z_{i',j'})$ 且满足 $i' = i$ 或 $j' = i$ 的时序游走 $P_{i',j'} \in \mathcal{P}$。上文分析了 $i' = i$ 和 $i < j'$ 的情形。我们观察到，根据构造，每个时序游走 $P_{i,j'}$ 都会在某个时间步 $t_{j'}$ 从某个 $\alpha^{j'}_v$ 返回到 c，并且存在一个顶点 $w^i_{\ell^\star_{j'}}$，对于某个 $\ell^\star_{j'} \in [kn]$，$\{w^i_{\ell^\star_{j'}}, c\}$ 上两个标签中较大的那个也为 $t_{j'}$。因此，对于所有 $j' > i$，$k(\ell^{\star\star} - 1) < \ell^\star_{j'} < k\ell^{\star\star} + 1$ 均成立。此外，对于所有 $i < j' < j''$，必有 $\ell^\star_{j'} \neq \ell^\star_{j''}$，否则时序游走 $P_{i,j'}$ 和 $P_{i,j''}$ 将会时间相交。

现在考虑 $i' < i$ 和 $i = j'$的情况。同样，我们可以观察到，根据构造，每个时序游走 $P_{i',i}$ 都必须在某个时间步 $t_{i'}$ 从 c 移动到某个 γ^i_v，并且存在一个顶点$\tilde{w}^i_{\ell^\star_{i'}}$，对于某个 $\ell^\star_{i'} \in [kn]$，$\{\tilde{w}^i_{\ell^\star_{i'}}, c\}$ 上的两个标签中较大的那个也为 $t_{i'}$。通过类似于前面所作的一系列论证，我们可以得出结论：对于所有 $i' < i$，都必须成立 $k(\ell^{\star\star} - 1) < \ell^\star_{i'} < k\ell^{\star\star} + 1$；对于所有 $i' < i'' < i$，都必须成立 $\ell^\star_{i'} \neq \ell^\star_{i''}$。最后，根据$\mathcal{G}$ 的构造，对于所有 $i' < i < j'$，必须成立 $\ell^\star_{i'} \neq \ell^\star_{j'}$。注意，$k(\ell^{\star\star} - 1) - k\ell^{\star\star} + 1 = k - 1$。由于存在 $k - 1$ 条不同的时序游走 $P_{i',j'}$，满足 $i' = i$ 或 $j' = i$，我们得到如下结论：每条满足 $i < j'$ 的时序游走 $P_{i,j'}$，都必须在某个$e \in E$通过移动到下一个可能的顶点 β_e 离开 c。同样，每条满足 $i' < i'$ 的时序游走 $P_{i',i}$，都必须在某个 $e \in E$通过从此前最近的可能顶点 β_e 移动而到达 c。这意味着每个 $P_{i,j'}$ 和 $P_{i',i}$ 都占据 c

连续三个时间步。此外，我们知道，P_i 无法从 c 移动到顶点 $\alpha_v^{j'}$,β_e或 $\gamma_v^{i'}$ ，而不与某个 $P_{i,\ j'}$ 或 $P_{i',\ i}$ 发生时序相交。然而，如果 P_i 在某个 $\ell \in [n]$ 从 c 移动到某个顶点 $w_\ell^i, x_\ell^i, y_\ell^i, \tilde{w}_\ell^i, \tilde{x}_\ell^i, \tilde{y}_\ell^i$ ，那么三步之后它必须移回 c 。因此，不存在连续三个时间步使 c 未被 P_i占据。由此可知， P_i与 $P_{i,\ j'}$时序相交。

□

引理9和引理10基本上表明，我们的归约中的顶点选择按预期发挥作用。由此得到以下推论。

推论 11。假设 $(\mathcal{G} = (V, (E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Walks$\tilde{}$ 的是实例，且令 $\mathcal{P}$ 为一个解。令 P_i、 $P_i \in \mathcal{P}$ 分别为连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走。存在一个 $\ell \in [n]$ ，使得 P_i 访问 y_ℓ^i ，且 $\tilde{P}_i$ 访问 $\tilde{y}_\ell^i$。此外， P_i 不访问任何满足 $\ell' \neq \ell$ 的 $y_{\ell'}^i$ ，且 $\tilde{P}_i$ 不访问任何满足 $\ell' \neq \ell$ 的 $\tilde{y}_{\ell'}^i$ 。

接下来，我们将展示在是实例中连接 $(s_{i,\ j},\ z_{i,\ j})$ 的时序游走 $P_{i,\ j}$ 对于 $i < j$ 的一个性质。

引理 12。假设 $(\mathcal{G} = (V, (E_t)^{t\in[T]})\ S)$ 是 Temporally Disjoint Walks 的是实例，且令 $\mathcal{P}$ 为一个解。令 $P_{i,\ j} \in \mathcal{P}$ 为连接 $(s_{i,\ j},\ z_{i,\ j})$ 的时序游走，其中 $i < j$。存在一个满足 $u \in V_i$ 和 $v \in V_j$ 的 $e = \{u, v\} \in E$ ，使得 $P_{i,\ j}$ 具有如下转移

- $(s_{i,j}, c, t_1)$, (α_u^j, c, t_3), (β_e, c, t_5), (γ_v^i, c, t_7),
- (c, α_u^j, t_2), (c, β_e, t_4), (c, γ_v^i, t_6), $(c, z_{i,\ j}, t_8)$,

对于某个 $t_1 < t_2 < \ldots < t_8$。

证明。时序游走 $P_{i,j}$ （带有 $i < j$）从 $s_{i,j}$ 开始，然后移动到中心顶点 c。因此，$P_{i,j}$ 的第一次转移是 $(s_{i,j}, c, t_1)$，其中 t_1。在下一个时间步，$P_{i,j}$ 必须移动到某个顶点 α_u^j ，满足 $u \in V_i$；否则，$P_{i,j}$ 会与连接 $(s_{i',j'}, z_{i',j'})$ 的其他时序游走 $P_{i',j'}$ 或连接 (s_1, z_1) 的时序游走 P_1 在时间上相交。由此可知，$P_{i,j}$ 的第二次转移是 (c, α_u^j, t_2)，对于 $t_2 = t_1 + 1$；而 $P_{i,j}$ 的第三次转移是 (α_u^j, c, t_3)，其中 $t_3 > t_2$。

接下来，我们考察 $P_{i,j}$ 的最后三次转移。最后一次显然是某个 t_8 对应的 $(c, z_{i,j}, t_8)$。前一次转移必须发生在时间 $t_7 = t_8 - 1$ ，否则 $P_{i,j}$ 会与另一条连接 $(s_{i',j'}, z_{i',j'})$ 的时序游走 $P_{i',j'}$ ，或与连接 (s_k, z_k) 的时序游走 P_k 在时间上相交。因此，$P_{i,j}$ 的倒数第二次转移是某个 $v \in V_j$ 对应的 (γ_v^i, c, t_7)，而倒数第三次转移是某个 $t_6 < t_7$ 对应的 (c, γ_v^i, t_6)。

剩下要说明的是，在 $P_{i,j}$ 的第三次转移与倒数第三次转移之间会发生什么。设 $P_i, P_i \in \mathcal{P}$ 分别是连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走。根据推论 11，存在一个 $\ell \in [n]$ ，使得 P_i 访问 y_ℓ^i ，而 $\tilde{P}_i$ 访问 $\tilde{y}_\ell^i$。此外， P_i 不会访问任何满足 $\ell' \neq \ell$ 的 $y_{\ell'}^i$ ，且 $\tilde{P}_i$ 不会访问任何满足 $\ell' \neq \ell$ 的 $\tilde{y}_{\ell'}^i$ 。因此，根据构造，存在一个大小为 $4(k-1)$ 的区间，在此区间内， c 既不被 P_i 占据，也不被 $\tilde{P}_i$ 占据（当游走分别移动到 y_ℓ^i 和 $\tilde{y}_\ell^i$ 时）；我们将该区间与颜色 i 对应。类似地，还存在一个这样的区间与颜色 j 对应。对于 (α_u^j, c, t_3)，时序游走 $P_{i,j}$ 进入对应颜色 i 的区间；对于 (c, γ_v^i, t_6)，则离开对应颜色 j 的区间，否则 $\tilde{}$ $\tilde{}$ $P_{i,j}$ 将与至少一个游走时间相交： P_i, P_i, P_j, P_j，其中 $P_j, P_j \in \mathcal{P}$ 分别是连接 (s_j, z_j) 和 $(\tilde{s}_j, \tilde{z}_j)$ 的时序游走。我们可以观察到，至多有 $k-1$

连接 $(s_{i',\ j'}, z_{i',\ j'})$ 的游走 $P_{i',\ j'} \in \mathcal{P}$ 可以进入和离开对应某种颜色的区间。这样的区间有 k 个，这样的游走有 $\binom{k}{2}$ 条，而每条这样的游走都需要进入至少两个这样的区间。根据鸽巢原理，如果 $P_{i,\ j}$ 进入了两个以上的区间，也就是说，进入颜色 i 对应的区间、颜色 j 对应的区间，以及至少一个其他颜色对应的区间，那么就有一个区间被 k 条游走访问，这与假设矛盾；该假设认为 $\mathcal{P}$ 中的所有时序游走两两时间不相交。因此，$P_{i,\ j}$ 离开颜色 i 对应的区间后，下一次访问中心顶点 c 时必须处于颜色 j 对应的区间。这只有在 $e = \{u, v\} \in E$ 和 $P_{i,\ j}$ 能够使用带有 $t_3 < t_4 < t_5 < t_6$ 的转移 (c, β_e, t_4) 和 (β_e, c, t_5) 时才有可能。 □

现在我们可以证明定理 8 了。

定理 8 的证明。令 $(\mathcal{G} = (V, (E_t)^{t\in[T]}\ S)$ 为构造 4 针对 Multicolored Clique instance $G = (V_1 \uplus \ldots \uplus V_k, E)$ 描述的 Temporally Disjoint Walks 实例。实例 $(\mathcal{G} = (V, (E_t)^{t\in[T]}\ S)$ 显然可以在多项式时间内计算出来，并且我们可以观察到 $|S| \in O(k^2)$。我们证明，$(\mathcal{G} = (V, (E_t)^{t\in[T]}\ S)$ 是 Temporally DisjointWalks 的是实例，当且仅当 G 包含一个大小为 k 的团。

($\Rightarrow$)：假设 $(\mathcal{G} = (V, (E_t)^{t\in[T]}\ S)$ 是 Temporally Disjoint Walks 的是实例，并令~$\mathcal{P}$为一个解。考虑颜色 i，并令 $P_i, P_i \in \mathcal{P}$ 分别为连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$ 的时序游走。根据推论 11，存在一个 $\ell \in [n]$，使得 P_i 经过 y^i_ℓ，且 $\tilde{P}_i$ 经过 $\tilde{y}^i_\ell$。此外，P_i 不会经过任何满足 $\ell' \neq \ell$ 的 $y^i_{\ell'}$，而 $\tilde{P}_i$ 不会经过任何满足 $\ell' \neq \ell$ 的 $\tilde{y}^i_{\ell'}$。注意，这意味着 ℓ 被唯一确定。我们称 v^i_ℓ 为颜色 i 对应的选定顶点。现在令 X 为所有颜色对应的选定顶点的集合，即 $X = \{v^i_\ell \in V_i \mid i \in [k]$，并且 v^i_ℓ 是颜色 $i\}$ 对应的选定顶点。我们声称 X 是 G 中的一个团。注意，根据定义，X 恰好包含每种颜色的一个顶点。

反设 X 不是 G中的一个团。那么存在 v^i_ℓ、$v^j_{\ell'} \in X$，使得 $i \neq j$，且 $\{v^i_\ell, v^j_{\ell'}\} \notin E$。考虑连接 $(s_{i、\ j}、\ z_{i、\ j})$ 的时序游走 $P_{i、\ j} \in \mathcal{P}$。根据引理 12，存在某个 $e = \{u, v\} \in E$，使得 $u \in V_i$ 且 $v \in V_j$，从而 $P_{i、\ j}$ 会经过顶点α^j_u、β_e，并且 γ^i_v。令 $P_{i、}$ $\tilde{P}_i \in \mathcal{P}$ 分别为连接 $(s_{i、}$ $z_i)$ 和 $(\tilde{s}_{i、}$ $\tilde{z}_i)$ 的时序游走，并且~令 $P_{j、}$ $P_j \in \mathcal{P}$分别为连接 $(s_{j、}$ $z_j)$ 和 $(\tilde{s}_{j、}$ $\tilde{z}_j)$ 的时序游走。根据 $\mathcal{G}$的构造~ ~（构造 4），时序游走 $P_{i、}$ $_{j、}$ $P_{i、}$ $P_{i、}$ $P_{j、}$ P_j 只有在 $u = v^i_\ell$ 且 $v = v^j_{\ell'}$ 时才能两两时间不相交，因此 $e = \{v^i_\ell, v^j_{\ell'}\} \in E$，矛盾。

($\Leftarrow$)：假设 G 是 Multicolored Clique 的一个是实例，并设 $X \subseteq \bigcup_{i=1}^k V_i$ 满足：对所有 $i \in [k]$，$|X \cap V_i| = 1$；令 G中的一个多色团为 $X \subseteq \bigcup_{i=1}^k V_i$。我们按如下方式构造$(\mathcal{G}, S)$的一个解 $\mathcal{P}$。对于每个 $i \in [k]$，选取 $a_i \in [n]$ 使得 $v^i_{a_i} \in X \cap V_i$。对于每个$(s, z) \in S$，我们必须给出一条时间(s, z)-游走，然后证明这些游走两两时间不相交。

我们从 $i、\ j \in [k]$（满足 $i < j$）的$(s_{i,j}, z_{i,j})$开始。令 $v := v^i_{a_i}$、$v' := v^j_{a_j}$，以及 $e := \{v, v'\}$。注意，由于 $v, v' \in X$，边 e 存在于 G中。$(s_{i,j}, z_{i,j})$-游走$P_{i,j}$从$s_{i,j}$出发，在层 $E^i_{,1}{}^j$ 移动到 c，随后在层 $E^i_{,2}{}^j$ 继续前往 α_v。它停留在那里，直到层$E^i_{4k(a_i-1)+4(j-1)}$，此时返回 c。接着，它在层 $E^i_{4k(a_i-1)+4(j-1)+2}$移动到 β_e。然后，游走在层$E^j_{4k(a_j-1)+4i}$ 移动到 c，再前往$E^j_{4k(a_j-1)+4i+2}$中的 $\gamma_{v'}$。最后，在层 $E^i_{,\ f-1}{}^j$ 和 $E^i_{,f}{}^j$ 中，它经过 c 到达$z_{i,j}$。

接下来考虑游走 P_i 和 $\tilde{P}_i$，它们分别连接 (s_i, z_i) 和 $(\tilde{s}_i, \tilde{z}_i)$，对应于 $i \in [k]$。在 E^i_0中，游走 P_i从 s_i移动到 c。对于 $\ell \in [(a_i - 1)k]$，它从 c移动到 w^i_ℓ（位于层 $E^i_{4\ell-3}$），然后在 $E^i_{4\ell}$中返回 c。在层 $E^i_{4(a_i-1)k+1}$ 中，游走接着前往 $y^i_{a_i}$，并停留在那里，直到它返回

c 在 $E^i_{4a_ik+1}$中。对于每个 $\ell \in [(a_i+1)k,\ nk]$，游走从 c 移动到 x^i_ℓ（位于 $E^i_{4\ell-2}$ 中），然后在 $E^i_{4\ell+1}$中返回 c。在 E^i_{4kn+4}中，游走到达其目标 z_i。对于 $\tilde{P}_i$ 的游走也类似。

现在必须论证，所构造的游走两两不会发生时间相交。多个游走唯一共同使用的顶点是 c，因此它们不可能在其他顶点发生时间相交。它们也不可能在层 c 中的 $E^{i,\ r^j}$处发生时间相交，因为这些层中只有 $P_{i,\ j}$ 占据 c。在层 E^i_r 中，游走 P_i、$\tilde{P}_i$、$P_{i,\ j}$（其中 $j \in [i+1,k]$）和 $P_{j,\ i}$（其中 $j \in [i-1]$）占据顶点 c。游走 P_i 只在层 E^i_r（其中）占据 c

- r 模 $4 \in \{0,1\}$ 且 $r \le 4\,(a_i-1)\,k$ 或
- r 模 $4 \in \{1,2\}$ 且 $r \ge 4\,(a_i+1)\,k+1$。

游走 $\tilde{P}_i$ 只在层 E^i_r（其中）占据 c

- r 模 $4 \in \{2,3\}$ 且 $r \le 4\,(a_i-1)\,k$ 或
- r 模 $4 \in \{0,4\}$ 且 $r \ge 4\,(a_i+1)\,k+1$。

因此，这两者也不会时间相交。同时，游走 $P_{i,\ j}$ 和$P_{j,\ i}$ 仅占据层 $E^i_{4k(a_i-1)+4+(j-1)}\ \cdots$、$E^i_{4k(a_i-1)+4+(j-1)+2}$，因此它们彼此也不相交，且不与~ P_i 和 P_i相交。

□

4 时间不相交路径/游走的算法

本节中，我们介绍两种新算法，一种用于 Temporally Disjoint Paths，另一种用于 Temporally Disjoint Walks。对于 Temporally Disjoint Paths，我们在第 4.1 节提出一种固定参数算法，其参数为源汇对的数量与底层图的反馈边数的组合。对于 Temporally Disjoint Walks，我们在第 4.2 节给出一种以源汇对数量为参数的固定参数算法，该算法要求输入的底层图是一条路径。

在这两种情况下，我们推广了已有结果，并解决了 Klobas 等人提出的开放问题 [18]。此外，第 3 节中的计算复杂性结果表明，从分类角度来看，我们的算法结果已无法得到显著改进。

4.1 时间不相交路径算法

本节中，我们提出一种固定参数算法，用于求解 Temporally Disjoint Paths，其参数为源汇对数量与底层图反馈边集数的组合。这推广了 Klobas 等人 [18] 针对时序森林提出的固定参数算法；该算法以源汇对数量为参数，用于求解 TemporallyDisjoint Paths。定理 2 表明，我们大概无法用更小的参数（例如反馈顶点数或树宽）替代底层图的反馈边集数，同时仍保持固定参数可解性。

定理 13. Temporally Disjoint Paths 在以源汇对的数量 $|S|$ 和底层图的反馈边数的组合作为参数时是固定参数可解的。

该算法的总体思路如下。对于任意源汇对，底层图中连接该源汇对的路径数量可以由其反馈边数的某个函数界定。两条此类路径相交的次数也可以用同样的方法界定。因此，对于给定的路径集，这类交点的总数可以由反馈边集数和源汇对数量的某个函数界定。对于每个这样的交点，我们可以考虑时序路径以何种顺序经过该交点的所有可能性。这样便能获得足够的信息，以便在多项式时间内验证源汇对之间的连接方式及顺序是否可行。

定理 13 的证明。令 $(\mathcal{G} = (V, (E_t)^{t\in[T]}) \ S)$ 为 Temporally Disjoint Paths的一个实例，且其底层图是一条路径。令 G 表示 $\mathcal{G}$的底层图。回顾一下，$\hat{} \ S$ 表示 V 中作为 S的源点或汇点出现的所有顶点的集合。

由于我们只考虑解中的时序路径，因此可以执行以下预处理步骤。我们穷尽地移除所有在底层图 G 中度至多为一且既非源点也非汇点的顶点 $v \in V$，即 $v \notin S$。对于 $(s, z) \in S$，一条时序 (s, z)- 路径不可能经过这些顶点。令 G' 表示预处理步骤后的底层图，并令 F 为 G' 的一个最小反馈边集。若顶点 v 满足以下任一性质，则称其为 G' 中的有趣顶点。

- 顶点 v 是源点或汇点，即 $v \in \hat{S}$。
- 顶点 v 与一条反馈边相 incident。
- 顶点 v 在 G'中的度至少为三。

令 D表示有趣顶点的集合。我们有， G'中有 $O(|F| + |S|)$ 个有趣顶点，也就是 $|D| \in O(|F| + |S|)$。此外，G'中有 $O(|F| + |S|)$ 个有趣路径段，也就是 G' 中的路径，这些路径以有趣顶点为起点和终点，且没有内部顶点是有趣顶点 [4, 引理 2]。令 $\mathcal{P}$ 表示有趣路径段的集合。显然，对于 $(s, z) \in S$，每条时序 (s, z)-路径都遵循 G' 中的一条路径，而该路径由有趣路径段构成。对于每个 $(s, z) \in S$，都有 $2^{O(|F|+|S|)}$ 种不同的有趣路径段集合，能被一条时序 (s, z)-路径访问。对于时序 (s, z)-路径 P，我们将 P 所访问的有趣路径段集合称为 P的配置。注意，对于P 遍历其配置中的有趣路径段，存在唯一的顺序和方向。考虑所有源汇对，我们得到 $2^{O(|F|\cdot|S|+|S|^2)}$ 种可能的配置集合，对应于一个解中的时序路径。

对于我们考虑的每一组配置，我们考察带有 $(s, z) \in S$ 的时间 (s, z)-路径以何种顺序占据有趣顶点 D。对于一组固定的配置，单个固定有趣顶点有 $|S|^{O(|S|)}$ 种可能的排序。因此，由于 $|D| \in O(|F| + |S|)$，所有有趣顶点的排序共有 $|S|^{O(|F|\cdot|S|+|S|^2)}$ 种可能的组合。然而，其中某些排序可能彼此不兼容，也就是说，无法在不使时间 (s, z)-路径发生时间相交的情况下实现这些排序。考虑一组固定的配置。我们向这些排序集合添加以下约束。

- 如果连接 $(s, z) \in S$和 $(s', z') \in S$ 的两条时序路径 P, P' 在其配置中具有相同的特殊路径段，那么它们在该路径段两个端点处必定具有相同的相对顺序。

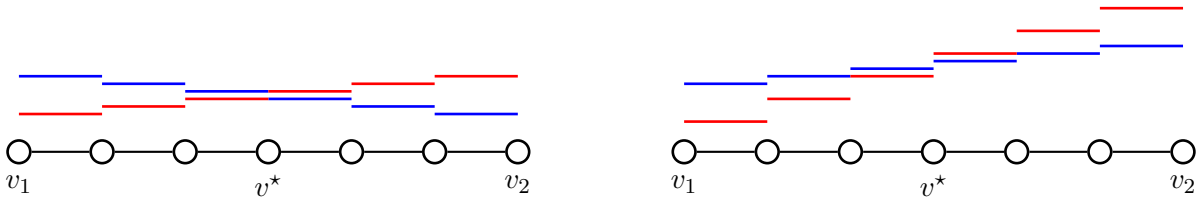

图5：从有趣顶点 v_1 到有趣顶点 v_2 的特殊路径段，以及沿该路径段行进的两条时序路径（红色和蓝色）的示意图。彩色边的水平位置表示时间标签。红色路径先经过顶点 v_1 ，蓝色路径随后经过。蓝色路径先经过顶点 v_2 ，红色路径随后经过。在两侧，红色时序路径都沿着从 v_1 到 v_2 的特殊路径段行进。在右侧，蓝色时序路径也沿着从 v_1 到 v_2的路径段行进；在左侧，蓝色时序路径则沿着从 v_2 到 v_1的路径段行进。我们可以看到，在这两种情况下，路径都在 $v^\star$处发生时间相交。

如果两条时序路径沿相同方向经过该特殊路径段，那么它们不可能在不发生时间相交的情况下彼此“超越”。也就是说，如果它们占据路径段第一个顶点的顺序与占据最后一个顶点的顺序相反，那么路径段中必定存在一个顶点，使两条路径在此处时间相交。类似地，如果它们沿相反方向经过该路径段，那么较早的时序路径必须先完成对该路径段的遍历，较晚的时序路径才能开始遍历；否则它们就会发生时间相交。图5对此作了说明。

此外，我们可以将每组排序视为 $<_1$ 上的一个偏序，其中 $D \times S$表示有趣顶点与源汇对的所有组合。对于每个源汇对（s， z） $\in S$，时序（s， z）路径的配置隐式定义了路径访问有趣顶点的顺序。考虑所有源汇对对应的这些顺序，我们得到另一个偏序 $<_2$，其定义在 $D \times S$上。令 $<^\star$ 表示关系 $<_{12}$ 在 $D \times S$上的传递闭包，其中 $x <_{12} y$ 当且仅当 $x <_1 y$ 或 $x <_2 y$。如果 $<^\star$ 是一个偏序，也就是说，不存在 x、 $y \in D \times S$ 使得 $x <^\star y$ 且 $y <^\star x$，那么 $<_1$ 和 $<_2$ 具有一个共同的线性扩张，我们称之为迭代顺序。如果满足这一条件，且这组排序也符合上述约束，那么我们称这组排序为有效。请注意，我们可以在多项式时间内检查给定的一组排序是否有效；若有效，还可以在多项式时间内计算出一个迭代顺序。

给定一个配置及一组有效排序，我们将证明，对于所有 $(s, z) \in S$，都可以在多项式时间内检查它是否可由两两时间不相交的 (s, z)-路径实现。具体而言，该算法执行以下步骤。

1. 从底层图 G 中反复移除度为 ≤ 1 的顶点，但源点或汇点（即属于 $\hat{S}$ 的顶点）除外。令 G' 为所得的（静态）图。

2. 计算 G' 的一个最小反馈边集 F。

3. 令 $V^{\geq 3}$ 表示 G' 中所有度至少为三的顶点。将森林 $G' - F$ 划分为一组极大路径 $\mathcal{P}$，其端点属于 $D = \bigcup_{e \in F} e \cup V^{\geq 3} \cup \hat{S}$，且中间顶点的度均为 2。由引理 2] 可知， $|\mathcal{P}| \in \mathcal{O}(|F| + |S|)$ [4, 。

这里， D 是有趣顶点的集合， $\mathcal{P}$ 是有趣路径段的集合。

4. 遍历所有时序 (s, z)-路径的配置，其中 $(s, z) \in S$.

5. 对于所有具有 $(s, z) \in S$ 的时序 (s, z)-路径，考虑一种固定配置 .
遍历所有有效排序集合，其中解中的时序 (s, z)-路径按照 D 占据顶点。

6. 考虑一个固定的有效排序集合。为 $D \times S$ 计算迭代顺序，并按该顺序遍历有趣顶点与源汇对的组合。设 $(v, (s, z))$ with $v \in D$ and $(s, z) \in S$ 为当前组合。

• 若 $v = s$，则令 $v^+ \in D$ 表示连接 s 和 z 的时序路径所经过的下一个有趣顶点。令 v' 为从 v 到 v^+ 的路径段中紧邻 v^+ 之前的顶点。计算一条从 v 到 v' 的前缀最早时间路径 P（每个顶点的最早可能到达时间），使其遵循从 v 到 v^+ 的路径段。若不存在这样的时序路径，则丢弃当前配置与有效排序集合的组合。否则，对于 P 中的每个转移 (v, w, t)，移除所有与 v 或 w 相接且具有时间标签 $t' \leq t$ 的时间边。

• 如果 $v = z$，则令 $v^- \in D$ 表示连接 s 和 z的时序路径上先前经过的有趣顶点。令 v' 为从 v^-到 v 的路径段中紧邻 v之前的顶点。令 $(\{v', v\}, t)$ 为连接 v' 和 v 且时间标签最小的时间边。如果不存在这样的时间边，则丢弃当前配置与有效排序集的组合。否则，移除所有关联 v' 或 v 且时间标签为 $t' \leq t$的时间边。

• 如果 $s \neq v \neq z$，则令 $v^+ \in D$ 表示连接 s 和 z 的时序路径上接下来经过的有趣顶点，并令 $v^- \in D$ 表示连接 s 和 z的时序路径上先前经过的有趣顶点。令 v_1 为从 v^- 到 v 的路径段中紧邻 v 之前的顶点，并令 v_2 为从 v 到 v^+ 的路径段中紧邻 v^+之前的顶点。计算一条从 v_1 到 v_2 的前缀最早时间路径P（每个顶点的最早可能到达时间），该路径在v_1 之后立即经过 v，然后沿着从 v 到 v^+的路径段前进。如果不存在这样的时序路径，则丢弃当前配置与有效排序集的组合。否则，对于 P中的每个转移 (v, w, t)，移除所有关联 v 或 w 且时间标签为 $t' \leq t$的时间边。

继续处理下一个有趣顶点与源汇对的组合。如果没有更多组合，则输出 YES。

7. 如果所有具有有效排序集合的配置组合都已被舍弃，则输出 NO。

由于有 $2^{O(|F|\cdot|S|+|S|^2)}$种可能的配置，以及 $|S|^{O(|F|\cdot|S|+|S|^2)}$ 个可能的有效排序集合，该算法的运行时间为 $2^{O((|F|\cdot|S|+|S|^2)^2\cdot\log|S|)}\cdot|\mathcal{G}|^{O(1)}$。注意，只需多项式开销，该算法也可以输出解。

根据前面的论证，很容易验证，如果算法输出 YES，那么我们面对的是一个是实例。

对于另一个方向，假设我们面对的是一个是实例，因此存在一个解 $\mathcal{S}$。对于每个 $(s, z) \in S$， $\mathcal{S}$ 中对应的时序 (s, z)-路径遵循底层图中的一条路径

G。由于时序路径不会重复经过顶点，因此可以彻底移除所有度为一的顶点（源点或汇点除外）。由此可知，$\mathcal{S}$ 中的每条时序 (s,z)-路径都遵循 G' 中的一条路径。考虑 G' 中对应的路径。每条路径都可以分割为若干特殊路径段，这些路径段的集合隐式定义了它们经过这些路径段的顺序和方向。因此，我们可以假设，在算法的某次迭代中，对于每个 $(s,z) \in S$，我们都在考虑 $\mathcal{S}$ 中时序 (s,z)-路径经过的特殊路径段集合，也就是该路径的配置。自然，我们还可以得出，每个有趣顶点都按一定顺序被 $\mathcal{S}$ 中的时序 (s,z)-路径（的某个子集）占据。根据算法描述中的论证，如果 $\mathcal{S}$ 中连接 $(s,z) \in S$ 和 $(s',z') \in S$ 的两条时序路径经过同一个特殊路径段，那么它们在该路径段两个端点处的相对顺序必须相同。因此，排序集合是有效的，算法也会考虑这些排序。

最后请注意，不失一般性，我们可以假设 $\mathcal{S}$ 中的时序 (s,z)-路径以一种前缀优先的方式遍历这些有趣的路径段（在所有彼此时间不相交的路径中），直到路径段中的倒数第二个顶点；而最后一个顶点则在最早可能时间被访问。若非如此，我们只需将时序路径段（直到倒数第二个顶点）替换为前缀优先的路径段，并将最后一次转移替换为最早可能的转移。由此可得，该算法输出 YES。 □

4.2 时间不相交游走算法

本节针对底层图为路径的情况，介绍一种以源汇对数量为参数的固定参数算法，用于解决 Temporally Disjoint Walks。回顾一下，底层图为路径的时序图称为时间线图。Klobas 等人 [18] 证明了在时间线图上，Temporally Disjoint Walks 是 NP难问题，并针对时序森林给出了一个以源汇对数量为参数的 Temporally Disjoint Paths 固定参数算法。定理 8 意味着，我们大概无法将这一固定参数算法改用于 Temporally Disjoint Walks。不过，对于时间线图的情况，我们可以实现可处理性。这回答了 Klobas 等人 [18] 提出的一个开放问题。

定理 14. Temporally Disjoint Walks在线图上的问题关于源汇对的数量 $|S|$ 是固定参数可解的。

在证明定理 14 之前，我们先研究实例Temporally Disjoint Walks 中使其游走长度总和最小的解 $\mathcal{S}$ 的性质（我们的算法将生成这样的解）。我们证明，可以用 $|S|$ 的某个函数来界定此类解中时序游走改变方向的次数上限 $\mathcal{S}$。此外，我们还证明，方向变化总是发生在 S 中源点和汇点“周围”的“区域”内（这些区域的大小可以用 $|S|$ 的某个函数来界定）。直观来说，这使我们能够遍历所有可能的情况：连接各源汇对的时序游走以何种方向、多少次以及按何种顺序经过 S 中源点和汇顶点周围的区域。给定这样一种可能情况后，我们就掌握了足够的信息，可以检查是否存在能够实现这种行为的时间不相交游走。

在本节余下部分，设 $\mathcal{S}$ 是 Temporally DisjointWalks 的一个实例的解，且该解使其时序游走长度之和最小。我们首先证明：如果 $\mathcal{S}$ 中的一条时序游走改变了方向，那么 $\mathcal{S}$ 中必定存在另一条时序游走，以如下方式促成这一行为。

引理 15。设 W 是 $\mathcal{S}$ 中一条时序 (s,z)-游走，且 (a,b,t),(b,a,t') 在 W 中相邻，其中 $t<t'$。那么，存在 $\mathcal{S}$ 中一条时序 (s',z')-游走 W'，使得 W' 中有 (c,a,t'') 或 (a,c,t'')，其中 $t<t''<t'$。

证明。设 (a',a,t_{-1}) 是 W 中位于 (a,b,t) 之前的转移，并设 (a,a'',t'_{+1}) 是 W 中位于 (b,a,t') 之后的转移。现在考虑通过从 W 中移除转移 (a,b,t)、(b,a,t') 得到的 $\hat{}W$。显然，$\hat{}\hat{}$ 是一个时序 (s,z)-游走，且其长度小于 W 的长度。由于 $W\notin\mathcal{S}$，必定存在一个 $W'\in\mathcal{S}$，它与 $\hat{}W$ 在时间上相交，但不与 W 相交。比较 $W\hat{}\hat{}$ 和 W，唯一一个被 W 占据的时间长于被 W 占据时间的顶点是顶点 a。时序游走 W 在时间 t_{-1} 至时间 t 占据 a，随后又在时间 t' 至时间 t'_{+1} 占据它（其中$t_{-1}<t<t'<t'_{+1}$）；而时序游走 $\hat{}W$ 在时间 t_{-1} 至时间 t'_{+1} 占据顶点 a。因此，$\hat{}W$ 占据 a 而 W 未占据的时间段是 (t,t')。由此可知，W' 必须在某个满足 $t<t''<t'$ 的时间 t'' 至少占据一次 a。假设 $\hat{}W$ 在时间t''到达 a，那么 $\hat{}W$ 必须包含一个转移 (c,a,t'')，其中某个顶点为 c。如果 $\hat{}W$ 从未到达 a，那么它必定从 a 出发，并包含一个转移 (a,c,t'')，其中某个 c。 □

根据引理 15，我们可以通过归纳证明：如果 W_0 在 $\mathcal{S}$ 中的时间游走改变了方向，那么 $\mathcal{S}$ 中就存在一系列时间游走，它们在 W_0 之前改变方向，之后接着 $\mathcal{S}$ 中的一条时间游走，该游走要么从其源点出发，要么到达其汇点。

引理 16. 设 W_0 是 $\mathcal{S}$中的一条时序（s_0,z_0）游走，并且（a_0,b_0,t_0）、（b_0,a_0,t'_0）连同 $t_0<t'_0$在 W_0中相邻。那么，存在时序（s_1,z_1）游走 W_1、...、时序（s_r,z_r）游走 W_r，它们位于 $\mathcal{S}$中；并且存在 $a_1,b_1,t_1,t'_1,\ldots,a_r,b_r,t_r,t'_r$，使得：

- 对于每个 $1\le i<r$，（a_i,b_i,t_i）、（b_i,a_i,t'_i）在 W_i、$t_{i-1}<t_i\le t'_i<t'_{i-1}$和$a_{i-1}=b_i$中相邻。
- 要么是（a_r，b_r，t_r），要么是（b_r，a_r，t_r），位于 W_r、$t_{r-1}<t_r<t'_{r-1}$、$a_{r-1}=b_r$ 中，并且要么是 $b_r=z_r$，要么是$b_r=s_r$。

证明。考虑 W_0。我们对 t'_0-t_0 使用归纳法证明该命题，即 W_0 离开顶点 a_0 到它返回之间的时间差。根据引理 15，存在一条位于 $\mathcal{S}$ 中的（s'，z'）-游走 W'，其中在 W' 中有（c，a_0，t''）或（a_0，c，t''），且 $t_0<t''<t'_0$。若 $a_0=s'$ 或 $a_0=z'$，则有 $r=1$ 和 $W'=W_r$，命题成立。

否则，我们有如下情况。假设 $s'\neq a_0\neq z'$。将 $c=a_1$、$a_0=b_1$、$s'=s_1$、$z'=z_1$和 $W'=W_1$重新命名。根据引理 15，（a_1，b_1，t_1）或（b_1，a_1，t_1）与$t_0<t_1<t'_0$ 是 W_1的转移。然而，由于 $s_1\neq b_1=a_0\neq z_1$，这两个转移必定在 W_1中连续，其中（a_1，b_1，t_1）是第一个，（b_1，a_1，t'_1）是第二个，且对某个$t'_0>t'_1>t_1>t_0$成立；否则，W_1 会与 W_0时间相交。

现在，W_1是 $\mathcal{S}$中的一条（s_1,z_1）-游走，且（a_1,b_1,t_1）、（b_1,a_1,t'_1）在 W_1中相邻，其中$t'_1-t_1<t'_0-t_0$。因此，根据归纳法可知，存在（s_2,z_2）-游走 W_2、.....、（s_r,z_r）-游走W_r，它们都在 $\mathcal{S}$中；并且 $a_2,b_2,t_2,t'_2,\ldots,a_r,b_r,t_r,t'_r$满足：

- 对于每个 $2\le i<r$，（a_i,b_i,t_i）、（b_i,a_i,t'_i）在 W_i中相邻，并且 b_i比 u_i更靠近 a_i，$t_{i-1}<t_i\le t'_i<t'_{i-1}$和 $a_{i-1}=b_i$。
- 在 W_r, $t_{r-1}<t_r<t'_{r-1}$, $a_{r-1}=b_r$ 中，（a_r,b_r,t_r）或（b_r,a_r,t_r）成立，并且 $b_r=z_r$ 或 $b_r=s_r$成立。

因此，引理得证。 □

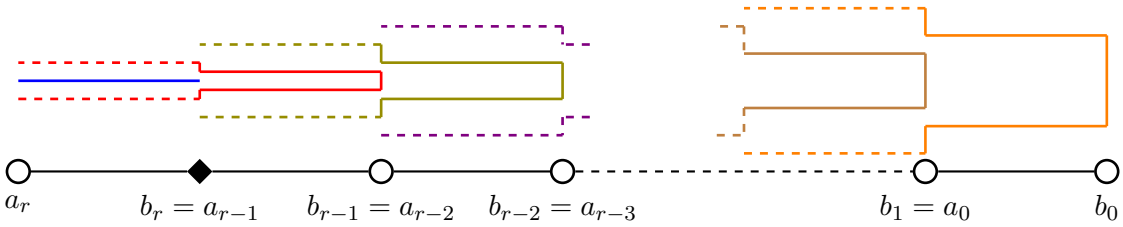

图6：引理 16 和推论17的示意图。彩色路径表示时序的 (s_r, z_r)-到(s_0, z_0)-游走。彩色边的水平位置表示时间标签。菱形顶点 $b_r = a_{r-1}$ 等于 s_r 或 z_r ，因此是一个感兴趣的顶点。由于$r \le |S|$， b_r 和 b_0 之间的距离也至多为 $|S|$。

由引理 16，我们可以得出两个重要推论，它们将帮助我们设计算法并证明其正确性。图 6 给出了示意图。第一个推论是， $\mathcal{S}$ 中时序游走的方向变化不会离源点或汇顶点太远。

推论17。设 W是 $\mathcal{S}$中的一条(s, z)-游走，满足(a, b, t)、(b, a, t')在 W中相邻，且$t < t'$。那么，存在 $\mathcal{S}$中的一条(s', z')-游走 $W' \neq W$，使得对于某个 a'，(s', a', t'')或(a', z', t'')中至少有一个出现在 W' 中，其中 $t < t'' < t'$，并且在该时序图的底层路径中， a和 a'之间的距离至多为 $|S|$。

$\mathcal{S}$ 中的时序游走不会过于频繁地改变方向，这是第二个推论。

推论18. 设 W 是 $\mathcal{S}$中的一个（s, z）游走。那么，在 W中，最多只有 $2|S|$ 对形如（a, b, t）、（b, a, t'）且满足 $t < t'$的三元组相邻。

现在，我们已经具备证明定理 14 所需的全部要素。

定理 14 的证明。设（$\mathcal{G} =$（V,（E_t）$_{t\in[T]}$） S）是 Temporally Disjoint Walks的一个实例，且 $\mathcal{G}$ 的底层图是一条路径。令 G 表示 $\mathcal{G}$的底层图。回想一下，$\hat{}$ S 表示 V 中作为 S的源点或汇点出现的所有顶点组成的集合。根据推论17，我们知道，不失一般性，可以假设解中的所有时序游走仅在距某个顶点（在 G中的距离）至多为 $|S|$ 的顶点处改变方向。令$D = \{v \in V \mid \exists v' \in \hat{}S$ ，使得 $\text{dist}_G(v, v') \le |S|\}$ 表示 V 中解里的某条时序游走可能改变方向的所有顶点组成的集合。注意到 $|D| \le 4|S|^2$。根据推论18，我们知道，不失一般性，解中的所有时序游走至多改变方向$2|S|$次。因此，对于（s, z）中的每个源汇对 $\in S$，我们需要考虑解中的时序（s, z）游走改变方向位置的 $|S|^{O(|S|)}$ 种可能。考虑所有源汇对后，我们需要考虑解中时序游走改变方向位置的 $|S|^{O(|S|^2)}$ 种可能配置。

考虑一种特定的配置。现在我们分析需要考虑时序游走的多少种不同相对顺序。为此，我们将每条时序游走视为由至多 $2|S|$ 个时序路径段组成，也就是说，在每个路径段的端点处，游走会改变方向（或开始/结束）。总共有 $2|S|^2$ 个时序路径段。注意，这些路径段中任意两个 P、P' 都满足以下性质：它们要么不会经过公共顶点，要么如果会经过，那么对于所有公共顶点，要么 P 都先经过这些顶点，要么 P' 都先经过这些顶点。否则， P 和 P' 就会时序相交。由此可知，存在

一个所有路径段的全序，使得每当两个路径段经过公共顶点时，该顺序就确定了这两个路径段中哪一个先经过每个公共顶点。总之，路径段共有 $|S|^{O(|S|^2)}$ 种可能的排序。

然而，重新连接所有路径段以形成相应的时序游走时，某些排序可能无法得到两两时间不相交游走。设 P_1 和 P_2 是某个时序游走 W 中的两个连续路径段，并且 P_1 是紧接在 P_2之前的路径段，那么该排序必须满足两个要求。

1. 路径段 P_1 必须在排序中位于路径段 P_2 之前。

2. 设顶点 v 是 P_1 的端点，也是 P_2的起点，这意味着 W 在顶点 v处改变了方向。那么，对于每个包含顶点 v的路径段 P'，在排序中，P'必须位于 P_1 和 P_2 之前，或者 P' 必须位于 P_1 和 P_2 之后。

第一个要求必须满足，否则连接 P_1 和 P_2 就无法得到时序游走。为了解释为什么必须满足第二个要求，设 W'是包含路径段 P'的时序游走。如果 $W = W'$，那么第一个要求就不满足。如果 $W \neq W'$，那么这两个时序游走将在顶点 v处时间相交。

如果满足上述两项要求，我们就称路径段的排序是有效的。给定一个排序，我们显然可以在多项式时间内检查它是否有效。

算法现在按如下方式进行：

1. 枚举解中的时序游走改变方向位置的所有可能配置。2. 对于每种配置，枚举该配置隐式给出的路径段的所有有效排序。3. 对于每种具有有效排序的配置，按照该排序遍历路径段。对于每个路径段，计算一条从起点到该路径段端点的前缀最早时间路径 P（该路径在每个顶点处的到达时间均为最早可能到达时间）。如果不存在这样的时序路径，则丢弃当前的配置与有效排序组合。否则，对于 P 中的每个转移 (v, w, t)，移除所有与 v 或 w 相接且具有时间标签 $t' \leq t$ 的时间边。继续处理下一个路径段。如果没有后续路径段，则输出 YES。

4. 如果所有满足有效排序的配置组合都已被舍弃，则输出 NO。

由于我们有 $|S|^{O(|S|^2)}$ 种可能的配置和 $|S|^{O(|S|^2)}$ 种可能的有效排序，该算法的运行时间为 $|S|^{O(|S|^4)} \cdot |\mathcal{G}|^{O(1)}$。请注意，该算法也可以在多项式开销下输出解。

根据前面的论证，很容易验证，如果算法输出 YES，那么我们面对的是一个是实例。

对于另一个方向，假设我们面对的是一个是实例。那么，存在一个解 $\mathcal{S}$，使其时序游走的长度之和最小。推论17和推论18表明，解中的每条时序游走在距离 $|S|$ 以内、且在 S 中作为源点或汇点出现的顶点处，改变方向的次数至多为 $2|S|$。因此，我们可以将每条时序游走分段为

至多 $2|S|$ 条时序路径，这些路径的端点属于集合 D （在证明开头定义）。如果定义路径段 P 小于路径段 P'，当且仅当这两个路径段有公共顶点，且每个公共顶点被 P 占据的时间都早于被 P' 占据的时间，那么这些路径段就构成一个偏序集。注意，对于所有有公共顶点的路径段数对 P、P'，P 必然小于 P'，或反之；否则，P 和P' 将会时序相交。因此，偏序的任意线性扩张都是路径段的一种有效排序。

由此可知，存在一种配置组合，其有效排序与该解一致。最后请注意，不失一般性，我们可以假设解中的时序路径段均为前缀优先（在彼此不时间相交的路径段中）；若并非如此，我们只需将某个时序路径段替换为一个前缀优先的路径段即可。由此可得，算法输出 YES。

□

5 结论

在 Klobas 等人 [18], 的研究基础上，我们针对 Temporally Disjoint Paths和 Temporally DisjointWalks 给出了其参数化计算复杂性的近乎完整图景，其中考察了底层图的图结构参数与源汇对数的组合。对于这两个问题变体，我们证明，即使底层图是星形图，以顶点数为参数的问题仍然是 W[1]-困难的，这表明仅限制底层图的结构不足以保证可处理性。因此，我们将源汇对数作为附加参数加以考察。对于 Tem-porally Disjoint Paths，我们证明，即使将源汇对数与底层图的点覆盖数结合起来，也很可能不足以实现固定参数可处理性。不过，我们证明，以反馈边数与源汇对数的组合作为参数时，该问题属于固定参数可解类。对于 Temporally Disjoint Paths，我们证明，即使底层图是星形图，以源汇对数为参数时，该问题仍然是 W[1]-困难的。然而，若底层图是路径图，则以相同参数化时该问题属于固定参数可解类。我们的结果揭示了该问题的路径变体与游走变体之间出人意料的差异，并解答了 Klobas 等人 [18] 提出的开放问题。

我们还留下了若干有待未来研究的方向。我们的困难性结果排除许多图结构参数作为进一步实现可处理性的选择。不过，还有一些候选参数既与点覆盖数和反馈边集无关，在星形图上的值也很大，因此引出了以下问题。

- 当参数取为源汇对的数量与底层图的割宽或带宽的组合时，
  Temporally Disjoint (Paths/Walks)属于固定参数可解还是 W[1]-困难?

在多智能体路径规划（MAPF）中，人们通常希望找到这样的解：使每个智能体到达其目的地 [30] 所需步数或动作数的总和或最大值最小。在我们的设定中，时序路径或游走的转移次数对应步数，而最后一次转移与第一次转移的时间标签之差（也称为持续时间）对应动作数（其中等待一个时间步视为一个动作）。我们可以观察到，在定理 1 和定理 2 的归约中，时序路径或游走的转移次数是固定的，这表明寻找转移次数较少的解仍然很难。我们认为，

持续时间可能是一个更有希望的参数，因为它在定理 2 和定理 8 的归约中都很大，由此引出以下问题。

- 当参数为源汇对的数量与解中任意时序（路径/游走）的最大持续时间的组合时，Temporally Disjoint (Paths/Walks)属于固定参数可解还是 W[1]-难?

最后，我们尚未解决这些结果是否也适用于非严格情形：在该情形下，时序（路径/游走）使用时间标签非递减（而非递增）的转移。我们推测，所有结果都可以推广到这种情形。

参考文献

[1] S. Almagor 和 M. Lahijanian。可解释的多智能体寻路。载于第19届国际自主智能体与多智能体系统会议（AAMAS '20）论文集，第34–42页，2020年。3[2] A. Andreychuk、K. Yakovlev、P. Surynek、D. Atzmon 和 R. Stern。连续时间多智能体寻路。《人工智能》，305:103662，2022年。3[3] D. Atzmon、R. Stern、A. Felner、G. Wagner、R. Bart´ak 和 N.-F. Zhou。鲁棒多智能体寻路与执行。《人工智能研究杂志》，67:549–579，2020年。3[4] M. Bentert、A. Dittmann、L. Kellerhals、A. Nichterlein 和 R. Niedermeier。介数中心性的 Brandes 算法自适应版本。《图算法与应用杂志》，24(3):483–522，2020年。25, 26[5] A. Casteigts、A.-S. Himmel、H. Molter 和 P. Zschoche。在等待时间约束下寻找时序路径。《算法学》，83(9):2754–2802，2021年。2[6] M. Cygan、F. V. Fomin、L. Kowalik、D. Lokshtanov、D. Marx、M. Pilipczuk、M. Pilipczuk 和 S. Saurabh。《参数化算法》。施普林格，2015年。5[7] R. G. Downey 和 M. R. Fellows。《参数化复杂性基础》。施普林格，2013年。5[8] M. R. Fellows、D. Hermelin、F. Rosamond 和 S. Vialette。多区间图问题的参数化复杂性。《理论计算机科学》，410(1):53–61，2009年。8, 17[9] J. Flum 和 M. Grohe。《参数化复杂性理论》，《理论计算机科学丛书》EATCS 丛书第XIV卷。施普林格，2006年。5[10] S. Fortune、J. Hopcroft 和 J. Wyllie。有向子图同胚问题。《理论计算机科学》，10(2):111–121，1980年。1[11] E. F¨uchsle、H. Molter、R. Niedermeier 和 M. Renken。时序图中的延迟鲁棒路线。载于第39届理论计算机科学专题国际研讨会（STACS）论文集，2022年。2

[12] O. Goldreich。在图推广的15数码问题中寻找最短移动序列是 NP 难问题。载于《复杂性与密码学研究：随机性与计算交互作用杂录》，第1–5页。施普林格，2011年。2[13] P. Holme 和 J. Saram¨aki。《时序网络理论》。施普林格，2019年。2[14] K. Jansen、S. Kratsch、D. Marx 和 I. Schlotter。重新探讨箱子数量固定的装箱问题。《计算机与系统科学杂志》，79(1):39–49，2013年。6[15] R. M. Karp。组合问题的计算复杂性。《网络》，5(1):45–68，1975年。1[16] K. Kawarabayashi、Y. Kobayashi 和 B. Reed。二次时间内解决不相交路径问题。《组合理论杂志》B辑，102(2):424–435，2012年。1[17] D. Kempe、J. Kleinberg 和 A. Kumar。时序网络的连通性与推断问题。《计算机与系统科学杂志》，64(4):820–842，2002年。3[18] N. Klobas、G. B. Mertzios、H. Molter、R. Niedermeier 和 P. Zschoche。时间中的无干扰游走：时间不相交路径。载于第30届国际人工智能联合会议（IJCAI）论文集，第4090–4096页，2021年。2, 3, 4, 8, 17, 24, 28, 32[19] B. Korte、L. Lov´asz、H. J. Pr¨omel 和 A. Schrijver。路径、流与 VLSI布局。施普林格，1990年。2[20] M. Latapy、T. Viard 和 C. Magnien。用于对随时间变化的交互进行建模的流图与链接流。《社会网络分析与挖掘》，8(1):61:1–61:29，2018年。2[21] J. Li、P. Surynek、A. Felner、H. Ma、T. S. Kumar 和 S. Koenig。大型智能体的多智能体路径规划。载于第33届 AAAI人工智能会议（AAAI ’19）论文集，第7627–7634页，2019年。3[22] H. Ma、T. S. Kumar 和 S. Koenig。考虑延迟概率的多智能体路径规划。载于第31届 AAAI人工智能会议（AAAI ’17）论文集，第3605–3612页，2017年。3[23] O. Michail。时序图导论：一种算法视角。《互联网数学》，12(4):239–280，2016年。2[24] R. Neil 和 P. D. Seymour。不相交路径——综述。《SIAM代数与离散方法杂志》，6(2):300–305，1985年。1[25] R. Neil 和 P. D. Seymour。图小类。XIII. 不相交路径问题。《组合理论杂志》B辑，63(1):65–110，1995年。1[26] O. Salzman 和 R. Stern。多智能体路径规划及多智能体取送问题中的研究挑战与机遇。载于第19届国际自主智能体与多智能体系统会议（AMAAS ’20）论文集，第1711–1715页，2020年

. 2

[27] G. Sharon、R. Stern、A. Felner 和 N. R. Sturtevant。用于最优多智能体寻路的基于冲突的搜索。《人工智能》，219：40–66，2015年。3[28] A. Slivkins。有向无环图中边不相交路径的参数化可处理性。《SIAM离散数学杂志》，24(1)：146–157，2010年。1[29] T. S. Standley。为协作式寻路问题寻找最优解。载于：第24届AAAI人工智能会议论文集（AAAI ’10），第173–178页，2010年。3[30] R. Stern。多智能体路径规划概述。载于：《人工智能》——第5届RAAI暑期学校，第96–115页，多尔戈普руд内，俄罗斯，2019年。施普林格。2, 3, 32[31] R. Stern、N. R. Sturtevant、A. Felner、S. Koenig、H. Ma、T. T. Walker、J. Li、D. Atzmon、L. Cohen、T. K. S. Kumar、E. Boyarski 和 R. Bart´ak。多智能体寻路：定义、变体和基准测试。载于：第12届国际组合搜索研讨会（SOCS）论文集，第151–159页，2019年。2, 3[32] J. ˇSvancara、M. Vlk、R. Stern、D. Atzmon 和 R. Bart´ak。在线多智能体寻路。载于：第33届AAAI人工智能会议论文集（AAAI ’19），第7732–7739页，2019年。3[33] H. Wu、J. Cheng、Y. Ke、S. Huang、Y. Huang 和 H. Wu。时序路径计算的高效算法。《IEEE知识与数据工程汇刊》，28(11)：2927–2942，2016年。5[34] J. Yu 和 S. M. LaValle。图上最优多机器人路径规划的结构与难处理性。载于：第27届AAAI人工智能会议论文集（AAAI ’13），第1443–1449页，2013年。2