arXiv:2611.33127v1 [cs.DM] 24 Nov 2026

NoGo 的胜负判定是 PSPACE 完全的

Kyle Burke [1]、Alfie Davies* [2] 和 Craig Tennenhouse [3]

[1]佛罗里达南方学院，美国[2]达勒姆大学计算机科学系，英国[3]新英格兰大学数学与数据科学学院，美国

摘要

我们解决了 Burke 和 Hearn 提出的一个未决问题 [5]，并证明组合博弈 NoGo 在网格上的复杂度是 PSPACE 完全的。这是该游戏通常进行的几何形式，包括 2011 年班夫研讨会上的锦标赛。除这一结果外，我们还证明，在网格上采用反常玩法时，该问题同样是 PSPACE 完全的。我们使用从有界双人约束逻辑出发的归约，证明了这两个结果。最后，我们证明，在连通二分次三图上采用正常玩法时，该问题也是 PSPACE 完全的。

1 引言

组合博弈论研究的是没有机会行动、信息完全的双人博弈 [1, 2, 14]。NoGo 是这类博弈之一，是围棋的一个变体，其中提子（包括自提）和停着均属非法。2005年，John Moore 在 Sensei's Library 上以“反阿塔里围棋”之名介绍了这一游戏的早期版本 [12]。Neil McKay 则独立发明了 NoGo 2011 [10, p. 396]。

为完整起见，我们简要介绍 NoGo 的规则：黑方和白方轮流在方格棋盘的空点上放置一枚己方颜色的棋子。一个 组是同一颜色棋子的一个连通分量。一个组的气是与其相邻的空点。如果一步棋能使每个组（无论属于哪种颜色）都至少有一口气，那么这步棋就是合法的。（在围棋中，若一个组失去所有气，就会被提子，其棋子也会从棋盘上移除。）在正常玩法中，玩家若轮到自己时无法走出合法着法，就会输；而在反常玩法中，玩家若轮到自己时无法走出合法着法，则获胜。[1] 相同的规则定义了图

*由 Leverhulme Trust研究项目资助RPG-2024-182 提供支持。[1]NoGo 可在 https://kyleburke.info/DB/combGames/noGo.html 进行正常玩法，也可在 https://kyleburke.info/DB/combGames/noGoMisere.html 进行反常玩法。

上的 NoGo 可推广至任意有限图，其中顶点对应棋盘上的点，边表示邻接关系。

算法组合博弈论中的一个基本问题是可胜性：也就是说，判断在给定局面 [7]下，轮到行动的玩家是否有必胜策略。本文分析黑方先手时，在常规玩法和反常玩法下是否有必胜策略。Burke 和 Hearn 报告称，Graph NoGo 在平面图上的 NP困难性已于 2011年班夫研讨会 上确立，当时还举办了 NoGo锦标赛 [5, 第 408] 页。他们后来证明，该问题在平面图上是 PSPACE 完全，并提出在方格棋盘上是否也成立 [5,定理 4 和开放问题 1]。在 Buchin、Hagedoorn、Kostitsyna 和 van Mulken 证明点与方格为 PSPACE 完全后，这个开放问题再次受到关注 [3]。我们的主要结果对他们的问题给出了肯定回答，适用于从空棋盘开始经对弈可达的局面。结合近期关于著名组合博弈反常玩法版本复杂度的研究 [4], ，我们还证明了反常玩法下同样具有这一困难性。

定理1.1： *NoGo* 在空的 $n \times n$ 方格棋盘上通过合法着法可达的局面中，无论采用正常玩法还是反常玩法，都是 *PSPACE* 完全的。

Burke 和 Hearn 首先将有界双人约束逻辑（B2CL）归约为平面 Col，再将平面 Col 归约为平面 Graph NoGo [5]。对于定理1.1，我们直接在方格棋盘上实现 B2CL 小装置。我们用导线连接这些小装置；导线通常由一串白子构成，每颗白子有两口气，分别位于两个小装置中。如果黑方填掉其中一口气，那么无论导线有多长，黑方都能在另一端落子提掉这串棋子（这是非法着法）。

尽管 Burke 和 Hearn 在关于 Graph NoGo 的平面图困难性结果中没有明确说明，但检查他们的小装置归约可以发现，该问题在最大度不超过 11 的平面图上也是困难的。定理1.1表明，该问题在最大度不超过 4 的平面图上是困难的。困难性是否会延续到最大度不超过 3 的图（即次三度图）？我们证明答案是肯定的，但不要求图具有平面性。

定理1.2。 在连通二分次三图上，*Graph NoGo* 在正常玩法下是PSPACE-完全的，即使对于其博弈树为公平的局面也是如此。

在 Moore 于 2005年 首次发布有关反阿塔里围棋（即 NoGo）的帖子时，他指出，这种游戏“玩起来很像点与方格” [12]。事实证明，这一观察极具预见性。在点与方格的对偶描述中，每个方格变成一枚硬币，每条边变成一根线；画出方格的一条边，就会切断那根线。这一概念可以推广到任意（多重）图，形成 Strings-and-Coins；其正常玩法变体称为 Nimstring。具体来说：

我们有称为硬币的顶点，它们之间的边称为线，玩家轮流移除边（即切断线）；如果某位玩家释放了一枚硬币，就可以再玩一回合。

为证明定理 1.2，我们从一个称为 Coins-are-Lava 的进一步变体归约而来。在这个游戏中，玩家像之前一样轮流剪断弦，但剪断连到任一枚硬币的最后一根弦的玩家会立即输掉游戏。（分数无关紧要。）在简单图上，这就是 Gallant、Gunther、Hartnell 和 Rall 所研究的“删边游戏”；他们分析了该游戏在路径和具有特殊对称性的图上的情形 [8]。Demaine 和 Diomidova 在 [6,定理 1] 中证明，Coins-are-Lava 是 PSPACE 完全的（在他们的一般多重图设定下）。我们改造他们的构造，得到 Coins-are-Lava 的连通二部图实例，进而得到定理 1.2 所断言的次三度图实例。基本思路是将每枚硬币替换为一条由棋子构成的路径，使其形成一个组；再将每根弦替换为两个组（一个白方组、一个黑方组）共享的一口气。这样，剪断连到一枚硬币的最后一根弦，就等同于占据一个组的最后一口气。

根据定理 1.2，通过一种直接的构造，我们还可以在次三度图的每条边都进行统一的 $\ell \geq 1$ 次细分后，实现完全相同的公平组合游戏树。这使我们的结果与 Johnson 等人针对由禁子图定义的图类提出的 C123 框架联系起来。该框架的第一个条件 (C1) 涉及有界树宽图类上的可解性。我们在此建立了所需的两个困难性条件：次三度图上的困难性 (C2)，以及对任意大的 ℓ，这些图的统一ℓ次细分上的困难性 (C3)。由此，立即可从 [11,定理 4 和 6] 的证明得出：对于每个具有一个非路径且非细分爪图连通分量的固定图 H，Graph NoGo 在H-子图自由图上是 PSPACE 完全的。因此，此论证未涵盖的情况仅限于 H 是路径和细分爪图的不交并。

2 方格网格上的正常玩法

为了证明正常玩法下的 NoGo 是 PSPACE-困难的，我们从 B2CL 进行归约。从 B2CL 归约时，与其直接从一般的 B2CL 局面归约，更常见的做法是从构造于 [9] 中的困难实例（如 [5, 4]）进行归约。为此，我们需要构造用于实现变量、与、或、分裂和选择等作用的小装置，同时还需要构造一个目标装置及其他装置，使整个系统能够协同运作。这样做的原因在于，借助这些小装置，B2CL 能够赋予我们平面性，而这对于网格上的 NoGo 是必需的。尽管这类归约看起来像是从公式博弈（尤其是正合取范式）进行的归约，但实际上它们源自 B2CL [9]。

所需的五种小装置有类似公式的解释。变量

必须在一名玩家（在我们的设定中是黑方）先在其上落子时传递激活信号；如果另一名玩家（白方）先落子，则传递未激活信号。随后，玩家必须按如下方式落子：与在且仅在两个输入都激活时产生激活信号；或在且仅在至少一个输入激活时产生激活信号；分裂在输入激活时产生两个激活的输出，否则两个输出都未激活。最后，选择在输入未激活时产生两个未激活的输出；否则由黑方选择其中一个输出使其激活。

我们构造的总体方案是：由黑棋包围的白棋路径导线将各个小装置连接起来。如果导线处于激活状态，那么输入端的白棋路径仍会有一口气，使黑方可以在导线的输出端落下一枚黑棋。黑方必须在每个小装置中落下一定数量的棋子，否则就会输。通常，只要能够堵住输入导线，黑方就不必落子封堵输出导线。

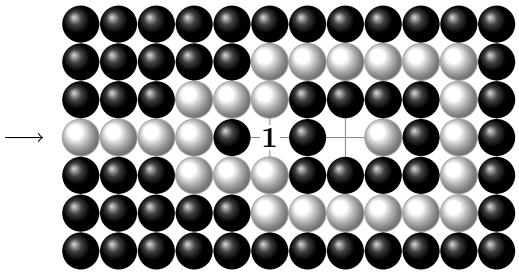


图1：目标组件。输入从左侧进入。

如果黑方在每个组件中按照规定的次数落子，并且能够在我们在图1中描述的目标组件上落子，那么黑方将赢得这局游戏。在该组件中有两个空位，但只有空位1可以落子，且只有黑方能够落子；此外，只有当输入白色导线的另一端有一口气时，黑方才能在空位1落子。因此，只有当输入到该组件的信号处于激活状态时，黑方才能进行这一步。

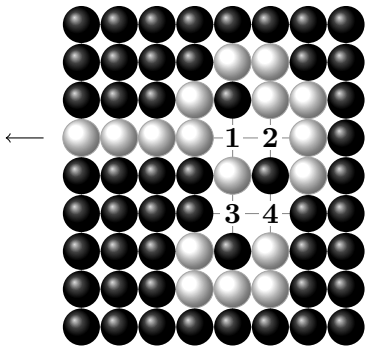


图2：变量组件。输出信号从左侧离开。

接下来我们描述图2所示的变量组件。这里有四个空位（标记为1、2、3和4），但只有黑方可以在1和3上落子，只有白方可以在2和4上落子。此外，1和2中至多只能落子一个，3和4中至多只能落子一个，1和3中至多只能落子一个，2和4中至多只能落子一个。因此，每位玩家在该组件上至多可以落子一次。如果黑方先下，那么选择1而不是3对黑方始终没有好处，因为在1处落子会使输出失去激活状态。如果白方先下，那么白方应当选择在4处落子，这样黑方在该组件上唯一可下的位置就是1，从而使输出失去激活状态。

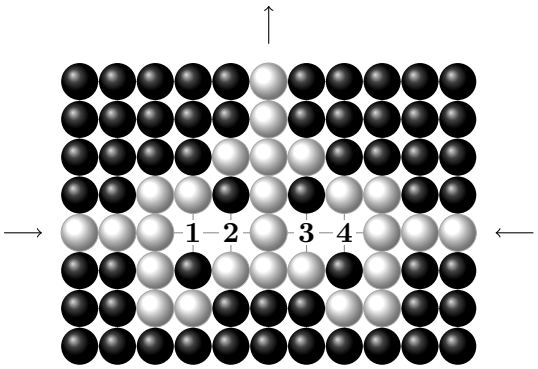


图3：或组件。输入位于左侧和右侧；输出从顶部引出。

在图3所示的或组件中，有两对开放位置，标记为1、2、3和4；这些位置都只能由黑方落子，并且每对中只能落子一次。如果两个输入都未激活，则黑方只能在2和3处落子，从而使输出失效。如果任一输入（或两者）处于激活状态，则黑方至少可以在1和4中的一个位置落子，同时使输出保持激活状态。

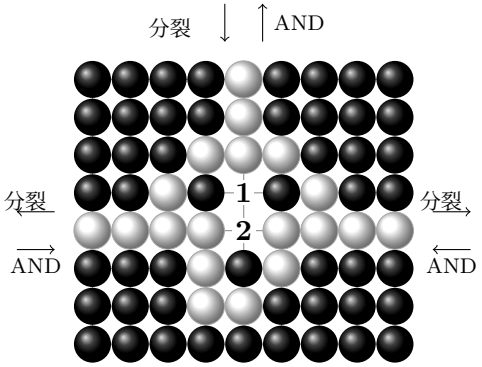


图4：与组件和分裂组件。对于与组件，输入位于左侧和右侧；输出从顶部引出。对于分裂组件，输入从顶部进入，输出从两侧引出。

如图4所示，与组件只有一对开放的空位，

分别标记为1和2。只有黑方可以在这些位置落子，而且只能选择其中一个空位。如果两个输入都处于激活状态，黑方就可以在2处落子并激活输出。反之，如果任一输入处于未激活状态，黑方就必须在1处落子，从而使输出失活。

这里无需单独设置分裂组件，因为我们可以复用与组件（图4），并交换输入和输出。如果输入处于激活状态，那么在1处落子即可激活两个输出。如果输入处于未激活状态，黑方就必须在2处落子，从而使两个输出都变为未激活。

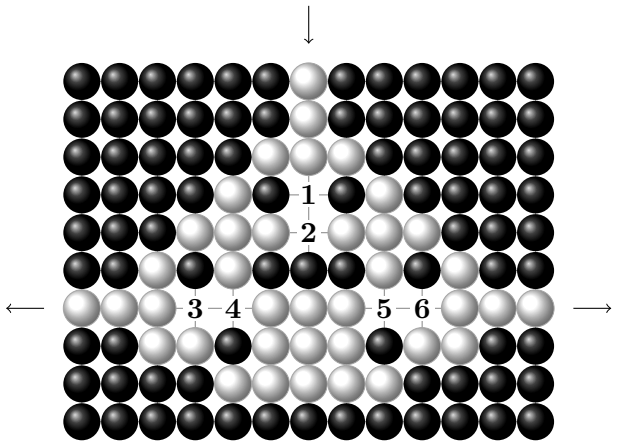


图5：选择组件。输入从顶部进入，输出从两侧引出。

图5所示的选择组件包含三对空位，分别标记为1和2、3和4、5和6。只有黑方可以在这些空位落子，而且每对空位只能落子一次。如果输入处于未激活状态，黑方就必须在2处落子，随后必须在3和6处落子，从而使输出变为未激活。如果输入处于激活状态，黑方可以在1处落子，然后选择在4和6处落子（激活左侧输出并使右侧输出变为未激活），或在3和5处落子（激活右侧输出并使左侧输出变为未激活）。

定理 *1.1*在正常玩法下。 NoGo 属于 PSPACE，因为在一个 $n \times n$ 网格上，博弈树的深度和分支数最多为 n^2，且可以在多项式时间内检查走法是否合法，因此我们可以在 PSPACE 内遍历整个博弈树。

为证明 PSPACE-困难性，我们将其从 B2CL 归约而来。利用本节截至目前所述的小装置，我们可以构造一个局面，使黑方能够进行次数等于

$$x = a + 2b + c + d + 3e,$$

其中， a是变量组件的数量， b 是或组件的数量，c 是与组件的数量， d 是分裂组件的数量， e

是选择组件的数量。在变量组件上完成对弈后，黑方将在构造的其余部分先行，因此，如果我们让白方拥有$x-a$步自己的棋，那么黑方获胜当且仅当他们能在目标组件上获得额外的一步。（白方还将在变量组件上进行 a 步。）我们可以通过放置一大片带有 $x+1-a$ 个单点空位的白子来为白方提供这些步数。只有白方能在这些空位中落子，而且他们可以使用其中除一个以外的所有空位。

因此，黑方先手能够获胜，当且仅当其能在每个组件以及目标组件上各按规定次数落子；这也正是其能在源 B2CL 局面中获胜的条件。

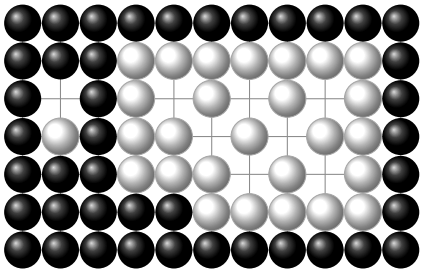


图 6：黑色背景内部的棋盘状白方储备区，以及左侧黑色背景的一个冻结气。在这个具体示例中，白方储备区用 26 个白子和 7 个内部空位替代了 33 个黑棋，使白方拥有 6 步，而黑方没有可走的步数。

由于白色线路可以将黑色背景分隔成若干区域，我们为每个黑色背景连通分量设置一个冻结气：它由一个与单个空点相邻的白色单子构成，周围全部是黑棋，如图 6 左侧所示；任何玩家都无法在此处落子。对于白方的 $x-a$ 等待步数，我们放置一个棋盘状白方储备区，其中有$x+1-a$个由白子包围、两两不相邻的内部空位，如图 6 右侧所示（当 $x-a=0$时，则为一个单独的非活动内部空位）。黑方永远无法在这些空位落子，而白方可以在最后一个空位变为非法之前，在其中任意 $x-a$ 个空位落子。

为了将构造放入一个黑白棋子数量相等的 $n\times n$ 网格中，我们用黑棋填充棋盘，并选择足够大的 n，使已占据点的数量为偶数，且小装置外有充足空间。接着，我们可以轻松地用“惰性”白子扩充白方预留区（以相邻的黑色填充棋子替换，不改变任何空点，也不切断黑色背景与其冻结气之间的连接），直到黑白棋子数量相等。最后，任何黑白棋子数量相等的合法 NoGo 局面，都可以从黑方先行的空 $n\times n$ 棋盘到达：删除任意棋子都能保持局面合法（其组中每个仍然存活的连通分量都会因腾出的点而多一口气），因此通过归纳法交替放置棋子

从空棋盘开始的每一步都保持合法，并最终得到一个黑方先行的局面。

□

3 方格网格上的反常玩法

为了证明 NoGo 的反常玩法版本是 PSPACE-困难的，我们将再次从 B2CL 归约，因此需要变量、与、或、分裂和选择组件，以及一个目标组件。

与常规玩法困难性证明一样，信号通过白棋导线传递，这些导线由黑棋“绝缘”，且两端的气都无法由白方落子。与常规玩法版本不同，导线中传递的激活信号方向相反：如果输入端被黑棋堵住，导线就是激活的，这意味着输出端必须保持开放。

我们重复使用常规玩法构造中的目标组件（见图1）。当且仅当与输入端相连的白方棋块此前有一个空气时（即处于未激活状态），黑方才有额外的一步棋。（在目标组件中，双方都不能在最右侧的空位落子，而白方永远不能在左侧的空位落子。）

这意味着，如果黑方能够激活通向目标的信号，就无法在目标处落子。只要黑方没有在别处作弊，缺少这最后一手就意味着黑方会比白方先无子可下，从而赢得比赛。

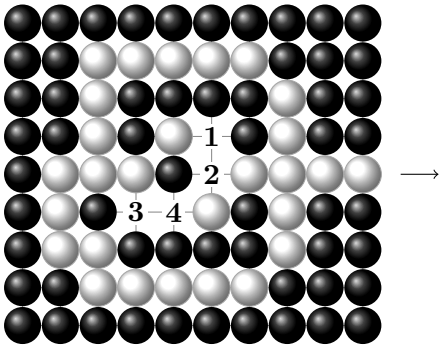


图7：变量组件。输出从右侧离开。

图7所示的变量组件有四个空位（标记为1、2、3和4），但双方玩家都不能在空位1或3落子。在空位2和4中，双方都更愿意在空位2落子。如果白方在此落子，输出就会处于未激活状态。如果黑方在此落子，输出就会处于激活状态。无论哪位玩家在空位2落子，另一位玩家都能在空位4落子，并在之后的对局中占据该位置。

图8展示了或组件，其中有三个空位（标记为1、2和3），都只能由黑方落子。如果任一输入处于激活状态，那么黑方

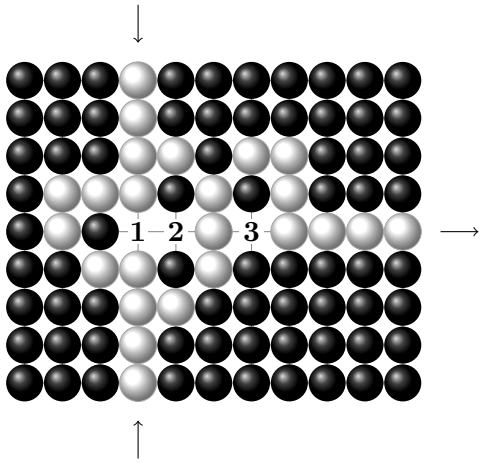


图8：或组件。输入从顶部和底部接入，输出从右侧引出。

只能在2或3落子，但不能两处都下。在这种情况下，黑方应在3落子，以保持输出处于激活状态。然而，如果两个输入都处于未激活状态，黑方则可以合法地在1和3都落子，也可以只在2落子。黑方会选择后者，因为这样可以少走一步，输出也会继续保持未激活状态。

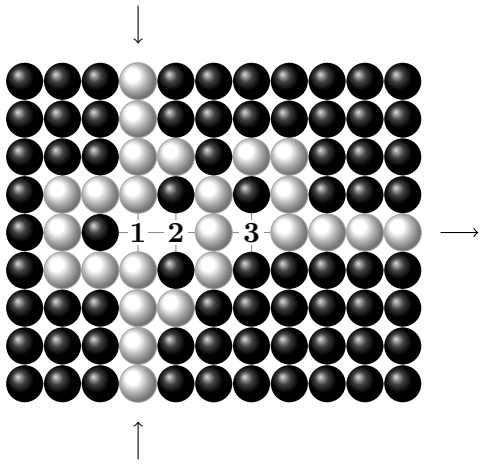


图9：与组件。输入从顶部和底部接入，输出从右侧引出。

与组件（如图9所示）与或组件非常相似，只是有一枚棋子从黑色变成了白色。这个组件也有三个空位（标记为1、2和3），只有黑方可以下。这次，如果任一输入未激活，那么1就可以下（因此下在3会使1处多出一步棋），所以黑方应下在空位2，让输出保持未激活。如果两个输入都已激活，黑方就可以下在3，使1和2都无法下

并激活输出。

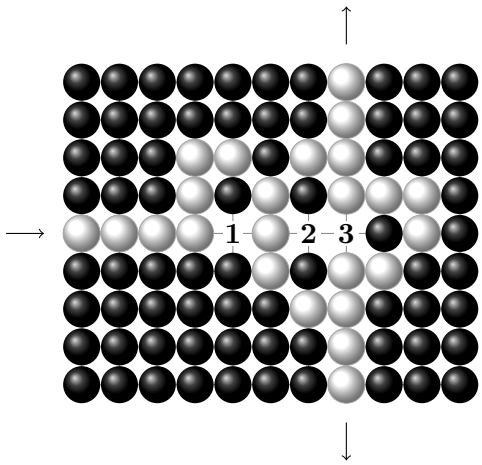


图10：分裂组件。输入从左侧进入，输出分别从顶部和底部引出。

分裂组件如图10所示，其中有三个仅黑方可以落子的空位，标记为1、2和3。如果输入处于激活状态，那么黑方在该组件中只有一步可走，可以落在2或3上。黑方应选择落在3上，以激活两个输出。如果输入处于未激活状态，那么三个空位都可以落子，因此黑方应选择落在2上（而不是分别落在1和3上），这样会使输出保持未激活状态。

选择组件如图11所示，其中有五个空位，但其中一个双方玩家都无法落子。其余四个（标记为1、2、3和4）都只有黑方可以落子。如果输入处于激活状态，那么1无法落子，黑方可以选择落在3或4上，激活相应的输出，并使2无法落子。如果输入处于未激活状态，那么黑方若落在3或4中的任意一个位置，1仍然可以落子，黑方就需要额外走一步。不过，如果黑方落在2上，那么3和4都无法落子，两个输出也都处于未激活状态。

定理 *1.1*的反常玩法证明。 NoGo 属于 PSPACE，因为在一个 $n \times n$ 网格上，博弈树的深度和分支数最多为 n^2，并且可以在多项式时间内检查走法是否合法，因此我们可以在 PSPACE 内遍历整个博弈树，就像常规玩法版本一样。

为证明 PSPACE-困难性，我们从 B2CL 进行归约。利用本节前文所述的小装置，我们构造出一个局面，使得黑方在非围棋目标小装置中预定的总着数为

$$x = a + b + c + d + e,$$

其中， a、 b、 c、 d、 e 分别是变量组件、或、与、分裂和选择组件的数量。我们使用以下方式为白方提供 $x - a$ 步额外着法

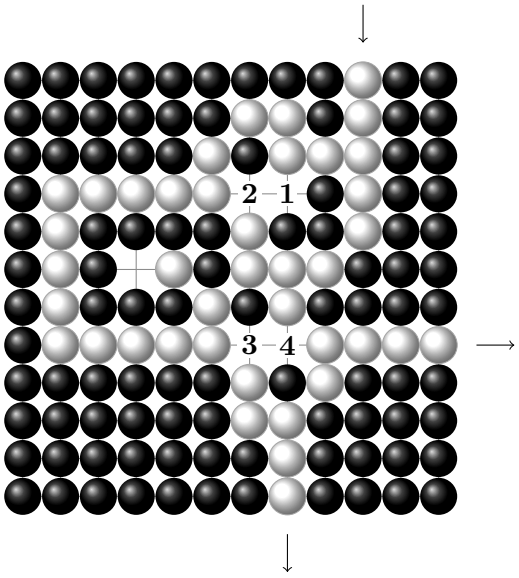


图 11：选择组件。输入从顶部进入，输出分别从右侧和底部离开。

一个带有 $x+1-a$ 个内部空位的棋盘状白方储备区（如图 6 所示），因此白方共有 x 步着法（其中a 步用于变量组件， $x-a$ 步用于储备区）。由于黑方先行，在反常玩法下，当且仅当黑方至多经过 x 步后无路可走时，黑方获胜。在每个逻辑门中，空位 2 在黑方于该逻辑门中落子之前始终合法（这迫使黑方在每个逻辑门中至少落子一次）；同样，白方也可以通过在黑方尚未使用的变量组件中落子于 2，并且仅在黑方于 2 落子后才落子于 4，迫使黑方在每个变量组件中至少落子一次。因此，当且仅当黑方能够在每个小装置中各用一步并将信号激活至目标（从而避免在那里额外使用一步）时，黑方才能至多用 x步耗尽着法；而这当且仅当黑方赢得源 B2CL 局面。

为了使归约后的局面能够从黑方走子的空白 $n \times n$ 网格到达，我们按照正常玩法证明中的相同方式扩展棋盘，并加厚白方储备。

□

4 次三次图困难性

我们通过从 Coins-are-Lava 归约来证明定理 1.2。回顾一下，
Coins-are-Lava 是一种在多重图上进行的无偏规则集：顶点称为硬币，边称为弦。在玩家的回合中，他们必须剪断一条弦（即删除一条边）。如果玩家移除了连接某枚硬币的最后一条弦，从而释放了这枚硬币，他们就会输。等价地，我们可以禁止释放硬币的着法，以采用正常玩法视角：如果玩家

无法进行合法着法，那么他们就会输。本节末尾，我们将证明以下引理；目前暂缓证明，以便先展示针对 Graph NoGo 的归约。

引理 4.1. *在正常玩法下，Coins-are-Lava 在没有孤立顶点的连通二分多重图上是 PSPACE 完全的，输入形式为这些图的线列表。*

引理 4.1 足以证明 Graph NoGo 的次三次图困难性。请记住，我们对这一归约采用的总体策略是用组替换硬币、用气替换线；我们利用引理 4.1 所保证的二分性，确保线表示由颜色相反的组共享的气。

定理 *1.2.* 设 G 是一个连通二分多重图（没有孤立顶点）。将一个部集中的硬币着为黑方，另一个部集中的硬币着为白方。将每个度数为 d 的硬币替换为一条由 $2d-1$ 个同色棋子组成的路径，并将其 d 条线分配给每隔一个棋子（第一个、第三个，依此类推），直至路径的最后一个棋子。然后，对于每条线，添加一个与分配给它的两个棋子相邻的空点。图 12 给出了一个具体示例。

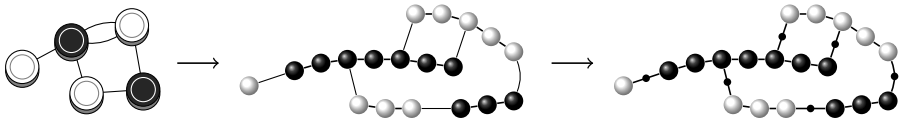


图 12：从二部的 Coins-are-Lava 到二部的 Graph NoGo。我们将硬币着为黑方和白方，仅用于示意二部划分。

所得棋盘必然是连通且简单的，即使 G 中的两个硬币之间由多条绳索连接也是如此。此外，每个棋子沿其路径至多有两个邻点，并且至多有一个对应于绳索的空点，而每个空点的度为 2；因此，该棋盘是次三次的。最后，我们可以通过将所有被分配了绳索的棋子放入一个部集，将其余棋子放入另一个部集，从而对该图进行正确的二着色；因此，该图是二部的。

最初，原多重图中的每枚硬币都由所得 Graph NoGo 局面中的一个组表示，而该组的气恰好是连接到该硬币的绳索。注意，没有硬币是孤立的，因此该局面是合法的 Graph NoGo 棋局。我们将一条绳索的关联气被任一玩家填满称为该绳索已被切断。我们断言，在整个对弈过程中，每个组的气恰好对应于其关联硬币的未被切断的绳索。

要理解这一点，可以考虑两个硬币之间代表一条线的点。它与每种颜色的一个棋子相邻。在这口气上落子（由

任一玩家落子）会与相邻的唯一一个同色组相连，并使两个组各失去一口气。它不会增加新的气，因为两个相邻点都不是空点。因此，只有当两个组都已有另一口气时，任一玩家才能在这口气上落子。

因此，在每一轮行棋之后，黑方和白方拥有相同的着法集合：博弈树是公平的。未切断的线按照我们所描述的方式决定着法。该构造的耗时与给定多重图的规模呈线性关系，因此由引理 4.1 可得 PSPACE-困难性。问题属于 PSPACE 显然成立，因为每一步着法都会填充一个点（并且我们可以通过寻找组及其气来检查是否存在合法着法）。因此，我们得到了所需结果。

□

由于一个博弈等于 0，当且仅当前一位玩家获胜，因此可以看出，定理 1.2 还告诉我们：判定这些公平局面（满足相关假设）中某一个是否等于 0，是 PSPACE 完全的。（这是因为 PSPACE 在取补运算下封闭。）

对于 $\ell \geq 1$，图的均匀 ℓ-细分 是将每条边替换为一条包含 ℓ个内部顶点（度为 2）的路径。

推论 4.2： 对于固定的 $\ell \geq 1$，在均匀 ℓ-细分的连通二分次三图上，*Graph NoGo* 在正常玩法下仍然是PSPACE完全的，即使对于博弈树是公平的局面也是如此。

证明。 考虑定理 1.2 的证明中构造的棋盘的一条边。它至少与一个棋子相接；如果它的两个端点都是棋子，则根据构造，它们具有相同的颜色。现在考虑这条边的 ℓ-细分，并将所有新点填上此前与该边相接的相同颜色。对每条边执行此过程，会将每个组扩展为与之前相同的气，因此整个博弈树得以保留。该构造只需多项式时间即可完成，因此结论成立。

□

定理 1.2 和推论 4.2 共同给出了 Johnson 等人 [11] 的条件 C2 和 C3。我们将说明由其框架和结果直接得到的禁子图推论。注意，细分爪图是通过对 $K_{1,3}$ 的每条边进行任意次数的细分得到的（即它是一个三足蜘蛛图）。如果一个图不含 $\mathcal{H}$ 的任何成员作为子图，我们称其为 $\mathcal{H}$ -子图自由图。

推论 4.3. 若 H 是一个固定的有限族，其中包含的图均非空，且都不是路径的不交并或细分爪图，则在不含 H 子图的图上采用正常玩法时，*Graph NoGo* 是 PSPACE-完全的。

证明。 选择 $\ell \geq 1$，使其大于 $\mathcal{H}$ 中每个图的顶点数。在推论 4.2 的假设所涉及的棋盘上，每个圈都包含多于 ℓ 个顶点，并且任意两个度为 3 的顶点之间的距离都大于

ℓ。因此，任何至多包含 ℓ 个顶点的连通子图都是一棵至多具有一个分支顶点的次三树。这种树必然是路径或细分爪图。因此，$\mathcal{H}$ 中没有任何成员是这些棋盘的子图，由推论 4.2 可得结论。

□

4.1 二部多重图上的 Coins-are-Lava

为证明引理 4.1，我们从 Schaefer 的正合取范式博弈 [13,定理 3.6] 进行归约。黑方和白方轮流认领变量，黑方将其认领的每个变量设为真，白方则将其认领的每个变量设为假。如果所得赋值满足给定公式，黑方获胜。

在完整证明该结果之前，我们先通过一个简单示例来说明这一构造；这样有助于更清楚地理解一般构造。考虑公式 $(x \vee y) \wedge z$。黑方先手时，可以通过先认领 z来确保获胜；白方认领 x或 y 中的一个后，黑方将认领另一个，从而得到一个为真的公式。

为了在 Coins-are-Lava 中表示这个公式博弈，我们写下三行：

$$A = (x, y, 1),$$
$$B = (z),$$
$$C = (0).$$

如果某一行的端点中多数为真，我们就称该行为真。于是，第A 行中额外的 1 意味着：当且仅当 x 或 y 中至少有一个为真时，该行才为真（即具有多数）；也就是说，当且仅当 $x \vee y$。第 B 行的逻辑值为 z。由于第 C 行始终为假，因此，当且仅当 A 和 B 都为真时，多数行才为真；这当且仅当 $(x \vee y) \wedge z$，而这正是我们试图模拟的公式。

我们为该公式构造的 Coins-are-Lava 局面如图 13 所示。每条双线表示一条可能包含多根线的*绳索*（即一条重边），其*宽度*为其中包含的线的数量。每一行都通过一条*导线* [6, Figure 5], 与根 r 相连；这条导线由两条在中间硬币处相交的绳索组成；r 边的绳索宽度为 N，行这一侧的绳索宽度为 N^2。随后，从每个行顶点引出一条宽度为 N^3 的绳索，连接到它的每个端点。例如，从A引出的三条绳索分别连接到 x、y 和 1。

需要牢记的过程如下：能够赢得公式博弈的玩家，首先会根据其必胜策略在变量路径上进行操作。如果黑方能够赢得公式博弈，这将使黑方拥有多数真行；如果白方能够赢得公式博弈，则会使白方拥有多数假行。随后，获胜玩家会在根处利用这一多数：他们切向错误值的行（即黑方切断通向假值的导线，而白方切断通向真值的导线），从而迫使最后剩下的线在 r

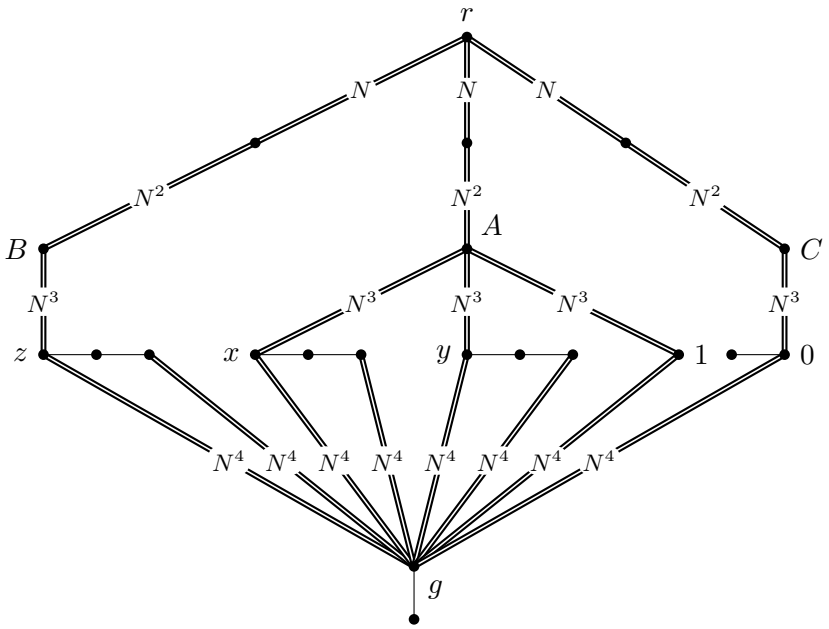


图 13：引理 4.1 证明中归约里 $(x \vee y) \wedge z$ 的构造。单线表示一根单根弦，而标记为 w 的双线表示宽度为 w 的绳索。

处通向一行所需的值。然后，他们在该行处再次运用同样的思路：例如，如果黑方获胜，那么黑方会迫使得到一行真行，而该行自身又拥有多数真条目，因此黑方可以再次操作多数，切断通向假值的导线；这样，黑方就会使该行最后剩下的线指向真值。现在，奇偶性已经设置为：真条目对应奇数次切割，而假条目对应偶数次切割，因此无论哪种情况，公式博弈的获胜者都会进行最后一次切割。让我们看看黑方如何在我们的 $(x \vee y) \wedge z$ 示例中做到这一点。

每个变量都有一条由两根单根弦组成的路径（一个 P_3）。切断标记端的弦会将变量设为真，而切断另一端的弦则会将其设为假；注意，路径的中间硬币只与这两根弦相邻，因此它们不能同时被切断。常量 1 和 0 以特殊形式绘制：1 可以视为其真边已被切断的变量，0 则是其假边已被切断的变量，不过可以省略一些额外部分，如构造所示。

每条变量路径的两端都通过宽度为 N^4 的绳索连接到“地”顶点 g，而 g 还有一片专属的额外叶节点（它永远不能被切断，否则会使该叶节点孤立）。

现在，假设黑方在 z 处剪断线，白方在 y-路径未标记（假）的末端剪断线，接着黑方在 x 处剪断线。那么 x 和 z 为真，y 为假，而两个真行是 A 和

B。请注意，我们实际上并不要求白方在这些变量上配合我们的走法；事实上，他们可以随意落子，而黑方只需多走一回合，就能让另一个变量为真。关键在于，我们把绳索做得足够粗，这样白方无论在哪里落子都无关紧要；更一般地说，正合取范式公式的胜方总会遵循我们描述的策略，不受对手行动影响（在我们的演示例子 $(x \vee y) \wedge z$ 中，胜方是黑方）。

按我们描述的方式认领变量（从而赢得公式）之后，黑方将注意力转向宽度为 N 的绳索；此时他们的目标是迫使位于 r 的最后一根弦通向真行（即 A 或 B 之一）。起初，位于 r 的 $2N$ 根弦通向 A 或 B，而只有 N 通向 C。因此，黑方每一步都会朝着 C 切割，直到它消失。无论白方如何下棋，至少会有一根指向真行的弦留下。随后，黑方继续在 r 随意切割，直到只剩下一根弦。假设最后一根弦通向 A。

现在，黑方将直接切断 A 与其导线中间硬币之间、宽度为 N^2 的绳索。位于 r 的最后一根弦托住中间硬币，而通往 x、y 和 1 的绳索也托住 A，因此黑方可以合法地切断所有这类弦。之后，黑方再次采用相同的多数程序：他们要确保与 A 相邻的最后一根弦连接到子句中的真部分（也就是本例中的 x 或 1）。注意，x 和 1 起初总共有 $2N^3$ 根弦，而假项 y 只有 N^3 根。此外，我们可以在构造中选取足够大的 N，使得截至这一阶段，游戏中已进行的切割次数少于 N^3。这样，黑方就可以朝着 y 切割，然后在 A 随意切割，直到只剩下一根弦通向某个真条目，例如 x。

考虑 B 导线的中间硬币。它通向 r的所有弦都已被剪断，因此它通向 B 的最后一根弦永远无法被剪断。这样一来，在黑方剪断从 B 到 z的绳索时，B 仍与其相连；另一端的 z 仍有一根绳索通向 g。同样的论证也适用于黑方剪断从 C 到 0 的绳索。这就是我们让通向 g 的绳索宽度为 N^4的原因：我们选择足够大的 N，使得即使白方每一回合都用来剪断这些绳索中的一根，到这个阶段它仍会留有一根弦。

接下来要看谁剪下最后一根弦。在连接 r 与 A的导线上，r处有一根单根弦；另外两根导线通向 r的弦都已被剪断，因此每根导线都必须留有一根弦通向其所在行。再加上从 A 到 x的弦、g处的叶字符串，以及四条路径（x、y、z和 0）上各一根单根弦，共计九根弦。

现在计算从路径两端到 g 的弦数。回想一下，黑方通过在 x 处剪断弦，将 x 置为真。现在，来自 A 的最后一根弦连着 x，而叶节点连着 g，因此绳索中 x 与 g 之间的每根弦都可以剪断。失去弦的另外三个端点是 z、1，以及 y 路径上未标记的假端点。每个端点都必须恰好留有一根弦连接到 g。路径中仍有弦连接到中间硬币的另外四个端点，可以剪掉所有连接到

g.

因此，游戏结束时恰好还剩下 $9+3=12$ 根弦。如果我们选中的端点是假而非真（例如，来自 A 的最后一根弦连向 y 而非 x），那么它仍会保留弦，因此游戏结束时会剩下 13 根弦，而不是 12 根。我们选择 N 为奇数，这样棋盘初始时的弦数就是奇数；初始时有八根单根弦和 19 根绳索，因此奇数个 N 会使总数为奇数。因此，通过迫使一个端点为真，黑方就能迫使剪断奇数根弦，从而走出最后一步并获胜。同样的奇偶性论证也适用于一般构造：行数始终为奇数，且每行的端点数也为奇数，而一个真条目同样可以多剪一根弦到 g。

有了一个演示例子作为基础后，现在我们转而证明一般性结果：引理 4.1。完整的构造只使用了我们目前见过的技巧，只是将其应用于更一般的情形。

引理 *4.1*。由于每一步操作都会截断一条线，并且可以在多项式时间内轻松检查截断是否合法，因此其属于 PSPACE 是显然的，剩下需要证明的是 PSPACE-困难性。设 F 是 Schaefer 的正合取范式博弈的一个实例。我们可以假设 F 含有 $c \geq 1$ 个子句，且每个子句至少包含一个变量（否则该构造是平凡的）。

对于包含 k 个变量的每个子句，我们通过列出其变量，然后追加 $k-1$ 个 1 的副本来构造一行；这样可以确保该行的多数端点需要 k 个真端点。1 的副本提供了其中 $k-1$ 个所需端点，因此，当且仅当其变量中至少有一个为真时，该行才为真。最后，我们再添加 $c-1$ 行，每行仅由一个 0 构成；这意味着，只有当我们有 c 个真行时，所得 $2c-1$ 行中才会有多数为真，而这恰好当且仅当所有 c 个子句都得到满足。

现在，我们按照图 13 的方式构建棋盘。每行对应一枚硬币，并用一根专属导线将它连接到根 r，导线在根端的宽度为 N，在行端的宽度为 N^2。

对于 F 的每个变量，我们构造一条由两根单根弦组成的路径，并在其中一端标上该变量的名称。我们再构造一条路径，两端分别标为 1 和 0，但省略 1 端的弦。中间硬币必须保留通向 0 的弦。这样，1 就相当于一个已赋值为真的变量，而 0 则相当于一个已赋值为假的变量。

对于每行中的每个条目，我们用一根宽度为 N^3 的绳索，将该行对应的硬币连接到标有该条目名称的硬币。因此，包含相同变量 x 的两行都会连接到同一枚标有 x 的硬币；而同一行中出现两个 1，则会有两根绳索从该行连接到同一枚标有 1 的硬币。最后，我们用宽度为 N^4 的绳索将每条路径的两端都连接到 g，并为 g 配置一个专属叶节点。所得的多重图是连通的，而且显然也是二部的。

为选择 N，我们将每条绳索和每根单弦各计一次，并将总数称为

t。取 $N = 2t + 3$。我们这样选择，是为了确保当一名玩家剪断宽度至多为 w的绳索中的弦时，另一名玩家无法剪断一整条宽度为 Nw的绳索。事实上，将单弦的宽度视为 1，则第一名玩家最多有 tw 根宽度至多为 w 的弦可剪。等他们剪完后，对手最多又走了一回合，因此总共最多剪断了 $2tw + 1 < Nw$ 根弦。我们将对宽度为 1、 N、 N^2 和 N^3的情况使用此上界。还要注意， N 是奇数，而棋盘最多有 tN^4 根弦，这相对于公式的规模是多项式的。

用 P 表示在公式博弈中拥有必胜策略的玩家，并用 P' 表示其对手。随着棋盘上的对局进行， P 会记录相应公式博弈中的对局过程。每回合， P 都会按照自己在公式博弈中的必胜策略认领相应的变量（若 P 是黑方，则将变量设为真；若 P 是白方，则将变量设为假）。如果 P' 设置了一个在公式博弈中尚未被认领的变量，那么 P 就将该变量记录为对手所下的一步；否则，如果 P' 在其他位置落子，那么 P 就任意记录一个尚未被认领的变量，作为对手所下的一步。

这样一来，公式博弈中所有仍未被认领的变量，在棋盘上也仍然处于未设置状态。因此， P 总能按照自己在公式博弈中的必胜策略走出规定的一步。记录完所有认领后，P 会将棋盘上仍未设置的变量全部设置完毕（如果 P' 在其他位置落子，就会出现这种情况）。如果 P 是黑方，那么他在每个子句中都设置了一个真变量。如果 P 是白方，那么她将某个子句中的所有变量都设为了假。因此，多数行的取值符合 P 的期望。

考虑根部 r处的弦。每一行都从其导线靠近根部的一端开始，带有 N 根弦。由于多数行具有所需的值，通向这些行的导线之间一开始至少比其余导线多出 N 根弦。到目前为止切断的弦少于 N 根，因此这些导线仍处于领先位置。在每个回合中，只要通向错误值行的导线中还存在弦， P 就可以在 r处切断一根朝向该行的弦。通向错误值行的所有弦都消失后， P 继续在 r 处切断，直到只剩下一根弦。这些切断是合法的，因为每个中间硬币必然仍保留着通向其所在行的较宽绳索 N^2 。因此， r 处最后剩下的那根弦必然通向一个具有所需值的行。

现在， P 将切断所选行与其导线的中间硬币之间、宽度为 N^2 的绳索。r 处的最后一根弦固定着中间硬币，而通往端点的宽度为 N^3 的绳索仍固定着该行，因此 P 可以合法地切断整条宽度为 N^2 的绳索。绳索一旦消失，该行仍必须保留一根通往某个端点的弦。在这里，我们采用与之前相同的多数论证：每个端点贡献了 N^3 根弦，多数端点具有期望值，而截至目前切断的弦少于 N^3 根。因此，通过先朝着值错误的端点切断， P 可以确保所选行上的最后一根弦通向具有期望值的端点。

最后， P 切断其他各行通往其端点的所有剩余弦。每一行都由其导线的中间硬币固定，而该中间硬币已经失去了通往 r 的所有弦，因此必须至少保留一根通往该行的弦。在另一端，端点硬币仍保留着通往 g 的绳索，因为截至目前切断的绳索少于 N^4 条。

P 现在可以任意进行操作，直到没有合法的切割可做。游戏结束时，每条 $2c-1$ 导线上恰好剩下一条弦；选定的导线保留其位于 r的弦，其他每条导线则保留一条位于其行末端的弦。我们还要计算一条从选定行连到其选定入口的弦，以及一条连到 g处叶节点的弦。

假设 F 有 n 个变量。对每个变量，其两个端点硬币中恰有一个仍与其中间硬币相连。再加上从 0 连到其中间硬币的弦，共有 $n+1$ 条弦。同样，有 $n+1$ 个端点硬币没有弦连到其中间硬币：每个变量对应一个，此外还有 1 个。除非这些硬币被选定行持有，否则每个都必须保留一条连到 g 的弦。连到 g 的其他所有弦都可以切断，因为叶节点持有 g。

如果选定入口为假，那么它仍与其中间硬币相连。因此，刚才计数的所有 $n+1$ 个端点硬币都必须保留一条连到 g的弦。这会在 $2c-1$ 条导线上的弦、选定行的弦以及叶字符串之外，再增加 2($n+1$) 条弦，总数为奇数。如果选定入口为真，那么它就是 $n+1$ 个本来需要一条连到 g的弦的硬币之一。选定行持有它，因此剩下的弦少一条，总数为偶数。

棋盘起初有奇数根弦： $2c-1$每一行都有奇数个端点，因此从行到端点的绳索总数为奇数；每根导线贡献两根绳索，而 g 每个变量有两根绳索，一根连到 0，另一根连到 1。因此，绳索总数为奇数，且每根绳索的宽度都是奇数。此外，还有 $2n+2$ 单根弦：每个变量对应两根，一根连接 0 和其中间硬币，另一根连接 g 和它的叶节点。因此，真条目会产生奇数次切割，假项则会产生偶数次。黑方先行，所以无论哪种情况， P 都会进行最后一次切割。

□

致谢

我们感谢 user30471 在 TeX Stack Exchange 上提供的 TikZ 代码，我们对其进行了改编，用于绘制围棋棋盘图。

参考文献

[1] M. H. Albert、R. J. Nowakowski 和 D. Wolfe。《博弈课：组合博弈论导论》。A K Peters, Ltd.，马萨诸塞州韦尔斯利，2007 年。

[2] Elwyn R. Berlekamp、John H. Conway 和 Richard K. Guy。《Winning Ways for Your Mathematical Plays》，第 1 卷。A K Peters, Ltd.，第二版，2001 年。[3]Kevin Buchin、Mart Hagedoorn、Irina Kostitsyna 和 Max van Mulken。Dots & Boxes 是 PSPACE 完全问题。载于 Filippo Bonchi 和 Simon J. Puglisi 编，《第 46 届数学基础与计算机科学国际研讨会（MFCS 2021）》，第 202 卷《莱布尼茨国际信息学论文集（LIPIcs）》，第 25:1–25:18 页。Schloss Dagstuhl – Leibniz 信息学中心，2021 年。doi:10.4230/LIPIcs.MFCS.2021.25.[4] Kyle Burke、Caroline Cashman、Alfie Davies、Kanae Yoshiwatari 和 Francesca Yu。即使在平面图上，Misère Partizan Arc Kayles 也是 PSPACE 完全问题，2025 年。 arXiv:2511.21888.[5] Kyle Burke 和 Robert A. Hearn。PSPACE 完全的双色平面放置博弈。《国际博弈论杂志》，48(2):393–410，2019 年。 doi:10.1007/s00182-018-0628-8.[6] Erik D. Demaine 和 Jenny Diomidova。Strings-and-Coins 和 Nim-string 是 PSPACE 完全问题。《Integers》，21B:A7，2021 年。 doi:10.5281/zenodo.10939674.[7] Erik D. Demaine 和 Robert A. Hearn。使用算法进行博弈：算法组合博弈论。载于 Michael H. Albert 和 Richard J. Nowakowski 编，《Games of No Chance 3》，第 56 卷《数学科学研究所出版物》，第 3–56 页。剑桥大学出版社，2009 年。 doi:10.1017/CBO9780511807251.002.[8] Robert P. Gallant、Georg Gunther、Bert L. Hartnell 和 Douglas F. Rall。图上的边删除博弈。《组合数学与组合计算杂志》，57:75–82，2006 年。[9] Robert A. Hearn 和 Erik D. Demaine。《Games, Puzzles, and Computation》。A K Peters, Ltd.，2009 年。[10] Melissa Huggan 和 Richard J. Nowakowski。合并博弈：GO-CUT 和 SNO-GO。载于 Urban Larsson 编，《Games of No Chance 5》，第 70 卷《数学科学研究所出版物》，第 395–402 页。剑桥大学出版社，2019 年。 doi:10.1017/9781108641999.019.[11] Matthew Johnson、Barnaby Martin、Jelle J. Oostveen、Sukanya Pandey、Daniël Paulusma、Siani Smith 和 Erik Jan van Leeuwen。禁图复杂性框架 I：框架。《Algorithmica》，87(3):429–464，2025 年。 doi:10.1007/s00453-024-01289-2。

[12] John Moore。反阿塔里围棋。Sensei's Library，第1版，2005年4月1日。https://senseis.xmp.net/?AntiAtariGo:v1。[13] Thomas J. Schaefer。关于一些双人完全信息博弈的复杂性。计算机与系统科学杂志，16(2)：185–225，1978年。 doi:10.1016/0022-0000(78)90045-4。[14] Aaron N. Siegel。组合博弈论，数学研究生教材第146卷。美国数学学会，2013年。 doi:10.1090/gsm/146。