受约束彩色令牌交换问题的复杂性阈值

Davide Bilò 拉奎拉大学信息工程、计算机科学与数学系，意大利

Stefano Leucci 拉奎拉大学信息工程、计算机科学与数学系，意大利

Andrea Martinelli
拉奎拉大学信息工程、计算机科学与数学系，意大利

arXiv:2601.09681v1 [cs.CC] 14 Jan 2026

摘要

考虑下面这个谜题：一片农田由若干块田地组成，每块田地里住着一位农夫、一只狐狸、一只鸡或一只毛毛虫。相邻田地里的生物可以交换位置，但狐狸不能遇到农夫，鸡不能遇到狐狸，毛毛虫也不能遇到鸡。目标是判断是否存在一个交换序列，能在避免任何不想要的相遇的同时，将这些生物重新排列成指定的最终配置。

上述谜题可以表述为一个具有 $k = 4$种颜色（即生物类型）的彩色令牌交换问题实例，其中只有某些颜色对可以交换。我们证明，即使表示农田的图是平面三次图，此问题仍然是 PSPACE-困难的。我们还证明，当涉及的生物类型不超过三种时，该问题是多项式时间可解的。为此，我们提供了一种更一般的算法，只要生物类型之间所有允许的交换所诱导的图为生成星，该算法就能判定任意 k取值下的实例。

我们的结果解决了 [Yangand Zhang 在 IPL 2025]，中明确提出的一个开放问题。该文证明了八种生物类型的情形是 PSPACE 完全的，但对于生物类型数量在三到七种之间的情形，其复杂度仍未确定。

2012 年 ACM学科分类计算理论 →问题、归约与完全性；计算数学 → 图算法

关键词与短语彩色令牌交换、令牌滑动、朋友与陌生人

补充材料本文中的 PSPACE 完全性归约所产生的交互式实例可在此处获取：
`https://www.isnphard.com/g/constrained-colored-token-swapping/`。

1 引言

设有一片由多个田地组成的农田，每块田地里住着农夫、狐狸、鸡或毛毛虫之一，如图1所示。相邻两块田地里的生物可以交换位置，并在途中相遇。然而，狐狸不想遇见农夫，鸡不想遇见狐狸，毛毛虫也不想遇见鸡。目标是重新安排这些生物的位置，使最终每块田地都由指定类型的生物占据，并且过程中不发生任何不希望出现的相遇。

这个谜题是一个著名脑筋急转弯的推广，可以表述为图上的彩色令牌交换 （CTS）问题 [15]，，并附加一项限制：只有部分交换是允许的。在 CTS 问题中，我们有一个无向连通的基图 G，其中每个顶点都包含一个令牌，其颜色为 k 种可能的颜色 之一，颜色取自 $\{1, 2, \ldots, k\}$，相同颜色的令牌无法区分。相邻顶点上的两个令牌可以交换，问题要求给定一个初始配置

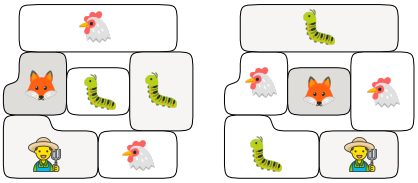


图1 脑筋急转弯中田地和生物的一种可能排列。

以及令牌在 G 上的最终 排列，求出一个交换序列，将初始配置变换为最终配置。由于所有 CTS 实例都可以求解，科学界一直专注于研究该问题的优化版本，其目标是使交换次数最少 [8, 11, 15, 17, 19]。

在受约束版本的 CTS（简称 CCTS）中，我们还会给定一个交换图，其顶点集恰好是令牌的颜色{1、2、.....、k}，并且只有当两个令牌的颜色在交换图中相邻时，才允许交换它们。图 2 展示了 CCTS 的一个实例。十五数码是一种由 CCTS 描述的著名谜题 [18]：每个编号方块都可以建模为一个具有独特颜色的令牌，而空白方块则可以视为第十六种颜色的额外令牌，它可以与任何其他颜色交换。更广泛地说，CCTS 推广了 Kornhauser、Miller、Spirakis 研究的卵石移动问题 [13]，该问题将十五数码推广到任意图和多个空白位置。与 CTS 不同，CCTS 的实例不一定可解，因此有必要研究其判定版本。

所有令牌颜色均互异的 CCTS 版本称为朋友与陌生人（FAS）问题，该问题由 [5] 提出。在他们的论文中，作者证明了朋友与陌生人图的若干结构性质。该图的顶点表示令牌的所有可能排列，且当两个配置之间仅相差一次允许的交换时，这两个顶点相邻。 [9] 还证明了朋友与陌生人图的其他结构性质。关于计算复杂度， [21] 证明，即使基图是次三次且为平面的，FAS 仍是 PSPACE 完全的。实际上， [21] 中讨论的 FAS 实例等价于颜色数为 8、交换图如图 2(d) 所示的 CCTS 实例。在论文的总结部分，作者提出一个尚未解决的问题：当颜色数介于 3 和 7 之间时，CCTS 的计算复杂度如何？[1]

我们的结果。我们通过证明 CCTS 在 $k \geq 4$ 情形下是 PSPACE-困难的，而在 $k = 3$ 情形下可于多项式时间内判定，从而解决了 [21] 留下的开放问题。

特别是，只要交换图包含一条有四个顶点的诱导路径，PSPACE 困难性结论就成立；上述脑筋急转弯便是这种情况，即使基图是三次平面图也成立。另一方面，我们实际上证明了一个更一般的结果：无论基图的结构如何，只要交换图是星形图，CCTS 问题就能在多项式时间内判定。这意味着，当 $k = 3$ 时，该问题可在多项式时间内求解，因为交换图要么是不连通的，要么是完全图，要么是星形图，而前两种情况都容易处理。这个结果扩展了 Kornhauser、Miller、Spirakis [13] 对图上卵石移动问题给出的刻画。

[1] 当颜色只有两种时，CCTS 可在多项式时间内判定，因为交换图必定为空图或完全图（后一种情况下，CCTS 实例实际上就是 CTS 实例）。

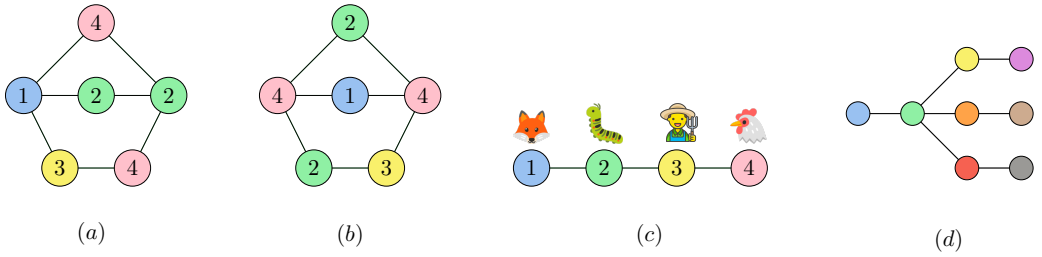


图 2 (a)、(b) 和 (c)：与图1中的脑筋急转弯对应的 CCTS 实例的初始配置 (a)、最终配置 (b) 和交换图 (c)。(d)： [21]中给出的 CCTS PSPACE 困难性归约所使用的交换图。

相关工作。 据我们所知，除了引言中提及的工作外，尚无其他关于 CCTS 的研究。关于 CTS，已知当图是星形图或路径 [3], 、具有 O 种颜色的团 [20], 、具有 O 种颜色的完全二部图 [2], ，以及当 $k = 2$ [20]时，该问题可在多项式时间内求解。在所有令牌颜色互不相同这一特殊情形下，该问题在团 [4], 、环图 [10],、扫帚图[12, 17], 、棒棒糖图 [12], 、路径的平方图 [7], 、完全分裂图 [22], 、完全二部图 [20], 和余图 [16]上也可在多项式时间内求解。另一方面，即使所有令牌颜色互不相同，CTS 在最大度为 3 [20], 的平面二部图、团 [3], 、完全二部图 [2], 和树 [1],上仍是 NP 困难的。最后，我们指出 CTS 是 APX 困难的 [15] ，其可近似性也受到了广泛关注（参见 [8]及其中的参考文献）。

预备知识。 下文不失一般性地假设，基图中包含交换图中每种颜色至少一个令牌；否则，可以等价地考虑这样一个实例：从交换图中删除所有未使用的颜色（即顶点）。此外，我们假设交换图是连通的；否则，该问题可归约为求解多个 CCTS 实例，其中每个实例具有：(i) 原交换图的一个连通分量 C 所诱导的交换图；以及 (ii) 原基图中所有且仅包含 C中某种颜色令牌的顶点所诱导的基图。最后，我们假设初始配置和最终配置中每种颜色的令牌数量相同；否则，显然可知这是一个否实例。

我们将在下一节介绍 PSPACE 困难性结果，第 3 节讨论适用于星形图交换图的算法，并在第 4 节给出一些总结性评论。

2 PSPACE 完全性

本节将证明，即使基图G 是三次平面图，且交换图是由 4 个顶点构成的路径，CCTS 仍然是 PSPACE 完全的。不失一般性，我们假设颜色为 i 的令牌只能与颜色为 $i-1$ （如果存在这种颜色）或$i+1$ （如果存在这种颜色）的令牌交换，如图1 (c) 所示的交换图。

我们通过从三次平面与/或约束图上的配置到配置非确定性约束逻辑问题 [6],进行归约来证明我们的结果。为简洁起见，我们将其简称为 NCL。要定义 NCL 问题，考虑一个无向三次平面图 G ，其中每条边的权重为{1，2}。我们将

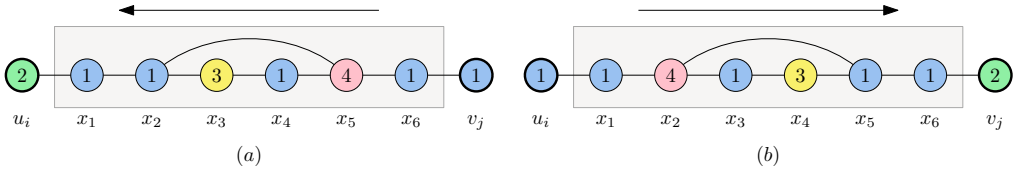


图3 一个边构件，分别朝向 u (a) 和 v (b)，并由灰色矩形包围。边构件的连接点是顶点 x_1 和 x_6。图中以粗体显示了模拟顶点 u 和 v 的构件的连接点 u_i 和 v_j。

权重为 1（分别为 2）的边称为轻边（分别称为重边）。[2] 图 G 中的每个节点[3] 恰好与三条边相连，且为一个AND 节点或一个OR 节点：AND 节点有一条相连的重边和两条相连的轻边，而 OR 节点相连的三条边均为重边。图 G 的一个配置是其各条边的一种方向安排；若对每个节点 v，指向节点 v 的入边权重之和至少为 2，则称该配置有效。这意味着，如果 v 是 OR 节点，那么 v 至少有一条相连的边必须指向 v。如果 v 是 AND 节点，则必须满足以下蕴含关系：若与 v 相连的重边背离 v，则与 v 相连的两条轻边都必须指向 v。NCL 问题给定同一图 G 的两个有效配置，分别称为初始配置和最终配置；目标是判断是否可以通过一系列边翻转从初始配置得到最终配置，且这些边翻转产生的中间配置始终有效。已知该问题是 PSPACE 完全的 [6]。

为了将一个 NCL 实例归约为一个 CCTS 实例，我们展示如何用合适的构件子图替换配置中的每条边、OR节点和 AND 节点。每个 AND 节点或 OR节点的构件都有三个不同的顶点指定为连接点，分别对应与该节点关联的三条边。每个边构件有两个顶点指定为连接点。如果 NCL 实例中的一条边 e 与节点 v 关联，那么 v的构件中的一个合适连接点将通过一条无向边与 e的构件中的一个连接点相连。构件之间只会通过交换放置在其连接点上的令牌进行交互；唯一能够跨越构件边界的令牌是颜色 1 或 2 的令牌，而颜色 3 和 4 的令牌则限制在其起始构件内。将颜色 1 的令牌视为“填充令牌”，并着重关注颜色 2 的令牌，会更便于理解。这是因为翻转一条边的唯一方法，是将一个颜色 2 的令牌 t 移入该边的构件；因此，t 会从旧边头所在的节点构件移至新边头所在的节点构件。

在以下小节中，我们首先描述各个构件，然后论证归约的正确性。为简单起见，我们先给出一种归约，其中 CCTS 的基图是平面且最大度为 3；随后修改该归约，在保持平面的同时得到一个三次基图。

2.1 边构件

边构件用于对 NCL 配置中的边 $e = (u, v)$ 进行建模，且轻边和重边使用相同的构件。它由一条包含 6 个顶点的路径组成，顶点为 x_1，x_2、...、x_6（按从一个端点到另一个端点的顺序排列）

[2] 在 [6] 这些边称为红色和蓝色。为了避免与令牌颜色混淆，我们使用“轻”和“重”这两个术语。[3] 我们使用术语 节点 指代 NCL 图中的顶点，使用术语 顶点 指代 CCTS 图中的顶点。

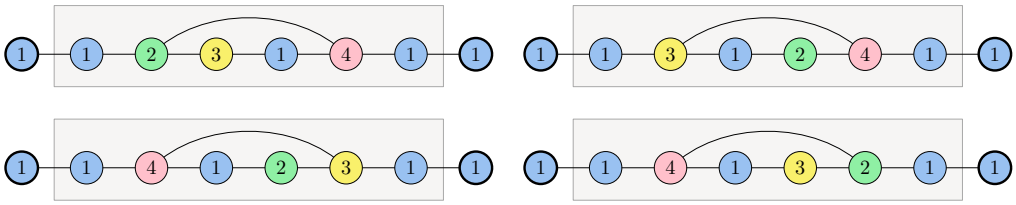


图4 从左上到右下：图3 (a) 中颜色 2 的令牌用于穿过构件时所经历的四个中间配置，这一过程会使构件翻转其方向，最终变为图3 (b) 所示的配置。为简洁起见，未显示所有交换。

再加上额外的边 (x_2, x_4)；它通过边 (u_i, x_1) 与表示节点 u 的构件相连，其中 u_i 是属于 u 构件的连接点；它还通过边 (x_6, v_j) 与表示节点 v 的构件相连，其中 v_j 是属于 v 构件的连接点。

构件的基本思想如下：令牌的排列始终定义一个方向，该方向可以朝向 u （远离 v），也可以朝向 v （远离 u）。构件的方向与 NCL配置中的边 e 的方向一致。构件无法自行改变方向，但如果它从其所指向的节点（的构件）处接收到一个颜色 2 的令牌，则可以改变方向。这样便可以执行一系列令牌交换，在将令牌移入相反节点（的构件）的同时翻转构件的方向。

当构件的方向未发生变化时，它处于图3所示的两种配置之一，二者互为镜像。从 x_1 到 x_6 读取的令牌颜色序列，要么是$\langle 1,1,3,1,4,1\rangle$，要么是$\langle 1,4,1,3,1,1\rangle$。前者对应于朝向 e 的方向，即朝向 u；后者对应于朝向 e 的方向，即朝向 v。一般而言，所有可达配置都将颜色 4 的令牌限制在 x_2 或 x_5 上。前一种情况下，我们称构件指向 v （背离 u）；后一种情况下，构件指向 u（背离 v）。

假设构件当前朝向 u，在 u_i 上放置一个颜色 2 的令牌，并在 v_j 上放置一个颜色 1 的令牌，如图3(a)所示。此时，可以通过图4中（简要）展示的交换序列反转其方向。在这一序列中，颜色 2 的令牌穿过构件，从左向右移动，从而使颜色 3 和 4 的令牌能够交换位置，最后停在顶点 v_j 上。请注意，翻转后， u_i 上会有一个颜色 1 的令牌。

一个重要的观察是，虽然颜色 2 的令牌可以从 v 一侧进入构件，但这种令牌无法穿过构件，因为它会被放置在 x_5 中的颜色 4 令牌挡住。即使有多个颜色 2 的令牌从 v 一侧进入构件，这一点仍然成立。此外，即使两个颜色 2 的令牌试图同时从 u 一侧穿过构件到达 v 一侧，构件也会按预期发挥作用。事实上，这些令牌中只有一个能到达构件的 v 一侧，而且这只有在翻转构件方向后才有可能。无论如何，另一个令牌仍会被限制在 u 一侧，位于 x_1 或 x_2 中。

当构件的方向朝向 v 时，类似的论证同样成立：从 v 一侧进入的颜色 2 令牌可以反转构件的方向；当两个这样的令牌从 u 进入，或多个颜色 2 令牌从 u 一侧进入时，构件也能正常工作。

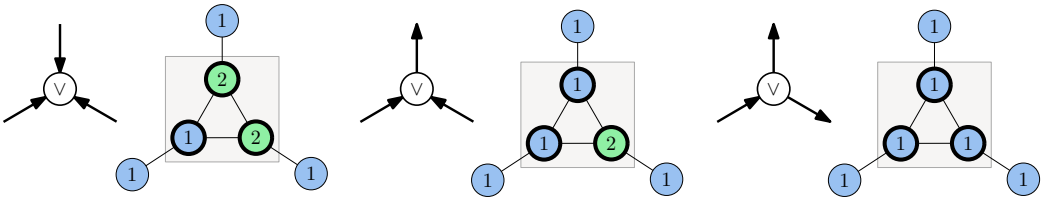


图 5 与 OR 节点相连的边的三种不同方向，以及每种情况下相应 OR 构件中的令牌配置（由灰色矩形框出）。OR 构件的所有顶点都是连接点，并以粗体显示。相邻边构件的连接点显示在灰色矩形之外。构件中颜色 2 令牌的数量才是相关因素，其所在的具体顶点并不重要。

2.2 OR 构件

OR 构件用于模拟 NCL配置中的每个 OR节点 v，由一个包含 3 个顶点的团组成 v_1，v_2，v_3；这些顶点分别充当与 v 相接的 G 的三条边的边构件的连接点。构件的顶点中只会有颜色 1 或颜色 2 的令牌，并且颜色 2 的令牌至多有 2 个。这些令牌在构件顶点中的具体位置并不重要。在构件的所有配置中，朝向 v 的、与 v 相接的边的数量至少为 $\eta + 1$，其中 η 是构件中颜色 2 令牌的数量。这确保在任何时候，至少有一条相接的边必须朝向 v（见图 5）。

如果 $\eta \geq 1$ 和连接到一般顶点 v_i 的某个边构件指向内侧，我们可以翻转该边构件：先在 v_i 放置一个颜色 2 的令牌，这最多需要与 OR 构件的某个顶点 v_j 进行一次交换，然后用该令牌穿过边构件。反过来，要翻转当前朝外且连接到顶点 v_i 的边构件，我们先确保 v_i 中有一个颜色 1 的令牌（这总能通过与某个顶点 v_j 最多进行一次交换来保证），然后穿过边构件，从而将一个颜色 2 的令牌移入 v_i。

注意，两个颜色 2 令牌可以从同一个连接点离开构件并进入同一个边构件；然而，最多只有一个令牌能够穿过该边构件（如果其当前方向允许，则通过翻转它），另一个则会被阻挡。

2.3 AND 构件

AND 构件用于对 NCL 配置的每个 AND 节点 v 进行建模。它由一条包含 6 个顶点的路径组成 v_1，$v_2, \ldots, v_6$（按从一端到另一端的顺序排列），其中 v_2 和v_4 用作与 v相连的两条轻边（的边构件）的连接点，而 v_5 则用作与 v 相连的重边（的边构件）的连接点（见图 6）。

构件中的令牌排列主要有两种（类别），我们将其称为上状态 和下状态。在上状态中，顶点 $v_1, \ldots, v_4$ 上令牌的颜色序列为$\langle 2, 4, 2, 4\rangle$，$v_6$ 上有一个颜色 3 令牌，而 v_5 上有一个颜色为{1,2}之一的令牌。图 6 (a) 和 (b) 展示了示例。在下状态中，v_1 上有一个颜色 3 令牌，v_3 和 v_5 上各有一个颜色 4 令牌，v_6 上有一个颜色 2 令牌，而v_3 和 v_4 上的令牌颜色为{1,2}之一。图 6 (c) 和 (d) 展示了这两种可能的排列。

构件的行为如下：在下状态中，与 {v1} 相连的重边（对应于边构件

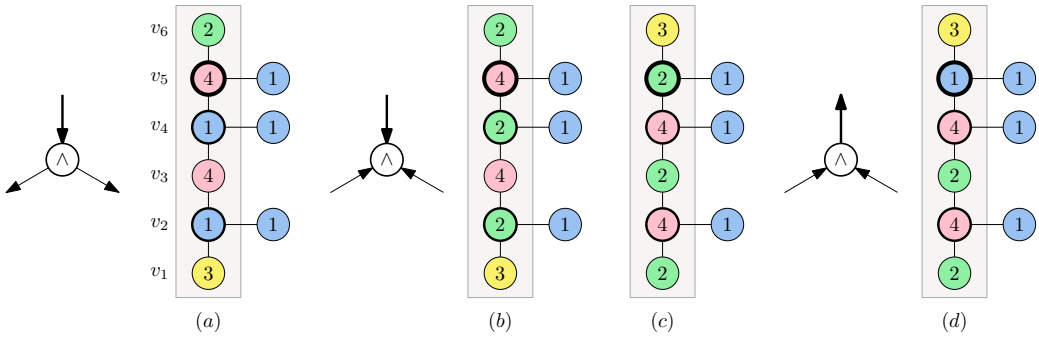


图 6 AND 构件（用灰色矩形框出）的几种可能配置，以及 NCL 实例中对应的 AND 节点和与其相连的边（显示在构件左侧）。构件的连接点以粗体显示，线条粗细表示对应的边是轻边还是重边。边构件相邻的连接点显示在灰色矩形之外。(a) 和 (b) 展示了 AND 构件处于下状态时令牌的两种可能排列。(c) 和 (d) 展示了处于上状态时的两种排列。请注意，可以通过将颜色 3 的令牌从 v_1 滑动到 v_6，从 (b) 转换到 (c)，反之亦然。

）必须朝向 v，且不能翻转，因为颜色 2 的令牌无法进入边构件。事实上，任何这样的令牌都会被顶点 v_5 中颜色 4 的令牌挡住。两条轻边可以朝任意方向，也可以自由地重新定向，前提是它们另一端的构件允许。当一条轻边朝向（分别为背离） v 时，可以在该边的连接点上放置一个颜色 2 的令牌（分别放置一个颜色 1 的令牌）。在上状态中，与 v 相连的轻边必须朝向 v，且不能翻转（因为能够到达相应边构件的颜色 2 令牌，已被顶点 v_2 和 v_4 中颜色 4 的令牌分别“锁”在顶点 v_1 和 v_3 中）。与 v 相连的重边可以朝向或背离 v，并且可以通过让颜色 2 的令牌沿该边移动来翻转该边；该令牌可以被放入顶点 v_5，也可以从中移出。

可以沿着构件中的路径，将颜色 3 的令牌从顶点 v_1 滑动到顶点 v_6，从而实现从下状态到上状态的转换。由于颜色 3 的令牌只能与颜色 2 或颜色 4 的令牌交换，因此这要求顶点 v_2 和 v_4 中含有颜色 2 的令牌，这会迫使（对应于）轻边的（边构件）朝向 v。这也会使颜色 2 的令牌位于 v_5 上。类似地，可以将颜色 3 的令牌从顶点 v_6 滑动到顶点 v_1，实现从上状态到下状态的转换。这样会锁定一个颜色 2 的令牌；该令牌必须最初放置在 v_5 中，并被锁在顶点 v_6 中，从而迫使重边朝向 v。此外，另外两个颜色 2 的令牌会从 v_1 和 v_3 移动到轻边的连接点。

请注意，虽然构件可能处于既非上状态也非下状态的中间状态，但让构件中的令牌保持这种排列从来都不是明智之举。事实上，在任何这种配置中，顶点 v_6 上仍有一个颜色 2 的令牌，而且还会阻止翻转一条或两条轻边；也就是说，它比下状态限制更多。

2.4 组合构件

CCTS实例的初始配置和最终配置分别通过将初始配置和最终配置中的每条边、与顶点和或顶点替换为

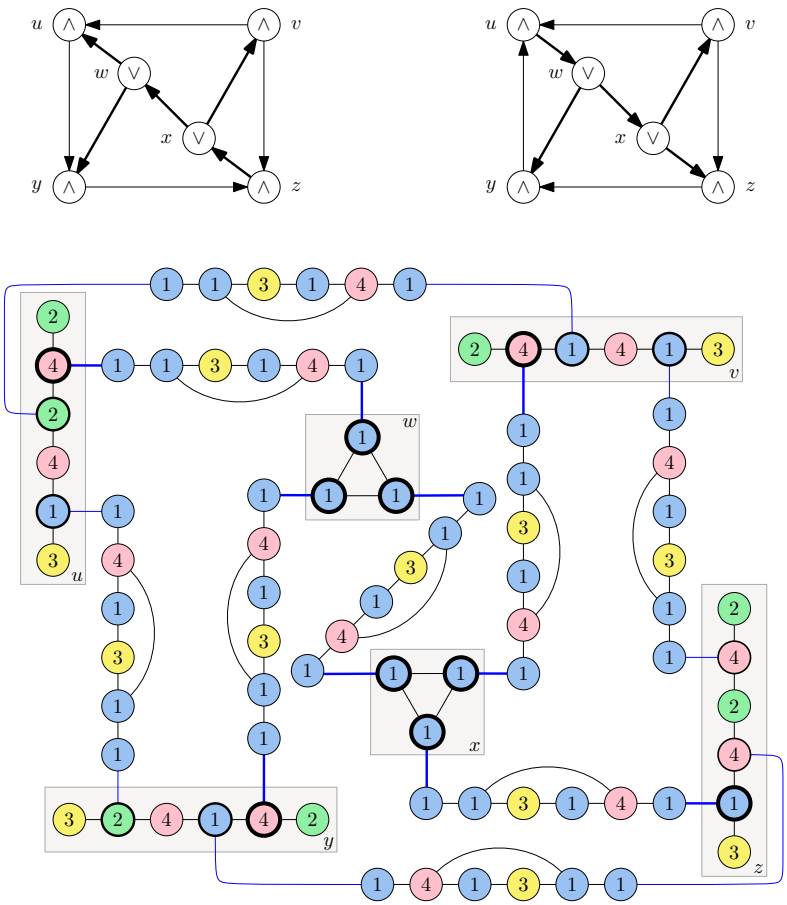


图7 左上和右上分别为 NCL 实例的初始配置和最终配置。重边以较粗的线条表示。底部：将我们的归约应用于 NCL 初始配置后得到的 CCTS 实例的初始配置。AND 和 OR 构件分别位于灰色矩形中，矩形标有 NCL 图中对应节点的名称；连接 AND/OR 构件连接点与边构件连接点的边以蓝色显示，其粗细表示 NCL 实例中对应的边是轻边还是重边。

上文所述的相应构件来获得，其中令牌的适当排列用于反映NCL实例中的边方向。图7展示了一个NCL实例，以及将我们的归约应用于初始NCL配置后得到的CCTS配置。图7的交互式版本可在`https://www.isnphard.com/g/constrained-colored-token-swapping/`查看。

由于 NCL 实例的图是平面图，且每个构件都是外平面图，因此得到的图也是平面图。此外，很容易验证 CCTS 实例中的每个顶点的度至多为 3。现在我们证明，CCTS 实例可解当且仅当 NCL 实例可解，从而确立了具有 4种颜色的 CCTS 的 PSPACE 完全性。[4]

[4] CCTS 属于 PSPACE 已在 [21]中得到证明。

NCL 的解意味着 CCTS 的解。假设 NCL 实例可解，并设 $\langle e_1, e_2, \ldots, e_k \rangle$ 是一系列需要翻转的边，它们将初始配置转换为最终配置，同时确保所有中间配置都有效。设 G_ℓ 为执行前 ℓ 次边翻转后得到的有向图，则 G_0 （分别为 G_k）与初始（分别为最终）NCL 配置一致。

我们可以按照相同的顺序模拟每一次边翻转，从而求解该 CCTS实例。更准确地说，在第 ℓ 次边翻转之前，CCTS 配置将与直接从 $G_{\ell-1}$ 应用我们的归约所得到的配置完全相同；我们将展示一系列令牌交换，将该配置转换为对 G_k 应用我们的归约所得到的配置。为描述第 ℓ 次边翻转的实现方式，设 u 和 v 分别为 e_ℓ 在 $G_{\ell-1}$ 中的头节点和尾节点，因此需要重新定向 e_ℓ，使其从指向 u 改为指向 v。设 u_i （分别为 v_j）是 u 的构件（分别为 v 的构件）中的连接点，该连接点与 e_ℓ（的边构件所建模的某个连接点）相 incident。我们分多个步骤进行，每个步骤可以包含零次、一次或多次令牌交换。

第一步是将最初位于 u构件某个顶点上的一个颜色 2 令牌，放到顶点 u_i上（如果它尚未在那里）。如果 u 是 OR 节点，那么 e_ℓ 是一条轻边，并且 u 在 $G_{\ell-1}$ 中除 e_ℓ之外至少还有一条入边。于是， u的构件中至少有一个颜色 2 令牌，它要么已经位于 u_i，要么可以通过一次交换移动到 u_i。否则， u 是 AND 节点，我们根据 e_ℓ是轻边还是重边采取不同的处理方式。如果 e_ℓ 是轻边，那么 u 在 $G_{\ell-1}$中有一条入重边，而 u的构件处于下状态的 AND 构件。此时，顶点 x 上已经有一个颜色 2 令牌，无需交换。如果 e_ℓ 是重边，那么 u 在 $G_{\ell-1}$、 $u_i = u_5$处有两条入轻边，并且 u的构件处于下状态，且 u_2 和u_4上各有一个颜色 2 令牌。我们将 u构件中的颜色 3 令牌滑动，使其转为上状态，从而将一个颜色 2 令牌放到 u_5上。

第二步将表示 e_ℓ 的边构件的方向从指向u 翻转为指向 v。具体做法是使用颜色 2 的令牌 t，它此时位于u上对应 e_ℓ的连接点，穿过 e_ℓ的构件。因此， t 从 u_i 移至 v上对应 e_ℓ的连接点，也就是移至 v_j。

第三步也是最后一步，可以重新排列 v的构件中的令牌；仅当v 是 AND 节点且 e_ℓ 是重边时才需要这一步。在这种情况下， e_ℓ 在 $G_{\ell-1}$中背离 v，这意味着与 v 相连的两条轻边都朝内。此时， v的构件处于上状态；在第二步之后，顶点 v_5 上有一个颜色 2 的令牌。我们通过将颜色 3 的令牌从 v_6 滑动到 v_1，使构件转为下状态。

CCTS 的解意味着 NCL 的解。现在假设该 CCTS实例可解。我们证明，这意味着存在一个边翻转序列，可以解出该 NCL 实例。具体而言，我们考察一个解出该 CCTS实例的交换序列；每当边构件中的颜色 3 和 4 的令牌发生交换时，我们就相应地翻转 NCL 实例中的对应边，使其方向与构件的新方向一致。我们声称，在这样的边翻转序列中遇到的所有 NCL 配置都是有效的。由于边构件的方向始终与对应 NCL 边的方向一致，这意味着执行所有边翻转后得到的配置，恰好就是所求的 NCL 实例的最终配置。

给定任意 CCTS 配置、一个节点 w 以及一条与 w 相接的边 e，我们将 w_e 表示为 w 的构件中与 e 的构件的某个连接点相连的连接点。我们将 w_e 的扩展版本定义为包含 w_e 及以下所有顶点的顶点集

e 的构件中可从 w_e 到达且无需经过带有颜色 4 的令牌的所有顶点。

考虑一个一般的交换操作：在表示某条边 $e = (u, v)$ 的边构件中，交换颜色 3 和颜色 4 的令牌。假设（不失一般性） e 的构件在交换前指向 u，交换后转而指向 v。要改变 e 的构件方向，必须与颜色 2 的令牌交互；由于这次翻转，该令牌从 u_e 的扩展版本移至 v_e 的扩展版本。

如果 u 是 OR 节点，那么在交换前， u 的构件至少有两个相邻边构件指向 u （见第 2.2 节）。这意味着翻转 e 后仍会得到一个 NCL 配置，其中 u 有一条入边，因而该配置是有效的。

如果 u 是 AND 节点，我们根据 e 是轻边还是重边，分两种情况讨论。如果 e 是轻边，那么 u_e 是 u_2 或 u_4，而 e 的构件之所以能够翻转，最初必定是因为有某个颜色 2 的令牌进入了 u_e 的扩展版本。这意味着翻转后立即， u 的构件不可能处于上状态，因此它在 u_6 中包含一个颜色 2 的令牌。只有当连接到 v_5、代表一条重边的边构件指向 u 时，这种情况才可能发生。于是，在得到的 NCL 配置中，节点 u 有一条重边入边，因此该配置有效。如果 e 是重边，那么 u_e 是 u_5；要使 e 的构件从指向 u 翻转为指向 v，唯一的方式是将一个颜色 2 的令牌从 u_e 的扩展版本移至 v_e 的扩展版本。只有当 u 的构件中的颜色 3 令牌放置在 v_6 时，这种情况才可能发生，也就是说，只有当 u 的构件处于上状态时才可能发生。这意味着与 u 相邻的两条轻边的边构件都指向 u。因此，交换后，节点 u 有两条轻边入边，这说明得到的 NCL 配置有效。

将结果推广到三次平面基图。以上论证表明，当交换图是包含 4 个顶点的路径，且基图是平面次三次图时，CCTS 是 PSPACE 完全的。通过在边构件和 AND 构件中添加虚拟顶点和边，对其进行“填充”，即可将上述结果推广到三次图。这些虚拟顶点和边不会改变构件的作用，但能确保所有顶点的度恰好为 3。[5] 图 8 (a) 和 (b) 分别展示了两种虚拟顶点配置，可将顶点的度从 1 提高到 3，以及从 2 提高到 3。图 8 (c) 和 (d) 分别展示了填充后的边构件和 AND 构件。请注意，虽然颜色 2 的令牌可以与额外添加的颜色 1 虚拟顶点交换，但这毫无用处，因为这种操作不会使任何涉及非虚拟顶点的额外交换成为可能；而处于虚拟顶点中的颜色 2 令牌只能返回其离开时所在的同一个非虚拟顶点。填充后的构件不再是外平面的，但仍然是平面的，而且所有连接点都可以位于嵌入的外面上（如图 8 的嵌入所示），因此最终基图仍保持平面性。

我们已经确立了本节的主要结果，陈述如下：

▶**定理 1.** 即使将 *CCTS* 问题限制在交换图为含有4 个顶点的路径、且基图为三次平面图的情形下，该问题仍是PSPACE-完全的。

3 星形图的多项式时间算法

本节介绍一种多项式时间算法，用于求解交换图为星形图的 CCTS 问题，简称 CCTS* 。这也表明

[5] 无需特别处理或门构件，因为其中所有顶点的度数本来就是 3。

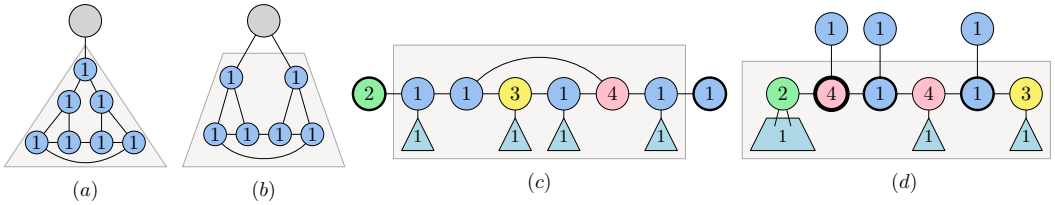


图 8 (a) 和 (b)：两种由度为 3、包含颜色 1 令牌的虚拟顶点构成的配置，它们可将顶点 v （灰色）的度从 1（在 (a) 中）或 2（在 (b) 中）增加到 3。(c) 和 (d)：分别对图3 (a) 所示配置中的边构件，以及图 6 (a) 所示配置中的 AND 构件进行填充后的版本。这里，三角形表示按 (a) 所示方式由虚拟顶点构成的填充，梯形则表示按 (b) 所示方式进行的填充。

当颜色数为 3 时，CCTS* 是多项式时间可解的，因为三顶点上唯一的连通非完全交换图是星形图。不失一般性，我们采用以下假设：(i) 交换图中心的颜色为 1；(ii) 在 CCTS* 的初始配置和最终配置中，颜色为 1 的令牌占据相同的顶点集。事实上，如果条件 (ii) 尚未满足，可以先不考虑其他令牌的位置，将颜色为 1 的令牌移动到其最终位置，然后考虑将所得配置转变为目标最终配置的实例。由于交换操作可逆，这不会将一个是实例变成否实例，反之亦然。

注意，颜色 1 的令牌可以与任何其他颜色的令牌交换，而除此之外不可能发生其他交换。因此，这类实例也可以等价地视为图上的滑动令牌问题，其中被颜色 1 的令牌占据的顶点视为空白，即不含令牌。这样一来，剩下的令牌颜色均不为 1，一步操作就是将任意此类令牌沿一条边从其当前顶点滑动到相邻的空白顶点。该问题要求判断：给定一个初始配置和一个最终配置，能否通过一系列操作从前者到达后者。当这类实例中的所有令牌颜色各不相同时，这正是图上的鹅卵石移动问题 (PMG)，它是对含有一个或多个空位的 15数码问题的图推广，并由 Kornhauser、Miller 和 Spirakis 进行了研究，见 [13].[6]

特别地，[13] 给出了 PMG 可解实例的刻画，可在多项式时间内检验。显然，可以通过测试所有可能的 保色双射，来判断一个 CCTS* 实例是否可解；这些双射将初始配置中颜色为 $c \neq 1$ 的每个令牌映射到最终配置中颜色为 c 的令牌。每个这样的保色双射都对应一个 PMG 实例，其可解性可利用 [13]的结果进行检验。然而，在最坏情况下，这种朴素方法可能需要测试超指数数量的双射。接下来，我们将扩展 [13] 针对 PMG 采用的技术，以在多项式时间内求解 CCTS* 。

我们重新采用 [13], 这一关键要素： $\sim$ 这是一种等价关系， $\sim$ 允许我们将 CCTS* 实例中被令牌占据的顶点划分为等价类。这些等价类仅取决于哪些顶点包含颜色不为 1 的令牌，而不取决于具体的令牌颜色。令 $\mathcal{B} \subseteq V(G)$ 表示图$G = (V, E)$中的空白顶点集，并在不属于 $\mathcal{B}$ 的顶点上定义关系 $\sim$如下：对于 $u, v \in V \setminus \mathcal{B}$，若存在一系列交换，将最初位于顶点 u 的令牌移动到顶点 v，则称顶点 $u \sim v$ 与之相关，即

[6] 许多技术细节和证明在 [13] 中省略了，可在 Kornhauser 的硕士论文 [14], 中找到； [13] 正是基于该论文。

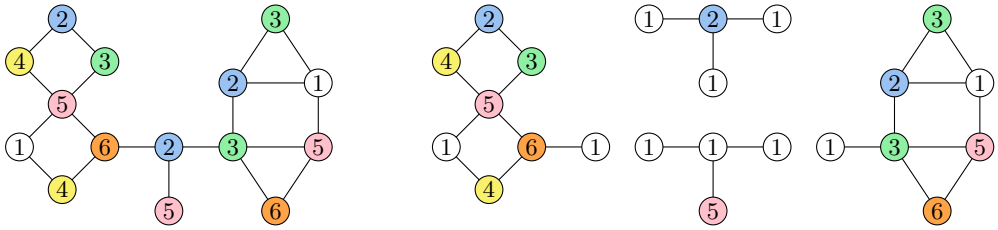


图 9 (a) CCTS 的一个实例 $\mathcal{I}^\star$ 的初始配置（交换图的中心颜色为 1）。(b) $\mathcal{I}$ 的传递子实例 $\mathcal{I}_C$ 的初始配置。

最初位于 u 的令牌可以在不改变最终空白顶点集的情况下移动到顶点 v。此外，对于顶点 $u \in V \setminus \mathcal{B}$，如果存在一系列交换，将最初位于 u 的令牌移动到顶点 v（不一定保持所得配置中的空白顶点集），则称顶点 $v \in V$ 可由 u 到达。我们用 $R(u)$ 表示所有可由 u 到达的顶点 v 所组成的集合；对于顶点集 $U \subseteq V \setminus \mathcal{B}$，令 $R(U) = \bigcup_{u \in U} R(u)$。

▶ **引理 2.** $\sim$ 是 $V \setminus B$ 上的等价关系。

证明。 显然，$u \sim u$ 对于所有 $u \in V \setminus \mathcal{B}$，只需考虑空交换序列，因此 $\sim$具有自反性。

要证明 $\sim$ 具有对称性，即 $u \sim v$ 蕴含 $v \sim u$，设 A 为令牌的初始排列，考虑一个交换序列 S，它将 u 中的令牌移至 v，同时保持空白顶点集合不变；设 B 为由此得到的令牌排列。具体而言，我们可以将 S 视为一列边（边可能重复），其中每条边描述了对其两个端点上的令牌进行交换，且两个端点中恰有一个为空白。设 S^{-1} 为将 S中的边顺序反转后得到的序列。若将 S^{-1} 应用于 B，则会得到配置 A。因此，从配置 A 应用 S^{-1}（这是可行的，因为空白顶点集合相同），会将 v 上的令牌移至顶点 u，同时保持空白顶点集合不变。

最后，我们证明 $\sim$ 是传递的。若 $u \sim v$ 和 $v \sim w$，则令 S_1（分别为 S_2）表示一个交换序列，它将 u（分别为 v）中的令牌移动到顶点 v（分别为 w），且不改变空白顶点集合。将 S_1 与 S_2 连接得到的序列，会把初始位于 u的令牌移至 w，且不改变空白顶点集合。◀

令 $\mathcal{C} = (V \setminus \mathcal{B})/\sim$ 为 $\sim$的等价类集合。对于每个等价类 $C \in \mathcal{C}$，我们考虑由 $C \cup R(C)$中的顶点所诱导的 G 的子图 $G_C = (V_C, E_C)$，并定义 CCTS* 的一个子实例 $\mathcal{I}_C$，其中交换图保持不变。在初始（分别为最终）配置中，C 中的顶点所含令牌的颜色与 CCTS* 实例的初始（分别为最终）配置中相同。在初始和最终配置中，$R(C) \setminus C$ 中的顶点都含有颜色为 1 的令牌。示例见图 9。请注意，所得实例的初始配置和最终配置中，同一颜色的令牌数量可能不同；在这种情况下，该实例显然是一个否实例。

如果 CCTS* 只有一个关于$\sim$的等价类，我们就称该实例为传递的。[13] 展示了如何在多项式时间内计算$\mathcal{C}$ 以及所有 $R(C)$（对于 $C \in \mathcal{C}$），并证明了这样一些实例是传递的：C 中的顶点不是空白，而 $R(C) \setminus C$ 中的顶点是空白。于是，我们有以下结论：

▶ **引理 3** ([13]). 对于所有等价类 $C \in \mathcal{C}$，实例 I_C 是传递的。

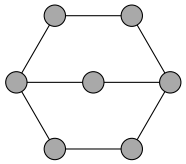


图10 定理6和引理 7的陈述中提到的图 T_0 。

如上所述，每当一个 CCTS* 实例 $\mathcal{I}$ 对除 1 以外的每种颜色恰好有一个令牌时，$\mathcal{I}$ 就等价于一个 PMG 实例。因此，它的传递子实例 $\mathcal{I}_C$ of $\mathcal{I}$ 恰好是 [13]为 PMG 计算出的那些实例，据此我们将以下结果调整为本文的术语。

▶**引理4** ([13], 改述)。 考虑一个 *CCTS* 的传递实例 $I^\star$ ，其中除颜色1外，每种颜色恰有一个令牌；令 C 为其关于 $\sim$的等价类集合。当且仅当所有实例 I_C（其中 $C \in C$）均可解时，实例 I 可解。

我们可以证明，类似的结论适用于所有 CCTS 实例。

▶**引理5**。 考虑一个 *CCTS* 的传递实例 $I^\star$ ，令 C 为其关于 $\sim$的等价类集合。当且仅当所有子实例 I_C（其中 $C \in C$）均可解时，实例 I 可解。

证明。假设 I 可解，考虑一个 S 求解 I 的交换序列，并定义由 S诱导的保色双射 μ 。更确切地说，如果交换序列 S 将一个令牌从初始配置中的顶点 u 移至最终配置中的顶点 v ，那么μ 就将初始配置中位于 u的令牌映射到最终配置中位于 v 的令牌。双射 μ 诱导出一个实例 I' ，其中除颜色 1 外，每种颜色恰有一个令牌；该实例可由序列 S求解。于是，引理4表明，I' 的所有传递子实例 I'_C 均可解。接着，由于对于每个等价类 $C \in \mathrm{C}$，实例 I'_C 是由 I_C的一个保色双射诱导的，因此 I'_C 可解意味着 I_C 也可解。

现在假设对于 $C \in \mathcal{C}$ 的所有子实例 $\mathcal{I}_C$ ，它们都是可解的。设 μ_C 是由$\mathcal{I}_C$ 的一个解所诱导的保颜色双射，并设 $\mathcal{I}'_C$ 是对应的 CCTS* 实例，其中除颜色 1 外的每种颜色恰有一个令牌（该实例是可解的）。考虑函数 μ ，它将一个一般令牌 t（最初位于属于等价类 C 的顶点上）映射到 $\mu_C(t)$。由于 $\mu_C(t)$ 必须放置在 C 中的某个顶点上，函数 μ 是一个双射，并且按构造保颜色。

考虑由 μ 诱导的实例 $\mathcal{I}'$ 。根据构造，实例 $\mathcal{I}'_C$ 恰好是$\mathcal{I}'$ 的传递子实例。因此，根据引理4，$\mathcal{I}'$ 是可解的，这意味着$\mathcal{I}$ 也是可解的。◀

3.1 求解传递实例

在 [13]，中，作者给出了 PMG 的可解传递实例的一种刻画，该刻画可在多项式时间内检验。我们将这种刻画改述为这样的 CCTS*实例：除颜色 1 外，每种颜色恰有一个令牌。

▶**定理6** ([13], 改述)。 考虑一个传递的 *CCTS* 实例 $I^\star$ ，其中每种颜色（除1 之外）恰有一个令牌，且其基图 G 既不是环，也不是图T_0 图*10*中的图。此类实例可在多项式时间内求解；当且仅当以下条件全部成立时，它是一个否实例：*(i)* 颜色为1的令牌恰有一个，

(ii) G 是双连通二分图；*(iii)* 由令牌的初始位置和最终位置所确定的置换是奇置换。[7]

利用上述刻画，我们可以给出以下判定准则，用于判断多个令牌可能具有相同颜色时的传递 CCTS* 实例是否可解。

▶**引理 7.** 考虑一个传递的 *CCTS** 实例 I，其基图 G 既不是环，也不是图*1*中的图 T_0 。此类实例可在多项式时间内求解；当且仅当以下条件全部成立时，它是一个否实例：*(i)* 所有令牌的颜色互异；*(ii)* G 是双连通且二分的；*(iii)* 由令牌初始位置和最终位置所诱导的置换（由于条件 *(i)* 而定义良好）是奇置换。

证明。 根据定理6，我们只需证明，若 $\mathcal{I}$ 中包含两个或更多个颜色相同的令牌 $c^* > 1$ 那么它是一个是实例。

我们定义两个保色双射 μ'，μ''，将初始配置和最终配置中的令牌对应起来。第一个双射 μ' 可以任意选取（其存在性由初始配置和最终配置中相同颜色的令牌数量相等保证）。令 T 表示初始配置中颜色不为 1 的令牌集合。第二个双射 μ'' 定义如下：对所有 $t \in T \setminus \{t_1, t_2\}$，$\mu''(t) = \mu'(t)$，$\mu''(t_1) = \mu'(t_2)$，且 $\mu''(t_2) = \mu'(t_1)$。

双射 μ'（分别为 μ''）给出了 CCTS 的一个实例$\mathcal{I}'$（分别为$\mathcal{I}''$），其中所有颜色不为 1 的令牌颜色互异。

由于$\mathcal{I}'$ 和$\mathcal{I}''$ 中令牌的初始位置与最终位置所诱导的置换奇偶性不同（因为两者相差一次对换），因此至少有一个这样的实例$\mathcal{I}^* \in \{\mathcal{I}',\mathcal{I}''\}$ 不满足定理6的条件 (iii)，因而是可解的。

注意到，由于 μ' 和 μ'' 保持颜色不变，任何能解$\mathcal{I}^*$ 的序列也能解$\mathcal{I}$，由此即可得出该结论。◀

由于基图为环或 T_0 的 CCTS 实例可以在多项式时间内求解，[8] 引理 7 的刻画蕴含以下结论。

▶**定理 8.** *CCTS* 问题可以在多项式时间内求解。

4 结论

在本文中，我们通过证明当颜色数量从 3 增加到 4 时，CCTS 的复杂度会发生剧烈变化（假设 P $\neq$ PSPACE），解决了 [21] 中遗留的问题。具体而言，我们证明了当交换图包含一个含 4 个顶点的诱导路径时，该问题是 PSPACE-困难的。这留下了这样一个问题：当交换图为 P_4-自由时，如何刻画 CCTS 的计算复杂度。朝这一方向迈出的一个有希望的第一步，或许是研究完全二分交换图；对于这类图，我们相信该问题可以在多项式时间内求解。

[7] 每个置换 π 都可以写成若干对换的乘积，也就是说，可以写成交换两个不同元素的乘积。更准确地说，每个置换 π 都可以写成偶数个对换的乘积，或者奇数个对换的乘积，但不能两者兼是；π 的奇偶性 定义为这类对换数量的奇偶性。[8] 当基图为 T_0时，可以通过暴力搜索求解该实例。当基图为一个环时，当且仅当最终配置保持除 1 以外的令牌颜色的循环序列不变，才可达。

参考文献

1 Oswin Aichholzer、Erik D. Demaine、Matias Korman、Anna Lubiw、Jayson Lynch、Zuzana Masárová、Mikhail Rudoy、Virginia Vassilevska Williams 和 Nicole Wein。树上的令牌交换问题的困难性。载于 Shiri Chechik、Gonzalo Navarro、Eva Rotenberg 和 Grzegorz Herman（编），第30届欧洲算法年会，ESA 2022，德国柏林/波茨坦，2022年9月5日至9日，LIPIcs 第244卷，第3:1–3:15页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2022年。URL：https://doi.org/10.4230/LIPIcs.ESA.2022.3，doi:10.4230/LIPICS.ESA.2022.3。 2 Davide Bilò、Maurizio Fiusco、Luciano Gualà 和 Stefano Leucci。交换混杂啤酒以保持其凉爽。载于 Andrei Z. Broder 和 Tami Tamir（编），第12届算法趣味国际会议，FUN 2024，意大利撒丁岛拉马达莱娜岛，2024年6月4日至8日，LIPIcs 第291卷，第5:1–5:18页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2024年。URL：https://doi.org/10.4230/LIPIcs.FUN.2024.5，doi:10.4230/LIPICS.FUN.2024.5。 3 Édouard Bonnet、Tillmann Miltzow 和 Pawel Rzazewski。令牌交换及其变体的复杂度。Algorithmica，80(9)：2656–2682，2018年。URL：https://doi.org/10.1007/s00453-017-0387-0，doi:10.1007/S00453-017-0387-0。 4 Arthur Cayley。LXXVII。关于置换理论的注记。《伦敦、爱丁堡和都柏林哲学杂志与科学杂志》，34(232)：527–529，1849年。 5 Colin Defant 和 Noah Kravitz。图上的朋友与陌生人行走。Combinatorial Theory，1，2021年。 6 Robert A. Hearn 和 Erik D. Demaine。通过非确定性约束逻辑计算模型证明滑块拼图及其他问题的 PSPACE 完全性。《理论计算机科学》，343(1-2)：72–96，2005年。URL：https://doi.org/10.1016/j.tcs.2005.05.008，doi:10.1016/J.TCS.2005.05.008。 7 Lenwood S. Heath 和 John Paul C. Vergara。通过短交换进行排序。《J. Comput. Biol.》，10(5)：775–789，2003年。 doi:10.1089/106652703322539097。 8 Sam Hiken 和 Nicole Wein。令牌交换近似困难性的改进。载于 Anne Benoit、Haim Kaplan、Sebastian Wild 和 Grzegorz Herman（编），第33届欧洲算法年会，ESA 2025，波兰华沙，2025年9月15日至17日，LIPIcs 第351卷，第57:1–57:16页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik，2025年。URL：https://doi.org/10.4230/LIPIcs.ESA.2025.57，doi:10.4230/LIPICS.ESA.2025.57。 9 Ryan Jeong。二分图上的二分朋友与陌生人行走。《Discret. Math.》，349(1)：114667，2026年。URL：https://doi.org/10.1016/j.disc.2025.114667，doi:10.1016/J.DISC.2025.114667。 10 Mark Jerrum。寻找最短生成元序列的复杂度。《理论计算机科学》，36：265–289，1985年。doi:10.1016/0304-3975(85)90047-7。 11 Sheldon B. Akers Jr. 和 Balakrishnan Krishnamurthy。对称互连网络的群论模型。《IEEE Trans. Computers》，38(4)：555–566，1989年。 doi:10.1109/12.21148。 12 Jun Kawahara、Toshiki Saitoh 和 Ryo Yoshinaka。通过匹配、令牌交换及一种变体进行置换路由的时间复杂度。《J. Graph Algorithms Appl.》，23(1)：29–70，2019年。URL：https://doi.org/10.7155/jgaa.00483，doi:10.7155/JGAA.00483。 13 Daniel Kornhauser、Gary L. Miller 和 Paul G. Spirakis。协调图上的卵石移动、置换群的直径及其应用。载于第25届计算机科学基础年会，美国佛罗里达州西棕榈滩，1984年10月24日至26日，第241–250页。IEEE计算机学会，1984年。 doi:10.1109/SFCS.1984.715921。 14 Daniel Martin Kornhauser。协调图上的卵石移动、置换群的直径及其应用，1984年。硕士论文。 15 Tillmann Miltzow、Lothar Narins、Yoshio Okamoto、Günter Rote、Antonis Thomas 和 Takeaki Uno。令牌交换的近似与困难性。载于 Piotr Sankowski 和 Christos D. Zaroliagis（编），第24届欧洲算法年会，ESA 2016，丹麦奥胡斯，2016年8月22日至24日，LIPIcs 第57卷，第66:1–66:15页。Schloss Dagstuhl

- 莱布尼茨信息学中心，2016。URL: https://doi.org/10.4230/LIPIcs.ESA.2016.66, doi:10.4230/LIPICS.ESA.2016.66. 16 Caio Henrique Segawa Tonetti、Vinícius Fernandes dos Santos 和 Sebastián Urrutia。余图上令牌交换问题的多项式时间算法。RAIRO 运筹学，58(1):441–455，2024。URL: https://doi.org/10.1051/ro/2023134, doi:10.1051/RO/2023134. 17 Theresa P Vaughan。扫帚图上的置换分解。组合数学与组合计算杂志，30:129–148，1999。18 Richard M Wilson。图谜题、同伦与交错群。组合理论杂志 B 系列，16(1):86–96，1974。19 Katsuhisa Yamanaka、Erik D. Demaine、Takehiro Ito、Jun Kawahara、Masashi Kiyomi、Yoshio Okamoto、Toshiki Saitoh、Akira Suzuki、Kei Uchizawa 和 Takeaki Uno。图上的带标签令牌交换。理论计算机科学，586:81–94，2015。URL: https://doi.org/10.1016/j.tcs.2015.01.052, doi:10.1016/J.TCS.2015.01.052. 20 Katsuhisa Yamanaka、Takashi Horiyama、J. Mark Keil、David G. Kirkpatrick、Yota Otachi、Toshiki Saitoh、Ryuhei Uehara 和 Yushi Uno。图上的彩色令牌交换。理论计算机科学，729:1–10，2018。URL: https://doi.org/10.1016/j.tcs.2018.03.016, doi:10.1016/J.TCS.2018.03.016. 21 Chao Yang 和 Zhujun Zhang。朋友与陌生人问题是 PSPACE 完全的。信息处理快报，190:106573，2025。URL: https://doi.org/10.1016/j.ipl.2025.106573, doi:10.1016/J.IPL.2025.106573. 22 Gaku Yasui、Kouta Abe 和 Katsuhisa Yamanaka。完全分裂图上的带标签令牌交换。SIG技术报告，115(84):87–90，2015。