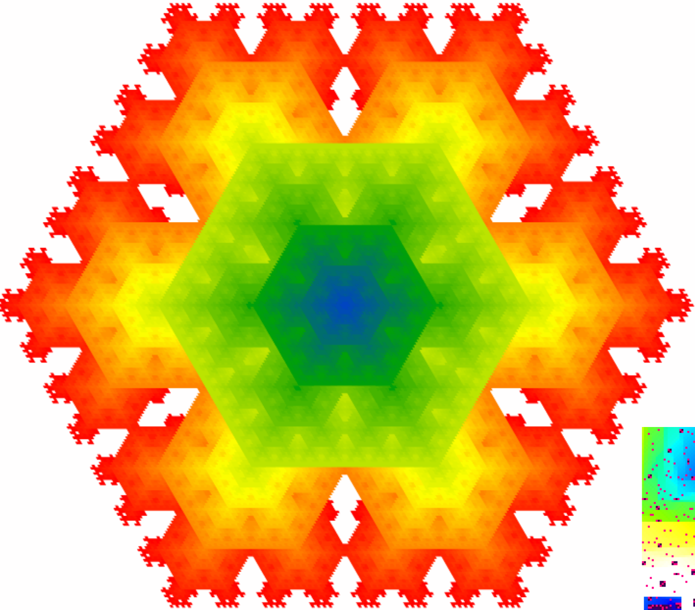
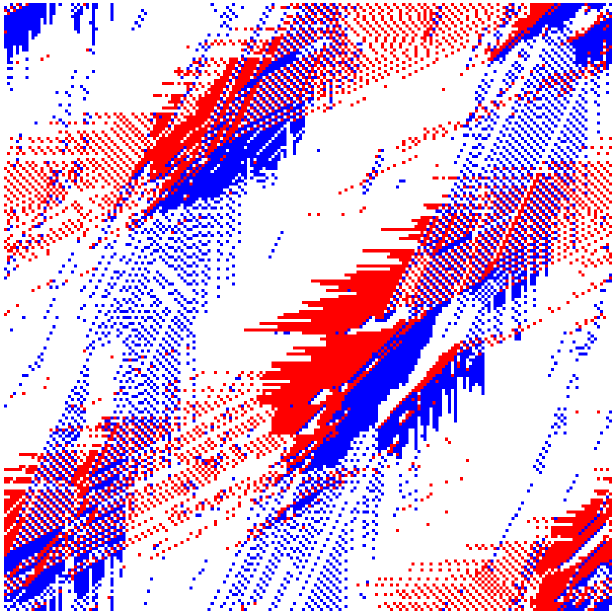
Is seeing believing?

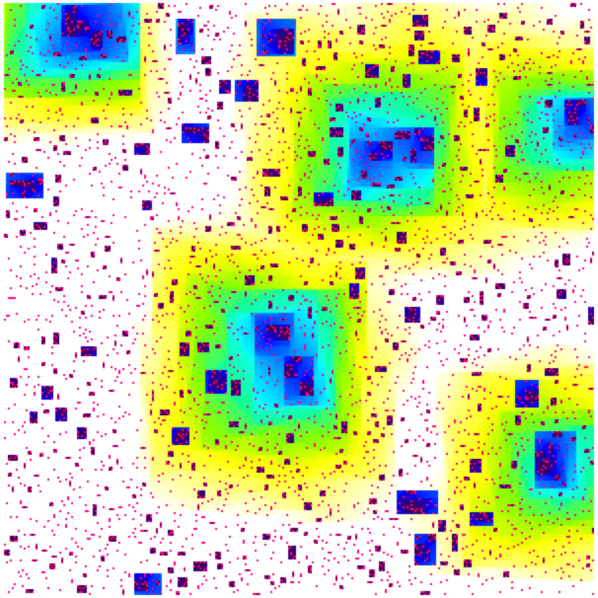
Cellular automata in theory and experiment

Alexander E. Holroyd, University of British Columbia



CUMC 2008





Cellular automaton:

- regular lattice of cells
- cell can be in finite number of possible states
  (e.g. alive/dead, full/empty)
- local rule for updating states

Idealized models of real-world systems

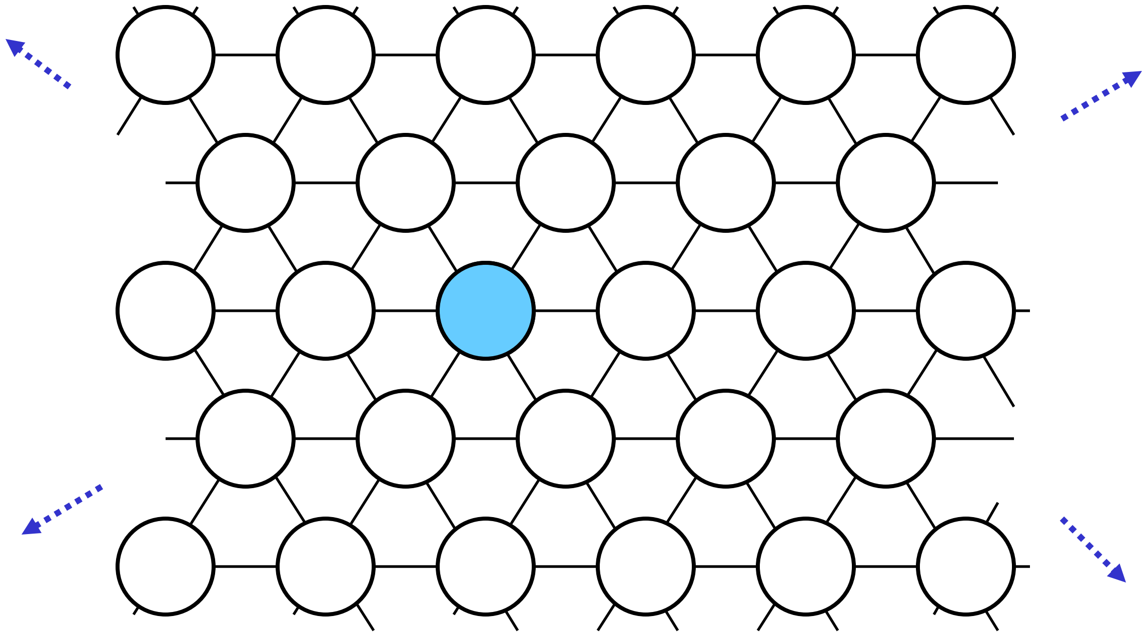
Easy to describe

(Sometimes) astonishing behaviour...

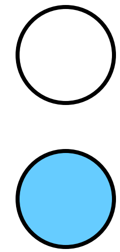
Mathematical analysis challenging and surprising...

Packard's snowflake models (1984)

Triangular lattice



Cells:



empty (water vapour)

full (ice)

Start with one full cell

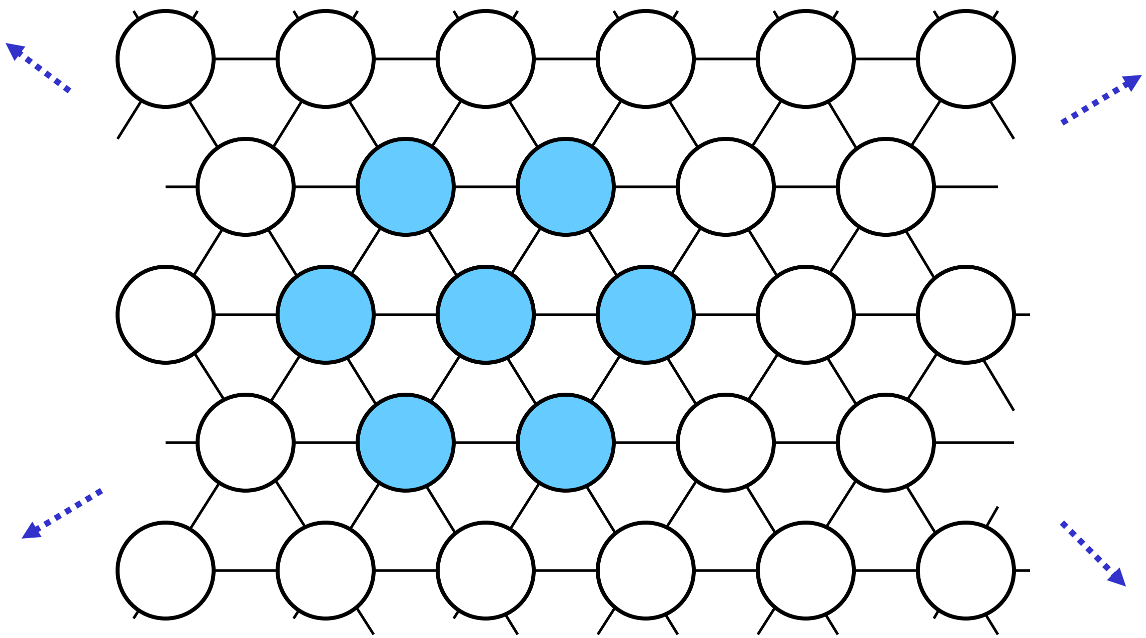
Update rule:

full $\rightarrow$ full

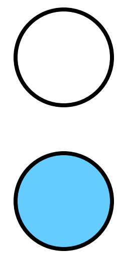
empty $\rightarrow$ full if has 1,4,5 or 6 full neighbours

Packard's snowflake models (1984)

Triangular lattice



Cells:



empty (water vapour)

full (ice)

Start with one full cell

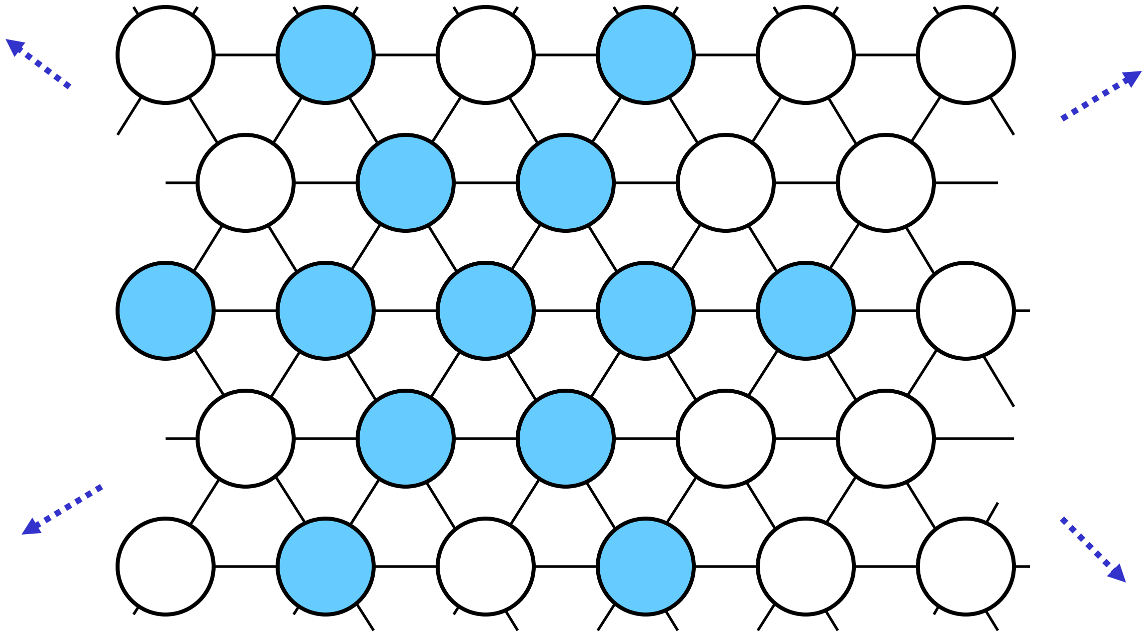
Update rule:

full $\to$ full

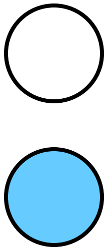
empty $\to$ full if has 1,4,5 or 6 full neighbours

Packard's snowflake models (1984)

Triangular lattice



Cells:


empty (water vapour)

full (ice)

Start with one full cell

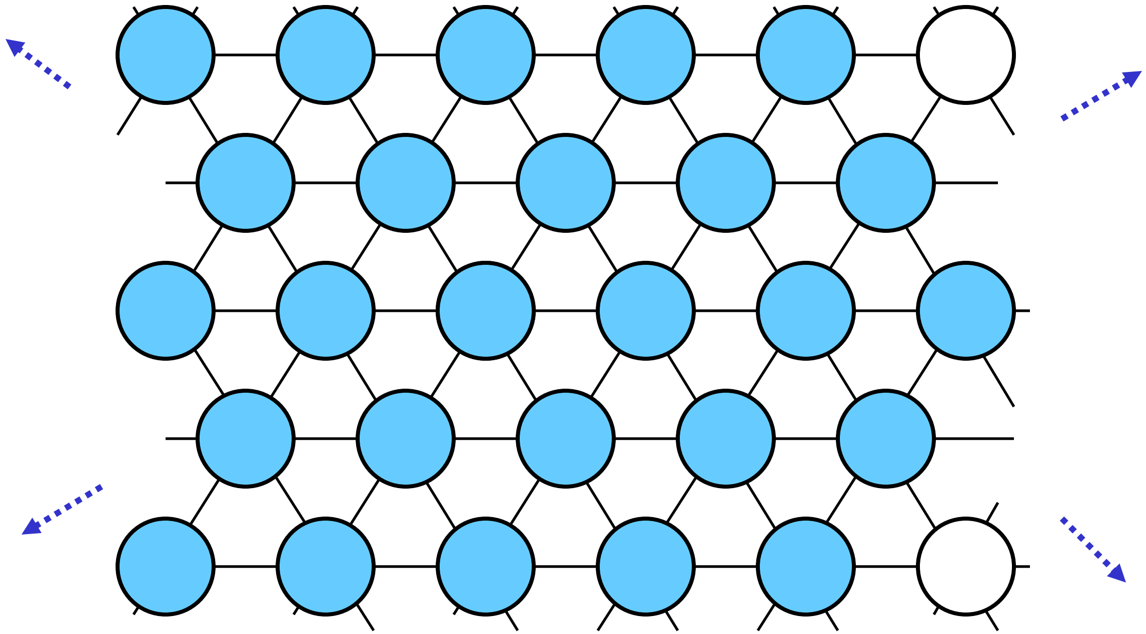
Update rule:

full → full

empty → full if has 1,4,5 or 6 full neighbours

Packard's snowflake models (1984)

Triangular lattice



Cells:

empty (water vapour)

full (ice)

Start with one full cell

Update rule:

full $\rightarrow$ full

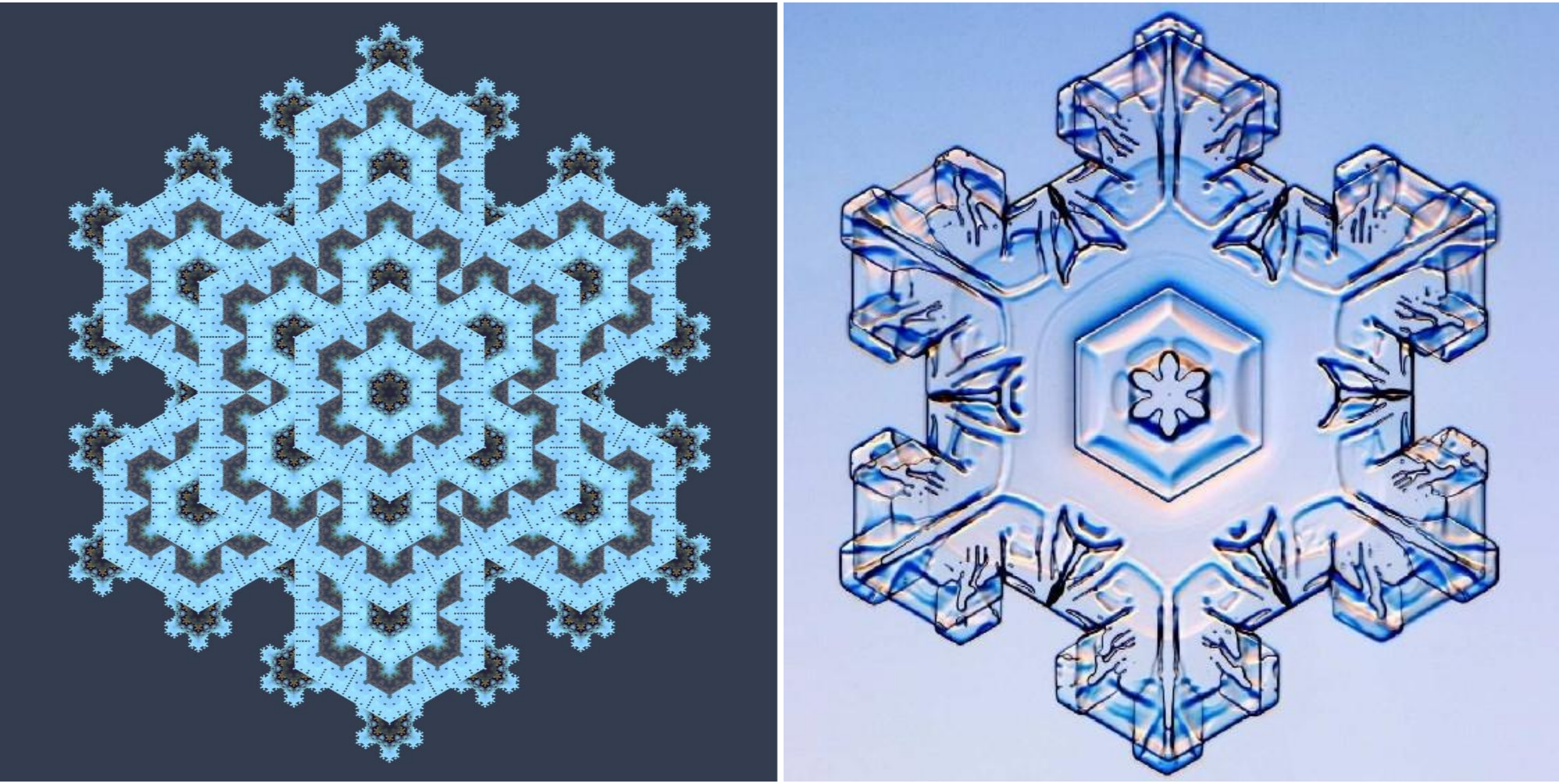
empty $\rightarrow$ full if has ~~1,4,5~~ or 6 full neighbours

1,3,5 or 6

1,5 or 6 etc.

(16 interesting rules)

More! (Mirek's Cellebration)



Source: Janko Gravner

"An elementary schoolchild could look at any of the gorgeous pictures of computer screens in Packard's collection and instantly identify it as a snowflake."
– Steven Levy

"Simulation by computer may be the only way to predict how certain complicated systems evolve. [. . .] The only practical way to generate the [Packard snowflake] pattern is by computer simulation." – Stephen Wolfram

Questions:

behaviour as time $\to \infty$?
shape of outer boundary ?
internal holes?

Let S = set of eventually full cells (in the infinite lattice)

Guess (from simulations):

$S_{1456} = S_{1346}$ = the entire lattice

S_{1345}, S_{156} have holes (etc.)

but:

Theorem (Gravner and Griffeath, 2006)

S_{1456} has holes!! (but not within distance 10^9=1000000000 of the origin!)

S_{1346} = entire lattice

S_{1345}, S_{156} have holes...

Theorem (Gravner and Griffeath)

The density

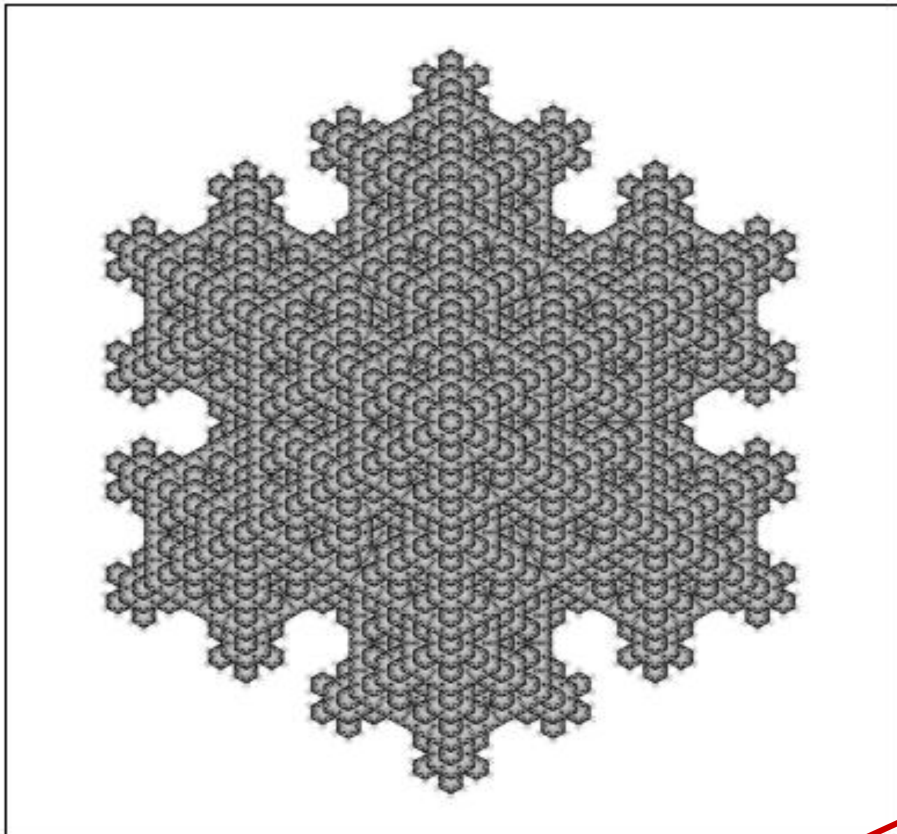
$$\rho := \lim_{n\to\infty} \frac{\#(S\cap[-n,n]^2)}{\#[-n,n]^2}$$

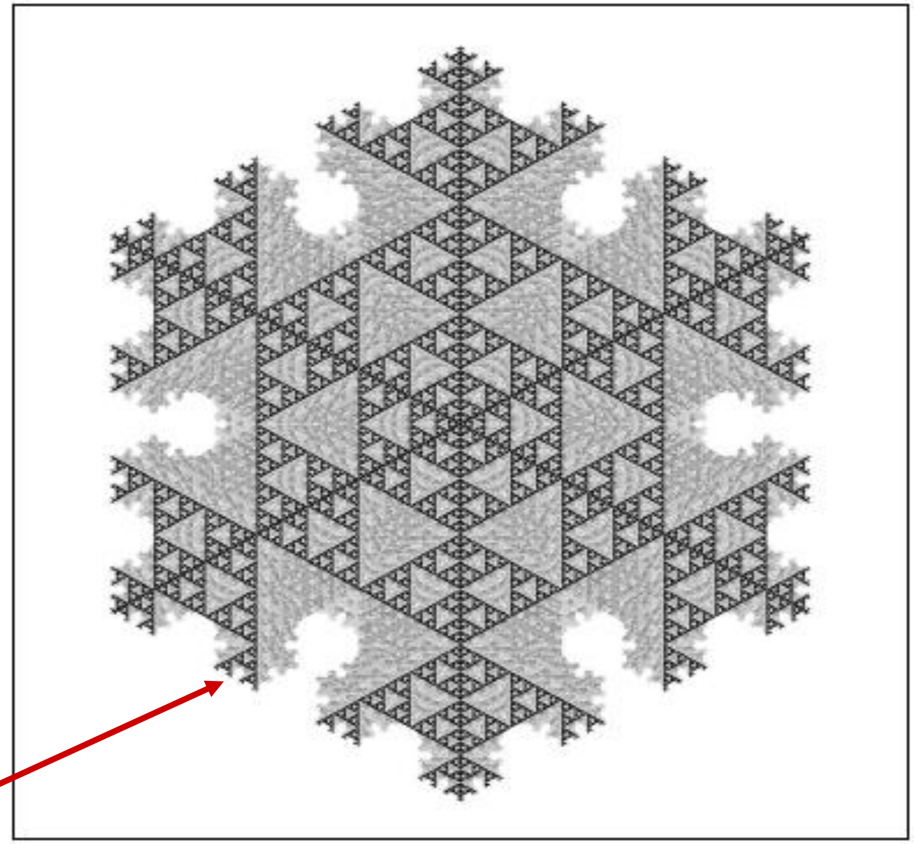
exists for all the models, and

$\rho_{13}=\rho_{135}=5/6$, $\rho_{134}=\rho_{1345}=21/22$, $\rho_{135}=\rho_{1356}=\rho_{1346}=\rho_{13456}=1$,

$\rho_1=0.635\pm0.001$, ρ_{14} ,ρ_{145} $=0.969\pm0.001$,
$\rho_{15}=0.803\pm0.001$, $\rho_{16}=0.740\pm0.001$, $\rho_{156}=0.938\pm0.001$,

$0.995 < \rho_{146} < 1$, $0.9999994 < \rho_{1456} < 1$.



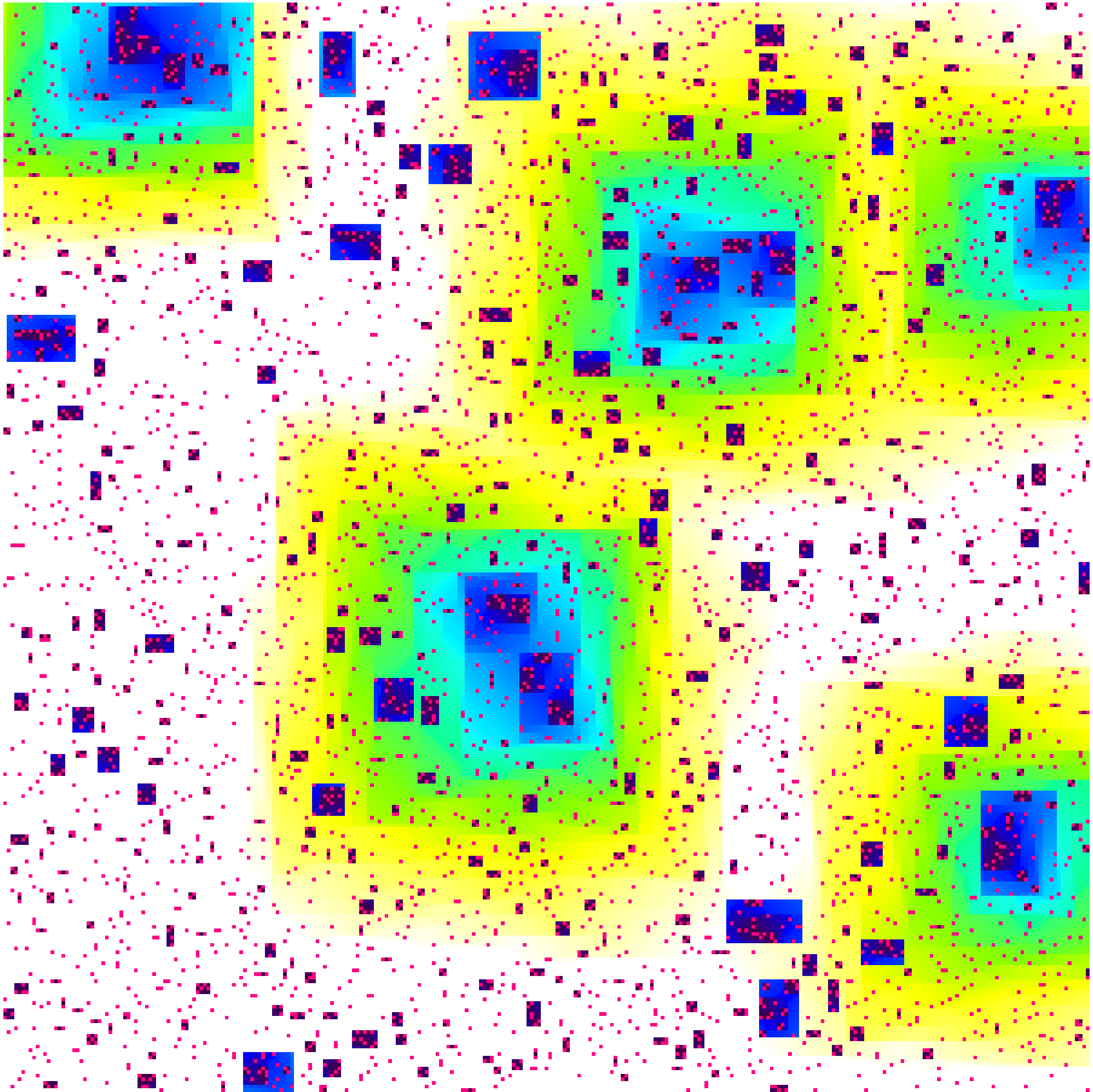


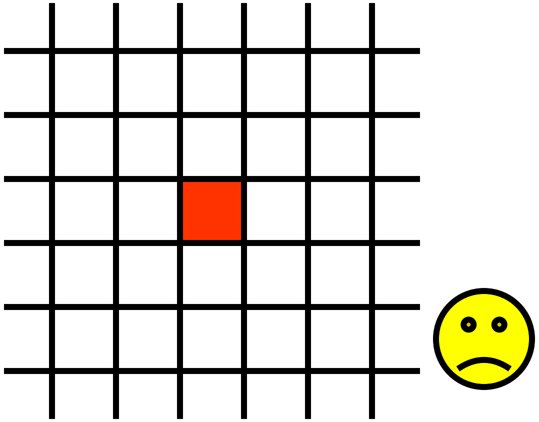
Key tool in proof:

Cells x such that:

time when x becomes full = distance from O to x

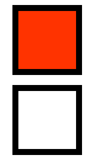
Bootstrap Percolation Model





square lattice ($\mathbf{Z}^2$)

Cells:

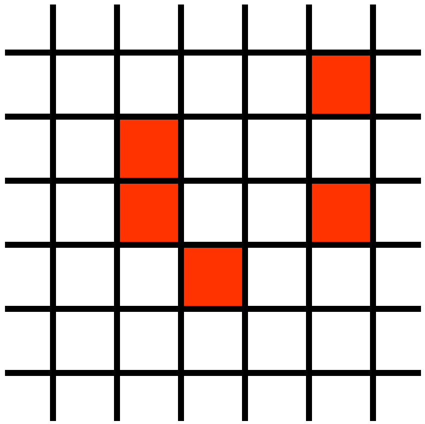


full
empty

Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has $\geq$ 2 full neighbours



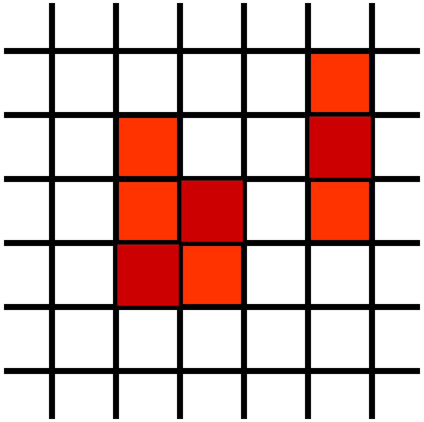
square lattice

Cells: full
empty

Update rule:

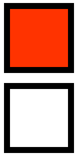
full $\rightarrow$ full

empty $\rightarrow$ full if has $\geq$ 2 full neighbours



square lattice

Cells:



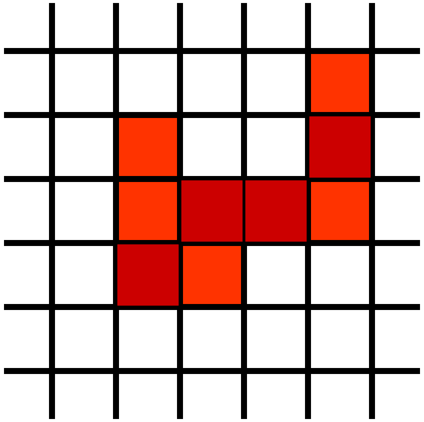
full

empty

Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has $\geq$ 2 full neighbours



square lattice

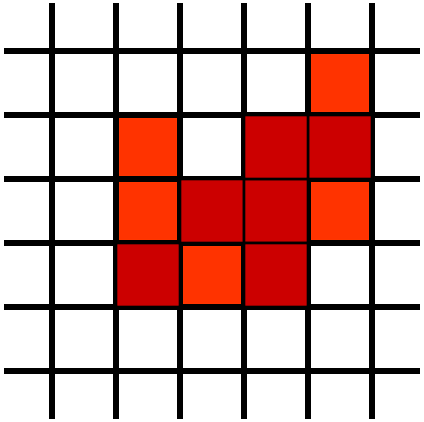
Cells: full

empty

Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has ≥ 2 full neighbours



square lattice

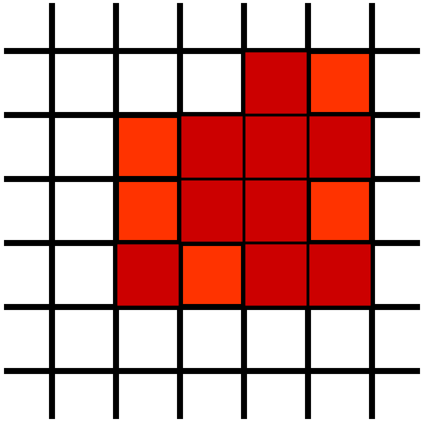
Cells: full

empty

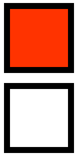
Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has ≥ 2 full neighbours



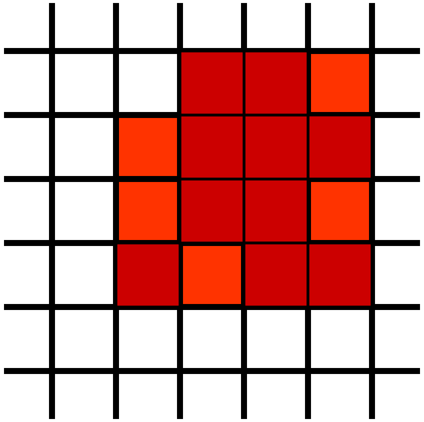
square lattice

Cells:  full / empty

Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has ≥ 2 full neighbours



square lattice

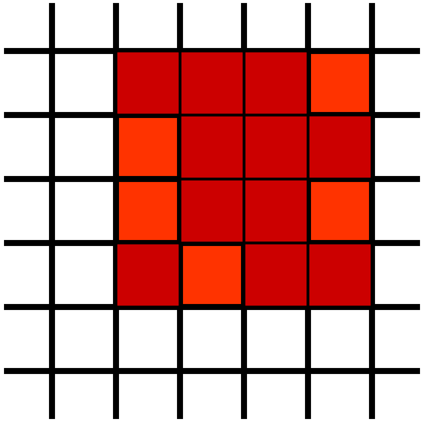
Cells: full

empty

Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has $\geq$ 2 full neighbours



square lattice

Cells: full

empty

Update rule:

full $\rightarrow$ full

empty $\rightarrow$ full if has $\geq$ 2 full neighbours

Random starting state:

Fix 0 < p < 1. Start with each cell:

full with probability p

empty with probability 1-p

independently for different cells.

Simulations

Guess: for some $p_{crit} \approx 0.04$,

if $p > p_{crit}$, every cell eventually full
if $p < p_{crit}$, not every cell eventually full

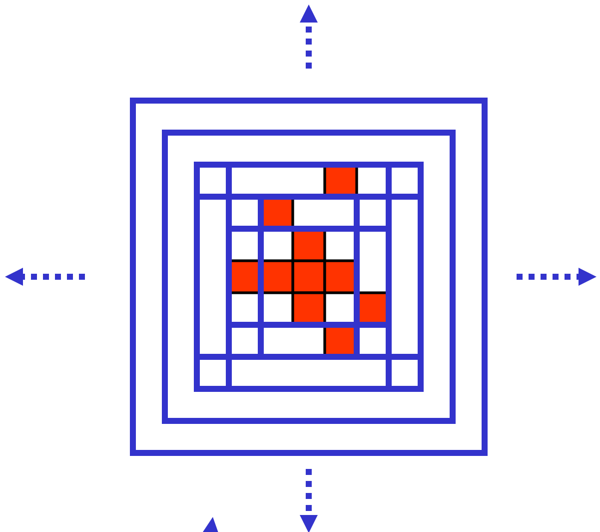
but

Theorem (Van Enter 1987) For any $p > 0$,

$$P(\text{every cell eventually full}) = 1.$$

Proof:

One way to fill everything:



P(fill everything) $\geq$ P(this) = $p^5 \left[(1-(1-p)^3)(1-(1-p)^5)(1-(1-p)^7)\cdots\right]^4$

For $0 < a_n < 1$,

$$\prod_{n=1}^{\infty}(1-a_n) > 0 \text{ if and only if } \sum_{n=1}^{\infty} a_n < \infty.$$

For p>0, $(1-p)^3+(1-p)^5+(1-p)^7+\cdots < \infty$,

so P(fill everything) > 0.

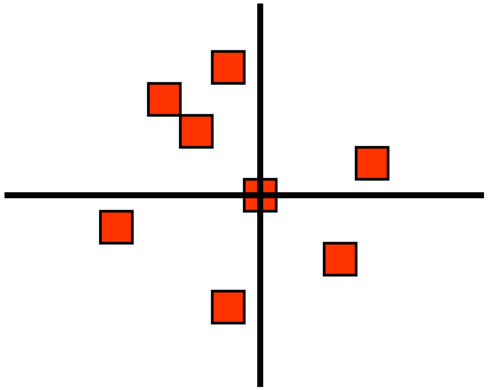
Theorem (Zero-One Law): For any translation-invariant event A on the space of p-coin flips on the lattice Z^d,

$$P(A) = 0 \text{ or } 1.$$

not affected by translating all coins

E.g. {the origin is initially full}

not translation-invariant



{every cell is eventually full}

is translation-invariant

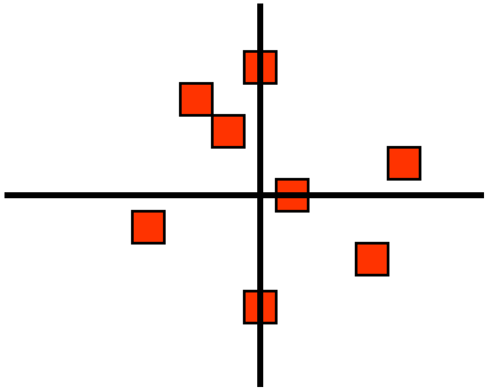
Theorem (Zero-One Law): For any translation-invariant event A on the space of p-coin flips on the lattice Z^d,

$P(A) = 0$ or 1.

not affected by translating all coins

E.g. {the origin is initially full}

not translation-invariant



{every cell is eventually full}

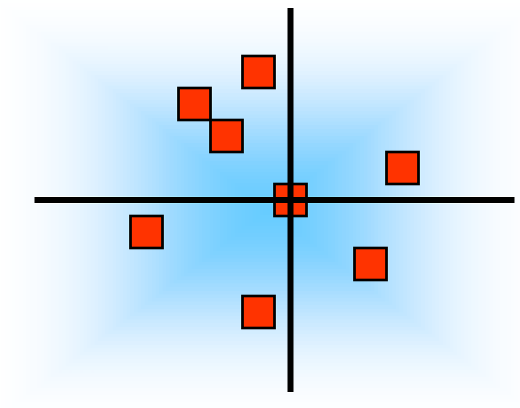
is translation-invariant

Theorem (Zero-One Law): For any translation-invariant event A on the space of p-coin flips on the lattice Z^d,

$$P(A) = 0 \text{ or } 1.$$

not affected by translating all coins

E.g. {the origin is initially full}
not translation-invariant



{every cell is eventually full}
is translation-invariant

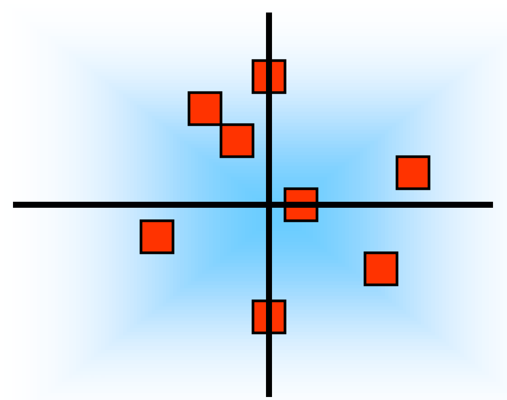
Theorem (Zero-One Law): For any translation-invariant event A on the space of p-coin flips on the lattice Z^d,

$$P(A) = 0 \text{ or } 1.$$

not affected by translating all coins

E.g. {the origin is initially full}
not translation-invariant

{every cell is eventually full}
is translation-invariant



So P(every cell eventually full) = 0 or 1
but P(every cell eventually full) > 0 (from before)

so P(every cell eventually full) = 1.

□

Proof of Zero-One Law

For any event A, any $\varepsilon > 0$, can find an approximation A_ε depending only on coins in a box of size $n = n(\varepsilon)$:

translation by n

symmetric difference

$$P(A \,\Delta\, A_\varepsilon) < \varepsilon$$

so $$P(T^n A \,\Delta\, T^n A_\varepsilon) < \varepsilon.$$

Independence: $$P(A_\varepsilon \cap T^n A_\varepsilon) - P(A_\varepsilon)\, P(T^n A_\varepsilon) = 0$$

so $$|P(A \cap T^n A) - P(A)\, P(T^n A)| < 4\varepsilon.$$

But A translation-invariant, so $T^n A = A$!

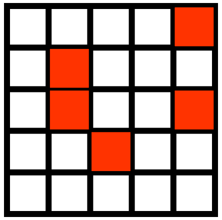
$$|P(A) - P(A)^2| < 4\varepsilon$$

$$P(A) - P(A)^2 = 0$$

$$P(A) = 0 \text{ or } 1.$$ $\square$

Going further:

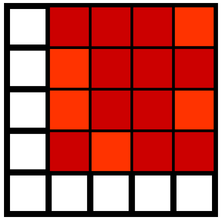
Consider model on an L by L square.



L=5

Going further:

Consider model on an L by L square.



L=5

Theorem (Aizenman and Lebowitz, 1989)

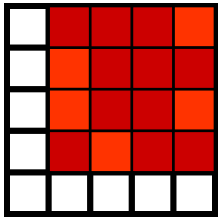
Let $p \to 0$ and $L = e^{a/p}$.

If $a > C$ then $P(\text{fill square}) \to 1$;

if $a < c$ then $P(\text{fill square}) \to 0$.

Going further:

Consider model on an L by L square.



L=5

Theorem (Holroyd, 2003)

Let $p \to 0$ and $L = e^{a/p}$.

If $a > \lambda$ then $P(\text{fill square}) \to 1$;

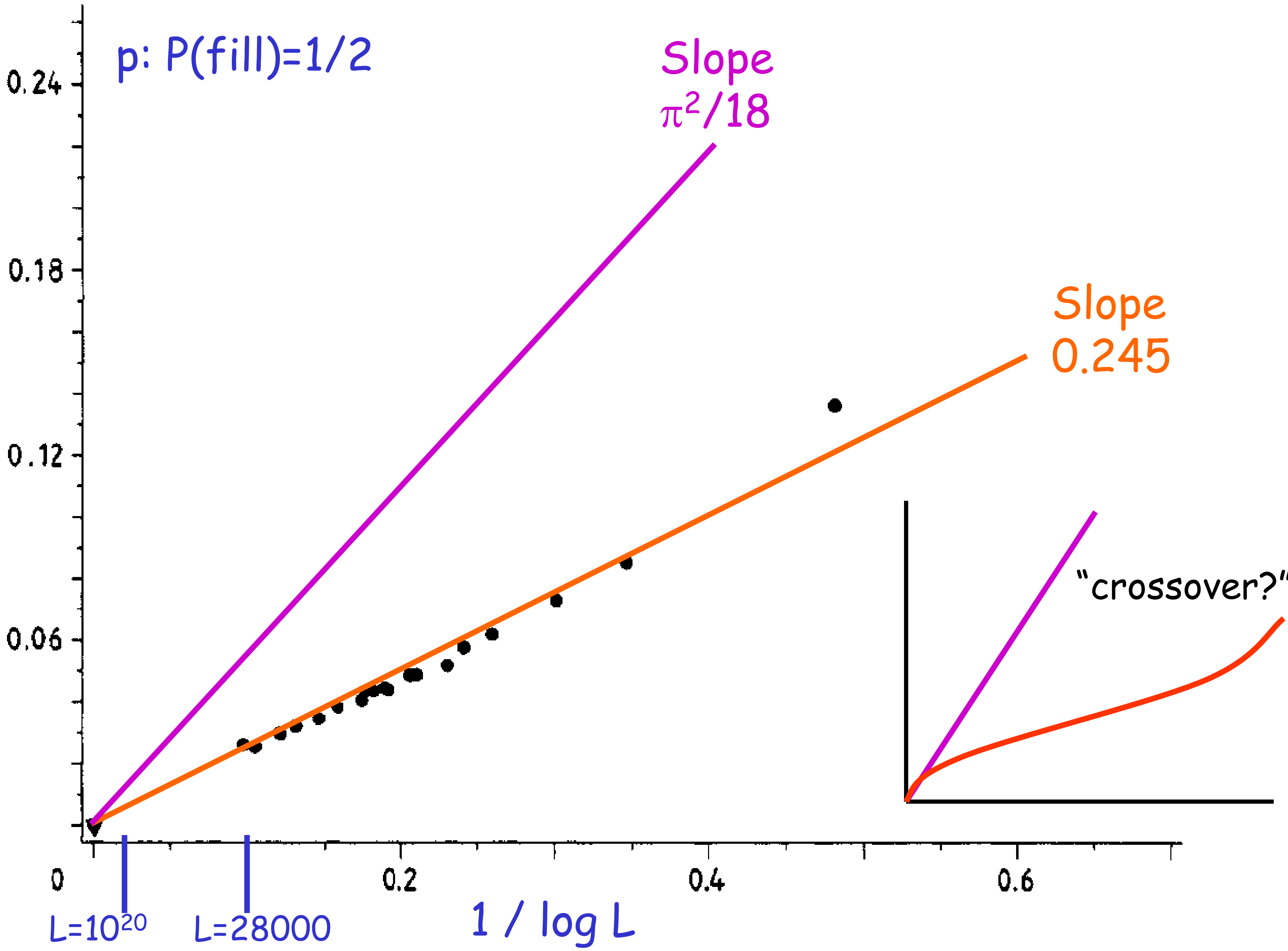
if $a < \lambda$ then $P(\text{fill square}) \to 0$,

where $\lambda = \pi^2/18$.

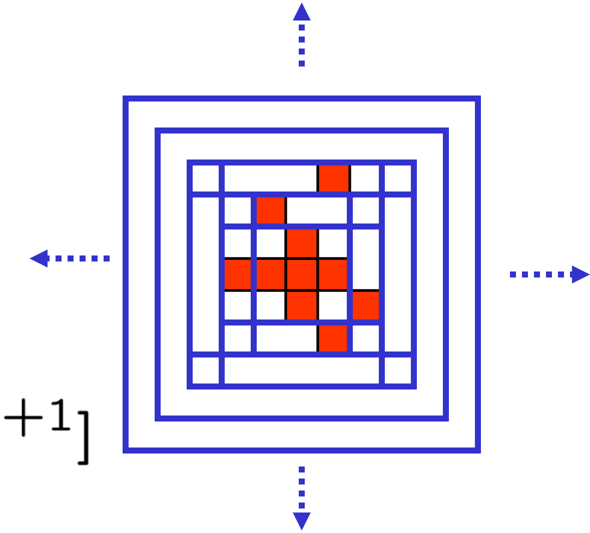
Simulation prediction (Adler, Stauffer, Aharony 1989):

$\lambda = 0.245 \pm 0.015$

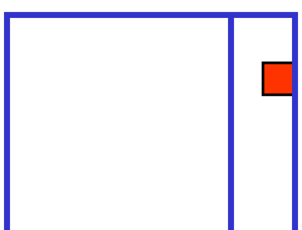
but $\pi^2/18 = 0.548311\ldots$!


p: P(fill)=1/2
Slope
$\pi^2/18$
Slope
0.245
"crossover?"
0.24
0.18
0.12
0.06
0
0.2
0.4
0.6
L=10^{20}
L=28000
1 / log L

$$P(\text{this}) = p^5 \left[(1-(1-p)^3)(1-(1-p)^5)(1-(1-p)^7)\cdots\right]^4$$



$$\begin{aligned}
\log P(\text{this}) &= 5\log p + 4\sum_{n=1}^{\infty} \log[1-(1-p)^{2n+1}] \\
&\approx 4\sum_{n=1}^{\infty} \log[1-(e^{-p})^{2n}] \qquad (p \text{ small}) \\
&\approx 4\frac{1}{2p}\int_0^{\infty} \log(1-e^{-x})\,dx \\
&\approx -\frac{2}{p}\,\frac{\pi^2}{6}.
\end{aligned}$$



$$\Rightarrow \pi^2/18$$

And further...

Understanding the slow convergence:

Theorem (Gravner, Holroyd, 2008)

Let $p \to 0$ and $L = e^{a/p}$.

If $a(L) > \lambda - c/\sqrt{\log L}$ then $P(\text{fill square}) \to 1$;

if $a < \lambda$ then $P(\text{fill square}) \to 0$,

where $\lambda = \pi^2/18$.

Need $L \mapsto L^4$ to halve the "error"!

$1/\sqrt{\log 28000} = 0.31\ldots$

$1/\sqrt{\log 10^{20}} = 0.15\ldots$

Biham-Middleton-Levine traffic model (1992)

Each cell of $\mathbf{Z}^2$ contains:

North-facing car (↑)
or East-facing car (→)
or empty space (**0**).

At odd time steps, each ↑ tries to move one unit North (succeeds if there is a **0** for it to move into).

At even time steps, each → tries to move one unit East (succeeds if there is a **0** for it to move into).

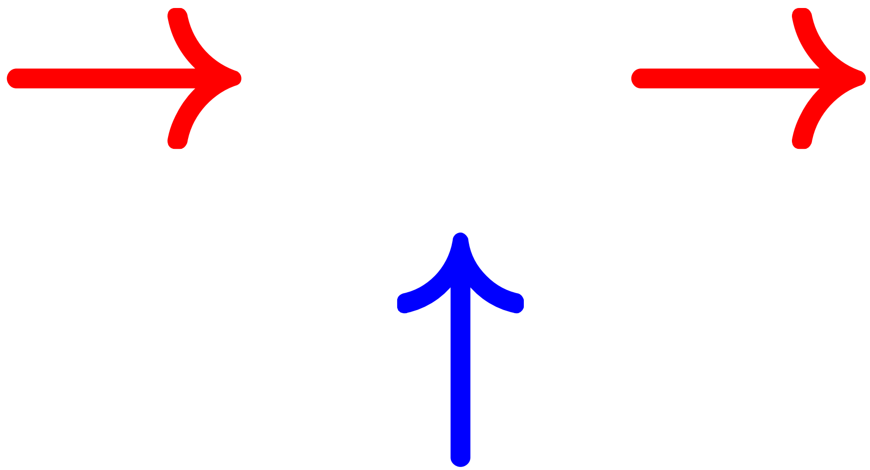
→ →

↑

0

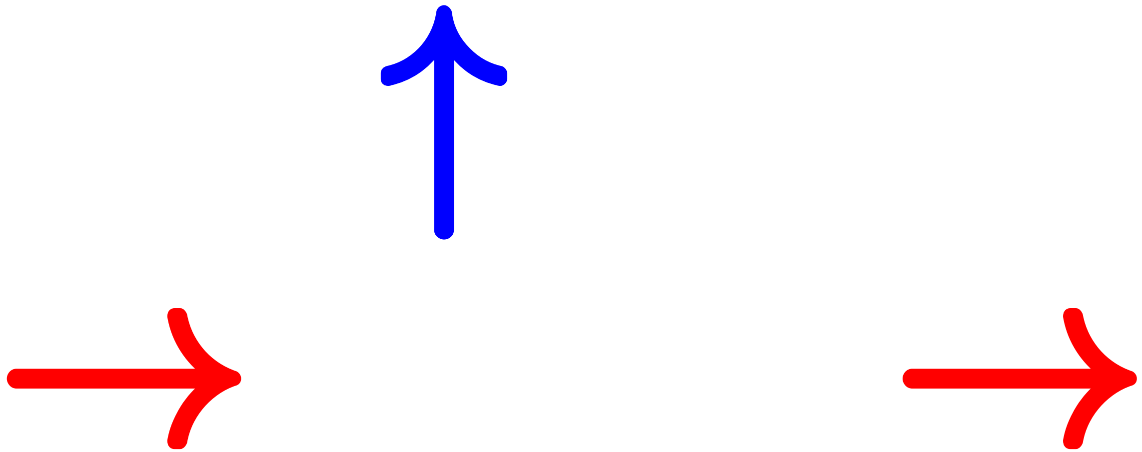
$\rightarrow \ \rightarrow$

$\uparrow$



$\rightarrow$ $\uparrow$ $\rightarrow$

→ ↑ →



↑

→ →

→

Random initial configuration:

$0 < p < 1$

Each cell of $\mathbf{Z}^2$ contains:

North-facing car (↑) with probability $p/2$
East-facing car (→) with probability $p/2$
empty space (**0**) with probability $1 - p$

independently for different sites.

Simulation

Conjecture. For some $0 < p_J < 1$,

$p > p_J$: every car eventually stuck

$p < p_J$: no car eventually stuck

Conjecture. For $0 < p_F < 1$,

$p < p_F$: every car eventually free flowing

$p > p_F$: no car eventually free flowing

Question. $p_F = p_J$?

Intermediate behaviour on finite torus ?

(D'Souza 2005)

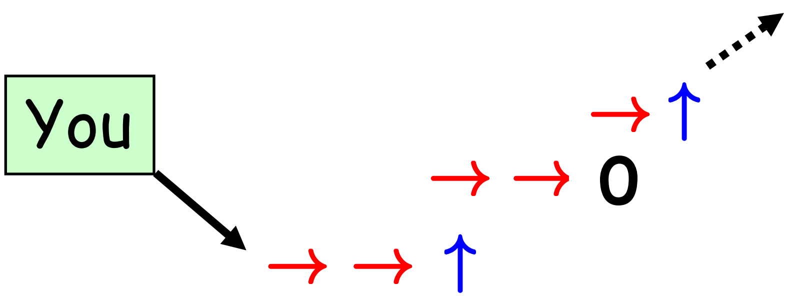
Only rigorous result:

Theorem (Angel, Holroyd, Martin 2005).
For some $p_1 < 1$,
if $p > p_1$ then P(all cars eventually stuck) = 1.

In fact, for $p > p_1$, some cars *never* move...

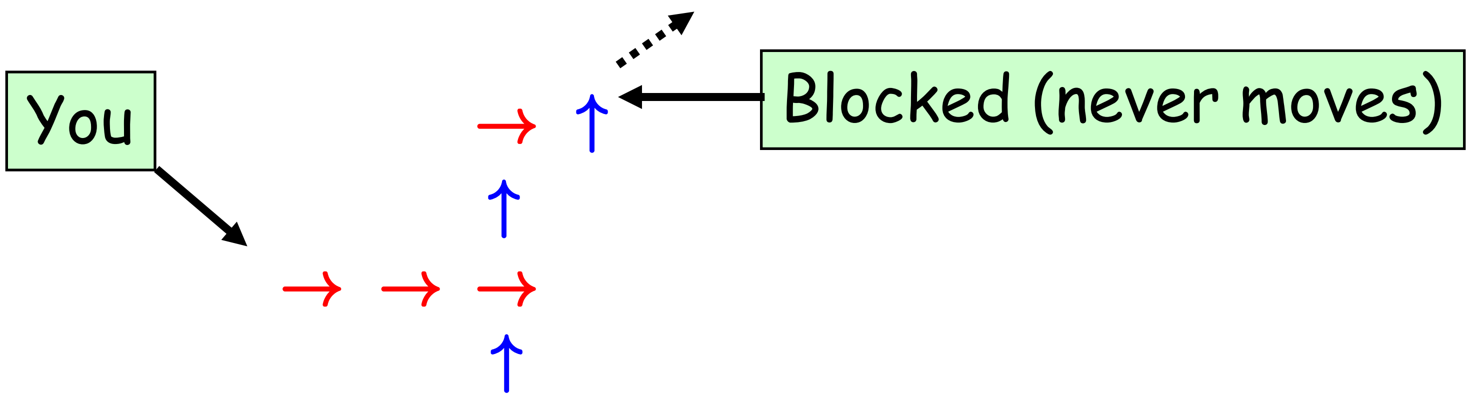
Proof

Easy case: $p = 1$.
Any car is blocked an infinite chain of others:

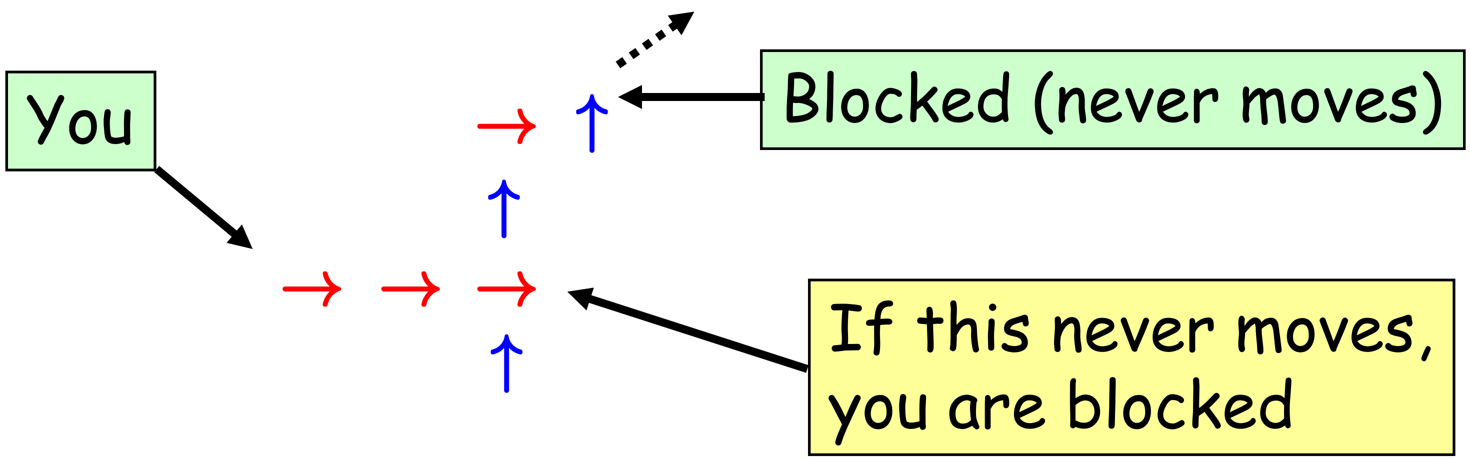


Argument does not work for $p < 1$.
Chain will be broken by an empty space.

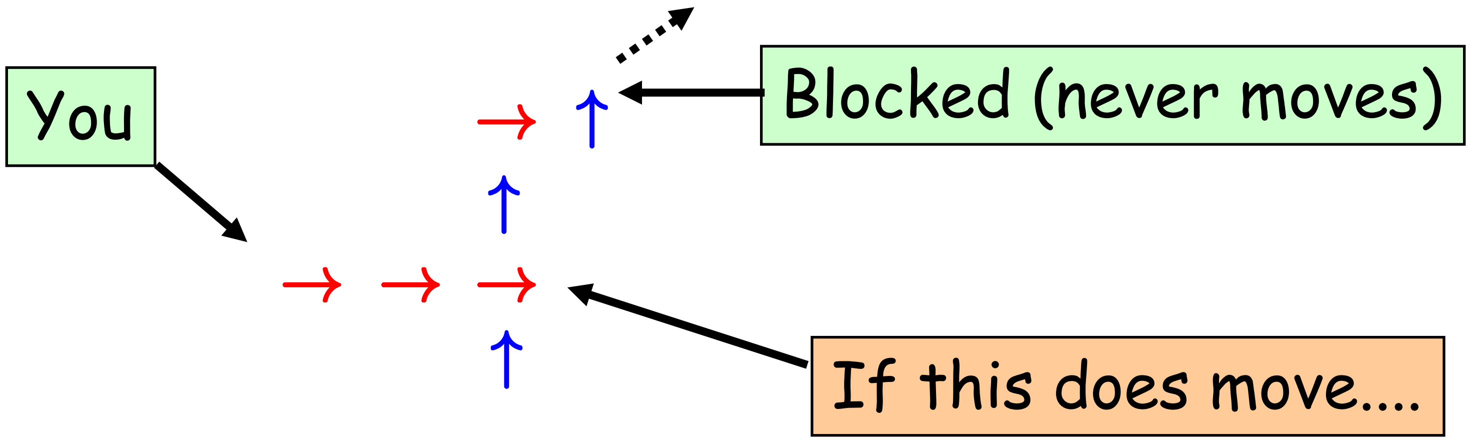
Another way for a car to be blocked:



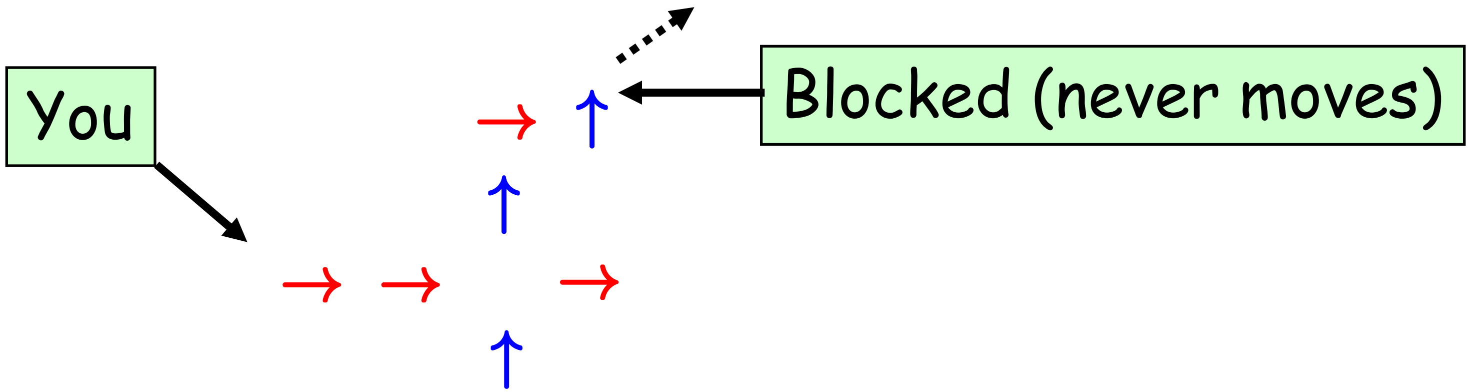
Another way for a car to be blocked:



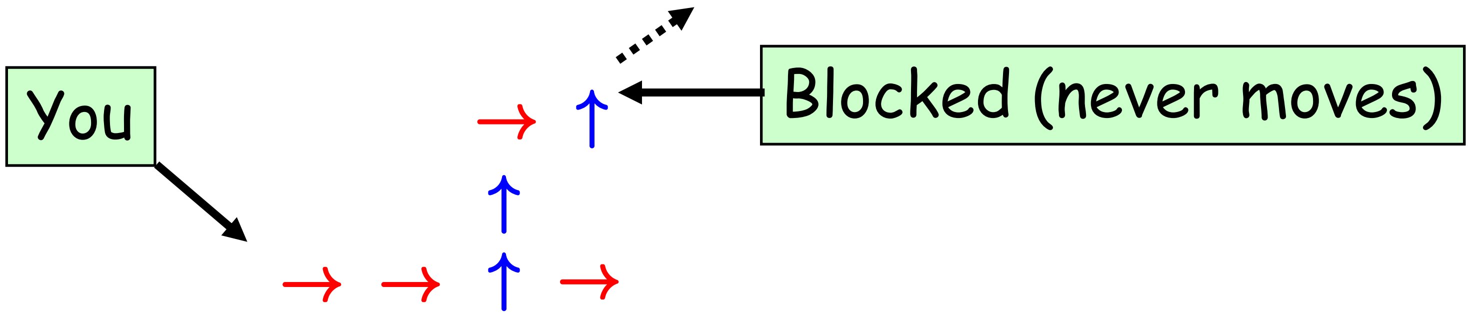
Another way for a car to be blocked:



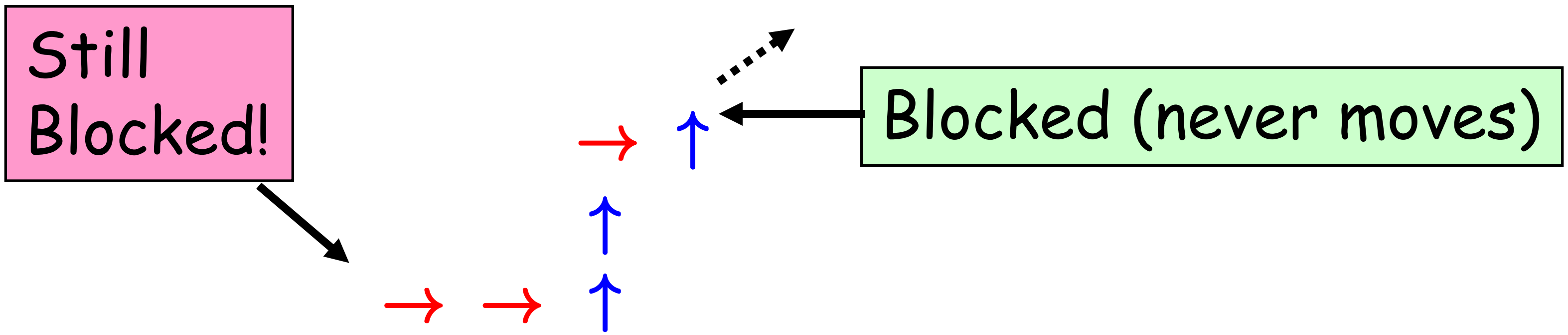
Another way for a car to be blocked:



Another way for a car to be blocked:

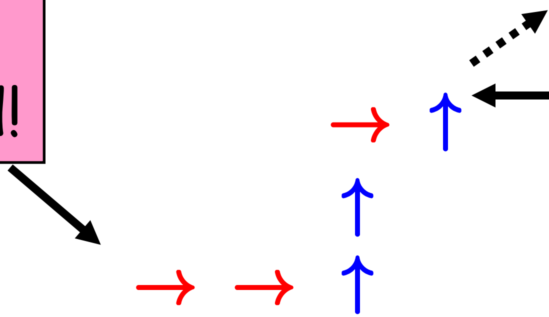


Another way for a car to be blocked:

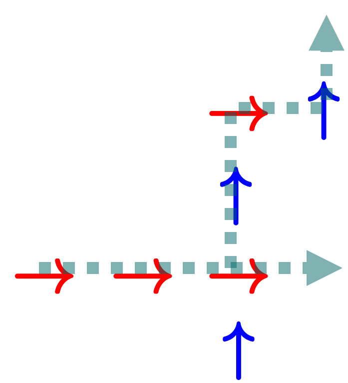


Another way for a car to be blocked:

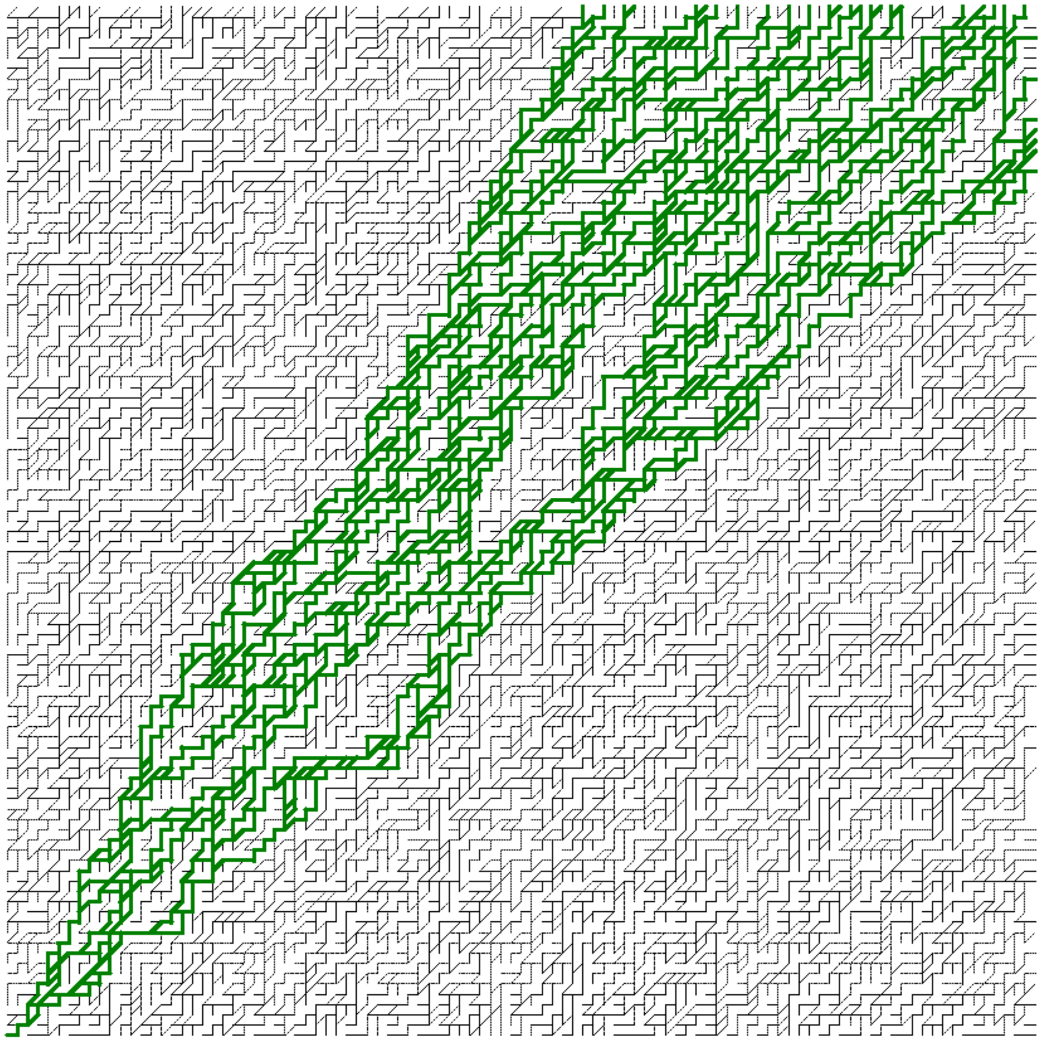
Still Blocked!



Blocked (never moves)



So 2 blocking paths...



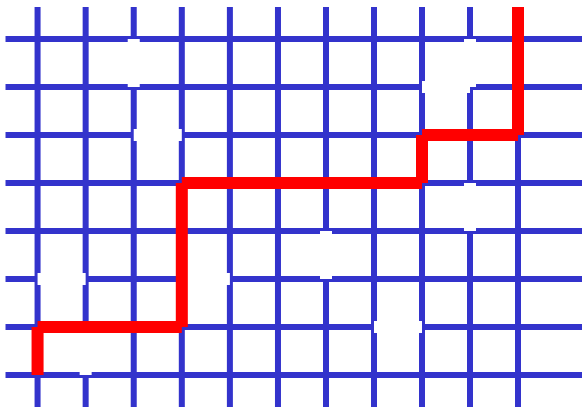
Blocking paths (both types) for one car when p = 1.

For p close to 1, some will survive.

Proof uses percolation theory:

delete a small fraction of connections
at random from the lattice.

In ≥ 2 dimensions, infinite paths remain.



(But not in 1 dimension.)