Part III Essays Easter 2023

E528F

Attach to Front of Essay

Essay Number:

37

Essay Title:

Activated Random Walks

ACTIVATED RANDOM WALKS

May 2, 2024

Abstract

In this paper, we explore the Activated Random Walks (ARW) which is similar to the Abelian Sandpile model but more tractable. It is an important model to study Self-organised criticality which in turn is resonant with seismic activity and other natural occurrences. Recently, it was shown by Gaudillière, Asselah and Forien that in every dimension $d \geq 1$ and for every sleeping rate λ, the critical density $\mu_c(\lambda)$ for the ARW on $\mathbb{Z}^d$ is always less than 1. This is going to be our star theorem. We introduce concepts like Diaconis-Fulton construction which helps to see results like the Monotonicity lemma which are otherwise very hard to see. It is a journey to the theorem by Gaudillière, Asselah and Forien and we continuously explore the ARW model during this journey through other results.

Contents

1 INTRODUCTION

1.1 Self-organised criticality.

Consider the phenomenon where a system builds up energy slowly over a long period of time and then releases it in intermittent bursts. Typical examples of this will be earthquakes, wildfires, landslides, and avalanches. To explain this phenomenon, Bak, Tang and Wiesenfield coined the term **Self-organised criticality** and gave an example, inspiring the idea of **Abelian Sandpile**, that has now grown popular among the Statistical Physicists.

Suppose there is a pile of sand resting on a table. The current slope of the sandpile is η and we are sprinkling sand from the above. At any time, we ask these two questions:

- Is η increasing or decreasing?
- How much sand falls off the table?

One notices that if the pile is flat, then the grains of sand will just sit on the top and increase the slope. If the slope is very steep, then adding sand grains will disturb it and may cause lots of 'sand grains' to tumble down and decrease the slope. The grains can also interact with each other causing an avalanche. There is some critical slope η_c, where the slope is trying to converge to with time. This is self organised criticality as the system drives itself to the critical state without the need of tuning of the parameters.

1.2 Ordinary criticality and Self-organised criticality.

Consider the box $\Lambda_n = [-n, n]^d$ in $\mathbb{Z}^d$ and suppose the initial configuration is independent Bernoulli number of particles of some parameter at each vertex. Suppose, at a constant rate, a new particle is added to a site $x \in \Lambda_n$ chosen uniformly at random and these particles then diffuse in the box by jumping to their nearest neighbour sites independently at rate λ. Whenever they are alone at a vertex, they go to sleep at some rate. If they hit the boundary, then they get killed. This is a diffusion-reaction mechanism as the partilces are diffusing in the lattice but there are two opposing forces or reactions to this: the 'sleeping' and the 'killing' phenomenon. Between every two additions of new particles, this diffusion-reaction mechanism is run at a high speed enough to stabilise the system just before the next addition of a particle. Does it seem similar to the Abelian Sandpile model on the finite table? Then, as $t \to \infty$, the average density of the particles goes to a limit μ_c though we didn't input any relevant parameter (indpendent of the sleeping and jumping parameters). This is again ***self-organised criticality***.

Now, expand this box to infinity from all the sides and don't add any new particle this time. Let $\eta_t(x)$ denote the state at vertex x and μ denote the Bernoulli parameter. μ is the average density of particles in the lattice and this stays constant throughout as we neither create nor annihilate the particles. We observe that there is a certain Bernoulli parameter $\tilde{\mu_c}$ such that whenever $\mu < \tilde{\mu_c}$, $\eta_t(x)$ at every x eventually becomes a constant (***fixates***). If $\mu > \tilde{\mu_c}$, then this doesn't happen. So, here the criticality depends on the parameter and hence, this is ***ordinary criticality***.

Density Conjecture: $\tilde{\mu_c} = \mu_c$[4]

1.3 Abelian Sandpile and Activated Random Walk.

Consider the lattice $\mathbb{Z}^d$. This is going to be our 'infinite table' and the sandgrains are going to be the collection of indistinguishable particles present at the vertices. As soon as a vertex has $2d$ number of particles, it topples and sends one particle to each neighbouring vertex. Now, these neighbouring particles may have $2d$ number of particles and then they topple, and this continues. This starts an 'avalanche'. This is the **Abelian Sandpile** model in simplest words.

A bit more explored model or easier to explore model related to the Abelian sandpile is the **Activated Random Walk** model which is defined as follows:

Let $X_0, X_1, X_2, \ldots \overset{\text{iid}}{\sim} \text{Bernoulli}(p)$ denote the number of particles at each vertex of $\mathbb{Z}^d$ at time $t = 0$. Each particle has two Poisson clocks: when the first one rings, the particle jumps to a neighbouring vertex and on the ringing of the second clock, the particle goes to sleep only if it is alone at a vertex ($A \to S$). So, particles on the graph $\mathbb{Z}^d$ can be in two states: ***active*** (A) and ***sleeping*** (S) at any time t. This entire process happens in continuous time. There can be multiple particles at any vertex and they don't interact with each other. Let the jump rate be j and sleep rate be λ. Then, $j = 0$ for a sleeping particle and 1 for active particles. A sleeping particle becomes an active one as soon as an active particle enters its site giving the transition: $A + S \to 2A$.

Remark 1.1 *In fact, instead of distributing the number of particles as iid Bernoulli variables, we can consider any translation-ergodic initial configuration. We define the Activated Random Walk properly in section.*

1.4 Analogy between Abelian Sandpile and Activated Random Walk.

The major difference between the Activated Random Walk and Abelian Sandpile is that in the latter, there is a threshold i.e. the number of particles that any vertex will hold before toppling all the particles together to the nearest neighbours. In case of Activated Random Walk, toppling at a vertex is not controlled by any threshold but rather by a Poisson clock.

However, there is analogy between these two models. The movement of active particles from any vertex plays the role of particles toppled at any vertex in the Abelian Sandpile. This movement can wake up the sleeping particles which can then wake up other sleeping particles creating an avalanche similar to that in the Abelian Sandpile model. Moreover, the density of the sleeping particles acts very similar to the slope of the sandpile in the model given by Bak, Tang and Wiesenfield.

2 DIACONIS-FULTON CONSTRUCTION

Let $\mathbb{N}_{\mathfrak{s}} = \mathbb{N}_0 \cup \mathfrak{s}$. The configuration of the ARW at any time t is given by $\eta_t \in \mathbb{N}_{\mathfrak{s}}^{\mathbb{Z}^d}$. Here, $\eta_t(x) = k$ means that at time t and at vertex x, there are k particles: $k \in \mathbb{N}_0$ means that there are k active particles and $k = \mathfrak{s}$ means that there is a sleeping particle at vertex x.

If we start the ARW with a very small Bernoulli percolation parameter p and a large enough sleeping rate λ, then we will notice that after a very large time, all the particles go to sleep in distinct vertices. We say that the system has ***fixated***. This is very similar to the fixation we had introduced before - the number of particles at each vertex remains the same after a long time. A natural question arises here: for any given $\lambda \in (0, \infty)$, do we have $p_c < 1$ such that whenever $p < p_c$, the system fixates and whenever $p > p_c$, the system never fixates? This result was fully addressed by Asselah, Forien, and Gaudillière in 2022. We will primarily focus on this result while building tools to deal with it.

In section 1.2, we introduced an interesting system, the driven-dissipative system, where particles get killed as soon as they hit the boundary and proposed the density conjecture. Currently, we do not have an answer to this question. Moreover, similar to Percolation theory, we don't know about the state of the system at the critical parameter for several dimensions.

However, there are other simpler questions which are hard to answer due to the following difficulties:

- The system is not attractive i.e. unlike the driven-dissipative system, the density doesn't converge to the critical value itself and hence it is hard to find a monotonic sub-sequence of configurations.
- The number of particles is conserved which gives rise to long-range effects and hence, we cannot use Energy-Entropy arguments.

We have a way to combat the first difficulty but we can't do much in case of the second difficulty. The first one is overcome by using *Diaconis-Fulton* construction.

2.1 Diaconis-Fulton construction.

We now turn $\mathbb{N}_{\mathfrak{s}}$ into an ordered set: $0 < \mathfrak{s} < 1 < 2 < \dots$. Also, we define the absolute value of $\mathfrak{s}$ to be 1 so that $|\eta_t(x)|$ denotes the number of particles at any site x. Define $[[\eta_t(x)]]$ to be the number of active particles at site x and at time t.

We need a few more definitions to avoid errors. We define $\mathfrak{s} + 1 = 2$ to represent the transition $S + A \to 2A$ i.e. the sleeping particle came in contact with an active one and became an active particle. This transition happens at rate ∞. Also, define $1 \cdot \mathfrak{s} = \mathfrak{s}$ and $n \cdot \mathfrak{s} = n$ to represent the transitions $A \to S$ and $(n-1)A + S \to nA$. Again, define $\mathfrak{s} - 1 = 0$ and $\mathfrak{s} \cdot \mathfrak{s} = \mathfrak{s}$ to represent the transitions that the sleeping particle is waken up by force and removed from the site and the sleeping clock alarm rings again for the sleeping particle which doesn't change anything respectively.

2.2 Poisson clocks.

The jump rate is 1 and sleeping rate is λ and so at each site x, the rate at which the the poisson clock of the vertex x (the combination of all the sleeping and jumping clocks of all the particles present at the vertex) rings is $(1 + \lambda)[[\eta_t(x)]]$. However, creating the Poisson clock in this manner means that we need to know the total number of particles at each vertex at all times. We need a method to avoid this issue so that we can reduce the time-complexity cost.

Hence, we choose to generate the Poisson clocks in the following manner: We first draw an infinite and independent collection of Poisson Point processes (θ_i) of intensity $1 + \lambda$ for $i \in \mathbb{N}$. Each of these processes θ_i takes values in R_+. Now, one might wonder that if we start with some initial configuration η_o with $[[\eta_0]]$ number of particles, then why do we need an infinite number of clocks? The initial configuration η_0 is random and hence we don't know how many particles are going to be there. Drawing the sequence of clocks (θ_i) in this manner makes the clocks independent of the number of the particles. However, after drawing this sequence, we are only going to use the Poisson point processes with indices $i \leq [[\eta_0]]$. Each of these Poisson Point processes is a combined clock i.e. sleep alarms and jump alarms in one clock.

At the first introduction of Poisson clocks, we had said that each particle has 2 clocks - one for sleeping and one for jumping. We still want to maintain that. Hence, to do so, we now say that when one of these clocks θ_i ring, we choose a particle at random (uniform distribution) and with probability $\frac{\lambda}{1+\lambda}$, the sleep instruction is implemented if it is alone at the vertex (if it is not, this instruction is overridden). With probability $\frac{1}{1+\lambda}$, the jump instruction is implemented (overridden if the particle is asleep). Note that the effect of each particle having two 'personal' Poisson clocks and everyone sharing the clocks is same. Moreover, in the former approach, we need to keep track of more information. Hence, the latter is better.

Lemma 1. *When the Poisson clock θ_i rings, with probability $\frac{\lambda}{1+\lambda}$, the sleep instruction is implemented and with probability $\frac{1}{1+\lambda}$, the jump instruction is implemented.*

Proof. Now, θ_i is a Poisson Point process and is created by the superposition of two independent Poisson Point processes $(S_{u,v})$ and $(J_{u,v})$ with rates λ and 1 respectively. Hence, by the theory of marking, we get that $(S_{u,v})$ must have rate $p_1(1 + \lambda)$ and $(J_{u,v})$ must have rate $p_2(1 + \lambda)$, where p_1 is probability of that the sleep instruction is implemented and p_2 is the probability that the jump instruction is implemented. Solving for p_1 and p_2, we get that $p_1 = \frac{\lambda}{1+\lambda}$ and $p_2 = \frac{1}{1+\lambda}$. $\square$

2.3 Toppling formalism.

In [4], T Rolla has given has extensive information on Toppling formalism. Inspired from that, we now define two operators τ_{xy} and $\tau_{x\mathfrak{s}}$. The first operator denotes that the particle at vertex x is toppled to the neighbouring vertex y. The second operator denotes that the particle at vertex x is put to sleep. So, for any configuration η, the effects of the operators are:

$$\tau_{xy}\eta(z) = \begin{cases} \eta(x) - 1, & \text{if } z = x \\ \eta(y) + 1, & \text{if } z = y \\ \eta(z), & \text{otherwise} \end{cases} \qquad \tau_{x\mathfrak{s}}\eta(z) = \begin{cases} \mathfrak{s} \cdot \eta(x), & \text{if } z = x \\ \eta(z), & \text{otherwise} \end{cases}$$

Vertex x is said to be unstable for the configuration η whenever $\eta(x) \geq 1$. We can make an unstable vertex stable by toppling the particles there by applying τ_{xy} and $\tau_{x\mathfrak{s}}$. Toppling an unstable vertex is ***legal***. However, if there is just a sleeping particle at x, then we accept for now that we can topple it as well i.e. we accept the transition $\mathfrak{s} - 1 = 0$. In this case, we say that the toppling is ***acceptable***.

Consider the field of instructions $\tau = (\tau^{x,j})_{x\in\mathbb{Z}^d, j\in\mathbb{N}}$. This looks very similar to τ_{xy} 'kind of thing' that we introduced earlier. In this case, for each vertex x, we have a set of topplings to execute in a particular order indexed by j. However, we don't have the information about the the neighbouring vertex where the particle goes after toppling. This explicit information is not required while calculating several important quantities such as final particle positions and total occupation times.

So, let τ be any such field of instructions. We will work out everything with respect to this field of instructions. Let h be the vector which counts the number of topplings that have been executed for each vertex x till now. We define $h = 0$ at time $t = 0$. Now, we define the toppling operator Φ_x.

The toppling operator Φ_x takes two inputs: configuration and the vector h and spits out the updated configuration and the updated vector h after toppling particle at x. So,

$$\Phi^\tau_x(\eta, h) = (\tau^{x,h(x)+1}\eta, h + \delta_x).$$

Note that the index of the instruction from τ being executed for any vertex x is same as the 'number of topplings done at x till now' plus 1. So, the $h(x) + 1$-th information for the vertex x is being executed. Φ^τ_x is legal if $\eta_x \geq 1$ before using the toppling operator and similarly, it is acceptable if $\eta_x \geq \mathfrak{s}$ before execution.

Suppose we are given an ordered list of vertices $\alpha = (x_1, x_2, \ldots, x_j)$. Here, this list is telling us that first the toppling operator should act on x_1, then on x_2, and so on. These vertices need not be distinct. But there is a small restriction. We are seeing this vector with respect to the field τ. Suppose x_1 occurs multiple times in the vector α. Then, on the first occurrence, $(\tau^{x,1})$ is executed, on the second occurrence, $(\tau^{x,2})$ is executed and so on. So, the execution must happen according to the order defined in τ.

Now, we want to define: $\Phi^\tau_\alpha = \Phi^\tau_{x_j}\Phi^\tau_{x_{j-1}}\ldots\Phi^\tau_{x_1}$ but there is a small issue. It may happen that after executing $\Phi^\tau_{(x_1,x_2,\ldots,x_k)}$ on η, we may find that in the configuration $\Phi^\tau_{(x_1,x_2,\ldots,x_k)}\eta$, x_{k+1} may have no particle at all or a sleeping particle. We want to avoid the first case and in the second case, we can consider removing a sleeping particle as an acceptable toppling. So, we get that Φ^τ_α is legal (acceptable) only when $\Phi^\tau_{x_l}$ is legal (acceptable) for $\Phi^\tau_{(x_1,x_2\ldots,x_{l-1})}\eta$ for each $l = 1, 2, \ldots, j$. In this case, we say that α is τ legal (acceptable). If $\Phi^\tau_\alpha(\eta, 0) = (\eta', h)$ and η' is stable in $U \subset \mathbb{Z}^d$, we say that α stabilizes η in U.

Now, we define the odometer function m. $m_\alpha = m(\alpha)$ gives the number of times any site is toppled while applying Φ^τ_α i.e. the number of time any vertex appears in the vector α. So, $m \in \mathbb{N}_0^{\mathbb{Z}^d}$. Let $U \subset \mathbb{Z}^d$. Now, we define the ododmeter of η in U:

$$m^\tau_{U,\eta} = \underset{\alpha\subset U,\ \alpha \text{ legal}}{sup} m_\alpha$$

So, $m^\tau_{U,\eta}$ takes an initial configuration η and a subset U as inputs and gives the largest possible τ-legal odometer function where toppling is only allowed on the vertices of U. We will see that this largest odometer function is the necessary number of topplings needed to stabilise η in U.

Also, we define:

$$||m||_A = \sum_{x\in A} m(x).$$

The Diaconis-Fulton leads to some very useful properties which are hard to see otherwise. We will use V in place of $\mathbb{Z}^d$ to make the equations look a bit less cluttered.

Monotonicity lemma 1. *Suppose η and η' are configurations in $\mathbb{N}^V_{\mathfrak{s}}$ such that $\eta' \leq \eta$ and $U \subset V$, then $m^\tau_{U,\eta} \geq m^\tau_{U,\eta'}$.*

This lemma is just telling us that if we have two configurations such that one contains more particles than the other at each site then for any subset U in the lattice V, we need more topplings at each site in U to stabilise the configuration with more number of particles in U. This lemma is very powerful as it doesn't assume anything about the type of topplings and order of the topplings. However, this holds only when the same field of instructions is used to stabilise both the configurations.

Monotonicity lemma 2. *Suppose α is an acceptable sequence of topplings that stabilizes η in $U \subset V$ and β is any legal sequence of topplings in U for the same configuration and field of instructions, then $m_\alpha \geq m_\beta$.*

Abelianness lemma 1. *If α and β are τ-acceptable sequences of topplings for some configuration η such that $m_\alpha = m_\beta$, then $\Phi_\alpha \eta = \Phi_\beta \eta$.*

Abelianness lemma 2. *If α and β are τ-legal sequences of topplings contained in some subset U of V such that they both stabilize η in U, then $m_\alpha = m_\beta = m^\tau_{U,\eta}$. By **Abelianness lemma 1**, we get that $\Phi^\tau_\alpha(\eta, 0) = \Phi^\tau_\beta(\eta, 0)$.*

This lemma is telling us that the odometer function of the legal sequences lying in some subset U stabilizing η in that subset is unique. Now, let $A \subset U$. Then using Abelianness lemma 2, we get that $||m^\tau_{U,\eta}||_A$ is well defined due to uniqueness and is $\sum_{x \in A} m^\tau_{U,\eta}(x)$.

If $||m^\tau_{U,\eta}||_U < \infty$, then we say that η can be stabilised in U using finite number of topplings. However, if this is equal to ∞, then we say that η cannot be stabilised in U. In the former case, we use the notation $\sigma^\tau_{U,\eta}$ to represent the final unique U-stabilised configuration obtained by only toppling the sites in U.

Theorem 1:

In a Markov chain with finite state space, there is always at least one closed class. The time to enter one of these closed classes starting from any state is almost surely finite.

Intuition. Consider any Markov chain with finite state space. Now, all the states cannot be transient as if this were the case, then there would not have been anywhere to go after finally leaving all the states. So, there is at least one recurrent class/closed class as it can be shown that with probability 1 after a finite time a transient state is never visited again. As the chain has finite number of states so the sum of all these finite leaving times of transient states must be finite. Hence, almost surely there is a finite time when the chain finally enters a closed class starting from any transient state. If we are already starting from a closed class, there is nothing to show. □

Remark 2.1 *Suppose, V is finite. Hence, the Markov chain comprising of all the possible configurations will be finite. If the number of particles is less than or equal to the number of vertices in V, then we see that the Markov chain has exactly 2 kinds of classes: each of the stabilizing configurations form a closed class and the rest of the configurations form a single large transient class.* ***So, the system stabilizes after a finite number of topplings almost surely.*** *Hence, given the field of instructions τ and an initial configuration η, there can be only one possible stabilising configuration. All other stabilising configurations will have probability 0 by Abelianness lemma 2. We denote the unique final configuration obtained in this case by σ^τ_η.*

3 THE DEVELOPMENTS

Before moving to prove our theorem and building further techniques, it will be helpful to pen down the facts and progress about the ARW model in sequential order:

- Rolla and Sidoravicius: The probability that the stays active always is increasing in μ and satisfy the 0-1 law under the measure $\mathbb{P}_\mu^\lambda$ (measure introduced in 3.4).
- The value of $\mu_c(\lambda)$ is same when the initial configuration is drawn from any translation-invariant ergodic measure with average density μ.
- Stauffer and Taggi [ST18]: When the sleeping rate λ is small enough, we have $\mu_c(\lambda) < 1$ for $\mathbb{Z}^d$ where $d \geq 3$. Also, $\mu_c(\lambda) \geq \frac{\lambda}{\lambda+1}$ for $\mathbb{Z}^d$, where $d \geq 1$ and $\lambda \in [0, \infty)$.
- Taggi [Tag19]: $\mu_c(\lambda) < 1 \ \forall \ \lambda \in (0, \infty)$ on $\mathbb{Z}^d$ with $d \geq 3$.
- Basu, Ganguly, Hoffman: $\mu_c(\lambda) < 1$ for small enough λ on $\mathbb{Z}^d \ \forall \ d \geq 1$.
- Asselah, Rolla, Schapira [ARS19]: $\mu_c(\lambda) = \mathcal{O}(\sqrt{\lambda})$ for λ small enough.
- Hoffman, Richey and Rolla [HRR20]: for any λ, $\mu_c(\lambda) < 1$ in dimension 1.
- In dimension 2, Forien and Gaudilliere have shown that $\mu_c < 1$ when λ is small enough.

We now mention 4 crucial theorems stated in [2].

3.1 Existence of a Universal Phase transition:

Rolla, Sidoravicius and Zindy proved the existence of a phase transition in the Activated Random Walk model in all the lattices $\mathbb{Z}^d$, where $d \geq 1$.

Theorem 2:

For every lattice $\mathbb{Z}^d$, where $d \geq 1$ and every sleep rate $\lambda \in (0, \infty]$, there exists a critical value $\mu_c(\lambda)$ such that whenever we have any translation-ergodic intial distribution with no sleeping particles and an average density of the particles μ, the Activated Random Walk model on $\mathbb{Z}^d$ fixates almost surely if $\mu < \mu_c(\lambda)$ and stays active almost surely if $\mu > \mu_c(\lambda)$.

Gideon Amir, Ori Gurel-Gurevich and Eric Shellef have shown that $\mu_c(\lambda) \leq 1$ always. This tells us that even in the Internal Diffusion Limited Aggregation (IDLA) model where sleep rate $\lambda = \infty$, there exists a critical value $\mu_c(\infty) \leq 1$ such that whenever $\mu > \mu_c(\infty)$, the system stays always active.

Also, note that though we are dealing with the lattice where we place independent Bernoulli(μ) number of particles at each vertex, the theorem given here is more general i.e. it is applicable to any translation-ergodic initial configuration with no sleeping particles.

3.2 Existence of a non-trivial active phase in all dimensions:

Theorem 3:

There exists a constant $\kappa_d > 0$ dependent only on the dimension of the lattice such that $\forall$ $\mu \in (0,1)$ and $\lambda > 0$ satisfying

$$\kappa_d(2d\lambda)^\mu < \mu^\mu(1-\mu)^{1-\mu} \tag{1}$$

for all translation-ergodic initial configurations on $\mathbb{Z}^d$ with no sleeping particles and an average density of particles μ and sleep parameter λ, the Activated Random Walk model almost surely always stays active.

Suppose, we choose any small value of μ. Then, we can choose λ sufficiently small satisfying this condition. Hence, for such a small valued-pair, the system will almost surely always remain active and so in any dimension, there always exists a non-trivial phase transition.

3.3 Translating the $\mathbb{Z}^d$ problem to many problems of discrete toruses:

Consider the torus $\mathbb{Z}_n^d = (\mathbb{Z}/n\mathbb{Z})^d$. This is a discrete torus. Consider the regular torus $(\mathbb{R}/n\mathbb{Z})^d$ but where all the points with integer coordinates light up. This mesh of lit bulbs forms the discrete torus $\mathbb{Z}_n^d$. Now, we start with an initial configuration on this torus such that the number of particles is not greater than the number of vertices on this torus. Consider the Markov chain made of all the possible configurations made of sleeping and active particles. There will be a finite number of such configurations as $\mathbb{Z}_n^d$ has finite number of vertices. Also, there exist at least one closed state. Hence, using Theorem 1, we get that the stabilisation time (fixation time), $\mathcal{T}_n$, is almost surely finite. We also denote the probability measure for the Activated Random Walk by $\mathbb{P}_\mu^\lambda$, which has sleep rate λ and starts with the initial configuration where very vertex has independent Poisson (μ) number of particles. We will talk more about this measure and the construction of this measure after this section.

Theorem 4:

In every dimension $d \geq 1$, whenever $\mu \in (0,1)$ and $\lambda > 0$ satisfy condition (1) given in Theorem 3, $\exists$ $c > 0$ such that the stabilisation times $\mathcal{T}_n$ of the Activated random Walks on the respective toruses $\mathbb{Z}_n^d$ satisfy

$$\mathbb{P}_\mu^\lambda(\mathcal{T}_n < e^{cn^d}) < e^{-cn^d}. \tag{2}$$

So, whenever condition (1) is satisfied in any dimension d, the stabilisation times of all the n-discrete toruses $\mathbb{Z}_n^d$, where $n \in \mathbb{N}$, is exponentially large with an overwhelming probability. Though, we have stated this theorem for the case when the initial distribution starts with independent Poisson number of particles at each vertex, this theorem can be easily changed to our case i.e. independent Bernoulli number of particles at each vertex.

Theorem 5:

In any dimension $n \geq 1$, for every finite sleep rate $\lambda > 0$ and every density $\mu < \mu_c(\lambda)$, condition (2) stated in Theorem 4 about the stabilisation times doesn't hold.

Remark 3.1 *This theorem combined with Theorem 4 gives us a way to transform the original ARW problem to infinite number of 'easy' problems of discrete toruses. This works as follows: In any dimension $d \geq 1$, for any sleep rate $\lambda > 0$ and $\mu < \mu_c$, by theorem (5), we have that condition (2) doesn't hold. Hence, if condition (2) holds for some $\mu \in (0,1)$ and $\lambda > 0$, then $\mu \geq \mu_c(\lambda)$. This transforms the original ARW problem on the lattice $\mathbb{Z}^d$ to infinite number of easy problems on the n-toruses i.e. checking condition (2) on toruses $(\mathbb{Z}_n^d)_{n\in\mathbb{N}}$.*

This also gives us a way to prove Theorem (2). Condition (1) $\implies$ condition (2). Hence, if condition (1) is satisfied then $\mu \geq \mu_c(\lambda)$. Now, condition (1) is a strict inequality and so the possible values of μ must be an open set. Hence, $\mu > \mu_c(\lambda)$ if condition (1) is satisfied. Using theorem (2) at the last step, we have now proved theorem (3).

3.4 Construction of the measure $\mathbb{P}_\mu^\lambda$:

Consider the field of instructions $(\tau^{x,j})_{x\in V, j\in\mathbb{N}}$. First we want to construct the measure $\mathbb{P}^\lambda$ on the set of all the field of instructions. We assume that all the constituent instructions $\tau^{x,j}$ are independent of each other. Now, using the claim introduced in the subsection on Poisson clocks, we get that for every $x \in V$ and $j \in \mathbb{N}$,

$$\mathbb{P}^\lambda(\tau^{x,j} = \tau_{x,\mathfrak{s}}) = \frac{\lambda}{1+\lambda}$$

and for every neighbour y of x,

$$\mathbb{P}^\lambda(\tau^{x,j} = \tau_{xy}) = \frac{1}{(1+\lambda)D}.$$

where D is the degree of the vertex x.

Since all the constituent instructions are independent of each other so $\mathbb{P}^\lambda((\tau^{x,j})_{x\in V, j\in\mathbb{N}})$ can be calculated by adding the probabilities of the constituent instructions.

Now, consider all possible initial configurations with independent Poisson number of particles with density μ at each vertex of the discrete torus. The measure $\mathbb{P}_\mu$ measures the probability of each such configuration in the usual way. So, it is a probability measure on the measurable space $(\mathbb{N}_\mathfrak{s}^V, \mathcal{P}(\mathbb{N}_\mathfrak{s}^V))$, where $\mathcal{P}(\mathbb{N}_\mathfrak{s}^V))$ denotes the power set. Now, we define $\mathbb{P}_\mu^\lambda$ to be the product measure $\mathbb{P}^\lambda \otimes \mathbb{P}_\mu$. So, any initial configuration η_0 and field of instructions τ are independently distributed according to the measures $\mathbb{P}^\lambda$ and $\mathbb{P}_\mu$ respectively and jointly their probability is given by $\mathbb{P}^\lambda \otimes \mathbb{P}_\mu$.

3.5 Complete definition of the Activated Random Walk.

Suppose $G = (V, E)$ is a vertex-transitive graph. Given a random field of instructions $\tau = (\tau^{x,j})_{x\in V, j\in\mathbb{N}}$ distributed according to $\mathbb{P}^\lambda$, the Activated Random Walk is a stochastic process $(\eta_t, h_t)_{t\geq 0}$ where:

1. η_0 follows a translation-ergodic distribution on $\mathbb{N}_\mathfrak{s}^V$ with some finite density μ
2. η_0 is independent of τ
3. $h_0 = 0$
4. Each site $x \in V$ has a transition rate of $(1+\lambda)|\eta_t(x)|1_{\eta_t(x)\geq 1}$ (combination of all the Poisson clocks at x) from (η_t, h_t) to $\Phi_x^\tau(\eta_t, h_t)$.

4 PROOF OF THEOREM 4

Our goal is to show that in any dimension $d \geq 1$, whenever $\mu \in (0,1)$ and $\lambda > 0$ and they satisfy condition (1), then $\exists \quad c > 0$ such that $\mathbb{P}^{\lambda}_{\mu}(\mathcal{T}_n < e^{cn^d}) < e^{-cn^d} \quad \forall \quad n \in \mathbb{N}$. So, if we manage to get a lower bound for $\mathcal{T}_n$ whose probability of being less than e^{cn^d} is less than e^{-cn^d}, then we are done. Consider any discrete n-torus $\mathbb{Z}^n_d$.

4.1 Steps of the proof of theorem 4.

Step 1: *Fixing the initial number of particles.*

Choose $\mu' < \mu$ still satisfying condition (1). We will prove later on that with an overwhelming probability, there are at least $k = \lceil \mu' n^d \rceil$ particles. Now, using the Monotonicity lemma 1, it suffices to show the required bound on the stabilisation time $\mathcal{T}_n$ when there are exactly k particles on the n-torus. We will come to this step again with a rigorous proof.

Step 2: *Fixing a subset on the n-torus of size k.*

All the k particles on the the n-torus eventually go to sleep on exactly k sites. We fix such a set A of size k. Now, if we have the probability of $\mathcal{T}_n$ being less than e^{cn^d} when they all eventually settle on A, then we can just add this probability over all possibilities of A i.e. $\binom{n^d}{k}$-many to get the total probability.

Step 3: *Particles not allowed to sleep anywhere else.*

Now, if we change the probability measure $\mathbb{P}^{\lambda}$ so that particles can longer sleep on the vertices outside A i.e. the probability that $\tau^{x,j} = \tau_{x,\mathfrak{s}}$ is 0. We denote the probability of any field of instructions τ under this effect as $\mathbb{P}^{\lambda,A}(\tau)$. Then, for every x outside A,

$$\mathbb{P}^{\lambda,A}(\tau^{x,j} = \tau_{x,\mathfrak{s}}) = 0$$

and for every neighbour y of x

$$\mathbb{P}^{\lambda,A}(\tau^{x,j} = \tau_{xy}) = \frac{1}{D}$$

and for every x inside A, it is same as $\mathbb{P}^{\lambda}$.

Under this effect, just the sleep instructions on vertices outside of A are overridden. In the original case also, the fixating set was A so the sleeping particles outside A in the original scenario were going to be waken up after some time. Hence, under this new measure, we have stochastically reduced the time to finally arrive on the sites of A and sleep. Hence,

Theorem 6:

For each initial configuration η of active particles on the torus $\mathbb{Z}^d_n$, $\forall\ A \subset \mathbb{Z}^d_n$ with $|A| = |\eta|$ and for every $M \in \mathbb{N}$,

$$\mathbb{P}^{\lambda}(\{||m^{\tau}_{\eta}||_A < M\} \cap \{\sigma^{\tau}_{\eta} = \mathfrak{s}1_A\}) \leq \mathbb{P}^{\lambda,A}(||m^{\tau}_{\eta}||_A < M)$$

So, $\mathbb{P}^{\lambda,A}(||m^{\tau}_{\eta}||_A < M)$ is an upper bound on our required probability. Though we have tried to present the intuition here but a rigorous proof of this is given in [2].

Step 4: *Starting with the initial configuration 1_A.*

We further aim to find an upper bound on this probability. One way to do this is not to start with any arbitrary configuration but with 1_A. The idea behind this is that if we start with any arbitrary initial configuration with $|A|$ number of active particles such that they are all eventually going to sleep on the sites of A, then concentrating on any particle, we note that there is a finite number of topplings after which the particle comes and sits on a vertex of A. Using Abelianness lemma 1, we know that *the order of topplings don't matter as long as the odometer functions and the field of instructions are same.* Hence, we can now choose the next particle after making it sit on some vertex of A. We do this same thing with the k-th particle as well. Hence, after a finite number of topplings, all the particles can be found to be sitting on A, all in active form. So, after several topplings, we are starting with the initial configuration 1_A.

Hence, starting with the initial configuration 1_A directly should stochastically take less number of topplings. We make this fact precise by proving the following:

Theorem 7:

Suppose, we have fixed the set A and let $\eta : \mathbb{Z}_n^d \to \mathbb{N}_{\mathfrak{s}}$ be any initial configuration such that $|\eta| = |A|$ i.e. η has $|A|$ number of active and total particles. Then, under the measure $\mathbb{P}^{\lambda,A}$, $||m_\eta^\tau||_A$ stochastically dominates $||m_{1_A}^\tau||_A$ i.e. $\forall \quad M \in \mathbb{N}$

$$\mathbb{P}^{\lambda,A}(||m_\eta^\tau||_A < M) \leq \mathbb{P}^{\lambda,A}(||m_{1_A}^\tau||_A < M).$$

Before embarking on the proof of this theorem, we introduce the Strong Markov Property.

Strong Markov Property[2]: Let τ be any random field of instructions distributed according to $\mathbb{P}^{\lambda,A}$ and β be any sequence of topplings implemented till now according to τ. Let $\mathcal{F}$ be the event that β is finite. For the stabilisation of the initial configuration on the torus, we may need to carry out more topplings. Let γ be any such remaining sequence of legal topplings. Then, given $\mathcal{F}$, γ is still distributed according to $\mathbb{P}^{\lambda,A}$ and independent of the instructions revealed while constructing β.

Proof (Theorem 7). We had defined on page 5 that $m_\eta^\tau = \sup_{\alpha \subset \mathbb{Z}_n^d,\ \alpha\ \tau\text{-legal}} m_\alpha$. So, $||m_\eta^\tau||_A = \sup_{\alpha \subset \mathbb{Z}_n^d,\ \alpha\ \tau\text{-legal}} ||m_\alpha||_A$. So,

$$\mathbb{P}^{\lambda,A}(||m_\eta^\tau||_A < M) = \mathbb{P}^{\lambda,A}(\forall\ \alpha \text{ legal for}(\eta, 0), ||m_\alpha||_A < M). \tag{3}$$

Let τ be any random field of instructions distributed according to $\mathbb{P}^{\lambda,A}$. We carry out the process as stated in Step 4 to ensure that each site of A contains an active particle. We call β the legal sequence of topplings needed to reach this stage (not unique) and $\mathcal{F}$ the event that the total number of topplings done according to β is finite. Now, we supposed that $|\eta| = |A|$. Consider the corresponding Markov chain. There is just one possible closed state: each site of A containing exactly one particle. Using Theorem 1, we get that $\mathcal{F}$ happens almost surely. Suppose, $\mathcal{F}$ occurs and let β be any such sequence of legal topplings. Then, $\Phi_\beta^\tau(\eta, 0) = (1_A, m_\beta)$. Now, let γ be any 'remaining' sequence of topplings legal for $(1_A, m_\beta)$. Consider the concatenation of β and γ i.e. '$\beta + \gamma$'. Now, '$\beta + \gamma$' is $\tau-$legal for $(\eta, 0)$. Hence, $||m_\beta + m_\gamma||_A < M$ for any such γ if $||m_\eta^\tau||_A < M$.

So,

$$\begin{aligned}\mathbb{P}^{\lambda,A}(\forall\ \alpha \text{ legal for}(\eta,0), ||m_\alpha||_A < M) &\leq \mathbb{P}^{\lambda,A}(\forall\ \gamma \text{ legal for}(1_A, m_\beta), ||m_\gamma + m_\beta||_A < M|\mathcal{F}) \\ &\leq \mathbb{P}^{\lambda,A}(\forall\ \gamma \text{ legal for}(1_A, m_\beta), ||m_\gamma||_A < M|\mathcal{F})\end{aligned}$$

Now, using the Strong Markov Property, we get that conditioned on $\mathcal{F}$, such a γ is independent of instructions revealed while constructing β (hence, independent of m_β) and distributed according to $\mathbb{P}^{\lambda,A}$.

So,

$$\begin{aligned}\mathbb{P}^{\lambda,A}(\forall\ \gamma \text{ legal for}(1_A, m_\beta), ||m_\gamma||_A < M|\mathcal{F}) &= \mathbb{P}^{\lambda,A}(\forall\ \gamma \text{ legal for}(1_A, 0)), ||m_\gamma||_A < M|\mathcal{F}) \\ &= \mathbb{P}^{\lambda,A}(\forall\ \gamma \text{ legal for}(1_A, 0)), ||m_\gamma||_A < M).\end{aligned}$$

since $\mathbb{P}(\mathcal{F}) = 1$.

So, using equation (3), we get that $\mathbb{P}^{\lambda,A}(||m^\tau_\eta||_A < M) \leq \mathbb{P}^{\lambda,A}(||m^\tau_{1_A}||_A < M)$. Hence, the initial configuration 1_A provides an upper bound on the required probability. □

After step 4, our problem has become a lot simpler. We neither have to worry about any random initial configuration nor about particles sleeping outside of A. Hence, after such simplification, our first attempt should be to create an algorithm or 'random' sequence of topplings that always stabilizes the initial configuration. We do it as follows:

Step 5: *Reducing the problem to one-dimensional random walk with a negative drift*

We first order the sites of A: $(x_1, x_2, \ldots, x_k)$ where $k = |A|$.

- We start with x_1. Either the particle at x_1 goes to sleep with probability $\frac{\lambda}{1+\lambda}$.
- Or we get a loop i.e. that is the particle visits a sequence of sites both inside and outside A before coming back to x_1. Then, we topple it again and this process carries on until this particle goes to sleep.
- We then focus on the particle at vertex x_2. We do a similar thing with the particle at x_2. If there is a loop, then this particle may go to site x_1 and may wake it up. If this happens, then after the particle comes back to x_2, we start with x_1 again.
- However, we introduce a small change here. Whenever, we are dealing with the particle at the site x_k, we only care if the particle wakes up the particle at the site x_{k-1} and ignore what happens in other previous sites.
- This may not give a stabilizing sequence of topplings but will provide us with a lower bound on the number of topplings needed to stabilise.

Remark 4.1 *We will see in Theorem 8 that whatever ordering of the sites of A we choose, the time to reach the last vertex according to the algorithm described in Step 5 stochastically dominates the time to reach the final state in a Markov chain with negative drift. However, the negative drift of this* ***second chain*** *will depend on the order of the sites and on A. A natural question follows: Can we get an ordering of the sites so that irrespective of A and the ordering of the sites, we get a bound on this negative drift?*

Thankfully, the answer to this question is a Yes and we will explore this in depth in section 4.3.

Step 6: *Ordering the sleeping sites to get a negative drift on the second Markov chain*

We order the sites of any given fixation set of sites A using Greedy approach. This then helps us to show that the product of distances between the sites is bounded above by a constant which depends only on the dimension n of the torus $\mathbb{Z}_n^d$! We then use this in the expression to calculate the drift of the second Markov chain which gives us condition (2) in Theorem 4 after some computations.

We now give a rigorous proof of **step 5** and a part of **Remark 4.1.**

4.2 Coupling and reducing to a problem of random walk in one dimension.

Theorem 8:

Let $A = (x_1, , x_2, \ldots, x_k) \subset \mathbb{Z}_n^d$ be an ordered sequence of vertices with $k = |A|$ and τ be a random field of instructions distributed according to $\mathbb{P}^{\lambda,A}$. Consider the Markov chain with state space {1, ..., k+1} and transition probabilities given by:

1. $p(1,1) = \frac{1}{1+\lambda}$
2. $p(j,j+1) = \frac{\lambda}{1+\lambda} \;\; \forall j \in \{1,\ldots,k\}$
3. $p(j,j-1) = \frac{1}{1+\lambda}\mathbb{P}_{x_j}(T_{x_{j-1}} < T^+_{x_j}) \;\; \forall j \in \{2,\ldots,k\}$
4. $p(j,j) = \frac{1}{1+\lambda}\mathbb{P}_{x_j}(T^+_{x_j} < T_{x_{j-1}}) \;\; \forall j \in \{2,\ldots,k\}$
5. $p(k+1,k+1) = 1$

where $\mathbb{P}_{x_j}$ is the Probability measure associated to the symmetric random walk on the torus $\mathbb{Z}_n^d$ starting from x_j, T_{x_j} is the first hitting time of the vertex x_j, $T^+_{x_j}$ is the second hitting time of x_j and $\mathbb{P}_j^A$ is the Probability measure for the Markov chain starting from j. Then, for any $M > 0$,

$$\mathbb{P}^{\lambda,A}(||m^\tau_{1_A}||_A \le M) \le \mathbb{P}_1^A(T_{k+1} \le M)$$

Proof: Let $A = \{x_1, \ldots \; x_k\} \subset \mathbb{Z}_n^d$ with $|A| = k$ and τ be a random field of instructions distributed according to $\mathbb{P}^{\lambda,A}$.

It suffices to couple the measures $\mathbb{P}^{\lambda,A}$ and $\mathbb{P}_1^A$ in such a way that $||m^\tau_{1_A}||_A \ge T_{k+1}$ a.s. Here, $||m^\tau_{1_A}||_A$ is the total number of topplings done on A while stabilising A according to τ starting from the initial configuration $\eta = 1_A$. Now, if we can create a legal sequence of topplings α (not necessarily a stabilising sequence of topplings) such that $||m_\alpha||_A \ge T$, then using Monotonicity lemma 2, we are done.

1. Introducing the **Markov chain** on the vertices of A:

1. Set a counter variable $J(t)$ which indicates which site we have to topple in the $t-th$ step. The process starts with $J(0) = 1$ and ends after $(t-1)$ steps if $J(t) = k+1$.
2. The process will guarantee that before each t, all the particles occupy all the vertices of A and that sites with indices $i \ge J(t)$ are all active.

3. **Case 1:** At each step t, if $J(t) \leq k$, then we topple the site $x_{J(t)}$. With probability, $\frac{\lambda}{1+\lambda}$, the particle at $x_{J(t)}$ falls asleep with probability $\frac{\lambda}{1+\lambda}$.
4. In this case, we then set $J(t+1) = J(t) + 1$ and proceed with the vertex $x_{J(t)+1}$.
5. **Case 2**: If the particle jumps to its neighbour then we continue to topple the sites it until it gets back to its original site. Note that it can't sleep on any other vertex and using Theorem 1, we get that it returns back after a finite number of topplings a.s.
6. If $J(t) \geq 2$ then with probability $\mathbb{P}_{x_{J(t)}}(T_{x_{J(t)-1}} < T^+_{x_{J(t)}})$, the particle during its journey from $x_{J(t)}$ to this site visits the vertex $x_{J(t)-1}$.
7. Hence, it makes the particle there active. We ignore what it does to the particles in other vertices. In this case, after the particle returns back to the site $x_{J(t)}$, we set $J(t+1) = J(t) - 1$ and continue this process.

2. So, we have two chains:

1. Markov chain with state space comprising of all the possible values of $J(t)$ i.e. $\{1, \ldots, k+1\}$ and transition probabilities given by p.
2. Markov chain with state space $\{1, \ldots, k+1\}$ and transition probabilities given by p.

From now on we will call the Markov chain in (2) the **second Markov chain**.

3. **Coupling mechanism:** So, both the chains are same however both are in different contexts. So, we can introduce a Uniform-$[0,1]$ random variable before each step $t \geq 1$ so that we are always in the same state in both the chains.

So, if we denote the number of steps in the second Markov chain till it stops by T, then due to the coupling we have $T = T_{k+1}$. This process gives a finite (by theorem 1 again) sequence of topplings α legal for $(1_A, 0)$ such that $||m_\alpha||_A \geq T$ which implies that $||m^\tau_{1_A}||_A \geq T_{k+1}$ by Monotonicity lemma 2. $\square$

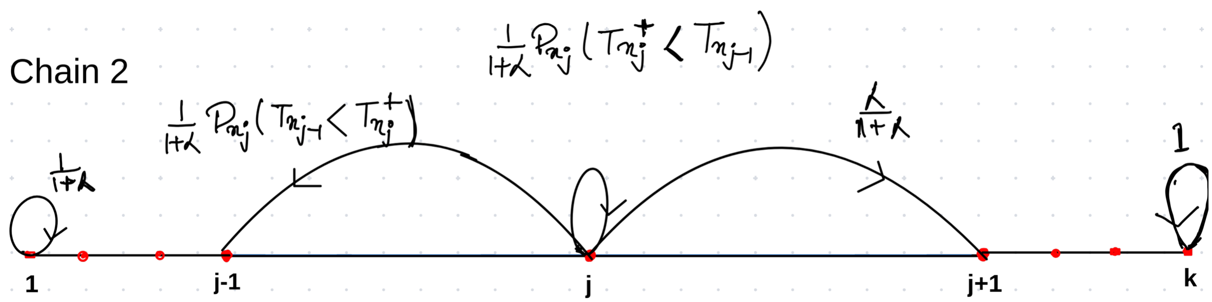


Figure 1: The one dimensional Markov chain governing the toppling procedure.

Chain 1

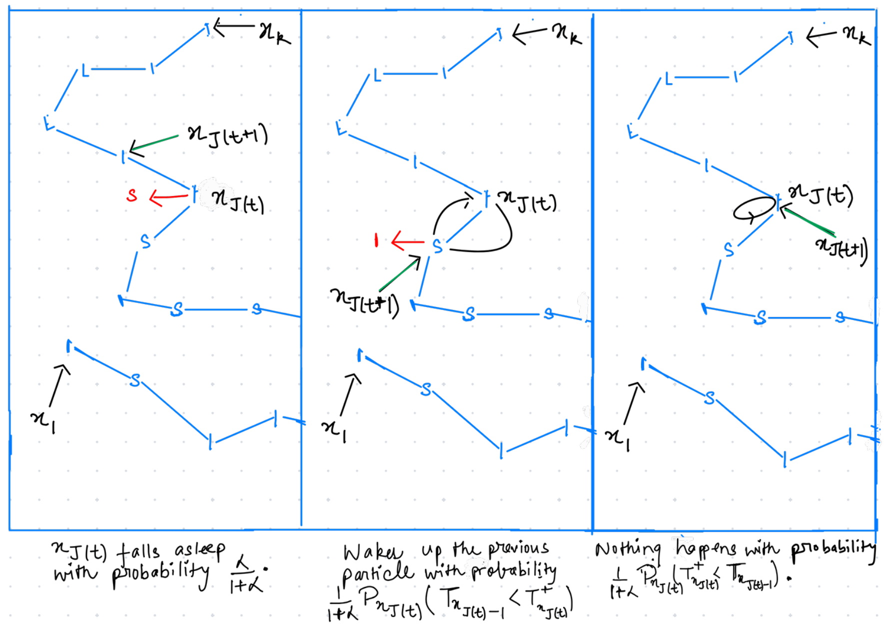


Figure 2: The chain is being viewed just after time t with 3 possibilities. Green arrow is indicating the vertex that will be toppled at time $t+1$ i.e $x_{J(t+1)}$-th vertex. Red arrow indicates the current transition.

4.3 Algorithm for ordering the sites.[2]

In Step 6 in section 4.1, we mentioned that if we order the sites according to the Greedy algorithm, then we will get an upper bound on the product of distances between the sites which depended only on the dimension and then we can use it to control the negative drift. We make it clear now.

We use the Greedy approach to create such an ordering. We first choose x_1 arbitrarily. Then, suppose we have created the ordered list $(x_1, x_2, \ldots, x_j)$. We choose the $(j+1)-$th vertex as follows: We set $A_j = A \setminus \{x_1, x_2, \ldots, x_j\}$. We define

$$x_{j+1} = \underset{x \in A_j}{\arg\min}\ d(x_j, x).$$

Also, we define $l_j = d(x_j, x_{j+1})$. Here, d is the d_1 metric but on the torus $\mathbb{Z}_d^n$. So, for any $x, y \in \mathbb{Z}_n^d$,

$$d(x,y) = min\{d(z_1, z_2) : \pi(z_1) = x, \pi(z_2) = y, z_1, z_2 \in \mathbb{Z}^d\}$$

Here, π is the map that projects points from the lattice $\mathbb{Z}^d$ onto the torus $\mathbb{Z}_n^d$.

Lemma 6. *Consider any dimension $d \geq 1$. Then, $\exists$ a universal constant $\kappa_d > 0$ which depends on only the dimension d such that for all $n \in \mathbb{N}$ and for all $A \subset \mathbb{Z}_n^d$, the distances $l_1, l_2, l_3, \ldots, l_{|A|-1}$* satisfy

$$\prod_{j=1 \text{ to } |A|-1} l_j \leq (\kappa_d)^{n^d}.$$

Proof. Fix any dimension d and n. Consider any $A \subset \mathbb{Z}_n^d$. We now order the sites according to our Greedy algorithm. Suppose $(x_1, x_2, \ldots, x_k)$ with $|A| = k$. We now define the set $\mathcal{A}_L = \{j \in \{1, \ldots, k-1\} : l_j \geq L\}$. Suppose, $k_1, k_2 \in \mathcal{A}_L$. Then, $d(x_{k_1}, x_{k_1+1}) \geq L$ and $d(x_{k_2}, x_{k_2+1}) \geq L$. Hence, $d(x_{k_1}, x_{k_2}) \geq L$ by construction. So, any two vertices with indices in $\mathcal{A}_L$ have distance at least L units between them.

Now, consider the open ball $B_d(x, r) = \{y \in \mathbb{Z}_n^d : d(x, y) < r\}$ where d is the metric we introduced while defining the algorithm. Since the distance between the vertices with indices in the set $\mathcal{A}_L$ is at least L units so the collection of balls $\{B_d(x_k, \lceil \frac{L}{2} \rceil) : k \in \mathcal{A}_L\}$ are disjoint to each other.

So, if we put an open ball of radius $\lceil \frac{L}{2} \rceil$ around every such x with indices in $\mathcal{A}_L$, then we can cover at most all the points of $\mathbb{Z}_n^d$ without any overlapping. Hence,

$$\sum_{x \in \mathcal{A}_L} |B_d(x, \lceil L/2 \rceil)| \leq |\mathbb{Z}_n^d|.$$

Now, due to the symmetry of the torus, $|B_d(x, \lceil L/2 \rceil)| = |B_d(0, \lceil L/2 \rceil)| \leq |\mathbb{Z}_n^d|$ and so

$$|\mathcal{A}_L| \leq \frac{|\mathbb{Z}_n^d|}{|B_d(x, \lceil L/2 \rceil)|} \leq \frac{|\mathbb{Z}_n^d|}{|B_d(0, \lceil L/2d \rceil)|} \tag{4}$$

as $d \geq 1$.

A natural question is: *What is the maximum value that L can take such that $\mathcal{A}_L \neq \phi$?*

Consider any torus $\mathbb{Z}_n^d$ and unwrap it so that this torus becomes a d- dimensional cube. Due to the symmetry of the torus and the opposite sides being connected in the cube, the distance of any 2 points on the torus must be less or equal to the distance of the center of the cube from any of the corners of the d- dimensional cube. To go from any of the corners to the center, we need to take $(n/2)$ steps in each direction.

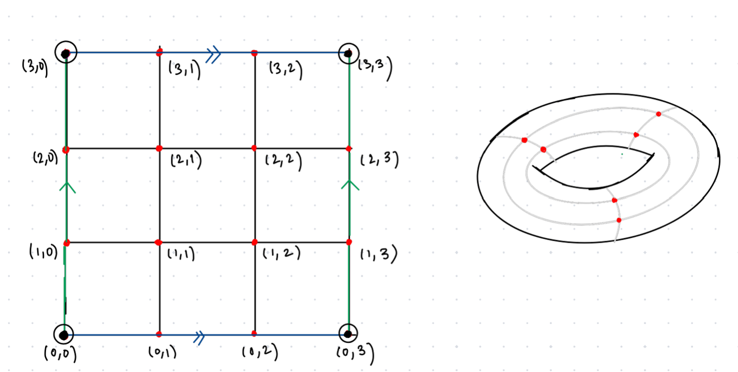


Figure 3: The discrete torus $\mathbb{Z}_3^2$ can be seen as a rectangular grid with edges identified. Alternatively, we can see visualize it as the unwrapped version of the torus. The maximum distance is between the corner point and center points.

So, according to the metric d on the torus, this distance is at most $\underbrace{n/2 + n/2 + \ldots + n/2}_{d-times} = \frac{nd}{2}$. So, $L \leq \frac{nd}{2}$ and hence, $\frac{L}{2d} \leq \frac{n}{4}$.

It is not hard to see that although the open balls $B_d(0,r)$ and $B_{d_1}(0,r)$ (the d_1 open ball in $\mathbb{Z}^d$) are different but due to the very similar nature of the metrics, both of them contain the same number of points. So,

$$|B_d(0,r)| = |B_{d_1}(0,r)| \tag{5}$$

Now, consider any two points x and y in $\mathbb{Z}^d$ such that $d_\infty(x,y) < \lceil \frac{\lceil L/2d \rceil}{d} \rceil$.Then, $d_\infty(x,y) < \frac{\lceil L/2d \rceil}{d}$. So, $d_1(x,y) < \lceil L/2d \rceil$. Now, taking $x = 0$, we get that $y \in B_{d_1}(0, \lceil L/2d \rceil)$. Hence,

$$|B_d(0, \lceil L/2d \rceil)| = |B_{d_1}(0, \lceil L/2d \rceil)| \geq |B_\infty(0, \left\lceil \frac{\lceil L/2d \rceil}{d} \right\rceil)| \tag{6}$$

where we used (5) in the first equality.

Using the inequality $\left\lceil \frac{\lceil m \rceil}{n} \right\rceil \geq \lceil \frac{m}{n} \rceil$ in (6), we get that $|B_d(0, \lceil L/2d \rceil)| \geq |B_\infty(0, \lceil L/2d^2 \rceil)| = (2\lceil \frac{L}{2d^2} \rceil - 1)^d \geq (\frac{L}{2d^2})^d$. So, we get that $|\mathcal{A}_L| \leq \frac{n^d}{(\frac{L}{2d^2})^d} = (\frac{2d^2 n}{L})^d$ by inputting the obtained bound in (4). Now,

$$\sum_{j=1 \text{ to } k-1} ln(l_j) \leq \sum_{m=0 \text{ to } \infty} (|\mathcal{A}_{2^m}| - |\mathcal{A}_{2^{m+1}}|) ln(2^{m+1})$$

This is because instead of summing over $ln(l_j)'$s, we can first partition $\mathbb{R}$ into sets of the form $[2^m, 2^{m+1})$, then try to find the number of indices whose distances lie in this range through $|\mathcal{A}_{2^m}| - |\mathcal{A}_{2^{m+1}}|$ and then multiply each of these cardinalities with the log of the maximum possible distance between the points and take the sum.

Now, rearranging the indices, we get that

$$\sum_{m=0 \text{ to } \infty} (|\mathcal{A}_{2^m}| - |\mathcal{A}_{2^{m+1}}|) ln(2^{m+1}) = \sum_{m=0 \text{ to } \infty} |\mathcal{A}_{2^m}| ln(2) \leq n^d ln(\kappa_d)$$

where κ_d is given by

$$\kappa_d = 2 \exp\left((2d^2)^d \sum_{m=0 \text{ to } \infty} \frac{m+1}{2^{md}} \right) = 2 \exp\left(\frac{(2d^2)^d}{1 - 2^{-d}} \right). \quad \square$$

4.4 Using the ordering of the sites to control the negative drift of the second Markov chain.

Theorem 9:

Let $\lambda > 0, \mu \in (0,1)$ and $A \subset \mathbb{Z}_n^d$ with $|A| = \lceil \mu n^d \rceil$. We now consider the Markov chain with the sites of A ordered using the Greedy algorithm. Let $(x_1, x_2, \ldots, x_k)$ be the ordering with $k = |A|$. Then, for any $M \in \mathbb{N}$, the absorption time of the chain satisfies

$$\mathbb{P}_1^A(T_{k+1} \leq M) \leq \frac{M}{1 \wedge (2d\lambda)} (\kappa_d (2d\lambda)^\mu)^{n^d}.$$

Proof. Suppose, $\lambda > 0, \mu \in (0,1)$ and $A \subset \mathbb{Z}_n^d$ with $|A| = \lceil \mu n^d \rceil$. Now, we modify the existing Markov chain on the states $1, 2, \ldots, k+1$ by changing the transition probability at the k-th state:

$$p(k+1,k) = p(k,k+1) = \frac{\lambda}{1+\lambda}.$$

We have just added a directed edge going from the $(k+1)$-th state to the k-th state. This modification doesn't change our interested quantity i.e. T_{k+1} but does something special: it makes our markov chain reversible.

Define

$$\nu(j) = \prod_{1 \le i < j} \frac{p(i,i+1)}{p(i+1,i)} \qquad \forall j \in \{1, 2, \ldots, k+1\} \tag{7}$$

and define $\nu(1) = 1$. Before moving forward, we introduce a small result that will be helpful to us.

Lemma 7. *Consider an irreducible Markov chain with transition probabilities $\{P_{ij}\}$. Suppose that $\{\pi_i\}$ is a set of positive numbers summing to 1 such that $\pi_i P_{ij} = \pi_j P_{ji}$ $\forall$ i, j. Then, $\{\pi_i\}$ form the stationary distribution for this chain and this chain is reversible.*

Now, we continue towards our target. Our claim is that $\nu(j)$ forms a scaled version of the stationary distribution for this markov chain. Consider any $i \in 1, 2, \ldots, k$. Then from (7),

$$\nu(i)p(i,i+1) = \nu(i+1)p(i+1,i).$$

This chain is irreducible and $\nu(j)$ is never 0. Hence, we could have applied theorem 9 if the sum would have been 1 but that doesn't stop us as we now know that the scaled version of ν is the stationary distribution and that will be enough!

One might wonder: we can introduce such a measure in case of every irreducible chain and this will make it reversible. However, that is not the case as one more important condition is that $\nu(j) > 0$ for all possible values of j. This condition ensures that not only states can increase freely but the states can also decrease with the same fluidity.

Now,

$$\begin{aligned}
\mathbb{P}_1^A(T_{k+1} \le M) &= \sum_{t=1 \text{ to } M} \mathbb{P}_1^A(T_{k+1} = t) \\
&= \sum_{t=1 \text{ to } M} \quad \sum_{1=j_0, j_1, \ldots, j_t = k+1} \quad \prod_{i=1 \text{ to } t} p(j_{i-1}, j_i) \\
&= \sum_{t=1 \text{ to } M} \quad \sum_{1=j_0, j_1, \ldots, j_t = k+1} \quad \prod_{i=1 \text{ to } t} \frac{\nu_{j_i}}{\nu_{j_{i-1}}} \, p(j_i, j_{i-1}).
\end{aligned}$$

$\prod_{i=1 \text{ to } t} \frac{\nu_{j_i}}{\nu_{j_{i-1}}} \, p(j_i, j_{i-1})$ is same as the probability of starting from $k+1$-th state and going down to state 1 by time t using the sequence of states $\{k+1, j_{t-1}, \ldots, j_1, 1\}$.

Hence,

$$\mathbb{P}_1^A(T_{k+1} \le M) = \sum_{t=1 \text{ to } M} \nu(k+1)\mathbb{P}_{k+1}^A(J(t) = 1) \le \sum_{t=1 \text{ to } M} \nu(k+1) = M\nu(k+1). \tag{8}$$

Now,

$$\begin{aligned}
\nu(k+1) &= \prod_{j=1 \text{ to } k} \frac{p(j,j+1)}{p(j+1,j)} \\
&= \prod_{j=1 \text{ to } k-1} \frac{p(j,j+1)}{p(j+1,j)} \\
&= \lambda^{k-1} \prod_{j=1 \text{ to } k-1} \frac{1}{\mathbb{P}_{x_{j+1}}(T_{x_j} < T^+_{x_{j+1}})} \\
&\leq \lambda^{k-1} \prod_{j=1 \text{ to } k-1} (2d)d(x_j, x_{j+1}) \\
&\leq (2d\lambda)^{k-1}(\kappa_d)^{n^d}.
\end{aligned}$$

Using this derived inequality in equation (8), we get that $\mathbb{P}^A_1(T_{k+1} \leq M) \leq M(2d\lambda)^{k-1}(\kappa_d)^{n^d}$. Using the fact that $k = \lceil \mu n^d \rceil$, we get the desired result. □

Remark 4.2 *This proof displays the power of reversibility of a Markov chain because without reversibility, $\mathbb{P}^A_1(T_{k+1} = t)$ is very hard to deal with!*

4.5 Proving a lighter version of Theorem (4).

Now, we are going to prove a 'lighter' version of Theorem 4 where in the initial configuration, we don't have iid Poisson(μ) number of particles. Instead, we work with any given initial configuration (not random). At this moment, we will see how all the results that we have derived so far come together to prove our result. Compared to previous results, this will be easier to prove.

Theorem 10:

In every dimension $d \geq 1$, whenever $\mu \in (0,1)$ and $\lambda > 0$ satisfy condition (1) given in Theorem (3), $\exists \; c > 0$ such that for n large enough, we have

$$\sup_{\eta \in \mathbb{N}_{\mathfrak{s}}^{\mathbb{Z}_n^d} : |\eta| \geq \lceil \mu n^d \rceil} \mathcal{P}^\lambda(||m^\tau_\eta|| < e^{cn^d}) < e^{-cn^d}.$$

Proof. Let $\lambda > 0$ and $\mu \in (0,1)$ be such that they satisfy condition (1). So,

$$\kappa_d(2d\lambda)^\mu < \mu^\mu(1-\mu)^{1-\mu}.$$

Then, $\frac{\kappa_d(2d\lambda)^\mu}{\mu^\mu(1-\mu)^{1-\mu}} < 1$. Hence, $\exists \quad c > 0$ such that $\frac{\kappa_d(2d\lambda)^\mu}{\mu^\mu(1-\mu)^{1-\mu}} < e^{-2c}$. Let η be any initial configuration (deterministic) and A the sleeping set of vertices such that $|A| = |\eta| = k = \lceil \mu n^d \rceil$.

So, the theorem states that the total number of topplings on the torus $\mathbb{Z}^d_n$ done according to the random field of instructions τ starting from the initial configuration η is exponentially large with an overwhelming probability.

Now,

$$\mathbb{P}^\lambda(||m^\tau_\eta|| \leq M) = \sum_{A:|A|=k} \mathbb{P}^\lambda(\{||m^\tau_\eta|| \leq M\} \cap \{\sigma^\tau_\eta = \mathfrak{s}1(A)\})$$

(this is very close to the form in Theorem 6. Just we need $||m^\tau_\eta||_A$ in place of $||m^\tau_\eta||$)

$$\begin{aligned}
\sum_{A:|A|=k} \mathbb{P}^{\lambda}(\{||m_{\eta}^{\tau}|| \le M\} \cap \{\sigma_{\eta}^{\tau} = \mathfrak{s}1(A)\}) &\le \sum_{A:|A|=k} \mathbb{P}^{\lambda}(\{||m_{\eta}^{\tau}||_A \le M\} \cap \{\sigma_{\eta}^{\tau} = \mathfrak{s}1(A)\}) \\
&\le \sum_{A:|A|=k} \mathbb{P}^{\lambda,A}(||m_{\eta}^{\tau}||_A \le M) \\
&\le \sum_{A:|A|=k} \mathbb{P}^{\lambda,A}(||m_{1_A}^{\tau}||_A \le M) \text{ (By Theorem 7)} \\
&\le \sum_{A:|A|=k} \mathbb{P}_1^{A}(T_{k+1} \le M) \\
&\le \sum_{A:|A|=k} \frac{M}{1 \wedge (2d\lambda)} (\kappa_d (2d\lambda)^{\mu})^{n^d} \text{ (By Theorem 8)} \\
&= \left({}^{n^d}\mathcal{C}_{\lceil \mu n^d \rceil}\right) \frac{M}{1 \wedge (2d\lambda)} (\kappa_d (2d\lambda)^{\mu})^{n^d}. \qquad (9)
\end{aligned}$$

Now, $\left({}^{n^d}\mathcal{C}_{\lceil \mu n^d \rceil}\right)$ is of the form $\left({}^{m}\mathcal{C}_{\lceil \mu m \rceil}\right)$, where $m \to \infty$.

This allows us to use Stirling's approximation by which $m! \sim \sqrt{2\pi m}\left(\frac{m}{e}\right)^m$.

Now, $m! \sim \sqrt{2\pi m}\left(\frac{m}{e}\right)^m$ when $m \to \infty$ and so using this, we get that

$$\begin{aligned}
\left({}^{m}\mathcal{C}_{\lceil \mu m \rceil}\right) &= \frac{m!}{\lceil \mu m \rceil!(m - \lceil \mu m \rceil)!} \sim \frac{m^m \sqrt{2\pi m}}{(\mu m)^{\mu m}\sqrt{2\pi \mu m}(m - \mu m)^{m-\mu m}\sqrt{2\pi(1-\mu)m}} \\
&= \mathcal{O}\left(\frac{m^m \sqrt{2\pi m}}{\mu^{\mu m} m^{\mu m}\sqrt{2\pi \mu m}\, m^{m-\mu m}(1-\mu)^{m-\mu m}\sqrt{2\pi(1-\mu)m}}\right) \\
&= \mathcal{O}\left(\frac{\sqrt{m}}{\mu^{\mu m}\sqrt{\mu m}(1-\mu)^{m-\mu m}\sqrt{(1-\mu)m}}\right) \\
&= \mathcal{O}\left(\frac{1}{\mu^{\mu m}(1-\mu)^{m-\mu m}\sqrt{m}}\right).
\end{aligned}$$

So, $\exists \quad 0 < C < \infty$ such that $\left({}^{m}\mathcal{C}_{\lceil \mu m \rceil}\right) \le \frac{C}{\mu^{\mu m}(1-\mu)^{m-\mu m}\sqrt{(1-\mu)m}}$. Substituting this in (9), we get that

$$\mathbb{P}^{\lambda}(||m_{\eta}^{\tau}|| \le M) \le \left(\frac{C}{\mu^{\mu n^d}(1-\mu)^{n^d - \mu n^d}\sqrt{n^d}}\right) \frac{M}{1 \wedge (2d\lambda)} (\kappa_d (2d\lambda)^{\mu})^{n^d}$$

Taking $M = e^{cn^d}$, we get that

$$\mathbb{P}^{\lambda}(||m_{\eta}^{\tau}|| \le e^{cn^d}) \le \frac{C}{(1 \wedge (2d\lambda))n^{d/2}} \left(\frac{e^c \kappa_d (2d\lambda)^{\mu}}{\mu^{\mu}(1-\mu)^{1-\mu}}\right)^{n^d}.$$

Here, we used the same c for which we had earlier shown that $\frac{\kappa_d(2d\lambda)^{\mu}}{\mu^{\mu}(1-\mu)^{1-\mu}} < e^{-2c}$. Hence, using this relation, we get that

$$\mathbb{P}^{\lambda}(||m_{\eta}^{\tau}|| \leq e^{cn^d}) \leq \frac{C}{(1 \wedge (2d\lambda))n^{d/2}} e^{-cn^d}.$$

Choosing n large enough, we can make $(1 \wedge (2d\lambda))n^{d/2} \geq 1$. Also, this doesn't depend on the initial configuration. Hence, for n large enough,

$$\underset{\eta \in \mathbb{N}_{\mathfrak{s}}^{\mathbb{Z}_n^d}: |\eta| \geq \lceil \mu n^d \rceil}{sup} \mathcal{P}^{\lambda}(||m_{\eta}^{\tau}|| < e^{cn^d}) < e^{-cn^d}.$$

by Monotonicity lemma 1. $\square$

Remark 4.3 *Here, we had with us a key assumption that made it simple for us. The assumption was that* $|\eta| > \lceil \mu n^d \rceil$.

4.6 Concluding the proof of Theorem (4).

Now, we wish to consider the case where η is any initial configuration with iid Poisson(μ) number of particles at each site. We claim that when n is very large, with overwhelming probability, the number of particles is always $\geq \mu n^d$ in the initial configuration with iid Poisson(μ) number of particles at each site. More precisely,

Lemma 8: *Let* $\lambda \in (0, \infty), \mu \in (0, 1)$ *be such that they satisfy condition (1) and* η_0 *be any initial configuration with iid Poisson(μ) number of particles at each site. Consider any* $0 < \mu' < \mu$ *still satisfying condition (1) in Theorem (3). Then,* $\exists \quad c > 0$ *such that*

$$\mathbb{P}_{\mu}(|\eta_0| < \mu' n^d) \leq e^{-cn^d}.$$

Proof. $\mathbb{P}_{\mu}(|\eta_0| < \mu' n^d) = \sum\limits_{j < \mu' n^d} \frac{(\mu n^d)^j}{j!} e^{-\mu n^d}$.

Also, $\left(\frac{(\mu n^d)^{j+1}}{(j+1)!}\right) / \frac{(\mu n^d)^j}{j!} = \frac{\mu n^d}{j+1} \geq 1$ since $j < \mu' n^d$ and n is large. Hence, $\frac{(\mu n^d)^j}{j!}$ increases in j. So,

$$\begin{aligned}
\mathbb{P}_{\mu}(|\eta_0| < \mu' n^d) &\leq e^{-\mu n^d} \left\lfloor \mu' n^d \right\rfloor \frac{(\mu n^d)^{\mu' n^d}}{(\mu' n^d)!} \\
&\sim e^{-\mu n^d} (\mu' n^d) \frac{(\mu n^d)^{\mu' n^d}}{(\mu' n^d)^{\mu' n^d} e^{-\mu' n^d} \sqrt{2\pi \mu' n^d}} \text{when } n \to \infty \text{ (By Stirling's)} \\
&= e^{-n^d(\mu - \mu')} \sqrt{\frac{\mu' n^d}{2\pi}} \left(\frac{\mu}{\mu'}\right)^{\mu' n^d} \\
&= e^{-n^d \left(-\mu' \log(\frac{\mu}{\mu'}) + (\mu - \mu')\right)}.
\end{aligned} \tag{6}$$

Now, we claim that $c = \left(-\mu' \log(\frac{\mu}{\mu'}) + (\mu - \mu')\right) > 0$ whenever $0 < \mu' < \mu$. If we can show that $c' = \left(\frac{\mu}{\mu'} - \log\left(\frac{\mu}{\mu'}\right)\right) > 1$, then that should be sufficient. Now, $\mu' = \mu + \epsilon$ for some $\epsilon > 0$. Putting this in the formula for c', we get that we have to show that $c' = 1 + \frac{\epsilon}{\mu'} - \log(1 + \frac{\epsilon}{\mu'}) > 1$.

Claim 1: $x - \log(1 + x) > 0$ *whenever* $x > 0$.

Proof. Let $f(x) = x - \log(1+x)$. Then, $f'(x) = \frac{x}{1+x}$. Now, $f'(x) > 0 \quad \forall x > 0$. So, f is strictly increasing. Now, $f(0) = 0$. Hence, $f(x) > 0 \quad \forall x > 0$.

So, using this result, we get that $\frac{\epsilon}{\mu'} - \log(1 + \frac{\epsilon}{\mu'}) > 0$ and hence, $c > 0$. So, we have proved **Lemma 7** as well. $\square$

Now, proving Theorem 10 in the case where the initial configuration has less than $\mu' n^d$ particles has become easier as we have shown through Lemma 7 that with overwhelming probability, the number of particles is always $\geq \mu' n^d$. When this is the case, we can simply apply Theorem 10. We make this formal here:

Theorem 11:

In every dimension $d \geq 1$, for every $\mu \in (0,1)$ and $\lambda > 0$ satisfying condition (1) in Theorem (3), for any initial configuration η_0 distributed according to $\mathbb{P}_\mu$ and a random field of instructions τ distributed according to $\mathbb{P}^\lambda$, $\quad \exists \quad c > 0$ such that for n large enough, the following holds

$$\mathbb{P}_\mu^\lambda(||m_{\eta_0}^\tau|| < e^{cn^d}) < e^{-cn^d}.$$

Proof. Now for any $c > 0$, $\mathbb{P}_\mu^\lambda(||m_{\eta_0}^\tau|| < e^{cn^d}) = \sum_{\eta \in \mathbb{N}^{\mathbb{Z}_n^d}} \mathbb{P}_\mu(\eta_0 = \eta) \mathbb{P}^\lambda(||m_\eta^\tau|| < e^{cn^d}) \leq \mathbb{P}_\mu(|\eta_0| < \mu' n^d) + \underset{\eta \in \mathbb{N}_{\mathbf{s}}^{\mathbb{Z}_n^d} : |\eta| \geq \mu' n^d}{sup} \mathcal{P}^\lambda(||m_\eta^\tau|| < e^{cn^d})$.

We used the fact that $\mathbb{P}^\lambda \otimes \mathbb{P}_\mu = \mathbb{P}_\mu^\lambda$ in the first equality.

Now, from Lemma 7, $\mathbb{P}_\mu(|\eta_0| < \mu' n^d) \leq e^{-c_1 n^d}$ for some $c_1 > 0$. Also, from Theorem 10, $\underset{\eta \in \mathbb{N}_{\mathbf{s}}^{\mathbb{Z}_n^d} : |\eta| \geq \mu' n^d}{sup} \mathcal{P}^\lambda(||m_\eta^\tau|| < e^{c_2 n^d}) < e^{-c_2 n^d}$.

Now, choose $c < \min(c_1, c_2)$ sufficiently small. Then,

$$\mathbb{P}_\mu^\lambda(||m_{\eta_0}^\tau|| < e^{cn^d}) < e^{-c_1 n^d} + e^{-c_2 n^d} < e^{-cn^d}.$$

for n large enough. $\square$

Note the that the result is in terms of the number of topplings. However, transforming this result to replace $||m_{\eta_0}^\tau||$ with the fixation time $\mathcal{T}_n$ isn't hard now and the reader can find it on Page 19 in [2].

5 INTRODUCTION TO THE PROOF OF THE MAIN CLAIM

5.1 The main claims.

In the more 'complete' definition of the ARW that we had introduced in 3.5, we had mentioned that the most general condition on the initial configuration is that it should follow a translation-ergodic distribution on $\mathbb{N}_{\mathfrak{s}}^V$ with finite density of particles. Keeping that in our minds, we reintroduce our main claims again that we need to prove:

Theorem 12:

In every dimension $d \geq 1$ and for every sleeping rate λ, the critical density $\mu_c(\lambda)$ for the ARW on $\mathbb{Z}^d$ is always less than 1.

More precisely, in dimension 2, in the lower sleep rate regime i.e. when $\lambda, \mu \to 0$, $\exists\ a > 0$ such that $\forall \lambda$ in this regime,

$$\mu_c(\lambda) \leq \lambda |\ln \lambda|^a$$

and in the higher sleep rate regime i.e. when $\lambda \to \infty$,

$$\mu_c(\lambda) \leq 1 - \frac{c}{\lambda(\lambda)^2}.$$

These bounds on $\mu_c(\lambda)$ have been recently given by Asselah, Forien and Gaudilliere in October, 2022. Note one thing that in the 'Developments' section, we had already mentioned that Taggi had shown that for every sleeping rate $\lambda \in (0, \infty)$, $\mu_c(\lambda) < 1$ when $d \geq 3$ and Hoffman, Richey and Rolla have shown an analogous result for dimension $d = 1$. So, $d = 2$ was a long standing problem. In fact, we also had upper bounds for these regimes in other dimensions. However, Asselah, Forien and Gaudilliere have refined these upper bounds which we mention here:

Theorem 13:

In dimension $d \geq 3$ and in the small sleep rate regime, $\exists\ c > 0$ such that

$$\mu_c(\lambda) \leq c\lambda$$

and in the large sleep rate regime, again $\exists\ c > 0$ such that

$$\mu_c(\lambda) \leq 1 - \frac{c}{\lambda \ln \lambda}.$$

Remark 5.1 *If we consider any $\mu < 1$ in the ARW model, then we can choose λ small enough so that condition (1) in Theorem (3) is satisfied. Hence, by Remark 3.1, we are in the activated phase always for such a pair. Hence, we have already shown that Theorem 12 holds in the low sleep rate regime through our previous work.*

5.2 The six stages of the Proof.

We now briefly discuss the six stages in which we are going to tackle the proof of Theorem 12. Theorem 13 can then be proved through some small changes. The reader can refer to [1] for the proof of Theorem 13.

Stage 1: **Transforming the problem to the problem of infinite number of toruses i.e. $((\mathbb{Z}_n^d)_{n\in\mathbb{N}})$.**

Our target is to show that $\mu_c(\lambda) < 1$. Theorems (4) and (5) allowed us to change this problem into problems on the toruses $\mathbb{Z}_n^d$. However, the transformed problems on the toruses (Remark 3.1) is still hard to solve. Luckily, we have an analogous result mentioned in Theorem (14) which covers both high and low sleep rate regimes. We mention it as Theorem (14).

Stage 2. **Decomposition over all possible sleep site sets.**

Again, we fix $\lceil \mu n^d \rceil$ number of particles as that will act as an upper bound for the probability $\mathbb{P}_\mu^\lambda(\mathcal{T}_n < e^{cn^d})$ and we sum over all possible set of sleeping sites A where the particles can settle.

This will result in a factor of ${}^{n^d}\mathcal{C}_{\mu n^d}$ infront while summing the probability as we had in Theorem (10). There we had the advantage that we could make λ small enough to outweigh this term. Here, we take advantge of the fact that ${}^{n^d}\mathcal{C}_{\mu n^d}$ is small when either $\mu \to 0$ ($\lambda \to 0$) or $\mu \to 1$ ($\lambda \to \infty$). Now, due to the Monotonicity theorem, we get that ***we don't need to worry about middle values of μ (and hence, middle values of λ) if we can show that Theorem (12) holds for the low sleep-rate regime and high sleep-rate regime.***

Stage 3. **Reduction to a simpler model.**

Once we have fixed such an A, we now simplify the model and try to get a good upper bound on the probability. We do the simplification in the following manner:

1. We suppress the sleeping instructions outside $\mathbb{Z}_n^d \backslash A$. By Theorem (6), this gives us an upper bound on the probability.

2. We start with the initial configuration 1_A. By Theorem (7), this again gives us an upper bound on the probability.

So, we start with one active particle on each site of A. Suppose, we start with a site of A. Now, we implement the instruction. If it is a sleep instruction, the particle goes to sleep and we deal with the next one. Suppose it is a jump instruction. Then, it will form a loop of vertices it visits. It may visit several vertices of A and wake them up if they were sleeping. We will topple the sites until it comes back to its sites. This process will go on until all the particles go to sleep on the sites of A.

We now introduce a Theorem analogous to the combination of condition (1) in Theorem (3), Theorem (4) and Theorem (5) that will work both in the large and small sleep rate regimes.

Theorem 14:

Let $d \geq 1$, $\lambda > 0$ and $\mu \in (0,1)$. Define

$$\psi(\mu) = -\mu \ln \mu - (1-\mu)\ln(1-\mu).$$

Suppose there exists $a > 0$ and $b > \psi(\mu)$ such that for large enough $n \in \mathbb{N}$ and for every $A \subset \mathbb{Z}_n^d$ with $|A| = \lceil \mu n^d \rceil$,

$$\mathcal{P}_\mu^{\lambda,A}(M_A \leq e^{an^d} \mid \eta_0 = 1_A) \leq e^{-bn^d},$$

then $\mu \geq \mu_c(\lambda)$.

Here, M_A denotes the number of topplings on A done to stabilize A.

Remark 5.2 *We are in a dilemma. Do you remember the random walk introduced in section 4.2? There, the active particle at x_j was allowed to activate only the sleeping particle at x_{j-1}. Since, we were in the low sleep-rate regime then, so it was easy to show that the number of topplings required is exponentially large in that case. In case of the higher sleep-rate regime, we want the particle at x_j to wake a few more particles so as to get exponential number of topplings since the particles in this case will stabilise more quickly. However, we don't want to wake up every particle that the particle at x_j visits as that will be tough to handle. So, this dilemma is resolved by introducing the concepts of Hierarchial dormitories, distinguished vertices and colored loops.*

Stage 4. **Hierarchial Dormitories**

After fixing the settling set A on the torus $\mathbb{Z}_d^n$, we now create a deterministic hierarchial structure: a decreasing sequence of subsets of A i.e. $(A = A_0 \supset A_1 \supset A_2 \ldots \supset A_{\mathcal{J}})$ and a sequence of decreasing partitions i.e. $\mathcal{C}_0 \supset \mathcal{C}_1 \ldots \supset \mathcal{C}_{\mathcal{J}}$ where $\mathcal{C}_k$ is a partition of A_k. Each of the constituent sets of a partition is called a cluster. The basic criteria of forming a cluster is that the vertices of the cluster should be near to each other i.e. it should be a dense structure so that its stabilization takes a large time. So, when λ is very small ($\mu \to 0$), the $n-$toruses will be sparsely filled with particles. Hence, in that case, the clusters of $\mathcal{C}_0$ are the singleton sets.

After constructing the partition $\mathcal{C}_j$, we form the partition $\mathcal{C}_{j+1}$ by merging the clusters in $\mathcal{C}_j$ which are very close to each other. At each level, it is required that the total size of each constituent cluster(in terms of the number of vertices) dominates an increasing sequence of real numbers. We also throw out clusters based on these conditions (especially, which are isoltaed) and at the the end, we obtain a single big cluster $\mathcal{C}_j$.

Stage 5. **Introducing Distinguished vertices and colored loops.**

5.1 Distinguished vertices. In every cluster C in every partition $\mathcal{C}_j$, we choose one vertex x and call it our distinguished vertex. What distinguishes it from the other vertices of the same cluster is that only its particle has the capacity to wake up other particles in the clusters $\mathcal{C}_1, \ldots, \mathcal{C}_{\mathcal{J}}$ during a loop.

5.2 A construction for Distinguished vertices.

We first create any arbitrary order of the vertices on the torus.

- For every $C \in \mathcal{C}_0$, we set x_C^* to be the smallest vertex in cluster C.
- For every $C \in \mathcal{C}_{j+1} \backslash C_j$, we have $C = D \cup E$, where $D, E \in \mathcal{C}_j$. So, we choose x_C^* to be the distinguished vertex of the larger cluster of D and E.

We find that still the number of loops emitted by the distinguished vertices is complex to handle. So, we now introduce the concept of colored loops getting a further lower bound on M_A.

5.3 Colored loops. Firstly, we take a collection of infinite of colors and label the colors as color 0, 1, and so on. Now, whenever a loop starts from the vertex x, we randomly assign this loop a color according to the distribution $\mathbb{P}(J+1 = k) = \frac{1}{2^k}\ \forall\ 1 \le k < \infty$. The aim behind introducing colored loops is to ensure that loops of different colors coming from any vertex x affect different, disjoint zones in the dormitory hierarchy. This reduces the complexity of the problem. This is made sure by defining

$$w(x,j) = \begin{cases} C_{j+1}(x)\backslash C_j(x), & \text{if } x \text{ is distinguished at level j} \\ \phi, & \text{if } x \text{ is distinguished at level 0 but not at level 0} \\ C_0(x), & \text{if } x \text{ is not distinguished at any level} \end{cases}$$

Here, $w(x,j)$ denotes the set of vertices that can waken up by a loop of color j from vertex x.

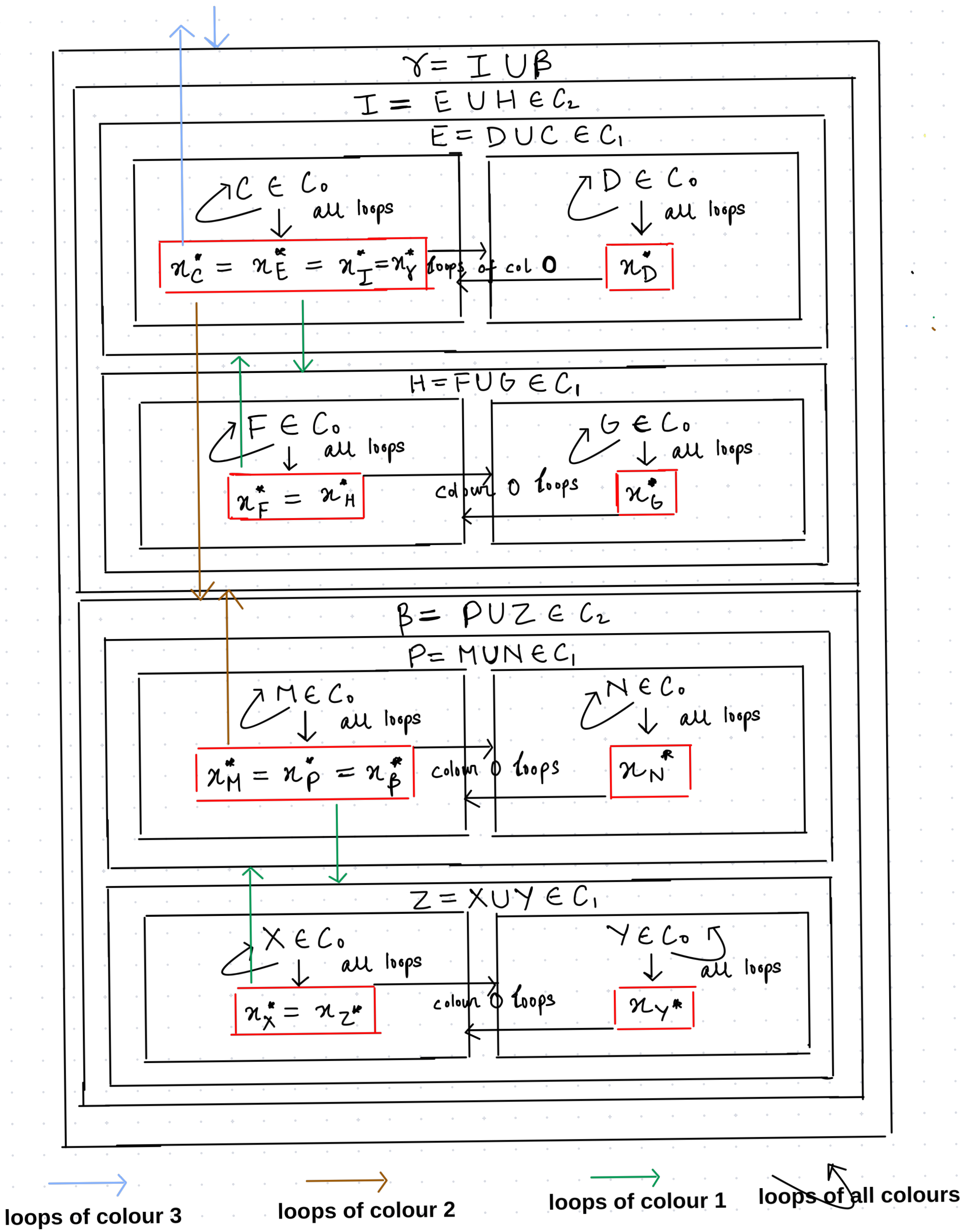
Stage 6. **Ping-pong rallies**[1]

In the Inductive step in [1], dormitory hierarchies in both low sleep rate regime and large sleep rate regime are constructed in way such that the number of topplings to stabilize each cluster of $\mathcal{C}_0$ is $\mathcal{O}(\exp|C|)$. In this case, one can then use induction and the idea of *Ping-pong rally* to prove the following claim:

The topplings needed to stabilize each cluster $C \in \mathcal{C}_j$ *is* $\mathcal{O}(\exp|C|)$ $\forall j \leq \mathcal{J}$.

As we will see from the definition of the Hierarchial dormitory after a bit that each cluster of $\mathcal{C}_{j+1}$ is either a cluster of $\mathcal{C}_j$ or a union of 2 clusters from $\mathcal{C}_j$.

For any $C \in \mathcal{C}_{j+1}$, $C = D \cup E$ where $D, E \in \mathcal{C}_j$. We stabilize this cluster as follows: we first completely stabilize D and then move on to stabilize E. While stabilizing E, with a positive probability, all the sites of D becomes active again. Then, we again stabilize D after stabilizing E. This ***Ping-Pong rally*** goes on until both of them stabilize. This stabilisation time is still stochastically bounded above by the actual stabilization time when the recursive toppling strategy is used with the Ping-Pong rally and it is found to be exponentially large in the size of the cluster C if this property held true for clusters in $\mathcal{C}_0$. We won't directly use Ping-pong rallies but we will use a result which is proved using Ping-Pong rally, colored loops and distinguished vertices - the 'Inductive Lemma'. For more details, the reader can refer to [1] where the proof of the Inductive lemma is given in Chapter 7.


$\gamma = I \cup \beta$
$I = E \cup H \in C_2$
$E = D \cup C \in C_1$
$C \in C_0$
all loops
$x^*_C = x^*_E = x^*_I = x^*_\gamma$
loops of col 0
$D \in C_0$
all loops
x^*_D
$H = F \cup G \in C_1$
$F \in C_0$
all loops
$x^*_F = x^*_H$
colour 0 loops
$G \in C_0$
all loops
x^*_G
$\beta = P \cup Z \in C_2$
$P = M \cup N \in C_1$
$M \in C_0$
all loops
$x^*_M = x^*_P = x^*_\beta$
colour 0 loops
$N \in C_0$
all loops
x^*_N
$Z = X \cup Y \in C_1$
$X \in C_0$
all loops
$x^*_X = x^*_Z$
colour 0 loops
$Y \in C_0$
all loops
x^*_Y
loops of colour 3
loops of colour 2
loops of colour 1
loops of all colours

6 GENERAL CONSIDERATIONS

We need several tools mentioned in Section 2 in [1] before embarking on the proof of Theorem (12). We mention them here.

6.1 Introducing a metric on the torus $\mathbb{Z}_n^d$.

In section 4.3, we had briefly introduced a variant of the d_1 metric on the torus. Here, we introduce a variant of the d_∞ metric.

First we define the map $\pi_n : \mathbb{Z}^d \to \mathbb{Z}_n^d$ that projects the points of $\mathbb{Z}^d$ onto the corresponding torus. Then, for any $x, y \in \mathbb{Z}_n^d$, the metric d is defined as:

$$d(x, y) = \inf \{d_\infty(a, b) : a, b \in \mathbb{Z}^d, \pi_n(a) = x, \pi_n(b) = y\}.$$

Given any $C \subset \mathbb{Z}_n^d$, we define its diameter as diam C $= \max_{x,y\ \in C} d(x, y)$. Also, the natural way to define the closed ball around x of radius r is $B(x, r) = \{y \in \mathbb{Z}_n^d : d(x, y) \leq r\}$. After defining the closed ball in this manner, we get that the volume of $B(x, r)$ is given by

$$|B(x, r)| = \begin{Bmatrix} (2r+1)^d & \text{if } n \geq 2r+1 \\ n^d & \text{otherwise} \end{Bmatrix}.$$

Also, we say that a set $C \in \mathbb{Z}_n^d$ is $r-$connected if for any $x, y \in C\ \exists$ a sequence $x = x_0, \ldots, x_m = y$ such that $d(x_i, x_{i+1}) \leq r$.

Now, we formally define the Dormitory Hierarchy.

6.2 Formal definition of Dormitory Hierarchy.

For every $A \subset \mathbb{Z}_n^d$, $D = (D_j)$ - a $\mathbb{N}-$ valued sequence and $v > 0$, we define the $(v, D)-$dormitory hierarchy on A as follows:

The $(v, D)-$dormitory hierarchy on A is a decreasing sequence of subsets $A \supset A_0 \supset \ldots \supset A_{\mathcal{J}}$ with $\mathcal{J} \in \mathbb{N}$ along with a sequence of partitions $\mathcal{C}, \mathcal{C}_0, \ldots, \mathcal{C}_{\mathcal{J}}$ where $\mathcal{C}_k$ is a partition of A_k such that

1. Every $C \in \mathcal{C}_j$ must satisfy $|C| \geq 2^{\lfloor j/2 \rfloor v}$ where $0 \leq j \leq \mathcal{J}$.
2. For $0 \leq \mathcal{J} - 1$ and for every $C \in \mathcal{C}_{j+1} \backslash \mathcal{C}_j$, diam $C \leq D_j$ and C must be the union of two clusters from $\mathcal{C}_j$ i.e. $C = D \cup E$ where D, E are in $\mathcal{C}_j$.
3. $\mathcal{C}_{\mathcal{J}}$ is Singleton.

In Condition (1), v is controlling the volume of clusters at each level. We throw out the clusters after each level which do not satisfy this condition. According to condition (2), either we choose to bring the cluster C from $\mathcal{C}_j$ if it satisfies condition (1) or we merge it with another cluster in C_j such that they are close to each other (controlled by D_j). At the end, we obtain a single large cluster.

Analogous to the random field of instructions τ in the Diaconis-Fulton construction, we have a random field of instructions this time as well but it contains more information now: it contains the original field of instructions $\tau = (\tau(x,h))_{x\in\mathbb{Z}_n^d, h\in\mathbb{N}}$ distributed according to $\mathbb{P}^{\lambda,A}$ and the field of colored loops at each vertex given by $(1+J(x,l))_{x\in\mathbb{Z}_n^d, l\in\mathbb{N}}$ distributed according to Geom(1/2).

6.3 Representation of the modified ARW model.

Now, after sampling such a random field of instructions, when we carry out the toppling process, we can reveal the complete information till now by keeping track of these 3 vectors.

After fixing the dormitory A, we consider the following sequences:

1) $(I(x,h))_{x\in A,h\in\mathbb{N}}$: Here, h represents the number of instructions (sleep or loop) that have already been implemented on x. Without knowing about the field of instructions, $I(x,h)$ is a Bernoulli random variable which takes value 1 with probability $\frac{\lambda}{1+\lambda}$ i.e. if the $(h+1)-$th instruction is a sleep instruction.

2) $(J(x,l))_{x\in A,l\in\mathbb{N}}$: Here, l represents the number of loops that have already been implemented on x. $J(x,l)$ is a random variable representing the color of the $l+1$-th loop on x and $J(x,l)+1$ follows a Geom(1/2) distribution.

3) $(\Gamma(x,j,l))_{x\in A,l\in\mathbb{N},j\in\mathbb{N}}$: $\Gamma(x,l,j)$ denotes set of vertices visited using the (l+1)-th loop of $j-$th color during the symmetric random walk on the torus starting from vertex x and killed at x.

If we don't know about the random field of instructions beforehand, then we can visualize each of the 3 vectors as iid random variables and $\Gamma(x,j,l)\cap w(x,j)$ represents the sites whose particles are awakened by the $(l+1)$-th loop of color j emitting from vertex x.

6.4 Step-toppling operator.

We denote the set of actives vertices in A by R. Our loop odometer function is $l : A \to \mathbb{N}_0$ and $h : A \to \mathbb{N}_0$ is the usual odometer function.

We want to introduce something similar to the operator Φ_α^τ (introduced in Page 4). There α was an ordered sequence of vertices such as $\alpha = (x_1, x_2, \ldots, x_l)$, $\Phi_\alpha^\tau = \Phi_{x_j}^\tau \Phi_{x_{j-1}}^\tau \ldots \Phi_{x_1}^\tau$, τ was a field of instructions and α was a $\tau-$legal sequence. We had also mentioned that $\Phi_x^\tau(\eta,h) = (\tau^{x,h(x)+1}\eta, h+\delta_x)$.

Analogous to the toppling operator Φ_x^τ is the **Step-toppling operator** $\Phi_x(R,h,l)$:

$$\Phi_x(R,h,l) = \begin{Bmatrix} (R\backslash\{x\}, h+\delta_x, l) & \text{if } i=1 \\ (R\cup((\Gamma(x,l(x),j)\cap w(x,j)), h+\delta_x, l+\delta_x) & \text{otherwise} \end{Bmatrix}.$$

Here, $i = I(x,h(x))$.

Note that the odometer function h here is counting the total number of loops (not total number of jump instructions on x) and sleep instructions on x.

6.5 Creating a toppling strategy for every cluster C that gets along with the Ping-pong rally.

Now, we want to create a toppling strategy for each cluster C in every C_j so that when Ping-pong rally is done to stabilise the cluster then still the stabilisation time is stochastically dominated by the actual stabilization time of the cluster C.

Now, for any cluster $C \subset \mathcal{C}_0$, we want to choose a vertex in C and try to stabilize it. We definitely prefer to topple the distinguished vertex first every time until it stabilizes since it emits many loops and so this will help us to achieve a very large stabilization time still stochastically dominated by the stabilisation time of the original ARW. Inspired by this, we define the $C-$toppling procedure $f : \mathcal{P}(C)\backslash\{\phi\} \to C$ such that $f(R) \in R$ and $x^*_C \in R \implies f(R) = x^*_C$. R is the set of all active vertices of C. So, the $C-$toppling procedure helps us to get the vertex that will be toppled now. Now, we define the procedure-toppling operator which is nothing but the application of the Step-toppling operator Φ_x on the site given by f.

So, $\Phi_C : \mathcal{P}(A) \times (\mathbb{N}^A)^2 \to \mathcal{P}(A) \times (\mathbb{N}^A)^2$ is such that $\Phi_C(R, h, l) = \Phi_{f(R\cap C)}(R, h, l)$ if $R \cap C \neq \Phi$ and $\Phi_C(R, h, l) = (R, h, l)$ if $R \cap C = \phi$.

We now describe how we use the procedure-toppling operator to stabilise each cluster C.

6.5.1 Recursive toppling strategy.

1. **Stabilization operator and stabilization at the 0th level.** Since, we have introduced the Toppling procedure so we can now introduce one more operator i.e. the Stabilization operator Stab_C. $\mathrm{Stab}_C : \mathcal{P}(A) \times (\mathbb{N}^A)^2 \to \mathcal{P}(A) \times (\mathbb{N}^A)^2$ such that if R_0 is the set of active vertices of A, h_0 and l_0 the odometers until now, then $\mathrm{Stab}_C(R_0, h_0, l_0) \to (R_\tau, h_\tau, l_\tau)$ where Stab_C returns the updated odometer functions and the configuration after the cluster C is stabilised using the $C-$toppling procedure f_C.

We stabilize every cluster $C \in \mathcal{C}_0$ by repeatedly apply the $C-$toppling procedure f_C i.e. by using the stabilization operator Stab_C once.

2. **Ping-Pong rally.** Let $j \in \{1, 2, \ldots, \mathcal{J}\}$ be such that the stab operator is defined for every $C \in \mathcal{C}_{j-1}$. Let $C \in \mathcal{C}_j\backslash\mathcal{C}_{j-1}$. From the definition of dormitory hierarchy, $C = D \cup E$ where D, E are in $\mathcal{C}_{j-1}$. Assume that $x^*_C = x^*_D$. If not, then we interchange the labels of D and E.

To stabilise the cluster C, we first stabilise D and then E and then again D if stabilising E awakens some sites in D. This goes on until both D and E have stabilized. In this case,

$$\mathrm{Stab}_D(R_{2i}, h_{2i}, l_{2i}) = (R_{2i+1}, h_{2i+1}, l_{2i+1})$$
$$\mathrm{Stab}_E(R_{2i+1}, h_{2i+1}, l_{2i+1}) = (R_{2i+2}, h_{2i+2}, l_{2i+2})$$

where $0 \le i \le \tau$ where τ is the number of rallies it takes to stabilize C.

Note that we stabilise D by repeatedly using its procedure toppling operator Φ_D and we stabilise E by repeatedly using the procedure toppling operator Φ_E. So, we just need toppling procedures for the clusters in $\mathcal{C}_0$.

Lastly, we introduce one more concept before moving to a few results and theorems needed for our proof.

6.6 Total number of topplings and loops during stabilisation.

Let $C \in \mathcal{C}_j$ for some $j \in \{1, 2, \ldots, \mathcal{J}\}$. Then, we represent the total number of sleep and loop instructions executed on the vertices of A during the stabilisation of C by $\mathcal{H}(C)$ i.e. $\mathcal{H}(C) = \sum_{x\in A} h_{\mathrm{stab}}(x)$ where h_{stab} is the odometer function obtained after applying Stab_C. We define the number of loops emitted by the distinguished vertex x^*_C during the stabilisation of C by $\mathcal{L}(C) =$

$l_{\text{stab}}(x_C^*)$. Hence, $\mathcal{S}(C) = h_{\text{stab}}(x_C^*) - l_{\text{stab}}(x_C^*)$ where $S(C)$ denotes the number of sleep instructions executed on x_C^* while stabilizing C.

6.7 Sufficient condition for activity using the function $\mathcal{H}$.

We now introduce a key theorem that will help us to establish that $\mu \geq \mu_c$. We had introduced a more general result in Theorem (14). However, the condition given by it is still hard to check. This result utilises the concepts of Dormitory hierarchy, colored loops, distinguished vertices and toppling procedures and its condition is easier to check.

Theorem 15

Let $d \geq 1, \lambda > 0$ and $\mu \in (0, 1)$. Suppose there exists $\kappa > \psi(\mu)$ such that for large enough $n \in \mathbb{N}$ and for every $A \subset \mathbb{Z}_n^d$ with $|A| = \lceil \mu n^d \rceil$, there exists a dormitory hierarchy $(A_j, \mathcal{C}_j)_{j \leq \mathcal{J}}$ and a toppling procedure f_C for every cluster $C \in \mathcal{C}_0$ so that through Recursive toppling strategy, we get the stochastic domination,

$$\mathcal{H}(A_{\mathcal{J}}) \succeq Geom(exp(-\kappa n^d)). \tag{10}$$

Then, we have $\mu \geq \mu_c(\lambda)$.

We now introduce a few results that will be crucial in proving Theorem 12.

6.8 Hitting probabilities on the torus.

Consider the closed ball $B[x, r]$ in the torus $\mathbb{Z}_n^d$ with the metric we introduced in section 6.1. Let $y \in B[x, r]$ be any site. A natural question is: What is the smallest probability with which a particle from vertex x hits vertex y before coming to x? We will see that the the answer to this question will be crucial to prove our theorem.

For this, we introduce the function $\Upsilon_d : r \in \mathbb{N}\backslash\{0\} \to \inf\{\mathbb{P}_x(T_y < T_x^+), n \in \mathbb{N}, x, y \in \mathbb{Z}_n^d : d(x, y) \leq r\}$.

Then, we have the following lemma:

1. When $d = 1$, we have $\Upsilon_1(r) = 1/(2r)$ for every $r \geq 1$.
2. When $d = 2$, there exists $K > 0$ such that $\Gamma_2(r) \geq K \ln(r)$ for every $r \geq 2$.
3. When $d \geq 3$, there exists $K = K(d) > 0$ such that $\Upsilon_d(r) \geq K$ for every $r \geq 1$.

6.9 Correlation between the colored loops.

For any dormitory hierarchy $(A_j, \mathcal{C}_j)_{j \leq \mathcal{J}}$ on some $A \subset \mathbb{Z}_n^d$, we have a very useful result that links the number of loops of color j performed by the distinguished vertex of any cluster $C \in \mathcal{C}_j$ during the stabilisation of C to the number of loops of color at most $j - 1$ from this vertex and the number of sleep instructions implemented on this vertex during the stabilisation of C.

Result 1.[1] *Let $d, n \geq 1$ and $A \subset \mathbb{Z}_n^d$ be any subset lying in this discrete torus. Fix any $\lambda > 0$ and $(A_j, \mathcal{C}_j)_{j \leq \mathcal{J}}$ dormitory hierarchy on A. For every $C \in \mathcal{C}_0$, let f_C be a toppling procedure on C. Then, for every $j \in \{0, \ldots, \mathcal{J}\}$ and $C \in \mathcal{C}_j$, we have*

$$\mathcal{L}(C, j) \text{ and } \sum_{i=1 \text{ to } \mathcal{T}} (X_i - 1) \text{ have the same distribution.}$$

Here, $\mathcal{T} = \mathcal{S}(C) + \mathcal{L}(C,0) + \ldots + \mathcal{L}(C,j-1)$ and $(X_i)_{i\geq 1}$ are i.i.d. geometric random variables with parameter

$$\frac{\lambda + 1 - 2^{-j}}{\lambda + 1 - 2^{-(j+1)}}$$

which are independent of $\mathcal{T}$.

6.10 Induction hypothesis and Induction Lemma.

So, for any $\lambda > 0$ and a nicely chosen $\mu \in (0,1)$, we want to show that for n large enough and for every $A \subset \mathbb{Z}_n^2$ with $|A| = \lceil \mu n^2 \rceil$, $\exists$ a hierarchial dormitory and a set of toppling procedures such that through the recursive toppling strategy, condition (10) holds true. Also, we need to find $\kappa > \psi(\mu)$ that doesn't not depend on A and n.

We approach this problem using our favourite tool i.e. Induction. Notice that $\mathcal{H}(A_{\mathcal{J}}) \geq |A| + \mathcal{L}(A_{\mathcal{J}}) \geq 1 + \mathcal{L}(A_{\mathcal{J}}, \mathcal{J})$ since we are executing at least $|A|$ number of sleep instructions on A. Hence, it suffices to show that $1 + \mathcal{L}(A_{\mathcal{J}}, \mathcal{J})$ stochastically dominates the Geometric random variable with parameter $\exp(-\kappa n^2)$.

Consider the following property and induction hypothesis:

If for some dormitory hierarchy $(A_j, \mathcal{C}_j)_{j \leq \mathcal{J}}$ on A, $\forall C \in \mathcal{C}_j$, $1 + \mathcal{L}(C,j) \succeq Geom(exp(-\alpha_j |C|))$, where (α_j) is some given sequence of positive reals, then we say that the property $\mathcal{P}$ holds for that j.

Our ***Induction hypothesis*** is that $\mathcal{P}(j)$ holds for all $1 \leq j \leq \mathcal{J}$ for this dormitory hierarchy $(A_j, \mathcal{C}_j)_{j \leq \mathcal{J}}$. So, if we choose the sequence (α_j) properly then we are done if the Induction hypothesis holds true in that case for every such A.

This is still very hard to verify. However, while describing the Ping-Pong rally, we mentioned that its role is to prove an 'Inductive lemma' described in Chapter 7 in [1]. This lemma will make it effortless to prove the Induction hypothesis in our case. We state the lemma now:

Lemma 9. [1] *Let $d \geq 1, \lambda > 0, v \geq 1, (D_j)_{j\in\mathbb{N}} \in (\mathbb{N}\backslash\{0\})^{\mathbb{N}}$ and let $(\alpha_j)_{j\in\mathbb{N}}$ be a sequence of positive real numbers such that*

$$\forall j \in \mathbb{N} \qquad \frac{4v(1+\lambda)2^{3j/2}}{(1-e^{-\alpha_j}v)\Upsilon_d(D_j)} \leq exp((\alpha_j - \alpha_{j+1})2^{j/2}v). \tag{11}$$

For every $n \geq 1$ and every $A \subset \mathbb{Z}_n^d$ equipped with a $(v,D)-$dormitory hierarchy $(A_j, \mathcal{C}_j)_{j\leq\mathcal{J}}$ and with a collection of toppling procedures $(f_C)_{C\in\mathcal{C}_0}$, if the property $\mathcal{P}(0)$ holds, then $\mathcal{P}(j)$ also holds for every $j \leq \mathcal{J}$.

7 PROOF OF THEOREM 12.

7.1 Proving Theorem 12 in the low sleep rate regime i.e. when $\lambda \to 0$.

Now, through section 6.10, it suffices to show that for each such A, $\exists$ a $(v, D)-$dormitory hierarchy $(A_j, \mathcal{C}_j)_{j\leq\mathcal{J}}$, κ, a set of toppling procedures - one for each cluster $C \in \mathcal{C}_0$ and a sequence (α_j) of positive reals such that

1. κ doesn't depend on A and n
2. Condition (11) holds true
3. $\mathcal{P}(0)$ holds true
4. $\alpha_{\mathcal{J}} \leq \frac{\kappa n^2}{|A_{\mathcal{J}}|}$
5. $\kappa > \psi(\mu)$

We call this set of conditions - **condition set I**.

For the existence of such a dormitory hierarchy, we rely on the following lemma:

Lemma 10.[1] *Let $d = 2$, $D_0 \geq 1$ and $D_j = 6^j D_0$ for every $j \geq 1$. For every $n \geq 1$ and every $A \subset \mathbb{Z}_n^2$ with $|A| \geq 288n^2/(D_0)^2$, there exists $\mathcal{J} \in \mathbb{N}$ and a $(1, D)-$dormitory hierarchy $(A_j, \mathcal{C}_j)_{j\leq\mathcal{J}}$ on A with $|A_{\mathcal{J}}| \geq |A| - 144n^2/(D_0)^2$ and where $\mathcal{C}_0$ contains only singletons.*

The reader is requested to refer to section (5) in [1] for a proof of this lemma.

Proof. Fix $d = 2$. For every $\lambda \in (0, 1)$, we define for $j \in \mathbb{N}$,

$$D_0 = \lceil 1/\lambda \rceil \quad \text{and} \quad D_j = 6^j \lceil 1/\lambda \rceil.$$

Set $\alpha_j = \ln(\frac{1+2\lambda}{2\lambda}) - \frac{a}{2}(1 - 2^{-j/4}) \ln|\ln\lambda|$, $a = \frac{2^{9/4}}{2^{1/4}-1}$, $\mu = \lambda|\ln\lambda|^a$ and $\mu' = \mu - \frac{144}{D_0^2}$.

Now, when the sleep rate $\lambda \to 0$, we have $\frac{1}{D_0^2} \sim \lambda^2 = o(\lambda|\ln\lambda|^a) = o(\mu)$. So, $\frac{1/D_0^2}{\mu} \to 0$ as $\lambda \to 0$. Hence, for λ sufficiently small, $\mu \geq \frac{288}{D_0^2}$. So, $|A| = \lceil \mu n^2 \rceil \geq \frac{288n^2}{D_0^2}$.

So, using Lemma 10, we get that for each such an $A, \exists\ \mathcal{J} \in \mathbb{N}$ and a $(1, D)-$dormitory hierarchy $(A_j, \mathcal{C}_j)_{j\leq\mathcal{J}}$ on A with $|A_{\mathcal{J}}| \geq |A| - 144n^2/(D_0)^2 \geq \mu' n^2$. Now, we use this obtained dormitory hierarchy to prove that condition set I holds true. Also, the toppling procedures in this case are all trivial as the clusters in $\mathcal{C}_0$ are all singleton.

Claim I: (2) in condition set I holds for λ small enough.

So, we need to show that

$$g(\lambda) = \sup_{j\in\mathbb{N}} \frac{1}{(\alpha_j - \alpha_{j+1})2^{j/2}} \ln\left[\frac{4(1+\lambda)2^{3j/2}}{(1 - e^{-\alpha_j})\Upsilon_2(D_j)}\right] \leq 1 \tag{12}$$

For every $j \in \mathbb{N}$, we have

$$\begin{aligned}\alpha_j - \alpha_{j+1} &= \frac{a}{2}(2^{-j/4} - 2^{-(j+1)/4}) \ln|\ln\lambda| \\ &= \frac{a(2^{1/4}) - 1}{2^{5/4+j/4}} \\ &= \frac{2\ln|\ln\lambda|}{2^{j/4}}.\end{aligned}$$

Also, from section 6.8, we get that $\exists K > 0$ such that for every $j \in \mathbb{N}$,

$$\begin{aligned}-\ln \Upsilon_2(D_j) &\leq \ln(\frac{\ln D_j}{K}) \\ &= \ln(j\ln 6 + \ln\lceil 1/\lambda\rceil) - K \end{aligned} \tag{13}$$

Put $a = \ln\lceil \frac{1}{\lambda}\rceil$ and $b = j\ln 6$. Then, we have the following form from (13): $\ln(a+b) - K$ where a is very large.

Now, $\ln(a+b) = \ln(a(1+\frac{b}{a})) = \ln(a) + \ln(1+\frac{b}{a})$. Also, $\ln(1+x) \leq x^a$ when x is very small and a is very large. Using this we get that, $\ln(a+b) \leq \ln(a) + (b/a)^a$.

So, $\ln(j\ln 6 + \ln\lceil 1/\lambda\rceil) - K \leq \ln\ln\lceil 1/\lambda\rceil + \frac{j\ln 6}{\ln\lceil 1/\lambda\rceil} - \ln K$.

Plugging this expression in (12), we get that when $\lambda \to 0$

$$g(\lambda) \leq \frac{1}{2\ln|\ln\lambda|}[C + \ln|\ln\lambda|] = \frac{1}{2} + o(1)$$

where C is some constant.

So, $g(\lambda) \leq 1$ for small enough λ. Hence, condition (11) holds true in this case. □

Claim II: (3) in condition set I holds.

The stabilization of any $C \in \mathcal{C}_0$ requires exactly one sleep instruction i.e. $\mathcal{S}(C) = 1$. Hence, from result (1), we get that $\mathcal{L}(C,0)+1$ has the same distribution as X_1 where $X_1 \sim \text{Geom}(\frac{2\lambda}{1+2\lambda})$.

Now, $\frac{2\lambda}{1+2\lambda} = e^{-\alpha_0} = e^{-\alpha_0|C|}$. Hence, $\mathcal{P}(0)$ holds true.

Hence, the conditions of Lemma (9) are satisfied. So, the induction hypothesis is true and $\mathcal{H}(A_{\mathcal{J}}) \succeq 1 + \mathcal{L}(A_{\mathcal{J}}, \mathcal{J}) \succeq \text{Geom}(\exp(-\alpha_{\mathcal{J}}|A_{\mathcal{J}}|))$. □

Claim III: Conditions (1) and (5) in condition set I hold true.

We set $\kappa = \alpha_\infty \mu'$ where $\alpha_\infty = \inf_{j\in\mathbb{N}} \alpha_j = \ln(\frac{1+2\lambda}{2\lambda}) - \frac{a}{2}\ln|\ln\lambda|$.

Now, when $\lambda \to 0$, we have

$$\alpha_\infty\mu' = (|\ln\lambda| - \frac{a}{2}\ln|\ln\lambda| + O(1))(\lambda|\ln\lambda|^a + O(\lambda^2)) = \lambda|\ln\lambda|^{a+1} - \frac{a}{2}\lambda|\ln\lambda|^a\ln|\ln\lambda| + O(\lambda|\ln\lambda|^a)$$

and,

$$\psi(\mu) = \lambda|\ln\lambda|^a(|\ln\lambda| - a\ln|\ln\lambda|) + O(\lambda|\ln\lambda|^a) = \lambda|\ln\lambda|^{a+1} - a\lambda|\ln\lambda|^a\ln|\ln\lambda| + O(\lambda|\ln\lambda|^a).$$

Hence, $\alpha_\infty\mu' > \psi(\mu)$ for small enough λ. □

Claim IV: condition (4) holds true is trivial to check.

Hence, by Theorem (15), $\mu \geq \mu_c(\lambda)$ and $\mu = \lambda|\ln\lambda|^a$ is an upper bound on $\mu_c(\lambda)$ when $\lambda << 1$. ■

7.2 Proving Theorem 12 in the high sleep rate regime i.e. when $\lambda \to \infty$.

Proof: Again through section (6.9), it suffices to show that for each $A \subset \mathbb{Z}_n^2$ (large enough n), $\exists$ a $(v, D)-$dormitory hierarchy $(A_j, \mathcal{C}_{\mathcal{J}})_{j \leq \mathcal{J}}$, κ, a set of toppling procedures - one for each cluster $C \in \mathcal{C}_0$ and a sequence (α_j) of positive reals satisfying the conditions in the **condition set I**. We mention it again here:

1. κ doesn't depend on A and n
2. Condition (11) holds true
3. $\mathcal{P}(0)$ holds true
4. $\alpha_{\mathcal{J}} \leq \frac{\kappa n^2}{|A_{\mathcal{J}}|}$
5. $\kappa > \psi(\mu)$

We will rely on the following lemma for the existence of a suitable dormitory hierarchy to be used in Lemma 9:

Lemma 11.[1] *Let $d = 2, r \geq 1$ and $D_j = 6^j \times 96r^3$ for every $j \in \mathbb{N}$. Then, for n large enough, for every $A \subset \mathbb{Z}_n^2$ with $|A| \geq n^2/2, \exists\ \mathcal{J} \in \mathbb{N}$ and a $(r^2, D)-$dormitory hierarchy $(A_j, \mathcal{C}_j)_{j \leq \mathcal{J}}$ on A with $|A_{\mathcal{J}}| \geq |A| - n^2/2$ with every every set $C \in \mathcal{C}_0$ being $8r-$connected (definition in section 6.1)and satisfying $|C \cap B(x, 4r)| \geq r^2\ \forall x \in C$.*

The reader is requested to refer to section 5 in [1] for a proof of this lemma.

Now, define for every $j \in \mathbb{N}$,

$$D_j = 6^j \lceil 1/\lambda \rceil, \quad \alpha_j = \frac{1 + 2^{-j/4}}{2}\alpha_0 \quad \text{and} \quad v = r^2 = \left\lceil \frac{8 \ln \lambda \sqrt{\lambda}}{\sqrt{K}} \right\rceil^2$$

where $\alpha_0 = \frac{K}{\lambda \ln \lambda}$.

Claim 1: (2) in condition set I holds true.

So, we aim to show that $g(\lambda) \leq 1$ (introduced in the previous proof).

For every $j \in \mathbb{N}$, $\alpha_j - \alpha_{j+1} = \frac{2^{1/4}-1}{2^{5/4}} \frac{\alpha_0}{2^{j/4}}$. Also, $\ln(1/\Upsilon_2(D_j)) = -\ln(\Upsilon_2(D_j)) \leq -\ln(K/\ln(D_j)) = \ln(j \ln 6 + \ln 96 + 3 \ln r) - \ln K \leq \ln \ln r + \ln j + C$ for some constant $C > 0$.

So,

$$\begin{aligned} g(\lambda) &= \sup_{j \in \mathbb{N}} \frac{1}{(\alpha_j - \alpha_{j+1})2^{j/2}r^2} \ln \left[\frac{4r^2(1+\lambda)2^{3j/2}}{(1 - e^{-\alpha_j r^2})\Upsilon_2(D_j)} \right] \\ &\leq \frac{16}{\alpha_0 r^2} \sup_{j \in \mathbb{N}} \frac{2 \ln 2 + 2 \ln r + \ln(1+\lambda) + 3j \ln 2/2 - \ln(1 - e^{-\alpha_j r^2}) - \ln \Upsilon_2(D_j)}{2^{j/4}} \end{aligned}$$

Now, when $\lambda \to \infty$, $\frac{16}{\alpha_0 r^2} \approx \frac{1}{4 \ln \lambda}$. Also, in this case, $2 \ln 2 + 3j \ln 2/2 = \mathcal{O}(1)$, $\ln(1+\lambda) \approx \ln \lambda$, $\ln(1-$

$e^{-\alpha_j}r^2) = \mathcal{O}(e^{-\alpha_j}r^2)$ and $\ln \Upsilon_2(D_j) = \mathcal{O}(\ln\ln r)$. Hence, we get that

$$
\begin{aligned}
g(\lambda) &\leq \frac{1}{4\ln\lambda}\left[2\ln r + \ln\lambda + \mathcal{O}(e^{-\alpha_j}r^2) + \mathcal{O}(\ln\ln r) + \mathcal{O}(1)\right] \\
&= \frac{1}{4} + \frac{\ln r}{2\ln\lambda} + o(1) \\
&= \frac{1}{4} + \frac{1}{2}\frac{\mathcal{O}(\ln\ln\lambda + \sqrt{\lambda})}{\ln\lambda} + o(1) = \frac{1}{2} + o(1).
\end{aligned}
$$

So, for large enough λ, $g(\lambda) \leq 1$. $\square$

Claim 2: (3) in condition set I holds true.

We still need to specify the $(v, D)-$dormitory hierarchy $(A_j, \mathcal{C}_j)_{j\leq\mathcal{J}}$ and the collection of toppling procedures to use Lemma (9). For the collection of toppling procedures, we rely on the following lemma:

Lemma 12.[1] *Assume that $d = 2$. There exists $\lambda_0 > 1$ such that, for every $\lambda \geq \lambda_0$, defining*

$$
r = \left\lceil \frac{8\ln\lambda\sqrt{\lambda}}{\sqrt{K}} \right\rceil \text{ and } \alpha_0 = \frac{K}{\lambda\ln\lambda}
$$

for every $D \in (\mathbb{N}\backslash 0)^{\mathbb{N}}$ and $n \geq 1$, if $A \subset \mathbb{Z}_n^2$ and $(A_j, \mathcal{C}_j)_{j\leq\mathcal{J}}$ is a $(r^2, D)-$dormitory hierarchy on A such that every set $C \in \mathcal{C}_0$ is $8r-$connected and dense in the sense of Lemma 11, then for every $C \in \mathcal{C}_0$, there $\exists$ a $C-$toppling procedure f such that $\mathcal{L}(C, 0)$ dominates a geometric variable with parameter $exp(-\alpha_0|C|)$.

(The reader can refer to Chapter 6 in [1] for the proof of this lemma.)

Let λ_0 be from Lemma 12. Then, we now choose $\lambda_1 \geq \lambda_0$ such that condition (2) in condition set I holds for all $\lambda \geq \lambda_1$ and $4\lambda_1(\ln\lambda_1)^2 \geq K$. From now, on we only consider $\lambda \geq \lambda_1$.

Now, we define $\mu = 1 - \frac{K}{8\lambda(\ln\lambda)^2}$. Then, $|A| = \lceil \mu n^2 \rceil \geq n^2/2$.

This gives us the $(v, D)-$dormitory hierarchy using Lemma 11.

Also, using Lemma 12, we get the collection of toppling procedures and that the property $\mathcal{P}(0)$ holds true. $\square$

Claim 3: (1) and (4) in condition set I hold true.

Hence, using Claim 1 and Claim 2, Lemma 9 is applicable here. Hence, $\mathcal{P}(j)$ holds true $\forall j \leq \mathcal{J}$.

So, $1 + \mathcal{L}(A_\mathcal{J}, \mathcal{J}) \succeq \text{Geom}(exp(-\alpha_\mathcal{J}|A_\mathcal{J}|))$.

Now by Lemma (11), $|A_\mathcal{J}| \geq |A| - n^2/2 \geq (\mu - 1/2)n^2$ and hence, we get that $\alpha_\mathcal{J}|A_\mathcal{J}| \geq \alpha_0(\mu - 1/2)n^2$.

Choosing $\kappa = \alpha_0(\mu - 1/2)$, we get that Claim 3 holds true. $\square$

Claim 4: (5) in condition set I holds true.

Now,

$$\begin{aligned}\kappa - \psi(\mu) &= \frac{\alpha_0}{2}\left(\mu - \frac{1}{2}\right) - \psi(\mu) \;=\; \frac{\alpha_0}{2} - \frac{\alpha_0 K}{16\lambda(\ln\lambda)^2} - \frac{\alpha_0}{4} - \mu|\ln\mu| - (1-\mu)|\ln(1-\mu)\\ &= \frac{K}{4\lambda\ln\lambda} + O\left(\frac{1}{\lambda^2(\ln\lambda)^3}\right) - \frac{K}{8\lambda\ln\lambda} + O\left(\frac{\ln\ln\lambda}{\lambda(\ln\lambda)^2}\right) \;=\; \frac{K}{8\lambda\ln\lambda} + o\left(\frac{1}{\lambda\ln\lambda}\right).\end{aligned}$$

So, $\kappa - \psi(\mu) > 0$ when λ is very large. ■

8 Conclusion

The ARW model, being a lattice model, exhibits a discernible connection to the Percolation model. Suppose we color code the state of each vertex in the ARW model. We color a vertex black when either the vertex doesn't have any particle or has a sleeping particle and we color it white otherwise. Then, the ARW model can be seen as a model where percolation is happening in continuous time - not just once! The vertices are continuously changing their states. The question of the existence of a density parameter $\mu < 1$ for any given λ can be translated into the following question: Does there exist $\mu < 1$ such that in the corresponding continuous time Percolation model, an infinite black cluster covering all the sites of $\mathbb{Z}^d$ appears eventually?

It is noteworthy that tackling the Percolation problem, where percolation occurs just once, can be challenging in itself. Hence, the complexity of the ARW problem can be estimated from this. I aim to uncover the relation between Percolation model and the ARW model in my next research work and the role λ plays in the continuous-time Percolation model.

References

[1] Asselah, Amine; Forien, Nicolas; Gaudillière, Alexandre (2022). *The Critical Density for Activated Random Walks is always less than 1.* arXiv:2210.04779.

[2] Forien, Nicolas; Gaudillière, Alexandre (2022). *Active Phase for Activated Random Walks on the Lattice in all Dimensions*. arXiv: 2203.02476

[3] Levine, Lionel; Silvestri, Vittoria (2023). *Universality Conjectures for Activated Random Walk*. arXiv: 2306.01698

[4] Rolla, Leonardo T. (2019). *Activated Random Walks on* $\mathbb{Z}^d$. Probability Surveys. DOI 10.1214/19-ps339