Minesweeper is NP-Complete

Notes by Melissa Gymrek
Based on a paper by Richard Kayes 2000

Minesweeper

- Reducing 3SAT to generalized Minesweeper
- Reducing cSAT to well-know version of Minesweeper

General Minesweeper

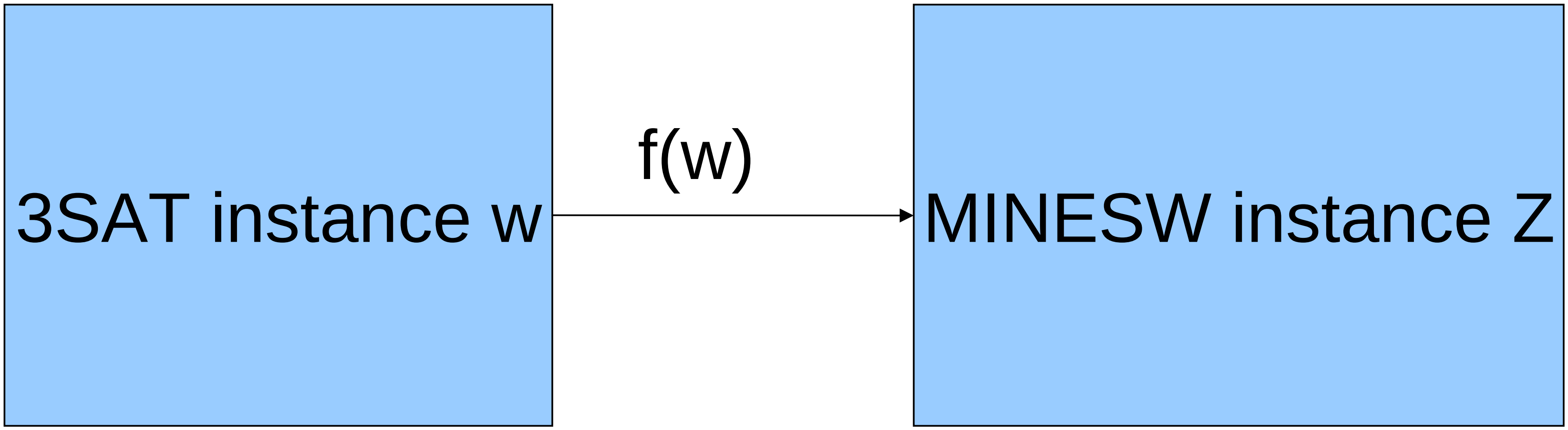
MINESWEEPER: { G, ξ |G is a graph and ξ is a partial integer labeling of G, and G can be filled with mines in such a way that any node v labeled m has exactly m neighboring nodes containing mines.}

Deciding if a graph is in the MINESWEEPER language is NP-complete:
- Polynomial time verification
- Reduce from 3SAT in polynomial time

Polynomial Time Verification

- For each node v labeled m:
  - Check that exactly m neighbors contain mines
  - O(E) time – clearly polynomial

Reduce from 3SAT



- Function f converts a 3SAT instance to a MINESW instance in polynomial time
- Z is satisfiable iff w is satisfiable

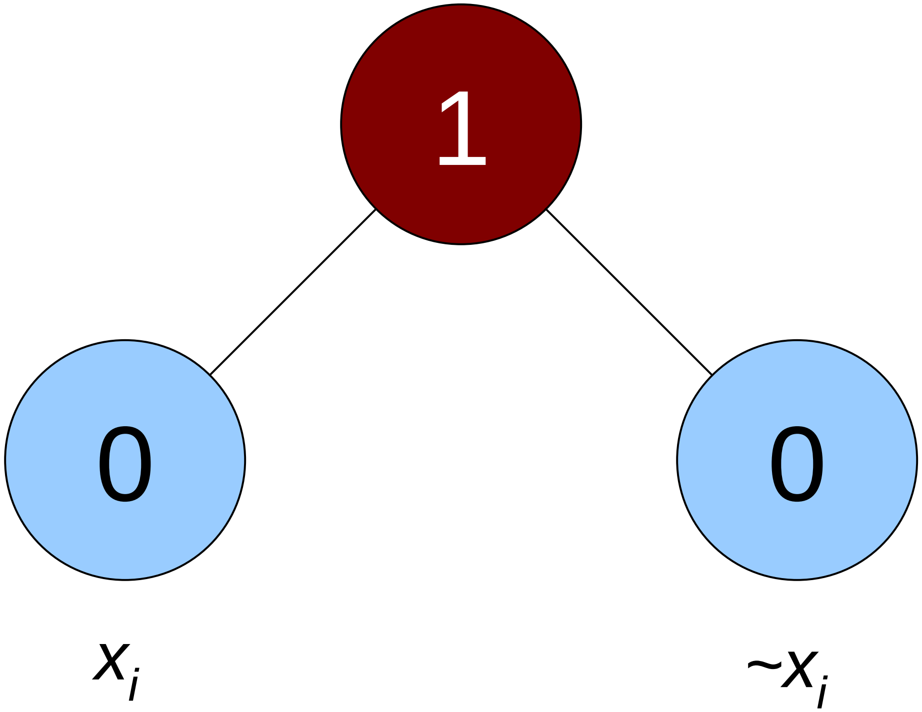
3SAT Review

Boolean 3CNF formula:
(A V B V C) ^ (~A V D V ~C) ^ …

N variables (A, B, C, D) in this instance
M clauses (here 2 clauses are shown)

Question: Is this boolean formula satisfiable?

3SAT → MINESWEEPER

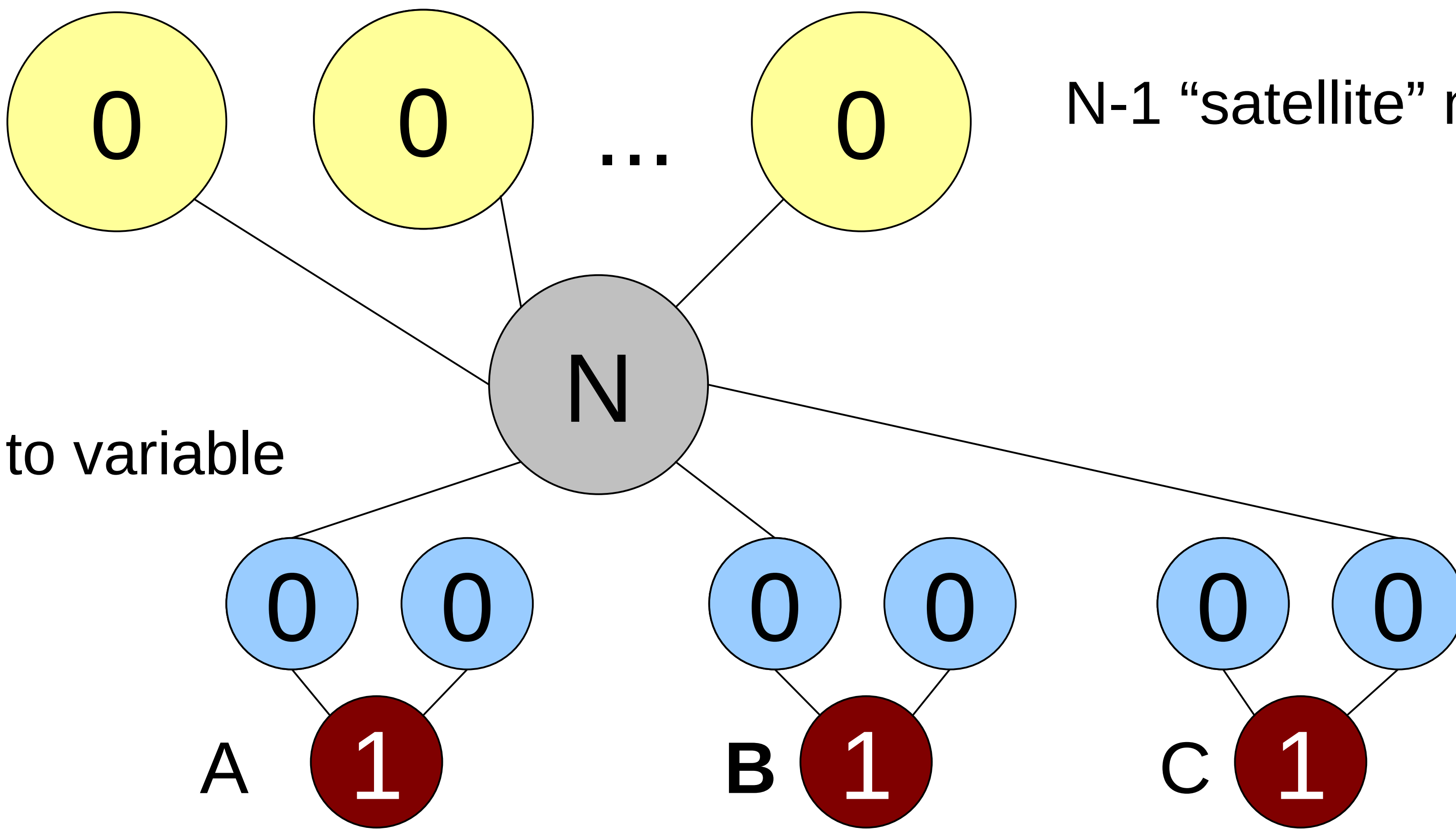


Make a gadget for each variable

3SAT → MINESWEEPER

For clause (A V B V ~C)

N-1 “satellite” nodes



Connect to variable gadgets

Make a gadget for each clause

3SAT → MINESWEEPER

- Conversion took polynomial time:
- 1 gadget for each of the N vars = O(N)
- 1 gadget for each of M clauses = O(MN)
- Total O(N(M+1)) time

Minesweeper as we know it

MINESWEEPER Problem: Given a rectangular grid partially marked with numbers and/or mines, some squares being left blank, determine whether there is some pattern of mines in the blank squares giving rise to the numbers seen.

Deciding if a graph is in the MINESWEEPER language is NP-complete:
- Polynomial time verification
- Reduce from cSAT in polynomial time

Wire

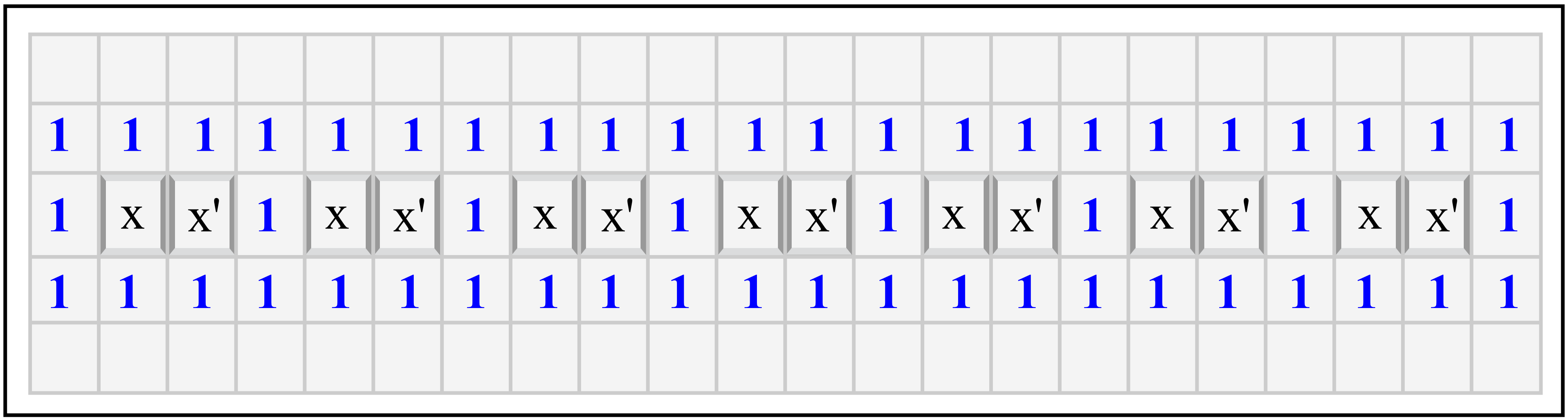


Image by MIT OpenCourseWare.

Either all the x's or all the x''s are mines. If it is the x's, we call it “true”, if the x''s, we call it “false”

Manipulating Wires

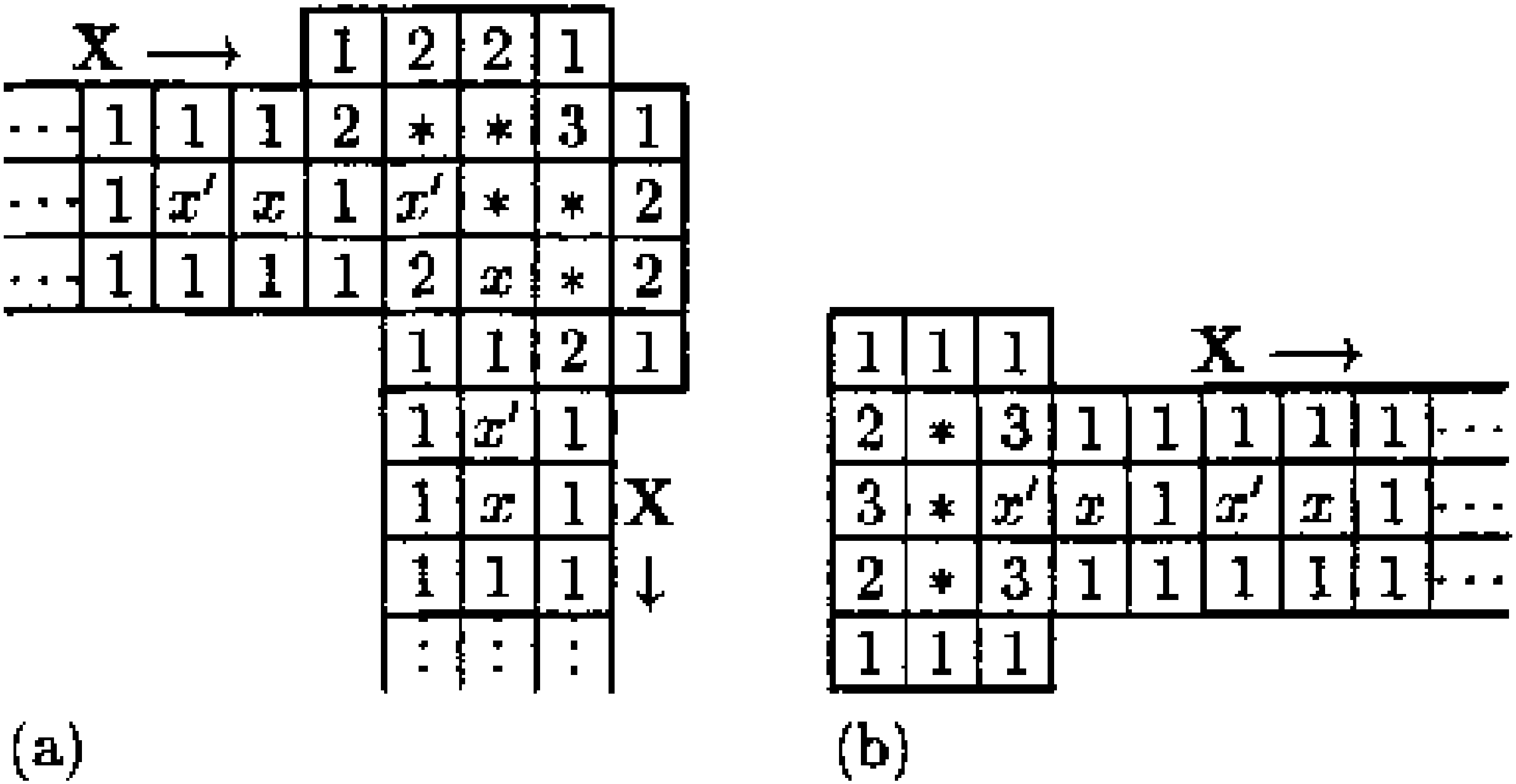


Figure 7. (a) A bent wire. (b) A terminated wire..

Kaye, Richard. "Minesweeper is NP-complete." *Mathematical Intelligencer* 22, no. 2 (2000): 9-15.

Minesweeper is NP-Complete, Kayes

Manipulating Wires

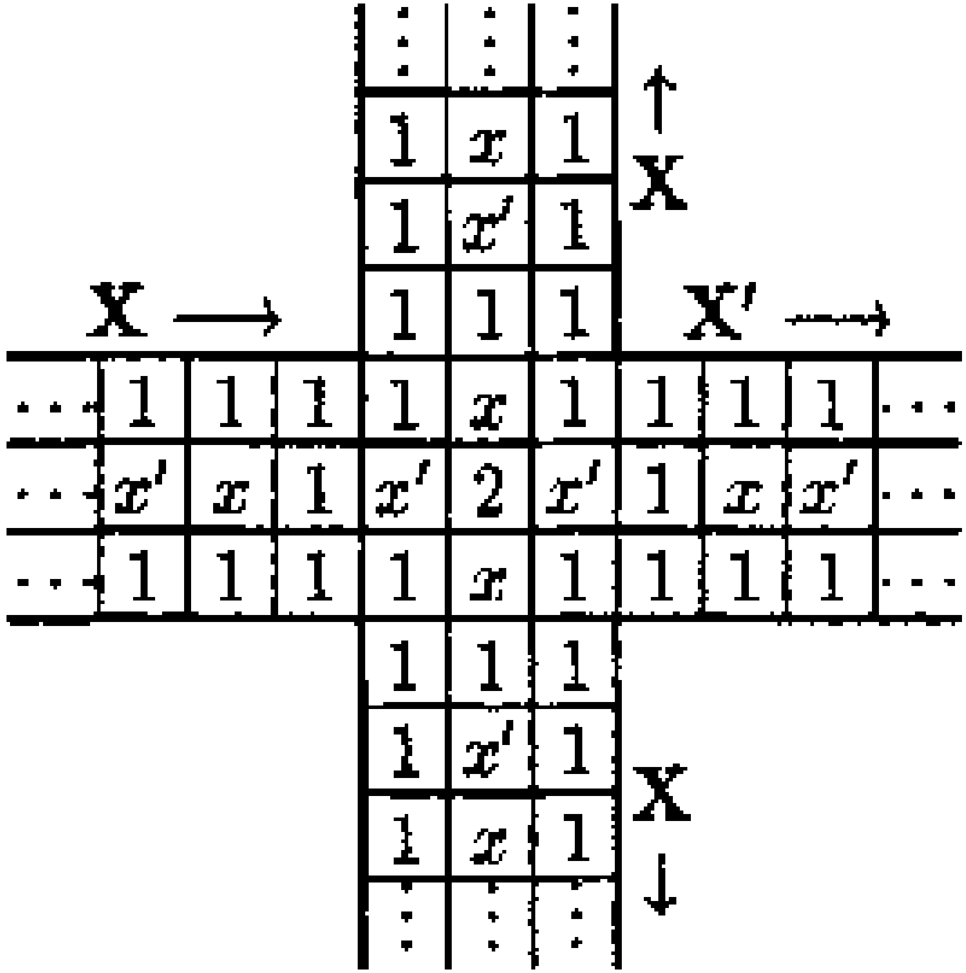


Figure 8. A three-way splitter.

Kaye, Richard. "Minesweeper is NP-complete." *Mathematical Intelligencer* 22, no. 2 (2000): 9-15.

Minesweeper is NP-Complete, Kayes

NOT gate

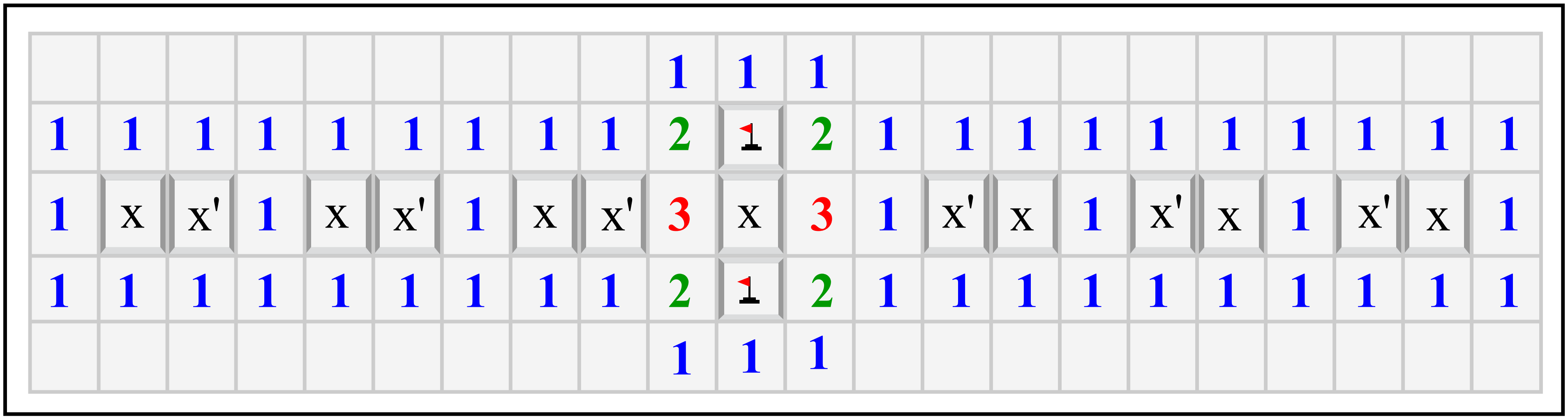


Image by MIT OpenCourseWare.

Inverts the sign of a wire

More gates

- We can now manipulate/invert wires

- Cross wires? First make planar XOR, then use XOR and three way splitter to cross wires

- We have NOT, and AND, universal!

More gates

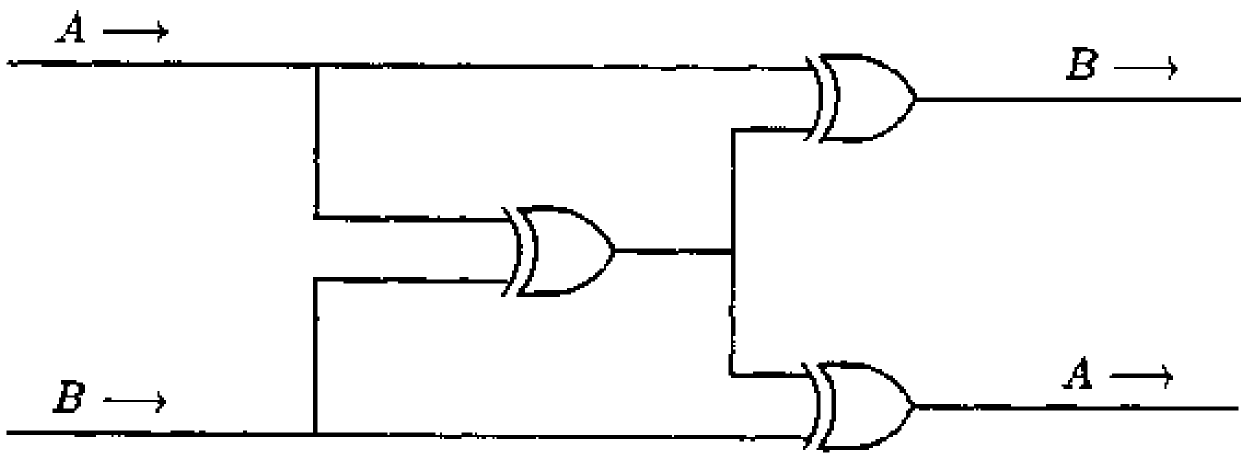


Figure 11. Crossing two wires with three XOR gates.

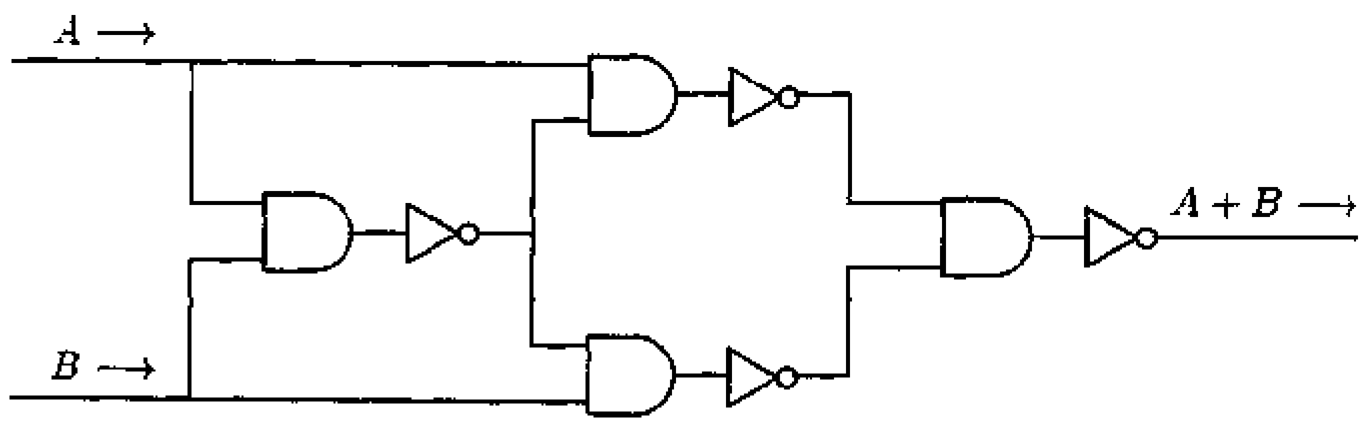


Figure 12. Making an XOR gate with AND and NOT gates.

Kaye, Richard. "Minesweeper is NP-complete." *Mathematical Intelligencer* 22, no. 2 (2000): 9-15.

Minesweeper is NP-Complete, Kayes

AND gate

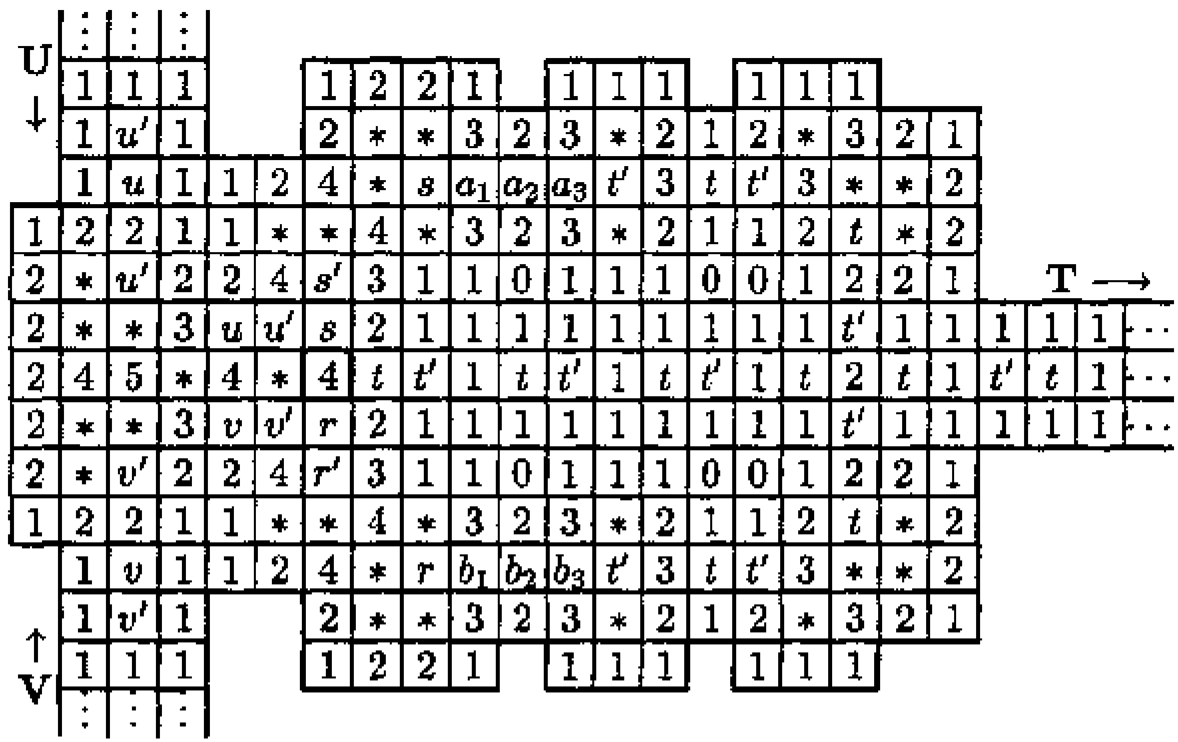


Figure 13. An AND gate.

Kaye, Richard. "Minesweeper is NP-complete." *Mathematical Intelligencer* 22, no. 2 (2000): 9-15.

NAND is universal!

- (A nand A) nand (B nand B) = A v B
- (A nand B) nand (A nand B) = A ^ B
- (A nand A) = ~A

Tetris is NP-complete

Ron Breukelaar, Erik Demaine,
Susan Hohenberger,
Hendrik Jan Hoogeboom,
Walter Kosters, David Liben-Nowell

published 2004

In Honor of your Intellectual Contribution to the Art of Tetris,

FOR PROVING NP-COMPLETENESS IN MAXIMIZATION OF LINES, TETRISES, PIECES PLAYED, OR MINIMIZATION OF SQUARE HEIGHT,

we masters of the Harvard Tetris Society hereby confer the title of

TETRIS MASTER

upon

Erik D. Demaine

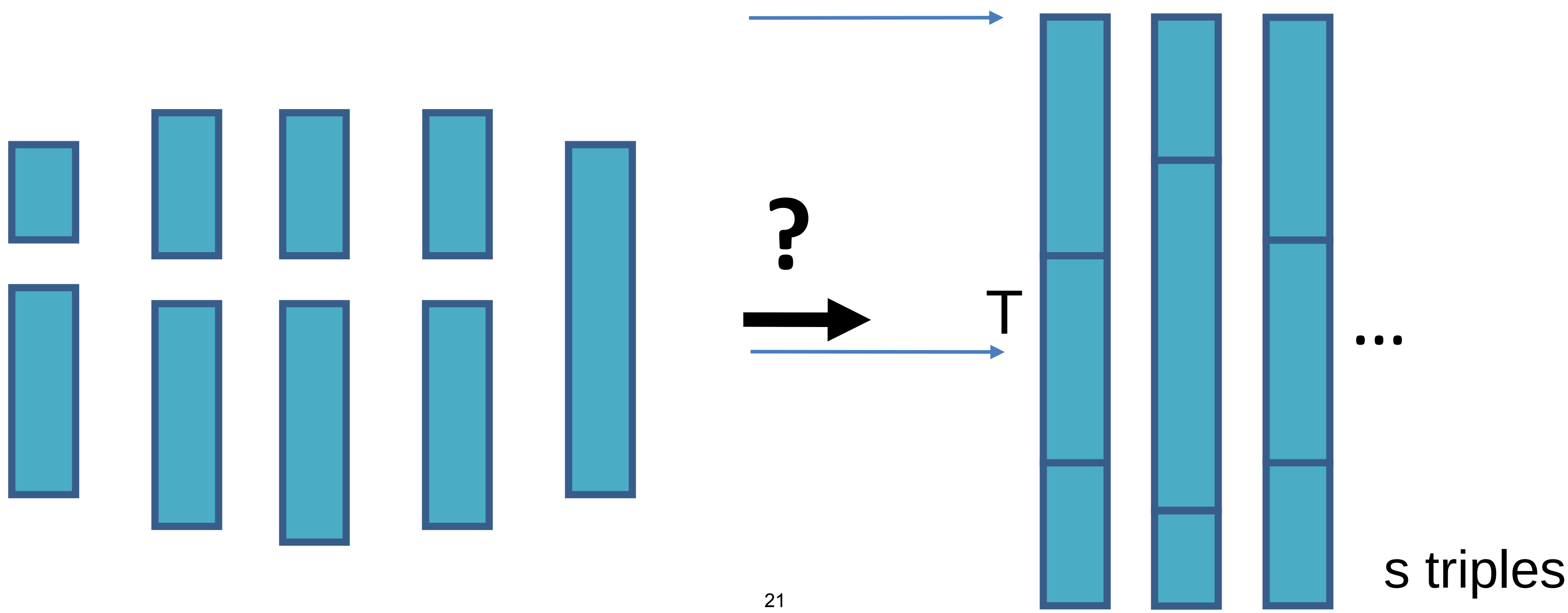
on the sixteenth day of the twelth month in the year 17 Anno Tetri (2002)

HTS President

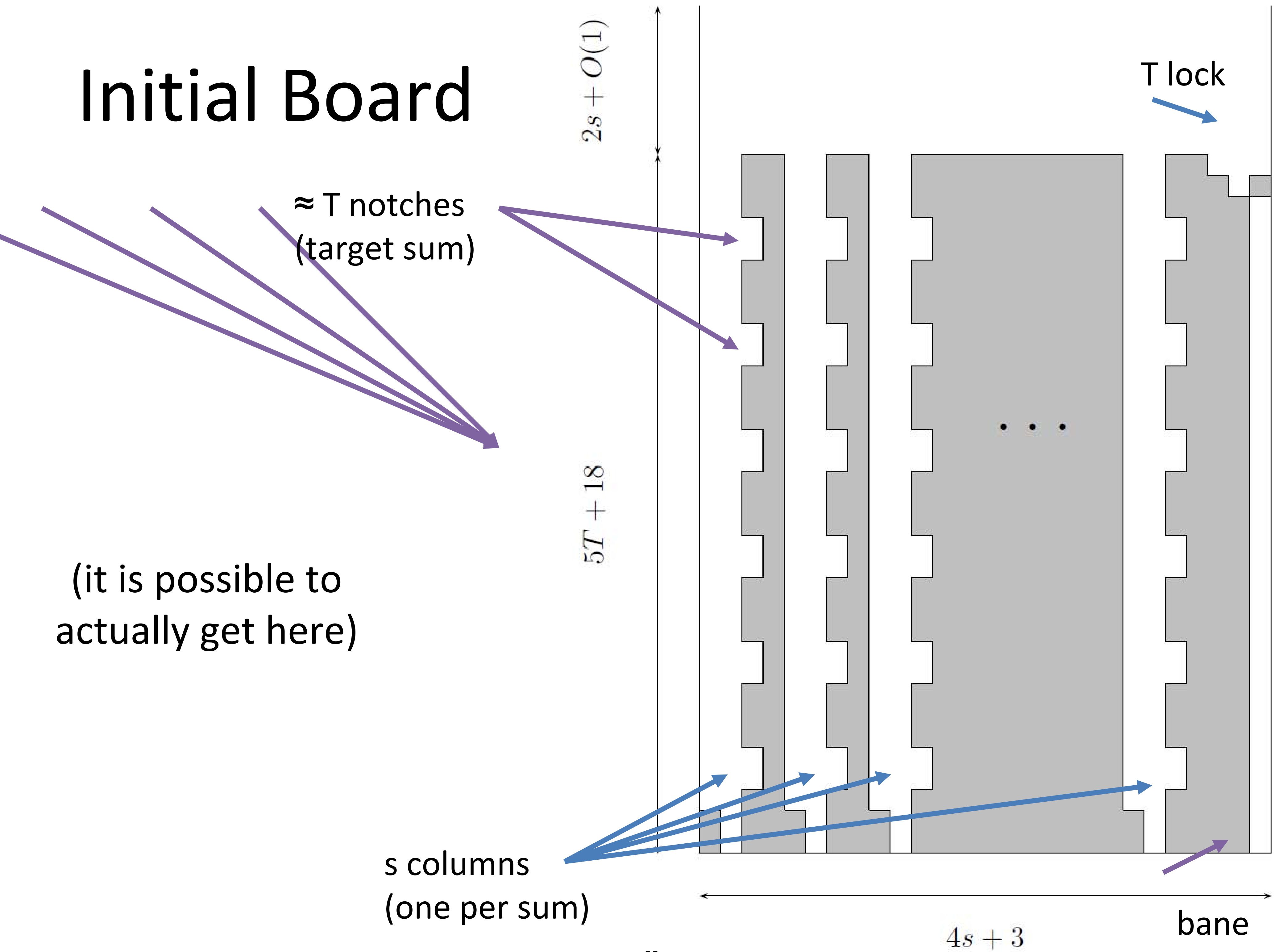
HTS Treasurer

3-Partition

- Given 3s integers $a_1, a_2, \ldots, a_{3s}$, can you partition into s triples with the same sum?
  - Know the sum must be $T = \sum a_i / s$
- This problem is **strongly NP-complete**: NP-complete even if a_i numbers are $s^{O(1)}$



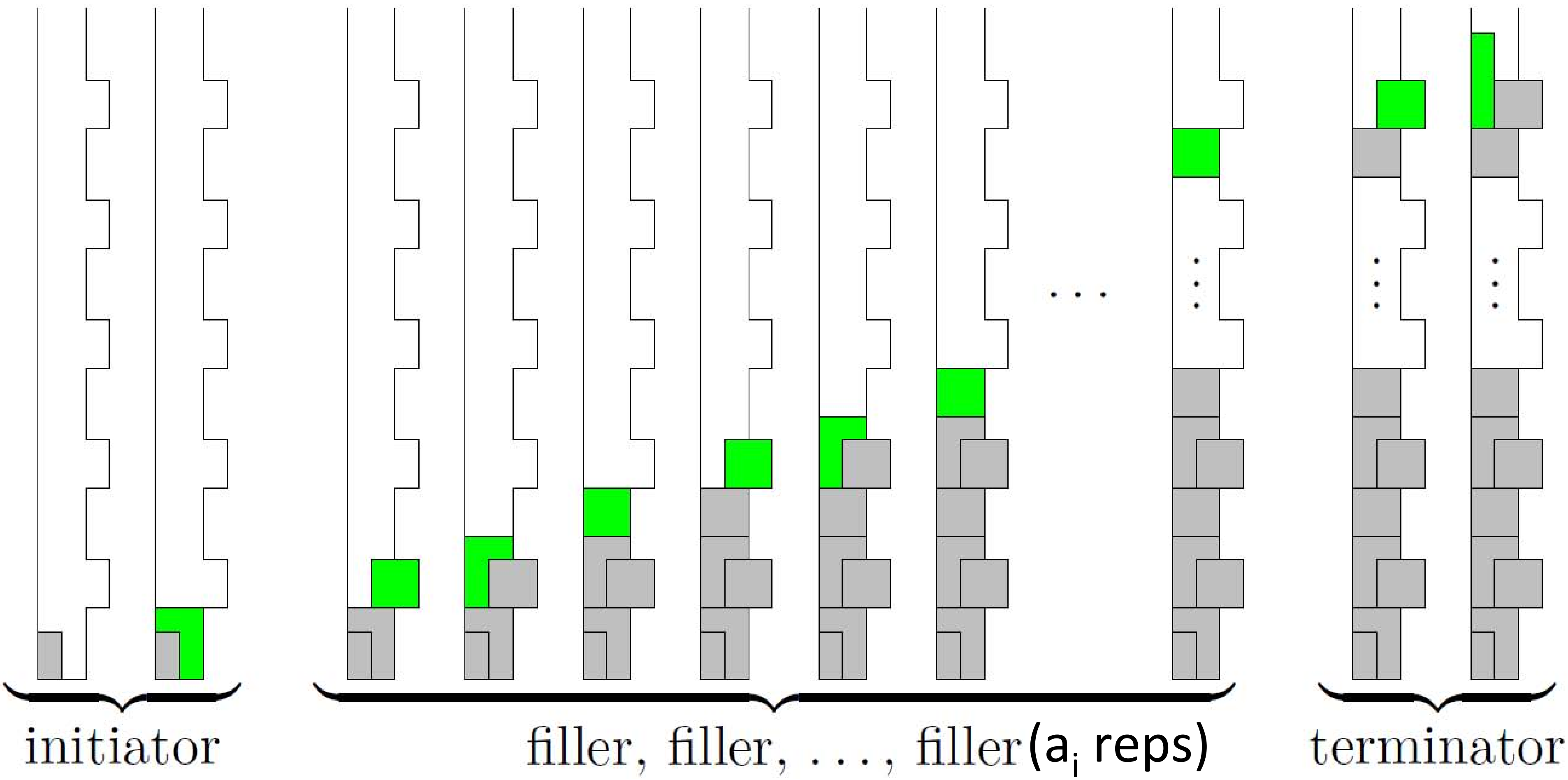
Initial Board



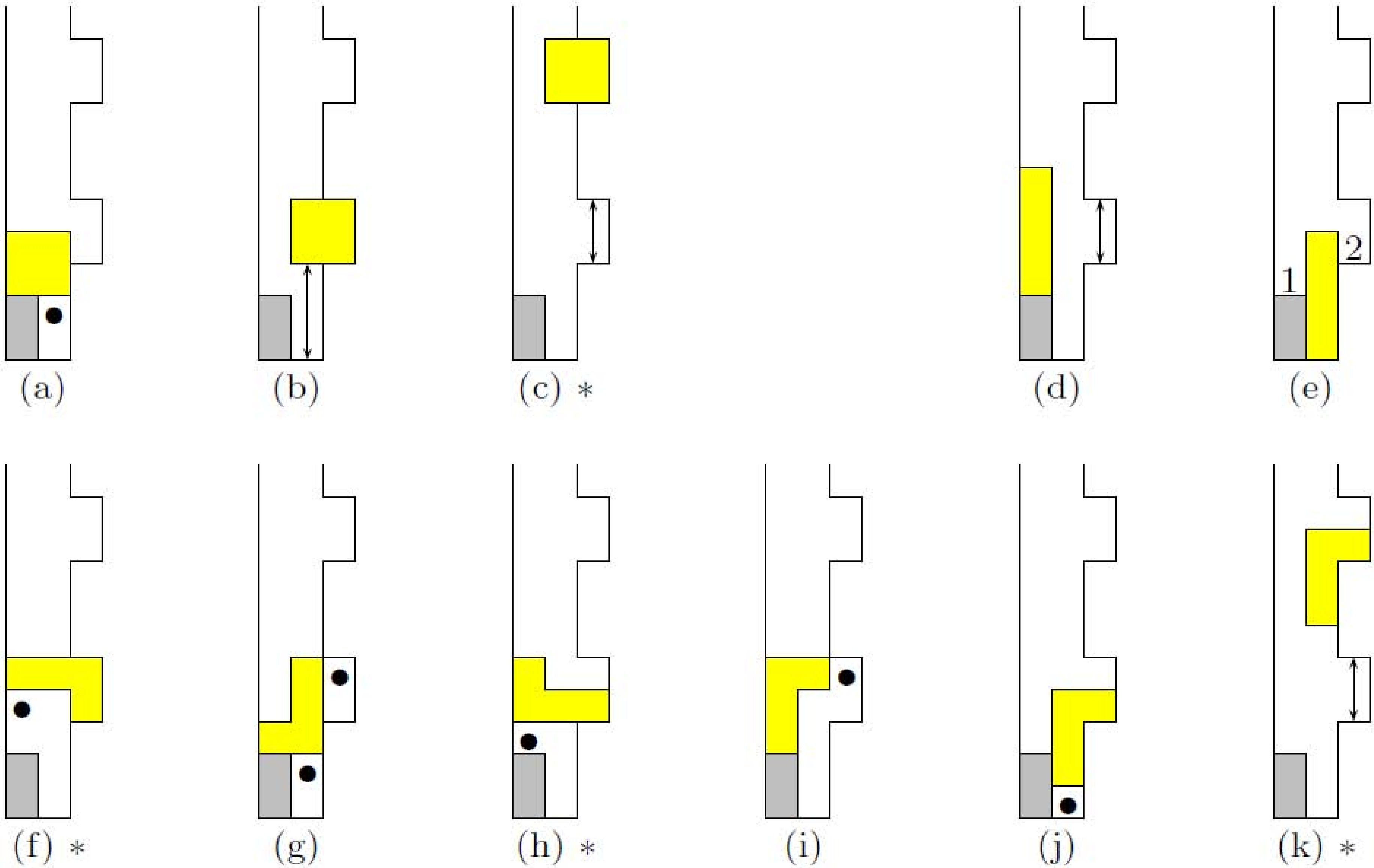
(it is possible to actually get here)

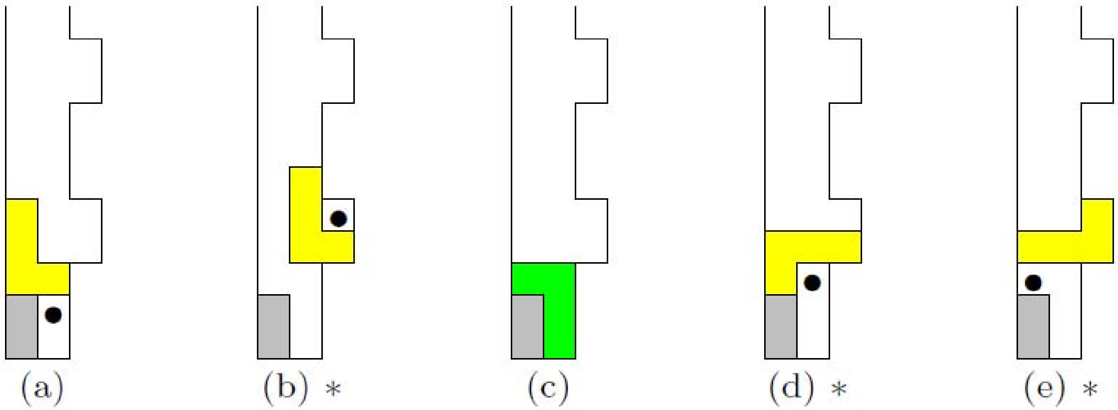
Piece Sequence

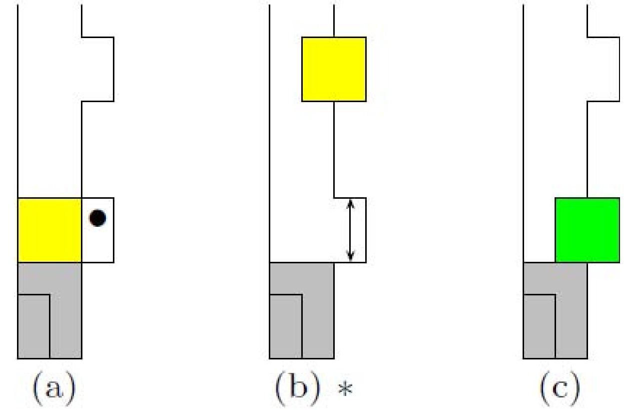
- For each input a_i:

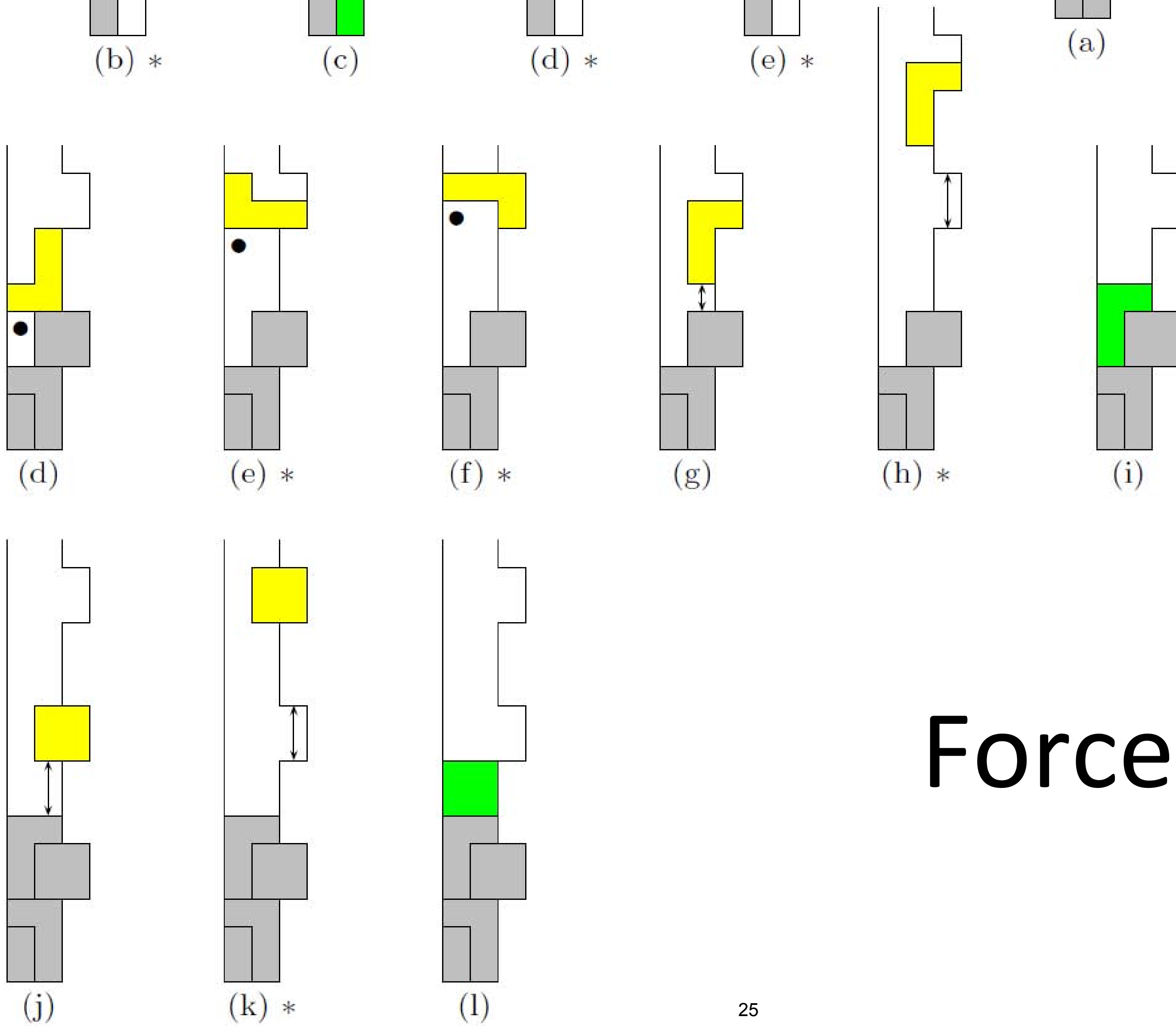


Failure to Launch



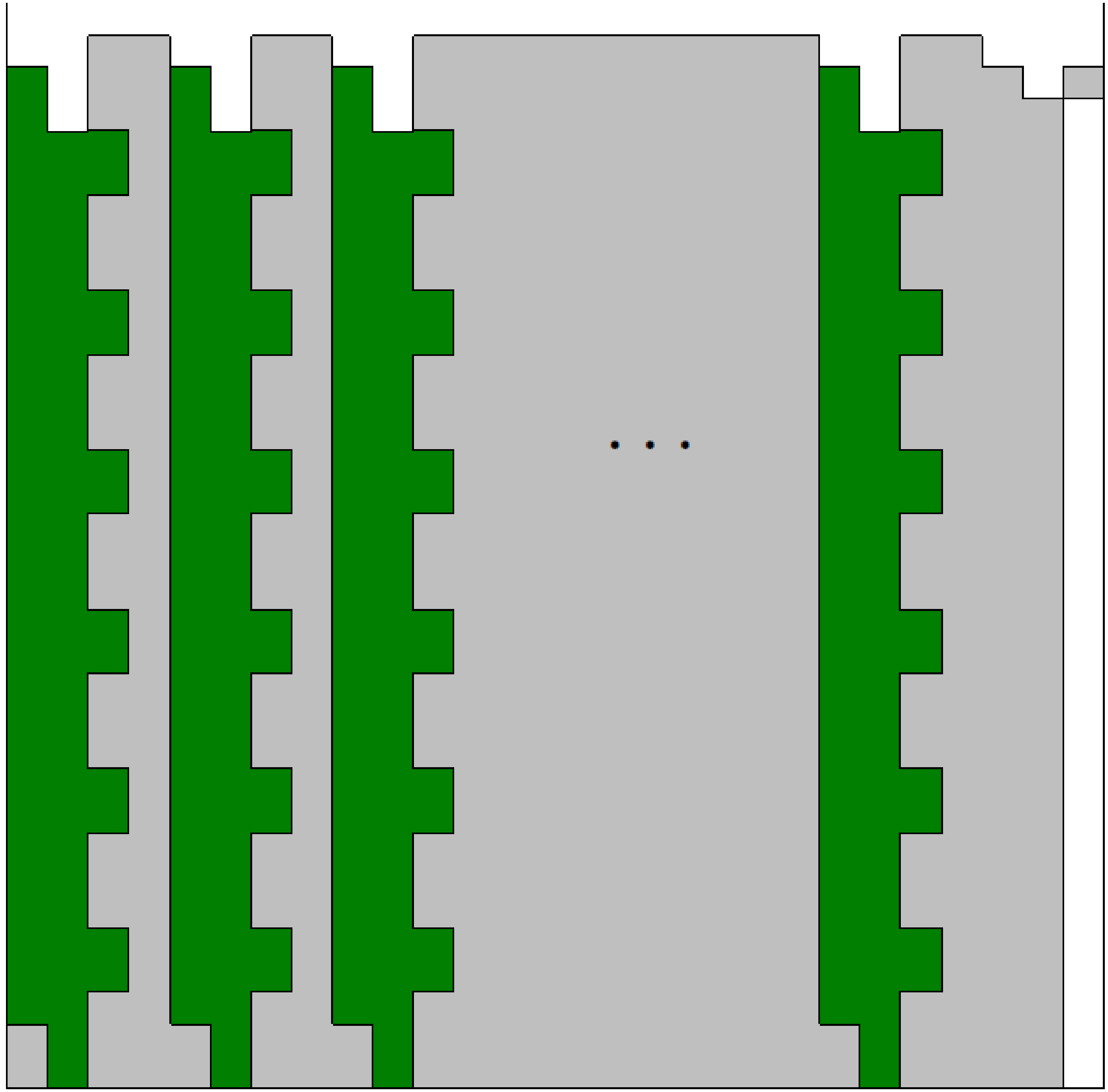




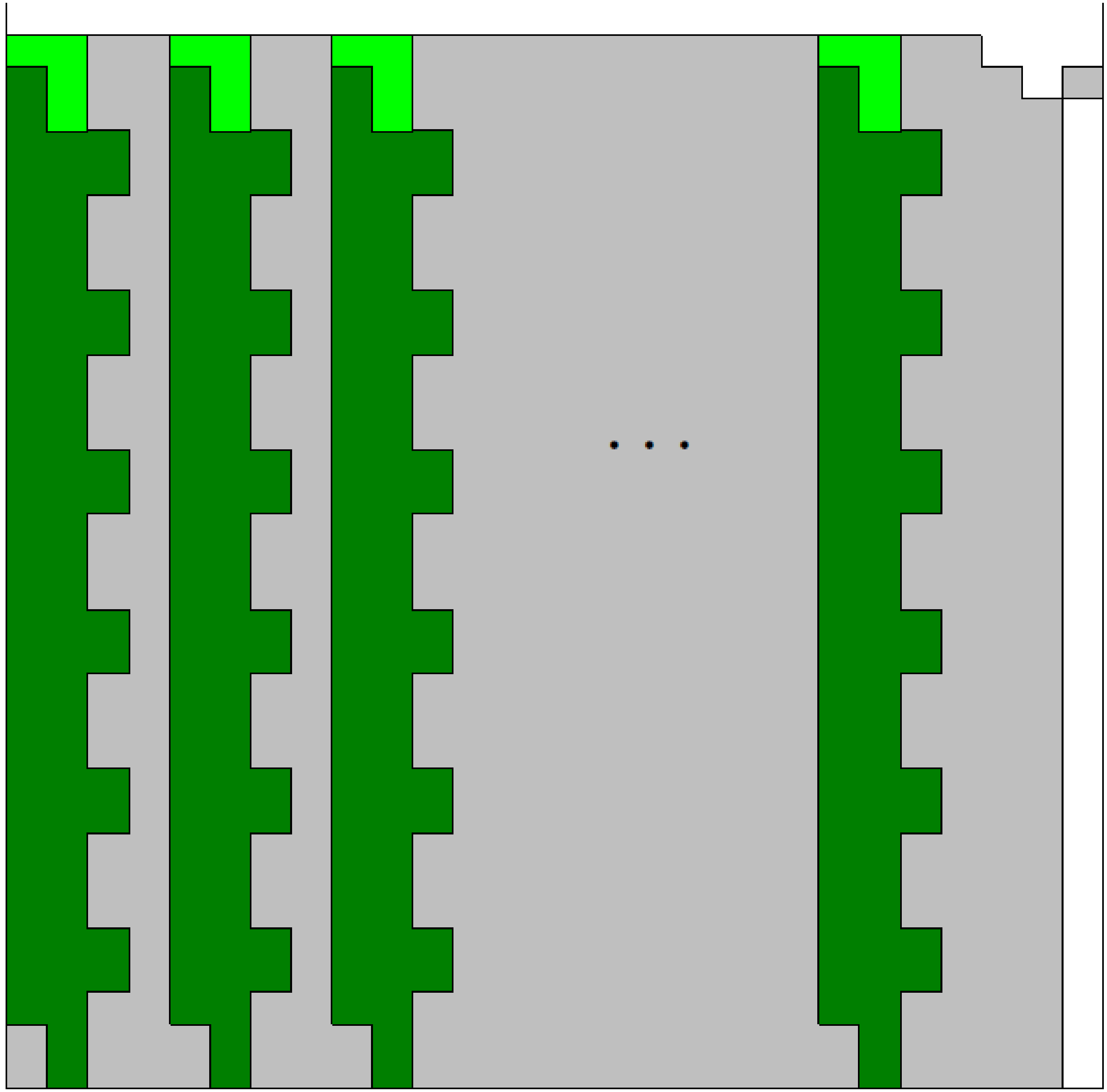


Forced Moves

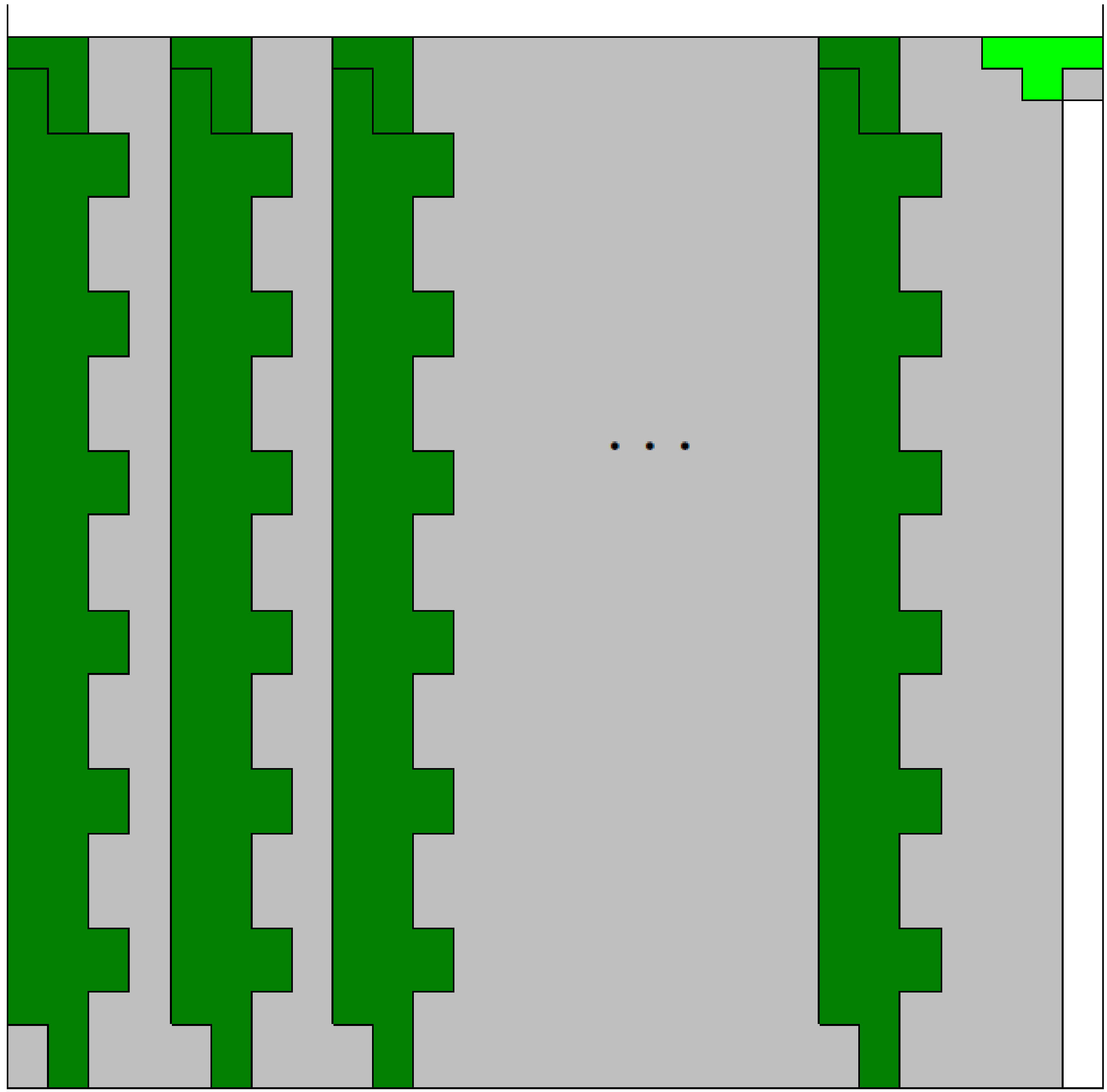
Finale Pieces



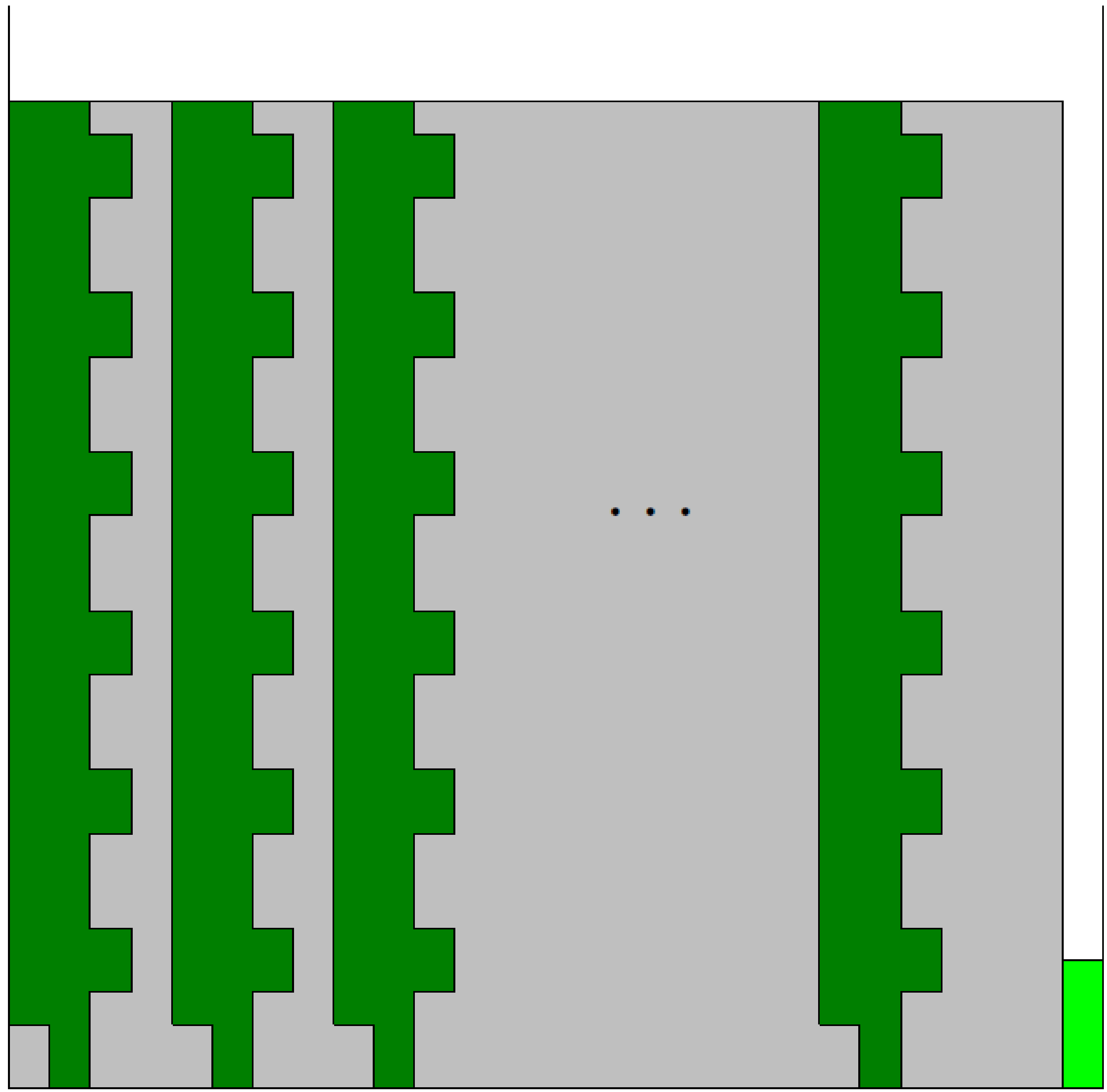
Finale Pieces



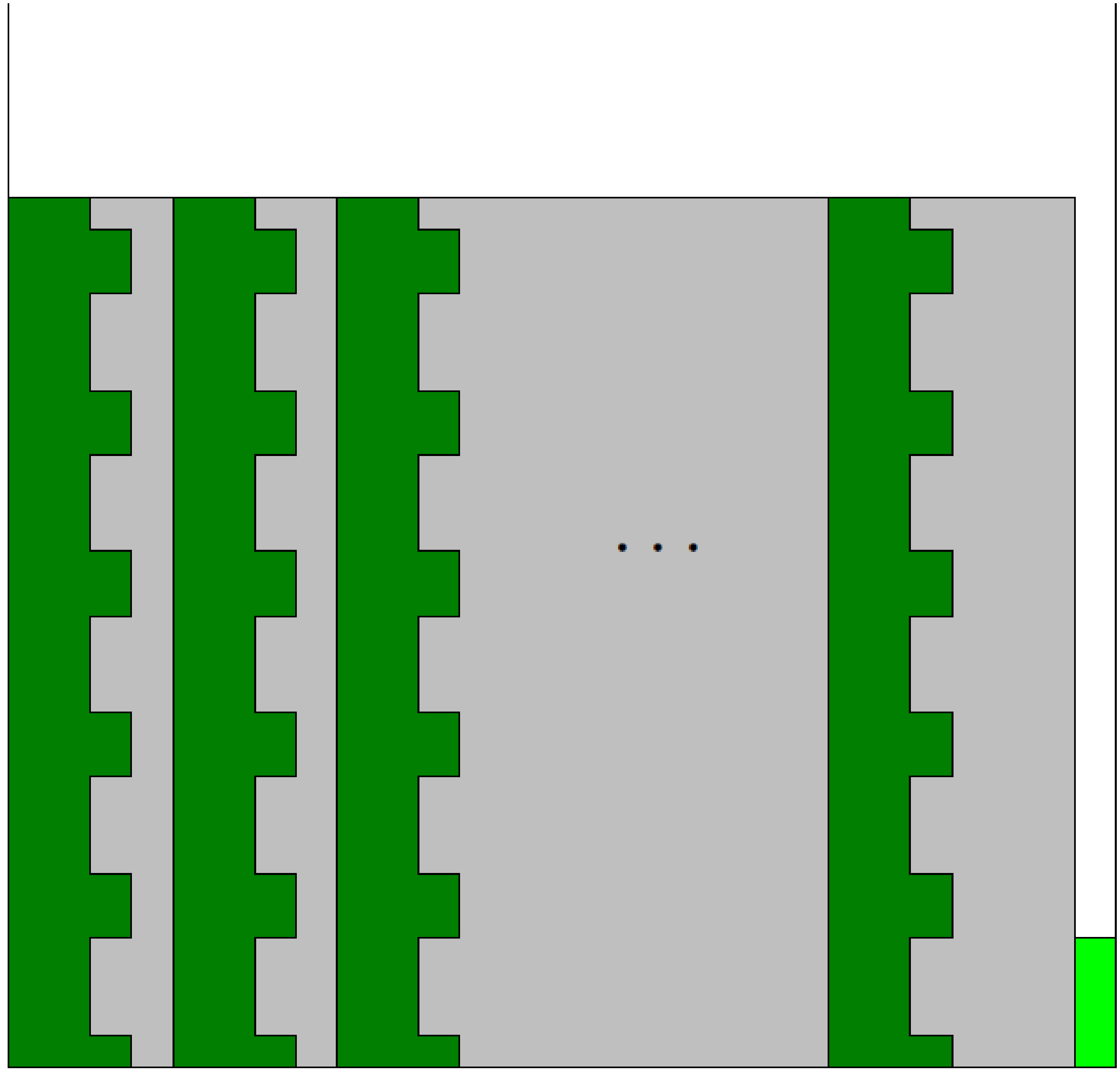
Finale Pieces



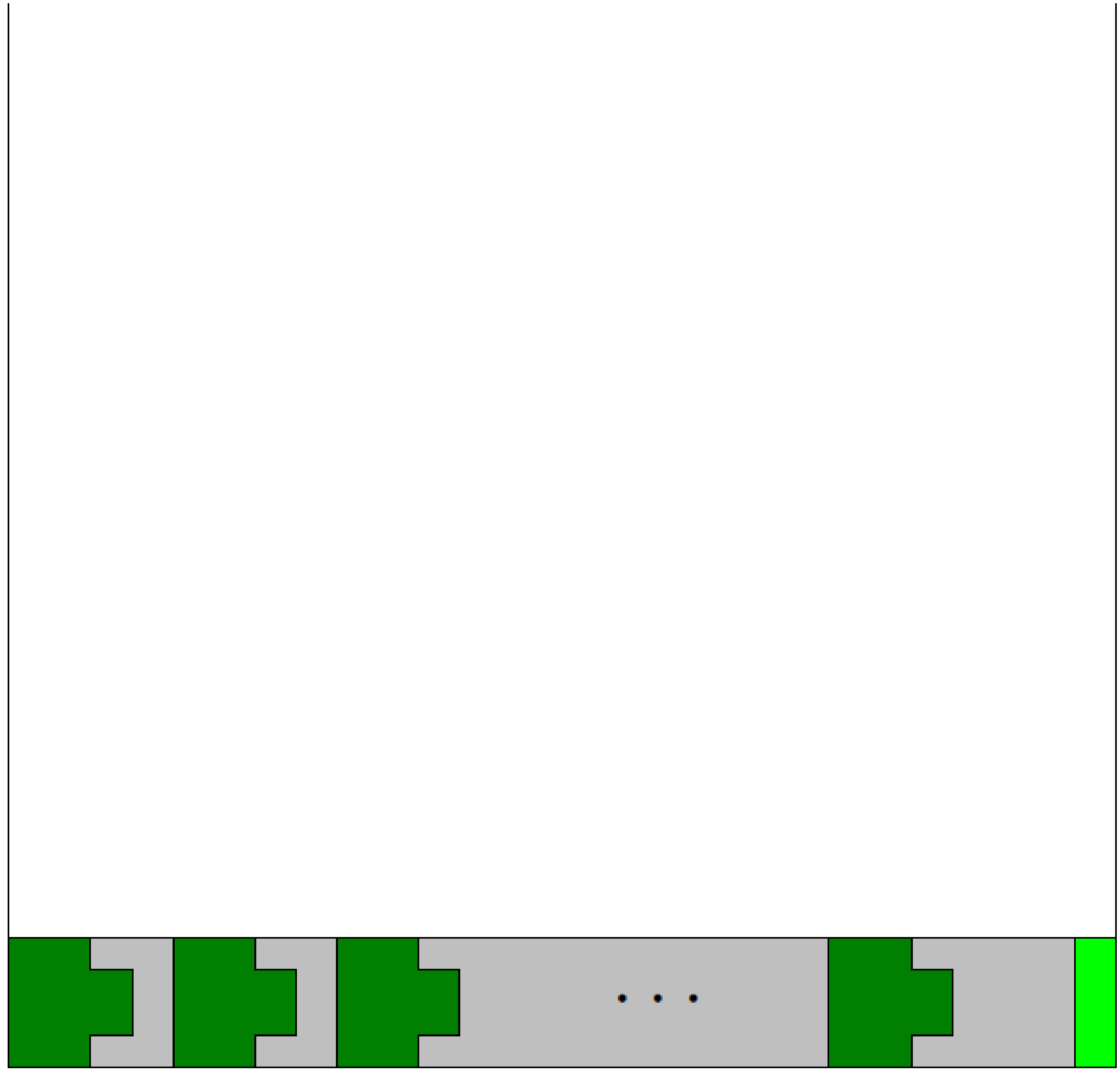
Finale
Pieces



Finale Pieces



Finale Pieces



Summary

- If there’s a 3-partition, can win Tetris:
  Get tons of lines, Tetrises, live forever, etc.
- If there’s no 3-partition, must lose Tetris:
  Die, no lines, no Tetrises, etc.

Open Problems

- What if the initial board is empty?
- What about Tetris with O(1) columns?
- What about Tetris with O(1) rows?
- What about restricted piece sets (e.g. just )?
- What if every move drops from high up (no last-minute slides)?
- Is two-player Tetris PSPACE-complete?
- What can we say about online (regular) Tetris?