Results for ARW on $\mathbb{Z}$

Jacob Richey

joint with: Riddhi Basu, Shirshendu Ganguly, Chris Hoffman, Leo Rolla

jfrichey@math.ubc.ca

October 22, 2020

ARW

Today: methods for ARW on $\mathbb{Z}$

Input for ARW:

- $\zeta \in [0, \infty)$ the 'mass' parameter
- $\lambda \in (0, \infty]$ the 'sleep rate'

The rules:

- The particles can be in two states: *active* and *sleepy*
- Each site starts with an active particle with probability ζ (iid)
- Active particles perform independent simple random walks at rate 1
- Sleepy particles do not move
- Active particles turn into sleepy particles at rate λ
- When two or more particles occupy the same site, all particles at that site become active

ARW **fixates** if each vertex is visited finitely many times.

Monotonicity + 0-1 law $\implies$ critical density ζ_c

Existence of the critical density

For each $\lambda \in [0, \infty]$, there is a critical density $\zeta_c(\lambda) \in [0, 1]$ satisfying

$$\mathbb{P}(\text{ARW}(\zeta, \lambda) \text{ fixates}) = \begin{cases} 1, \zeta < \zeta_c \\ 0, \zeta > \zeta_c \end{cases}$$

Driven-dissipative version of ARW:

1. Start with any(!) initial configuration of particles on$[-N, N]^d \subset \mathbb{Z}^d$
2. Run ARW dynamics, killing particles that hit the boundary of the box
3. When all particles are asleep, add a particle at a random site
4. Return to step 2

Different from ARW on a graph: no mass conservation!

Questions

- Critical density ζ_c?
- How quickly does the density converge? Fluctuations?
- Distribution of particles at criticality?

Questions

- Critical density ζ_c?
- How quickly does the density converge? Fluctuations?
- Distribution of particles at criticality?

Concrete problems:

- Time to fixate?
- Behavior of the odometer function?
- Existence of stationary distribution?

Results

Consider ARW on $\mathbb{Z}$.

Rolla, Sidoravicius '12

$\zeta_c \in [\frac{\lambda}{1+\lambda}, 1]$.

Basu, Ganguly, Hoffman '15

$\zeta_c \to 0$ as $\lambda \to 0$.

Hoffman, R., Rolla '20

For any $\lambda > 0$, $\zeta_c < 1$.

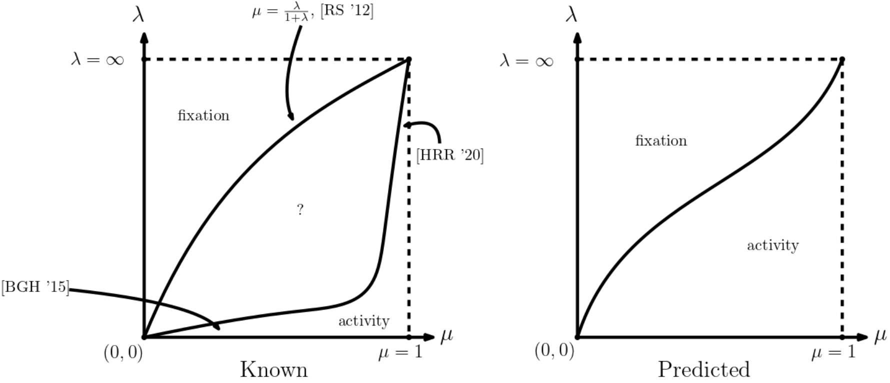


Figure: Phase diagram for ARW on $\mathbb{Z}$.

Results

Let T denote the time to fixation on $G = \mathbb{Z}/n\mathbb{Z}$.

Basu, Ganguly, Hoffman, R. '17

- For any $\lambda \in (0, \infty]$ and $\zeta < \frac{\lambda}{1+\lambda}$,

$$T < Cn \log(n)^2$$

  with high probability as $n \to \infty$ for some $C > 0$.
- For any $\zeta \in (0, 1)$, there exists $\lambda_0 > 0$ such that for $\lambda < \lambda_0$,

$$T > e^{cn}$$

  with high probability as $n \to \infty$ for some $c > 0$.

Site-wise representation:

Define iid instructions $\{\xi_{v,j}\}$ for $v \in G$ and $j \geq 1$ by

$$\xi_{v,j} = \begin{cases} \text{move the particle at } v \text{ to a uniform neighbor} \\ \text{of } v \text{ with probability } \frac{1}{\deg(v)(1+\lambda)} \\ \text{put the particle at site } v \text{ to sleep with probability } \frac{\lambda}{1+\lambda} \end{cases}$$

At each site v, the jth time we topple a particle at v, the state of the system changes according to $\xi_{v,j}$.

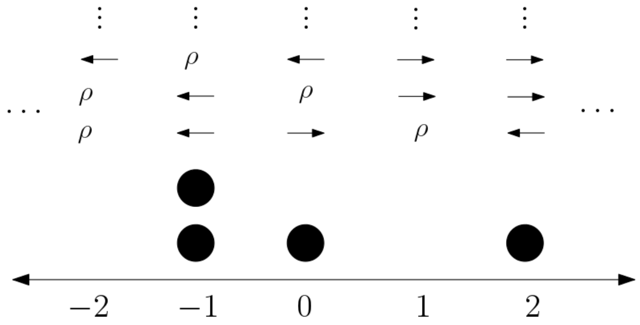


Figure: The stacks of instructions, visualized as lying above each vertex of the graph $G = \mathbb{Z}$. ρ represents a 'sleep' instruction.

Issue: what if the evolution of the system depends on the order of the instructions?

The order doesn't matter!

Abelian property

Consider any initial configuration η, and any (legal) sequence of stack instructions $\overline{\xi} = (\xi_{v^1,j^1}, \xi_{v^2,j^2}, \ldots, \xi_{v^N,j^N})$. If $\overline{\xi}'$ is any (legal) re-ordering of the instructions in $\overline{\xi}$, then $\overline{\xi}\eta = \overline{\xi}'\eta$.

Key idea: we can choose clever toppling sequences.

Rolla, Sidoravicius '12

ARW fixates almost surely on $G = \mathbb{Z}$ for $\zeta < \frac{\lambda}{\lambda+1}$.

Proof sketch: find 'traps' for the particles to fall asleep in.

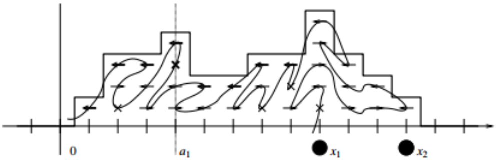


Figure: A diagram from [RS '12], showing the first trap a_1 for the particle x_1.

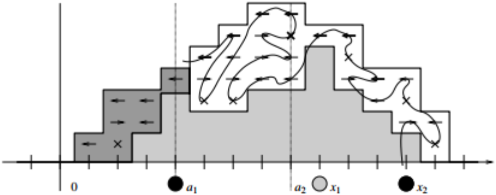


Figure: The trap a_2 for the particle x_2, obtained recursively by exploring the stack instructions.

Let x_k = position of kth particle to the right of 0, $k = 1, 2, \ldots$

Define the traps a_k recursively:

- $a_0 = 0$.
- For $k > 0$: send a ghost particle out from x_k, ignoring sleep instructions, until it hits a_{k-1}.
- a_k = leftmost site to the right of a_{k-1} where the second to last instruction seen by the ghost was a sleep instruction.

Particles follow the paths of their ghosts, except that they fall asleep in the trap.

Proof sketch: Trap setting succeeds if $a_{k-1} < x_k$ for all $k \in \mathbb{N}$.

Proof sketch: Trap setting succeeds if $a_{k-1} < x_k$ for all $k \in \mathbb{N}$.

On average, $x_k - x_{k-1} = \zeta^{-1}$ and $a_k - a_{k-1} = \frac{1+\lambda}{\lambda}$.

Proof sketch: Trap setting succeeds if $a_{k-1} < x_k$ for all $k \in \mathbb{N}$.

On average, $x_k - x_{k-1} = \zeta^{-1}$ and $a_k - a_{k-1} = \frac{1+\lambda}{\lambda}$.

By the LLN, $x_k \approx k\zeta^{-1}$, and $a_k \approx k \cdot \frac{1+\lambda}{\lambda}$.

Proof sketch: Trap setting succeeds if $a_{k-1} < x_k$ for all $k \in \mathbb{N}$.

On average, $x_k - x_{k-1} = \zeta^{-1}$ and $a_k - a_{k-1} = \frac{1+\lambda}{\lambda}$.

By the LLN, $x_k \approx k\zeta^{-1}$, and $a_k \approx k \cdot \frac{1+\lambda}{\lambda}$.

Thus if $\zeta < \frac{\lambda}{1+\lambda}$, $x_k > a_k$ for all k large a.s.

Proof sketch: Trap setting succeeds if $a_{k-1} < x_k$ for all $k \in \mathbb{N}$.

On average, $x_k - x_{k-1} = \zeta^{-1}$ and $a_k - a_{k-1} = \frac{1+\lambda}{\lambda}$.

By the LLN, $x_k \approx k\zeta^{-1}$, and $a_k \approx k \cdot \frac{1+\lambda}{\lambda}$.

Thus if $\zeta < \frac{\lambda}{1+\lambda}$, $x_k > a_k$ for all k large a.s.

So $\mathbb{P}(\text{fixation}) > 0$. By the 0-1 law, $\mathbb{P}(\text{fixation}) = 1$. $\square$

Basu, Ganguly, Hoffman, R. '17

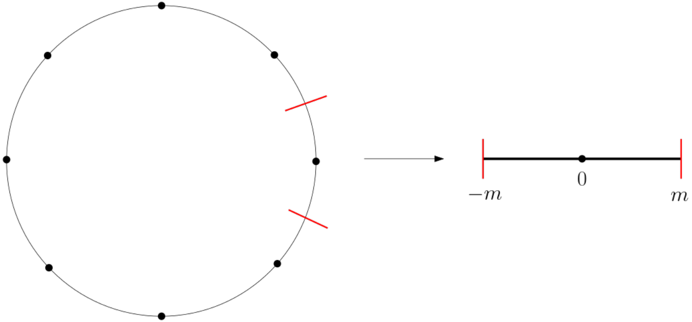
Consider ARW on $\mathbb{Z}/n\mathbb{Z}$ started from iid Ber(ζ) particles per site. For any $\lambda \in (0, \infty]$ and $\zeta < \frac{\lambda}{1+\lambda}$,

$$\# \text{ stack instructions to fixate} < Cn \log(n)^2$$

with high probability as $n \to \infty$ for some $C > 0$.

The fixation speed depends on the initial condition.

First step: gather $O(\log n)$ particles at each of $O\left(\frac{n}{\log n}\right)$ sites.



Focus on a single sub-interval.

How to adapt the traps for an interval?

Two-sided traps: ghosts start at 0, traps are set recursively at the boundary. Procedure fails if the traps reach 0.

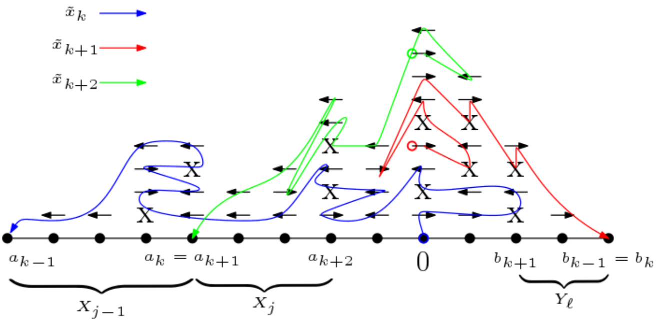


Figure: Setting traps 'in both directions' on an interval.

Internal erosion on an interval:

1. Start with the interval $X_0 = [-m, m] \cap \mathbb{Z}$.
2. Start a simple random walker from 0, stopped when she hits a boundary point $B \in \partial X_t$.
3. Remove the point B from X_0, to obtain $X_{t+1} = X_t \setminus \{B\}$.
4. Return to step 2

Idea: replace each segment $[j-1,j]$ and $[-j,-j+1]$ by an independent Exponential(j) length of rope, connect them all together, and initialize by lighting both ends on fire.

Properties of exponentials give a coupling between this process and the erosion process. Key computation: for $a, b > 0$,

$$\mathbb{P}^0(\text{hit } b \text{ before } -a) = \frac{a}{a+b} = \mathbb{P}(\text{Exp}(b) < \text{Exp}(a)).$$

($+$ memoryless-ness)

Levine, Peres, '07

Let $R(m)$ be the number of sites remaining when the origin is eroded. As $m \to \infty$,

$$\frac{R(m)}{m^{3/4}} \to_d \left(\frac{8}{3}\right)^{1/4} \sqrt{|Z|},$$

where $Z \sim N(0,1)$.

Note: the number of remaining sites is $O(m^{3/4}) = o(m)$.

Issue: at each stage, one of the traps moves a random distance – distributed as Geo$\left(\frac{1+\lambda}{\lambda}\right)$ – not distance 1.

We are still able to couple with the rope process, but the exponentials have random means. Many concentration estimates necessary.

Conclusion: the left and right side traps still shrink to 0 at the same rate (up to lower order stuff). Two-sided trap setting succeeds for $\zeta < \frac{\lambda}{1+\lambda}$.

Hoffman, R., Rolla '20

For any $\lambda > 0$, $\zeta_c(\lambda) < 1$.

Fix λ, choose ζ very close to 1, show that the odometer at 0 is infinite.

Hoffman, R., Rolla '20

For any $\lambda > 0$, $\zeta_c(\lambda) < 1$.

Fix λ, choose ζ very close to 1, show that the odometer at 0 is infinite.

Count how many particles exit a large interval $[0, N]$ after stabilizing.

Hoffman, R., Rolla '20

For any $\lambda > 0$, $\zeta_c(\lambda) < 1$.

Fix λ, choose ζ very close to 1, show that the odometer at 0 is infinite.

Count how many particles exit a large interval $[0, N]$ after stabilizing.

Initial condition is not iid (use RSZ '19)

Initial condition & dynamics:

- free particle
- carpet particle
- transit regions
- hole

Topple left-most block with a free particle

Carpet sometimes gets turned into free

Transit regions are all active, but never move

Holes can move inside their blocks

Initial condition & dynamics:

- free particle
- carpet particle
- transit regions
- hole

Topple left-most block with a free particle

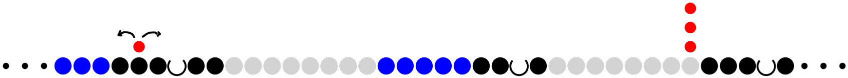
Carpet sometimes gets turned into free

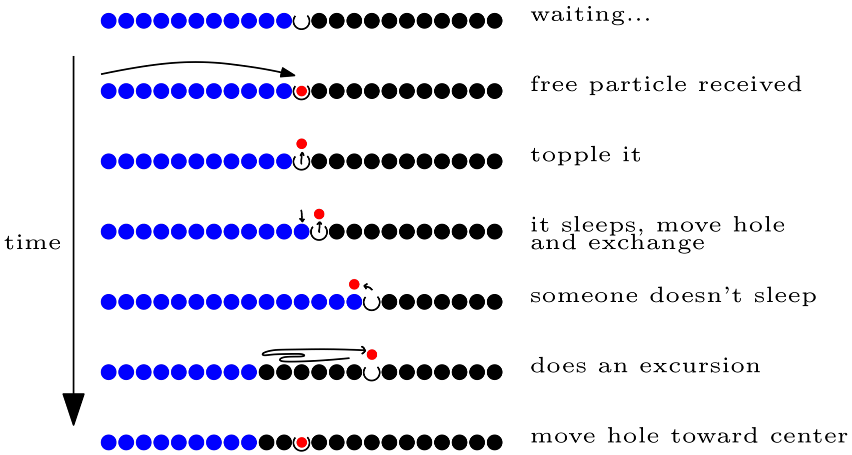
Transit regions are all active, but never move

Holes can move inside their blocks

Typical situation during the dynamics:

- free particle
- sleepy carpet particle
- active carpet particle




waiting...
free particle received
topple it
it sleeps, move hole
and exchange
someone doesn't sleep
does an excursion
move hole toward center
time

We want particles to take long enough excursions to make it to neighboring blocks.

We want particles to take long enough excursions to make it to neighboring blocks.

Problem: what if the hole hits the edge of the block?

Frozen configuration (bad event)

We want particles to take long enough excursions to make it to neighboring blocks.

Problem: what if the hole hits the edge of the block?

Frozen configuration (bad event)

Solution: the hole has a strong bias towards 0. Expected maximum distance reached by a SRW excursion $= \infty$

Blocks / transit regions (i.e. ζ) must be sufficiently large.

Main lemma: the probability that the *last* free particle to interact with a given block causes a ‘freeze’ is small.

Formulating this is tricky: we condition on previous blocks, and bound uniformly over all numbers of free particle inputs.

Main lemma: the probability that the *last* free particle to interact with a given block causes a 'freeze' is small.

Formulating this is tricky: we condition on previous blocks, and bound uniformly over all numbers of free particle inputs.

Successively condition on blocks, use single-block-estimate on each to upper bound total number of frozen blocks remaining at the end.