# THE SQUARE ACHIEVEMENT GAME ON $N \times N$ BOARDS

SEAN REID

Abstract. Two players alternately claim cells of an $n \times n$ grid; the first to hold all four corners of an axis-parallel square wins, and a full board without one is a draw. The first player has a winning strategy if and only if $n \geq 5$, which answers a question of Erickson. For $n \geq 6$ one strategy serves every board: a rule with no lookup table whose moves after the second reply are all forcing, built around the *plus*, a stone with the four cells at a common distance from it along the axes, a shape proposed for this game by Sirgedas in 2010. The rule is checked on the boards of size 6 to 17, and a simulation theorem carries one checked run on the $23 \times 23$ board to every larger board. The proofs are formalized in Lean 4, and the theorem for the rule states that the tree of its own moves is winning.

## 1. Introduction

Two players alternately claim empty cells of an $n \times n$ grid, the first player moving first. A player wins upon holding all four corners of a square with horizontal and vertical sides, of any size: the cells $(r, c)$, $(r, c+s)$, $(r+s, c)$, $(r+s, c+s)$ for some $s \geq 1$. If the board fills with no player holding such a square, the game is a draw.

Erickson posed the game and asked for the outcome under optimal play [6, 8, 7]. It is an achievement game in the sense of Harary [11], played in the Maker–Maker convention: both players pursue the same goal, and a player who loses the race may still secure a draw. The convention goes back to the positional games of Hales and Jewett [10]; a recent unified treatment of achievement conventions, with the strategy-stealing theory in the generality used here, is [9], and the general theory of positional games is in Beck [4].

Two boundary results were known. Jenrich [12] reported computer searches showing that the second player can secure a draw for $n = 3, 4$ and that the first player can force a win for $n = 5$. Bacher and Eliahou [3] proved that every two-coloring of the $15 \times 15$ grid contains four equal-colored cells at the corners of an axis-parallel square, and that 15 is least with this property. Since no completed game on a board of size 15 or more is square-free, strategy stealing and determinacy give the first player a win there. Between these results nothing was published: for $6 \leq n \leq 14$ neither a winning strategy nor a drawing strategy was known [8, 1], and no coloring argument can reach this range, since Bacher and Eliahou also exhibit square-free two-colorings of the $14 \times 14$ board. One idea had been sketched. Answering Erickson's question on MathOverflow in 2010, Sirgedas [13] proposed that the first player open in the center and win by making a *cross*, a stone with its four neighbors at a common distance along the axes, each arm forcing a reply; he noted that the cross can also be built on two stones in a row or column at distance two, worked out three of the nine first replies on the $7 \times 7$ board up to symmetry, and guessed that $n = 5$ was a draw. Concurrently with the present work, Adler [1, 2] announced computer searches settling the nine sizes $6 \leq n \leq 14$ as first-player wins, with full strategy trees certified for $n = 6$ and 7 and search verdicts for the

2020 *Mathematics Subject Classification.* Primary 91A46; Secondary 68V20, 05D10.

*Key words and phrases.* Positional game, achievement game, strategy stealing, formal verification.

rest, cross-checked by independent runs at the smaller sizes; Adler notes that the results are unrefereed and names a conceptual proof with an explicit strategy as the open direction.

**Theorem 1.1.** *In the square achievement game on the $n \times n$ board, the first player has a winning strategy if and only if $n \geq 5$. For $n \leq 4$ the game is drawn under optimal play: each player can secure at least a draw.*

Relative to Adler's announcement, what is new is a single strategy that wins every board from $6 \times 6$ up, stated in a paragraph, and a proof checked by a proof assistant. The strategy is a rule built around one shape, the plus of Definition 2.1, which is Sirgedas's cross: the arms of a plus force the second player's replies one at a time, and the last arm threatens two squares at once. The sketch of 2010 covered one board and a few replies and did not say which plus to aim for, how to prepare one when the replies obstruct it, or why one board's analysis should carry to another. The rule settles the first two by a search with a fixed preference order, and the simulation theorem of Section 6 settles the third.

**Theorem 1.2.** *For every $n \geq 6$, the first player following the plus rule of Section 3 wins the square achievement game on the $n \times n$ board.*

Two facts make Theorem 1.2 a finite computation. After the second reply the rule makes only forcing moves, so every later reply is a forced block and the game tree branches freely at two levels only. Checking the tree on one board means enumerating the second player's first two replies and following a chain of forced moves from each pair; this check is run on every board from 6 to 17 (Section 5). The rule reads only the top-left $17 \times 17$ sub-board and one cell computed from the second player's first two replies. A stone outside the sub-board can affect the game only through that cell, and the rule never forces a block there, so a game on any larger board is mirrored move for move by a game on the $23 \times 23$ board. A simulation theorem, stated for any strategy, turns one checked run on that board into a proof for every board of size at least 17 (Section 6). Theorem 1.1 then follows with an exhaustive refutation for $n \leq 4$ and a strategy table for $n = 5$, whose play from the opening $(2, 2)$ is not purely forcing (Section 4).

Everything is formalized in Lean 4 [5] on mathlib [14] (Section 7). The formal theorem for the rule states that the tree of the rule's own moves is winning, not only that a winning tree exists; the runs of the checkers are discharged by evaluation, and the two properties of the rule that the simulation theorem needs are proved as lemmas about its definition. The rule was extracted from an earlier lookup-table strategy, almost all of whose forcing sequences were arms of a side-1 plus at $(2, 2)$ or $(3, 2)$, the rest preparing the side-2 plus at $(3, 2)$. The 2011 versions of [12] contained an error in the search program, corrected in 2012; the formalization makes every search here discharge a statement about the game rather than about a program.

## 2. Definitions

Cells are pairs $(r, c)$ of nonnegative integers, row then column, with row 0 at the top and column 0 at the left in the figures, where the first player's stones are black and the second player's white, and $B_n = \{0, \ldots, n-1\}^2$. For $s \geq 1$ the square with top-left corner $(r, c)$ and side $s$ is $Q(r, c, s) = \{(r, c), (r, c+s), (r+s, c), (r+s, c+s)\}$; a set of cells contains a square if it contains some $Q(r, c, s)$. The case $s = 1$ is allowed, so the smallest winning shape is a $2 \times 2$ array, the convention of [3, 12]; tilted squares do not count. A *position* is a pair of disjoint subsets of $B_n$, the stones of the two players. The game is finite with perfect information, so exactly one of three holds: the first player can force a win, the second can, or both can force at least a draw.

A player *threatens* at a cell $z$ if $z$ is empty and $z$ together with three of that player's stones is a square; the opponent *blocks* by playing $z$. A move that makes exactly one threat is *forcing*; a move that makes two threats wins if the opponent has no threat of their own, since one block leaves the other. A *ply* is one move of either player, so the first player moves at the odd plies. The rule of Section 3 plays inside the *region* $B_6$ and reads the second player's stones inside the *base board* $B_{17}$.

2.1. **Pluses.**

**Definition 2.1.** The *plus* of side $s$ at the *center* $c = (r, k)$ has four *arms*, the cells $(r - s, k)$, $(r + s, k)$, $(r, k - s)$, $(r, k + s)$, and four *diagonals*, the cells $(r \pm s, k \pm s)$ (Figure 1). Two arms are *perpendicular* if one is above or below the center and the other to its left or right; the diagonal between them is the cell in the row of one and the column of the other. A plus lies in the region if its arms and diagonals do, that is, $s \leq r, k$ and $r + s, k + s \leq 5$.

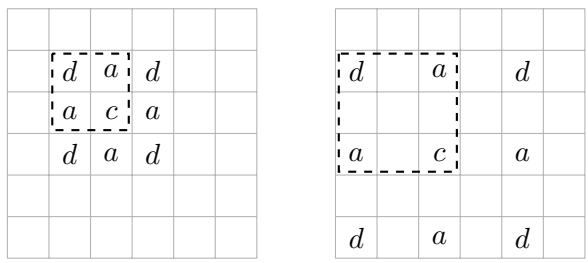


FIGURE 1. Left: the plus of side 1 at $c = (2, 2)$, with its arms $a$ and diagonals $d$. Right: the plus of side 2 at $c = (3, 2)$. In each, the dashed outline is one of the four squares of the plus, whose corners are the center, two perpendicular arms, and the diagonal between them.

The region holds 16 pluses of side 1, with centers in $\{1, \ldots, 4\}^2$, and four of side 2, with centers in $\{2, 3\}^2$; side 3 does not fit.

**Lemma 2.2.** *Suppose the first player holds the center $c$ and one arm $a$ of a plus of side $s$ whose four diagonals are empty. Playing an empty arm $b$ perpendicular to $a$ threatens at the diagonal between $a$ and $b$. If the first player also holds the arm $a'$ opposite to $a$, playing $b$ threatens at two cells, the diagonals between $a$ and $b$ and between $a'$ and $b$.*

*Proof.* Say $a = (r - s, k)$ and $b = (r, k - s)$. Then $c$, $a$, $b$ and $d = (r - s, k - s)$ are the corners of $Q(r - s, k - s, s)$, so with $c, a, b$ held and $d$ empty the first player threatens at $d$. With $a' = (r + s, k)$ also held, $c$, $a'$, $b$ and $(r + s, k - s)$ are the corners of $Q(r, k - s, s)$, and $(r + s, k - s) \neq d$. The other placements are symmetric. □

So from a center and one arm the first player reaches a double threat in three moves against a second player who has no threat and whose blocks make none. The two arms perpendicular to the held arm each force a block on a diagonal, and the opposite arm threatens at two diagonals at once (Figure 2). The opening $(2, 2)$ and the second move $(3, 2)$ are the center and an arm of the side-1 plus at $(2, 2)$, and an arm and the center of the side-1 plus at $(3, 2)$.

2.2. **Far stones and the hidden corner.** The rule's stones never leave $B_6$, and the second player's stones in a position the rule reaches are two free replies and forced blocks, the blocks in $B_6$ as well.

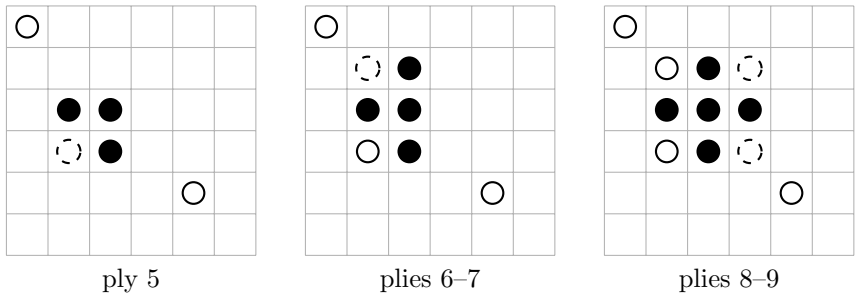


FIGURE 2. The basic line (Definition 2.8) against the replies $(0,0)$ and $(4,4)$. The first player holds $(2,2)$ and $(3,2)$; each arm threatens at the dashed cell, the second player blocks, and the last arm threatens twice.

**Lemma 2.3.** *Let $x \neq y$ be distinct corners of a square $Q(r,c,s)$ with $x, y \in B_m$. Then $Q(r,c,s) \subseteq B_{2m}$, and every corner of $Q(r,c,s)$ has its row or its column indexed below $m$.*

*Proof.* Two distinct corners either share a coordinate and differ by $s$ in the other, or differ by $s$ in both; either way $s \leq m-1$. Each coordinate of $(r,c)$ is bounded by the corresponding coordinate of a corner, so $r, c \leq m-1$, and the containment follows, indeed with $r+s, c+s \leq 2m-2$. If $x$ and $y$ share a row, the two remaining corners inherit their columns, both below $m$; a shared column is symmetric; and if $x$ and $y$ are diagonally opposite, all four corners lie in $B_m$. □

With $m = 6$, two stones of the region lie in a common square only if that square lies in $B_{12}$. Hence a second-player stone beyond $B_{17}$ shares no square with two blocks. A second-player threat through such a stone and a block must therefore have the other free reply as its third corner (Figure 3). The block in that situation is the one cell through which stones beyond $B_{17}$ can affect the game.

**Definition 2.4.** Let $x$ and $y$ be two cells each having a coordinate at least 12. Their *hidden corner* is a cell $b \in B_6$ such that $b$, $x$, $y$ are three corners of one square. The *abstraction* of a position consists of the first player's stones, the second player's stones inside $B_{17}$, and, once the second player has two stones, the hidden corner of the first two, if they have one.

**Lemma 2.5.** *Two cells $x \neq y$ have at most one hidden corner. If $Q(r,c,s)$ has a corner $b \in B_6$, has $x$ and $y$ among its other corners, and is not contained in $B_{17}$, then $b = (r,c)$, $s \geq 12$, each of $x$, $y$ has a coordinate at least 12, and $b$ is the hidden corner of $\{x, y\}$.*

*Proof.* Uniqueness. If $x$ and $y$ share a column, the square has side $s = |x_1 - y_1|$, and $b$ lies in the row of a member with row below 6, at distance $s$ to its left or right. Only one member can have row below 6: if both did, then $s < 6$ and the shared column, being the coordinate at least 12 of either, would put both candidates for $b$ at column 7 or more. For that member the cell to the right is excluded too: it lies in $B_6$ only if the member's column is below 6, and that column is the member's coordinate at least 12, its row being below 6. A shared row is symmetric. If $x$ and $y$ are diagonally opposite, the two remaining corners are $(x_1, y_2)$ and $(y_1, x_2)$, of which at most one lies in $B_6$, since otherwise all four coordinates of $x$ and $y$ would be below 6.

For the second claim, a second corner in $B_6$ would put the square inside $B_{12}$ by Lemma 2.3, against the hypothesis; so $b$ is alone in the region. Were $b$ not $(r,c)$, the corner $(r,c)$ would have both coordinates at most those of $b$ and would lie in $B_6$; so $b = (r,c)$. The corner

opposite $b$ has both coordinates at most $5 + s$, and the square leaves $B_{17}$ only if a coordinate reaches 17, so $s \geq 12$. Each of $x$, $y$ has a coordinate equal to $r + s$ or $c + s$, hence at least 12, and $b$ is their hidden corner by definition. □

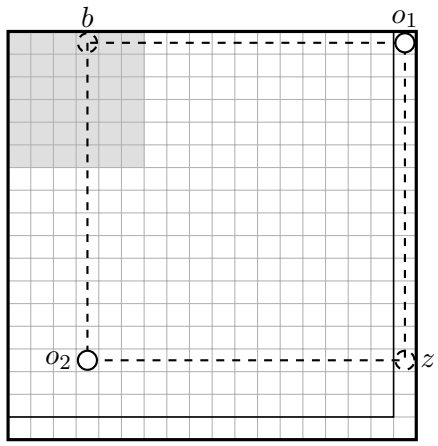


FIGURE 3. A hidden corner, on $B_{18}$. The inner outline is $B_{17}$ and the shaded square the region $B_6$. The second player's free replies $o_1 = (0, 17)$ and $o_2 = (14, 3)$ are two corners of the dashed square; its third corner $b = (0, 3)$ is their hidden corner, and $o_1$ and the fourth corner $z = (14, 17)$ lie beyond $B_{17}$. A second player forced to block at $b$ would threaten at $z$, a cell the rule does not read.

The rule therefore never makes a threat whose only block is the hidden corner (Definitions 2.7 and 2.8). It applies the restriction whenever the pair has a hidden corner, even when the square through it lies inside $B_{17}$ and the restriction is not needed.

*Remark* 2.6. The threshold 12 of Definition 2.4 and the base 17 are tied by $17 = 12 + 5$: a square through a region cell leaves $B_{17}$ only if its side is at least 12. With a threshold $t$ and the base $t + 5$ in their place, Lemma 2.5 holds exactly when $t \geq 11$. At $t = 10$ uniqueness fails: $(0, 10)$ and $(5, 10)$ would have the two hidden corners $(0, 5)$ and $(5, 5)$. So base 16 with threshold 11 would serve the lemma as well. The development uses $t = 2 \cdot 6 = 12$, the bound of Lemma 2.3, and the rule has been checked only with base 17.

2.3. **Tempo moves and lines.** When no plus can be completed directly, because the second player holds an arm or a diagonal or because the arms would not force, the rule first plays moves that each force a block and so add a stone. Threats in this subsection are those of the abstraction: squares of the first player inside $B_6$ and squares of the second player inside $B_{17}$.

**Definition 2.7.** A *tempo move* in a position is a cell of $B_6$ such that, after it is played, the second player has no threat and the first player has exactly one, at a cell other than the hidden corner, if the pair has one.

In a position where the second player has a threat, a tempo move must remove it, and the only way is to occupy its cell; so the second player has exactly one threat there and the tempo move is its block.

**Definition 2.8.** A *line* from a position consists of some tempo moves, each followed by its block, and then, in some order, the arms the first player does not yet hold of one plus in the

region whose center the first player holds and none of whose arms the second player holds. The arms are played until one makes at least two threats. After each move of the line the second player has no threat, and each arm before the last played makes exactly one threat, at a cell other than the hidden corner. The *key* of a line is

$$(\text{number of tempo moves},\ \text{number of arms not held},\ \text{side of the plus},\ \text{center of the plus}).$$

The rule prefers the smallest key in lexicographic order, cells ordered by row and then by column, then the line its search generates first.

The arms after the one that makes two threats are part of the line and of its key but are never played. A block may give the second player a threat; the next move of the line is then a tempo move that takes it. The rule searches for lines with at most three tempo moves. Whether a line exists, and which is preferred, depends only on the abstraction of the position.

## 3. The plus rule

Algorithm 1 is the move function and Algorithm 2 the line search it calls once, after the second reply. Both are stated as in the formal development. The second move $x_2(o_1)$ against a first reply $o_1$ is $(3, 2)$ unless $o_1$ is one of seven cells, for which Figure 4 gives the move.

**Algorithm 1** The move function.

1: **if** some empty cell completes a square of the first player **then**
2:  play the least such cell
3: **else if** some empty cell completes a square of the second player **then**
4:  play the least such cell
5: **else if** the first player has no stone **then**
6:  play $(2, 2)$
7: **else if** the first player does not hold $x_2(o_1)$ **then**
8:  play $x_2(o_1)$
9: **else**
10:  $L \leftarrow$ the line of Algorithm 2 from the abstraction after the second reply
11:  play the first cell of $L$ the first player does not hold

Steps 1 to 4 scan the squares of the abstraction: the first player's, which lie in $B_6$, and the second player's with all corners inside $B_{17}$. A second-player square with a corner beyond $B_{17}$ would pass through the hidden corner, where the rule never forces a block. Independently, the checker of Section 5 confirms at every node that the second player has no threat at all. If Algorithm 2 returns no line the rule is undefined; the checker certifies that this never happens at a position the rule reaches.

**Algorithm 2** The line search, from an abstract position $A$.

1: $F \leftarrow [(A, \varepsilon)]$ ▷ positions reached, with the tempo moves that reached them
2: **for** $d = 0, 1, 2, 3$ **do**
3: $R \leftarrow \varnothing$
4: **for all** $(X, p) \in F$ in order, such that the second player has no threat in $X$ **do**
5: **for all** pluses $P$ in the region whose center the first player holds in $X$ **do**
6: $T \leftarrow$ the arms of $P$ the first player does not hold in $X$
7: **if** $T \neq \varnothing$ and the second player holds no cell of $T$ **then**
8: $\pi \leftarrow$ the first ordering of $T$ such that $p\,\pi$ is a line from $A$
9: **if** $\pi$ exists **then** record $p\,\pi$ in $R$ with the key $(d, |T|, \text{side of } P, \text{center of } P)$
10: **if** $R \neq \varnothing$ **then return** the first line recorded in $R$ among those with the smallest key
11: $F \leftarrow$ the successors of the members of $F$, in order
12: **return** no line

The successors of $(X, p)$ in step 11 are the pairs $(X', p\,m)$ for the tempo moves $m$ in $X$, taken in cell order, where $X'$ is $X$ after $m$ and its block. They are taken from every member of $F$, including positions in which the second player has a threat; such a position offers no plus (step 4), but its tempo moves, which must remove the threat, are explored. Step 8 tries the orderings of $T$ in lexicographic order, the arms listed above, below, left of, and right of the center, so that two implementations choose the same line.

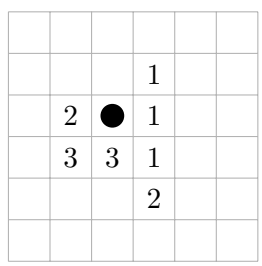


FIGURE 4. The second move. The black stone is the opening. Against a first reply on a cell marked 1 the second move is $(2, 1)$; on 2 it is $(2, 3)$; on 3 it is $(1, 2)$. Against every other first reply, on this board or beyond it, it is $(3, 2)$.

The exceptions are exactly the first replies against which $(3, 2)$ is occupied or fails. They were found by search: for each first reply on $B_6$ the second move is the first cell in the order $(3, 2)$, $(2, 3)$, $(2, 1)$, $(1, 2)$ against which the rest of the rule beats every second reply.

Table 1 lists two complete games in which the second player always blocks. In the first the replies $(0, 0)$ and $(4, 4)$ are away from the opening pair and the basic line wins at once, key $(0, 3, 1, (2, 2))$. In the second the reply $(3, 1)$ is a diagonal of the side-1 plus at $(2, 2)$ and an arm of the side-1 plus at $(3, 2)$, no line with fewer than two tempo moves exists, and the line has key $(2, 4, 2, (3, 2))$: two tempo moves and then the four arms of the side-2 plus at $(3, 2)$. Figure 5 shows the position before its last move. After the forced block at ply 10 of the first game and at ply 16 of the second, the second player does hold a threat, at the cell the first player completes next; Definition 2.8 constrains the second player's threats only after the first player's moves.

The line is chosen once, after the second reply, and followed. Since the blocks are forced, the line determines every position of the forced tree (Section 5) below the second reply. Until

| ply | first player | second player | what the move does |
|---|---|---|---|
| 1, 2 | $(2,2)$ | $(0,0)$ | opening; free reply |
| 3, 4 | $(3,2)$ | $(4,4)$ | second move; free reply |
| 5, 6 | $(2,1)$ | $(3,1)$ | arm; threat at $(3,1)$ through $Q(2,1,1)$, blocked |
| 7, 8 | $(1,2)$ | $(1,1)$ | arm; threat at $(1,1)$ through $Q(1,1,1)$, blocked |
| 9, 10 | $(2,3)$ | $(1,3)$ | arm; threats at $(1,3)$ and $(3,3)$; one is blocked |
| 11 | $(3,3)$ | | completes $Q(2,2,1)$; wins |
| 1, 2 | $(2,2)$ | $(0,0)$ | opening; free reply |
| 3, 4 | $(3,2)$ | $(3,1)$ | second move; free reply on a diagonal |
| 5, 6 | $(3,3)$ | $(2,3)$ | tempo move; threat at $(2,3)$ through $Q(2,2,1)$, blocked |
| 7, 8 | $(4,3)$ | $(4,2)$ | tempo move; threat at $(4,2)$ through $Q(3,2,1)$, blocked |
| 9, 10 | $(3,4)$ | $(4,4)$ | arm; threat at $(4,4)$ through $Q(3,3,1)$, blocked |
| 11, 12 | $(1,2)$ | $(1,4)$ | arm; threat at $(1,4)$ through $Q(1,2,2)$, blocked |
| 13, 14 | $(5,2)$ | $(5,4)$ | arm; threat at $(5,4)$ through $Q(3,2,2)$, blocked |
| 15, 16 | $(3,0)$ | $(1,0)$ | arm; threats at $(1,0)$ and $(5,0)$; one is blocked |
| 17 | $(5,0)$ | | completes $Q(3,0,2)$; wins |

TABLE 1. The basic line and the longest line as complete games.

the completing move, the first player's stones are the opening pair together with a prefix of the line. Committing to the line makes the forced tree a function of the reply pair, which Section 6 uses.

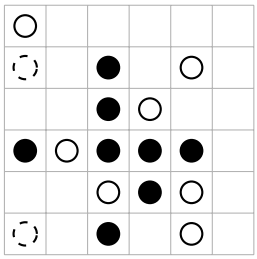

FIGURE 5. The second game of Table 1 after ply 15. The side-2 plus at $(3,2)$ stands with the second player's blocks on its diagonals $(1,4)$ and $(5,4)$, and the last arm $(3,0)$ threatens at the other two diagonals, dashed.

The Python implementation of the rule was run over every reply pair on each board from 6 to 17, on the anchor board $B_{23}$, and from every anchored position, both of Section 6, recording the line, its tempo moves, and the ply of the win when the second player blocks. On every board from 6 to 17, on the anchor board, and from every anchored position, the rule plays one of the same 79 lines; the 277,728 pairs on the anchor board have 43,424 distinct abstractions. Every abstraction that occurs on a board beyond 17 occurs on the anchor board or in an anchored position (Section 6), so the same holds on every board. These counts come from the Python implementation and the lines the checker prints; they are not part of the formal theorem. Table 2 splits the pairs by the number of tempo moves. Every pair won by a side-1 plus needs no tempo move and every pair won by a side-2 plus needs at least one, so the first column also counts the side-1 pairs. In 19 of the 1,155 pairs on $B_6$, all with three tempo

moves, a block gives the second player a threat that the next tempo move takes. The longest line, the second game of Table 1, has 6 first-player moves after the opening pair, and every game in which the second player blocks ends by ply 17; a reply that blocks nothing leaves a threat that the rule completes at its next move. So the rule wins every game by the 17th ply. This bound is computed from the 79 lines and is not part of the formal theorem.

| tempo moves in the line | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| reply pairs on $B_6$ | 985 | 91 | 54 | 25 |
| reply pairs on $B_{17}$ | 81,186 | 1,103 | 54 | 25 |

TABLE 2. The rule's lines by the number of tempo moves. The $54 + 25$ pairs needing two or three tempo moves have both replies in $B_6$ and are the same on every board; the other columns grow with the board.

## 4. Small boards and the $5 \times 5$ board

**Lemma 4.1.** *The second player has no winning strategy.*

*Proof.* Suppose the second player had a winning strategy $\tau$. Let the first player open with an arbitrary move and thereafter follow $\tau$ as if the extra stone were absent, making a further arbitrary move whenever $\tau$ selects an occupied cell. The winning condition is monotone in a player's stone set, so surplus stones never hurt (a surplus stone may itself complete a square, which ends the game in the first player's favor), and the first player wins; but $\tau$ was assumed to win against all first-player play, a contradiction. □

**Proposition 4.2.** *For $n \leq 4$ the first player has no winning strategy. With Lemma 4.1, these games are drawn under optimal play: each player can secure at least a draw.*

*Proof.* For $n \leq 2$ the board has at most four cells, so neither player ever holds four. For $n = 3, 4$ the claim is a statement about a finite game tree, verified by an exhaustive search that is part of the formal development: for every first-player move the search exhibits a reply after which the first player still cannot force a win. The result agrees with [12]. The step from "neither player can force a win" to "each player can secure a draw" is determinacy of the finite game, proved by induction on the number of empty cells. □

**Proposition 4.3.** *The first player has a winning strategy on $B_5$.*

*Proof.* The strategy is a table, verified by a checker that walks its complete game tree on $B_5$, every reply at every node, and finds every branch ending in a first-player square. The checker's soundness theorem and the run are part of the formal development (Section 7). □

The table was extracted from a full solution of the $5 \times 5$ game ($1.79 \times 10^8$ positions solved, with draws scored against the first player). Like the rule, it completes a square when it can and otherwise blocks one before consulting the table. It opens at $(2, 2)$, gives a second move for each of the 24 first replies, and after the second reply plays a sequence of forcing moves ending in a double threat where one exists. For 104 of the reply pairs, with the table's second move, the solver finds no winning continuation of at most eight moves in which every first-player move makes a threat, so the table plays a *quiet* move there, one making no threat, and continues with a subtable indexed by the reply (Figure 6). That no such continuation exists is a computation of the solver and is not formalized; the win of the table is.

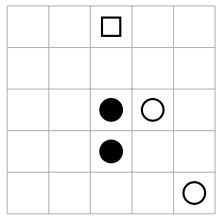

FIGURE 6. A quiet move of the $5 \times 5$ table. The first player opened at $(2, 2)$ and answered the reply $(4, 4)$ with the table's second move $(3, 2)$. After the second reply $(2, 3)$, one of the 104 pairs with no forcing continuation, the table plays $(0, 2)$, marked by the square, which makes no threat, and then consults a subtable indexed by the reply to it.

## 5. Verification

Fix a strategy $\sigma$ and a board size $n$. The *full tree* of $\sigma$ on $B_n$ is the game tree in which the first player's moves are given by $\sigma$ and the second player's moves range over all empty cells; if every branch ends in a first-player square, the first player wins. The $5 \times 5$ table is checked on its full tree.

The rule is checked on a smaller tree. The *forced tree* of $\sigma$ on $B_n$ has the first player playing $\sigma$'s move, the second player's replies ranging over all empty cells at the first two levels, and below every later node only over the cells that block a threat of the move just played. The checker walks this tree under guards:

(G1) after the second reply, every move of $\sigma$ wins outright or makes a threat;
(G2) every nonwinning move and every threat cell lies in $B_6$;
(G3) after every move of $\sigma$ the second player has no threat;
(G4) optionally, for a given cell $g$, no move of $\sigma$ has $g$ as its only threat cell.

A reply that blocks nothing leaves a threat standing, which the strategy completes at its next move, so every omitted reply loses in two plies.

**Proposition 5.1.** *Let $\sigma$ be a strategy that, whenever the first player's stones lie in $B_6$ and some cell completes one of the first player's squares, plays such a cell. If the forced tree of $\sigma$ on $B_n$ passes the guards and every branch ends in a first-player square, then the tree in which the first player plays $\sigma$'s move at every node is winning; in particular the first player wins on $B_n$.*

*Proof.* This is the soundness theorem of the forced checker, an induction over the checker's recursion (Section 7). At a node after the second reply, the move $m$ of $\sigma$ either completes a square or, by (G1), makes a threat. By (G3) no reply completes a second-player square. A reply that blocks a threat of $m$ is a child in the forced tree and is winning by induction. A reply that blocks nothing leaves a threat of $m$ standing, the first player's stones lie in $B_6$ by (G2), and the hypothesis makes $\sigma$ complete the threat at its next move, so that the winning tree is $\sigma$'s own. □

The rule satisfies the hypothesis: its first clause plays a completion found by its threat scan, and that scan agrees with the checker's whenever the first player's stones lie in $B_6$, which is proved as a lemma (Section 7).

The rule passes the forced tree on every board from 6 to 17, one run of the checker per board with $(n^2 - 1)(n^2 - 3)$ reply pairs at the root (Table 3 in Section 7). A Python implementation

of the rule and a full-tree verifier were used while the rule was being found. The checker executable prints the line the Lean rule commits to for every reply pair of $B_6$ and for every anchored position of Section 6, and a test of the Python implementation reproduces every one; the full-tree verifier confirms the win on $B_6$. Only the Lean runs enter the proof.

## 6. Boards of size at least 17

**Theorem 6.1.** *For every $n \geq 17$ the first player following the rule wins on $B_n$.*

By Lemma 2.5, a stone beyond $B_{17}$ can affect the rule's play only through a hidden corner, and the rule never forces a block there. The proof replays a checked run on $B_{23}$ as a run on $B_n$. Replies inside $B_{17}$ are read verbatim. A reply beyond $B_{17}$ is replaced, in the run on $B_{23}$, by a *stand-in*: a fixed cell beyond $B_{17}$ when the pair has no hidden corner, and an *anchor*, a cell beyond $B_{17}$ that forms with the other reply the same hidden corner, when it has one. The *anchor board* $B_{23}$ is the least board holding every anchor (Remark 6.3).

A *history* is the sequence of moves played so far, and for a history $h$ and a sequence $t$ of further moves $h\,t$ is their concatenation. A strategy $\sigma$ is given the board, and $\sigma_n(h)$ is its move after $h$ on $B_n$. The guards are those of Section 5.

**Theorem 6.2.** *Let $\sigma$ be a strategy, let $m, n \geq 17$, and let $x_1, x_2 \in B_6$. Let $h = (x_1, o_1, x_2, o_2)$ be a history on $B_n$ and $h' = (x_1, o_1', x_2, o_2')$ a history on $B_m$ such that for each $i$ either $o_i' = o_i \in B_{17}$, or $o_i \in B_n \setminus B_{17}$ and $o_i' \in B_m \setminus B_{12}$. Let $g$ be either no cell or a cell of $B_6$. Suppose that*

- (i) *on every history extending $h$ in which the first player's stones lie in $B_6$, $\sigma$ plays a cell completing one of the first player's squares whenever one exists;*
- (ii) *$\sigma_n(h\,t) = \sigma_m(h'\,t)$ for every sequence $t$ of further moves;*
- (iii) *the forced tree of $\sigma$ on $B_m$ below $h'$ passes the guards, with $g$ as the cell of (G4) if $g$ is a cell, and every branch ends in a first-player square;*
- (iv) *if $o_1$ or $o_2$ lies beyond $B_{17}$, then either $g$ is no cell and $\{o_1, o_2\}$ has no hidden corner (Definition 2.4), in particular if one of them has no coordinate at least 12, or $g$ is the hidden corner of $\{o_1, o_2\}$.*

*Then the tree in which the first player plays $\sigma$'s move at every node below $h$ on $B_n$ is winning.*

*Proof.* By induction along the forced tree of (iii). Consider a node of that tree together with the corresponding real position on $B_n$; the two differ only at the two reply slots, and the blocks played so far, all in $B_6$ by (G2), are shared. The two positions agree inside $B_{12}$: a real reply beyond $B_{17}$ and its stand-in both lie outside $B_{12}$, and every other stone is shared. By (ii) $\sigma$ plays the same cell $m$ in both. If $m$ completes a square in the anchor position, the square has three corners among the first player's stones, which lie in $B_6$, so it lies in $B_{12}$ by Lemma 2.3, and $m$ is empty and completes it on $B_n$ too. Otherwise $m \in B_6$ by (G2), and the threat cells of $m$ are the same on both boards, since each threat square has three corners in $B_6$ and so lies in $B_{12}$; by (G3) the second player has no threat in the anchor position.

No reply $r$ completes a second-player square on $B_n$. Suppose $Q$ were such a square.

Case 1: $Q \subseteq B_{17}$. Its corners other than $r$ are second-player stones inside $B_{17}$, so blocks or replies read verbatim, all shared with the anchor position, since the real replies beyond $B_{17}$ lie outside $Q$ and a stand-in only adds a stone. Then either $r$ is a stand-in and the anchor position already holds $Q$, or $r$ is empty in the anchor position and completes $Q$ there. The second case contradicts (G3) at the current node. In the first case, the anchor position holds no second-player square: the two replies alone are too few corners, so the last corner placed

was a block, and that block answered a threat that was standing after the previous move of $\sigma$, again contradicting (G3).

Case 2: $Q \not\subseteq B_{17}$. Two blocks among its corners would confine it to $B_{12}$ by Lemma 2.3, so its corners are $r$, one block $d \in B_6$, and both replies, and a reply lies beyond $B_{17}$: any three corners of a square include a diagonally opposite pair, and such a pair in $B_{17}$ confines $Q$ there by Lemma 2.3. By Lemma 2.5 $d$ is the hidden corner of $\{o_1, o_2\}$, which contradicts (iv): either the pair has none, or its hidden corner is $g$, and $g$ is never a block, because a block at $g$ arises only when $g$ is the only threat cell, which (G4) excludes, or when the second player chooses $g$ among two threat cells, in which case $\sigma$ completes the other by (i) and the branch ends.

A reply that blocks nothing leaves a threat, which $\sigma$ completes by (i). A block $r \neq g$ is a block in the anchor run as well, and the induction continues along the anchor branch, whose invariants are preserved: the new block lies in $B_6$ and is not $g$. □

The formal theorem is stated for any node of the anchor run reached by forced play, with the blocks so far in $B_6$ and $g$ not among the second player's stones, and is proved by induction on the checker's fuel.

For the rule, hypotheses (i) and (ii) are lemmas about its definition. Hypothesis (i) is the hypothesis of Proposition 5.1, which the rule satisfies as noted there. For (ii), the rule's move depends on a history only through its abstraction and through the second move of the first reply. A reply beyond $B_{17}$ and its stand-in both drop out of the abstraction, a reply inside $B_{17}$ is its own stand-in, and the stand-in pair is chosen with the same hidden corner as the real pair, so the two abstractions agree at every node; the second moves agree because every first reply beyond $B_6$, and every stand-in, has the default second move $(3, 2)$. Figure 7 shows a real history and its anchor history.

Pairs with a hidden corner $b = (b_1, b_2)$ are covered by three families of *anchored runs*, each a forced run of the rule on $B_{23}$ from the history $((2, 2), o_1', x_2(o_1'), o_2')$ with $b$ as the cell of (G4), preceded by a check that $b$ is the hidden corner of the anchor pair.

- The *far* family, for both replies beyond $B_{17}$: $o_1' = (b_1+17, b_2+17)$ and $o_2' = (b_1+17, b_2)$, two corners of $Q(b_1, b_2, 17)$; one run per $b \in B_6$, 36 runs.
- The *exact* family, for $o_1 \in B_{17}$ and $o_2$ beyond it. By Lemma 2.5, the square through $b$ and the pair is $Q(b_1, b_2, s)$ with $s \geq 12$, and it leaves $B_{17}$. The corner opposite $b$ bounds the square, so $o_1$ is not that corner; hence $o_1$ lies in the row or the column of $b$ at distance $s$, and $o_1 \in B_{17}$ puts that coordinate of $o_1$ below 17. Say $o_1 = (a, w)$ and $b = (a, h)$ with $a, h < 6$, $12 \leq w < 17$, and $s = w - h$. Then $o_1' = o_1$ and $o_2' = (a+s, w)$, the corner of $Q(a, h, s)$ opposite $b$, which lies beyond $B_{17}$ exactly when the square does. One run per $(a, w, h)$ with $a + w - h \geq 17$ and one per transposed triple, 70 runs.
- The *far-near* family, for $o_1$ beyond $B_{17}$ and $o_2 \in B_{17}$: the exact family with the two replies exchanged, 70 runs.

This gives 176 anchored runs in all.

*Proof of Theorem 6.1.* Let $n \geq 17$. The rule opens at $(2, 2)$ and answers the first reply $o_1$ with $x_2 = x_2(o_1) \in B_6$; both moves are legal and complete nothing, so it remains to show that the rule's tree below $h = ((2, 2), o_1, x_2, o_2)$ is winning for every reply pair. Apply Theorem 6.2 with $m = 23$, $\sigma$ the rule, and $h'$, $g$ as follows, calling a reply *far* if it lies beyond $B_{17}$.

(a) Neither reply far: $h' = h$, $g$ no cell, and (iii) is the branch at $h$ of the forced run of the rule on $B_{23}$; (iv) is vacuous.

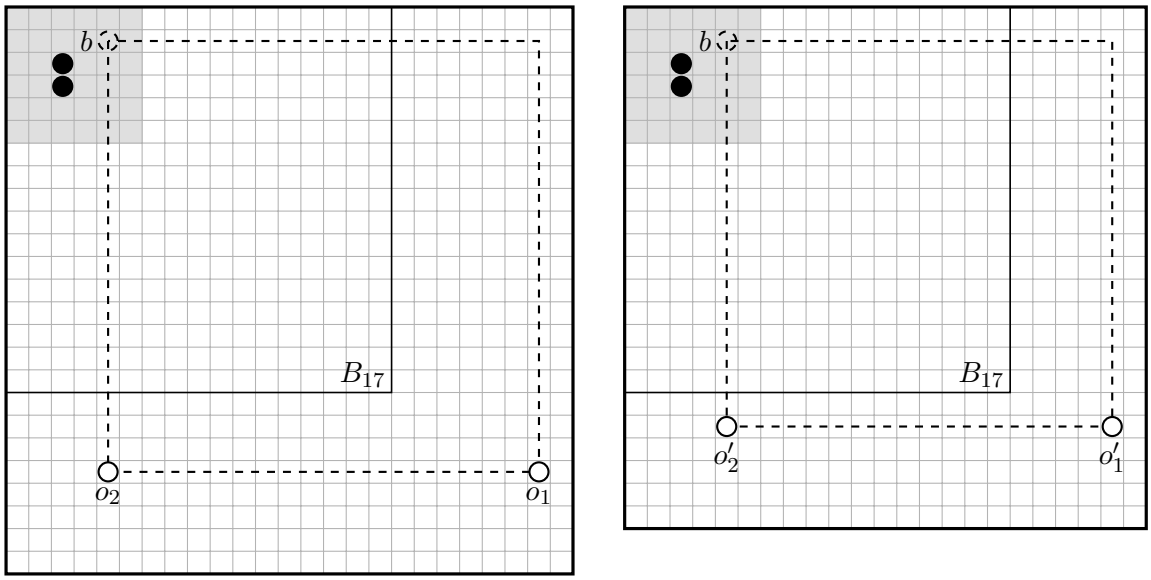


FIGURE 7. Left: a real history on $B_{25}$, with $B_{17}$ outlined and $B_6$ shaded. The first player holds $(2,2)$ and $(3,2)$; the replies $o_1 = (20,23)$ and $o_2 = (20,4)$, both beyond $B_{17}$, are two corners of the dashed square $Q(1,4,19)$ with hidden corner $b = (1,4)$. Right: the history on the anchor board $B_{23}$; the anchors $o'_1 = (18,21)$ and $o'_2 = (18,4)$ are two corners of $Q(1,4,17)$ with the same hidden corner. The two histories have the same abstraction, so the rule plays the same line from each.

(b) A far reply and no hidden corner: a far $o_1$ is replaced by $(22,22)$ and a far $o_2$ by $(22,21)$, $g$ is no cell, and (iii) is again a branch of the run on $B_{23}$. A stand-in forms no hidden corner with a cell of $B_{17}$: it shares no row or column with such a cell, and if the two are diagonally opposite corners of a square, the other two corners have a row or a column of at least 21. The two stand-ins share row 22, so a hidden corner of theirs would lie in row 21 or 23. So the stand-in pair has no hidden corner either.
(c) Both replies far with hidden corner $b$: the far family, $g = b$.
(d) $o_1 \in B_{17}$, $o_2$ far, hidden corner $b$: the exact family, $g = b$.
(e) $o_1$ far, $o_2 \in B_{17}$, hidden corner $b$: the far-near family, $g = b$.

In (c) to (e), (iii) is the anchored run, and (iv) holds by the case hypothesis. Hypothesis (i) is the lemma above; (ii) holds because the real and anchor pairs have the same hidden corner, checked at the head of each anchored run and shown in (b) for the stand-ins, and the same second move, so that the abstractions of $h\,t$ and $h'\,t$ agree for every $t$. □

*Remark* 6.3. The anchors of the far family reach coordinate $5 + 17 = 22$, those of the other two families $a + s \leq 5 + 16 = 21$, and the stand-ins sit at 22, so $B_{23}$ is the least board holding them all; any larger board would serve. The anchor board, the stand-ins, and the anchors are choices made in the proof, and the rule does not depend on them.

*Proof of Theorems 1.1 and 1.2.* Theorem 1.2 follows from the checked runs at 6 to 17 with Proposition 5.1, and from Theorem 6.1 for $n \geq 17$. Theorem 1.1 follows from Propositions 4.2 and 4.3 and from Theorem 1.2 with the moves forgotten. □

## 7. Formalization

The Lean development defines cells, boards, and squares as in Section 2 (`Core/Square.lean`), and the winning predicate `P1CanWin` (`Game/Rules.lean`) as an inductive family whose derivations are exactly strategy trees: one first-player move per node, every reply covered, and no derivation for a game that ends in a draw. The second player's win is defined from it by swapping roles after a move; the first player's ability to secure a draw is a second inductive family, whose branches may also end on a full board, and the second player's is defined from that by the same swap (`Game/Outcome.lean`). Strategy stealing (`second_player_no_win`, with the monotonicity step `P1CanWin.insert_left`) and determinacy (`P1CanWin.or_p2CanDraw`) are proved against these definitions with no computation. These definitions, a few dozen lines, are the whole specification; everything else is proved against them, and the development has no `sorry` and no unproved declaration. The two main statements are

```
outcome (n : ℕ) : (P1CanWin n ∅ ∅ ↔ 5 ≤ n) ∧ ((P1CanDraw n ∅ ∅ ∧ P2CanDraw n
                  ∅ ∅) ↔ n ≤ 4)
plus_wins_with {n : ℕ} (h : 6 ≤ n) : WinsWith plusMove n []
```

where `P1CanWin n A B` says that the player to move, holding $A$ against $B$ on $B_n$, can force a win. Running `lake build` checks everything, `lake exe checker n` replays the certificate for $B_n$, and `#print axioms` lists the axioms of a theorem.

The predicate behind Theorem 1.2 is `WinsWith`. For a strategy $\sigma$, a board size $n$, and a history $h$ with the first player to move, `WinsWith` $\sigma$ $n$ $h$ holds if $\sigma$'s move at $h$ is legal and either completes a square or admits at least one reply, no reply completes a second-player square, and `WinsWith` holds again after every reply. It has the constructors of `P1CanWin` with one extra premise on each, that the move is the one $\sigma$ returns, and implies `P1CanWin` by forgetting that premise. A checker that plays $\sigma$ at every node proves `WinsWith` only if its soundness proof never invents a move, which is why Proposition 5.1 needs its hypothesis, `TakesWinsOn`.

Three executable checkers are proved sound: `check`, which replays a table against all reply sequences (`check_sound`); the forced checker, generic over strategies (`checkForcedS_winsWith`, Proposition 5.1); and `refute` (`refute_sound`, Proposition 4.2). Each proof is an induction over the checker's recursion, whose guards supply the premises of the constructors, so soundness never inspects the strategy. The $5 \times 5$ table is `fiveTable` and Proposition 4.3 is `p1_wins_five`.

The rule is the definition `plusMove`; the file `Plus/Rule.lean`, which defines it with the abstraction and the line search, is about 250 lines. Hypothesis (i) of Theorem 6.2 for the rule is `plusMove_takesWins` and hypothesis (ii) is `plusMove_agree`; both are proved by unfolding the definition, and neither is an evaluation. The simulation theorem is `simulate`, and Theorem 6.1 is `plus_wins_with_of_seventeen_le`, whose body `plus_uniform_aux` dispatches the cases (a) to (e) to `simulate` with the run on the anchor board (`plus_forced_anchor`) and the three families `farAnchorsSafe`, `exactAnchorsSafe`, and `farNearAnchorsSafe`. Lemma 2.3 is not a separate Lean theorem; the containments it gives are proved where they are used: `three_corners_confine` and `square_filter_confine` confine a threat square to the region, and `escape_pick` and `escape_farm` confine a square with two region corners. The hidden corner of a pair is computed by `farmKey`, which the abstraction and the anchored runs use; the Lean identifiers call the hidden corner a *farm* (`farmKey`, `Key.farm`) and the cell of (G4) `hot`. A cell `farmKey` returns lies in $B_6$ and is a corner of a square through the pair (`farmKey_corners`), and the second claim of Lemma 2.5 is `escape_farm`: for a square as in the lemma, `farmKey` returns $b$. Hypothesis (iv) of Theorem 6.2 is stated through `farmKey`: $g$ is no cell and `farmKey` returns none, or `farmKey` returns $g$. The uniqueness claim is not

needed by the formal proof, which reads the one value `farmKey` returns. Lemma 2.2 is not used by the formal proof, which checks the lines directly. Theorem 1.2 is `plus_wins_with` and Theorem 1.1 is `outcome`, whose first half is `p1_wins_iff`.

| $n$ | pairs | time (s) | $n$ | pairs | time (s) |
|---|---|---|---|---|---|
| 5 | full tree | 1.5 | 12 | 20,163 | 10 |
| 6 | 1,155 | 0.9 | 13 | 27,888 | 16 |
| 7 | 2,208 | 1.3 | 14 | 37,635 | 24 |
| 8 | 3,843 | 1.9 | 15 | 49,728 | 36 |
| 9 | 6,240 | 2.9 | 16 | 64,515 | 52 |
| 10 | 9,603 | 4.4 | 17 | 82,368 | 72 |
| 11 | 14,160 | 6.8 | 23 | 277,728 | 499 |

Table 3. The runs of the checkers, timed with the checker executable on a MacBook Pro (16-inch, 2024) with an Apple M4 Max chip and 36 GB of memory. At 5 the enumerating checker walks the full tree of the table; from 6 up the forced checker runs the rule, with the number of reply pairs at the root. The run at 23 is the run on the anchor board; the 176 anchored runs take 0.34 seconds together.

Concrete boards are discharged by native evaluation (`native_decide`): the refutations at 1 to 4, the $5 \times 5$ run, one evaluation for the rule's runs at sizes 6 to 16, one for 17, one for the anchor board, and one per anchored family. The axioms of `outcome` and `plus_wins_with` are therefore `propext`, `Classical.choice`, `Quot.sound`, and one instance of Lean's evaluation axiom per certificate. The kernel checks the soundness theorems; the evaluations are performed by the Lean compiler and runtime, which are therefore trusted, the usual trust base of computer-assisted proofs in Lean. The whole development builds in about 17 minutes on the machine of Table 3, almost all of it the evaluations.

The Lean development and the Python code, with the $5 \times 5$ solver, the implementation of the rule, and the scripts that produce the numbers and board diagrams of this paper, accompany the paper as ancillary material and are at https://github.com/sean-reid/square-achievement, tag `v1.0.0`. The development builds with Lean 4.33.1 and mathlib at its tag `v4.33.1`.

## 8. Open questions

The rule wins by ply 17. What is the least ply by which the first player can force a win on $B_n$, and is there a rule of comparable length that attains it on every board? Other orderings of the lines, or other target shapes, might shorten the longest games at the cost of a longer statement.

The second move has seven exceptions. Is there an opening, or an opening with a fixed second move, from which a rule of this kind wins with none? The search of Section 3 did not vary the opening.

Whether a short rule exists for $B_5$, where the table needs quiet moves, is open.

Base 16 would serve Lemma 2.5 (Remark 2.6); whether the rule still wins when it reads only $B_{16}$ was not checked. Whether a rule of this kind can play inside a region smaller than $B_6$, or read less than the second player's stones inside its base board, for instance only those inside $B_6$ with a summary of the rest, is open.

## Acknowledgements

I thank Nicholas Tan for many games and for productive discussions that shaped the strategy presented here, and Anthony Gabriele for reviewing the presentation and content of the paper.

## References

1. K. Adler, *Is the square achievement game on the $n \times n$ grid resolved for $6 \leq n \leq 14$?*, MathOverflow, August 2026, Question 514742, https://mathoverflow.net/q/514742; accessed 2 October 2026.
2. ———, *square-achievement-game*, GitHub repository, https://github.com/keithadler/square-achievement-game, 2026, solvers, logs, and strategy certificates; accessed 2 October 2026.
3. R. Bacher and S. Eliahou, *Extremal binary matrices without constant 2-squares*, J. Comb. **1** (2010), no. 1, 77–100, https://doi.org/10.4310/JOC.2010.v1.n1.a6.
4. J. Beck, *Combinatorial games: Tic-Tac-Toe theory*, Encyclopedia of Mathematics and its Applications, vol. 114, Cambridge University Press, Cambridge, 2008, https://doi.org/10.1017/CBO9780511735202.
5. L. de Moura and S. Ullrich, *The Lean 4 theorem prover and programming language*, Automated Deduction – CADE 28 (A. Platzer and G. Sutcliffe, eds.), Lecture Notes in Computer Science, vol. 12699, Springer, Cham, 2021, https://doi.org/10.1007/978-3-030-79876-5_37, pp. 625–635.
6. M. Erickson, *Pearls of discrete mathematics*, CRC Press, Boca Raton, FL, 2010, The game is an exercise of Chapter 20, p. 171, with a hint on p. 252.
7. ———, *Square achievement game on a grid*, MathOverflow, June 2010, Question 29608, https://mathoverflow.net/q/29608; accessed 2 October 2026.
8. ———, *Square achievement game on an $n \times n$ grid*, Open Problem Garden, https://www.openproblemgarden.org/op/a_game_on_an_n_x_n_grid, 2010, accessed 2 October 2026.
9. F. Galliot and J. Sénizergues, *A unified convention for achievement positional games*, Preprint, arXiv:2503.18163v2 [cs.DM], 2026, Extended abstract in EuroComb'25, booklet of extended abstracts, pp. 473–478.
10. A. W. Hales and R. I. Jewett, *Regularity and positional games*, Trans. Amer. Math. Soc. **106** (1963), 222–229, https://doi.org/10.1090/S0002-9947-1963-0143712-1.
11. F. Harary, *Achievement and avoidance games for graphs*, Ann. Discrete Math. **13** (1982), 111–119, https://doi.org/10.1016/S0304-0208(08)73554-0.
12. T. Jenrich, *Guaranteed successful strategies for a square achievement game on an $n$ by $n$ grid*, Preprint, arXiv:1109.2341v4 [cs.DM], 2012, https://arxiv.org/abs/1109.2341v4.
13. T. Sirgedas, *Answer to "square achievement game on a grid"*, MathOverflow, July 2010, https://mathoverflow.net/a/33619; accessed 2 October 2026.
14. The mathlib Community, *The Lean mathematical library*, Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020) (J. Blanchette and C. Hriţcu, eds.), ACM, 2020, https://doi.org/10.1145/3372885.3373824, pp. 367–381.

Department of Mathematics and Statistics, San Jose State University, San Jose, CA 95192
*Email address*: sean.reid@sjsu.edu