

The density of oscillating agars in Conway's Game of Life

Results, methods and open problems. September 2026.

The problem

An **oscillating agar** is a pattern that fills the whole plane periodically and returns to itself after p generations. How large can its average density be?

- The conjectured maximum is **$1/2$** .
- For still lifes (period 1) this is a known theorem.
- For oscillators, the best published upper bound we found is **$1176/2087 \approx 0.563$** (LifeWiki, due to Keith Amling, 2024). Earlier bounds were $8/13$, $17/28$ and $320/559$.

Main results

1. Every oscillating agar, of any period, has density at most **$205847955/400000000 \approx 0.51462$** .

2. Every oscillating agar that avoids a fixed list of 769 small local patterns has density at most **$1/2$** .

3. Several lemmas proved by hand, including a short proof of the still-life bound, and a description of the only patterns that can still make the density exceed $1/2$.

The conjecture itself is still open.

Results labelled *computer-assisted* rely on the scripts described in Section 10. Nobody has yet checked them with an independent implementation.

Contents

1. Definitions
 2. Summary of results
 3. Results proved by hand
 4. The linear program
 5. Turning the LP into a proof
 6. Which patterns can make the density exceed $1/2$
 7. The two remaining families
 8. Agars with a small spatial period
 9. Approaches that did not work
 10. Reproducing the results
 11. Open problems
- Appendix: patterns

1. Definitions

Basic notions

- **Life** is the rule B3/S23 on the square grid. Each cell has 8 neighbours, labelled as in Figure 1.
- An **agar** is a pattern that repeats in both directions. Equivalently, it is a pattern on a finite torus. Every agar repeats along some rectangle, so rectangular tori are enough.
- An **oscillating agar of period p** has phases $A_0, \dots, A_{p-1}$. Each phase is the Life image of the one before, and $A_p = A_0$. Oscillators whose period divides p are included.
- Its **average density** is the mean, over the p phases, of the fraction of live cells.

NW	N	NE
W	x	E
SW	S	SE

Figure 1. The eight neighbours of a cell x .

Cells and deaths

- The **degree** of a live cell is its number of live neighbours.
- A live cell with at least 4 live neighbours dies next generation. We call this an **overcrowding death**. A live cell with at most 1 live neighbour dies of **isolation**.
- In a period-2 oscillator every cell has one of four **types**: S (always alive), D (always dead), X (alive in phase A only) or Y (alive in phase B only). X and Y cells are called **rotor** cells.

■ always alive (S) ■ alive in phase A only (X) ■ alive in phase B only (Y) □ always dead (D)

Windows

- A **window** is the 3×3 block around a cell, recorded at time t and at time $t+1$. Larger $h \times w$ windows are defined the same way.
- Each cell of a window is in one of four states: dead at both times, alive then dead, dead then alive, or alive at both times.
- A window is **reachable** if some pattern at time t produces it under Life. Of the 262,144 possible 3×3 windows, exactly **87,851** are reachable.

The patterns that matter

- A **dying triangle** is a set of three cells in one 2×2 block that all die in the same generation. Each of them has at least two live neighbours inside the triangle, so they die of overcrowding.
- A **vanishing block** is a 2×2 block with at least 3 live cells at time t and no live cells at time $t+1$.
- A **culprit window** is a window centred on a cell of a vanishing block that has a *surviving* neighbour directly above, below, left or right of it, outside the block.

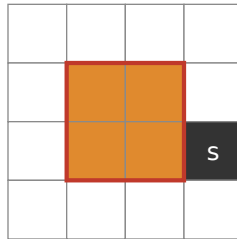


Figure 2. A vanishing block (orange: alive at t , dead at $t+1$). The cell s survives from t to $t+1$ and touches the block along an edge, so the windows centred on the block cells next to s are culprit windows.

2. Summary of results

Proved by hand

#	Result	Applies to
R1	Degree lemma: if every live cell has at most 3 live neighbours, the density is at most $1/2$	any periodic set
R2	Localisation: only cells with at least 4 live neighbours can end with positive charge	any configuration
R3	If no cell ever dies of overcrowding, the density is at most $1/2$	all periods
R4	Every live cell with at least 4 live neighbours has a neighbour that was just born	any evolution
R5	If no two rotor cells of the same type are adjacent, the density is at most $1/2$	period 2
R6	If live cells have k live neighbours on average, the density is at most $(k+4)/12$	any periodic set
R7	Splicing lemma: a smallest counterexample never repeats a pair of adjacent rows or columns	all periods
R8	Band-forcing lemma; the edge rows of family 2 cannot occur	period 2, strip forms
R9	Ring identity and linking lemmas for blinking blocks	period 2

Computer-assisted (rigorous if the code is correct)

#	Result	Type of check
R10	Every oscillating agar has density at most 0.5161291 (3×3 windows)	exact integer arithmetic
R11	Every oscillating agar has density at most $205847955/400000000 \approx 0.5146199$ (3×4 windows)	exact integer arithmetic
R12	Density $\leq 1/2 + o/10$, where o is the rate of overcrowding deaths; and $\leq 1/2 + 0.1213\tau$, where τ is the rate of dying-triangle cells (each up to about 10^{-6})	rounded dual
R13	Density $\leq 1/2$ if there are no dying triangles; also if there are no vanishing blocks; also if there are no culprit windows	exact rational arithmetic
R14	Main theorem: density $\leq 1/2$ if none of 769 listed window types occurs	exact rational arithmetic
R15	Density ≤ 0.500000010 if none of 3,350 listed 3×4 window types occurs	exact integer arithmetic
R16	Density $\leq 1/2$ for agars with a small vertical period (Section 8)	exact

3. Results proved by hand

3.1 The degree lemma (R1)

Lemma. Let Z be a periodic set of cells in which every cell has at most 3 neighbours in Z . Then Z has density at most $1/2$.

Stripes (full rows alternating with empty rows) have density exactly $1/2$, so the bound cannot be improved.

Proof. Call the cells of Z live. Give each live cell a charge of $+1/2$ and each dead cell a charge of $-1/2$. In a repeating block of n cells with L live cells the total charge is $L - n/2$, so we need to show that the total is at most 0.

Given a cell y and a direction (up, down, left or right), the two **side cells** of y are its neighbours on either side, perpendicular to that direction. Charge is moved by two rules, shown in Figure 3.

- **Rule 1.** Each live cell sends $1/4$ to each dead cell directly above, below, left or right of it, unless both side cells of that dead cell are live.
- **Rule 2.** A live cell x sends $1/4$ to a live neighbour w directly above, below, left or right of it, if both side cells of x are live and both side cells of w are dead.

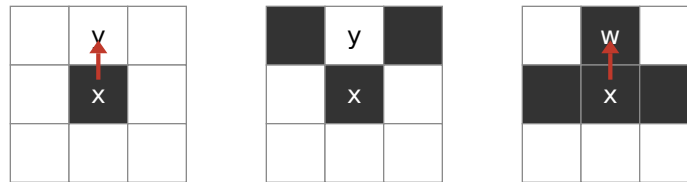


Figure 3. Left: Rule 1, the live cell x sends $1/4$ to the dead cell y above it. Middle: no transfer, because both side cells of y are live. Right: Rule 2, x sends $1/4$ to w , because x 's side cells are live and w 's side cells are dead.

No charge is created or destroyed, so the total stays the same. It remains to check that every cell ends with charge at most 0.

Dead cells. A dead cell only receives charge, and only by Rule 1.

- If 3 of its 4 direct neighbours are live, say N , W and E , then N sends nothing (its side cells W and E are live) while W and E each send $1/4$. It receives $1/2$.
- If all 4 are live, every transfer is blocked, so it receives nothing.
- If 2 or fewer are live, it receives at most $1/2$.

Live cells. A live cell x has at most 3 live neighbours. Let o be the number of live cells directly above, below, left or right of x . A dead direct neighbour of x can only be blocked if the two

diagonal neighbours of x beside it are both live.

- $o = 0$: x has at most 3 live diagonal neighbours, which can block at most 2 of the 4 transfers. So x sends at least $1/2$.
- $o = 1$, say W is live: at most one of the transfers to N, E, S is blocked, so x sends at least $1/2$. If W sends $1/4$ to x by Rule 2, then NW and SW are live, so NE and SE are dead and x sends $3/4$.
- $o = 2$: at most one diagonal neighbour is live, so nothing is blocked and x sends $1/2$.
- $o = 3$, say N, W and E : x has no other live neighbours, so it sends $1/4$ to S and, by Rule 2, $1/4$ to N .

So every cell ends at most 0, and the total is at most 0. ■

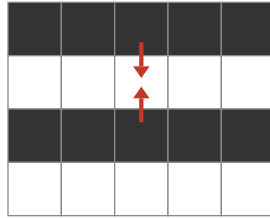


Figure 4. Stripes. Each live cell sends $1/4$ up and $1/4$ down and ends at 0. Each dead cell receives $1/4$ from above and $1/4$ from below and also ends at 0.

3.2 Consequences (R2 to R5)

- **R2.** The argument for a dead cell does not use degrees at all, and the argument for a live cell uses only that cell's own degree. So in any configuration, the only cells that can end with positive charge are live cells with at least 4 live neighbours.
- **R3.** In an oscillator, a live cell either survives (2 or 3 live neighbours), dies of isolation (at most 1) or dies of overcrowding (at least 4). If there are no overcrowding deaths, every phase satisfies the degree lemma.
- **R4.** A live cell with at least 4 live neighbours at time t had at most 3 at time $t-1$, because it was alive at t . So at least one of its neighbours was born at time t .
- **R5.** An X cell is reborn with exactly 3 live neighbours in phase B , all of type S or Y , so it has at most 3 S neighbours. To have at least 4 live neighbours in phase A it therefore needs an X neighbour. The same holds for Y cells.

3.3 Spectral bound (R6)

On a torus, the adjacency matrix of the 8-neighbour grid has eigenvalues $(1 + 2\cos \theta)(1 + 2\cos \phi) - 1$, which lie between -4 and 8 .

Let a set have density α , and suppose its cells have k neighbours in the set on average. Applying the quadratic form of the adjacency matrix to the centred indicator of the set gives $k - 8\alpha \geq -4(1 - \alpha)$, that is, $\alpha \leq (k + 4)/12$. In particular, an average of at most 2 live neighbours gives density at most $1/2$.

3.4 Splicing lemma (R7)

Take an oscillator on a torus. Suppose two pairs of adjacent rows are identical in every phase, possibly after a sideways shift. Cut the torus between the two pairs and reglue the pieces into two smaller tori (Figure 5).

- Every cell keeps the same neighbours, so both pieces are still oscillators.
- Their live cells add up to those of the original, so one piece has density at least that of the original.

So a smallest counterexample never repeats a pair of adjacent rows, or of adjacent columns. This bounds each dimension in terms of the other, but not by a fixed number.

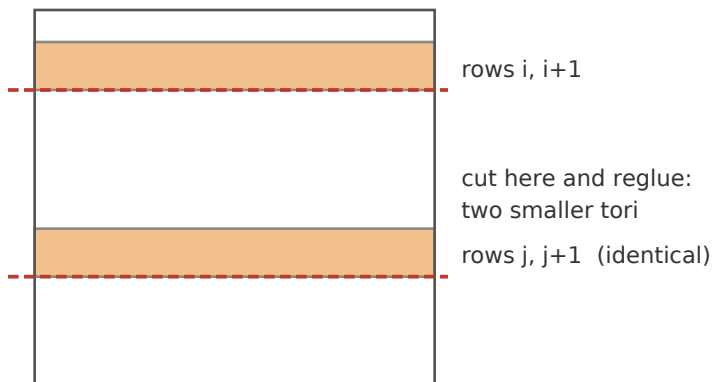


Figure 5. Splicing. If two pairs of adjacent rows agree, cutting at both places gives two smaller oscillators.

3.5 Lemmas about blinking blocks (R9, period 2)

A **blinking block** is a 2×2 block of X cells: alive in phase A, dead in phase B. Its **ring** is the 12 cells around it: 4 corner cells, each touching one block cell, and 8 edge cells, each touching two (Figure 6).

c	e	e	c
e			e
e			e
c	e	e	c

Figure 6. A blinking block (orange) and its ring: corner cells *c* and edge cells *e*.

Ring identity. Each block cell is reborn in phase A, so it has exactly 3 live neighbours in phase B. Those neighbours are of type S or Y, and all lie in the ring, since the block is empty in phase B. A corner cell counts for one block cell and an edge cell for two, so

$$c + 2e = 12, \text{ where } c \text{ and } e \text{ are the numbers of corner and edge cells of type S or Y.}$$

If no other X cells touch the block, the remaining ring cells are always dead. There are then at least 4 of them, and $(\text{number of S cells}) - (\text{number of D cells}) \leq 4$ in the ring. A small exhaustive search shows that 4 is attained.

Linking. An S cell touching two cells of a blinking X-block already has 2 live neighbours in phase A, and it must survive, so it has at most one more. Hence it can touch at most one X-block and at most one Y-block. If an S cell touches two cells of each of two separate blocks, one block is of type X and the other of type Y (Figure 7).

X	X	S	Y	Y
X	X		Y	Y

Figure 7. An S cell between an X-block and a Y-block. Two X-blocks in this position would give the S cell 4 live neighbours in phase A, and it would die.

4. The linear program

A single linear program (LP) gives an upper bound for oscillators of every period at once.

Variables

There is one variable $x_w \geq 0$ for each reachable window w : 87,851 variables for 3×3 windows and 2,175,813 for 3×4 windows.

Why every oscillator gives a solution

Take an oscillator of period p on a torus with n cells. Let x_w be the fraction of the $n \cdot p$ pairs (cell, time) at which the window is w . These numbers satisfy the following constraints.

- **Reachability.** Only reachable windows occur.
- **Spatial consistency.** For every pattern P , the windows whose left part is P and the windows whose right part is P have the same total frequency, because shifting by one column permutes the cells of the torus. The same holds for top and bottom parts.
- **Stationarity.** The patterns seen at time t and the patterns seen at time $t+1$ occur with the same frequencies, because $t+1$ runs through the same p phases as t .
- **Normalisation.** The x_w add up to 1.

The density of the oscillator is the total frequency of windows whose centre is alive at time t . None of this depends on p , so the maximum of the LP bounds the density of every oscillator.

Values

LP	Variables	Maximum
period 2 only, 3×3	66,320	16/31
period 2 only, 3×4	about 1.12 million	$88/171 \approx 0.51462$
all periods, 3×3	87,851	$16/31 \approx 0.5161290$
all periods, 3×4 (using symmetry)	547,016 groups of 2,175,813	≈ 0.51461982 (equal to 88/171 to 8 digits)

Symmetry

Reflecting left-right or up-down maps windows to windows and constraints to constraints (possibly with a change of sign). So there is an optimal solution that is unchanged by reflections, and windows can be grouped with their mirror images. This makes the LP up to 4 times smaller. The 3×3 solve went from about 38 seconds to about 3 seconds, and the 3×4 solve became practical.

5. Turning the LP into a proof

A solver's floating-point answer is not a proof. A set of multipliers checked in exact arithmetic is.

Weak duality

Choose any number u_r for each consistency or stationarity constraint. These constraints are all zero for a valid solution x , so

$$\text{density} = \sum_w x_w (c_w - (A^T u)_w) \leq \max_w (c_w - (A^T u)_w).$$

Here c_w is 1 if the centre of w is alive at time t and 0 otherwise. Any choice of u gives a valid bound, and checking it means computing a maximum over finitely many windows.

How the multipliers were checked

- **Rounded check** (R10, R11, R15). Take the multipliers from the solver, multiply by 10^8 , round to integers and compute the maximum in exact integer arithmetic. With symmetry, the multipliers are averaged over the four reflections.
- **Exact check** (R13, R14). Take an optimal basis from the solver HiGHS, compute the multipliers exactly as fractions with the library FLINT, and check every window with exact fractions. In every case the bound came out as exactly $1/2$.

6. Which patterns can make the density exceed $1/2$

Removing a class of windows from the LP and solving again shows whether that class is responsible for the gap between the LP bound and $1/2$.

Windows removed	LP maximum	Status
none	16/31	R10
windows centred on an overcrowding death	$1/2$	R3, by hand
dying triangles	$1/2$	exact (R13)
vanishing blocks	$1/2$	exact (R13)
culprit windows (2,480 types)	$1/2$	exact (R13)

6.1 The main theorem (R14)

Most culprit windows play no role. Removing only those that the current optimum uses, and repeating, gives:

Round	LP maximum	Windows removed
0	16/31	405 (family 1)
1	0.505279	364 (family 2)
2	$1/2$	none needed

Main theorem. Let E be the 769 window types removed in rounds 0 and 1. Every oscillating agar, of any period, in which no window of E occurs has average density at most $1/2$.

The list E depends on the solver: other runs gave lists of 458 and 464 types. Every run that ends with an exact bound of $1/2$ proves the theorem for its own list.

6.2 The same with 3×4 windows (R15)

Round	0	1	2	3
3×4 windows	0.514620	0.506944	0.503219	0.499999971
3×3 windows	0.516129	0.510634	0.502250	0.4999999998

This removed 3,350 window types and gave a checked bound of 0.500000010. These windows contain 736 different 3×3 culprit patterns, compared with 458 to 769 for 3×3 windows. So larger windows improve the bound for all agars, but they do not reduce the number of problem patterns.

7. The two remaining families

7.1 What the optimal LP solution looks like

The optimal LP solution is not an actual pattern. It is a list of windows with frequencies, each window consistent with its neighbours. Trying to fit these windows together shows:

- they fill 8×8 squares, but no 9×9 square;
- they fill strips up to 5 rows high of any length, but 6 rows only up to length 10, and 7 to 9 rows only up to length 8.

So the optimal solution describes a band that cannot be stacked. Two families of windows appear.

Family 1: the blinking-block band. Two rows of blinking 2×2 blocks, spaced 3 apart and alternating between phases A and B, with a row of always-alive cells between them (Figure 8). The band is 5 rows high, repeats every 6 columns and has density $8/15$.

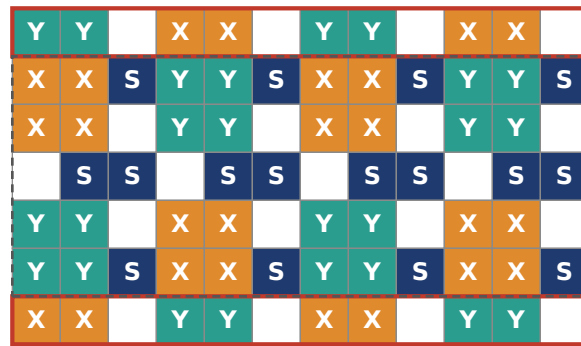
Family 2. A moving double stripe, in which two full rows die while full rows are born next to them, with half-filled rows at its edges (Figure 10).

7.2 Band-forcing lemma (R8)

Write each cell of the row just above the band as a pair (a_j, b_j) : $a_j = 1$ if it is alive in phase A and $b_j = 1$ if it is alive in phase B. Each cell in the top row of the band gives one condition:

```
cell (col 0, X):  b5 + b0 + b1 = 2      cell (col 1, X):  b0 + b1 + b2 = 2
cell (col 2, S):  a1 + a2 + a3 <= 1 and b1 + b2 + b3 <= 1
cell (col 3, Y):  a2 + a3 + a4 = 2      cell (col 4, Y):  a3 + a4 + a5 = 2
cell (col 5, S):  a4 + a5 + a0 <= 1 and b4 + b5 + b0 <= 1
```

The only solution is $a = (0,0,0,1,1,0)$ and $b = (1,1,0,0,0,0)$. So the row above the band must be **YY.XX.**, and by symmetry the row below must be **XX.YY.** (Figure 8). A computer check over 30 columns, without assuming that the pattern repeats, confirms this away from the ends of the band.



always alive (S)
 alive in phase A only (X)
 alive in phase B only (Y)
 always dead (D)

Figure 8. The band (dashed, two periods shown) and the rows it forces above and below (red).

Per 6 columns	Live cells (average over both phases)	Compared with 1/2
band (5 rows)	16 of 30	+1
forced row above	2 of 6	−1
forced row below	2 of 6	−1

So the band together with its two forced rows has density $20/42$, which is less than $1/2$. A real period-2 agar on a 6×14 torus consists of two such 7-row units and has density exactly $10/21$ (Appendix).

7.3 The edge rows of family 2 (R8)

Suppose the strip repeats every 2 columns. Then the row above an edge row alternates two values a and b (Figure 9). An edge cell marked “born” has no other live neighbours at time t , so it needs $a + b + a = 3$, which forces $a = b = 1$. But then the neighbouring cell, which must stay dead, also has 3 live neighbours and would be born. This is a contradiction.



Figure 9. Top: the row above, alternating a , b . Bottom: the edge row, with cells that must be born (teal) and a cell that must stay dead (x). Both see the same number of live cells above.

What is left is the ordinary moving double stripe, which has density exactly $1/2$ (Figure 10).

X	X	X	X
X	X	X	X
Y	Y	Y	Y
Y	Y	Y	Y

Figure 10. The moving double stripe on a 4-row torus: the X rows are alive in phase A and the Y rows in phase B.

7.4 Further computations

Question	Answer
Real period-2 agars of width 6 that contain the band	none of heights 7–13 or 15–18; best densities 0.476, 0.456 and 0.467 at heights 14, 19 and 20
A piece of band m columns long: largest possible excess of live cells nearby	+8 for $m = 6$, -1 for $m = 12$, -3 for $m = 18$, so long pieces are below $1/2$
Cells forced near a single blinking block with an S cell on one edge	none
Can one blinking block have S cells on two of its sides?	yes, for every combination tested
Can a culprit window sit at the centre of an 11×11 square of optimal windows?	no (9×9 is possible); this gives only a small improvement, to 0.516119
Is the optimal solution consistent with all pairwise correlations?	yes, up to 7×7 , so no argument based only on pairs of cells can rule it out

8. Agars with a small spatial period (R16)

If an agar repeats every h rows, it can be read as a sequence of columns, each column being one of finitely many states. Its density is at most $1/2$ exactly when a certain graph on pairs of columns has no cycle of positive weight, and this can be certified with integer labels on the nodes.

Period p	1	2	3	4	5	6
Vertical periods checked	≤ 6	≤ 5	≤ 3	≤ 2	≤ 2	1

The weaker two-layer statement (each cell of P has at most 3 neighbours in Q and vice versa, so $|P| + |Q|$ is at most the area) was also checked exactly for vertical period up to 4.

9. Approaches that did not work

Approach	Result	Reason
Charge rules or LPs using fixed-size windows	16/31 (3×3), ≈ 0.5146 (3×4)	the optimal solution is locally consistent up to 8×8
Larger windows only around dying triangles (4×4)	at least 0.5149	the problem patterns are more than 4 cells across
Requiring windows to have a long history	no change	all optimal windows have at least 4 generations of history
Slicing (choosing one phase per cell)	fails	a real agar has three adjacent rotor cells that cannot be separated
Moving excess charge one generation forward	fails	fails on about 12% of 5×5 patterns and on real agars
Sharing charge with nearby cells (radius 1)	fails	a computer search found real counterexamples
Column generation, warm starts	slower	the constraints are tightly linked
Local surgery on real oscillators	fails	the moving double stripe cannot be changed locally
Symmetrising the two-layer problem	fails	asymmetric pairs do better on odd-sized tori
Matching S cells to D cells along lines	fails	7 of 17 random period-2 agars have no suitable direction
Stability form of the degree lemma	fails	leads back to the charge-sharing problem
Reducing to a bounded period	no proof	cuts must match across the whole width

10. Reproducing the results

Setup

A fresh Python environment avoids conflicts between packages built for different NumPy versions.

```
conda create -n agar python=3.12 -y
conda activate agar
pip install numpy scipy highspy python-sat python-flint
```

Scripts

Script	What it does
life_agar_density.py	3×3 LP, removal of culprit windows, exact certificate (R14)
big_window_lp.py	LP for any window size: solve, remove culprits, checked bounds, symmetry (R10, R11, R15)
bound_16_31.py	short standalone computation of the 16/31 bound
band_forcing_check.py	computer check of the band-forcing lemma
strip_certificate.py	exact results for agars with a fixed spatial period (R16)

Main theorem (about 5 minutes on one core)

```
python life_agar_density.py certify

reachable windows: 87851
peel round 0: LP bound 0.516129; tight culprit windows 405
peel round 1: LP bound 0.505279; tight culprit windows 364
peel round 2: LP bound 0.500000; tight culprit windows 248
excluded 769 window types (saved to excluded_windows.pkl)
basis valid=True; exact system 6764 x 6764
EXACT BOUND = 1/2 (certificate saved to certificate.pkl)
```

Bounds for all agars

The 3×3 run takes about 1 minute. The 3×4 runs were done on a 16-thread machine, taking about 10 minutes per solve.

```
python big_window_lp.py certify --h 3 --w 3
RIGOROUS bound: density <= 51612905/100000000 = 0.516129050

python big_window_lp.py solve --h 3 --w 4 --sym --threads 16
RIGOROUS bound: density <= 205847955/400000000 = 0.514619887

python big_window_lp.py peel --h 3 --w 4 --sym --threads 16
peel round 3: bound 0.499999971 ... excluded 3350 window types
RIGOROUS bound: density <= 200000004/400000000 = 0.500000010
```

Other checks

- `python band_forcing_check.py` lists the possible types of each cell above a 30-column band. In columns 6 to 24 there is exactly one, repeating YY.XX.
- `python strip_certificate.py --width 5 --period 2` ends with PROVEN in under a minute.

11. Open problems

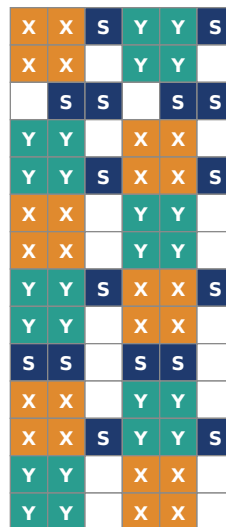
- **The two families.** By the main theorem, the conjecture is equivalent to this: in every oscillator, the 769 listed window types never push the density above $1/2$.
- **Arrangements.** Long pieces of band have density below $1/2$ once their forced rows are included. But a single blinking block forces nothing nearby and can have S cells on any side. What prevents short or irregular pieces from being packed densely?
- **Bounded period.** If a counterexample exists, must there be one with a bounded period in space or in time?
- **Independent checking.** The computer-assisted results R10 to R16 should be re-implemented independently.

Appendix: patterns

A real agar containing the band

A period-2 agar on a 6×14 torus, with density $10/21$. RLE for Golly, using a torus rule:

```
x = 6, y = 14, rule = B3/S23:T6,14
3o2bo$2o$b2ob2o$3b2o$2b4o$2o$2o$2b4o$3b2o$2ob2o$2o$3o2bo$3b2o$3b2o!
```



■ always alive (S) ■ alive in phase A only (X) ■ alive in phase B only (Y) □ always dead (D)

Figure 11. The same agar, one repeating block, by cell type.

The triangle agar

A period-2 agar on an 8×10 torus, with density 0.406. It contains three mutually adjacent rotor cells, each with three always-alive neighbours, which is the configuration that rules out slicing. Phase A, then phase B:

phase A	phase B
.00.....	.00.....
.....0.000	...000.000
0000.00.0.	0000.00.0.
...00.....	0.....
..00.00000	.00..00000
0.0.0..0..	0.0.0..0..
0.0.0.....	0.0.0.....
0..0.....0	0..0.....0