arXiv:2302.01405v1 [cs.CC] 2 Feb 2023

无限移动次数下单人国际象棋的复杂性

Josh Brunner* Lily Chung* MichaelCoulombe* ErikD. Demaine*Timothy Gomez* Jayson Lynch*

摘要

我们分析单人国际象棋谜题：输入是一个 $n \times n$ 棋盘，上面有若干颜色相同的标准国际象棋棋子，目标是通过一系列吃子移动，最终只剩下一枚棋子。以往的研究分析了单一棋子类型的这一谜题，其中每枚棋子最多只能进行两次吃子移动（如 chess.com 上的单人国际象棋谜题）。与之不同的是，我们研究每枚棋子都可以进行不限次数吃子移动的情形。我们证明，在一般的棋子类型模型中，任意单一棋子类型都能在多项式时间内求解，而任意两种标准国际象棋棋子的问题均为 NP 完全。我们还分析了一项限制（如 chess.com 上的谜题）：一种棋子类型仅有一枚，且必须成为最后存活的棋子。我们证明，在这种情况下，某些棋子类型组合变得容易求解，而另一些仍然是困难的。

1 引言

经典双人游戏国际象棋是 PSPACE完全 [Sto83] 或 EXPTIME完全 [FL81]，具体取决于移动次数是否被限制为多项式。近期研究分析了基于国际象棋的谜题，包括 PSPACE完全的助杀国际象棋和逆向国际象棋[BDHW20]，以及 NP 完全的单人国际象棋 [AMM22]。本文将单人国际象棋的分析扩展到每个棋子的移动次数不受限制的情况。

首先，我们回顾 Chess.com 上实现的标准单人国际象棋 [Che]。所有棋子颜色相同，可以吃掉除王以外的任何棋子。每次移动都必须吃子。目标是找到一系列移动（吃子），使棋盘上最后只剩一个棋子。如果棋盘上有王，它必须是最后剩下的棋子。此外，每个棋子最多可以移动 $k = 2$ 次。先前的研究 [AMM22] 将单人国际象棋推广到每个棋子移动次数任意受限的情况 k （棋盘大小也任意），并将这种游戏记作 (Generalized) Solo-Chess(S, k)，其中 S 是允许使用的棋子类型集合。他们证明， Solo-Chess({♟}，2) 和Solo-Chess(S, 1) 可在线性时间内求解，而 Solo-Chess({♝}，2)、 Solo-Chess({♜}， 2) 和 Solo-Chess({♛}，2) 是 NP 完全的。

本文分析了不限制每种棋子可移动次数，或等价地说，每种棋子的移动上限大于棋子数量时，单人国际象棋谜题的复杂度。我们将这种游戏记为 Solo-Chess(S)，这等价于 Solo-Chess(S, ∞)。我们还考虑了有无单个不可被吃的王这一限制的两种游戏：用^1 表示带有此限制的游戏，用它表示允许有多个可被吃的王的更一般情形。我们还将这一概念扩展到任意棋子类型 T 的 T^1，意即存在一个类型为 T 的不可被吃的棋子（因此它必须是最后留在棋盘上的棋子）。

*计算机科学与人工智能实验室，麻省理工学院，美国马萨诸塞州剑桥市 02139， {brunnerj,lkdc,mcoulomb,edemaine,dylanhen,tagomez7,jaysonl}@mit.edu

	♟	♚	♞	♜	♝	♛
♟[1]	P, 定理 2.2	P, 2.16	NP-c, 定理 3.9	OPEN	P, 定理 2.16	P, 定理 2.16
♚[1]	, 定理 3.7	P, 定理 2.2	NP-c, 定理 3.9	OPEN	NP-c, 定理 3.10	P, 定理 2.16
♞[1]	, 定理 3.1 NI	, 定理 3.1	P, 定理 2.2	NP-c, 定理 3.11 NI	, 定理 3.11 NI	, 定理 3.11
♜[1]	, 定理 3.2 N	, 推论 3.3	NP-c, 推论 3.3	P, 定理 2.2	NP-c, 推论 3.4	P, 定理 2.16
♝[1]	NP-c, 推论 3.5	NP-c, 推论 3.5	NP-c, 推论 3.5	NP-c, 推论 3.6	P, 定理 2.2	P, 定理 2.16
♛[1]	NP-c, 推论 3.4	NP-c, 推论 3.4	NP-c, 推论 3.4	NP-c, 推论 3.6	NP-c, 推论 3.4	P, 定理 2.2

表 1：总结了我们关于单一国际象棋棋子类型的结果（见对角线），以及关于两种国际象棋棋子类型的结果；其中一种（左侧的行标签）被限制为不可吃棋子，且为最后一个棋子。

我们的结果。我们证明，对于任意一种单一标准国际象棋棋子类型（$|S| = 1$），Solo-Chess(S) 都可以在多项式时间内求解。事实上，对于第 2.1 节中定义的一般棋子类型模型，这一结果同样成立。对于任意两种不同的标准国际象棋棋子类型（$|S| = 2$），且两者都不是不可被吃的棋子，我们通过多种归约证明 Solo-Chess(S) 是 NP 完全的。（本文讨论的所有问题都显然属于 NP。）

对于唯一的不可被吃的王 ♚[1]，我们证明，$S = \{♚^1, ♛\}$ 实际上可以在多项式时间内求解，根本原因是王的走法是后的走法的子集。另一方面，$S = \{♚^1, ♟\}$，$\{♚^1, ♞\}$，$\{♚^1, ♝\}$ 都是 NP 完全的。我们还针对若干其他形式为 $\{S^1, T\}$ 的棋子对给出了多项式时间算法和 NP 完全性结果，其中 S^1 不可吃；关于仅限于标准国际象棋棋子的结果，请参见表 1。

本文分为算法结果和困难性结果两部分。第 2 节介绍我们针对单一棋子类型（第 2.2 节）以及某些由可吃与不可吃棋子类型组成的棋子对（第 2.3 节）所得到的算法结果；这两部分结果都适用于第 2.1 节定义的一种非常一般的棋子类型模型。第 3 节介绍我们的困难性结果，主要分为两类：从哈密顿路径进行的归约（第 3.1 节）和从 SAT 进行的归约（第 3.2 节）。第 4 节最后提出一些开放问题。

2 算法结果

本节将介绍几种单人国际象棋情形的多项式时间算法。首先，在第 2.1 节中，我们定义一种广义的抽象移动概念，其中棋子必须始终吃掉另一枚棋子。接着，在第 2.2 节中，我们证明，在这个一般化游戏中，任何单一棋子类型（涵盖所有常规国际象棋棋子）都可以在多项式时间内求解。最后，在第2.3节中，我们考察两种棋子类型的情形，其中一种类型仅包含一枚不可吃棋子，并证明在许多情况下，这个问题也可以在多项式时间内求解。

2.1 广义国际象棋模型

我们的算法结果适用于一个非常一般化的棋子在棋盘上移动的模型。该模型涵盖了国际象棋中的所有标准棋子类型，以及许多其他仙灵国际象棋棋子，例如滑行棋子和跳跃棋子。

将棋盘定义为一组 L 位置以及一组棋子，每枚棋子都分配到一个唯一的位置。因此，任意两个棋子都不能占据同一位置，但某些位置可能是空的。

将一组位置序列 $\langle \ell_0, \ell_1, \ldots, \ell_k \rangle$ 定义为一次移动。如果 ℓ_0 和 ℓ_k 各有一个棋子，而 ℓ_1、……、ℓ_{k-1} 均为空，则这次移动有效（就吃子而言）。执行这样一次有效移动会移除位置 ℓ_k 上的棋子，并将位置 ℓ_0 上的棋子移动到位置 ℓ_k。

将某种棋子可以进行的一组移动（在有效的情况下）定义为一种棋子类型。实际上，棋子类型会针对每个可能的起始位置，列出在某些特定的中间位置为空时，该棋子可以移动到哪些位置。

例如，国际象棋棋子类型由其移动定义

$$\Big\{\big\langle(x,y),(x+1,y),\ldots,(x+i,y)\big\rangle,\ \big\langle(x,y),(x,y+1),\ldots,(x,y+i)\big\rangle,$$
$$\big\langle(x,y),(x-1,y),\ldots,(x-i,y)\big\rangle,\ \big\langle(x,y),(x,y-1),\ldots,(x,y-i)\big\rangle \ \Big|\ i>0\Big\},$$

而国际象棋棋子类型由其移动定义

$$\Big\{\big\langle(x,y),(x+s,y+2t)\big\rangle,\ \big\langle(x,y),(x+2s,y+t)\big\rangle \ \Big|\ s,t\in\{-1,+1\}\Big\},$$

其中，棋盘要么是无限的，要么我们将范围限制为所有位置都在棋盘上的移动。请注意，该示例实现了车的走法具有的阻挡特性——沿途的所有位置都必须为空——而另一个示例则不存在这种阻挡。一般而言，如果所有移动都是由两个位置组成的序列，那么该棋子类型就是对移动封闭的。

我们要求每种棋子类型 T都对移动封闭，这意味着，如果 $\langle\ell_0,\ell_1,\ldots,\ell_k\rangle$是 T中的一个移动，那么对于任意整数 $0\le i<j\le k$，$\langle\ell_i,\ell_{i+1},\ldots,\ell_j\rangle$ 都是 T中的一个移动。（所有非阻挡棋子类型都自动满足这一限制。）

棋子类型的定义支持不对称移动的棋子，例如国际象棋棋子：

$$\Big\{\big\langle(x,y),(x+s,y+1)\big\rangle \ \Big|\ s\in\{-1,+1\}\Big\}.$$

如果某种棋子类型对表示移动的序列取逆后仍然封闭，我们就称其为对称。以下是此模型无法表示的国际象棋类棋子类型示例：

1. 象棋和韩国象棋中的马和象，它们的移动可能被无法吃掉的棋子阻挡（因此它们的移动不对子移动封闭）；

2. 象棋和韩国象棋中的炮，吃子前必须有棋子供其跳过；以及

3. 跳棋，吃子后落在被吃棋子前方的空格中。

在单人国际象棋中，任何初始为空的位置都将永远保持为空，因此可以从棋盘中省略这些位置。更准确地说，如果 ℓ_i 初始为空，那么我们可以将任何移动 $\langle\ell_0,\ldots,\ell_i,\ldots,\ell_k\rangle$ 替换为 $\langle\ell_0,\ldots,\ell_{i-1},\ell_{i+1}\ldots,\ell_k\rangle$；而任何满足$\ell_0$ 或 ℓ_k 的移动 $\langle\ell_0,\ldots,\ell_i,\ldots,\ell_k\rangle$，如果其相关位置初始为空，则可以完全删除。这些更改不会改变单人国际象棋谜题的结果，因此从此我们假设所有位置最初都被占据。

现在，我们定义几个有用的结构，并证明一些关于一般单人国际象棋谜题及其解的重要事实。

定义 ***Split*** 操作：将给定的移动拆分为一系列子移动，沿途吃掉所有棋子。具体来说，假设有一个移动 $m=\langle\ell_0,\ell_1,\ldots,\ell_k\rangle$，以及一个包含 ℓ_0和 ℓ_k的 位置 集合 L。令 ℓ_{j_0}，ℓ_{j_1}、.....、$\ell_{j_{k'}}$ 为 ℓ_0，ℓ_1、.....、ℓ_k与 L相交后得到的子序列。将 Split(m，L) 定义为子移动序列$\langle\ell_{j_r},\ell_{(j_r)+1}$、.....、$\ell_{j_{(r+1)}}\rangle$，其中 $r=0$、1、.....、$k'-1$ 按递增顺序排列。如果 m 是一个在子移动下封闭的棋子类型 T的移动，那么 Split(m，L) 就是棋子类型 T的一系列移动。此外，如果 L 包含 m 中当前被棋子占据的所有位置，那么 Split(m，L) 就是一系列有效移动。

引理 2.1。设 r 为有效解中最后一个棋子所在的位置。

1. 对于除 r 以外的每个位置 v，解中恰好有一步从 v 移出。*2.* 令 v_0、v_1、.....为从 v_0 开始，反复跟随解中从 v_i 移出的唯一一步所得到的位置序列。该序列是有限的，并以 r 结束。

证明。我们分别证明每个性质。

1. 如果不存在这样的一步，那么 v 将永远被占据，这与解的有效性矛盾。在这一步移出之后，v 将永远空着，因此不可能再有第二步从 v 移出。2. 假设结论不成立。由于位置的数量有限，该序列必定会进入一个由 r 以外的位置构成的循环。但这样一来，解中就没有一步可以移出该循环，因此循环中的至少一个位置必定会永远被占据，这与解的有效性矛盾。

□

2.2 单棋子类型很简单

本节将证明，对于任何单一棋子类型，单人国际象棋谜题都可以在多项式时间内求解，即使是第 2.1 节定义的非常一般的棋子类型也不例外。

定理 2.2。 Solo-Chess($\{T\}$) 对于任何在子移动下封闭的单一棋子类型 T，都可以在多项式时间内求解。

证明。 定义即时捕获图 ICG：这是一个有向图，每个位置 ℓ对应一个顶点 ℓ，而对于每个在初始棋盘上有效（不受其他棋子阻挡）的移动 $\langle \ell_0, \ell_1, \ldots, \ell_k \rangle$，都存在一条有向边 (ℓ_0, ℓ_k)。根据上述假设，初始棋盘上的所有位置都被占据，因此此类移动只涉及 $k = 2$ 位置。该图可以在多项式时间内计算出来。

我们断言，当且仅当即时捕获图 ICG 存在一棵生成入树时，该实例才可解；也就是说，存在一组边，使得每个顶点都存在一条通往公共根 r 的唯一路径。[1] (对于对称棋子类型，即时捕获图是无向图，因此只需找到一棵生成树并为其指定根即可。)这一性质可以通过 Edmonds 1967 年算法 [Edm67], 或其 $O(|E| + |V| \log |V|)$ 优化算法 [GGST86] 在多项式时间内检验；也可以使用经过修改的深度优先搜索 [KV06，练习 6.10], 在 $O(|E| + |V|)$ 时间内检验。

如果存在生成入树，我们可以按如下方式求解该实例。任何包含多个顶点的入树都有一个叶子顶点，即没有入边且有一条出边的顶点。反复找到一个叶子，并执行从该叶子到其父节点（沿唯一出边到达的顶点）的相应移动，然后删除该叶子。每次移动都是有效的，因为它对应即时捕获图中的一条边；而且根据构造，相关的两个棋子尚未被吃掉。由于入树是生成的，最终我们会缩减为位于位置 r 的一枚棋子。

反过来，假设该实例可通过一系列移动求解。令 r 为解中最后一个棋子所在的位置。我们将证明 ICG 存在一棵以 r 为根的生成入树。只需证明 ICG 中存在一条从每个位置到 r [的有向游走 KV06，定理 2.5(d)]。

[1]通常以相反的方式定义（出）树：从根 r 到每个顶点都有唯一的路径。在这个证明中，我们需要反过来的版本 in 。通过反转图中的所有边，现有的出树算法即可用于入树。

令 v_0 为任意位置，并令 v_0，v_1，……，r 为引理 2.1 中的位置序列。对于每一对位置 (v_i，v_{i+1})，解中都有一次从 v_i 移动到 v_{i+1} 的移动 m 。于是 Split(m，L)（其中 L 是所有位置的集合）是一系列双位置移动；根据定义，每次移动都对应 ICG 中的一条边。因此，我们得到 ICG 中一条从 v_i 到 v_{i+1} 的游走。将这些游走首尾相接，便得到 ICG 中一条从 v_0 到 r 的游走。因此，ICG 存在一棵生成入树。 □

2.3 单英雄与恶棍问题很容易

本节将介绍一种算法，用于求解某些单人国际象棋谜题。这类谜题由 n 个同一棋子类型的棋子，以及一个第二种棋子类型的棋子组成；后者必须是棋盘上最后剩下的棋子（因此不可被吃掉）。这推广了单人国际象棋中的单王限制：如果初始谜题中有王，那么在有效解中，王必须是棋盘上最后剩下的棋子。

我们将唯一不可被吃掉的棋子称为英雄，将数量为 n 的棋子称为反派。令 S 为英雄棋子类型，令 T 为反派棋子类型。我们要求 $S \subseteq T$，也就是说，英雄可以进行的每一种移动，反派也都可以进行。我们还要求 T 是对称的。两个值得考虑的情况是：英雄为国际象棋王或兵，而反派为国际象棋后。在国际象棋棋子中，$\{S^1, T\}$ 包括{♚^{1}, ♛}、{♟^{1}, ♛}、{♟^{1}, ♝}、{♜^{1}, ♛}和{♝^{1}, ♛}。

该算法的基本思路是让反派吃掉反派，使棋子逐渐聚拢到一条路径上，英雄便可沿此路径移动，从而吃掉所有反派。这种方法的主要难点在于阻挡：英雄本身会阻止反派从其所在位置穿过，因此可能需要等英雄离开后再进行移动。

2.3.1 即时捕获图中的游走

设 V 和 B 是互不相交的位置集合。将即时捕获图 $\mathbf{ICG}(\boldsymbol{V}, \boldsymbol{B})$ 定义为一个有向图，其顶点集为 V；当且仅当 $\langle \ell_0, \ell_1, \ldots, \ell_m \rangle \in T$是一次反派移动，且 ℓ_1，ℓ_2，……，ℓ_{m-1}均不属于 $V \cup B$ 时，图中存在一条边 (ℓ_0，ℓ_m)。（注意，ℓ_1，ℓ_2，……，ℓ_{m-1}可能不属于 V。）此图表示：当反派位于V 中的位置，且不可移动且不可捕获的阻挡棋子位于 B中的位置时，所有有效反派移动所构成的图。

我们现在 证明一些有用的事实，以确保 ICG(V，B) 中存在（有向）游走。

引理 2.3。设 V 和 B 是不相交的位置集合。假设 $\langle \ell_0, \ell_1, \ldots, \ell_m \rangle \in T$ 是一个反派移动，使得 ℓ_0、$\ell_m \in V$ 和 $\{\ell_0, \ell_1, \ldots, \ell_m\} \cap B = \emptyset$。那么在 $\mathrm{ICG}(V$、$B)$ 中存在一条从 ℓ_0 到 ℓ_m 的有向游走。

证明。由于 ℓ_0，$\ell_m \in V$、ℓ_0 和 ℓ_m 都是 ICG(V，B) 的顶点。考虑反派移动 Split($\langle \ell_0, \ell_1, \ldots, \ell_m \rangle$，$V$) 所组成的序列。根据 Split 的定义，每个这样的移动 $\langle \ell_s, \ell_{s+1}, \ldots \ell_{s+r} \rangle$ 都具有以下性质：$\{\ell_{s+1}, \ell_{s+2}, \ldots, \ell_{s+r-1}\} \cap (V \cup B) = \emptyset$，而 ℓ_s，$\ell_{s+r} \in V$。因此，(ℓ_s，ℓ_{s+r}) 是 ICG(V，B) 的一条边。将这些边连接起来，我们便得到 ICG(V，B) 中一条从 ℓ_0 到 ℓ_m 的有向游走。 □

推论 2.4。设 V和 B是互不相交的位置集合，且 $m_0, m_1, \ldots, m_k \in T$是一个反派棋步序列，满足

1. m_i 的目的地是 m_{i+1}的起点；
2. 每个移动的起点和目的地都在 V中；并且
3. 没有任何移动经过 B中的位置。

然后那里 是 ICG(V, B) 中从 m_0 的起点到 的目的地的有向游走 m_k.

证明。 每次移动都满足引理 2.3 的条件，因此在 ICG(V, B) 中，存在一条从每次移动的起点到目的地的有向游走。将这些游走连接起来，就得到了所需的游走。

□

推论 2.5。 设 V 和 B 是不相交集合，且 V' 和 B' 是不相交集合，并且 $V \subseteq V'$ 和 $B' \subseteq B$。（注意这种不对称性。）假设在 ICG(V, B) 中存在一条从 ℓ_0 到 ℓ_1 的有向游走。那么，在 ICG(V', B') 中也存在一条从 ℓ_0 到 ℓ_1 的有向游走。

证明。 在 ICG(V, B) 中，从 ℓ_0 到 ℓ_1 的游走是一系列移动，每次移动的起点和目的地都在 $V \subseteq V'$ 中，并且没有一次移动经过 $B \supseteq B'$ 中的位置。因此，根据推论 2.4，在 ICG(V', B') 中存在一条从 ℓ_0 到 ℓ_1 的有向游走。 □

2.3.2 英雄路径

给定一个棋盘，英雄位置序列 是指位置序列 p_0，p_1，……，p_k，其中 p_0是英雄在给定棋盘上的初始位置。英雄路径 是具有额外性质的英雄位置序列：在初始棋盘上（没有进行任何其他移动时），英雄可以依次进行合法移动 $\langle p_0, \ldots, p_1 \rangle$、$\langle p_1, \ldots, p_2 \rangle$、……、$\langle p_{k-1}, \ldots, p_k \rangle$。对于国际象棋中的王或兵，这个定义尤其简单；但更一般地说，反派棋子可能会阻挡英雄移动。我们将在下面的引理 2.9 中证明，只需考察英雄沿着英雄路径移动的解；不过目前我们知道，英雄至少会沿着一条英雄位置序列移动。

对于固定的英雄位置序列 p_0，p_1、……、p_k（统称为 p），以及一个满足 $0 \le i < k$ 的指标 i，我们将 $H_i(p) = \text{ICG}(L \setminus \{p_0, p_1, \ldots, p_i\}, \{p_i\})$ 定义为反派在棋盘上仅完成前 i 次英雄移动后的即时捕获图（定义如上）（此时英雄位于 p_i，而 p_0，p_1、……、p_{i-1} 均为空）。若上下文清楚，我们就简写为 H_i（省略英雄位置序列 p）。

如果对于某个满足 $0 \le i < k$ 的指标 i，存在一条从 v 到 p_{i+1} 的有向游走，则称位于位置 v 的反派可被英雄位置序列 p_0，p_1、……、p_k强可捕获。直观地说，当英雄位于位置 p_i 时，反派可以移动到英雄所在的位置 p_{i+1}，随后在英雄下一次移动时被吃掉。根据定义，所有 p_1，p_2、……、p_k 都是强可捕获的。

引理 2.6。 设 $p_0, p_1, \ldots, p_k$ 是一个英雄位置序列。假设在 H_i 中存在一条从 v 到 p_j 的有向游走，且满足 $i < j$。那么 v 可由 $p_0, p_1, \ldots, p_k$ 强可捕获。

证明。 不失一般性，我们可以假设这条游走是最短的，因此 p_j 是这条游走首次到达 $\{p_{i+1}, p_{i+2}, \ldots, p_k\}$ 中任一位置的地方。于是，这条游走构成了一系列移动，每次移动的起点和目的地都在 $L \setminus \{p_0, p_1, \ldots, p_{j-1}\}$ 中，且不会经过 p_{j-1}（因为它们是 H_i 的边）。根据推论 2.4，在 H_{j-1} 中存在一条从 v 到 p_j 的有向游走，因此 v 可由 p_0，p_1、……、p_k 利用索引 $j-1$ 强可捕获。 □

接下来，我们将证明，只要知道英雄访问的位置序列，就足以解开谜题。也就是说，给定一条英雄路径 p_0，p_1，……，p_k，我们将说明如何在多项式时间内判断是否存在一个有效解，使英雄恰好沿着这组位置移动 k 步。具体而言，我们分两部分证明：所有反派都能被 p_0，p_1，……，p_k 强可捕获，是一个充要条件。

引理 2.7。 假设存在一个解，使英雄沿英雄位置序列 $p_0, p_1, \ldots, p_k$ 恰好移动 k 次。那么所有反派都能被 $p_0, p_1, \ldots, p_k$ 强可捕获。

证明。考虑位于位置 v_0 的反派。令 $v_0, v_1, \ldots, p_k$ 为将引理 2.1 应用于该解所得的位置序列。在第一个满足 $v_r \in \{p_0, p_1, \ldots, p_k\}$ 的位置 $v_r = p_{i+1}$ 处截断该序列，因此特别有 $v_j \notin \{p_0, p_1, \ldots, p_i\}$ 对于 $0 \le j \le r$。

对于每一对（v_j, v_{j+1}）（其中 $0 \le j < r$），解中都存在一个 $\langle v_j = \ell_0, \ell_1, \ldots, \ell_m = v_{j+1} \rangle$ 反派移动。这些反派移动必须在解中先于从 p_i 到 p_{i+1}的英雄移动发生，因为英雄必须向 p_{i+1}走出最后一步。因此，当每个反派移动发生时，$p_i \notin \{\ell_0, \ell_1, \ldots, \ell_m\}$ 仍被 p_i 占据。

因此，我们得到了一系列移动，其中每一步的起点和目的地都在 $L \backslash \{p_0, p_1, \ldots, p_i\}$ 中，并且没有一步经过 p_i。根据推论 2.4，在H_i中存在一条从 v_0 到 p_{i+1} 的有向游走，因此 v_0 是强可捕获的。□

引理 2.8。设 $p_0, p_1, \ldots, p_k$ 是一条英雄路径，且每个反派都能被 $p_0, p_1, \ldots, p_k$ 强可捕获。那么存在一个解，使英雄恰好进行 k 次移动，依次经过位置 $p_0, p_1, \ldots, p_k$。

证明。对于每个从位置 v 出发的反派，根据强可捕获的定义，我们得到一个满足 $0 \le i < k$ 的指标 i，称为该反派的 秩；同时还得到一条 H_i 中从 v 到 p_{i+1} 的有向游走。删除游走中的所有环后，可以假设该游走实际上是一条简单路径。

我们将按逆序输出一系列能够解决该实例的移动。按秩对反派排序，秩相同的任意排列，然后从秩最高到最低依次处理。对于每个满足 $0 \le i < k$ 的指标 i，将从 p_i 到 p_{i+1} 的英雄移动添加到输出中，并按如下方式处理秩为 i 的反派。对于每个从位置 v 出发且秩为 i 的反派，在从 v 到 p_{i+1} 的简单路径中，找到最早的一个位置 v'，使得一条到达或离开 v' 的移动已在我们的输出移动列表中。这样的位置总是存在，因为我们刚刚添加了一步到达 p_{i+1} 的英雄移动。将从 v 到 v' 的子路径所对应的反派移动序列（按逆序）添加到输出中。（特别地，如果 $v = v'$，我们就不添加任何移动。）将处理秩为 i 的反派时添加的每个反派移动的秩也定义为 i。

我们断言，不存在从 p_0、p_1、……、p_k出发的反派移动，也不存在任何秩为 i、吃掉 p_i 的反派移动。这是因为从 v到 p_{i+1}的路径不包含 p_0、p_1、……、p_i中的任何一个，因为它们不是 H_i的顶点；此外，所有涉及 p_i、p_{i+1}、……、p_k的英雄移动都已经存在。

我们必须证明，得到的移动序列是一个使用英雄路径 p_0、p_1、……、p_k的解。完整的英雄路径 p_0、p_1、……、p_k 按构造出现在该解中。除 p_k 之外，每个位置都恰好出现一次，作为输出中某个移动的起点；并且输出中没有从 p_k出发的移动。每个秩为 i 的反派移动发生时，英雄都位于 p_i，并且已经吃掉了 p_0、p_1、……、p_i中的所有顶点。没有反派移动会吃掉英雄，因为不存在任何秩为 i、吃掉 p_i的反派移动。

还需证明每一步移动在执行时都是合法的。移动可能不合法的原因有两种：它可能移动到或离开一个当前为空的位置，也可能被中间的棋子阻挡。移动的起点不可能为空，因为每个位置至多只会作为一着移动的起点出现一次。移动的目的地不可能为空，因为每一步移动要么是移动到 p_k，要么是移动到某个位置，而从该位置出发的唯一一步移动发生在更晚的时刻。根据英雄路径的定义，没有英雄移动会被阻挡：即使不移动反派，p_0、p_1、……、p_k 也必须是一连串合法的移动。没有反派移动会被阻挡，因为秩为 i 的反派移动只沿着 H_i 的边进行；根据 H_i 的定义，此类移动只有可能被 p_0、p_1、……、p_{i-1}阻挡，而英雄位于 p_i 时，这些位置都是空的。□

注意，引理 2.7 适用于任意英雄位置序列 $p_0 \ldots p_k$，但引理 2.8 要求英雄路径。这里存在一个技术问题，因为在英雄位置序列中，英雄移动可能会被阻挡，而解可能会通过移开反派来避免这种情况。不过，我们可以证明，总是存在另一种无需这样做的解：

引理 2.9. 假设一个谜题有解。那么，存在一种解，使英雄经过的位置序列构成一条英雄路径。

证明。 考虑只取解中的英雄移动子序列，并尝试在不进行任何反派移动的情况下执行这些移动。在此过程中，我们可能会遇到一个英雄移动 m，由于被某个尚未被吃掉的反派集合 V_m 阻挡而不合法。将每个这样的英雄移动 m 替换为 Split(m, V_m)，后者是一个合法移动序列。所得的英雄移动序列构成一条英雄路径 p_0，p_1，……，p_k，其中包含原始英雄位置序列，也包含所有这些 V_m 集合。

还需证明存在一个解，使得英雄沿着这条英雄路径恰好进行 k 次移动。根据引理 2.8，只需证明每个反派都能被 p_0，p_1，……，p_k 强可捕获。我们使用与引理 2.7 类似的论证来证明这一点。

考虑位于位置 v_0 的一个反派。令 v_0，v_1，……，v_r 为将引理 2.1 应用于原始解所得的位置序列。在第一个满足 $v_r \in \{p_0, p_1, \ldots, p_k\}$ 的位置 $v_r = p_{i+1}$ 处截断该序列，从而特别有：对于 $0 \le j \le r$，$v_j \notin \{p_0, p_1, \ldots, p_i\}$。

设 p_s 是反派从 v_{r-1} 移动到 p_{i+1} 时，英雄在原解中的位置；这样，上述所有反派移动都发生在解中英雄唯一一次离开 p_s 的移动之前。这些反派移动都不会经过 p_s，因为它们发生时该位置上有棋子。必有 $s < i+1$，因为在解中，直到反派最后一次移动到 p_{i+1} 之后，英雄才能移动到 p_{i+1} 或经过 p_{i+1}；而这次反派移动发生在英雄移动到 p_s 之后。

根据推论 2.4，在 H_s（p）中存在一条从 v_0 到 p_{i+1} 的有向游走，因此根据引理 2.6，v_0 可被 p_0、p_1、……、p_k 强可捕获。 □

根据上述引理，我们可以将谜题的解视为一条英雄路径，使所有反派都强可捕获。

接下来，我们定义“弱可捕获性”的概念，用来排除解的英雄路径的前缀。对于英雄路径 p_0、p_1、……、p_k，定义 G_i（p）$= ICG$（$L \setminus \{p_0, p_1, \ldots, p_i\}$，$\emptyset$）为移除 p_0、p_1、……、p_i 后棋盘的即时捕获图。若从 v 到 $p_{k'}$ 存在一条 G_k 中的有向游走，则称位于 v 的反派在经过 $\boldsymbol{p_0}$、$\boldsymbol{p_1}$、……、$\boldsymbol{p_k}$ 后弱可捕获，其中英雄路径 p_0、p_1、……、$p_{k'}$ 以 p_0、p_1、……、p_k 为前缀。尽管 H_i（p）是 G_i（p）的子图，弱可捕获性并不一定意味着沿同一条英雄路径强可捕获，因为只有后者可以选择一个索引 i；弱可捕获性必须使用 $i = k$。（直观地说，强可捕获性意味着反派实际上已经被吃掉，而弱可捕获性意味着反派将来可能被吃掉。）

我们证明，如果英雄路径不能至少使所有反派弱可捕获，那么它就不能作为解的潜在前缀：

引理 2.10. 设 $p_0, p_1, \ldots, p_k$ 是一条英雄路径，并假设存在一个解，其英雄路径以 $p_0, p_1, \ldots, p_k$ 为前缀。那么，每个反派要么可被 $p_0, p_1, \ldots, p_k$ 强可捕获，要么在 $p_0, p_1, \ldots, p_k$ 之后弱可捕获。

证明。 设 $p_0, p_1, \ldots, p_{k'}$ 是该解中的英雄路径，且设 v 是一个反派的位置。根据引理 2.7，v 可被 $p_0 \ldots p_{k'}$ 强可捕获。也就是说，存在一个索引 $i < k'$，以及一条从 v 到 p_{i+1}、位于 H_i 中的有向游走。如果 $i < k$，那么 v 可被 $p_0 \ldots p_k$ 强可捕获，因此假设 $i \ge k$。

根据推论 2.5，$H_i =$ ICG($L \setminus \{p_0, p_1, \ldots, p_i\}, \{p_i\}$) 中的一条有向游走可以扩展为 $G_k =$ ICG($L \setminus \{p_0, p_1, \ldots, p_k\}, \emptyset$) 中的一条有向游走。因此，在 G_k 中存在一条从 v 到 p_{i+1} 的有向游走，所以在 $p_0, p_1, \ldots, p_k$ 之后，v 是弱可捕获的。 □

2.4 有趣的英雄路径

Co考虑位于位置 v 的反派。定义一条英雄路径 $p_0 \dots p_k$，使其为 ***v*-有趣的** if

1. v 无法被 p_0、p_1、……、p_k强可捕获；并且
2. v 在经过 p_0、p_1、……、p_k后可被弱可捕获。

我们将证明，对于任意 v，v-有趣路径构成一棵树；并且对于某个 v，英雄路径具有 v-有趣前缀。这些性质共同限制了我们需要搜索的可能英雄路径的数量。

引理 2。 **11.** 如果 $p_0, p_1, \dots, p_k$ 是一条 v-有趣路径，那么该路径的每个前缀也都 $o.$

证明。令 $p_0, p_1, \dots, p_j$ 是 $p_0, p_1, \dots, p_k$ 的一个前缀，即 $j \le k$。

如果 v 被 $p_0 \cdots p_j$ 强可捕获，那么使用相同的指标，它也能被 $p_0 \dots p_k$ 强可捕获。

假设 v 在 $p_0, p_1, \dots, p_k$ 之后弱可捕获。那么存在某个英雄路径 $p_0, p_1, \dots, p_{k'}$，其前缀为 $p_0, p_1, \dots, p_k$，并且在 $G_k = \text{ICG}(L \setminus \{p_0, p_1, \dots, p_k\}, \emptyset)$ 中存在一条从 v 到 $p_{k'}$ 的有向游走。根据推论 2.5，这条有向游走可以延伸为 $G_j = \text{ICG}(L \setminus \{p_0, p_1, \dots, p_j\}, \emptyset)$ 中的一条有向游走。因此，v 在 $p_0, p_1, \dots, p_j$ 之后弱可捕获。

□

推论 2.12. 假设 $p_0, p_1, \dots, p_k$ ***with*** $k > 0$ 是一个最小解；也就是说，没有任何前缀也是一个解。那么，在位置 v 有一个反派，使得对所有 $j < k$，$p_0, p_1, \dots, p_j$ 都是 v-有趣的。

证明。由于 $p_0, p_1, \dots, p_{k-1}$ 不是一个解，根据引理 2.8，存在某个 v，它无法被 $p_0, p_1, \dots, p_{k-1}$ 强可捕获。根据引理 2.10，v 在 $p_0, p_1, \dots, p_{k-1}$ 之后弱可捕获。因此，$p_0, p_1, \dots, p_{k-1}$ 是 v-有趣的。最后，根据引理 2.11，对所有 $j <k$，$p_0, p_1, \dots, p_j$ 都是 v-有趣的。

□

引理 2.13。 假设反派棋子类型 T 是对称的。如果在英雄路径 $p_0, p_1, \dots, p_k$之后，v 是弱可捕获的，那么在 G_{k-1}中，v与 p_k 连通。

证明。根据弱可捕获的定义，存在一条英雄路径 $p_0, p_1, \dots, p_k, p_{k+1}, \dots, p_{k'}$，使得在 G_k中存在一条从 v 到 $p_{k'}$ 的路径 P_1。根据推论 2.5，P_1 可以扩展为 G_{k-1}中的一条路径 P_1'。

考虑英雄路径中英雄移动的后缀 $p_k, p_{k+1}, \dots, p_{k'}$。由于英雄棋子类型 S 是反派棋子类型 T 的子集，根据推论 2.4，我们知道这串英雄移动可以扩展为一条在 $\text{ICG}(\{p_k, p_{k+1}, \dots, p_{k'}\}, \emptyset)$ 中从 p_k到 $p_{k'}$的路径 P_2。根据推论 2.5，P_2可以扩展为 G_{k-1}中的一条路径 P_2'。

将 P_1' 与 P_2' 的反向路径拼接起来（根据反派移动的对称性），我们便得到了一条从v 到 p_k 的路径，位于 G_{k-1}中。

□

引理 2.14。 假设反派棋子类型 T是对称的。如果 p_0、p_1、……、p_k和 q_0、q_1、……、q_j是两条不同的英雄路径，且起点和终点相同，那么它们中至多有一条是 v-有趣的。

证明。反设两条路径都是 v-有趣的。令 $i + 1$ 为两条路径分歧的第一个索引，因此 $p_i = q_i$，但 $p_{i+1} \neq q_{i+1}$。根据引理 2.11，不失一般性地假设 k 和 j 是满足 $p_k = q_j$的最小整数 $> i$。根据引理 2.11，p_0，p_1，……，p_{i+1}是 v-有趣的。根据引理 2.13，在 G_i中存在一条从 v到 p_{i+1}的路径 P_1。在 P_1首次进入 $R = \{p_{i+1}, p_{i+2}, \dots, p_k, q_{i+1}, q_{i+2}, \dots, q_j\}$处将其截断；

不失一般性地，假设该点是某个 $i+1 \le r \le j$对应的 q_r。因此， P_1 成为 ICG $\big((L \setminus (p \cup q)) \cup \{q_r\}, R \setminus \{q_r\}\big)$中的一条路径。根据推论 2.5， P_1 可延伸为 $H_{k-1}(p)$中的一条路径 P_1'。

考虑对应英雄路径中的英雄移动后缀 q_r， q_{r+1}，……， q_j。根据英雄路径的定义以及 j 的极小性，这些移动都不会经过任何一个 p_{i+1}， p_{i+2}，……， p_{k-1}。由于英雄移动类型 S 是反派棋子类型 T 的子集，根据推论 2.4，我们知道这一系列英雄移动可以延伸为一条从 q_r 到 q_j、位于 ICG$(\{q_r, q_{r+1}, \ldots, q_j\}, \{p_{i+1}, p_{i+2}, \ldots, p_{k-1}\})$ 中的路径 P_2。根据推论 2.5， P_2 可以延伸为 $H_{k-1}(p)$ 中的一条路径 P_2'。

最后，将 P_1' 和 P_2' 拼接起来，我们得到 $H_{k-1}(p)$ 中一条从 v 到 $q_j = p_k$ 的路径。但这样一来， v 就被 p_0， p_1，……， p_k 强可捕获，这与它是 v- 有趣的相矛盾。 □

如果某条英雄路径对于某个反派位置 v 是 v-有趣的，则称其为有趣的。

引理 2.15。给定一条英雄路径 $p_0, p_1, \ldots, p_k$，我们可以在多项式时间内计算它是否有趣，以及是否为一个解。

证明。我们可以通过构造图 H_i 并检查对于每个 i， p_{i+1} 是否可从 Q 到达，来计算反派位置 v 是否强可捕获。

我们还可以通过构造图 G_k，并遍历所有位置，寻找一个位置 ℓ，使得在 G_k 中 ℓ 可从 v 到达，且在可能的英雄移动图中也可从 p_k 到达，从而计算反派位置 v 是否弱可捕获。 □

定理 2.16。 Solo-Chess$(\{S^1, T\})$ 对于任意两种在子移动下封闭的棋子类型$S \subseteq T$，只要 T 对称，就可以在多项式时间内求解。

证明。根据推论 2.12，最小解的每个前缀必然都是有趣的。此外，根据引理 2.14，每个位置至多有一条以其为终点的有趣英雄路径。

我们的算法通过搜索所有有趣英雄路径构成的树来找到最小解。根据引理 2.15，我们可以在多项式时间内判断一条英雄路径是否有趣，也可以判断它是否能解决该实例。位置、有趣路径以及待尝试的移动都只有多项式多个，因此该搜索需要多项式时间。 □

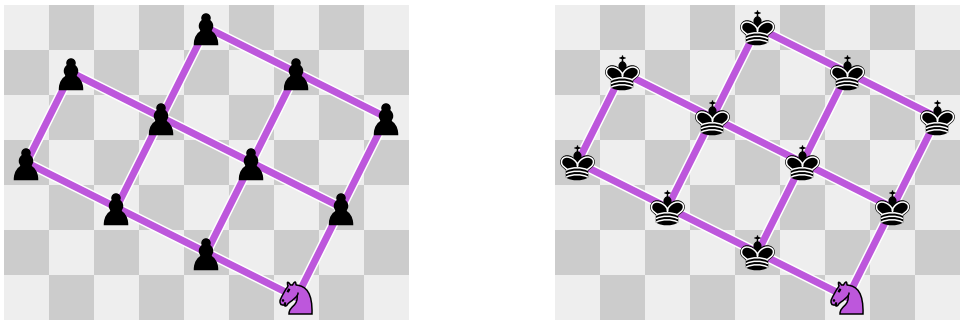
3 困难性结果

在本节中，我们证明，对于任意一组由 S 个不同标准国际象棋棋子组成的集合， Solo-Chess(S) 都是 NP 完全问题。这些归约在其中一种棋子类型被称为特殊时也成立，但需限制该棋子只能有一个，且它必须是棋盘上的最后一个棋子。这是单王限制 [1]的推广；对于其他棋子类型 T，我们使用相同的记号 T^1。但是，所有困难性归约在棋子类型 T 不受限制且可以被吃掉时也成立。

我们根据归约的起点将本节分为两个小节。在第 3.1 节中，我们给出多个从哈密顿路径归约而来的结果，其中特殊棋子需要访问其他棋子。在第 3.2 节中，我们从 3SAT 的一个特殊情形进行归约，其中特殊棋子负责设置变量并满足子句。

3.1 哈密顿路径归约

所有这些归约都来自最大度为 3 的网格图上的 HamiltonianPath 问题：该问题指定一个度为 1 的起始顶点，并可能还指定一个度为 1 的终止顶点；或者来自该问题的推广形式（例如，有时不需要满足网格图或度为 1 的条件）。通过对 [PV84] 作细微修改（见附录 A），可知该问题是 NP 困难的。



(a) 兵和一匹骑士　　(b) 王和一匹骑士

图 1：放置兵或王以及一匹骑士，以模拟具有指定起始顶点的网格图中的哈密顿路径。

当特殊棋子是骑士时，第一个归约尤其简单：

定理 3.1。 对于任意 $T \in \{\text{♟}, \text{♚}\}$, Solo-Chess($\{\text{♞}^1, T\}$) 和 Solo-Chess($\{\text{♞}, T\}$) 都是 *NP* 困难的。

证明。 该归约来自具有给定起始顶点的网格图中的哈密顿路径问题 s （引理 A.1）。图 1 展示了该归约的一个示例。我们将网格图旋转 $\arctan \frac{1}{2}$，并进行缩放，使相邻顶点之间构成合法的骑士移动。我们在网格图顶点处放置兵或王，除了 s，该处放置一匹骑士（构造中唯一的骑士）。兵或王无法吃子，因此无法移动。于是，骑士必须吃掉其他所有棋子；这一系列吃子对应一条从 s 出发的哈密顿路径。 □

接下来，我们给出一个从哈密顿路径到 Solo-Chess（$\{\text{♜}^1, \text{♟}\}$）的归约。该归约构成了证明其他若干种棋子组合具有 NP 困难性的基础，可通过缩放和/或旋转来实现。与前一个归约类似，这些构造背后的主要思路是放置许多无法移动的同类棋子，再放置一枚特殊类型的棋子，使其必须吃掉其他所有棋子，从而要求存在一条哈密顿路径。

Th定理 3.2. Solo-Chess($\{\text{♜}^1, \text{♟}\}$) 和 Solo-Chess($\{\text{♜}, \text{♟}\}$)是 *NP* 困难 .

证明。 该归约来自最大度为 3 的图中的哈密顿路径问题，其中给定起始顶点 s 和目标顶点 t，二者的度均为 1（引理 A.1）。

参见图2。除 s 和 t 以外，每个顶点由七个兵组成，分布在两行中；其中四个兵标记一个很宽、高为 2 的矩形的四个角，另外三个兵位于矩形底行，每个兵各构成一条连接至另一顶点的边的一半。更具体地说，若将每个顶点标记为 1 到 $|V|$ 之间的整数，并将每条边标记为 1 到 $|E|$之间的整数，则顶点 i将四个角上的兵放在位置 $(i, 3i)$，$(i, 3i+1)$，$(4|V|+i, 3i)$，$(4|V|+i, 3i+1)$；而连接顶点 i和 j的边 k则分别在顶点 i和顶点 j的矩形底行位置 $(|V|+k, 3i)$，$(|V|+k, 3j)$放置兵。因此，每列有零个或两个兵，每行有零到五个兵。对于起始顶点 s及其关联边 k_s，我们在 (k_s,1)处放置一个车。同样，对于目标顶点 t 及其关联边k_t，我们在 (k_t,0)处放置一个兵。

在这个构造中，没有任何兵能够吃子，因此只有车可以移动。车只能通过顶点的三条边兵进入或离开一个顶点，因此每个顶点最多只能进入一次：如果进入、离开后又再次进入，那么在度数最多为 3 的图中就再也无法离开，从而无法到达 t （其度数本身为 1，因此永远无法被

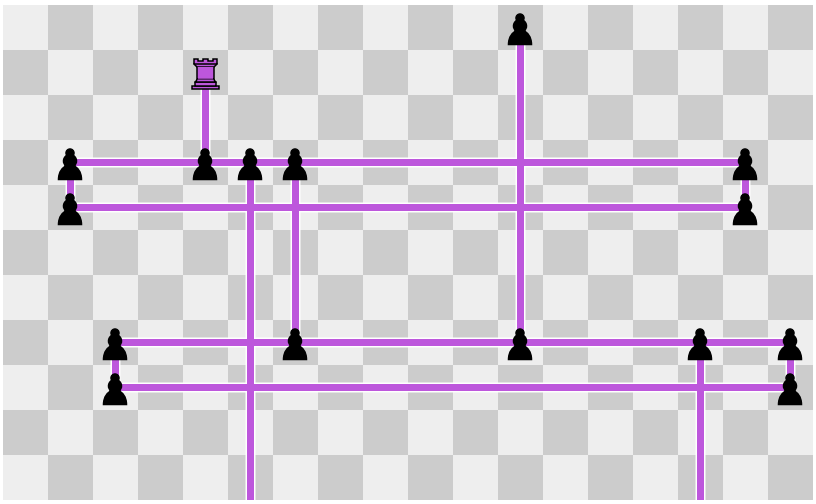


图2：在最大度为 3 的图中放置兵和一辆车，以模拟具有指定起点和终点顶点的哈密顿路径。图中展示了由一条边连接的两个顶点，以及起点和目的地顶点。

离开）。因此，为了访问所有兵，车必须恰好进入并离开除 s 和 t 以外的每个顶点一次。车一旦进入一个顶点，就必须访问该顶点的全部七个兵。沿着顶点矩形顺时针或逆时针移动，车便可以选择沿着与该顶点相连的另外两条边中的任意一条离开。因此，车从 s 的起始位置吃掉所有兵，当且仅当存在一条从 s 到 t 的哈密顿路径。

□

推论3.3。 对于任意 $T \in \{\text{♚}, \text{♞}\}$, Solo-Chess($\{\text{♜}^1, T\}$) 以及 Solo-Chess($\{\text{♜}, T\}$)，均为 *NP* 困难。

证明。 将定理3.2中的构造在两个维度上都放大 4 倍，并将每个兵替换为类型为 T 的棋子。这样的缩放可阻止王或骑士吃子，同时不影响车的走法。

□

推论 3.4。 对于任意 $S \in \{\text{♜}, \text{♛}\}$ 和 $T \in \{\text{♟}, \text{♚}, \text{♞}, \text{♝}\}$。 Solo-Chess($\{S^1, T\}$) 以及 Solo-Chess ($\{S, T\}$)，均为 *NP* 困难。

证明。 将定理3.2中的构造沿 x 方向放大 $h+1$ 倍，其中 h 是原构造的高度。（换句话说，在每一对相邻的棋子列之间增加 h 个空列。）这样缩放后，棋子之间的间距足够大，不会发生斜线吃子。如果 $S = \text{♛}$，用后替换车不会增加任何额外的斜线走法。最后，我们将每个兵替换为类型为 T 的棋子。如果 $T \in \{\text{♟}, \text{♝}\}$，这些棋子无法移动，因为没有斜线走法。如果 $T \in \{\text{♚}, \text{♞}\}$，我们还会在两个维度上额外放大 4 倍（如推论3.3所示），以确保这些棋子无法移动。

□

推论3.5。 对于任意 $T \in \{\text{♟}, \text{♚}, \text{♞}\}$, Solo-Chess($\{\text{♝}^1, T\}$) 以及 Solo-Chess($\{\text{♝}, T\}$) 都是 *NP* 困难。

证明。 将定理3.2中的构造旋转 $45°$，并按 $\sqrt{2}$缩放；见图3。此变换将车的走法变为象的走法：原本正交相邻的顶点现在变为对角相邻。将车替换为象，并将每个兵替换为一个棋子

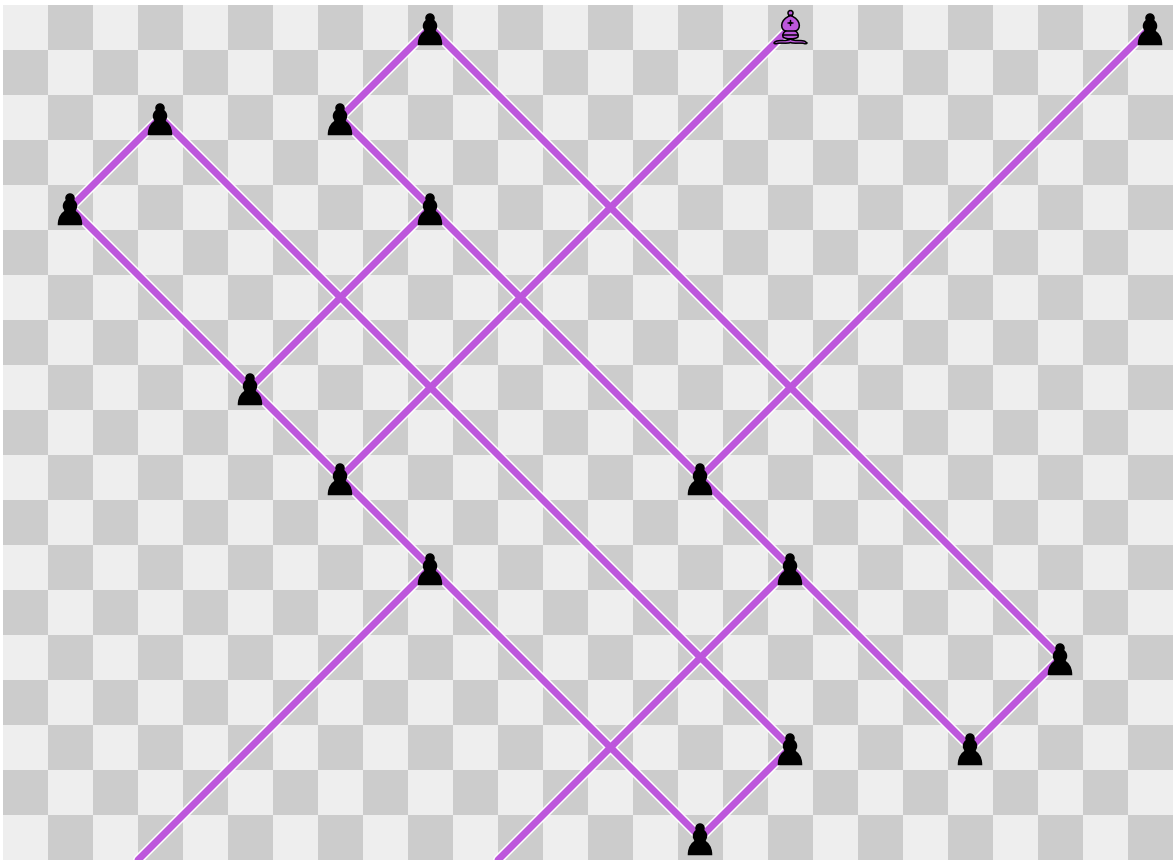


图3：通过旋转图2中的构造，放置兵和一枚象，以模拟最大度为 3 且指定起点和终点的图中的哈密顿路径。

类型为 T的棋子。对于 $T \in \{\text{♚}, \text{♞}\}$，我们在两个维度上都额外缩放4倍（如推论3.3所示），以确保这些棋子无法移动。 □

推论3.6. 任意 $T \in \{\text{♝}, \text{♛}\}$，Solo-Chess$(\{T^1, \text{♜}\})$ 和 Solo-Chess$(\{T, \text{♜}\})$ 都是 *NP* 困难。

证明。 首先，按照推论3.3中的做法，将定理3.2中的构造沿 x 方向缩放 $h+1$ 倍。然后，按照推论3.5中的做法，旋转 $45°$ 并缩放 $\sqrt{2}$ 倍。最初的缩放消除了原构造中的对角线对齐，从而防止棋子在旋转后的版本中出现正交对齐。然后，将车替换为类型为 T 的棋子，并将每个兵替换为车。

□

下一个归约同样来自哈密顿路径问题，但并未采用此前归约所用的相同框架。

定理 3.7. Solo-Chess$(\{\text{♚}^1, \text{♟}\})$ 和 Solo-Chess$(\{\text{♚}, \text{♟}\})$都是 *NP* 困难。

证明。 该归约从具有指定起始顶点的最大度为 3 的网格图上的哈密顿路径问题归约而来 s （引理A.1 ）。图 4 展示了构造示例。首先，我们将给定的网格图旋转 $45°$ 并按 $3\sqrt{2}$ 缩放，在顶点处及沿边放置兵。顶点处的兵绘制为蓝色。相邻顶点上的兵沿对角线相距三个格，且

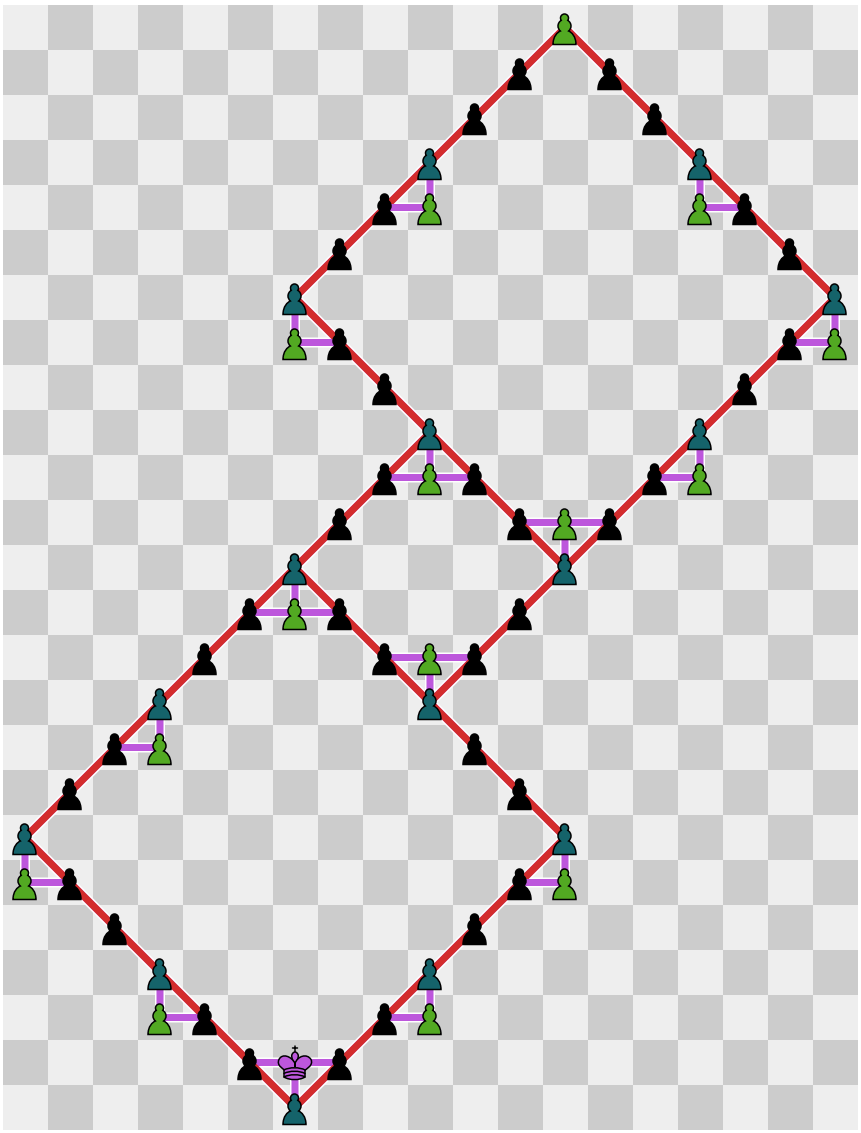


图4：在最大度为 3 的网格图中放置兵和一个王，以表示指定起始顶点的哈密顿路径问题。蓝色兵位于网格图顶点上，而绿色兵没有吃子移动。

它们之间有一条由两个（黑色）兵组成的对角链。构成网格图的所有这些兵都放置在棋盘的浅色格上。

假设兵向上吃子。现在，对于每个至少有一条向上的关联边的顶点，我们在相邻的深色格上放置一个绿色兵：如果有两条向上的关联边，就将它放在顶点上方；否则，就将它放在顶点下方。注意，只有向下的关联边的顶点不会放置绿色兵；在这种情况下，我们将该顶点上的兵涂成绿色。这样，每个顶点恰好对应一个绿色兵，该兵没有合法吃子，而所有其他兵都有合法吃子。所有其他兵（图4中的黑色或蓝色兵）都至少有一种合法吃子方式。王取代哈密顿路径问题起始顶点 s 上的绿色兵（图4中最靠下的顶点）。

如果存在一条哈密顿路径，那么我们断言存在一个有效的吃子序列，使得最后只剩下

王。王将沿着哈密顿路径吃子，并确保在每个顶点都绕行去吃掉绿色兵。不过，在王移动之前，所有位于哈密顿路径之外的格子上的兵，都会从最底下的兵开始，吃掉它们上方的兵。这些兵总是有合法吃子，因为根据我们的构造，每个兵要么有合法吃子，要么就是属于哈密顿路径的绿色兵。完成这些吃子后，剩下的兵就只位于哈密顿路径上，因此王只需沿着这条路径游走，将所有兵吃掉。

反过来，如果单人国际象棋问题存在有效解，那么我们声称底层网格图中必定存在一条哈密顿路径。由于绿色兵永远无法移动，王必定会在某个时刻吃掉每个绿色兵。因此，王的路径从王的初始位置开始，经过各个兵所在的位置，不会两次吃掉同一个格子，并且会吃掉每个绿色兵。由于该图的最大度数为 3，王至多访问每个顶点一次；又由于每个顶点都与一个绿色兵相邻，王的路径必须能够至少访问每个顶点一次。因此，王的路径给出了图中的一条哈密顿路径。（即使没有[1] 限制，这一论证仍然成立，因为如果王在吃掉所有绿色兵之前被吃掉，谜题就无法解出。）

□

至此，我们已经完成证明：对于任意两种标准国际象棋棋子， Solo-Chess(S) 都是 NP 完全的：

推论 3.8。 Solo-Chess(S) *对于任意一组由两种不同标准国际象棋棋子组成的棋子，都是 NP 完全的。*

证明。 推论 3.4 和 3.4 合起来涵盖了$\in S$ 或$\in S$的所有情形，剩下 $S \subseteq$ { ♕, ♔, ♘, ♗ }。推论 3.5 涵盖了$\in S$的所有剩余情形，剩下 $S \subseteq$ { ♟, ♔, ♘ }。定理 3.4 涵盖了$\in S$的所有剩余情形，剩下 $S \subseteq$ { ♟, ♔ }。定理 3.7 涵盖了最后一种情形 $S =$ { ♟, ♔ }。

□

3.2 SAT 归约

接下来，我们将讨论此前的归约未涵盖的某些棋子类型所对应的不可被吃限制。本节中的归约来自 3SAT 的一个特殊情形：每个文字的出现次数至多为两次。Tovey [Tov84 在定理 2.1]中证明了该情形为 NP 困难。

这里从平面 3SAT 变体进行归约较为方便。De Berg 和 Khosravi [dBK10 在定理 1] 中证明了平面单调 ***3SAT***[2]是 NP 困难的。在这种 3SAT 变体中，包含每个子句一个顶点、每个变量一个顶点、连接每个子句与其中所含变量的边，以及一条经过所有变量的哈密顿回路的图，必须具有平面嵌入。此外，在该嵌入中，包含正文字的所有子句必须位于哈密顿回路内，而包含负文字的所有子句必须位于其外；尤其是，每个子句要么完全由正文字组成，要么完全由负文字组成。（这个问题更准确的名称是“Sided Var-Linked Planar Monotone 3SAT” [Fil19]。）等价地，可以设想将变量沿平面中的一条直线排列，所有正子句（及其边）位于直线一侧，所有负子句位于另一侧。

我们还希望满足每个文字最多出现两次的条件。在附录 B 中，我们证明，结合了这两项限制的问题——每个文字最多出现两次的平面单调 3SAT——仍然是 NP 困难的。

[2]这里的***3SAT***指每个子句包含 *at most* 三个变量的合取范式可满足性，而不是每个子句包含 *exactly* 三个变量的可满足性（E3SAT）。

定理 3.9。 对于任意 $T \in \{\text{♟}, \text{♚}\}$, Solo-Chess($\{T^1, \text{♞}\}$)，该问题都是 *NP* 困难的。

证明。我们将问题归约自每个文字最多出现两次的平面单调 3SAT（引理 B.1）。参见图 5。

对于每个变量 x_i，我们构造一个变量组件，其中包含两条兵可通行路径。每条路径上都有两个对应于该变量文字的文字骑士（以绿色绘制）：左侧路径上的文字骑士对应正文字 x_i，右侧路径上的文字骑士对应负文字 $\neg x_i$。这些文字骑士通过对应于 3SAT 子句的骑士不相交路径彼此连通。每个文字最多出现两次这一限制确保我们有足够的文字骑士来构造该 3SAT 实例。

假设兵向上吃子。孤立的兵或王必须从棋盘底部向顶部穿行，依次经过每个变量组件。在每个变量组件处，它都面临两条路径的选择，因此可以经过该变量的正文字骑士或负文字骑士，但不能两者都经过。（王可以沿一条文字路径向上走，再沿另一条文字路径向下走，但这样就会被困住，无法到达构造顶部的终点骑士。）为了吃掉用于子句的骑士，每个子句中至少有一个文字骑士必须被兵或王经过。所有其他骑士，包括未用于子句的文字骑士，都与构件的两侧连通，因此无论选择哪条路径，都可以吃掉它们。因此，当且仅当 3SAT 实例可满足时，单人国际象棋实例才可解。

□

本节中的其他归约也类似；我们只需在每种情况下设计合适的变量构件。

定理 3.10. Solo-Chess($\{\text{♚}^1, \text{♝}\}$)是 *NP* 困难的。

证明。 我们从每个文字至多出现两次的平面单调 3SAT 归约而来，方法与定理 3.9 类似。不同之处在于，我们使用图 6 所示的变量组件。我们添加象，将每个子句中的文字象连接起来；或者将未使用的文字象同时连接到两条路径上，使其无论选择哪条路径都能被吃掉。

□

定理 3.11。 对于任意 $T \in \{\text{♝}, \text{♜}, \text{♛}\}$, Solo-Chess($\{\text{♞}^1, T\}$)，该问题都是 *NP* 困难的。

证明。 我们从每个文字至多出现两次的平面单调 3SAT 归约而来，方法与定理 3.9 类似。不同之处在于，我们使用图 7 所示的变量组件（针对 $T =$ 的情况）。请注意，即使将图中的后♛替换为车或象，所有非文字棋子仍与两条路径连通。我们添加后/车/象，将每个子句中的文字后/车/象连接起来；或者将未使用的文字后/车/象同时连接到两条路径上。

□

4 个开放问题

表 1 中的结果留下了两个情形的复杂性问题尚未解决：$\{\text{♚}^1, \text{♜}\}$ 和 $\{\text{♟}^1, \text{♜}\}$。我们猜测这两个问题都可以在多项式时间内求解，主要原因是王或兵无法从车旁边溜过，但仍需将第2.3节中的算法加以推广。同样，能否将该算法推广到非对称棋子，目前也是一个开放问题。

在之前的论文 [AMM22], 中， Solo-Chess($\{\text{♞}\}$, $O(1)$) 和 Solo-Chess($\{\text{♚}\}$, $O(1)$) 的复杂性仍未解决。或许可以尝试证明图上更多非阻挡棋子类型的困难性，该图可能还受到最大度数为8或为网格图的约束。

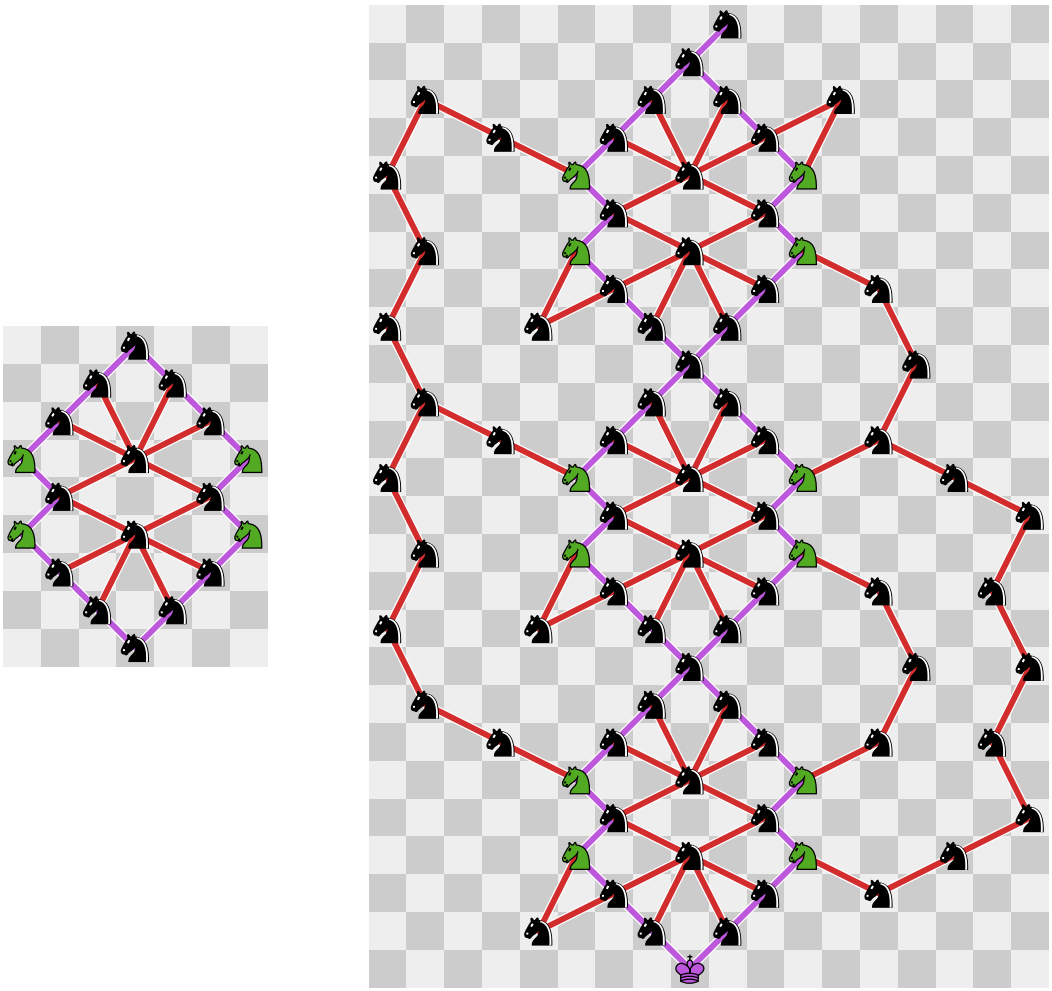


图 5： Solo-Chess({♚^{1}, ♞}) 的变量组件（左）及归约示例输出（右）。该实例对应于公式 $(x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_2 \vee \neg x_3)$。每个子句中至少必须访问一个绿色文字骑士。

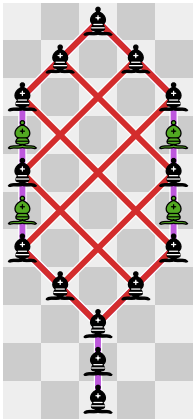


图 6： Solo-Chess({♚^{1}, ♝}) 的变量组件。

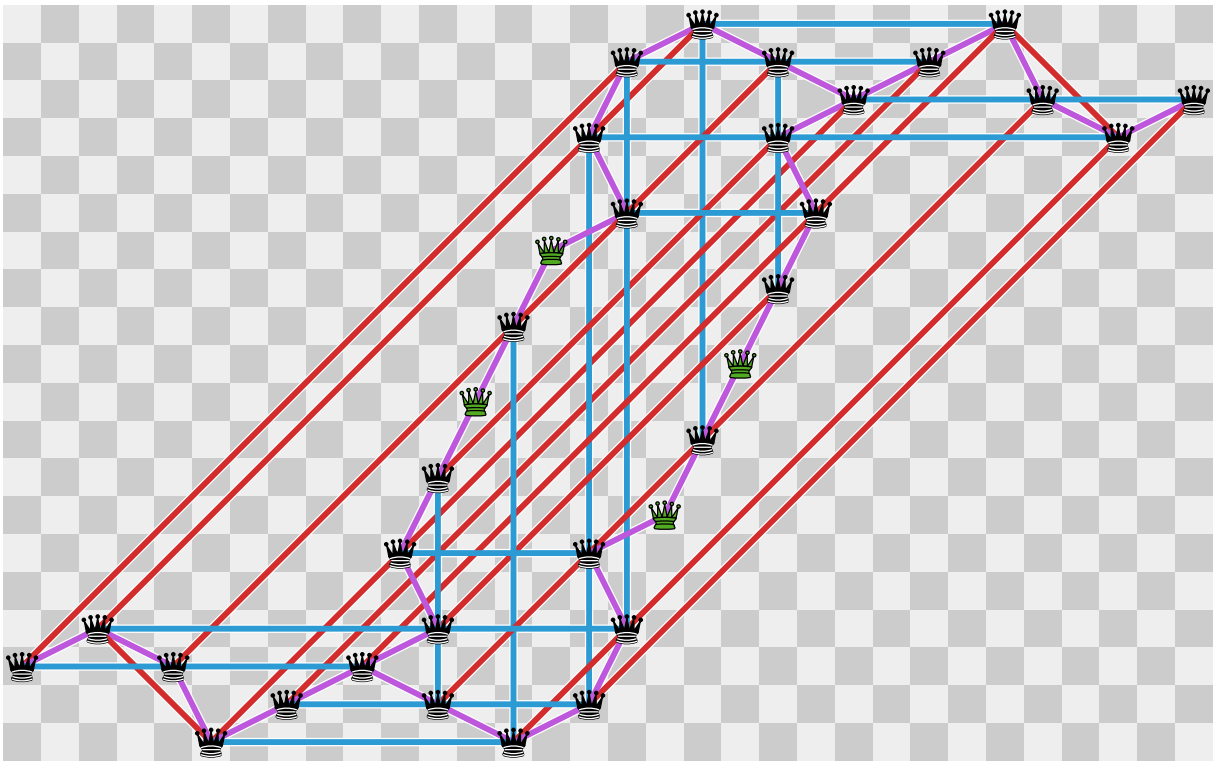


图 7： Solo-Chess({♞[1], ♛}) 的变量组件。

最后，尽管单人国际象棋谜题并非旨在确保解唯一，但确定该问题是否为 ASP 完全，以及统计解的数量是否为 #P 完全，仍然颇有意义。我们的部分归约是保解的，但并非全部如此。

致谢

这项工作始于 2019 年春季，在参加由 Erik Demaine 于麻省理工学院开设的“算法下界：与困难性证明共乐”（6.892）课程的学员们进行的数次长时间解题讨论中。感谢其他学员提供的见解与贡献。特别感谢 Dylan Hendrickson 就一种棋子类型的算法进行了有益的讨论。

本文的大多数插图均使用 SVG Tiler [https://github.com/edemaine/svgtiler] 绘制。国际象棋棋子图像基于维基百科上的 https://commons.wikimedia.org/wiki/Standard chess diagram，由 Colin M.L. Burnett 绘制，并依据 BSD 许可证授权。

参考文献

[AMM22] N. R. Aravind、Neeldhara Misra 和 Harshil Mittal。《即使对单人而言，国际象棋也很难》。载于 Pierre Fraigniaud 和 Yushi Uno 编，《第 *11* 届趣味算法国际会议论文集》，第 226 卷（*LIPIcs*），第 5:1–5:20 页，2022 年。[BDHW20] Josh Brunner、Erik D. Demaine、Dylan H. Hendrickson 和 Julian Wellman。《逆向分析与助杀国际象棋问题的复杂性：即使合作国际象棋也很难》。

载于 Yixin Cao、Siu-Wing Cheng 和 Minming Li 编，《第 *31* 届算法与计算国际研讨会论文集》，第 181 卷（$LIPIcs$），第 17:1–17:14 页，2020 年。[Che] Chess.com。单人国际象棋。https://www.chess.com/solo-chess。[dBK10] Mark de Berg 和 Amirali Khosravi。平面上的最优二元空间划分。载于 My T. Thai 和 Sartaj Sahni 编，《计算与组合数学》，第 216–225 页，柏林、海德堡，2010 年。Springer Berlin Heidelberg。[Edm67] Jack Edmonds。最优分支树。美国国家标准局研究杂志 *B* 辑，71:233–240，1967 年。[Fil19] Ivan Tadeu Ferreira Antunes Filho。布尔可满足性问题变体的特征。硕士论文，麻省理工学院，2019 年。[FL81] Aviezri S. Fraenkel 和 David Lichtenstein。计算国际象棋的完美策略需要指数级时间 n。组合理论杂志 *A* 辑，31:199–214，1981 年。[GGST86] Harold N. Gabow、Zvi Galil、Thomas H. Spencer 和 Robert Endre Tarjan。求解无向图和有向图中最小生成树的高效算法。$Combinatorica$, 6(2):109–122，1986 年。[IPS82] Alon Itai、Christos H. Papadimitriou 和 Jayme Luiz Szwarcfiter。网格图中的哈密顿路径。*SIAM* 计算杂志，11(4):676–686，1982 年。[KR92] Donald E. Knuth 和 Arvind Raghunathan。相容代表元问题。*SIAM* 离散数学杂志，5(3):422–427，1992 年。[KV06] Bernhard Korte 和 Jens Vygen。生成树与树形图。载于《组合优化：理论与算法》，第 119–141 页。Springer，2006 年。[PV84] Christos H. Papadimitriou 和 Umesh V. Vazirani。与旅行商问题相关的两个几何问题。算法杂志，5(2):231–246，1984 年 6 月。[Sto83] James A. Storer。论国际象棋的复杂性。计算机与系统科学杂志，27(1):77–100，1983 年。[Tov84] Craig A. Tovey。一个简化的 NP 完全可满足性问题。离散应用数学，8(1):85–89，1984 年。

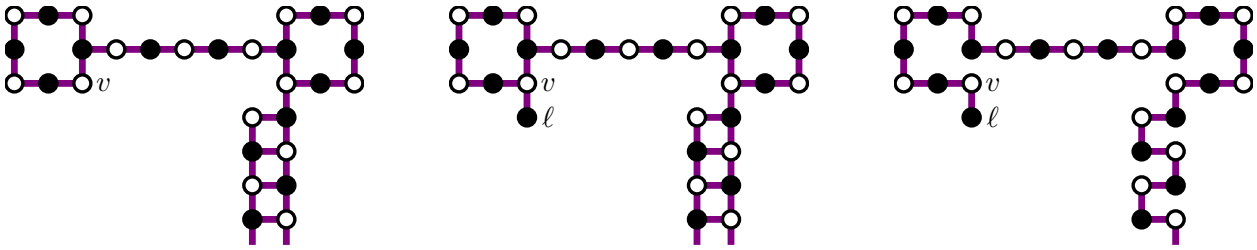
最大度为 3 的网格图中的哈密顿路径（指定起点/终点顶点）

Itai、Papadimitriou 和 Szwarcfiter [IPS82] 证明了，判断一个网格图是否存在指定起点和终点的哈密顿路径是 NP 困难的。[3] Papadimitriou 和 Vazirani [PV84]证明了，判断一个最大度为 *3* 的网格图是否存在哈密顿路径（不指定起点/终点）是 NP 困难的。这两项结果都不完全符合本文的需求：

引理 A.1. 判断一个最大度为 *3* 的网格图是否存在一条满足以下条件的哈密顿路径是 *NP* 困难的

1. 从指定的起始顶点 s 开始，该顶点的度为 1 （但未指定终止顶点）；或

[3]他们还描述了如何将这个问题归约为判断一个图是否存在哈密顿路径（不指定起点/终点），但他们的归约方式（在指定的起点和终点顶点各连接一个度为 1 的顶点）并不能显然地保证所得图仍为网格图。



(a) 基于 [PV84，图12(b)]的带连接销的哑铃　(b) 添加一个度为 1 的顶点　(c) 得到的哈密顿路径

图 8：从[PV84] 到最大度为 3 的网格图中的哈密顿路径的困难性归约，在 (a) 中的构件有两个副本。该图是哈密顿图，当且仅当 (b) 中的修改也是如此；这迫使哈密顿路径呈现出 (c) 中的形态，特别是从 ℓ 开始或在此终止。

2. 从指定的起始顶点 s 出发，并在指定的终止顶点 t结束；这两个顶点的度均为1。

证明。 我们修改 Papadimitriou 和 Vazirani [PV84]的证明。他们的证明将问题归约自平面有向图 G_1 ，其中每个顶点的入度为 2、出度为 1，或入度为 1、出度为 2。

他们对 G_1 [PV84，图 2] 所作的第一项修改构造了一个无向 图 G_2 ，使得 G_1 具有哈密顿回路，当且仅当 G_2 具有哈密顿路径。该修改的一部分 [PV84，图2(b)] 将 G_1 中的一个顶点 v_1 替换为一个包含两个度为 1 的顶点 vin'_1 和 vout_1的四顶点构件。显然，如果 G_2 具有哈密顿路径，那么它必然从 vin'_1开始，并在 vout_1结束。

接下来，他们的证明构造了一个最大度为 3 的网格图 G'_4 ，使得 G'_4 存在哈密顿路径，当且仅当 G_2 存在哈密顿路径。 G'_4 本质上是 G_2 （结果证明它是二分图）的网格绘图，经过常数倍放大，并将每个顶点和边替换为加粗的构件。 G_2中度为 1 的顶点 vin'_1 和 vout_1, 在 G_4 中分别映射为一个“哑铃”（两个长度为 8 的圈通过一条长度为 6 的路径相连），它通过“销连接”（单条邻接边）连接到一根“触手”（一个在转弯处带有长度为 8 的圈的 $2 \times n$ 矩形）；见图 8a。由于哑铃仅通过一条边（销连接）与图的其余部分相连，任何哈密顿路径都必须在每个这样的哑铃内开始或结束。特别地，我们可以选择哑铃内的某个起始或终止顶点，使其为连接两个圈的路径上、远离销连接那一端相邻的两个顶点之一；我们在图 8a 中将这样的顶点标记为 v 。根据 [PV84、引理] 或图 8c，如果指定该顶点 v 为起始或终止顶点，就能保留 G_4中哈密顿路径存在这一性质。该顶点 v 的度为 2，且它旁边有一个网格点（位于图 8a 中 v 的下方），没有相邻顶点，因此我们可以如图 8b 所示在该点添加一个度为 1 的顶点 ℓ ，使 v的度从 2 增至 3（同时保持最大度为 3）；此时，度为 1 的顶点 ℓ必定是任何哈密顿路径的起点或终点。图 8c 展示了任何哈密顿路径都必须采用的局部配置（尤其验证了哈密顿路径存在这一性质得以保留）。我们可以将来自 vin'_1 的顶点指定为起始顶点，并可选择将来自 vout_1 的顶点指定为终止顶点，从而证明这两种变体都是 NP 困难的。

□

出现次数受限的 B 侧变量链接平面单调 3SAT

引理 B.1。 *Sided Var-Linked Planar Monotone 3SAT* 是 *NP* 困难的，即使每个文字最多出现两次、每个变量最多出现三次。

Knuth 和 Raghunathan [KR92] 指出，任何变量链接平面 3SAT 实例都有直线布局。也就是说，可以将子句和变量绘制为平面上的水平线段，并用垂直线段连接关联变量与子句，使得除此之外，线段之间不会相交，且所有变量都位于同一条水平线上。因此，这一结果也适用于提供了这种直线布局的问题版本。

证明。 该归约来自 Sided Var-Linked Planar Monotone 3SAT。参见图 9。我们证明，每个变量都可以替换为一组新变量和子句，从而构造出一个等可满足的 Sided Var-Linked Planar Monotone 3SAT 实例；其中每个新变量的出现次数至多为三次，且其中至多两次具有相同符号。下文假设每个变量至少各有一次正、负符号的出现；任何缺少其中一种符号出现的变量都可以删除，而不影响可满足性。

设 x 是一个变量，它在子句 N_1、……、N_n 中有 n 次负出现，在子句 P_1、……、P_m 中有 m 次正出现。我们将 x 替换为两列变量 x_1、……、x_{n+m} 和 y_1、……、y_{n+m-1}。对于每个 $1 \le i \le m+n-1$，我们添加一个新的正子句 $x_i \vee y_i$ 和一个新的负子句 $\neg y_i \vee \neg x_{i+1}$。最后，对于每个 $1 \le k \le n$，我们将子句 N_k 中 $\neg x$ 的每次出现替换为 $\neg x_k$ 的一次出现；对于每个 $1 \le j \le m$，我们将子句 P_j 中 x 的每次出现替换为 x_{n+j} 的一次出现。至此，构造完成。每个新变量 x_i 或 y_i 至多出现三次，且相同符号至多出现两次。从图 9 可以看出，该构造保留了带侧平面性。我们必须证明所得实例与原实例等可满足。

在一个方向上，设 v 是原实例的一个满足赋值（即从变量到 $\{0,1\}$ 的映射）。定义一个新赋值 w，使得对于每个原变量 x，$w(x_i) = v(x)$ 且 $w(y_i) = \neg v(x)$。那么 w 就是新实例的一个满足赋值。

在另一个方向上，设 w 是新实例的一个满足赋值。定义原变量上的一个赋值 v，令 $v(x) = w(x_n)$，其中 n 是 x 的负出现次数。我们声称 v 是原实例的一个满足赋值。为此，只需证明对于每个具有 n 个负出现次数和 m 个正出现次数的变量 x，

$$\begin{aligned} w(x_k) &\ge v(x) \\ w(x_{n+j}) &\le v(x) \end{aligned} \tag{B.1}$$

对于 $1 \le k \le n$ 和 $1 \le j \le m$。因为这样一来，任何通过满足 $w(x_k) = 0$ 或 $w(x_{n+j}) = 1$ 而得到满足的子句 N_k 或 P_j，也会因 $v(x)$ 的取值而得到满足。

子句 $x_i \vee y_i$ 和 $\neg y_i \vee \neg x_{i+1}$ 共同要求 $x_{i+1} \to x_i$ 对于 $1 \le i \le m+n-1$。因此，w 在 $x_1, \ldots, x_{n+m}$ 上弱单调递减。由于 $v(x) = w(x_n)$，这立即得出 (B.1)。因此，Sided Var-Linked Planar Monotone 3SAT 的新实例与原实例等可满足。 □

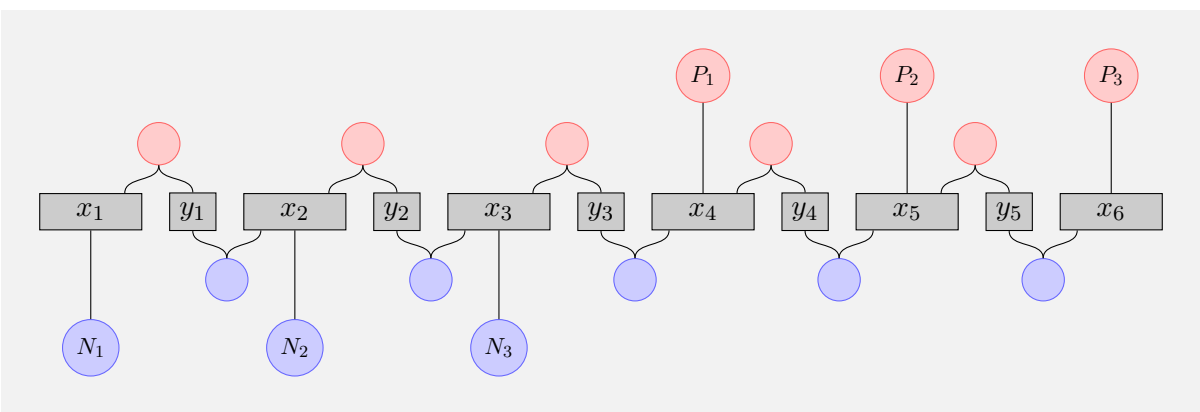


图 9：对有侧变量链接单调平面 3SAT 中的变量进行变换，以减少出现次数。变量线（红色）上方的子句为正，变量线（蓝色）下方的子句为负。上图：一个变量有三次正出现和三次负出现。下图：一组等价的子句和变量。每个变量最多出现三次，且相同符号的出现次数最多为两次。