arXiv:2603.09958v1 [cs.CC] 10 Mar 2026

仅使用一种方块类型的俄罗斯方块很难

MIT Hardness Group*

Josh Brunner† Erik D. Demaine† Della Hendrickson† Jeffery Li†

摘要

我们分析了在限定为单一多格骨牌类型时，俄罗斯方块消行（确定玩家能否使用给定的方块序列清除初始棋盘）和生存（确定玩家能否在初始棋盘中放置完所有给定方块之前避免失败）的计算复杂性。我们证明，对于除 P 之外的任意四格骨牌类型，在标准超级旋转系统（SRS）下，俄罗斯方块消行和生存都是 NP难问题，即使输入序列仅由指定数量的 P 方块组成。这些出人意料的结果推翻了一个关于仅使用 $7k$方块的俄罗斯方块计算复杂性的、已有 23 年历史的猜想（尽管我们的结果仅适用于特定的旋转系统）。作为推论，我们证明，对于任意正整数 $k \geq 1$，当方块序列必须能够由 $1 \times k$-袋随机生成器生成时，俄罗斯方块消行是 NP难问题。另一方面，在假设采用特定旋转模型的情况下，当输入序列仅由多米诺骨牌组成时，我们给出了俄罗斯方块消行和生存的多项式时间算法，从而解决了一个有 9 年历史的开放问题。在此过程中，我们还给出了使用 k方块时俄罗斯方块消行和生存的多项式时间算法（对于任意固定的 $k-1$），前提是顶部行在初始时为空，这表明我们的 NP 困难性结果需要顶部三行中存在已填充单元格。

1 引言

俄罗斯方块是最古老、最受欢迎的益智电子游戏之一，最初由阿列克谢·帕基特诺夫于 1984 年创作。俄罗斯方块多次进入主流媒体，最近一次是传记电影俄罗斯方块 [Tet23]，以及 16 岁的 Michael Artiaga 成为第一个通关任天堂红白机版俄罗斯方块全部 255 关的人这一新闻 [McM24]。

在俄罗斯方块的基础版本中，玩家会得到一个由单位正方形格组成的矩形网格（方格）。每一回合，一个四格骨牌（, , , , , , 之一）会在网格顶部生成，并在其下方方格为空的前提下，周期性地向下移动一个单位。玩家可以反复将该拼块向左移动一个单位、向右移动一个单位、向下移动一个单位，或将其旋转 $\pm 90°$。当拼块试图向下移动，而其任意部分停靠在已填充方格的顶部时，该拼块会“锁定”在原地并停止移动。如果拼块“锁定”在某一高度之上，或锁定在下一拼块将要生成的位置，玩家即告失败；否则，下一拼块会在网格顶部生成，游戏继续。完全填充一行会导致该行消除，该行上方的所有方格向下移动一个单位。更详细的规则见 [Wik]。

俄罗斯方块有许多官方和非官方的变体 [Tetb], , 会改变玩家生成的方块类型。例如，*ntris* [Kis11]和*Combinos* [Teta] 将方块类型推广为 ***k-ominoes***（由单位方格沿边粘合而成的连通形状）。

*人工标注的第一作者，用于强调其他作者（按字母顺序排列）作为一个平等团队开展工作。请在参考文献中列出所有作者（包括该作者），并将作者称为“MIT Hardness Group”（不要使用“et al.”）。†麻省理工学院计算机科学与人工智能实验室，瓦萨街32号，剑桥，马萨诸塞州02139，美国，{brunnerj,edemaine, della,jeli}@mit.edu

从计算复杂性的角度研究俄罗斯方块时，我们通常假设玩家获得一个初始棋盘状态，其中有一些已填充单元格，并且获得有限的方块序列，或数量无限的某种类型方块，因此这是一种完全信息博弈（如 [BDH+04]中所述）。两个主要目标是“消行”和“生存”（如 [DDE+17]中所述）。在俄罗斯方块消行问题中，我们要判定：在放置完有限方块序列中的所有方块，或放置一定数量的某种类型方块后，能否清空整个棋盘。在俄罗斯方块生存问题中，我们要判定：玩家在放置给定方块序列中的所有方块时，能否从给定的初始棋盘状态出发避免失败。先前研究表明，如果玩家获得的是有限方块序列，那么这些问题都是NP完全的；即使要在 $n^{1-\epsilon}$ [BDH+04]所述的各种度量指标上进行近似，或棋盘只有8列或4行 [ACD+20]，或使用不可旋转的多米诺骨牌或 k-ominoes来实现 $k \geq 3$ 消行或 $k \geq 4$生存 [DDE+17]，又或方块仅限于七种方块类型中的任意两种（假设采用特定的旋转系统） [MDHL24]，，以及仅限于下落方块或重力极高的情况 [MDHL24]。

俄罗斯方块的在线 版本也曾受到研究。在这种版本中，玩家看不到方块序列的大部分内容（不过可能会预览接下来的若干个方块），相关研究既来自俄罗斯方块社区 [DTW]，最近也有 [GV24]，的研究，表明不存在任何能够与任何离线算法相竞争的在线算法。

1.1 我们的结果

本文研究输入序列恰好包含一种多格骨牌类型这一限制下的俄罗斯方块消行与生存问题，解决了此前俄罗斯方块论文 [BDH+04、DDE+17、MDHL24] 提出的开放问题。

积极的一面是，对于输入序列仅由多米诺骨牌组成的情况，我们给出了俄罗斯方块消行和生存问题的多项式时间算法。这些结果假设多米诺骨牌采用一种特别简单的下落旋转模型，其中方块只能单调向下移动；其他旋转模型下的复杂性仍是开放问题。此外，消行结果也仅适用于无限（或足够长）的多米诺骨牌序列。除了这些限制，我们的结果解决了 [DDE+17] 提出的关于不同 n 的 n 三角形的三个剩余开放问题中的两个。（唯一尚未解决的开放问题是如何在不可旋转的多米诺骨牌序列中生存。）我们开发的关键子程序，是在棋盘顶部 $k-1$ 行为空的前提下，对于任意固定的 k，仅使用 $1 \times k$ 方块实现消行和生存的策略。（直观地说，只要你离失败还不太近，使用条形方块就很容易生存。）

在负面结果方面，我们证明，对于除 P 以外的任何四格骨牌类型，即使输入序列仅由指定数量的 P 方块组成，俄罗斯方块消行和生存问题仍然是 NP难的。这些结果基于俄罗斯方块指南（***SRS***） [Tetc]，中由 Tetris公司制定的超级旋转系统（SRS），该指南规定了所有现代（2001+）俄罗斯方块游戏应有的行为 [Tetd]。这些结果解决了 [MDHL24] 中关于 { , , , , , , } 单一四连方类型的大多数开放问题，并推翻了 23 年前的猜想：只使用方块的俄罗斯方块游戏可以在多项式时间内求解 [BDH+04]。这与我们关于顶部三行为空时的正面结果形成了鲜明对比。不过，我们的结果关键依赖于旋转系统，尤其是 SRS 中利用“踢墙”沿某些“阶梯”向上攀爬的能力。

作为一个具体推论，我们的 NP 困难性结果首次适用于带有暂存 功能的现代俄罗斯方块游戏；该功能允许玩家将一个方块暂时放在一旁以备后用。由于给定的方块类型都相同，暂存不会影响方块的放置顺序。

此外，我们关于仅使用方块进行俄罗斯方块消行的 NP 困难性结果 可以修改后证明：如果方块序列必须能够由适用于任意正整数 $k \geq 1$的“7k袋随机生成器”生成，那么俄罗斯方块消行也是 NP 难的。在***7***袋随机生成器中，方块序列被分成若干组（或称“袋”），每组包含 7 个方块，且恰好各有一个四格骨牌，并按照 7! = 5，040 种可能的顺序之一排列。 7***k***袋随机生成器 是这类随机生成器的一种自然扩展，其中

现在，序列被分成若干组，每组包含 $7k$ 个方块，且恰好包含 k 个每种四格骨牌，并按照 $(7k)!\,(k!)^7$ 种可能的顺序之一排列。

1.2 提纲

本文其余部分的结构如下。第2节详细介绍本文所考虑的旋转系统，包括用于四格骨牌的超级旋转系统（SRS），以及我们针对多米诺骨牌的下落旋转模型。第3节讨论我们关于使用 $1\times k$ 方块进行俄罗斯方块生存与消行的正面结果。第4节描述1-in-3SAT和图定向问题，我们的困难性证明即归约自这些问题。第5节描述仅使用方块、采用SRS的俄罗斯方块消行的困难性结果，以及如何修改该困难性结果，从而在方块序列必须能够由7袋随机生成器或针对 $k>1$ 的 $7k$ 袋随机生成器生成时，得到俄罗斯方块消行的NP 困难性。第6节描述仅使用方块或仅使用方块、采用SRS的俄罗斯方块消行的困难性结果，第7节描述仅使用方块、采用SRS的俄罗斯方块消行的困难性结果，第8节描述仅使用方块或仅使用方块、采用SRS的俄罗斯方块消行的困难性结果。第9节描述对我们的构造所做的修改，这些修改可得出采用SRS的俄罗斯方块生存相应的单一方块类型困难性结果。

2 旋转系统

2.1 超级旋转系统（SRS）

对于涉及四格骨牌的困难性结果，我们假设采用标准的超级旋转系统（***SRS***）[Tetc]。

每个拼块都有一个确定的旋转中心，如图1中的点所示，但 和 除外；它们的旋转中心是图1中 4×4 方格的中心。没有障碍时，所有非 四格骨牌都只绕旋转中心旋转（请注意，拼块无法旋转）。SRS 的关键特点是踢墙机制：尝试旋转四格骨牌时，如果受到阻挡，游戏会尝试将四格骨牌“踢”到四个备用位置之一，并依次测试；如果四个位置都不行，旋转就会失败。图2展示了踢墙过程的一个示例。墙踢的完整数据见表1和表2，以及 [Tetc]；图44则展示了各个测试位置相对于拼块的位置。值得注意的是，除 拼块外，SRS 墙踢对所有拼块或拼块对（即 $\leftrightarrow$ 和 $\leftrightarrow$ ）都具有垂直对称性，因此所有旋转都可以镜像。

	测试1	测试2	测试3	测试 4	测试 5
$0\to R$	$(0,\ 0)$	$(-1,\ 0)$	$(-1,\ +1)$	$(0,\ -2)$	$(-1,\ -2)$
$R\to 0$	$(0,\ 0)$	$(+1, 0)$	$(+1, -1)$	$(0, +2)$	$(+1,\ +2)$
$R\to 2$	$(0,\ 0)$	$(+1,\ 0)$	$(+1,\ -1)$	$(0,\ +2)$	$(+1,\ +2)$
$2\to R$	$(0,\ 0)$	$(-1,\ 0)$	$(-1,\ +1)$	$(0,\ -2)$	$(-1,\ -2)$
$2\to L$	$(0,\ 0)$	$(+1,\ 0)$	$(+1, +1)$	$(0, -2)$	$(+1,\ -2)$
$L\to 2$	$(0,\ 0)$	$(-1, 0)$	$(-1,\ -1)$	$(0,\ +2)$	$(-1,\ +2)$
$L\to 0$	$(0,\ 0)$	$(-1,\ 0)$	$(-1,\ -1)$	$(0,\ +2)$	$(-1,\ +2)$
$0\to L$	$(0, 0)$	$(+1, 0)$	$(+1, +1)$	$(0, -2)$	$(+1,\ -2)$

表 1： 踢出数据，适用于 、 、 和方块。0 表示默认方向，而 R、2 和 L 分别表示从默认方向顺时针旋转 90°、 180°和 270° 后达到的朝向。有序对（a，b）表示中心沿 x 方向平移 a 个单位，并沿 y 方向平移 b 个单位。 x 方向为正时表示向右， y 方向为正时表示向上。

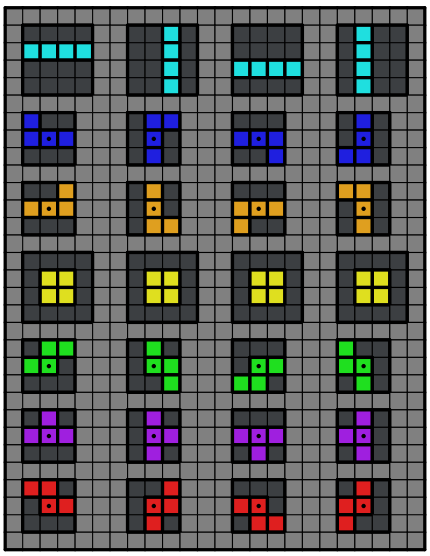
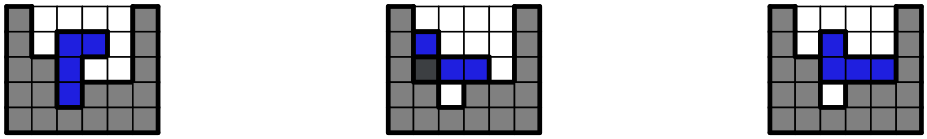


图1：所有四格骨牌，从上到下依次为：、、、、、、。第一列是方块生成时的默认方向，右侧每一列表示该方块绕其旋转中心进行一次 90° 顺时针旋转。



(a) **(b)** **(c)**

图2：SRS 踢墙系统的一个示例。假设 (a) 中的方块正在 逆时针旋转。测试1（即 $(0,0)$）会失败，原因是 (b) 中所示的深灰色方格。测试2（即 $(+1,0)$）会成功，如 (c) 所示，因此该方块会旋转到 (c) 中的位置。

这种在旋转过程中踢移四格骨牌的机制，使得一些操作成为可能；这些操作通常称为扭转或旋转。我们使用的所有旋转操作详见附录B、C、D和E。

2.2 下落旋转模型

对于使用多米诺骨牌的俄罗斯方块，我们假设一种旋转模型：方块旋转时位置单调降低。图3精确定义了这一下落旋转模型。一般而言，每当我们旋转一块多米诺骨牌时，其最低像素之一保持原位，另一个像素则移动到相邻且不更高的单元格，前提是该单元格未被占用。（如果该单元格已被占用，则旋转失败，方块不会旋转。）这一定义中有两种选择，分别对应顺时针旋转和逆时针旋转：当多米诺骨牌为水平时，我们可以选择左侧或右侧像素保持原位；当多米诺骨牌为垂直时，我们可以选择相邻单元格在左侧还是右侧。

3 用多米诺骨牌玩俄罗斯方块，$1 \times k$

Demaine等人 [DDE+17] 证明了使用不可旋转的多米诺骨牌进行俄罗斯方块消行的NP 困难性（其中禁止旋转）。他们将不可旋转的多米诺骨牌情形下生存的复杂性，以及可旋转多米诺骨牌情形下消行与生存的复杂性留作开放问题。在本节中，我们给出一个用于生存的多项式时间算法

	测试1	测试2	测试3	测试4	测试5
$0 \to R$	(0, 0)	(−2, 0)	(+1, 0)	(−2, −1)	(+1, +2)
$R \to 0$	(0, 0)	(+2, 0)	(−1, 0)	(+2, +1)	(−1, −2)
$R \to 2$	(0, 0)	(−1, 0)	(+2, 0)	(−1, +2)	(+2, −1)
$2 \to R$	(0, 0)	(+1, 0)	(−2, 0)	(+1, −2)	(−2, +1)
$2 \to L$	(0, 0)	(+2, 0)	(−1, 0)	(+2, +1)	(−1, −2)
$L \to 2$	(0, 0)	(−2, 0)	(+1, 0)	(−2, −1)	(+1, +2)
$L \to 0$	(0, 0)	(+1, 0)	(−2, 0)	(+1, −2)	(−2, +1)
$0 \to L$	(0, 0)	(−1, 0)	(+2, 0)	(−1, +2)	(+2, −1)

表2： 方块的踢出数据，记法与表1相同。

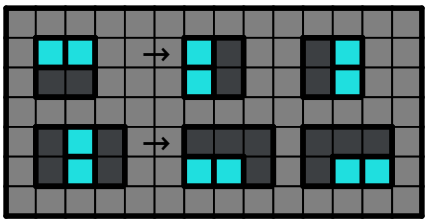


图3： 多米诺骨牌的下落旋转模型：当左侧的多米诺骨牌尝试顺时针或逆时针旋转时，如果相应位置没有障碍物，它就会变成右侧对应的多米诺骨牌。

针对可旋转多米诺骨牌，以及一个多项式时间算法，用于判定棋盘在无限/足够多方块下是否最终可消行。

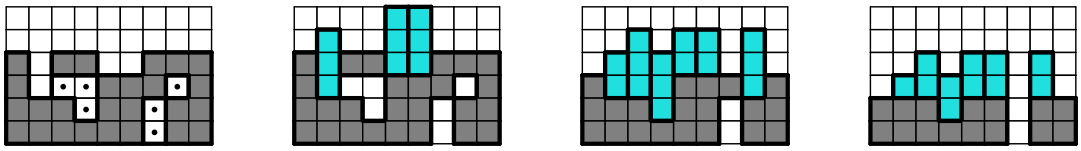
我们还给出一种策略：只要顶部 $k-1$ 行起始时为空，无论 k 固定为何值，它都能确保仅使用 $1\times k$ 个方块时始终存活。采用这一策略后，给定足够多这样的方块，我们可以在多项式时间内判定是否有可能最终完全清空棋盘。我们会将这一结果与 $k=2$ 一起用作多米诺骨牌算法的子程序，而 $k=4$ 也与我们的困难性结果形成了有趣的对照。

3.1 使用 $1\times k$ 拼块生存与消行

定理 3.1。 如果棋盘的高度至少为 $2k-1$，且顶部的 $k-1$ 行最初为空，那么就可以在无限序列的 $1\times k$ 拼块中生存下来。此外，还存在一种多项式时间算法，可用于判定棋盘最终能否被完全消除。

证明。 我们的策略分两个阶段进行。首先，消除所有空洞，其中空洞 指位于已填充单元格下方的空单元格。其次，放置水平棋子，使所有列中的拼块数量对 k 取模后相同。

第一阶段中，我们将说明如何消除最高的洞，同时不产生任何新空洞。然后，通过对洞的数量进行归纳，我们可以清除所有空洞。示例请参见图4。



(a) **(b)** **(c)** **(d)**

图4： 针对 $k=3$ 消除空洞的过程。(a) 中的点表示哪些空方格是空洞。

假设最高的洞 h 位于某一行 r 和某一列 c（如果第 r 行有多个洞，任意选一个即可）。根据洞的定义（并且因为这是最高的洞），洞的正上方有一个已填充单元格

。将该单元格所在的行称为 $r_\prime$。我们将给出一个过程来清除第 r' 行，从而减少空洞上方已填充单元格的数量。

要消除第 $r_\prime$ 行，我们找出在第 r' 行有空单元格的所有列。由于 $r_\prime$ 位于最高的洞上方，因此这些空单元格都不是洞，而且它们上方只有空单元格。在这些列中，我们反复放置竖直方块，直到该列中 r' 处的单元格被填满。现在我们将证明，始终可以放置这些方块，而且不会因为列堆得太高而导致失败。起初有 $k-1$ 行空行，因此最高的洞上方至少有 k 行（因为洞上方至少有一个已占用单元格，其上方还有 $k-1$ 个空单元格）。那么第 r' 行上方必定至少有 $k-1$ 行。由于覆盖第 $r_\prime$ 行的最后一个方块在 r' 上方最多有 $k-1$ 个格子，因此它可以放入棋盘，而不会超出顶部导致失败。

重复应用这一过程， c 空洞上方 h 的所有已填充单元格都会被消除，随后 h 就不再是空洞。这个过程永远不会产生新空洞，因为它只会放置竖直方块。

一旦没有空洞，就很容易永远存活下去：底行上必定有某个空单元格（否则它早已消行），而且因为它不是空洞，我们可以在该列放置一个竖直方块。通过不断在最低可用位置放置方块，我们还可以使最高已填充单元格的高度至多为 $k-1$，由于棋盘总高度至少为 $2k-1$，因此顶部至少有 k 空行。

现在我们给出一种算法，它要么从当前位置清空棋盘，要么证明棋盘从一开始就不可能被清空。在清空状态下，每一列中已填充单元格的数量模 k都相同（即为0）。放置一个竖直方块不会改变其所在列中已填充单元格数量模 k 的值。消除一行会使所有列的计数都改变1。因此，如果棋盘有可能被清空，我们就必须放置水平放置的方块，使所有列的计数相等。

我们用贪心策略判断这是否可行。对于该算法，我们将“在第 c列放置一个拼块”定义为：放置一个水平棋子，使其最左侧单元格位于第 c 列。首先，我们猜测要在最左列放置多少个水平棋子。对于每个后续列，我们必须在其中放置一定数量的水平棋子，使该列已填充单元格的数量与前一列的数量模 k相等。如果这样做后，所有列的单元格数量都模 k 相等，我们就成功了。由于在第一列放置的方块数量只有 k 种可能，我们可以尝试所有可能的值；如果没有任何一种能使所有列的单元格数量都模 k相等，我们就可以断定无法消行。

在此过程中，我们可能会产生一些空洞。每放置一个水平棋子后，我们都会再次运行消除空洞算法（第一阶段）。由于消除空洞算法结束时，顶部至少有 k 行空行，因此每添加一个水平棋子后，顶部至少有 $k-1$ 行空行，这使得消除空洞算法能够继续运行而不会失败。□

3.2 使用多米诺骨牌进行消行的算法

在本小节中，我们将回答 [DDE+17] 中关于仅使用可旋转多米诺骨牌在俄罗斯方块中生存和消行的开放性问题。在 [DDE+17]中，他们指出，消除一整行就足以永远生存下去。我们给出一种算法，用于判定在由多米诺骨牌组成的无限序列中，是否有可能最终消除一整行，从而判定其生存性。消除一行后，使用第3.1节中的算法，还可以判断是否能够将棋盘完全清空。

我们将使用第2.2节中的下落旋转模型。允许的旋转是：旋转后的位置与先前位置恰好在两个单元格中的一个重叠，且另一个单元格位于比先前更低的行。

我们给出一种算法，用来判断某一行 r 是否能够被清空。要判断是否有任何一行可以被清空，只需对每一行 r分别运行此算法。

如果从初始构型出发，可以将多米诺骨牌移动到骨牌的两个格子之一位于 $(r,\ c)$ 的位置，那么我们称单元格 $(r,\ c)$ 可达 。类似地，如果从初始构型出发，可以将多米诺骨牌移动到一对相邻的空格，那么这对空格就是一个可达多米诺骨牌位置。请注意，我们只要求多米诺骨牌能够经过该位置；即使因为没有支撑物而无法停在那里，该位置仍被视为可达。

如果每个拼块下方都有一个已填充单元格或多米诺骨牌，就称该多米诺骨牌构型为受支撑构型。因此，在受支撑构型中，对于每块多米诺骨牌，如果它是最后放置的，就不会掉落。首先，我们证明，只要每块多米诺骨牌所在的位置单独来看都是可达的（忽略其他多米诺骨牌），就能安排多米诺骨牌的放置顺序，从而构建出给定构型。如果一个多米诺骨牌所在的位置放置后，会导致另一多米诺骨牌所在的位置变得不可达，就称前者阻挡后者。此外，如果第二块多米诺骨牌至少有一个单元格正好位于第一块多米诺骨牌某个单元格的上方，就称第一块多米诺骨牌所在的位置支撑第二块多米诺骨牌所在的位置。

首先，我们为所有可达的空单元格建立一个排序，称之为路径顺序。示例请参见图5。我们将用这些单元格在列表中的位置为每个单元格标注。我们逐行处理，较高行中的单元格始终排在较低行之前。在每一行中，我们先处理正上方一行中有可达单元格位于其上方的所有单元格。（对于第一行，这指所有空单元格。）然后从这些单元格开始，沿着这一行向外延伸；每当首次遇到一个可达单元格时，就将其加入路径顺序。

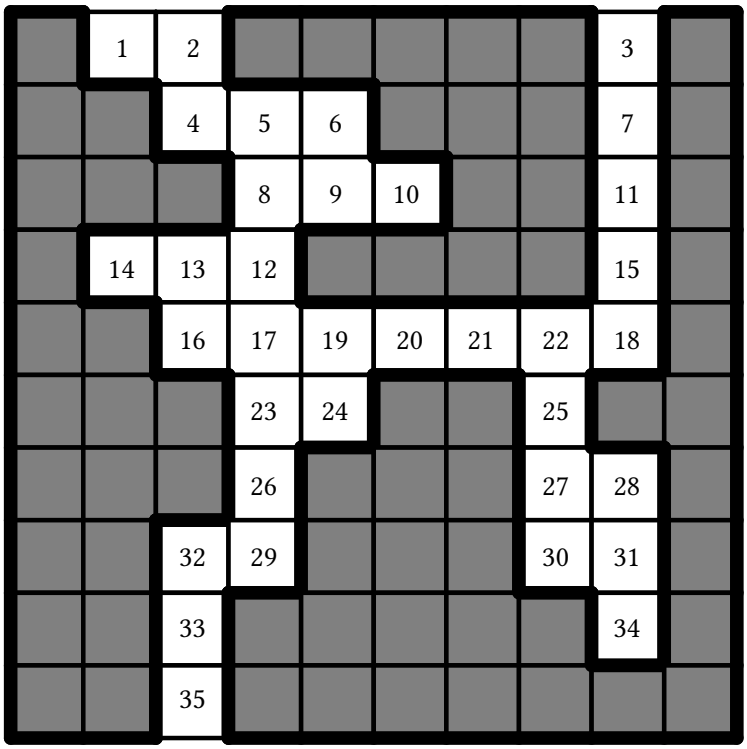


图5： 路径顺序　　示例棋盘状态中空单元格的。此处，较大的数字在路径顺序中出现得较晚　　.

引理 3.2. 多米诺骨牌可以沿着任意一条从顶部空位置出发、且 y 值单调递减的空单元格路径移动。

证明。从路径起点的顶部空位置开始放置多米诺骨牌。多米诺骨牌每次会占据两个单元格。要到达下一个格子，如果它与多米诺骨牌当前位置处于同一直线上，我们只需根据情况将多米诺骨牌向左、向右或向下移动一格。否则，利用下落旋转系统，

总能通过一次旋转，让多米诺骨牌覆盖下一个格子，同时腾出它所占据的两个格子中较早到达的那个。□

引理 3.3。 对于每个可达多米诺骨牌位置，都存在一条多米诺骨牌可以沿着它到达该位置的路径，并且这条路径按路径顺序单调递增。

证明。 我们将按相反顺序计算这条路径。先从多米诺骨牌的最终位置开始，按照路径顺序排列它的两个格子。然后，反复将一个相邻格子添加到路径开头，要求它在路径顺序上早于我们上一步添加的最后一个格子。按照路径顺序的构造方式，这样总能从顶部构造出一条通往多米诺骨牌所在位置的单调路径；根据引理 3.2，多米诺骨牌可以沿着这条路径移动。□

引理 3.4。 每个受支撑的多米诺骨牌构型都可以放置。

证明。 用每个多米诺骨牌两个格子中路径顺序较早的位置为它标记。我们按照标记顺序放置多米诺骨牌，标记最大的先放。每放置一个多米诺骨牌时，所有路径顺序小于该多米诺骨牌标记的格子都仍未被占据。因此，根据引理 3.3，存在一条该多米诺骨牌可以沿着到达其位置的路径。此外，所有支撑该多米诺骨牌的骨牌都必定已经放置，因为较低骨牌的标记总是大于较高骨牌的标记。□

既然我们知道任何配置都可以放置，接下来我们将说明如何找到一个能消行的配置（如果存在的话）。如果满足以下条件，就称某一列 c 是好列：

- (r, c)可达；并且
- 要么$(r + 1, c)$为空且可达，并且 r不是最顶行；要么列c 中 r 下方最近的已填充单元格与 r之间的距离为偶数。

换句话说，好列是指可以通过不断堆叠垂直多米诺骨牌，直到(r, c)被占据的列。坏列则是指任何不是好列的列。

如果满足以下条件，水平多米诺骨牌就能修复坏列 c：

- 该多米诺骨牌的位置可达；
- 它位于 r下方偶数格处；并且
- 它与 r之间的 c列中没有已填充单元格。

换句话说，如果现在可以在一块水平多米诺骨牌上方放置一叠（可以为空的）垂直多米诺骨牌，使（r，c）被占据，那么这块水平多米诺骨牌就能固定 c 列。注意，一块水平多米诺骨牌通常会同时固定它所占据的两列。

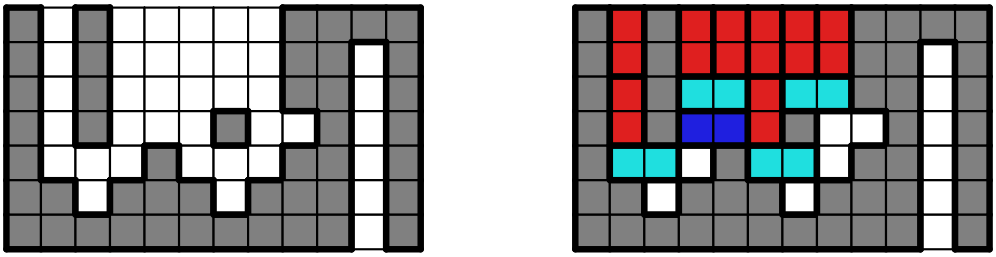
引理 3.5。 r 可被清除，当且仅当存在一个互不重叠的水平多米诺骨牌集合，能够固定每个坏列。

证明。 为了清除第 r 行，每个坏列中都必须有一块水平多米诺骨牌，因为坏列的定义意味着它不可能完全由垂直多米诺骨牌填满。此外，这块多米诺骨牌必须固定该列，否则水平多米诺骨牌上方的空白仍会使该列成为坏列。因此，只有存在一个互不重叠的水平多米诺骨牌集合，能够固定每个坏列，r 才可被清除。

现在我们证明，这样一组水平多米诺骨牌就足够了。如果有任何骨牌没有支撑，就在它们下方加上一摞水平多米诺骨牌，直到它们得到支撑。接下来，在每个有 r 空单元格、需要填满 r 的列中，都加上一摞垂直多米诺骨牌。根据水平多米诺骨牌固定列的定义，这样就能

成功填满先前坏列中 row r 的每个单元格；根据“好”的定义，这样也能填满先前好列中 row r 的每个单元格。

这是一个受支撑构型，因此根据引理 3.4，存在一种放置顺序。□



(a) (b)

图 6： 一种可能的棋盘状态，以及一组用于修复所有坏列的水平多米诺骨牌（青色）、使这些水平多米诺骨牌获得支撑的额外水平多米诺骨牌（深蓝色），以及有助于消除顶行的垂直多米诺骨牌（红色）。

现在，我们给出一种算法，用来找出引理 3.5 所需的水平多米诺骨牌集合。由于每块多米诺骨牌最多只能覆盖两列，我们可以使用一个宽度为 2 列的窗口，从左向右进行动态规划；扫描过程中，我们维护所有能放置水平多米诺骨牌、从而修复最后两列的方式。由于修复每列至多只需 1 块水平多米诺骨牌，因此修复前两列的方式最多有 $O(n^2)$ 种，所以动态规划最多需要 $O(n^3)$ 的时间。

定理 3.6。 存在一种多项式时间算法，可以判断一个俄罗斯方块局面是否可存活，以及使用无限序列的多米诺骨牌时该局面是否最终可清除。

证明。 上述算法会找出一组能够清除一行的多米诺骨牌（如果存在这样的组合）。根据引理 3.4、引理 3.5 和定理 3.1，这就给出了一种策略：如果有可能永远存活，该策略便能使局面永远存活；如果有可能最终清除棋盘，该策略便能最终清除棋盘。□

推论 3.7。 存在一种多项式时间算法，可以判断一个俄罗斯方块局面是否能通过一系列 k 多米诺骨牌存活下来。

证明。 如果该局面能够使用无限多个多米诺骨牌存活下来，那么显然对于任意有限 k也能够存活。根据定理 3.1 以及 [DDE+17] 的说明，唯一无法永远存活的情况是不存在任何可清除的行。在这种情况下，我们将多米诺骨牌装入可达位置。这很容易，因为这是一个二分图上的最大匹配问题；根据引理 3.4，任何这样的配置总是可以放置的。□

4 可满足性问题与图定向

这里，我们将讨论所有归约到仅使用一种四格骨牌类型的俄罗斯方块消行所需的可满足性问题和图定向问题。

我们针对 -三角形的归约将来自或涉及以下问题：

问题 4.1 (1-in-3SAT)。 给定一个合取范式（*CNF*）公式 ϕ，其中包含 m 个子句和 n个变量，且每个子句 C_j 都含有三个文字（即变量或其否定），判断是否存在一个对变量的满足真值赋值，使得 ϕ 中的每个子句恰好有一个真文字。

问题 4.2 (图定向)。 给定一个无向3-正则图 $G=(V,E)$，其中 V被划分为三个（互不相交的）节点子集 V_ℓ（文字节点）、V_C（子句节点）和$V_{\bar{C}}$（否定子句节点），判断是否存在一种对 G 的定向（即为 G中的所有边指定方向），使得：

1. V_ℓ 中的每个文字节点的入度要么为零，要么为三，
2. V_C 中的每个子句节点的入度为一，并且
3. $V_{\bar{C}}$ 中的每个否定子句节点的入度为二。

这两个问题都已知是NP完全问题；1-in-3SAT在 [Sch78], 中被证明是NP完全的，而图定向在 [HIN+17]中被证明是NP完全的。

我们对 -三角形的归约, -三角形，且 -三角形将来自以下问题：

问题 4.3 (平面双连通 {{0, 4}-in-4,1-in-4} 图定向)。 给定一个无向4正则平面双连通图 $G=(V,E)$，满足 V 被划分为两个（不相交的）节点子集 V_0 和 V_1，判断是否存在 G 的一种朝向（即，为 G 中所有边指定方向），使得每个节点 $v \in V_0$ 的入度为0 或4，且每个节点 $v \in V_1$ 的入度恰为1。

问题 4.4 (平面双连通 {{0, 4}-in-4,4 中取 3} 图定向)。 给定一个无向4正则双连通平面图 $G=(V,E)$，满足 V 被划分为两个（不相交的）节点子集 V_0 和 V_3，判断是否存在 G 的一种朝向（即，为 G 中所有边指定方向），使得每个节点 $v \in V_0$ 的入度为0 或4，且每个节点 $v \in V_3$ 的入度恰为3。

这些问题在没有双连通性条件的版本已在 [MAD+26] 中被证明是 NP完全的。我们证明，添加双连通性条件并不会改变这些问题的难度：

定理 4.5。 平面双连通 {{0，4}-*in*-4、1-*in*-4} 图定向和<|*style id='19'>{{*0，4}-*in-4*，3-*in*-4} 平面双连通图定向都是 *NP*难的。

证明。 我们关注平面双连通 {{0，4}-in-4，1-in-4} 图定向；对另一个问题的证明类似。我们从平面 {{0，4}-in-4，1-in-4} 图定向（不含双连通性条件） [MAD+26] 进行归约。假设不存在孤立顶点（因为可以移除这些顶点）。给定一个平面 {{0，4}-in-4， 1-in-4} 图定向实例，我们选择一个平面嵌入，使所有双连通分支共享一个公共的“外部”面（即没有任何双连通分支嵌套在另一个双连通分支内部）。然后，我们进行如下操作：

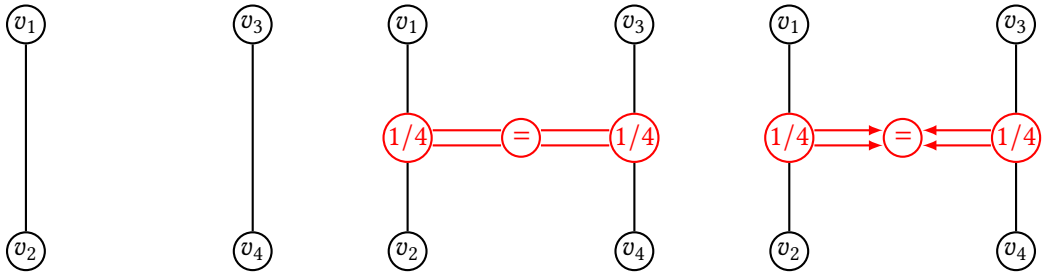
- 首先，对于某一对连通分支，我们分别从每个连通分支的“外侧”面上选取一条边，并用图7b所示的构件替换这两条边。该构件由三个新顶点和四条连接这三个顶点的新边组成，其中一个新顶点将其中一条边分成两条，另一个新顶点则将另一条边分成两条。我们反复进行这一操作，直到整个图连通。

- 接下来，我们对某对相邻双连通分支（即共享一个顶点的双连通分支）做同样的处理：从每个连通分支中各选取一条位于“外侧”面上的边，并用图7b所示的构件替换这对边。我们反复执行此操作，直到该图成为双连通图。

我们证明该归约是有效的。首先，由于构件所替换的边位于“外侧”面上，新增的边和顶点可以嵌入平面，同时保持平面性。此外，每当构件在第一阶段或第二阶段替换一对边时，连通分支或双连通分支的数量会分别减少（因为它连接了两个分离的分支，或在双连通分支之间引入了一条替代连接），因此该过程是有限的，并会在有限次迭代后得到一个双连通图。因此，所得的图定向实例是平面的且双连通的。

因此，只需证明图7b中的构件可以模拟一对不同的边。具体而言，{0，4}-入-4顶点必须按图7c所示的方式定向，因为另一种朝向会违反两个1-入-4顶点的要求。接着，在图7c所示的固定朝向下，两个1-入-4顶点都还需要一条入边和一条出边，因此变成1-入-2顶点，从而模拟与其相连的另外两个顶点之间的一条边。因此，该构件可以模拟一对不同的边，故可用它替换一对不同的边，而不影响图定向实例是否有解。

因此，我们已经证明了一种将平面{{0，4}-入-4，1-入-4}图定向归约为平面双连通{{0，4}-入-4，1-入-4}图定向的方法，所以后者是NP难的，正如所需。同理，平面双连通{{0，4}-入-4，4 中取 3}图定向也是NP难的。□



(a) 添加构件之前　　**(b)**　　**(c)** 强制边

图7： 用于连接 {{0,4}-入-4、1-入-4} 图定向中双连通分支的构件。1/4 表示 1-入-4顶点，而 = 表示一个 {0, 4}-入-4 顶点。该构件替换了两个不同双连通分支中的两条边（v_1、v_2）和（v_3、v_4）（其中的 v_i 不一定彼此不同）。

对于 -三角形和 -三角形，我们将使用平面双连通 {{0,4}-入-4、4 中取 3} 图定向；对于 -三角形，我们将使用平面双连通 {{0, 4}-入-4、1-入-4} 图定向。

最后，我们将从以下问题归约到 -三角形和 -三角形：

问题 4.6 (平面单调直线 3SAT)。 给定一个合取范式（*CNF*）公式 ϕ，它由 m 个子句和 n 个变量组成，并且这些变量可以在平面上按如下方式排列：

1. 每个变量都是 x轴上的一条水平线段；*2.* 每个子句要么全由正文字构成，要么全由负文字构成，并且是 x轴外的一条水平线段，通过 *3* 条垂直连接与变量相连；*3.* 全由正文字构成的子句位于 x轴上方；*4.* 全由负文字构成的子句位于 x轴下方；*5.* 除连接处外，线段之间不会交叉。

判断是否存在一个变量的满足赋值，使得 ϕ 中的每个子句都至少有一个真文字。

该问题已在 [dBK10] 中证明为 NP完全问题。

5-三角形消行

定理5.1。 即使将提供给玩家的序列中的方块类型限制为仅有一种，采用 *SRS* 的俄罗斯方块消行仍然是 *NP*难的。 ∎

这里，我们将从 1-in-3SAT 进行归约，并经过图定向。给定一个包含 n 个变量和 m 个子句的 1-in-3SAT实例，我们首先使用 Horiyama 等人在 [HIN+17 定理 2.3] 中给出的归约，在网格上构造一个对应的图定向实例；示例见图8。请注意，该实例由两部分组成：

- **上部**– 由 n 个变量环、从文字发出的垂直导线，以及由变量环与垂直导线交点形成的交叉点组成，
- **下部**– 由 m 个 [HIN+17 中子句/否定子句构造的副本组成，见图 5] （参见图9）；每个副本在水平方向占用 8 列，且任意两个副本不共用同一列。

此外，我们在图的顶点或边上的每个晶格点处放置一个红点。这些红点将指示我们在归约到俄罗斯方块消行的过程中放置主要构件的位置。

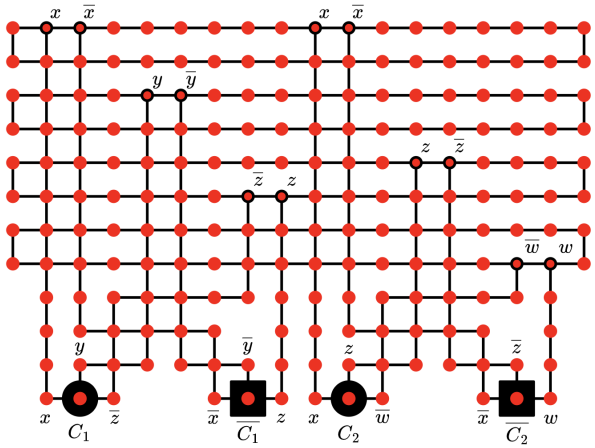


图8： 图定向实例示例，使用 1-in-3SAT实例 $\phi = (x \vee y \vee z) \wedge (x \vee z \vee w)$，并在图中的每个晶格点处标出红点。

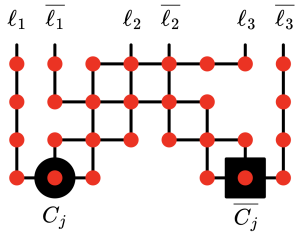


图9： 子句/否定子句构造。

5.1 一般结构

现在，我们使用构造出的图定向实例来构造俄罗斯方块清除实例。假设图定向实例需要一个包含 R 行和 C 列的网格。我们首先描述构造的一般结构，如图10所示。棋盘有 $60R + 10$ 行和 $48C + 16$ 列。构造的主体部分对应于图定向实例，以绿色标示，位于第9行至第 $60R + 8$ 行（含首尾）之间，以及第14列至第 $48C + 13$ 列（含首尾）之间。主体部分有一条由空方格组成的路径通向它，该路径主要位于棋盘最左侧的12列或最顶部的8行内，我们将其称为“长隧道”。棋盘的倒数第二列中还有4个额外的空方格，其中两个位于棋盘最顶部的两行，另外两个位于棋盘最底部的两行。我们只给玩家恰好足够的方块，使得清除棋盘需要使用全部方块；换句话说，如果构造中有 M 个空方格，我们就给玩家 $M/4$ 个方块。

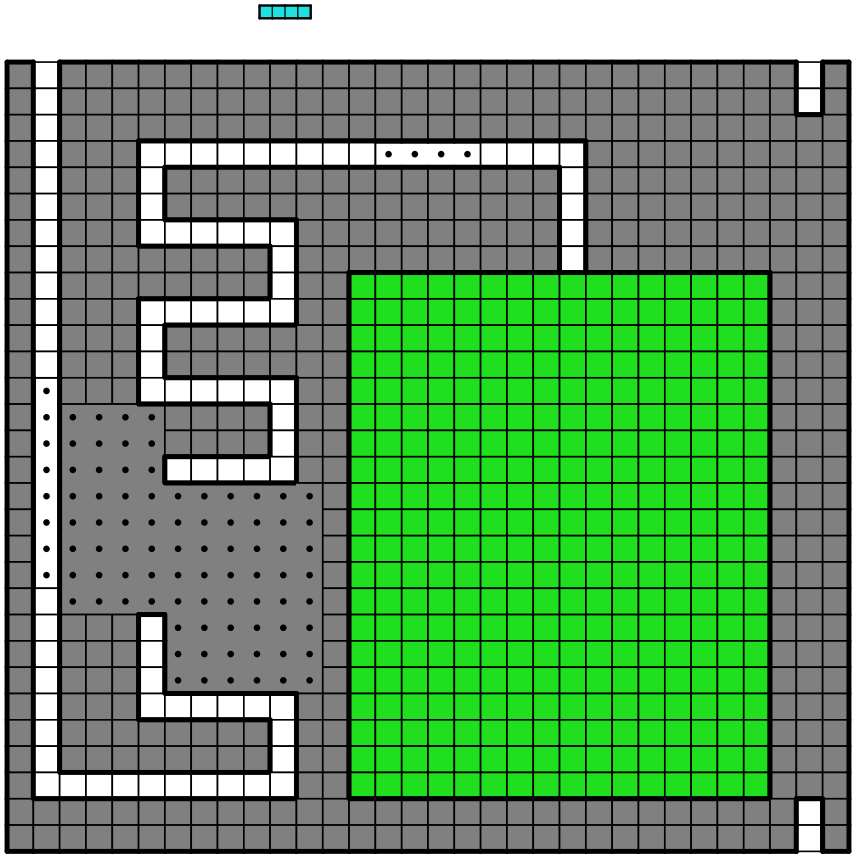


图10：仅使用方块进行俄罗斯方块消行的构造总体结构。构造的主体部分以绿色标示。

我们作出以下观察：

- 最靠右第二列中的4个空方格，必须在中间行消除后最后填入。

- 由于长隧道中有转弯，填满／铺满长隧道的方块摆放方式恰好只有一种，如图11所示；此外，方块可以（在SRS规则下）穿过长走廊进入构造的主体部分（参见附录B.1中的操作）。

- 如果在主体部分的构造完全填满之前，任何拼块在长隧道中锁定，

那么长隧道就会被阻挡，我们也就无法再清空棋盘——即使消去若干行，由于右上角有两个空方格，且该拼块阻挡了长隧道，也没有更多方块能够旋转进入主体部分的构造。因此，在主体部分的构造填满之前，不能将任何拼块放入长隧道。

- 在我们将任何方块放入长隧道之前，长隧道还会阻止与构造主体部分相对应的行发生消行。

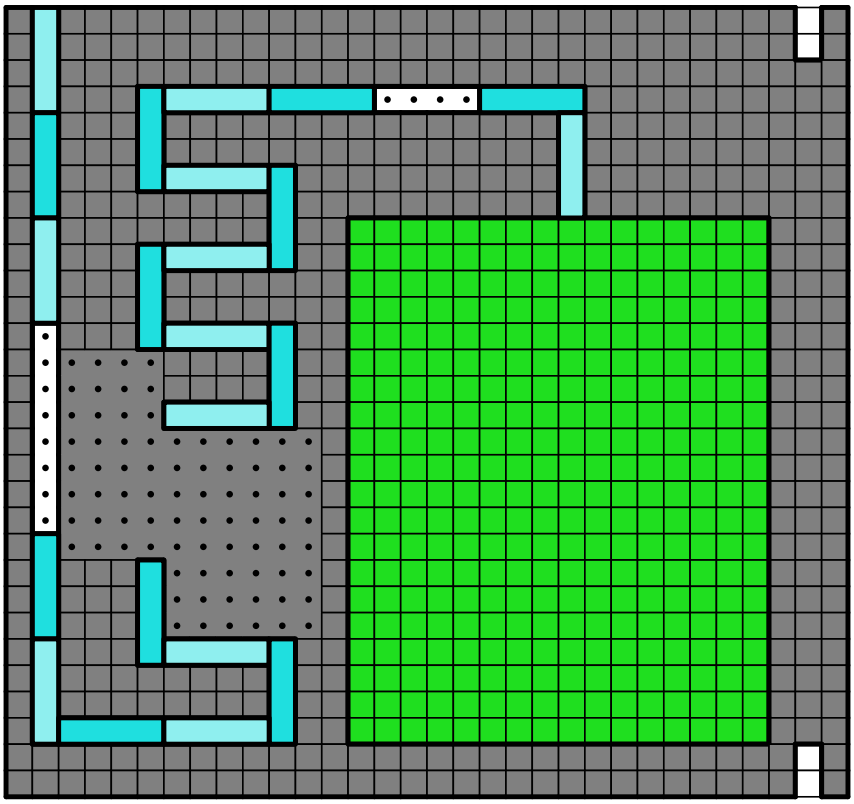


图11：用方块铺砌长隧道的唯一可能方式。

因此，我们需要能够用方块精确铺砌构造的主体部分，并附加一项限制：我们必须能够按照俄罗斯方块和SRS规则，将所有方块操作到位。

5.2 构件

现在，我们介绍构造主体部分中使用的构件。大致而言，这些构件类似于 [HIN+17]中2BAR-Tiling归约所使用的构件，只做了轻微修改，以便玩家能够将方块操作穿过这些构件。这意味着我们拥有拐角、直线、复制器、子句、否定子句以及交叉装置。

这里，我们将构件的入口称为**端口**（例如，水平线构件的左侧和右侧各有一个端口）；我们将端口称为上、左、右、下（对于复制器，两个下方端口都是下端口）。端口将以中心带点的空白（白色）方格表示。请注意，这些端口也充当“额外”的方格，有助于标示构件周围的“朝向”。

我们还会在每个构件的图示中标出特定方格作为“中心”，以便整体描述构造的主体部分；每个构件都有一个中心，复制器构件则有两个。值得注意的是，所有左端口和右端口都与某个中心位于同一行，所有上端口和下端口都与某个中心位于同一列。

一般来说，我们希望这些构件具有以下性质：

- 假设两个端口都存在，则可以进行以下形式的任意遍历：上 → {左、右、下}，左 ↔ 右（即既可以从左到右，也可以从右到左），{左、右} → 下。
- 构件可能的铺砌方式，大致与 [HIN+17]中用于 2BAR-Tiling 归约的对应构件的可能铺砌方式相符。
- 对于每种可能的铺砌方式，在 SRS 下，每个构件都可以用方块填充；如果构件有上端口，方块就从上端口进入，否则方块从左端口或右端口进入。（换句话说，方块能够移动到铺砌中需要它们到达的位置，尤其是在构件被方块部分填充时。）

我们使用的主要技巧是利用方块可以转过拐角这一事实——对方块进行旋转测试的第四项和第五项（如图14所示），可以看作是绕着一个方块的两个端格之一旋转。因此，假设活动片块周围的方格已填充，使得前三项测试均失败，那么活动片块就可以 90° 绕其两个端格中的任意一个旋转（类似于附录B.1中的多次旋转操作）。

5.2.1 角构件和线构件

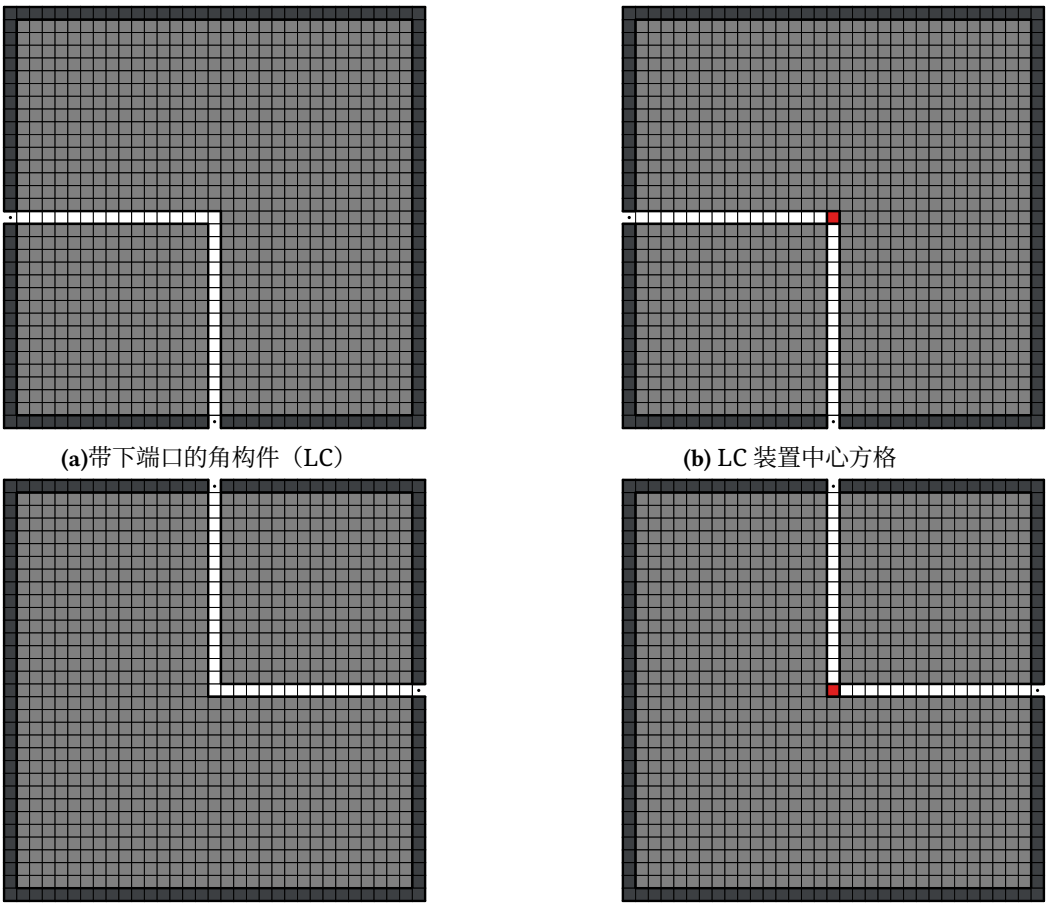
角构件相当简单；图12展示了两种定向以及中心方格（另外两种定向可通过沿垂直线反射得到）。将带有下端口的拐角称为下角（*LC*），将带有上端口的拐角称为上角（*UC*）。

由于拼块可以转过拐角，在 LC 装置中，可以从非下端口遍历至下端口；在 UC 装置中，可以从上端口遍历至非上端口。图 61 展示了所有可能的铺砌，以及确保拼块正确铺砌该构件的一种放置顺序。

由于拼块需要进入构件的主体部分，我们创建了一个特殊的入口角（*EC*）构件，如图13所示。可以验证，构件的上部必定按图13c所示的特定方式铺砌，而构件的其余部分则发挥普通 LC 装置的作用。由于拼块可以转过拐角，因此可以从构件的上部遍历至两个端口中的任意一个。图 62 展示了所有可能的铺砌，以及确保拼块正确铺砌该构件的一种放置顺序。

与普通拐角构件一样，垂直线（VL）构件正是[HIN+17]中的垂直线构件，如图14所示。由于方块可以转过拐角，因此可以从上端口遍历到下端口。图63展示了所有可能的铺砌方式，以及一种可确保方块正确铺砌该构件的放置顺序。

另一方面，水平线（HL）构件与 [HIN+17]中的水平线构件相似，但额外增加了一些方格，以便更轻松地从左端口遍历到右端口，反之亦然（因为玩家只需将水平放置的-方块从左向右滑动，或反向滑动）。HL 构件如图15所示。可以验证，中心矩形的上部必定以图15c所示的特定方式铺砌。图64展示了所有可能的铺砌方式，以及可确保方块正确铺砌该构件的放置顺序。请注意，每种铺砌方式都有两种顺序：一种适用于方块从左端口进入，另一种适用于



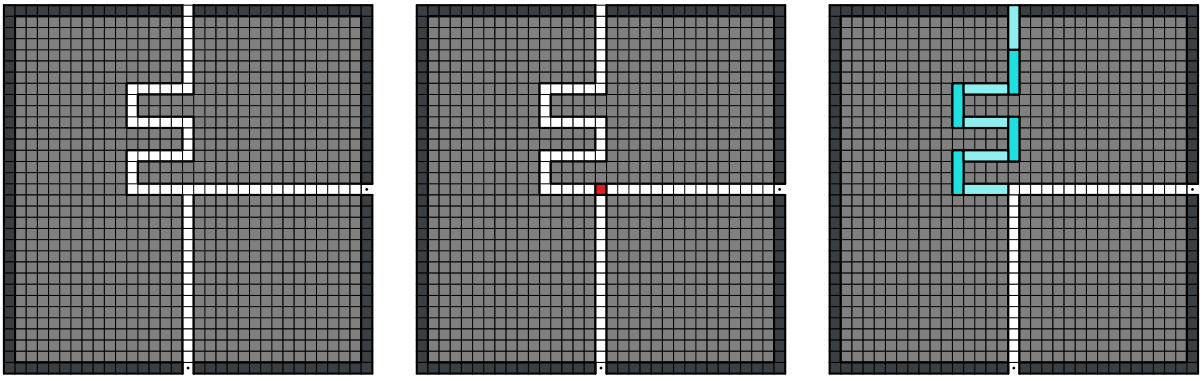
(a)带下端口的角构件（LC）

(b) LC 装置中心方格

(c)带上端口的角构件（UC）

(d) UC 装置中心方格

图12： 角构件。



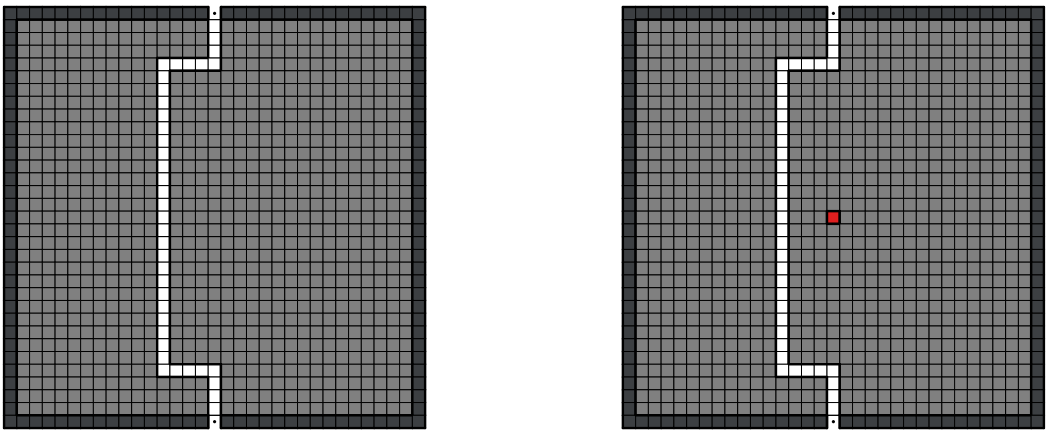
(a)

(b)装置中心方格

(c) 强制放置

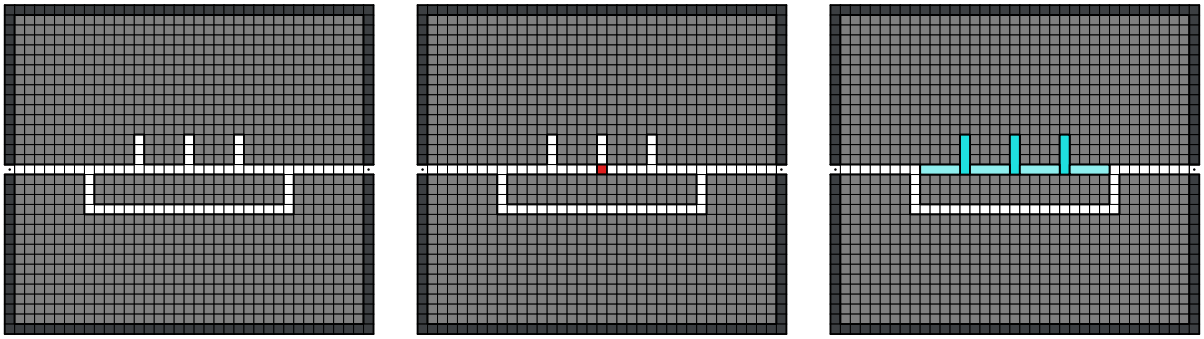
图13： 入口拐角（EC）构件。

如果方块从右端口进入。由于方块可以在必要时转过拐角，因此也可以进行这些铺砌。



(a) (b)装置中心方格

图14： 垂直线（VL）构件。



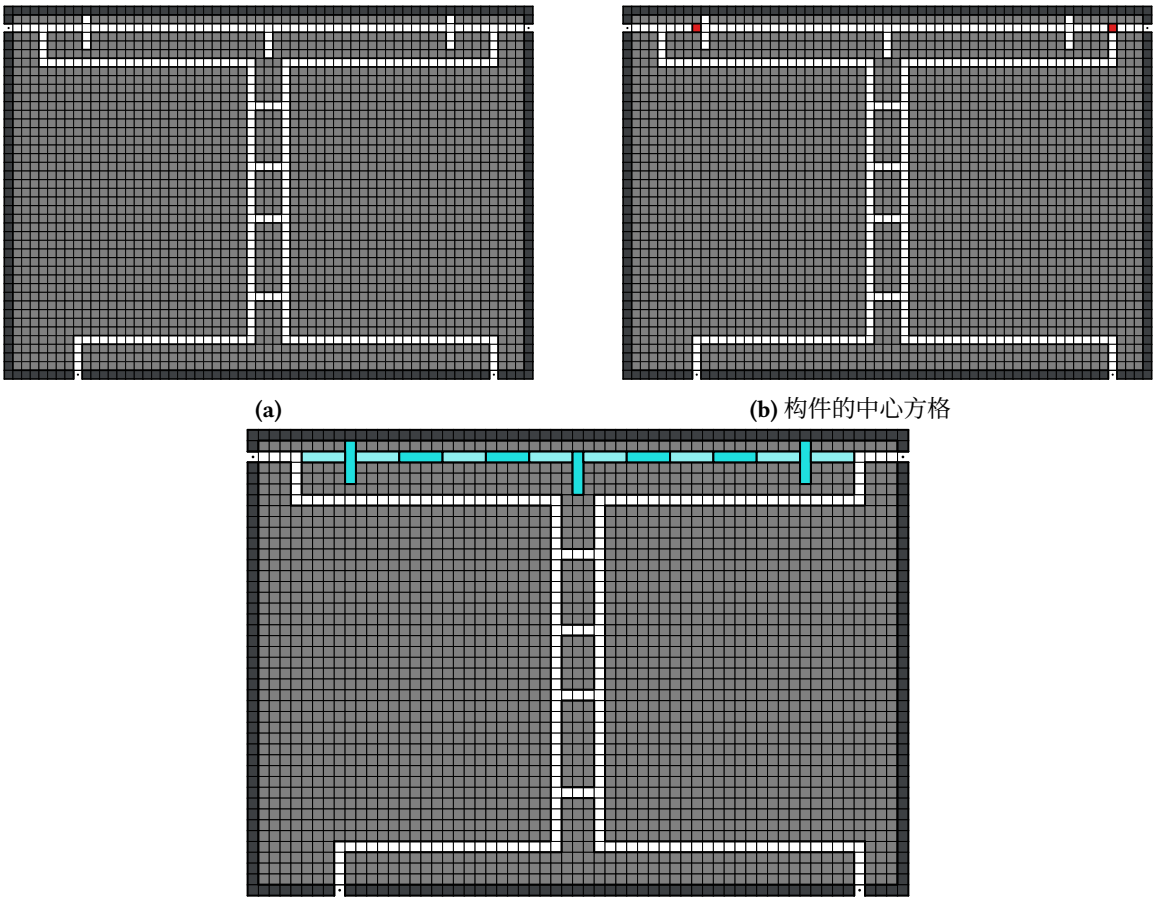
(a) (b)装置中心方格 (c) 强制放置

图15： 水平线（HL）构件。

5.2.2 复制器构件

复制器构件和 HL 小工具一样，与 [HIN+17] 中的复制器构件类似，但在左端口和右端口附近稍作调整，并增加了方格，以便更轻松地从左端口遍历到右端口，反之亦然（因为玩家只需将水平放置的 -拼块从左向右滑动，或反向滑动）。复制器构件如图16所示。可以验证，左端口区域和右端口区域之间的“桥”只能按照图16c所示的特定方式铺设。

由于拼块可以转过拐角，因此从左端口或右端口到任一下端口的穿行都是可能的，左端口与右端口之间的穿行也同样可能。图65展示了所有可能的铺砌，以及确保拼块能正确铺满构件的放置顺序。请注意，每种铺砌都有两种放置顺序：一种适用于拼块从左端口进入，另一种适用于拼块从右端口进入。之所以能够实现这些铺砌，是因为拼块可以根据需要转过拐角。所有铺砌中，进入编号为48和60的拼块时所需的旋转，都要求拼块在旋转前处于默认方向；不过，如果拼块处于相反的（180°）方向，玩家仍然可以在左端口稍右侧的区域或右端口稍左侧的区域内，通过两次绕拼块同一端方格转动的逆时针旋转，将拼块转回默认方向（该操作的示意图详见附录B.2）。



(a) **(b)** 构件的中心方格

(c) 强制放置

图 16： 复制器构件。

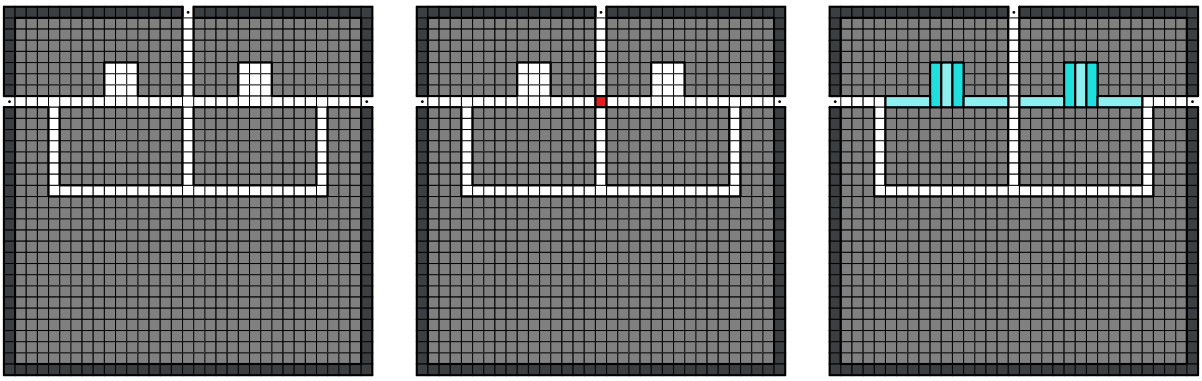
5.2.3 子句和否定子句构件

子句构件与其他构件一样经过了类似的改动并增加了额外方格，如图 17 所示。可以验证，左端口区域与右端口区域之间的“桥”只能按照图17c所示的特定方式铺砌。

由于方块可以转过拐角，因此从上端口到另外两个端口的穿行都是可行的，左右端口之间的穿行也同样可行。图 66 展示了所有可能的铺砌，以及能够确保方块正确铺满构件的放置顺序。

否定子句构件在中间部分还做了额外改动，以便在填充构件时，方块能够旋转到所需位置。否定子句构件如图 18 所示。可以验证，左端口和右端口区域之间的“桥”以及构件中间的额外结构，都被强制以图 18c 所示的特定方式铺砌。

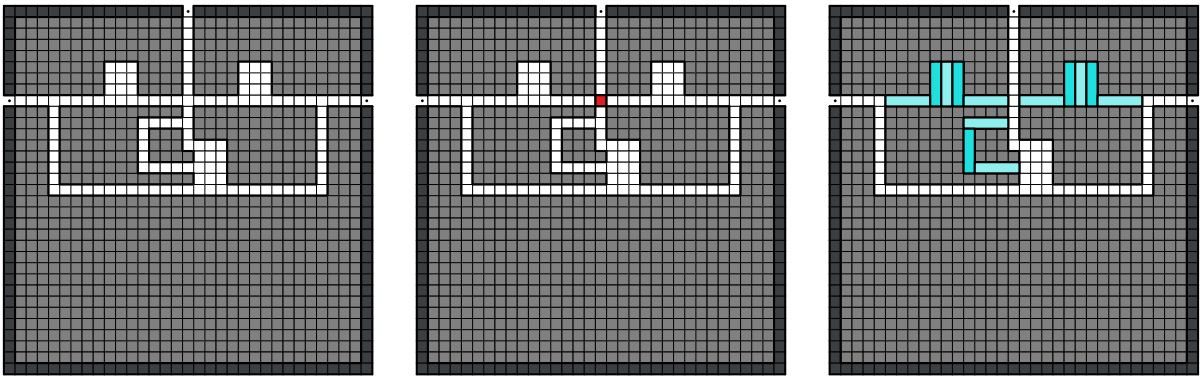
由于方块可以在拐角处转弯，从上端口到另外两个端口中任一端口的穿行都是可能的，左端口与右端口之间的穿行也同样可能。唯一可能的铺砌，以及为确保这些方块正确铺砌该构件而放置方块的顺序，如图 67 所示。值得注意的是第一种铺砌和第二种铺砌中编号为15的拼块——将该拼块移至该位置的操作，需要仔细注意该拼块在穿行经过中间



(a) **(b)**装置中心方格 **(c)** 强制放置

图 17： 子句构件。

结构时的朝向，但这是可行的，并在附录 B.3 中有详细说明。



(a) **(b)**装置中心方格 **(c)** 强制放置

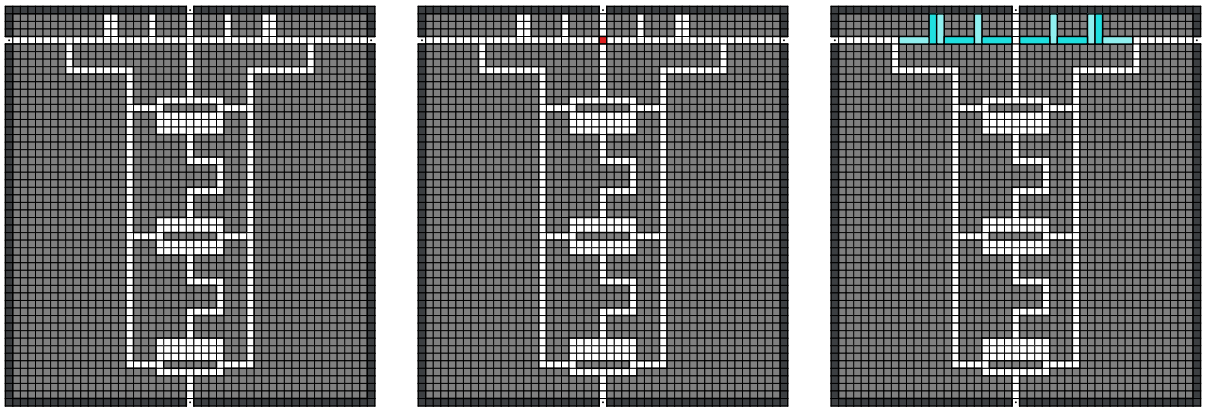
图 18： 否定子句构件。

5.2.4 交叉装置

图 19 所示的交叉装置是最复杂的构件，需要最详尽的说明。与 [HIN+17] 中的交叉装置相比，左端口和右端口附近略有改动，并增加了方格，以便更轻松地从左端口遍历到右端口，反之亦然。我们将用 腔室 一词来指代中间三个仅填充了一个 9×5 矩形的矩形。可以验证，左端口区域与右端口区域之间的“桥”只能按图 19c 所示的特定方式铺砌。

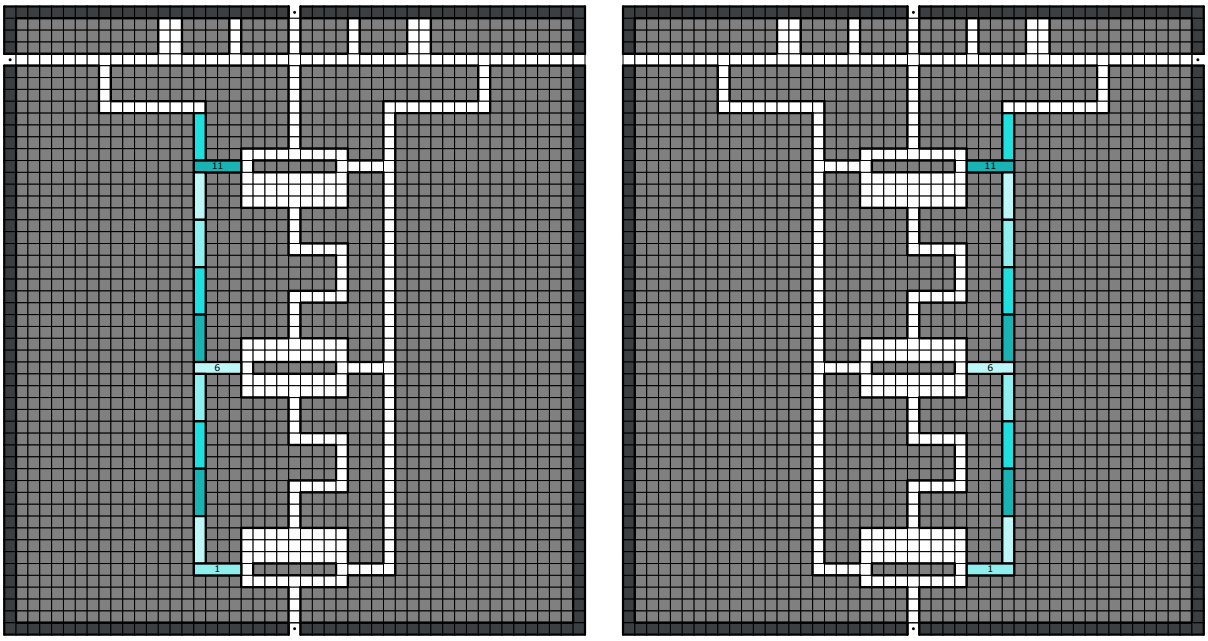
就遍历而言，从上端口到左端口、从上端口到右端口，以及在左、右端口之间的穿行都显然可行。不过，交叉装置有其特殊之处：根据所需的铺砌方式，如图 20 所示，我们希望在左侧或右侧预先放置一些方块。这样做是为了确保进行必要的旋转时，目标测试之前的所有墙踢测试都会失败，从而便于穿行至下端口。采用图 20 中的方块放置方式，可以实现从任何非下端口到下端口的所有穿行（有关如何让方块穿过构件中部腔室的操作示例，请参见附录 B.4）；此外，图 20 中的方块放置方式及顺序在构件的初始状态下都是可行的。

唯一可能的铺砌方式，以及为确保方块正确铺砌构件而放置方块的顺序，见图68。



(a) **(b)**装置中心方格 **(c)** 强制放置

图 19: 交叉装置。



(a) 左侧版本 **(b)** 右侧版本

Figure 20: 使交叉装置的下端口可访问的初始方块放置。

5.3 综合运用

现在我们描述构造的主体部分。取图定向实例，并将其嵌入俄罗斯方块棋盘，使得水平方向相邻的红点位于同一行，垂直方向相邻的红点位于同一列，且每个红点的坐标形式为（$60i + 38$, $48j + 37$），其中 i 和 j 为整数（这里，（1，1）表示棋盘左上角的方格）。对于每一对对应于一对文字的水平相邻红点，放置一个复制器构件，使复制器

构件的中心方格恰好落在红点上。在最靠上且最靠左的红点处，放置一个入口角部装置，使其中心方格恰好落在红点上。否则，在每个红点处放置相应的构件（角构件、垂直线、水平线、子句、否定子句、交叉），使其中心方格恰好落在红点上。

此外，使用水平连线（即一个 $1 \times n$ 矩形）或垂直连线（即一个 $n \times 1$ 矩形）连接相邻的构件。由于模4约束，所有连线的长度都比4的倍数少1，因此对于每条连线，两端的两个端口中恰有一个必须由用于铺砌该连线的方块覆盖，这有助于保留/传递有关构件朝向的信息。

5.4 正确性

既然我们已经介绍了构造的所有部分，现在证明该构造的正确性；换言之，当且仅当1-in-3SAT实例有解时，整个构造才能在SRS下用方块清除。根据第5.1小节中的讨论，当且仅当构造的主体部分能在SRS下用方块填满时，整个构造才能在SRS下用方块清除。

首先，如果主体部分在 SRS 规则下可以用拼块填满，那么我们必然能够用拼块铺砌该构造的主体部分。由此，我们可以读出各构件的朝向，从而得到指定向实例中各边的定向，进而得到1-in-3SAT实例的解。

现在，假设我们有一个 1-in-3SAT实例的解。利用这个解，我们可以快速（即在多项式时间内）确定图定向实例中各边的定向，进而确定各构件以及连接构件的连线的朝向。接下来，我们要证明能够通过操作，将拼块移至并放置在构造主体部分中的预定位置。

关键思路是利用 [HIN+17 中图定向实例的结构，见定理 2.3]。我们先填充构造中对应于下部的构件，定义见第5节开头。由于下部由 m 个图9所示的子句/否定子句构造副本组成，因此可以按照图21所示的相同方式填充这里的构件：

1. 沿从最左侧文字 ℓ_1 到子句构件 C_j 的路径填充构件，并连同子句构件 C_j 右侧的角构件一起填充，顺序为从路径上最底部最右侧的构件到最顶部最左侧的构件。2. 沿从最右侧文字 ℓ_3 到否定子句构件 C_j 的路径填充构件，并连同否定子句构件 C_j 左侧的角构件一起填充，顺序为从路径上最底部最左侧的构件到最顶部最右侧的构件。3. 沿倒数第二行填充剩余构件，从最左侧和最右侧的剩余构件开始，向中间列移动。

4. 沿倒数第三行从右到左填充剩余构件。

5. 从左到右填充剩余构件。

所有带有上端口的构件将由穿过上端口的方块填充，而所有其他构件（没有上端口）将由穿过仍未阻塞的左端口或右端口的方块填充（即，从某个文字到该构件仍存在一条未阻塞路径）。

构造的下部对应的构件全部填满后，我们就可以填充构造的上部对应的构件。为此，我们首先填满位于最左侧含有文字的垂直线路左侧的所有构件，

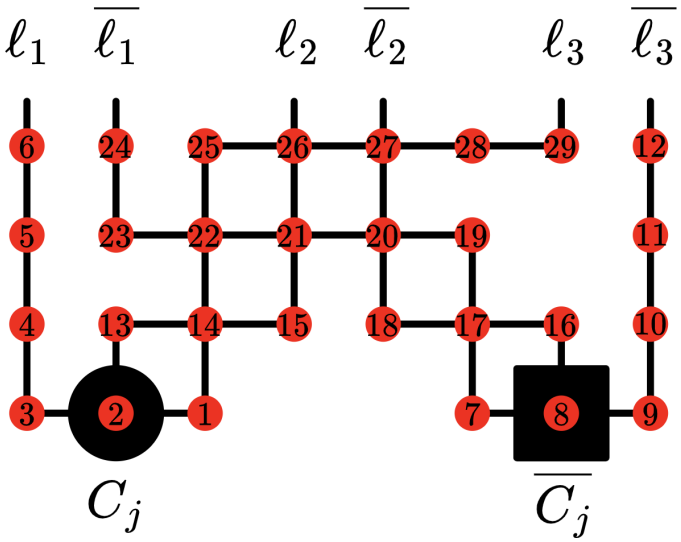


图21： 在每个子句/否定子句构造中填充各个构件的顺序。数字较小的先填充（因此，1 最先填充，29 最后填充）。

但不包括 EC组件，然后填充其余构件，从最底部一行开始，并在每行构件中从右向左进行。请注意，构造上部中带有上端口的构件只有交叉点和 UC组件，而且在整个构造中，将相关方块操作至这些构件的上端口很容易；除此之外，构造上部的其他构件在填充时，对方块的来源没有任何限制。

填充构件时，我们也会填充构件之间的连线——每当准备填充一个构件时，我们都会检查所有相邻的构件；如果其中任何构件已经填充，就会在填充当前构件之前，先填充连接当前构件与已填充构件的连线。

具体而言，这种在整体构造中放置方块的方法确保任何构件或连线都不会过早被堵住或无法进入，并且只要给定构件的有效朝向，所有构件和连线都能得到妥善填充。因此，给定 1-in-3SAT实例的一个有效解，我们就能确定一种在 SRS 规则下用方块填充构造主体部分的方法。

因此，当且仅当 1-in-3SAT实例有解时，才能在 SRS 规则下用方块填充构造的主体部分，这意味着我们的构造达到了预期效果。由此，我们证明了即使玩家只获得方块，采用 SRS 的俄罗斯方块消行仍然是 NP难问题，正如所需。

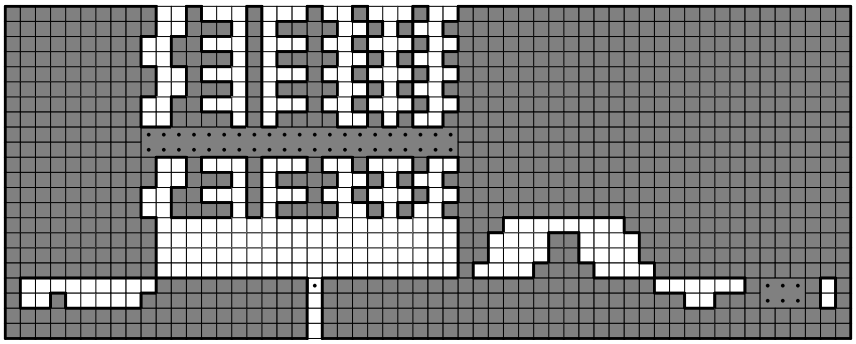
5.5 补充说明：袋

既然我们已经证明，即使玩家只拿到方块，俄罗斯方块消行仍然是 NP 困难的，那么实际上我们可以证明，在采用7袋随机生成器的情况下，俄罗斯方块消行也是 NP 困难的：

定理5.2。 即使方块序列必须能够由7袋随机生成器生成，采用 *SRS* 的俄罗斯方块消行问题仍然是 *NP*难。

这里，我们将定理5.1中的归约与 [MDHL24]中归约到两种方块类型时所采用的瓶形阻挡机制结合起来。更具体地说，设 $N + 1$ 表示清除定理5.1中归约结构所需的方块数量（包括从右数第二列中的四个空方格

）。然后，我们修改该结构的顶部两行，并在结构顶部额外添加 $2N + 4$ 行，如图22所示（图中展示的是 N 为偶数时的情况；如果 N 为奇数，则左侧隧道在顶部各行中的偏移会略有不同）。顶部的 $2N$ 行大致由同一个2行结构重复 N 次组成。输入序列由 $N + 5$ 个袋组成，共计 $7N + 35$ 个方块。这里，每个袋中方块的顺序并不重要。



Figu再22： 在 7 袋随机化下归约到俄罗斯方块消行所需的额外结构 er.

对于前 N 袋，非 形状的方块必须填入顶部两行中各自对应的隧道。具体来说，由于方块的放置位置是被强制确定的（否则方块的一部分会伸出最顶行），其他非 形状的方块必须以特定方式放置，使顶部两行被清空，以便正确放置下一个方块。这就迫使方块按图23所示放置。

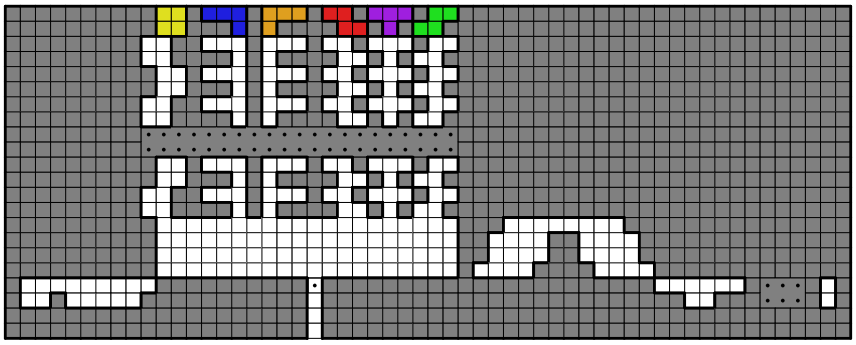
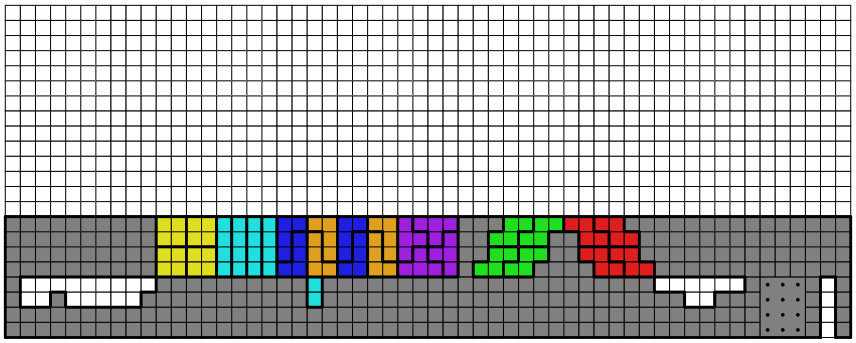


图23： 前 N 袋中非 形状方块的放置方式。

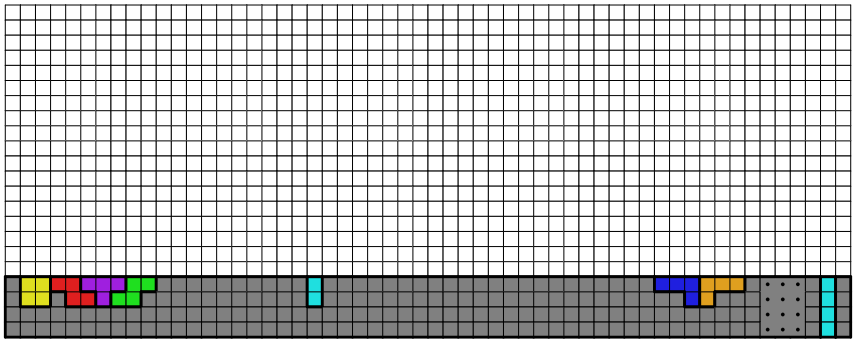
现在，对于前 N 袋，如果该方块是袋中的第一个或最后一个，那么所有隧道都会保持开放（因为非 形状的方块要么尚未放置，要么已经清除了顶部的行），因此我们可以让该方块穿过任意一条隧道，进入主归约结构的入口。否则，顶部两行中会放置 1 到 5 个不同的非 形状方块，这意味着至少有一条隧道仍然开放，因此我们可以让该方块穿过那条隧道，进入主归约结构的入口。无论哪种情况，每袋之后，无论袋中方块的排列顺序如何，顶部 2 行都会被清除，并且恰好有一个额外的方块被放入主归约结构中。

最后5袋用于清理剩余的行；图24展示了一种清理剩余行的可能方法。

额外结构并未显著改变正确性论证。消行策略仍必须采用以下形式：“填满嵌入的图定向结构，再填满长隧道，然后清除主归约结构周围顶部两行和底部两行”，但现在额外结构要求在整个结构顶部定期消行。未能填满



(a) 前4袋



(b) 最后一袋

图24： 清理剩余的行。

嵌入的图定向结构仍会导致最后一步（清除主归约结构周围顶部两行和底部两行）失败，因为主归约结构中的中间行没有被消除。此外，由于通往长隧道的入口宽度有限，只有方块可以进入主归约结构；所有其他方块都必须放置在顶部 $2N + 4$ 行内。因此，即使要求方块序列必须能够由7袋随机生成器生成，俄罗斯方块消行问题仍是 NP 困难的。

我们对这个归约再做一些补充说明：

- 如果允许玩家暂存方块，甚至可以任意调整输入序列中方块的顺序，即使最终顺序并非由7袋随机生成器生成，也依然可行。能够延迟放置特定方块（甚至选择方块的放置顺序）并不会影响棋盘是否可消行，因为只有方块能进入主归约结构，而这才是清空整个棋盘的关键。

- 这个结构可以轻松修改，以证明对于任意正整数 $k \geq 1$，**7*k*-袋随机生成器** 也是 NP 困难。首先，我们可以对主归约结构稍作调整，使 N/k 为整数，并给玩家 $N/k + 5$ 袋。然后，我们制作 k 个附加结构副本，它们现在占据 $2N/k + 4$ 行，并将它们并排放置在顶部 $2N/k + 4$ 行中，但只有一个副本通向主归约结构。仍然可以论证，在前 N/k 袋中，恰好有 k 个额外方块被放入主归约结构中，不过论证稍微复杂一些（特别是，基本思路是优先将非方块放入那些不通向主归约结构的附加结构副本中）。

6 -tris/ -tris消行

定理6.1。 即使给玩家的序列中方块的类型仅限于方块，或给玩家的序列中方块的类型仅限于方块，采用 *SRS* 的俄罗斯方块消行问题也是 *NP*难。

我们关注 方块； 方块的情形是对称的。

在此，我们将从平面双连通{4中取{0,4},4中取3}图定向进行归约。我们首先描述该构造的总体结构，它与-tris归约的总体结构相似，并如图25所示。构造的主体部分对应于平面双连通{4中取{0,4},4中取3}图定向实例，以绿色标出。有一条通向构造主体部分的空方格路径，它主要位于棋盘最左侧12列或最顶部8行之内，我们称之为“长隧道”；我们还将第8列与第11列之间（从左起）的垂直结构称为“可攀爬垂直结构”。棋盘最右侧三列中还有另外4个空方格。我们给玩家的方块数量刚好使得完成棋盘消行必须用上所有方块；换言之，如果构造中有 M 个空方格，我们就给玩家 $M/4$个方块。

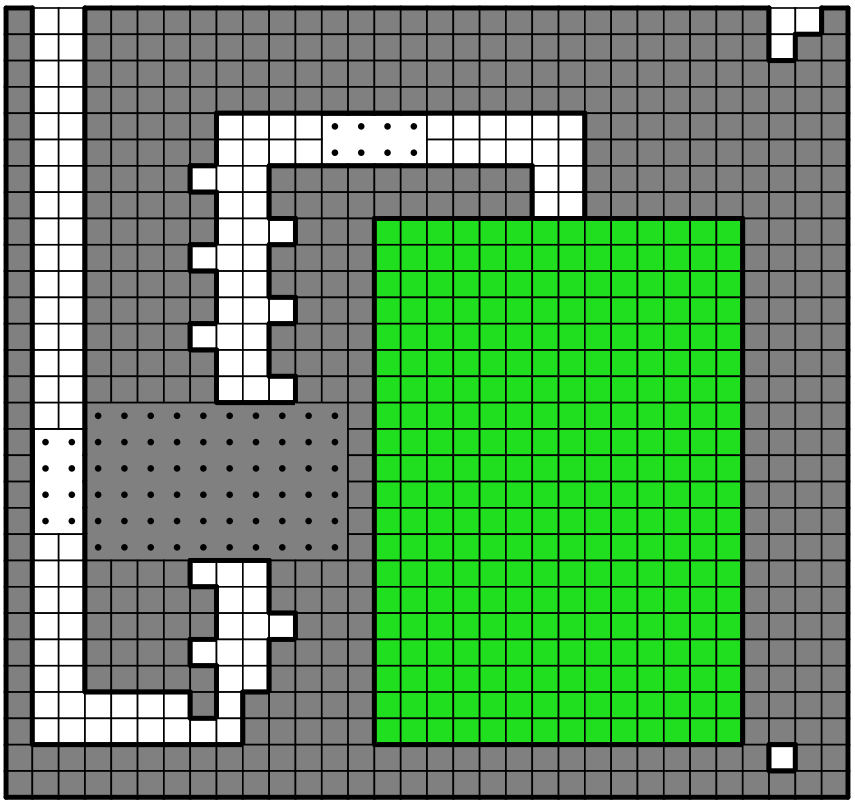


图25：仅使用方块进行俄罗斯方块消行的构造总体结构。构造的主体部分以绿色标示。

与 -tris归约类似，我们注意到，最右侧三列中的 4 个空方格必须是最后填充的方格，也就是在中间行消行之后。此外，如图 26 和 27 所示，用方块填充／铺满长隧道的方式恰好只有一种，而且可以（在 SRS 规则下）将方块穿过长隧道（甚至是可攀爬垂直结构）送入构造的主体部分（参见附录C.1中的操作）。在长隧道中锁定任何方块

都会在构造的主体部分完全填满之前封住其通道，导致棋盘变得无法消行。因此，在我们填满构造的主体部分之前，不能将任何方块放入长隧道。最后，在我们将任何方块放入长隧道之前，长隧道也会阻止与构造主体部分对应的行消行。

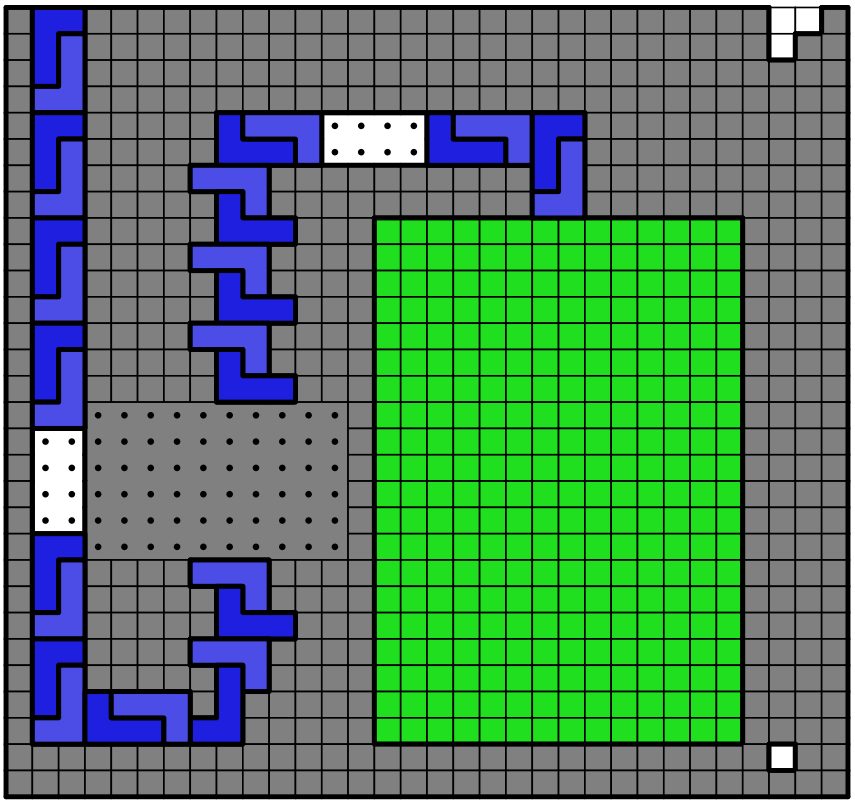
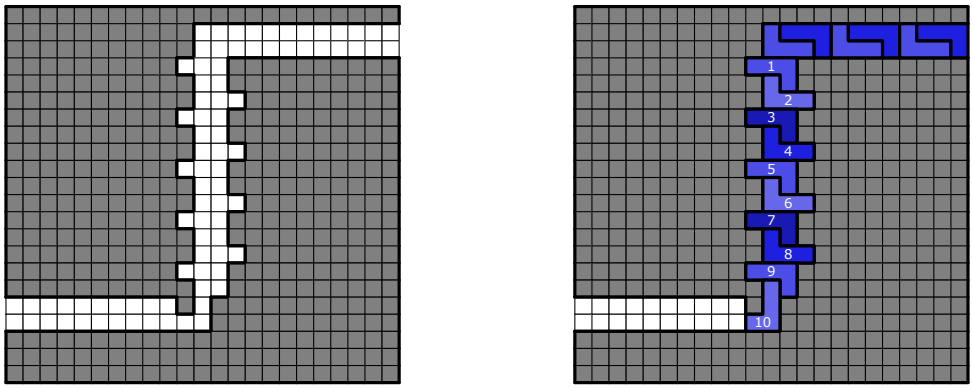


图26： 使用方块铺砌可攀爬垂直结构的唯一方式。



(a) 未填充　　**(b)** 已填充

图27： 进一步观察可攀爬垂直结构的一部分，其中包括该结构一个片段的方块排序。

因此，我们需要能够恰好用方块铺满构造的主体部分，同时还需满足一项额外限制：能够通过俄罗斯方块和 SRS 将所有方块操作到位。

规则。

6.1 构件

现在，我们介绍构造主体部分中使用的构件。与 -tris 构件一样，我们将入口称为端口，并用中心带点的空方格表示。我们需要 0 或 4 构件（对应入度必须为 0 或 4 的顶点）、4 中取 3 构件（对应入度必须为 3 的顶点）、水平线构件和 U 形转弯构件（对应顶点之间的边），以及一种名为奇偶校正器的特殊构件；如果不同构件的端口坐标模 4 后不相同，它可以帮助将这些端口的坐标模 4“对齐”。

与 -tris 构件一样，我们希望这些构件具有以下性质：

- 从具有较高 y坐标的端口到较低 y坐标的端口的任何遍历，或在具有相同 y坐标的端口之间的遍历，都是可能的。
- 构件的可能铺砌方式，要么对应于顶点周围各边的一种设定（若端口被铺砌中的一块拼块填充，则该边指向顶点；若端口未被铺砌中的一块拼块填充，则该边背离顶点），这适用于 0 或 4 构件和 3 入 4 构件的情形；要么对应于一条边的设定，这适用于其他构件的情形。
- 对于每一种可能的铺砌，每个单独的构件都可以在 SRS 下用方块填充：若构件有上端口，则方块来自上端口，否则方块来自左端口或右端口。（换言之，方块能够移动到铺砌中它们需要到达的位置，尤其是在构件被方块部分填充时。）

6.1.1 水平线和 U 形转弯构件

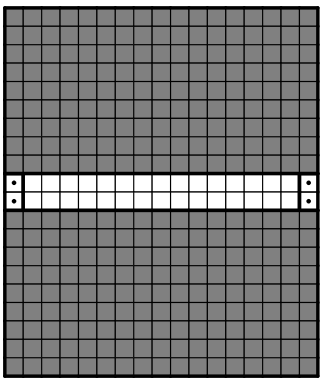
水平线（HL）构件和两种 U 形转弯（UL、UR）构件（分别为左-下-右版本和右-下-左版本）及其端口如图28所示。HL 构件以及 UL 和 UR 构件的垂直部分可根据需要按 4 的倍数延长。这些构件均不会改变端口坐标的模4值。

假设拼块处于默认方向或 180° 朝向（此时拼块高两格），HL 构件中显然可以实现端口之间的穿行。对于任一种 U 形转弯构件，通过在底角处进行若干次旋转并使用一次墙踢，任何拼块都可以从上端口穿行至下端口，并在下端口处保持默认方向或 180° 朝向。可能的铺砌方式，以及确保拼块正确铺砌构件的放置顺序，见图69和图70。

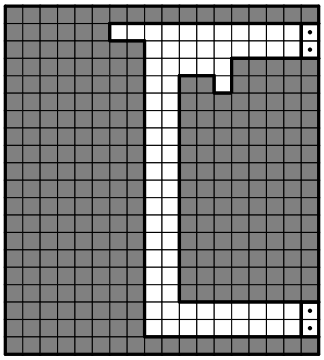
由于方块需要进入构件的主体部分，我们创建了一个特殊的入口角（*EC*）构件，如图 29 所示。由于长隧道的结构，构件的左上部分必须以特定方式铺砌，才能使构件的大部分发挥普通左上转弯装置的功能。借助必要的旋转和墙踢，可以从构件的上部遍历至两个端口中的任意一个，并使方块最终处于默认方向或 180° 朝向。图 71 展示了所有可能的铺砌方式，以及一种确保方块能正确铺砌构件的放置顺序。

6.1.2 0 或 4 和 4 中取 3 构件

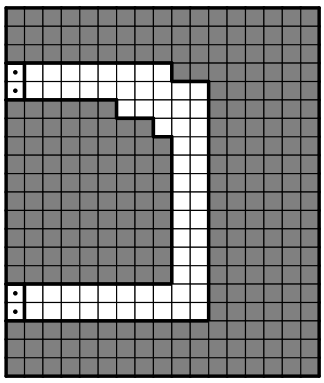
图30展示了 0 或 4 和 4 中取 3 构件及其端口。构件中的端口模 4 不一致，因此需要奇偶性修正构件。



(a) 水平线（HL）



(b)U 形转弯（UL；左-下-右）



(c) U 形转弯（UR；右-下-左）

图28

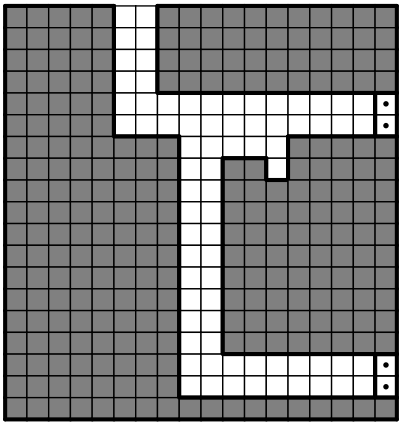


图29：入口拐角（EC）构件。

从上方端口到下方端口的穿行，以及构件的填充，可以通过旋转和一些踢墙来完成，路径经过构件的弯折处和中心区域。

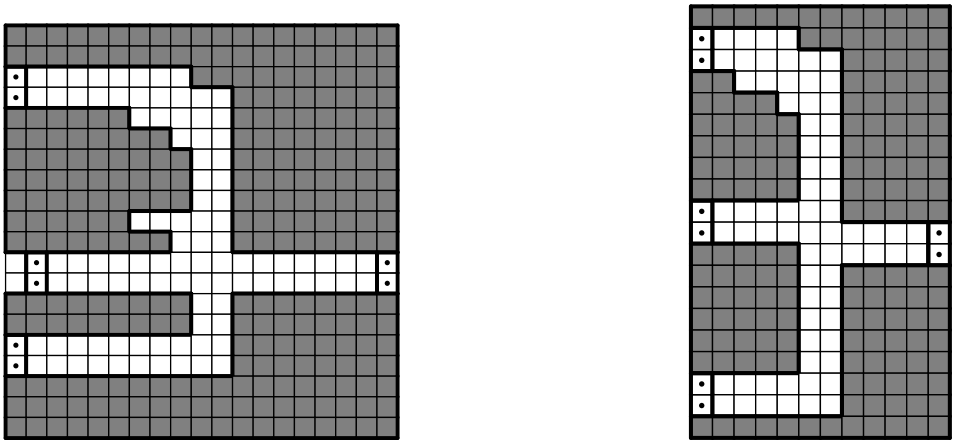
唯一可能的铺砌方式，以及一种确保方块正确铺砌该构件的放置顺序，见图72和图73。

6.1.3 奇偶校正构件

两个奇偶校正器（PF）构件及其端口如图31所示。上方的奇偶校正器将多米诺骨牌端口偏移（+1, +2）模4；下方的奇偶校正器将多米诺骨牌端口偏移（$-1, +1$）模4（此处，$+x$ 表示向右，$+y$ 表示向上）。这两个奇偶校正器共同涵盖了模4下所有可能的（x, y）组合。

通过旋转以及在构件弯折处踢墙，可以从上方端口穿行至下方端口，并完成构件的填充。

由于这些构件将连接到其他构件，而后者对铺砌方式有特定限制，因此每个奇偶校正器只有两种可能的铺砌方式，分别对应于覆盖其中一个多米诺骨牌接口而不覆盖另一个。图74展示了这些可能的铺砌方式，以及一种方块放置顺序；按照该顺序放置方块，可以确保方块正确铺砌该构件。



(a) 0 或 4　　(b) 4 中取 3

图30： 0 或 4 构件和 3 入 4 构件。

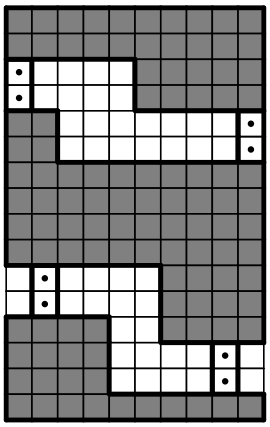


图31： 两个奇偶校正器（PF）构件。

6.2 将所有内容整合起来

现在我们描述构造的主体部分。取平面双连通的 {{0, 4}-in-4, 3-in-4} 图定向实例，并使用 [BK98] 中的线性时间 BICONN 算法，计算该图在网格上的嵌入。计算得到的嵌入还具有一些特殊性质，即：如果在最顶部边的左上方“弯折”处添加一个人工的“根顶点”，那么所有顶点都具有不同的 y坐标，并且每个点（顶点或边上的点）都可以通过一条从根顶点出发的“向下”（即单调递减 y坐标的）路径到达。我们将后一种性质称为向下可达。

接下来，我们将网格图的绘图嵌入俄罗斯方块棋盘，并在顶点周围留出足够的间距；在根顶点放置一个入口角部装置，并在相应的顶点放置 0 或 4 构件和 4 中取 3 构件（前者用于入度必须为 0 或 4 的顶点，后者用于入度必须为 3 的顶点）。最后，根据需要使用水平线、两个 U 形转弯以及两个奇偶校正器，连接 0 或 4 构件和 4 中取 3 构件（这些构件起到边的作用，并传递有关构件朝向的信息）。

6.3 正确性

既然我们已经介绍了构造的所有部分，现在就来证明该构造的正确性；换言之，当且仅当平面双连通 {{0，4}-in-4，4 中取 3} 图定向实例有解时，整个构造才能在 SRS 下用方块清除。根据第6节开头的讨论，当且仅当构造的主体部分能在 SRS 下用方块填满时，整个构造才能在 SRS 下用方块清除。

首先，如果在 SRS 规则下，构造的主体部分可以用方块填满，那么我们必然能得到主体部分的一种铺砌。从中，我们可以读出各构件的定向，从而确定各边的朝向，这就构成了平面双连通 {{0，4}-in-4，4 中取 3} 图定向实例的一个解。

现在，假设我们有一个平面双连通 {{0，4}-in-4， 4 中取 3} 图定向实例的解。利用这个解，我们可以快速（即在多项式时间内）确定所有构件的朝向。接下来，我们要证明可以通过操作，将方块移至构造主体部分中的目标位置。

关键思路是利用图嵌入的特殊性质。总体过程如下：

1. 首先，填充位于构造主体部分最底部各行上或与其接触的水平线和 U 形转弯构件（参见 [BK98] 的图2中最右侧的图）。2. 然后，从最低的 0 或 4 或 4 中取 3 构件开始向上操作，填充与通向“较低”（较小的 y-坐标）0 或 4 或 4 中取 3 构件的各条边相对应的水平线、U 形转弯或奇偶校正构件中的任意构件，随后填充该 0 或 4 或 4 中取 3 构件本身。3. 所有 0 或 4 或 4 中取 3 构件都填充完毕后，只剩下入口角以及入口角周围的一些构件（水平线、U 形转弯、奇偶校正器）。在填充入口角本身之前，先填充入口角周围的构件。

特别地，由于所有顶点的 y 坐标各不相同，在判断哪个 0 或 4 构件或 4 中取 3 构件“更低”时，我们无需处理平局。此外，由于该嵌入向下可达，只要按照逆向广度优先搜索顺序（从根顶点开始）填充构件，就能保证填满主体部分构造中的所有构件。因此，给定一个平面双连通 {{0,4}-in-4,3-in-4} 图定向实例的有效解，我们就能确定一种在 SRS 下用方块填充构造主体部分的方法。

因此，当且仅当平面双连通 {{0,4}-in-4, 3-in-4} 图定向实例有解时，构造的主体部分才能在 SRS 下用方块填充，这意味着我们的构造达到了预期效果。由此，我们证明了即使玩家只能获得方块，采用 SRS 的俄罗斯方块消行问题仍然是 NP难的，正如所需。

根据对称性，使用 SRS 的俄罗斯方块消行是 NP难的，即使玩家只被给予方块 s.

7 -三角形消行

定理7.1. 使用 *SRS* 的俄罗斯方块消行是 *NP*难的，即使给予玩家的序列中的方块类型仅限于方块。

这里，我们将从平面双连通 {{0,4}-in-4,1-in-4} 图定向问题进行归约。我们首先介绍该构造的一般结构，它与 -tris 和 -tris 归约的一般结构相似，并如图32所示。该构造的主体部分，对应于

平面双连通 {{0, 4}-in-4,1-in-4} 图定向实例，以绿色标出。一条由空方格组成的路径通向构造的主体部分；主体部分大多位于棋盘最左侧的23列或最顶部的11行内，我们将这条路径称为“长隧道”；我们还将从左数第7列和第22列之间的垂直结构称为“可攀爬垂直结构”。棋盘最右侧的三列中还有4个额外的空方格。如果构造中有 M 个空方格，我们就给玩家 $M/4$ 个方块。

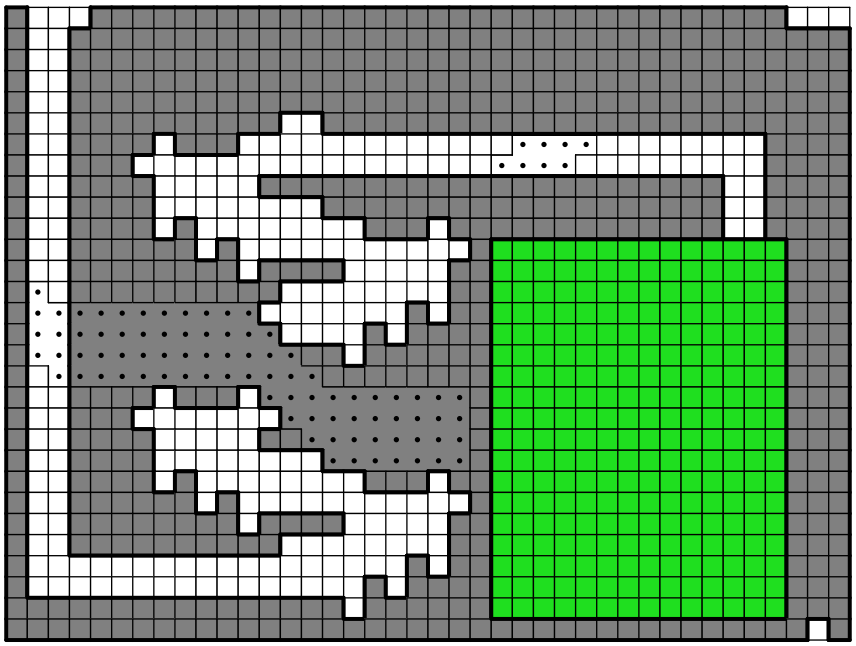


图32： 仅使用方块进行俄罗斯方块消行的构造整体结构。构造的主体部分以绿色标示。

与 -tris归约类似，我们注意到，最右侧三列中的4个空方格必须是在中间行消除后最后填入的方格。此外，如图33和图34所示，用方块填满长隧道恰好只有一种方式；而且，方块可以通过长隧道（甚至是可攀爬垂直结构）进入构造的主体部分（在 SRS 规则下；见附录D中的操作）。在构造的主体部分完全填满之前，若将任何方块锁定在长隧道内，就会堵住通往主体部分的通道，使棋盘无法消行。因此，在主体部分填满之前，不能将方块放入长隧道。最后，在我们将任何方块放入长隧道之前，长隧道还会阻止与构造主体部分对应的行消行。

因此，我们需要能够恰好用方块铺满构造的主体部分，此外还必须能够按照俄罗斯方块和 SRS 规则操作所有方块，使它们就位。

7.1 构件

现在，我们介绍构造主体部分所用的构件。与 -tris 构件类似，我们将入口称为**端口**并用中心带点的空方格表示。我们需要 0 或 4（对应入度必须为 0 或 4 的顶点）、1-in-4（对应入度必须为 1 的顶点）、水平线构件和 U 形转弯构件（对应顶点之间的边），

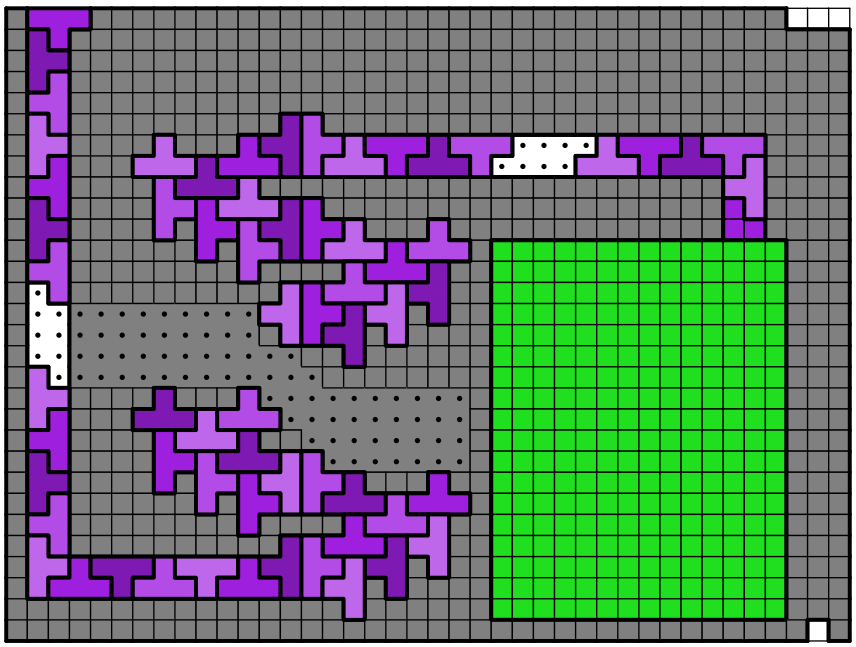
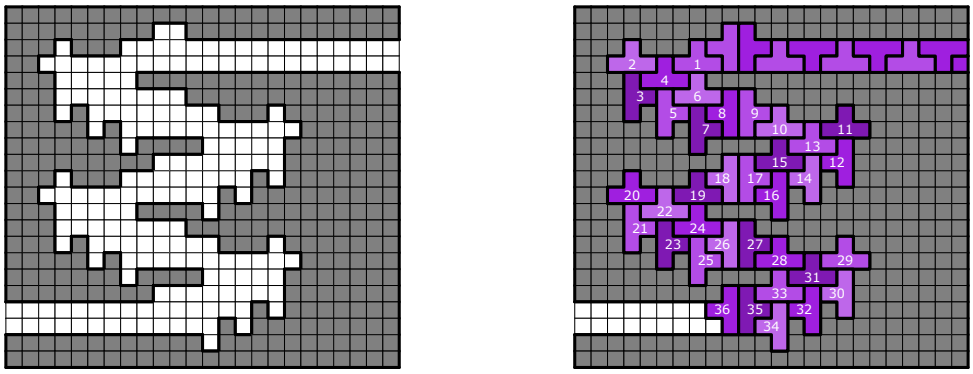


图33：用拼块铺砌可攀爬垂直结构的唯一可能方式。



(a) 未填充　　**(b)** 已填充

图34：近距离观　　察可攀爬垂直结构的一部分，包括其中一个片段的方块排序
攀爬结构的片段 垂直结构。

以及一种称为奇偶校正器的特殊构件，用于在各构件端口的坐标模 4 不同时，帮助将这些端口的坐标“对齐”到模 4 意义下。

与 -tris 构件类似，我们希望这些构件具有以下性质：

- 从较高 y坐标的端口遍历到较低 y坐标的端口，或在具有相同 y坐标的端口之间遍历，都是可行的。

- 构件的可能铺砌方式，要么对应于顶点周围边的设置（若铺砌中一个端口被拼块填充，则相应的边指向该顶点；若端口未被拼块填充，则边从该顶点指出），适用于 0 或 4 构件和 1-in-4构件；要么对应于一条边的设置，适用于其他构件。

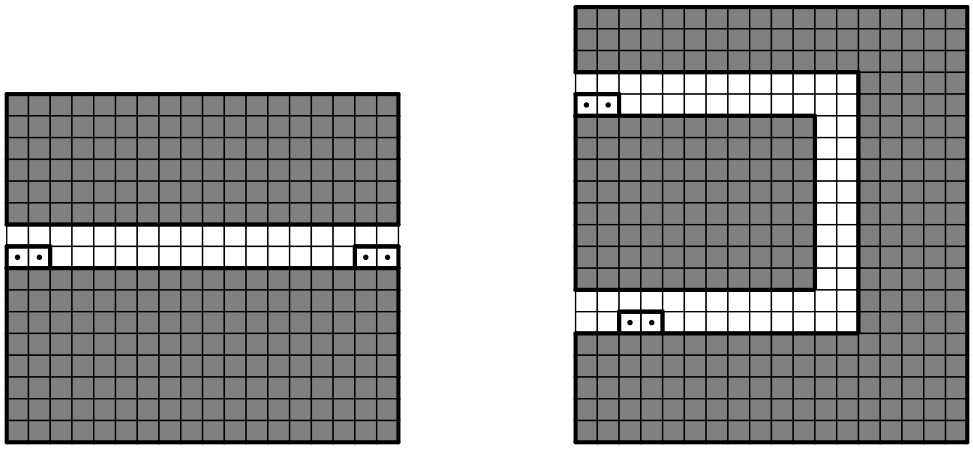
- 对于每一种可能的铺砌，每个构件都可以在 SRS 规则下用方块填充，如果构件有上端口，方块就从上端口进入；否则，方块就下

左端口或右端口进入。（换句话说，方块能够移动到铺砌中需要到达的位置，尤其是在构件已部分填充方块时。）

7.1.1 水平线与 U 形转弯构件

水平线（HL）和 U 形转弯构件及其端口如图35所示（图中仅展示了右-下-左版本；左-下-右版本与之对称）。HL 小工具和 U 形转弯构件的垂直部分可按需要延长 4 的倍数。HL 小工具不会改变端口坐标模4的值；不过，U 形转弯构件中的端口会偏移 (+2, +2) 模4。

假设方块处于默认方向或 180° 朝向（此时方块高两格），显然可以在 HL 小工具中穿行于各端口之间；而通过若干次旋转（并在填充过程中进行一些墙踢），任何方块都能从上端口穿行至下端口，并在下端口处以默认方向或 180° 朝向结束。可能的铺砌方式，以及确保方块正确铺砌构件的放置顺序，见图75和图76。



(a) 水平线（HL）　　**(b)** U 形转弯

图35

由于方块需要进入构件的主体部分，因此我们创建了一个特殊的入口角（*EC*）构件，如图36所示。由于长隧道的结构，构件的右上部分必须以特定方式铺砌，才能使构件的大部分发挥普通 U 形转弯构件的功能。通过必要的旋转和墙踢，可以从构件的上部遍历到两个端口中的任意一个，且方块最终会处于默认方向或 180° 方向。所有可能的铺砌方式，以及确保方块能正确铺砌构件的放置顺序，都显示在图77中。

7.1.2 0 或 4 和 1-in-4 构件

0 或 4 和 1-in-4 构件及其端口如图37所示。这些构件中的端口在模4意义下不一致，因此需要奇偶性修正构件。

通过在弯曲处和构件中心区域进行旋转以及一些踢墙，可以实现从上方端口到下方端口的穿行，并完成构件的填充。

所有可能的铺砌，以及确保方块正确铺砌构件时方块可放置的顺序，都显示在图78和图79中。

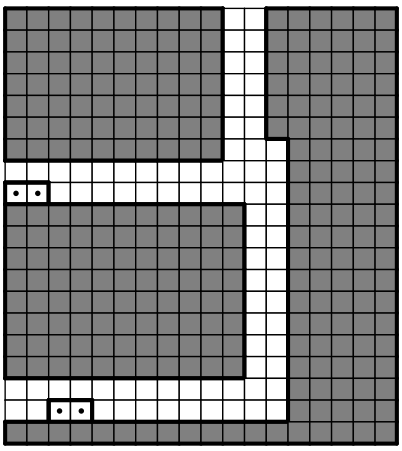
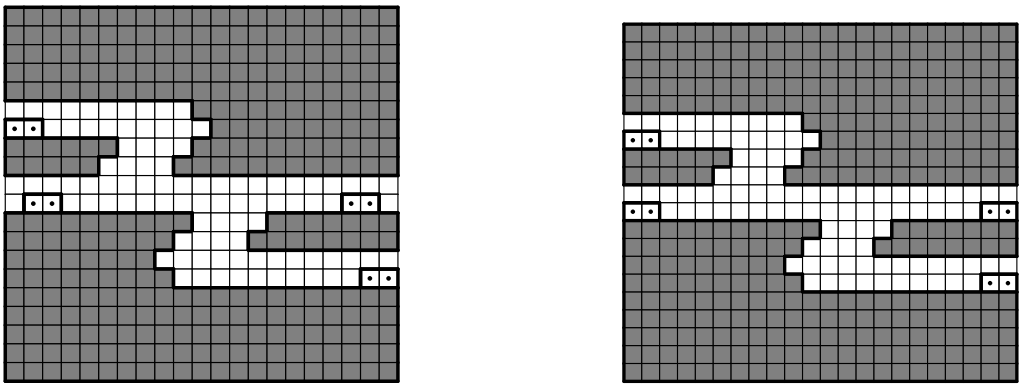


图36： 入口拐角（EC）构件。



(a) 0 或 4　　(b) 1-in-4

图37： 0 或 4构件和1-in-4构件。

7.1.3 奇偶校正构件

两个奇偶校正器（PF）构件及其端口如图38所示。第一个奇偶校正器将多米诺骨牌端口按 $(0,\ +1)$ 模 4 移动，第二个奇偶校正器将多米诺骨牌端口按 $(-1,\ +1)$ 模 -4 移动（此处，$+x$ 表示向右，$+y$ 表示向上）。这两个奇偶校正器共同覆盖模4下所有可能的 $(x,\ y)$ 组合。

通过旋转以及在构件弯折处进行一些墙踢，可以从上方端口穿行至下方端口，同时填充构件。

由于这些构件将连接到其他构件上，而后者对铺砌方式有特定限制，因此每个奇偶校正器只有两种可能的铺砌方式，分别对应于覆盖其中一个多米诺骨牌接口而不覆盖另一个。图 80 展示了这些可能的铺砌方式，以及一种放置方块的顺序，以确保方块能够正确铺砌该构件。

7.2 将所有内容整合起来

现在我们来介绍构造的主体部分，它与 -tris 构造类似。取平面双连通的 {{0,1}-in-4,1-in-4} 图定向实例，并使用 [BK98] 中的线性时间 BICONN 算法，计算该图在网格上的嵌入。和之前一样，如果在最顶部那条边的右上方“拐弯”处添加一个虚拟的“根顶点”，那么所有顶点的 y坐标都不相同，并且

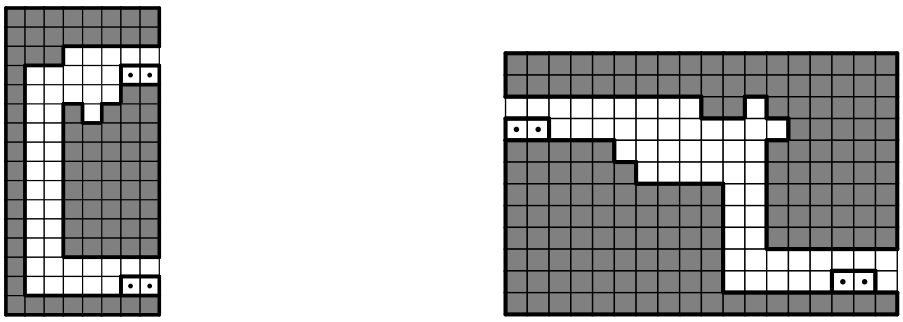


图38： 两个奇偶校正器（PF）构件。

该嵌入向下可达。

从这里开始，我们将网格图的绘图嵌入俄罗斯方块棋盘中，并在顶点周围留出足够的间距；在根顶点处放置一个入口角部装置，并在相应的顶点处放置 0 或 4 构件和 1-in-4构件（前者用于入度必须为 0 或 4 的顶点，后者用于入度必须为 1 的顶点）。最后，根据需要使用水平线、U 形转弯以及两个奇偶校正器，将 0 或 4 构件和 1-in-4构件连接起来（它们起到边的作用，并传递有关构件定向的信息）。

7.3 正确性

正确性论证与 -tris 的正确性论证密切相关。具体而言，如果构造的主体部分能够在 SRS 下用方块填满，那么我们必然可以得到一个用方块对构造主体部分进行铺砌的方案；这会给出构件的定向，进而给出平面双连通 {{0，4}-in-4，1-in-4} 图定向实例的一个解。此外，如果我们有一个平面双连通 {{0、4}-in-4，1-in-4} 图定向实例的解，那么我们就能快速地（即在多项式时间内）得到所有构件的定向。随后，我们可以采用与 -tris归约类似的方式进行操作，并将方块放置到构造主体部分各自所需的位置，从而得到一种在 SRS 下用方块填满构造主体部分的方法。

因此，当且仅当平面双连通 {{0,4}-in-4, 1-in-4} 图定向实例有解时，主体部分的构造才能在 SRS 规则下用方块填满，这意味着我们的构造按预期发挥了作用。因此，我们证明了即使玩家只能获得方块，采用 SRS 规则的俄罗斯方块消行问题仍然是 NP难的，正如所期望的那样。

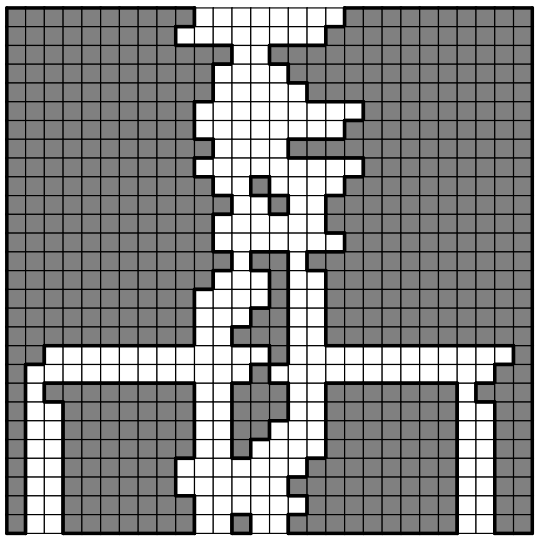
8 三角形/三角形消行

定理8.1。 即使给玩家的序列中方块的类型仅限于三角形，采用 *SRS* 规则的俄罗斯方块消行仍然是 *NP*难的；或者，即使给玩家的序列中方块的类型仅限于三角形，采用 *SRS* 规则的俄罗斯方块消行仍然是 *NP*难的。

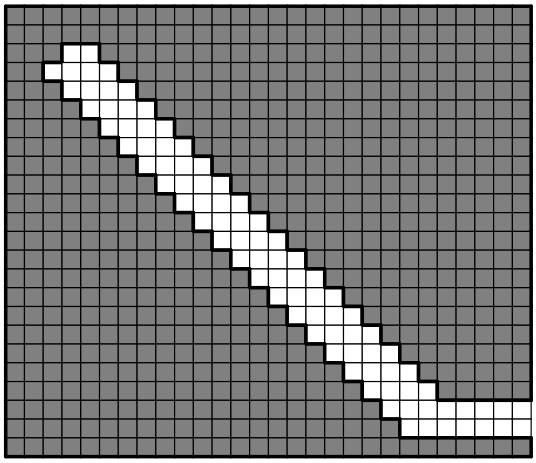
我们聚焦于 三角形； 三角形的情形具有对称性。

这里，我们将从平面单调直线 3SAT 进行归约。为此，我们首先构造变量构件和子句构件（以及用于连接它们的连线），并构造一个“可攀爬垂直结构”。这些构件如图 39 所示。

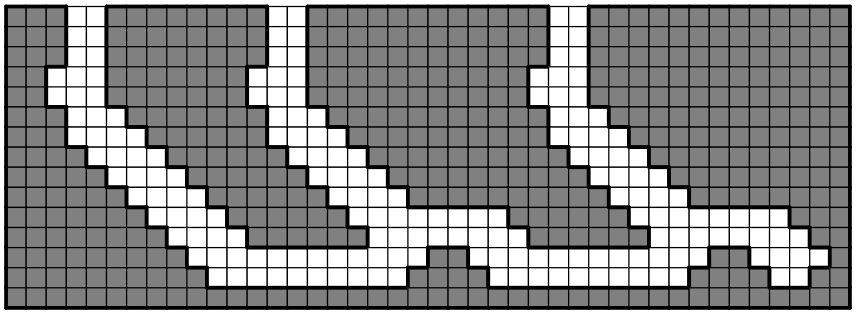
变量装置的核心思路是，根据初始拼块的放置位置，有两种设置方式，如图40所示；每种方式都允许方块通过其中一条隧道，但代价是



(a) 带有两条连线的变量构件（一条位于“真”侧，另一条位于“假”侧）



(b) 可攀爬垂直结构



(c) 子句构件

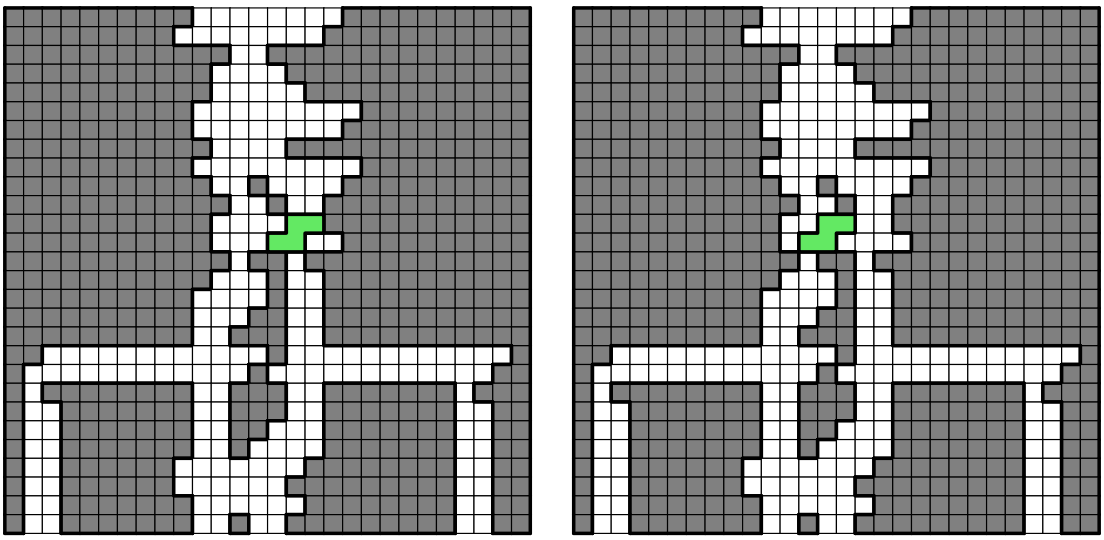
图39： 用于 -三角形的构件。

另一条隧道会被堵住。在能够消除行之前，不存在能让两条隧道同时畅通的方块放置方式。一旦可以消除行，就能将变量装置压缩成一种允许方块通过两条隧道的形态，如图 81 所示。附录 E.4、E.5 和 E.6 中描述的特定操作，可用于填充变量装置和/或穿过其中。

对于子句构件，它有三个入口，分别对应三条连接。一旦其中一个入口可以进入，子句构件的大部分区域都可以填满，可能只有少数几个方格除外，如图 82 所示。这些可能未被填满的方格不会影响我们后续的论证；如有必要，可以在最后将它们填满。不过，只要至少有一个入口可以进入，便能够填满的那些方格会促使一些行被消除，从而帮助结构的其他部分变得可进入。附录 E.1、E.2 和 E.3 中描述的特定操作可用于填充和／或穿过子句构件。

可攀爬垂直结构使我们能够防止在该结构变得可进入之前发生消行，并且可以从底部向上填充（见图 83）。附录 E.1 和 E.2 中描述的特定操作可用于填充和／或穿过可攀爬垂直结构。

现在我们已经有了这些构件，下面描述总体结构；示例见图 41。给定平面单调直线型3-可满足性问题的一个实例，我们可以取其平面直线型排布，



(a) “真”设置 **(b)**“假”设置

图40： -三角形变量设置。

将整个排布旋转90度，使第一条变量线段朝顶部方向旋转，再将每条子句线段各自旋转90度，并重新排列子句线段与连线，使排布仍保持平面且直线型。得到该排布后，我们可以在变量线段和子句线段所在位置插入变量装置和子句构件，用连线连接这些构件，并在最后一个变量装置下方添加一些断开的连线，通向可攀爬垂直结构；该结构向上延伸，并位于其余结构的左侧，止于顶行下方两行处。我们还在最上方的变量装置之上额外添加一份变量装置最上面两行的副本，并在最右侧的若干列中加入一些空方格，以防止最上面两行被过早消除。对于方块序列，若总体结构中有 M 个空方格，我们给玩家 $M/4$ 个方块，这样，若玩家想清空棋盘，任何方块的任何部分都不能放置在总体结构的最顶行之上。

8.1 正确性

现在我们证明该构造的正确性；换句话说，当且仅当平面单调直线 SAT 实例存在一个解时，整个构造能在 SRS 规则下用方块清除。

首先，如果该构造能够在 SRS 下被消除，那么在某种特定的变量子句设置下，每个子句都必须可以到达。具体来说，根据该结构的布局方式，以及任何拼块的任何部分都无法放置在结构最顶行上方这一事实，顶部两行必须最后消除（因为顶部两行中的空方格与最右侧几列中的其他空方格对齐），这意味着只有消除与子句构件能够消除的行相对应的行，才能到达可攀爬垂直结构。这意味着必定存在一种特定的变量子句设置方式，使每个子句都可以到达。换言之，存在一种变量赋值，使得每个子句至少包含一个真文字，这意味着该平面单调直线 SAT 实例有解。

现在，假设我们有一个平面单调直线 SAT 实例的解。利用该解，我们执行以下操作来消除该构造：

1. 根据解设置变量构件的状态（真、假）。

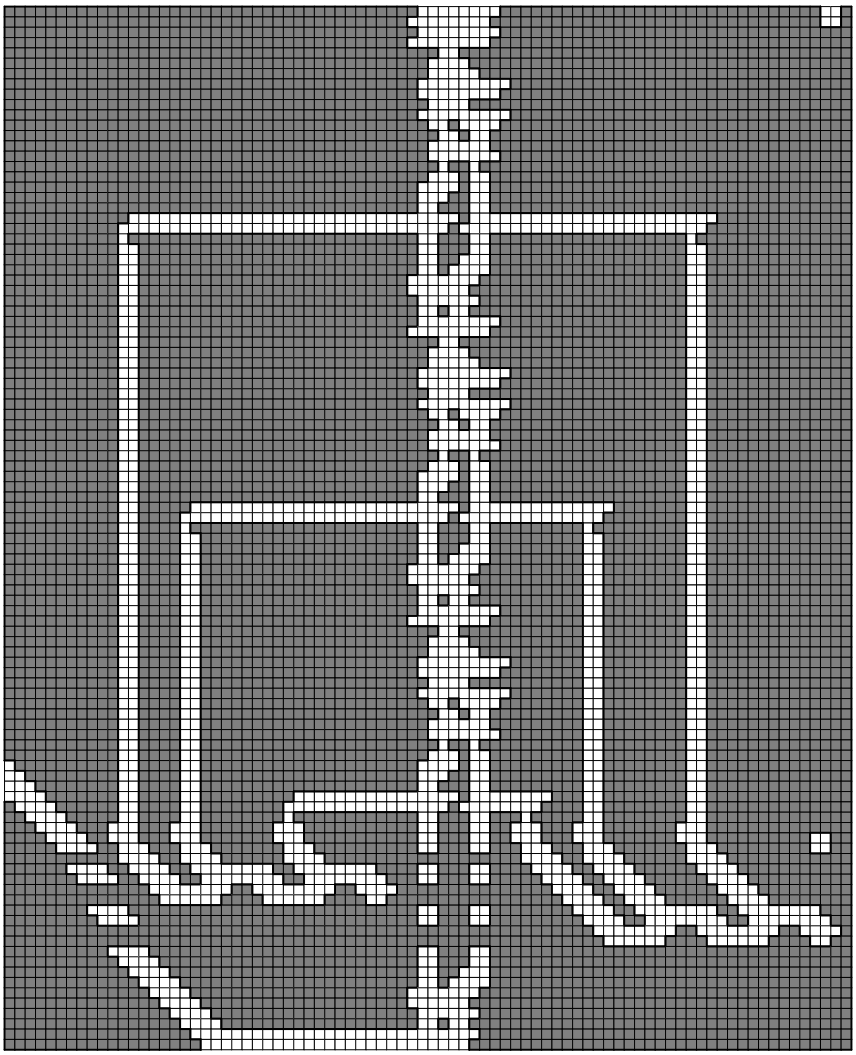


图 41： -tris约的一般结构。

2. 如图 82 所示，以三种方式之一填充所有子句构件（由于解保证所有子句构件均可访问，因此这是可行的），从而消除一些行，并使结构的底部区域变得可访问。

3. 填满可攀爬垂直结构，因为现在已经可以进入该结构的底部区域。这使得包含变量构件的行能够被清除。4. 从最顶部的变量开始，填满并清除变量构件，以及连接到子句构件的相关连线，如图 81 所示（现在之所以可行，是因为包含变量构件的行可以被清除）。5. 清除最右侧几列中的 8 个方格，以及子句构件中剩余的空方格（如有必要）。

该方法保证，只要平面单调直线 SAT 实例存在一个有效解，我们就能消除整个构造。因此，如果平面单调直线 SAT 实例有解，那么在 SRS 规则下，就能用方块消除整个构造。

由于给定一个平面单调直线 SAT 实例后，可以在多项式时间内构造出这一构造；并且在 SRS 规则下，整个构造能够被方块消除，当且仅当平面单调直线 SAT 实例有解，因此我们证明了，即使玩家只能获得方块，使用 SRS 的俄罗斯方块消行问题仍是 NP难，正如所需。

利用对称性，即使玩家只获得方块，带 SRS 的俄罗斯方块消行问题仍然是 NP难的。

9 生存

定理 9.1。对于任意方块类型 $P \in \{$, , , $\}$，带 SRS 的俄罗斯方块生存问题是 *NP* 难的，即使限制玩家所获序列中的方块类型仅为 P。

我们先讨论 -tris 情形。证明的大部分内容与带 SRS 且仅使用方块的俄罗斯方块消行证明相同；不过，我们主要修改了构造的最右两列，如图 42 所示。这里，我们将最右两列设置为棋盘状图案，以防止任何方块被放置在最右两列中（否则玩家就会输），并防止任何行被消除。此外，与俄罗斯方块消行构造相比，我们少给玩家一个方块，因此玩家可以填满整个棋盘，只留下最右两列中的空方格；当且仅当相应的 1-in-3SAT 实例有解时，玩家才能存活。通过这一修改，我们证明了带 SRS 的俄罗斯方块生存问题是 NP 困难的，即使玩家只获得方块。

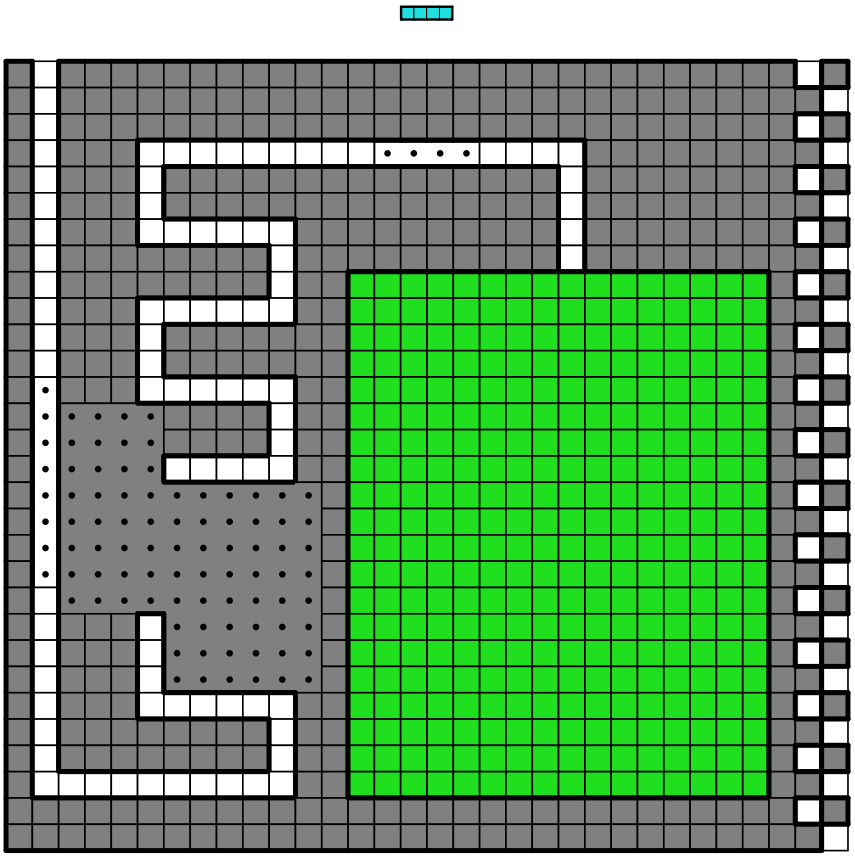


图 42：仅使用方块的俄罗斯方块生存构造的总体结构。构造的主体部分以绿色标出。

采用类似的方法（修改最右两列），即使玩家只获得类型为 P 的拼块 $P \in \{$, , $\}$，也能证明使用 SRS 的俄罗斯方块生存是 NP 困难的。

对于 ，除了修改最右两列外，我们还需要进行其他修改，因为整体结构依赖消行来进入结构的所有部分，而在归约到俄罗斯方块生存时，我们希望避免这种情况。在这里，我们移除可攀爬垂直结构中的拼块，并按照图43所示修改每个子句构件，使每个子句构件都连接到一个由空方格构成的 $(2C_1 + 2) \times (2C_2 + 2)$ 矩形，其中 $C_1 = \Theta(C_2)$，且 C_2 足够大，使得不属于这些矩形的空方格数量小于 C_1C_2（这样的 C_1、C_2 确实存在，因为这类空方格的数量只会随 C_1+C_2 线性增长）；并在必要时延长任何“连线”（即任何不属于变量构件、子句构件或可攀爬垂直结构的长度或宽度为2的走廊），确保这些矩形彼此之间以及与任何现有构件都不相交。对于方块序列，我们在输入序列中给玩家 mC_1C_2 个拼块（其中 m 是平面单调直线 SAT 实例中的子句数量）。

关于正确性，请注意，由于不属于任何附接在子句构件上的矩形的空方格数量少于 C_1C_2，玩家要想在不失败的情况下放入 mC_1C_2 个方块，唯一的办法就是所有附接在子句构件上的 $m(2C_1+2) \times (2C_2+2)$ 矩形都至少包含 1 个方块，这意味着所有 m 个矩形，从而所有 m 个子句构件，都必须可达。此外，如果所有 m 个子句构件都可达，那么每个 $(2C_1 + 2) \times (2C_2 + 2)$ 矩形都能轻松容纳 C_1C_2 个方块，这意味着所有 mC_1C_2 个方块都可以在不失败的情况下放置。因此，即使玩家只获得方块，我们也能证明采用 SRS 的俄罗斯方块生存问题是 NP 困难的。利用对称性，我们也能证明该问题是 NP 困难的。

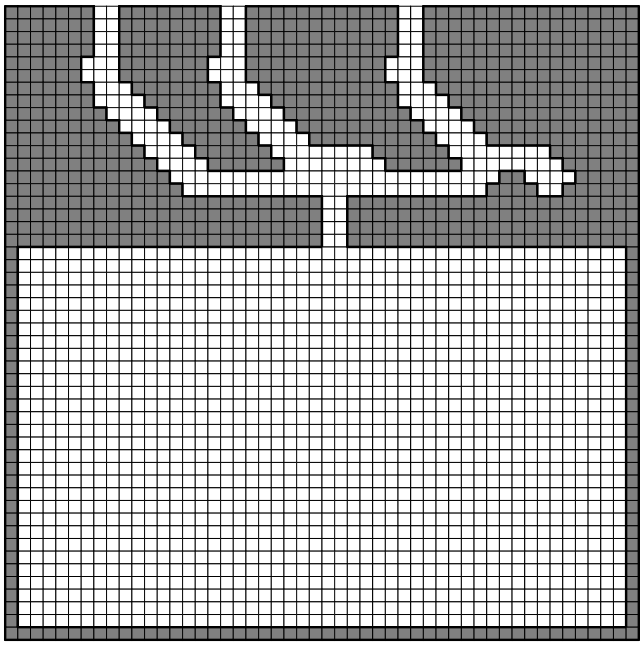


图43： 仅使用方块时用于俄罗斯方块生存的（修改后的）子句构件

10 个开放问题

[DDE+17] 和 [MDHL24] 中的一些开放问题仍未解决。对于可旋转多米诺骨牌，一个尚未解决的问题是：如果我们假设采用不同的旋转模型，例如适用于多米诺骨牌的 SRS 版本，那么俄罗斯方块消行和生存是否仍可在多项式时间内求解。在我们针对可旋转多米诺骨牌的俄罗斯方块消行和生存问题所设计的多项式时间算法中，我们假设旋转中心

只会单调向下移动。虽然可以将该算法调整为适用于允许旋转中心向上移动的不同旋转系统，但具体细节会棘手得多，而且该问题也可能是 NP难。

对于四格骨牌类型，最受关注的开放问题是方块的俄罗斯方块消行和生存的计算复杂性。对于消行问题，虽然方块无法旋转，这简化了问题（因为方块会参与旋转），也让我们相信它很可能属于 P，但由于消行机制的存在，要证明这一点仍不简单。对于生存问题，我们有证据表明它是 NP难的，因为 2×2 正交凸网格多边形中的方格装箱是 NP难的 [AS26], 不过，还有一些棘手的细节需要解决，尤其是要确保所有方块都能按照俄罗斯方块规则成功放置。

另一个开放问题是，使用 SRS 且只有一种方块类型时，俄罗斯方块消行和生存是否属于 #P难或 ASP 完全。[1] 具体来说，我们的构造不太可能是 c-monious[2]，因为每种构件类型的填充方式数量可能因铺砌方式不同而不同。是否存在另一种构造，能对某个 c而言满足 c-monious?

最后，我们的大多数结果都分析了有限输入序列下的俄罗斯方块消行和生存。如果玩家获得的不是有限序列，而是无限多个给定类型的方块，俄罗斯方块消行和生存的复杂性会如何变化?

致谢

本文起源于 2023 年秋季麻省理工学院开设的《算法下界：与困难性证明的乐趣》（6.5440）课程中的开放问题探讨，该课程由 Erik Demaine 授课。我们感谢该课程的其他参与者——特别是 Eugeniya Artemova、Cecilia Chen、Lily Chung、Jenny Diomidova、Holden Hall、Jonathan Li、Jayson Lynch、Frederick Stock 和 Ethan Zhou——感谢他们参与有益的讨论，并营造了鼓舞人心的氛围。

参考资料

[ACD+20] Sualeh Asif、Michael Coulombe、Erik D. Demaine、Martin L. Demaine、Adam Hesterberg、Jayson Lynch 和 Mihir Singhal。即使只有 $O(1)$行或列，俄罗斯方块仍然是 NP难问题。《信息处理学报》，28:942–958，2020年。[AS26] Mikkel Abrahamsen 和 Jack Stade。用单位正方形装箱、覆盖和划分简单多边形的困难性。《*SIAM*计算杂志》，FOCS24:29–84，2026年。[BDH+04] Ron Breukelaar、Erik D. Demaine、Susan Hohenberger、Hendrik Jan Hoogeboom、Walter A. Kosters 和 David Liben-Nowell。俄罗斯方块很难，甚至难以近似。《国际计算几何与应用杂志》，14(1–2):41–68，2004年。会议版本发表于 COCOON 2003。[BK98] Therese Biedl 和 Goos Kant。一种改进的正交图绘制启发式算法。《计算几何：理论与应用》，9(3):159–180，1998年。[dBK10] Mark de Berg 和 Amirali Khosravi。平面上的最优二元空间划分。载于第*16*届年度计算与组合国际会议（*COCOON 2010*）论文集，第216–225页，越南芽庄，2010年7月。施普林格。

[1]回想一下，NP 搜索问题是***ASP*** 完全 [YS03]，如果所有 NP 搜索问题都能归约到它（包括解之间存在多项式时间双射）[2]若某种归约 ***c*-解数保持**，则它恰好将解的数量扩大 c倍。

[DDE+17] Erik D. Demaine、Martin L. Demaine、Sarah Eisenstat、Adam Hesterberg、Andrea Lincoln、Jayson Lynch 和 Y. William Yu。Total Tetris：使用单格骨牌、多米诺骨牌、三格骨牌、五格骨牌……的俄罗斯方块。《信息处理学报》，25:515–527，2017年。[DTW] Hard Drop 俄罗斯方块维基。永远游玩。https://harddrop.com/wiki/Playing forever。访问日期：2026年1月8日。[GV24] Matthias Gehnen 和 Luca Venier。俄罗斯方块并非竞争性的。载于 Andrei Z. Broder 和 Tami Tamir 编，《第十二届算法趣味国际会议（FUN 2024）论文集》，LIPIcs 第291卷，第16:1–16:16页，意大利拉马达莱纳，2024年。[HIN+17] Takashi Horiyama、Takehiro Ito、Keita Nakatsuka、Akira Suzuki 和 Ryuhei Uehara。用三格骨牌或条形骨牌铺砌多边形的复杂性。《离散与计算几何》，58:686–704，2017年。[Kis11] Shaunak Kishore。ntris，2011年。https://erikdemaine.org/ntris/。[MAD+26] MIT Hardness Group、Zachary Abel、Erik D. Demaine、Jenny Diomidova、Jeffery Li 和 Zixiang Zhou。平面图定向框架及其在 KPlumber 和多格骨牌铺砖中的应用。arXiv:SUBMITTED，2026年。[McM24] Liv McMahon。青少年声称首次实现俄罗斯方块“重生”。英国广播公司，2024年10月8日。https://www.bbc.com/news/articles/c70wnrg678lo。[MDHL24] MIT Hardness Group、Erik D. Demaine、Holden Hall 和 Jeffery Li。少种类方块俄罗斯方块。载于 Andrei Z. Broder 和 Tami Tamir 编，《第十二届算法趣味国际会议（FUN 2024）论文集》，LIPIcs 第291卷，第24:1–24:18页，意大利拉马达莱纳，2024年。[Sch78] Thomas J Schaefer。可满足性问题的复杂性。载于第十届年度ACM计算理论研讨会（STOC 1978）论文集，第216–226页，1978年。[Teta] TetrisWiki。Combinos。https://tetris.wiki/Combinos。访问日期：2026年1月8日。[Tetb] TetrisWiki。粉丝游戏列表。https://tetris.wiki/List of fan games。访问日期：2026年1月8日。[Tetc] TetrisWiki。超级旋转系统。https://tetris.wiki/Super Rotation System。访问日期：2026年1月8日。[Tetd] TetrisWiki。俄罗斯方块指南。https://tetris.wiki/Tetris Guideline。访问日期：2026年1月8日。[Tet23] 俄罗斯方块，2023年。Apple TV+ 电影，由 Jon S. Baird 执导，Noah Pink 编剧。[Wik] 维基百科。俄罗斯方块。https://en.wikipedia.org/wiki/Tetris。访问日期：2026年1月8日。[YS03] Takayuki Yato 和 Takahiro Seta。寻找另一解的复杂性与完备性及其在谜题中的应用。《电子、通信与计算机科学基础IEICE汇刊》，E86-A(5):1052–1060，2003年。另见《信息处理学会特别兴趣组研究报告》2002-AL-87-2，2002年。

拼块旋转图

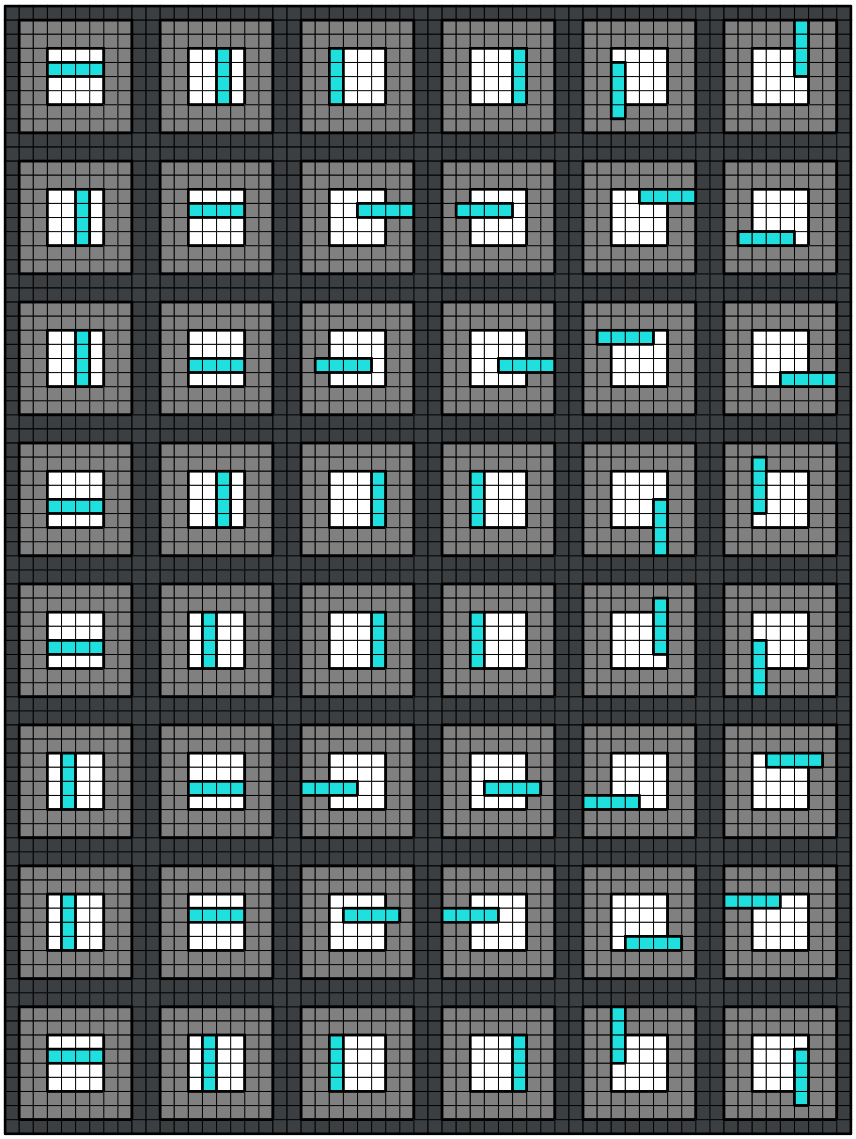


图44： 拼块所有旋转的测试。从上到下，旋转依次为：$0 \to R$, $R \to 0$, $R \to 2$,$2 \to R$, $2 \to L$, $L \to 2$, $L \to 0$, $0 \to L$。0 表示默认方向，而 R、2 和 L 分别表示从默认方向顺时针旋转 90°、180° 和 270° 后达到的朝向。

B 特定 -tris 操作

B.1 操作1

图45所示的这一操作用于让方块穿过长隧道，进入构造的主体部分（经由 EC 组件）。这一操作也大致说明了方块如何“转过拐角”。在 (a) 中，拼块可以处于任一水平朝向（默认方向或旋转 180° 后的朝向）；在 (h) 中也可以处于任一水平朝向，不过拼块在 (a) 中的朝向会影响它在 (h) 中的朝向。

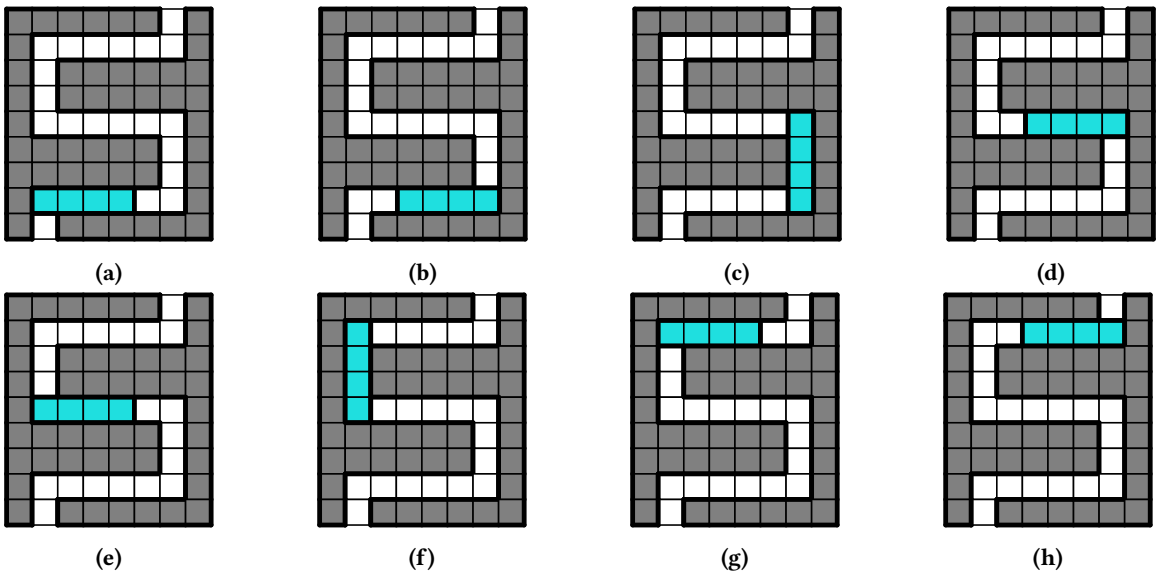


图45： 操作　第1步。这是可逆的；也就是说，从 (a) 到 (h) 和从 (h) 到 (a) 都是可能的　e.

B.2 操作2

图46所示的这一操作，或关于垂直线镜像对称的类似操作，可用于在将片件48或片件60放入复制器构[illegible]时，拼块处于错误朝向（即非默认水平朝向）的情况。

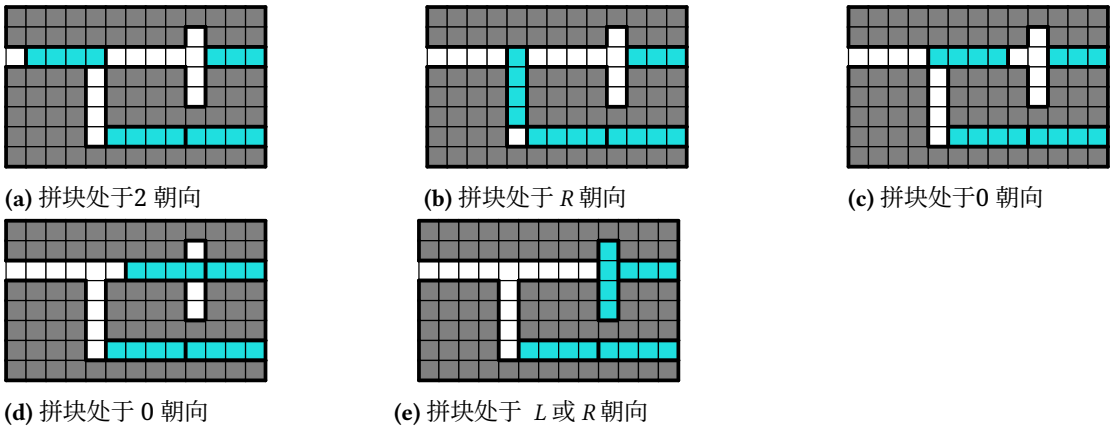
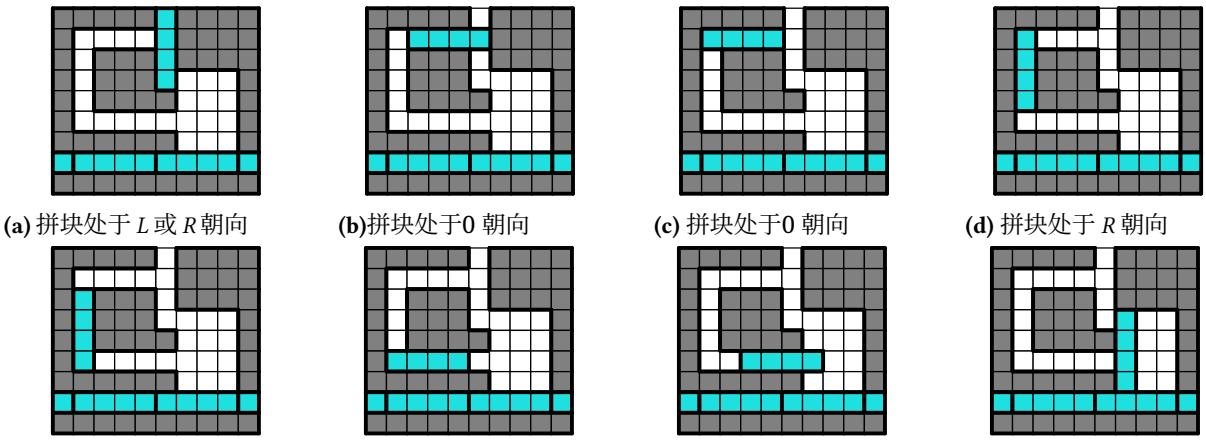


图46： 操作2。图注标明了拼块的朝向。0 表示默认方向，而 R、2和 L 分别表示从默认方向顺时针旋转 90°、 180° 和 270° 后达到的朝向。

B.3 机动3

图 47 所示的这一操作用于将方块移至否定子句构件的第一种和第二种铺砌中编号为15的拼块所在的位置。



(a) 拼块处于 L 或 R 朝向 **(b)**拼块处于0 朝向 **(c)** 拼块处于0 朝向 **(d)** 拼块处于 R 朝向

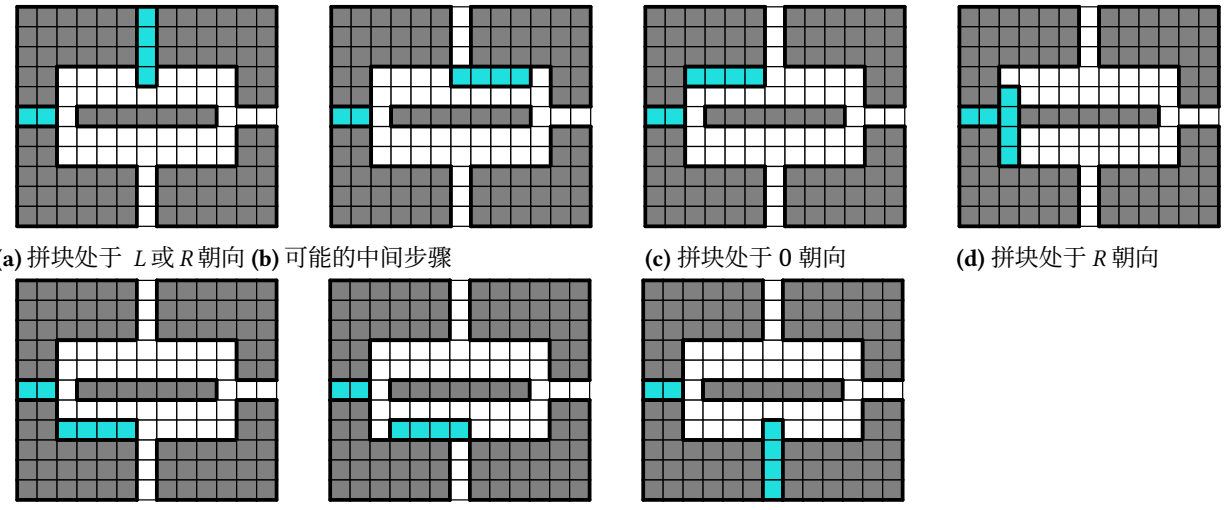
(e) 拼块处于 R 朝向 **(f)** 拼块处于2 朝向 **(g)** 拼块处于 2 朝向 **(h)** 拼块处于 L 或 R 朝向

图 47： 机动3。图注标明拼块的朝向。0 表示默认方向，而 R、[illegible] L 分别表示从默认方向顺时针旋转 90°、180° 和 270° 后到达的朝向。

B.4 操作 4

图 48 所示的这一操作及类似操作用于使方块通过 acrossovergadget 的中[illegible]分（尤其是腔室）。

图中展示了进入中间腔室的操作，不过步骤 (e) 要求拼块向下移动一次。根据腔室中是否已有其他方块，步骤 ([illegible] (d) 可能会受到更多限制（要么只能选择 $R \to 0 \to R$，要么只能选择 $L \to 2 \to L$），但无论步骤 (d) 中的朝向是（L 还是 R），都可以让拼块到达 (e) 中所示[illegible]置。其他腔室的步骤也类似。



(a) 拼块处于 L 或 R 朝向 **(b)** 可能的中间步骤 **(c)** 拼块处于 0 朝向 **(d)** 拼块处于 R 朝向

(e) 拼块处于2 朝向 **(f)** 拼块处于2 朝向 **(g)** 拼块处于 R 朝向

图 48： 机动4。图注标明拼块的朝向。0 表示默认方向，而 R、[illegible] L 分别表示从默认方向顺时针旋转 90°、180° 和 270° 后达到的朝向。

C 特定 -tris 操作

C.1 操作1

此操作如图49所示，用于向上移动 方块，使其穿过可攀爬垂直结构。

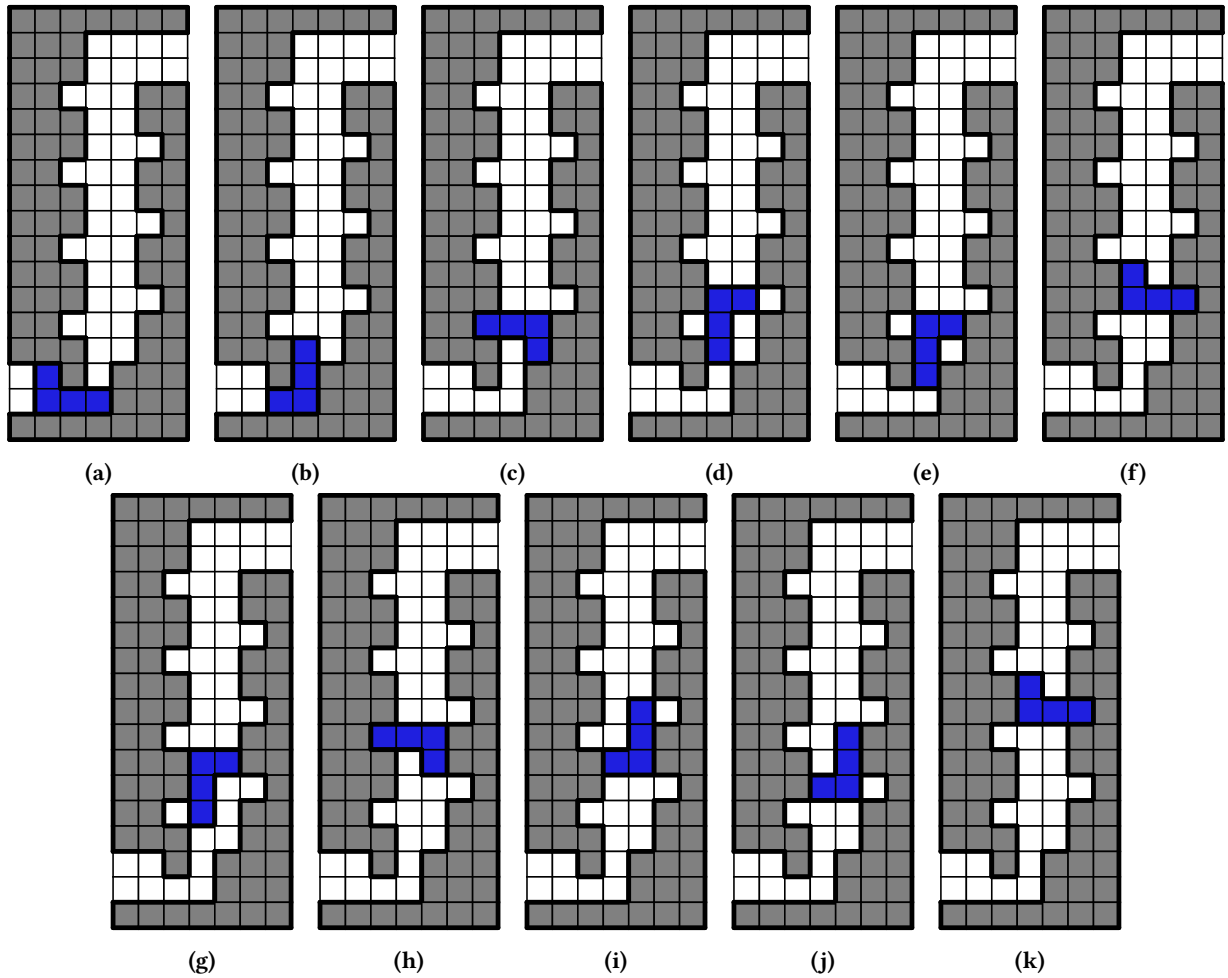


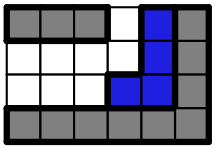
图49： 操作1。(a)至(f)用于将拼块移入可攀爬垂直结构， (f)至(k)用于沿可攀爬垂直结构向上攀爬一个区段。

C.2 操作2

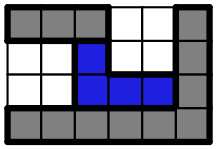
图50所示的这一操作用于在某些构件中转过拐角。

C.3 机动3

图51所示的这一操作用于将部分方块塞入奇偶校正构件和入口角部装置中的指定位置。

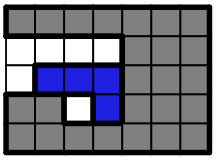


(a)

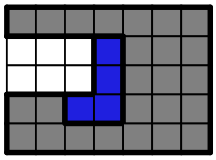


(b)

图50：操作2



(a)

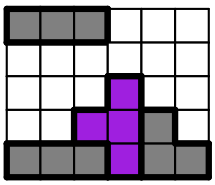


(b)

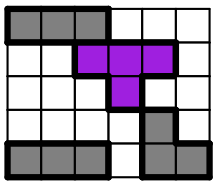
图51：机动3

D 特定 -三角形操作

我们利用这些操作将方块向上推，或将方块塞入构件或可攀爬垂直结构中的特定位置。这些操作具有对称性；我们展示了从默认方向旋转至经 90° 逆时针旋转后所达朝向，以及从该朝向旋转回默认方向的操作。对于从默认方向经 90° 顺时针旋转后所达的朝向，也存在类似的操作。

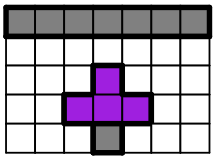


(a)

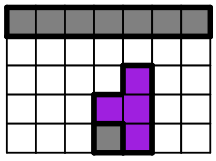


(b)

图52：操作1



(a)



(b)

图53：操作2



(a) (b)

图54：机动3

E 特定 -tris 操作

点标示的方格是拼块的中心。

E.1 操作1

图 55 所示的这一操作用于使方块向上穿过子句构件以创建可攀爬的垂直结构。

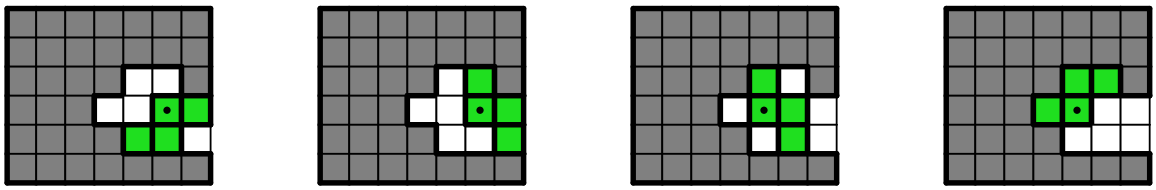


(a) (b)

图 55：操作1

E.2 操作2

图 56 所示的这一操作用于将部分方块放入子句构件和可攀爬垂直结构中的相应位置。



(a) (b) (c) (d)

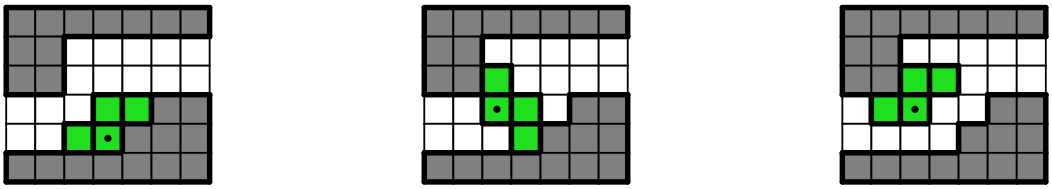
图 56：操作2

E.3 机动3

图57 所示的这一操作用于在子句构件中将方块从左向右移动。

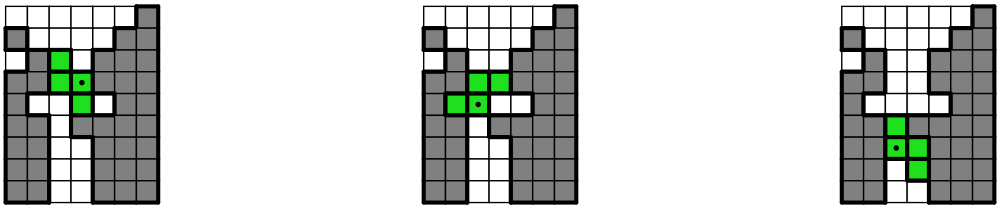
E.4 机动4

图58所示的这一操作用于将方块穿过变量装置的“False”隧道。



(a) (b) (c)

图57：机动3

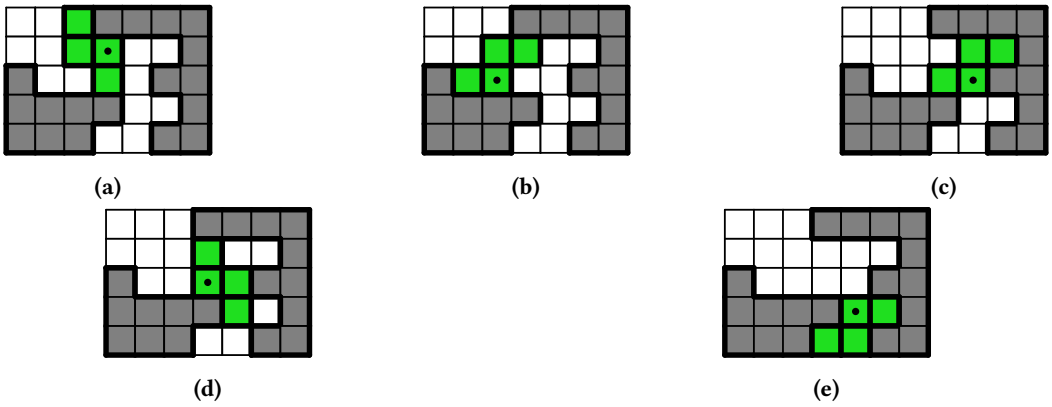


(a) (b) (c)

图58：机动4

E.5 操作5

图59所示的这一操作用于将方块9放入图 81(b)中



(a) (b) (c)

(d) (e)

图59：操作5

E.6 操作6

图60所示的这一操作，以及该操作的一种变体（将拼块的初始位置向左移动 1 格），用于让方块穿过变

构件中的某些间隙。



(a) (b)

图60：操作6

F-三角形构件铺砌

方块上的数字表示放置顺序；数字较小的先填入（因此，1 表示铺砌中最先放置的拼块）。

G-三角形构件铺砌/填充

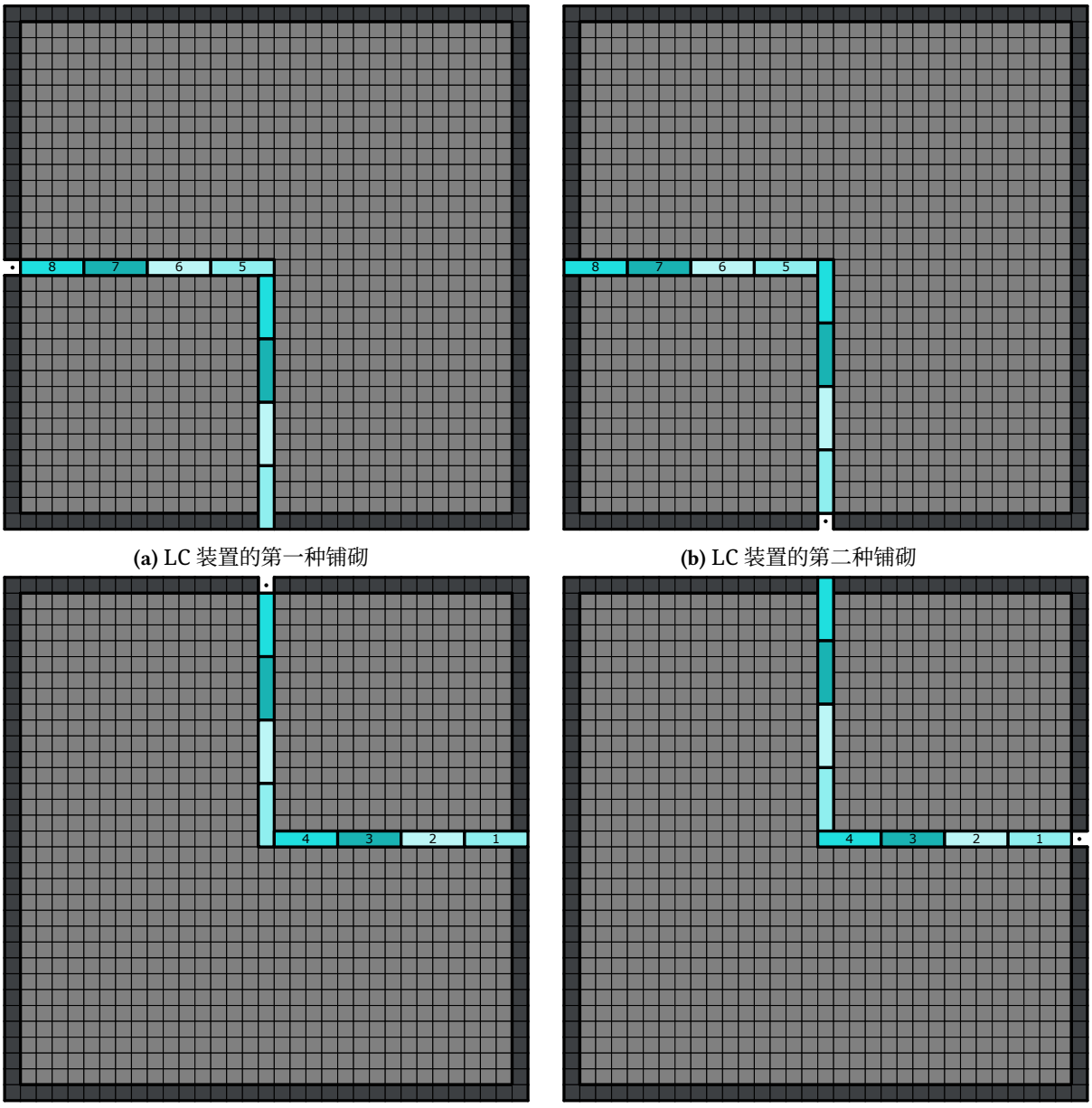
与三角形中一样，方块上的数字表示放置顺序，数字较小的先放置。

H-三角形构件铺砌/填充

与.....中一样 -三角形中，方块上的数字表示放置顺序，数字较小的先放。

I -三角形构件铺砌/填充

与 -三角形中一样，方块上的数字表示放置顺序，数字较小的先放。



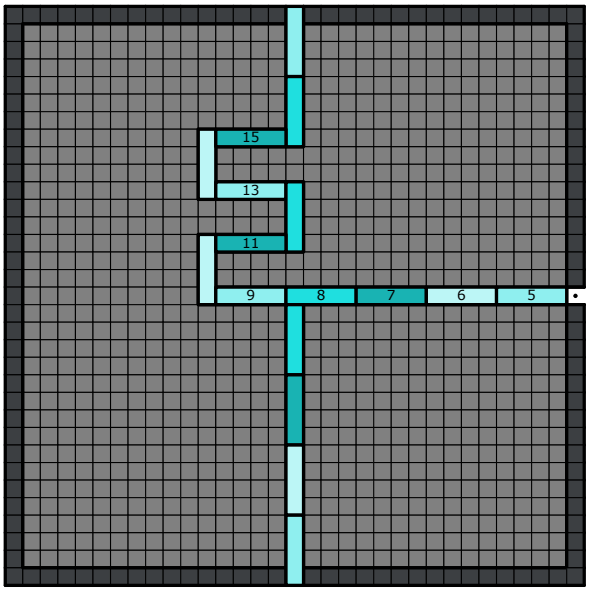
(a) LC 装置的第一种铺砌

(b) LC 装置的第二种铺砌

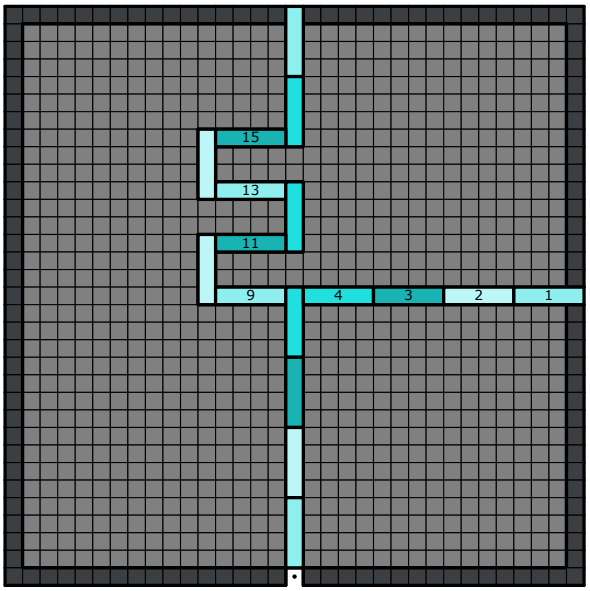
(c)UC 装置的第一种铺砌

(d) UC 装置的第二种铺砌

图61： 角构件的铺砌 + 放置顺序

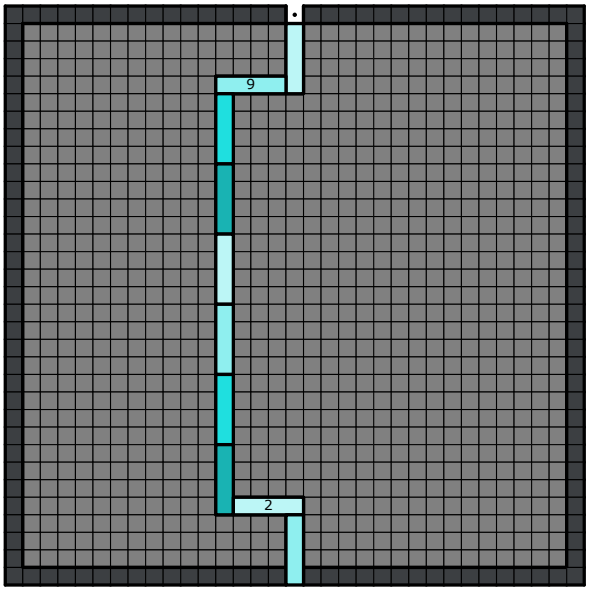


(a) 第一种铺砌

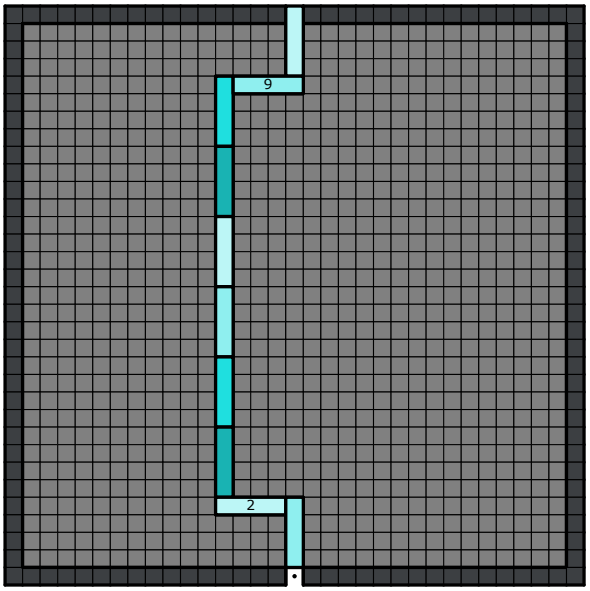


(b) 第二种铺砌

图62： 入口角部装置的铺砌 + 放置顺序

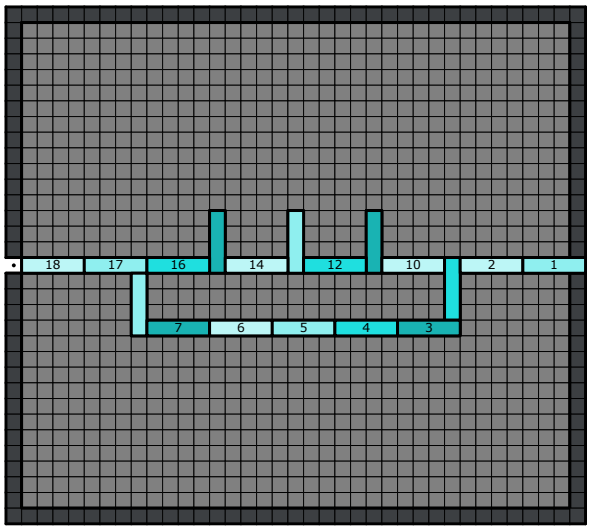


(a) 第一种铺砌

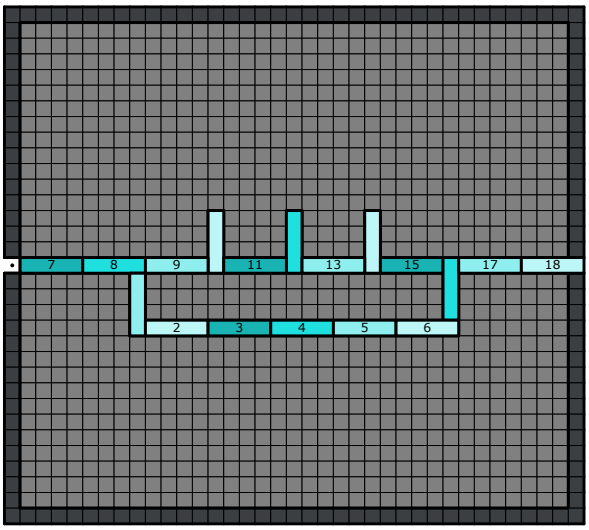


(b) 第二种铺砌

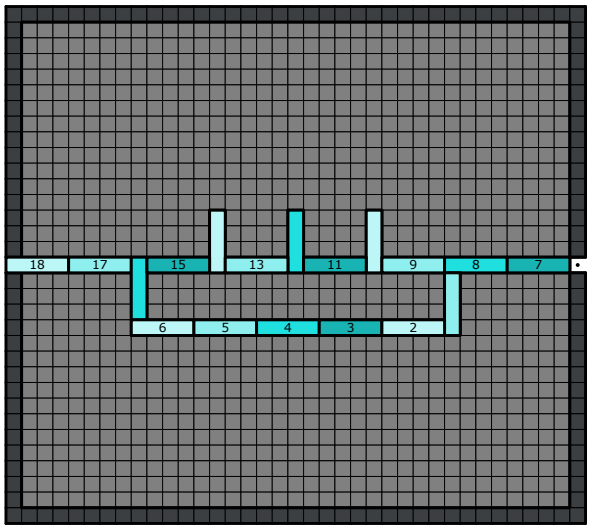
图63： 垂直线装置的铺砌 + 放置顺序



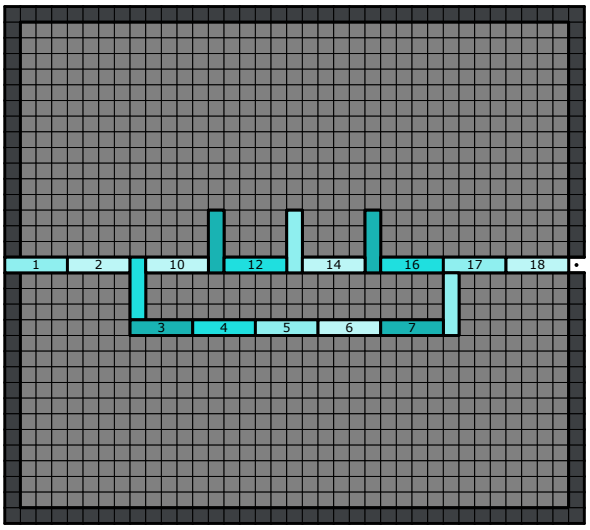
(a) 第一种铺砌，左端口



(b) 第一种铺砌，右端口

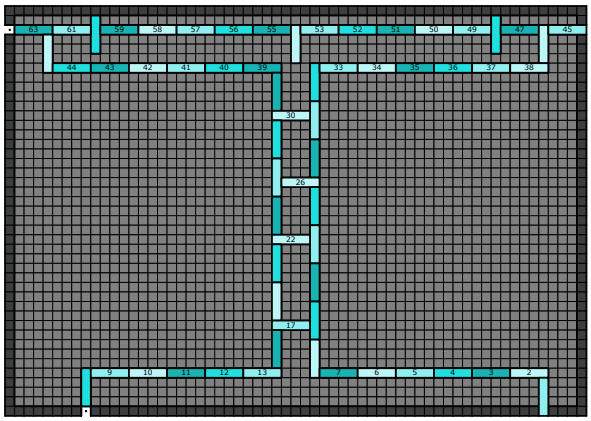


(c)第二种铺砌，左端口

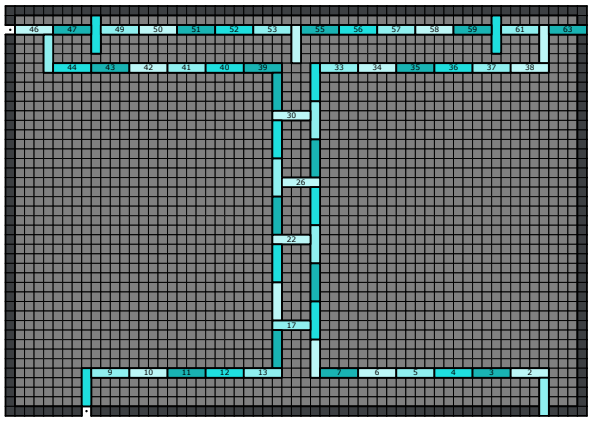


(d) 第二种铺砌，右端口

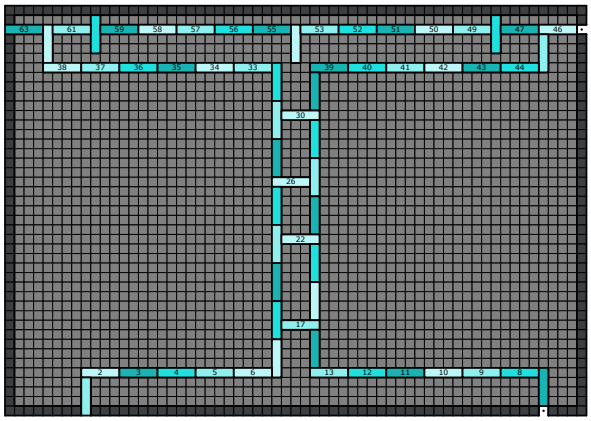
图64：HL 小工具的铺砌 + 放置顺序



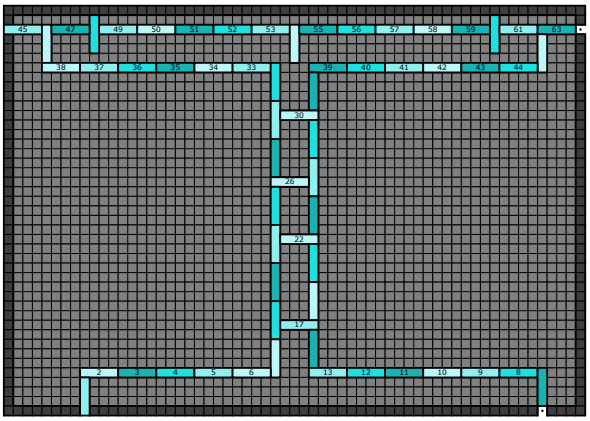
(a) 第一种铺砌，左端口



(b) 第一种铺砌，右端口

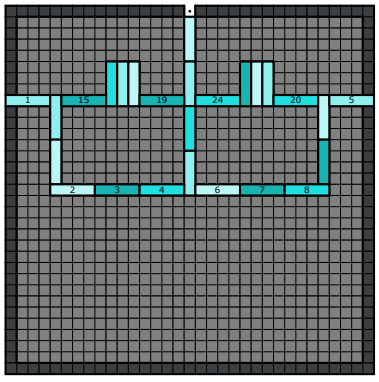


(c)第二种铺砌，左端口

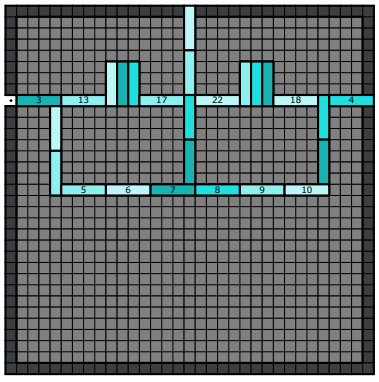


(d) 第二种铺砌，右端口

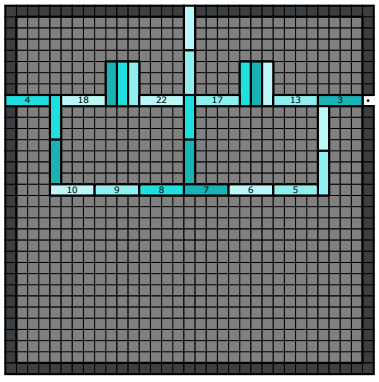
图65： 复制器构件的铺砌 + 放置顺序



(a) 第一种铺砌

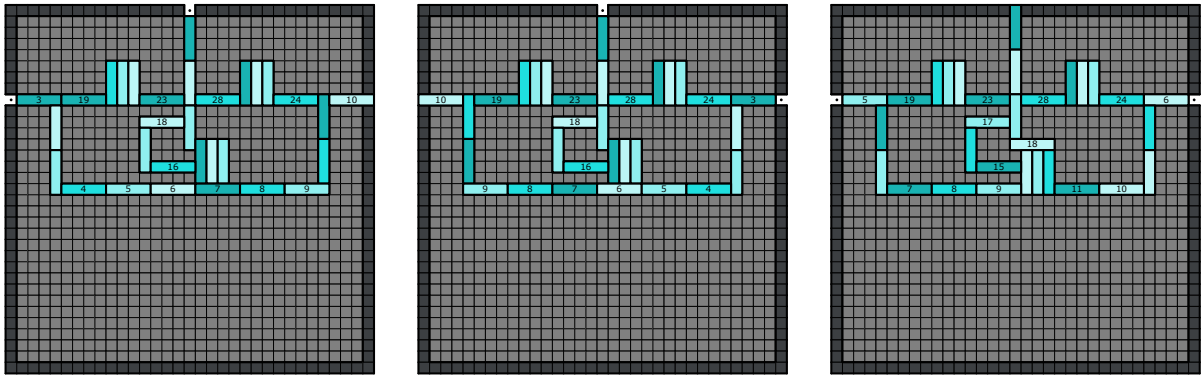


(b) 第二种铺砌



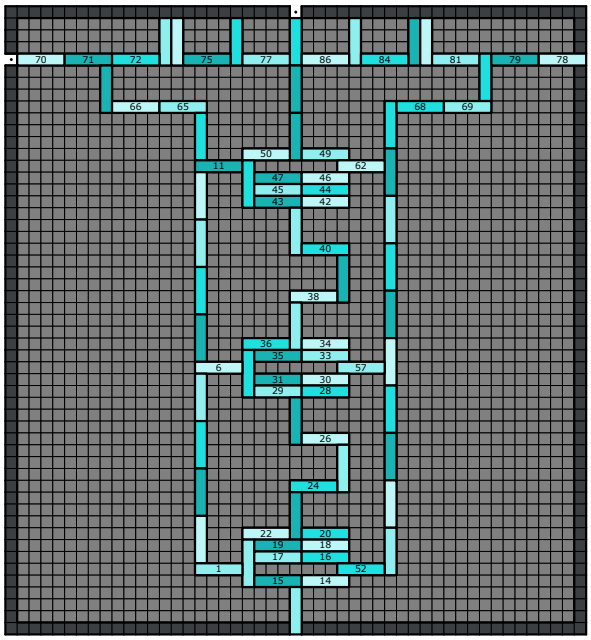
(c) 第三种铺砌

图 66： 铺砌 + 子句构件的放置顺序

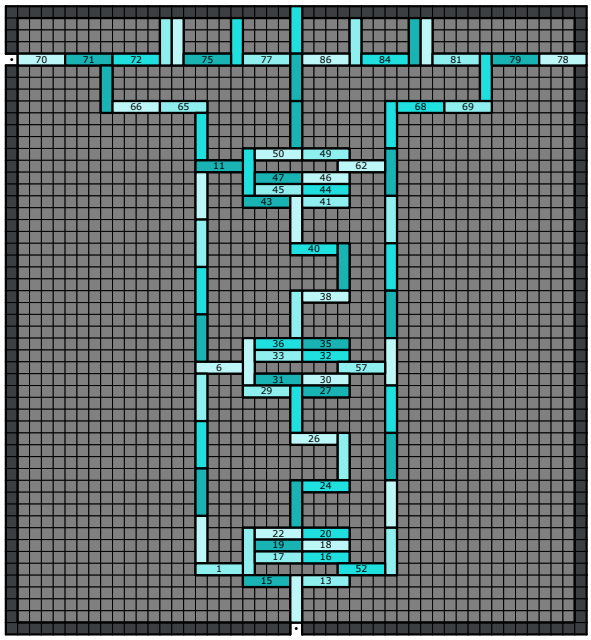


(a) 第一种铺砌 **(b)** 第二种铺砌 **(c)** 第三种铺砌

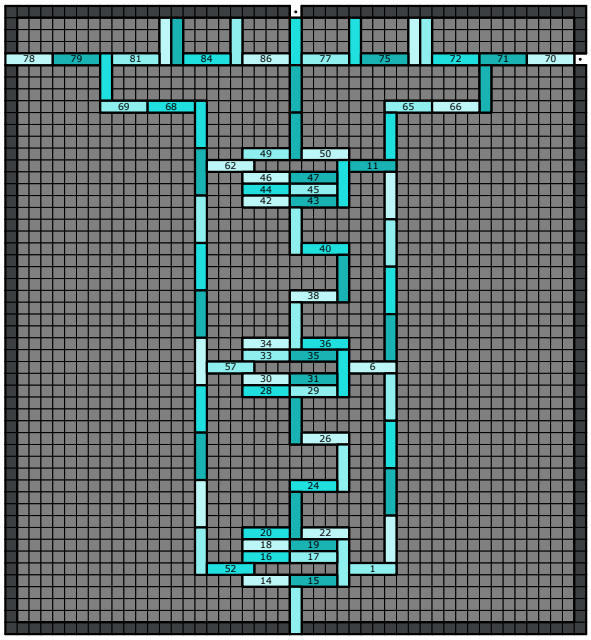
图 67： 否定子句构件的铺砌 + 放置顺序



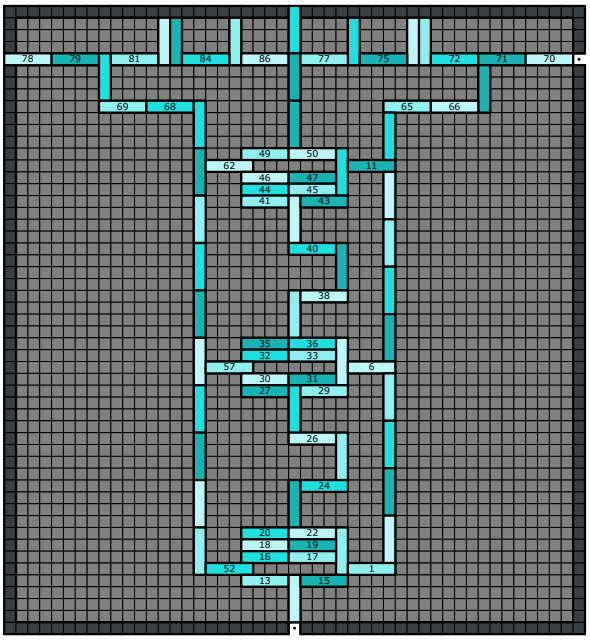
(a) 第一种铺砌



(b) 第二种铺砌

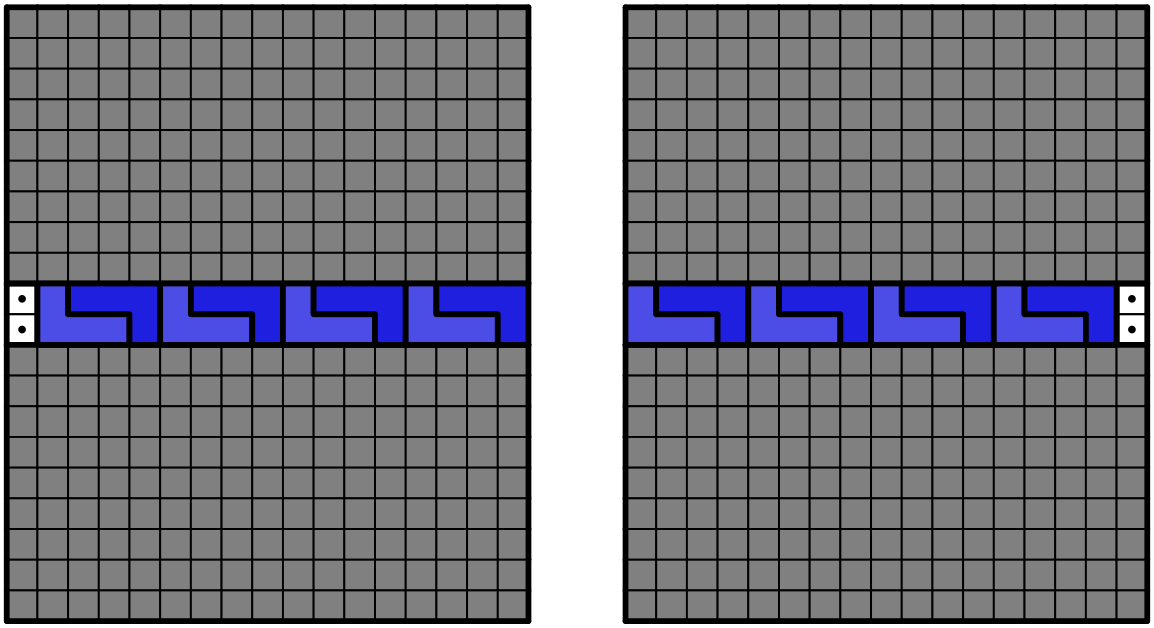


(c)第三种铺砌



(d) 第四种铺砌

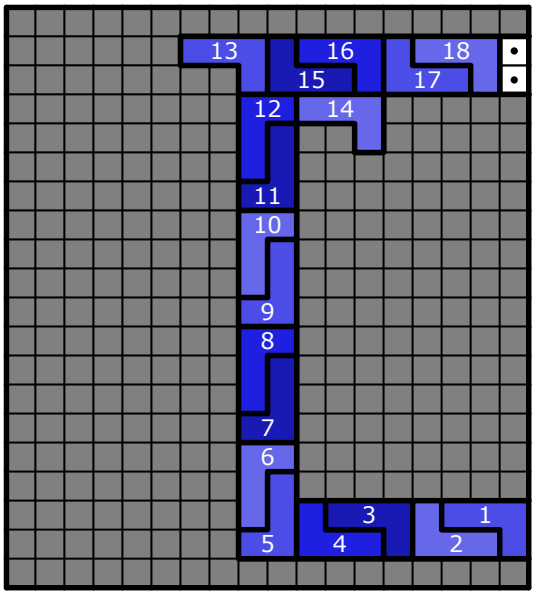
图68：交叉装置的铺砌 + 放置顺序



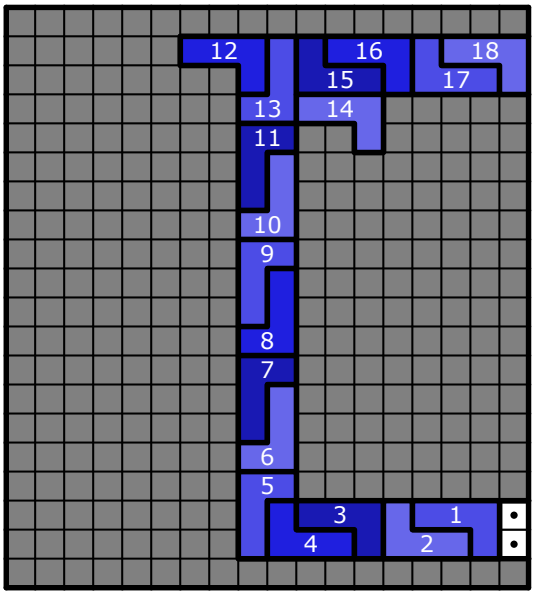
(a) 第一种铺砌

(b) 第二种铺砌

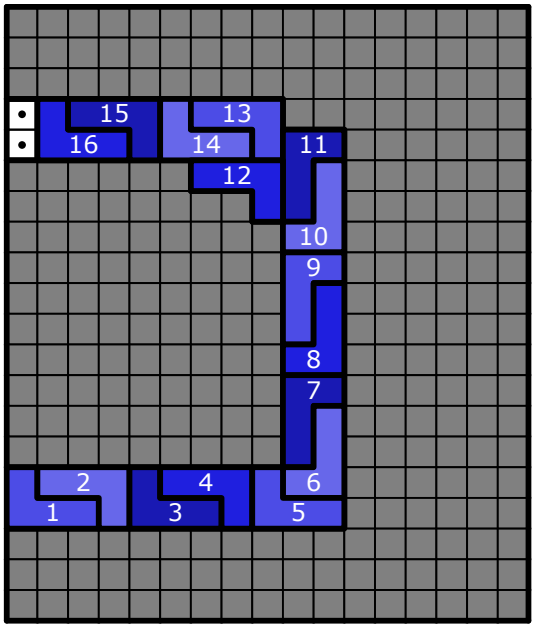
图 69：HL 小工具的铺砌



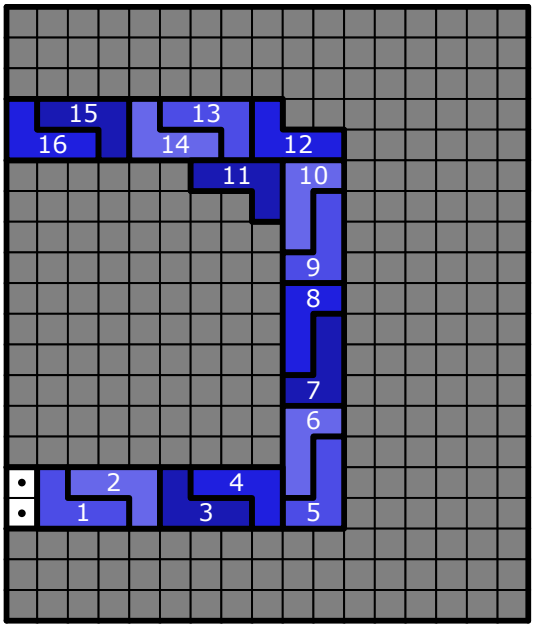
(a) 左上转弯装置的第一种铺砌



(b) 左上转弯装置的第二种铺砌

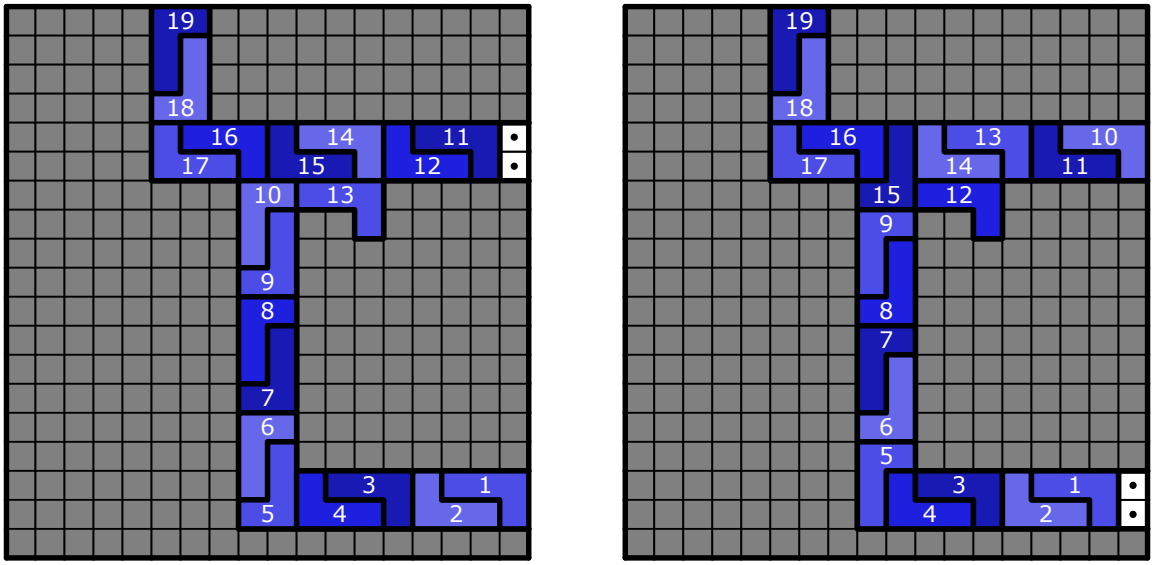


(c)右上转弯装置的第一种铺砌



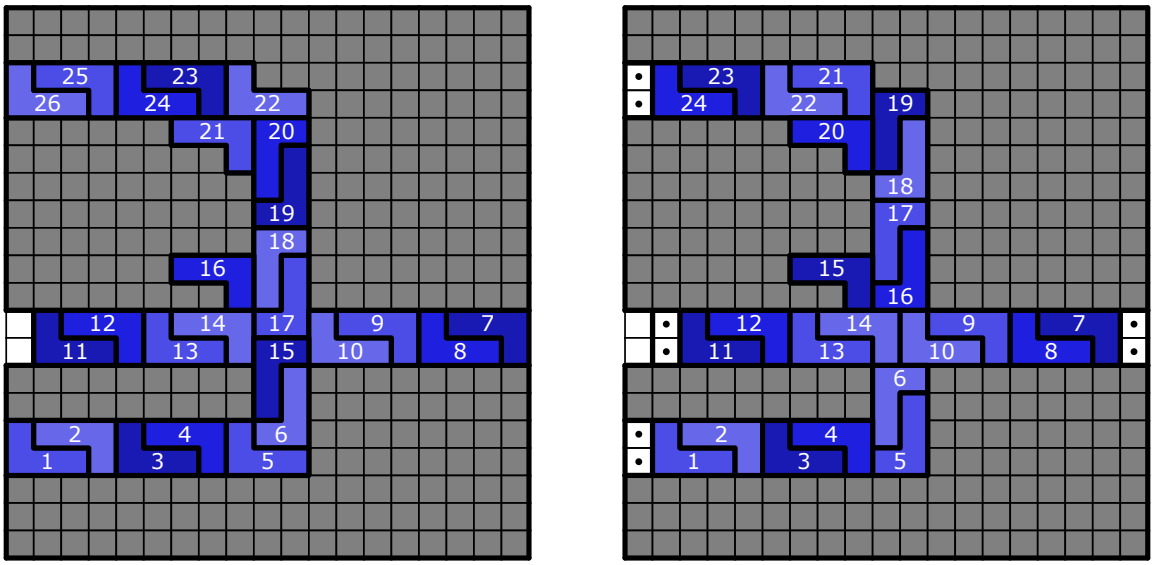
(d)右上转弯装置的第二种铺砌

图70： U形转弯构件的铺砌 + 放置顺序



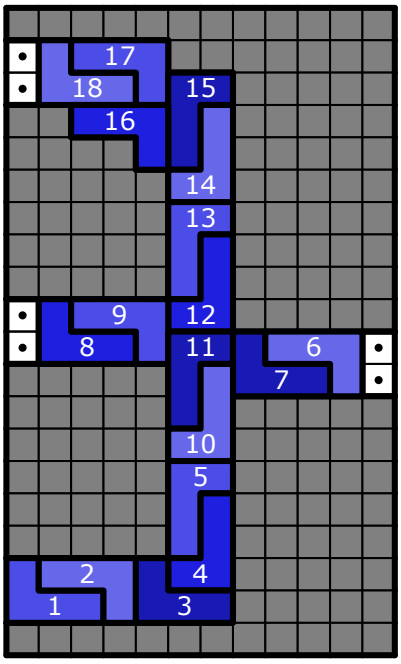
(a) 第一种铺砌　　(b) 第二种铺砌

图 71： 铺砌 + EC组件的放置顺序

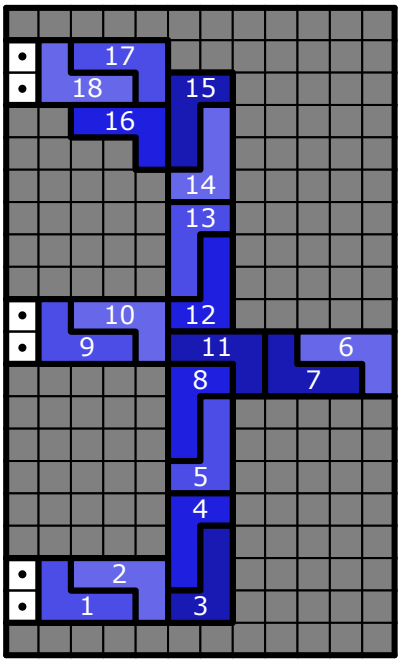


(a) 第一种铺砌　　(b) 第二种铺砌

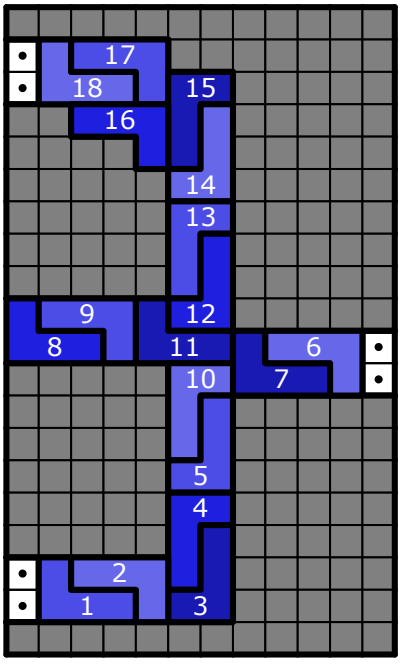
图 72： 铺砌 + 0或4构件的放置顺序



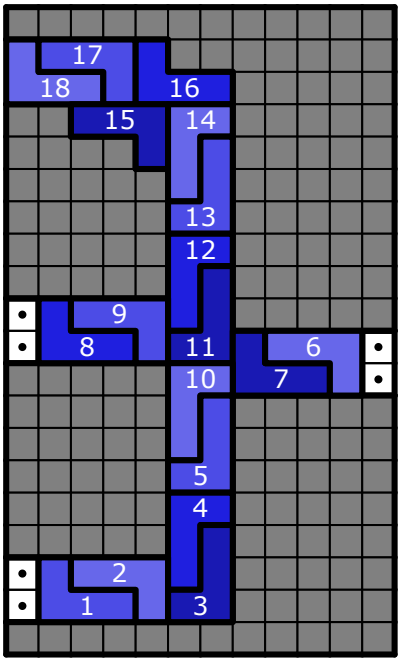
(a) 第一种铺砌



(b) 第二种铺砌

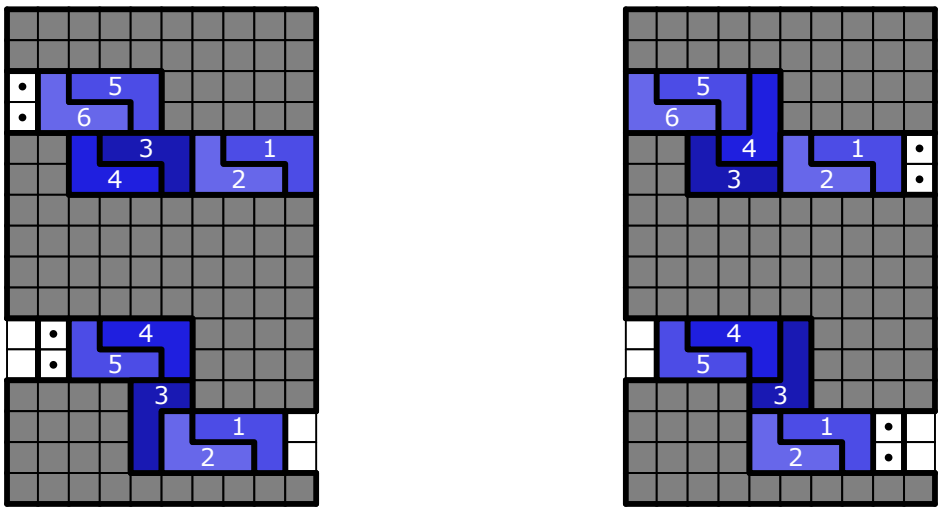


(c)第三种铺砌



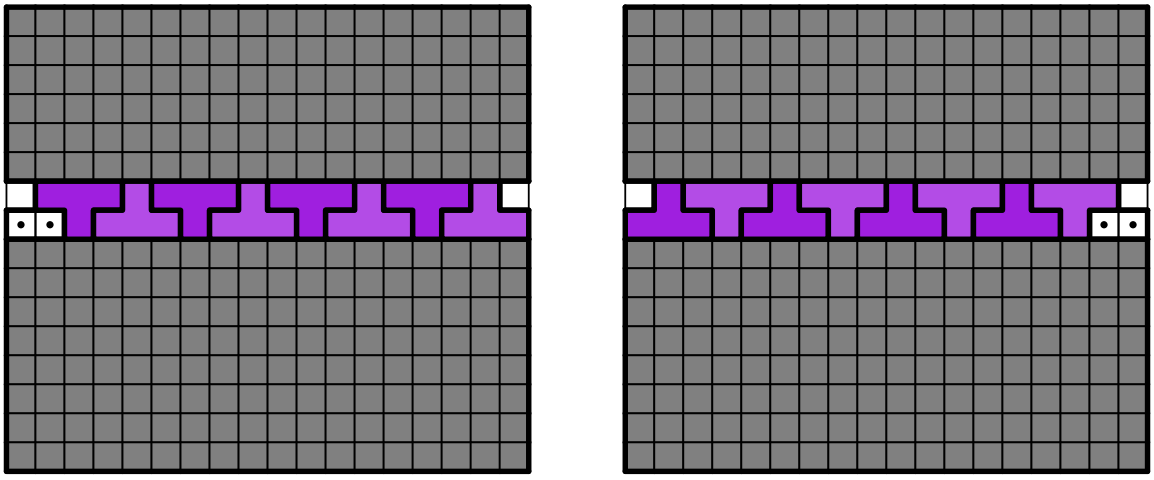
(d) 第四种铺砌

图73： 铺砌 + 3 入 4 构件的放置顺序



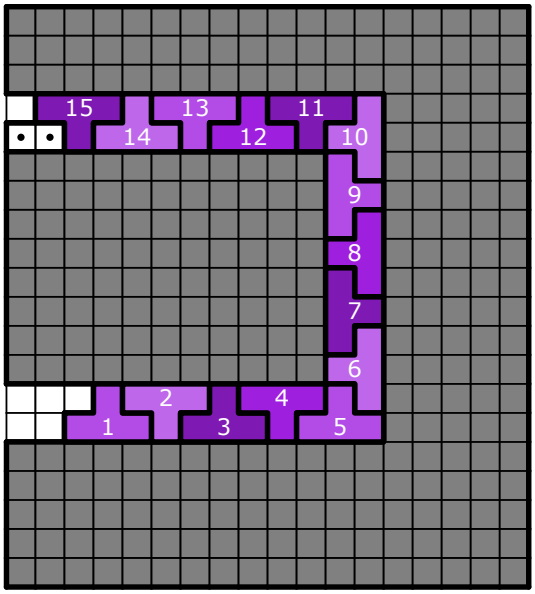
(a) 第一种铺砌　　**(b)** 第二种铺砌

图74： PF 装置的铺砌 + 放置顺序

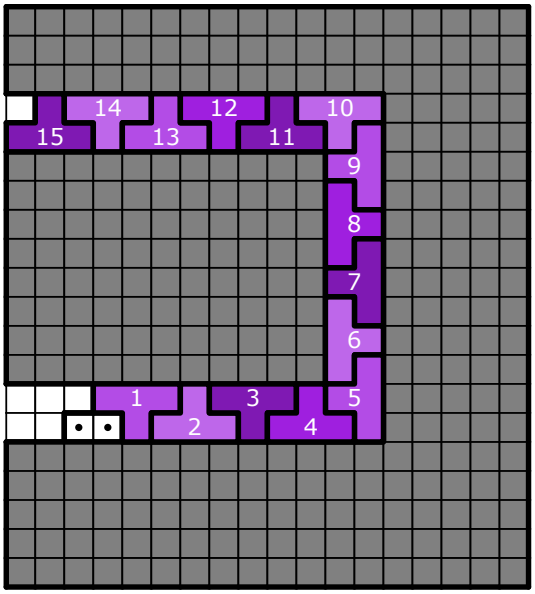


(a) 第一种铺砌　　**(b)** 第二种铺砌

图75： HL 小工具的铺砌

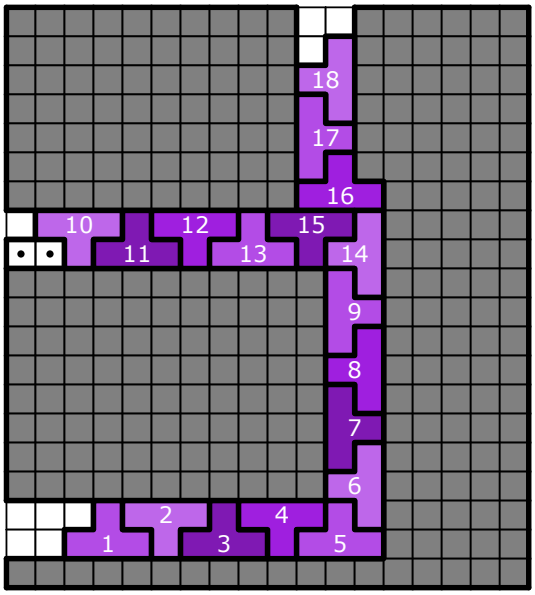


(a) 第一种铺砌

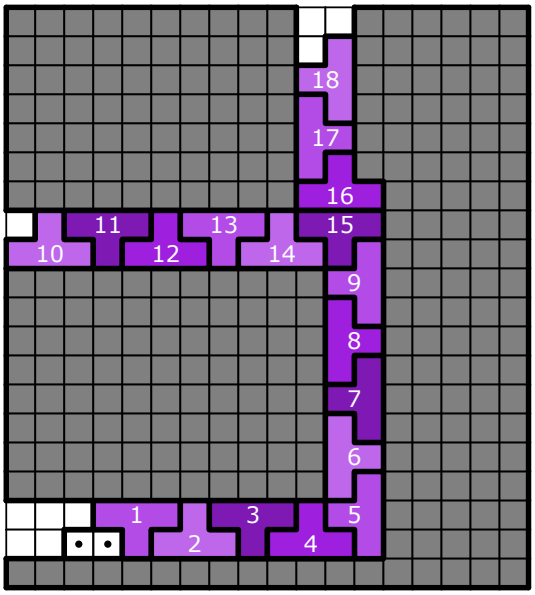


(b) 第二种铺砌

图76： U形转弯构件的铺砌 + 放置顺序（左-下-右情形与之对称）

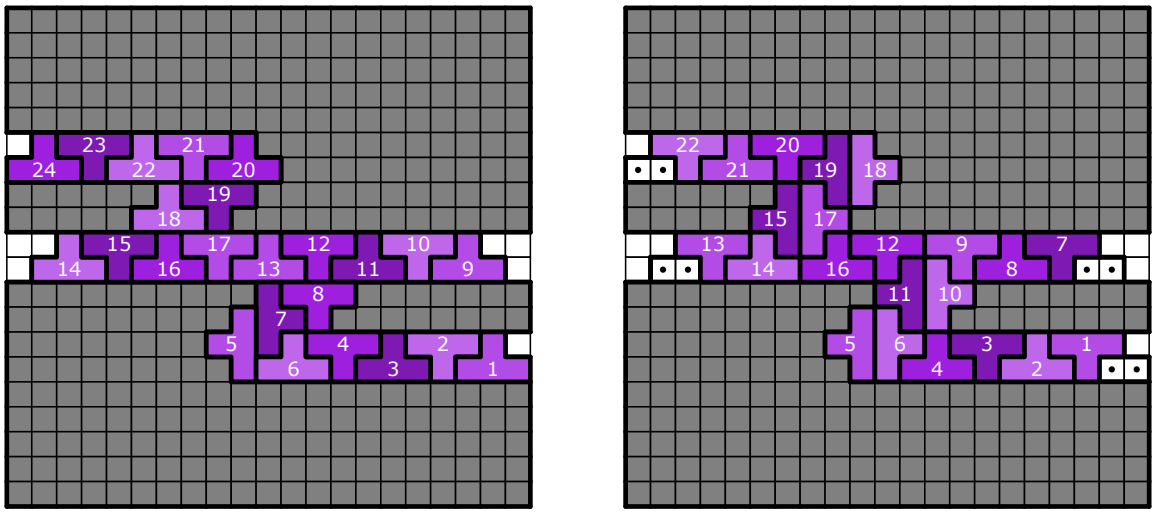


(a) 第一种铺砌



(b) 第二种铺砌

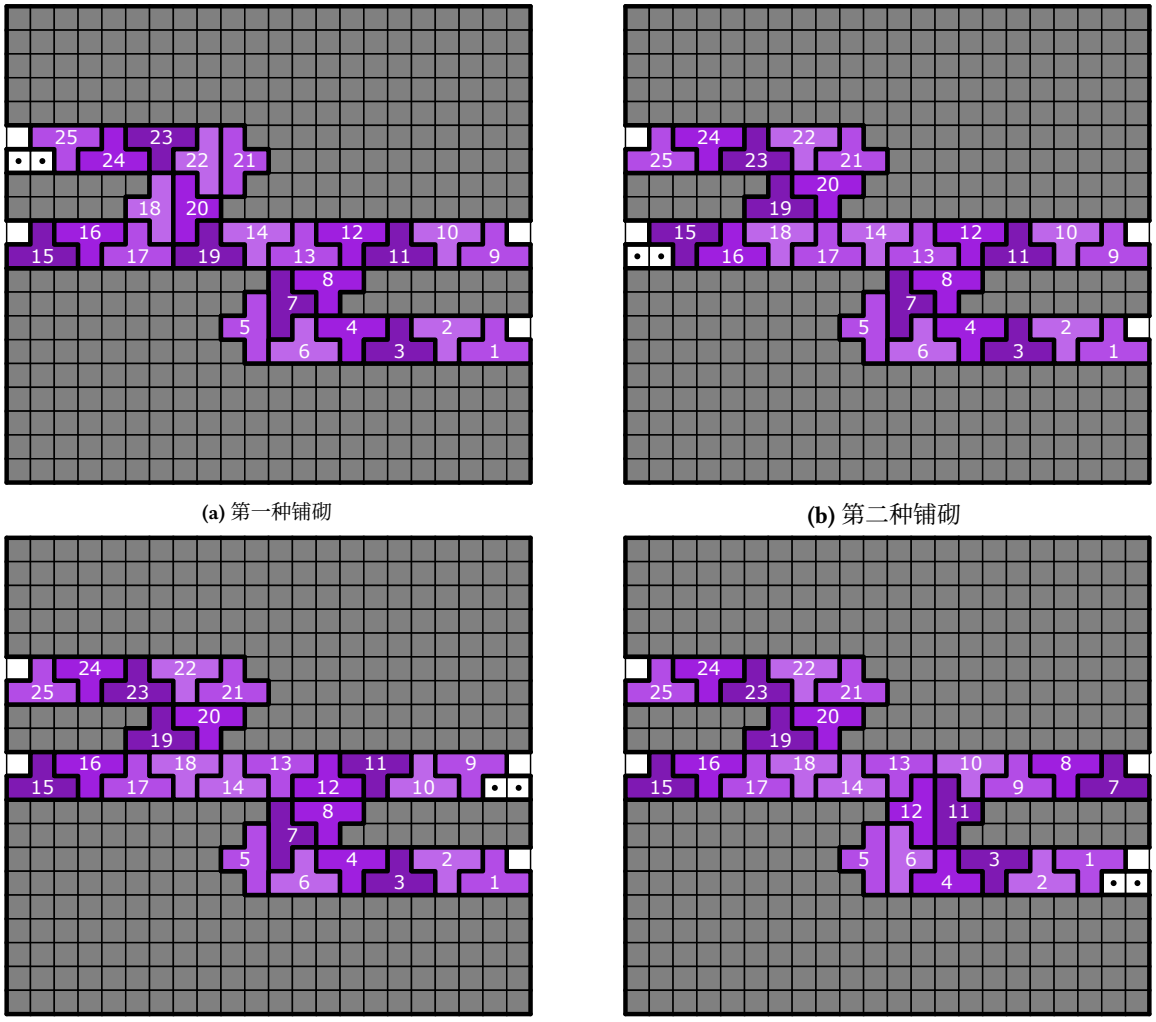
图77： EC组件的铺砌 + 放置顺序



(a) 第一种铺砌

(b) 第二种铺砌

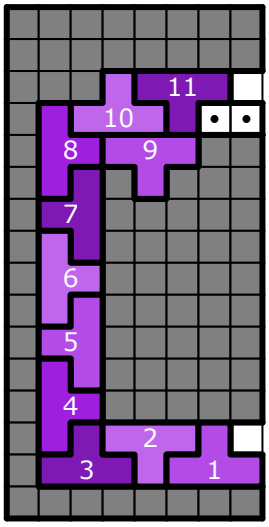
图78： 铺砌 + 0或4构件的放置顺序



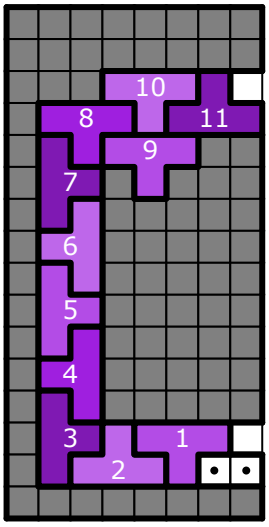
(a) 第一种铺砌 **(b)** 第二种铺砌

(c)第三种铺砌 **(d)** 第四种铺砌

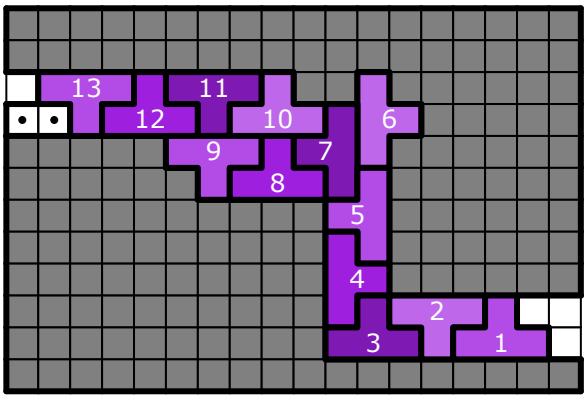
图 79： 铺砌 + 1-in-4构件的放置顺序



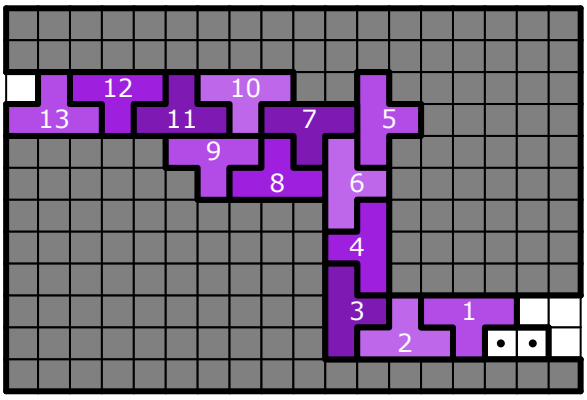
(a) 第一个 PF 构件的第一种铺砌



(b) 第一个 PF 构件的第二种铺砌

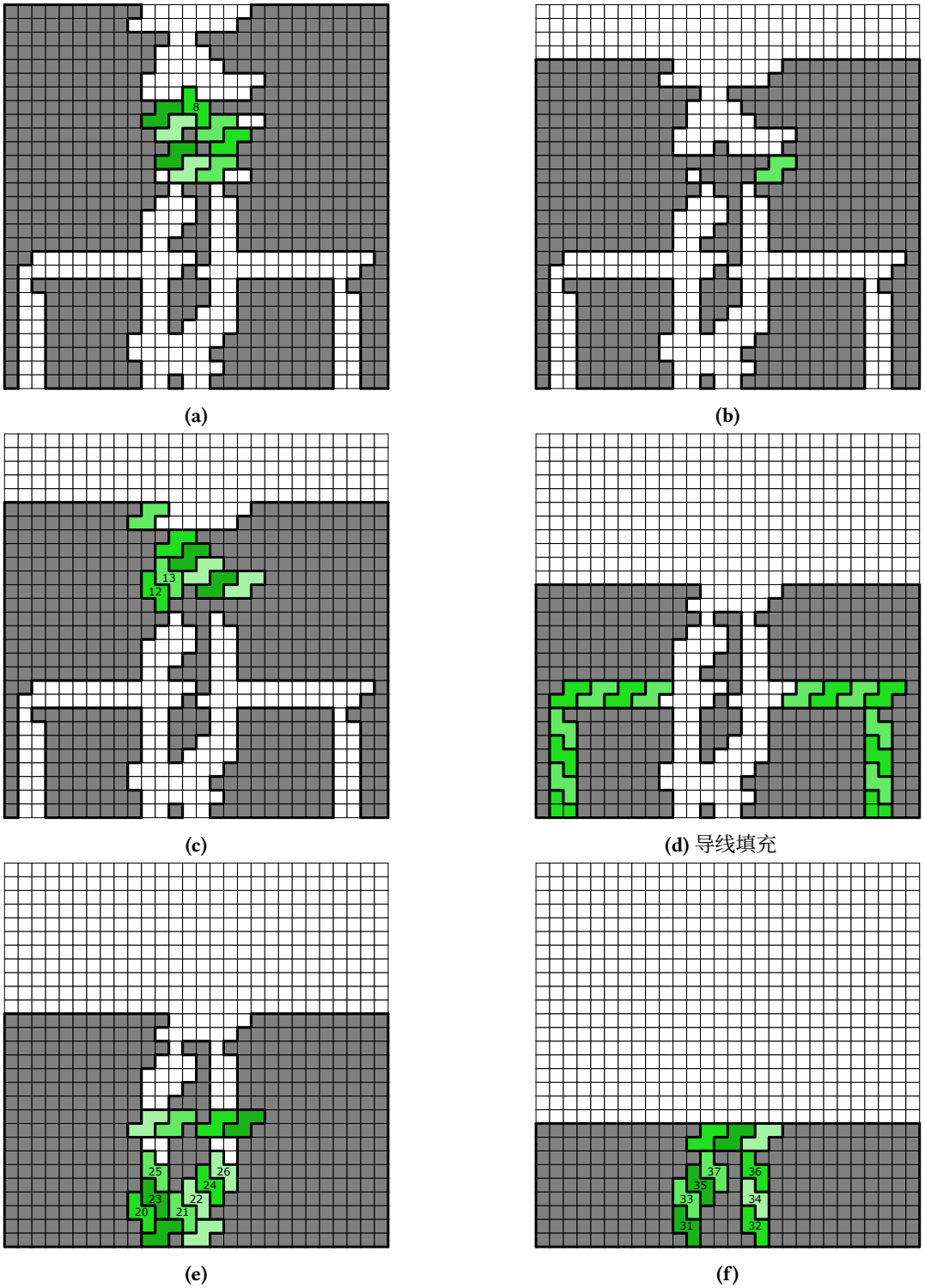


(c) 第二个 PF 构件的第一种铺砌



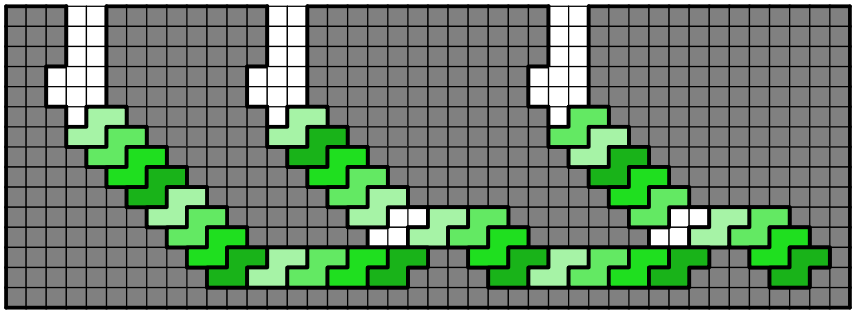
(d) 第二个 PF 构件的第二种铺砌

图 80： PF 构件的铺砌 + 放置顺序

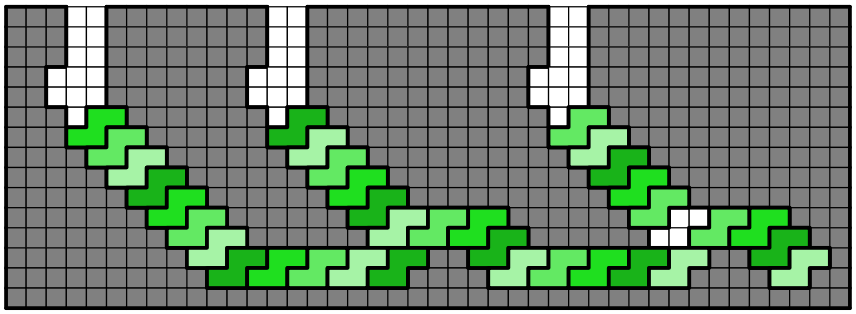


(a) **(b)** **(c)** **(d)** 导线填充 **(e)** **(f)**

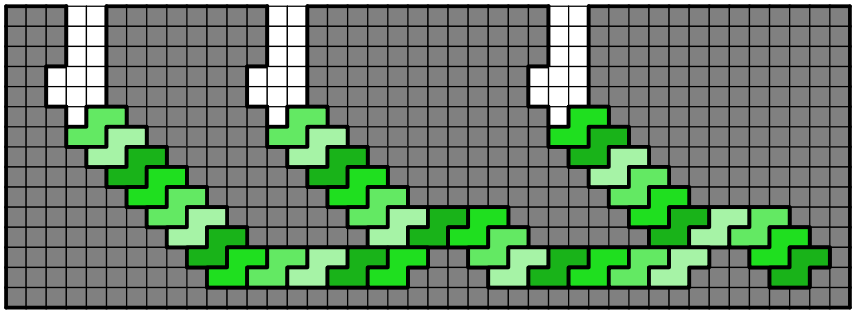
图 81： 变量构件的放置顺序，假定采用“真”设置（“假”设置会交换 (a) 中的“1”和“2”），且该变量一次正向出现、一次负向出现（出现次数不同意味着需要在 (d) 和 (e) 中放置额外方块以清空连线）



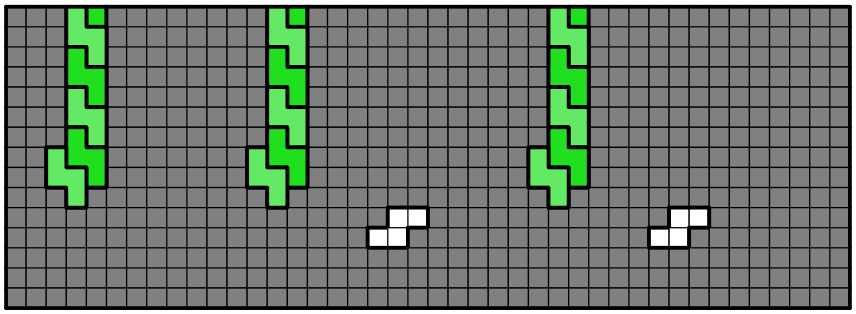
(a) 通过第一入口进行填充



(b)通过第二入口进行填充



(c)通过第三入口进行填充



(d) （在可攀爬垂直结构填充完成后进行）导线填充

图 82： 子句构件的已填充 + 放置顺序

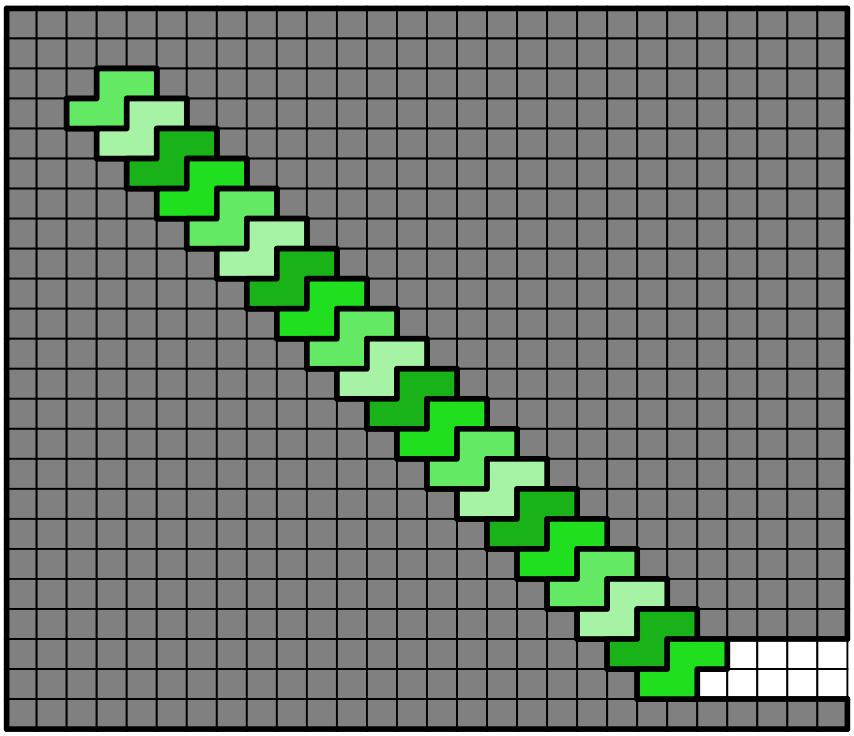


图 83： + 可攀爬垂直结构的铺砌放置顺序