THE CRITICAL SLEEP RATE FOR ACTIVATED RANDOM WALK ON THE CYCLE WITH ONE CHIP PER VERTEX

HANNAH CAIRNS, SHIRSHENDU GANGULY, LIONEL LEVINE

ABSTRACT. Activated random walk (or ARW) is a random process with chips on a graph that may fall asleep or wake each other up. It has one parameter, the sleep rate λ. In the extreme case where ARW starts with one chip on every vertex of a cycle of length L, we find a precise boundary $\lambda = \frac{1}{2}\log L$ between 'fast sleep' and 'slow sleep' regions.

CONTENTS

1. ACTIVATED RANDOM WALK

We describe the activated random walk process. Let $G = (V, E)$ be an undirected graph, and let λ be a nonnegative real number called the *sleep rate*. Each vertex on the graph is in one of the following states:

- it can be empty;
- it can have one *sleeping* chip on it;
- or it can have any positive integer number of *activated* chips on it.

Date: October 24, 2022.

The graph starts in some initial configuration, and then proceeds step by step. At each step, we choose a vertex $v \in V$ with at least one activated chip on it. If there are none, the process has *stabilized* and we are done.

We are allowed to choose any suitable vertex we want, with the constraint that we can't ignore an active vertex forever: if a vertex is active at some point in the process, it must be chosen at some later time.

Once we have chosen a vertex with an activated chip on it, we flip a weighted coin that comes up heads with probability $q = \lambda/(1+\lambda)$.

- *If the coin is heads*, and v has only one activated chip, that chip falls asleep. If there are two or more chips on v, then nothing changes.
- *If the coin is tails*, then we move one of the chips on v to one of the adjacent vertices $w \sim v$, chosen uniformly at random.

If an activated chip lands on a vertex with a sleeping chip, the sleeping chip wakes up, so that there are now two activated chips on that vertex. When many chips are sleeping in a region, the arrival of a single activated chip can start a powerful chain reaction.

This is a so-called *abelian network*, meaning that neither the final configuration nor the number of times each vertex is chosen depends on the arbitrary choices which we make.

With our setup, if a vertex has more than one chip on it, none of the chips will move until our weighted coin comes up tails, which takes $\lambda + 1$ flips on average. If λ is large, a typical activated chip will follow a very lazy random walk until it finds an empty vertex, and then fall asleep right there.

2. ARW ON THE CYCLE WITH ONE CHIP PER VERTEX

Our graph will be the cycle of length L, and we'll start with one activated chip on every vertex. We run our process until all the chips fall asleep, and measure the total number of steps taken.

Our main theorem is as follows.

Theorem 2.1. *Suppose that we run the activated random walk process with sleep rate λ on the cycle of length L, with one chip on each vertex.*

If λ is less than $\frac{1-\varepsilon}{2}\log L$ for $\varepsilon > 0$, the expected sleep time is at least $e^{L^{\varepsilon}}$. If λ is greater than $\frac{1+\varepsilon}{2}\log L$, the expected time is at most $128L^2(1+\varepsilon)/\varepsilon$, and on the boundary line $\lambda = \log L$, the expected time is at most $400L^2\log L$.

The lower bound in the slow case, when $\lambda < \frac{1-\varepsilon}{2}\log L$, is Theorem 3.1. The upper bound in the fast case is Theorem 3.3.

In the fast case, the expected sleep time is $O(L^2)$, but the tail probabilities decay more slowly. This behavior is described in Theorem 3.10.

We also discuss the higher-dimensional analogue. To fall asleep quickly in three or more dimensions, the sleep rate has to be of order L^d, so high that all the chips are likely to fall asleep before any of them move. In two dimensions, the rate can be slightly smaller: we can fall asleep quickly if the rate is $\sim L^2/\log L$. These statements are proven in Theorems 4.3, 4.4.

2.1. **Sorted configurations.** Let's say that the process is *sorted* if there is one chip on each vertex, and the activated chips are in a single contiguous segment of the cycle. The cycle is symmetric, so all that matters is the length of the active segment.

The abelian property allows us to choose any vertex with an active chip on it. If the process is sorted, we always choose the rightmost vertex in the active segment. When we do that, the active chip will fall asleep with probability $\lambda/(1+\lambda)$. If that happens, then the process will still be organized, and there will be one fewer chip in the active segment.

If the chip doesn't fall asleep, which happens with probability $1/(1+\lambda)$, then it will move to one of the adjacent vertices. That vertex already had a chip on it, so there are now two chips, and they will both wake up if they weren't awake already. We continue to activate any vertex with two chips on it at each step. Eventually, a chip will fall back into the empty vertex.

Another way to think of this is that the chip which we started with carries out a simple random walk on the cycle until it returns to its original position, waking up every sleeping chip that it visits. This won't separate the sleeping segment into two pieces, because the walk starts from a vertex in the active segment. Therefore, once the chip returns, the process is again sorted.

2.2. **Big steps.** If the process is sorted, say with x activated chips, we run the process as above until it's sorted again with some random number X of active chips. We call this a *big step*.

If we start with $x = 0$, then all chips are asleep and the process has stopped. If not, then there is a single segment of vertices with active chips. We choose the active vertex on the right end of the segment. The process is sorted, so there is only one chip on it. With probability $\lambda/(1+\lambda)$, the chip falls asleep, and the process is sorted again with one fewer active chip.

If that doesn't happen, then the chip moves one vertex to the right or left, and begins a simple random walk on the cycle which stops when it reaches its original position. Every chip that is visited during the walk will wake up, although some may be already activated.

Let J be the maximum distance from the origin that is reached by the simple random walk before it stops. If the walk goes all the way around the cycle and comes back to the original position from the other side, let $J = L$. By the gambler's ruin calculation, $\mathbb{P}(J \geq j) = 1/j$ when $j \leq L$.

If the chip moves to the right, it enters the segment of sleeping chips, and it wakes up every chip it reaches before it returns to the starting vertex.

If the walk crosses the whole segment, then all the chips wake up. So the count of activated chips after the big step is $(J+x) \wedge L$.

If the chip moves to the left on the first step, then it has to cross the rest of the activated segment before it starts waking up sleeping chips. So in this case the number of activated chips is $((J+1) \vee x) \wedge L$. Here we add one to J because we moved one vertex left before starting, so we begin on the second vertex from the right in the activated segment.

Summarizing, if $x \in \{0, \dots, L\}$ is the total number of activated chips before a big step, then the number of activated chips afterward is

$$X = \begin{cases} & \text{with independent probability} \\ x-1 & q \\ ((J+1) \vee x) \wedge L & \frac{1}{2}(1-q) \\ (J+x) \wedge L & \frac{1}{2}(1-q) \end{cases}$$

We can use this transition as the single step of a Markov chain, which we will call the *total process*. It tracks the total number of activated chips in the random walk after each big step. Studying this Markov chain will give us answers about our activated random walk.

3. Lower and upper bounds on the total process

In this section we find lower and upper bounds on the expected time for the total process to reach 0 from L, proving Theorem 2.1. The proof relies heavily on the fact that the total process never decreases by more than one at any step, which lets us derive recurrences for several functions.

3.1. **A positive recurrence for the increments.** Let $h(x)$ be the expected number of steps that the total process takes to reach zero from x. Let $\delta h(x) = h(x+1) - h(x)$. We'll find a simple expression for $\delta h(x-1)$ in terms of $\delta h(x), \dots, \delta h(L-1)$.

If $x \neq 0$, then $h(x) = 1 + \sum_y p_{xy} h(y)$, where p_{xy} are the transition probabilities for the total process. Writing this out explicitly,

$$h(x) = 1 + qh(x-1) + \frac{1-q}{2} \sum_{j=1}^{\infty} \frac{h(((j+1) \vee x) \wedge L)}{j(j+1)} + \frac{1-q}{2} \sum_{j=1}^{\infty} \frac{h((j+x) \wedge L)}{j(j+1)}$$

for $x = 1, \dots, L$. For convenience, let's set $h(x) := h(L)$ for $x > L$, so that we can write this as

$$h(x) = 1 + qh(x-1) + \frac{1-q}{2} \sum_{j=1}^{\infty} \frac{h((j+1) \vee x)}{j(j+1)} + \frac{1-q}{2} \sum_{j=1}^{\infty} \frac{h(j+x)}{j(j+1)}.$$

Rearrange this to get the expression

$$q(h(x)-h(x-1)) = 1 + \frac{1-q}{2}\sum_{j=1}^{\infty}\frac{h((j+1)\vee x)-h(x)}{j(j+1)} + \frac{1-q}{2}\sum_{j=1}^{\infty}\frac{h(j+x)-h(x)}{j(j+1)}.$$

Here we have used the fact that $\sum_{j=1}^{\infty} 1/j(j+1) = 1$.

Replace $h(b)-h(a) = \delta h(a)+\cdots+\delta h(b-1)$ and collect the coefficients of each $h(j)$. For example, we rewrite

$$\sum_{j=1}^{\infty}\frac{h((j+1)\vee x)-h(x)}{j(j+1)} = \sum_{j=1}^{\infty}\sum_{\substack{i\geq x,\\ i<j+1}}\frac{\delta h(i)}{j(j+1)} = \sum_{i\geq x, j\geq i}\frac{\delta h(i)}{j(j+1)} = \sum_{i\geq x}\frac{\delta h(i)}{i}.$$

Do this for both sums and solve for $\delta h(x-1)$. We get the recurrence

$$\delta h(x-1) = \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x}^{\infty}\delta h(j)\left(\frac{1}{j}+\frac{1}{j-x+1}\right) \tag{1}$$

for $x = 1,\dots,L$, with the boundary condition $\delta h(x) = 0$ for $x \geq L$.

We can use this as a recurrence to estimate the function δh. Note that all the coefficients are nonnegative.

3.2. **A lower bound for the expected time.** When λ is smaller than $\frac{1}{2}\log L$, we can get a stretched-exponential lower bound for the expected time from this recurrence.

Theorem 3.1. *If $\lambda < \frac{1-\varepsilon}{2}\log L$, the expected sleep time is at least $e^{L^{\varepsilon}}$.*

Proof. Let $z := 1-e^{-2\lambda}$. Let $f(x) := z^{x-L+1}/q$. We'll prove by induction that $\delta h(x) \geq f(x)$, going downward from $x = L-1$ to 0.

First, $\delta h(L-1) = 1/q = f(L-1)$, so the base case holds. Suppose $\delta h(j) \geq f(j)$ for $j \geq x$. Then

$$\begin{aligned}
\delta h(x-1) &= \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x}^{L-1} \delta h(j)\left(\frac{1}{j} + \frac{1}{j-x+1}\right) \\
&\geq \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x}^{L-1} \frac{\delta h(j)}{j-x+1} \\
&\geq \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x}^{L-1} \frac{f(j)}{j-x+1} \\
&\geq \left(\frac{1}{q} - \frac{1}{2\lambda}\sum_{j=L}^{\infty} \frac{f(j)}{j-x+1}\right) + \frac{1}{2\lambda}\sum_{j=x}^{\infty} \frac{f(j)}{j-x+1}
\end{aligned}$$

We chose our constant z so that the sum $\sum_{j=1}^{\infty} z^j/j = \log 1/(1-z)$ is equal to 2λ, and so the second summand is

$$\frac{1}{2\lambda}\sum_{j=x}^{\infty}\frac{f(j)}{j-x+1} = \frac{z^{x-L}}{2\lambda q}\sum_{j=x}^{\infty}\frac{z^{j-x+1}}{j-x+1} = \frac{z^{x-L}}{2\lambda q}\sum_{j=1}^{\infty}\frac{z^j}{j} = \frac{z^{x-L}}{q} = f(x-1).$$

This also means that

$$\frac{1}{2\lambda}\sum_{j=L}^{\infty}\frac{f(j)}{j-x+1} \leq \frac{1}{2\lambda}\sum_{j=L}^{\infty}\frac{f(j)}{j-L+1} = \frac{1}{q},$$

so the term in parentheses is positive. Therefore, $\delta h(x-1) \geq f(x-1)$.

That's the induction step, so we can proceed by induction and conclude that $\delta h(x) \geq z^{x-L+1}/q$ for $0 \leq x \leq L-1$.

The expected time is $h(L) \geq \delta h(0)$, which is at least z^{-L+1}/q. However, z is at least q, because $(1-z)^{-1} = e^{2\lambda}$ is at least $1+\lambda = (1-q)^{-1}$, so we can use the simpler bound

$$h(L) \geq (1-e^{-2\lambda})^L \geq \exp(Le^{-2\lambda}).$$

We are assuming that $\lambda < \frac{1-\varepsilon}{2}\log L$, so $e^{-2\lambda} > e^{(1-\varepsilon)\log L} = L^{1-\varepsilon}$. Therefore, the expected sleep time is at least

$$\exp(Le^{-2\lambda}) \geq \exp(L/L^{1-\varepsilon}) = \exp(L^{\varepsilon}).$$

This proves the first half of Theorem 2.1. □

3.3. **An upper bound on the expected number of big steps.** If the sleep rate λ is at least $\frac{1}{2}\log L$, then the lower bound from the last section is 1. We can get a better bound just by observing that it takes at least one step for each chip to fall asleep, so it takes at least L big steps to get from L to 0.

In this section, we will get an upper bound on the expected number of big steps. If $\lambda \geq \frac{1+\varepsilon}{2}\log L$ for $\varepsilon > 0$, we get the bound $128L(1+\varepsilon)/\varepsilon$, and for any sleep rate $\lambda \geq \frac{1}{2}\log L$, we get a uniform bound $400L\log L$.

3.3.1. *Our goal: a function that satisfies the inequality.* We start with the recurrence from before:

$$\delta h(x-1) = \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x}^{L-1}\left(\frac{1}{j} + \frac{1}{j-x+1}\right)\delta h(j).$$

We can lower the top bound of the sum to $L-1$ because $\delta h(j) = 0$ for $j \geq L$.

We'll exhibit a function that satisfies the inequality

$$f(x) \geq \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x+1}^{L-1}\left(\frac{1}{j} + \frac{1}{j-x}\right) f(j). \tag{2}$$

A function like that must be an upper bound on δh by an easy induction, as we'll see in Theorem 3.6.[1]

The coefficients in the sum add up to about $\log L$ when $x \approx L$, and add up to about $2\log L$ when $x \approx 0$. So $\delta h(x)$ grows only slowly for most x, but it starts to grow sharply when x is close to zero. So our explicit upper bound will have to have a singularity at $x = 0$, but it shouldn't be too bad because we want to bound $h(L) \leq \sum \delta h(x) \leq \sum f(x)$ by something of order L.

It turns out that we can use functions of the form $a\log^n(L/x) + b$, and we spend the rest of this section finding a particular function that works.

3.3.2. *Wrestling with the inequality.* We'll start by proving a related inequality about a function ψ which has that form more or less. The proof uses two lemmata about a delicate and unpleasant integral inequality. Those are proven immediately afterward.

Once that ordeal is over, we write down an explicit upper bound for δh in Theorem 3.5 and prove it in Theorem 3.6, which gives us an upper bound for the expected number of big steps until sleep.

Theorem 3.2. *Let* $\psi(x) := \log^2(L/x) + \frac{1}{16}$ *for* $x \geq 1$ *and* $\psi(0) = 3\psi(1)$. *Then*

$$\psi(x) \geq \frac{1}{100\log L} + \frac{1}{\log L}\sum_{j=x+1}^{L-1}\left(\frac{1}{j} + \frac{1}{j-x}\right)\psi(j).$$

Note. The proof should go through for any function $\log^n(L/x) + c$ where $n \geq 2$ and $0 \leq c < 1/16$. The statement is probably true even when $c = 1$, although the approximations we use don't work.

[1]We can push this inequality back through the calculation earlier in reverse to see that it holds if and only if $F(X_n) + n$ is a supermartingale, where $F(x) = f(0) + \cdots + f(x-1)$ and X_n is the total process stopped at zero.

Proof. If $x = 0$, then the right side is at most

$$\frac{2\psi(1)}{\log L}\left(1+\frac{1}{2}+\cdots+\frac{1}{L-1}\right) \le 2\psi(1)\frac{\log L+1}{\log L} \le 3\psi(1).$$

If $x \ge 1$, then we can take $y = x/L$, $\delta = 1/L$ and plug it into Lemma 3.3. We do indeed have $0 < \delta \le y \le 1-\delta$, so we get the inequality

$$\varphi(y)\log 1/\delta - \varphi(y+\delta)\left(\frac{\delta}{y+\delta}+1\right) - \int_{y+\delta}^{1}\left(\frac{1}{t}+\frac{1}{t-y}\right)\varphi(t)\,dt \ge \frac{1}{100}.$$

where $\varphi(y) = \log^2(1/y)+\frac{1}{16}$ as in the lemma. Substitute $y = x/L, \delta = 1/L$, and also substitute $s = Lt$ in the integral:

$$\psi(x)\log L - \left(\psi(x+1)\left(\frac{1}{x+1}+1\right) + \int_{x+1}^{L}\left(\frac{1}{s}-\frac{1}{s-x}\right)\psi(s)\,ds\right) \ge \frac{1}{100}.$$

We have made the substitution $s = Lt$.

The integral is an upper bound on the part of the sum from $x+2$ to L:

$$\int_{x+1}^{L}\left(\frac{1}{s}-\frac{1}{s-x}\right)\psi(s)\,ds \ge \sum_{j=x+2}^{L-1}\left(\frac{1}{j}-\frac{1}{j-x}\right)\psi(j),$$

and the other part is the first summand, the one for $j = x+1$. Therefore,

$$\psi(x)\log L - \sum_{j=x+1}^{L-1}\left(\frac{1}{j}+\frac{1}{j-x}\right)\psi(j) \ge \frac{1}{100}.$$

Divide the whole thing by $\log L$ and move the sum to the right-hand side to get the inequality in the statement. □

We'll use this to show that $f(x) = 32\psi(x)/\varepsilon$ satisfies the inequality (2), but first we need the supporting Lemma 3.3 and another one, Lemma 3.4.

Lemma 3.3. *Let* $\varphi(y) := \log^2(1/y)+\frac{1}{16}$. *If* $0 < \delta \le y < 1-\delta$, *then*

$$\varphi(y)\log 1/\delta - \varphi(y+\delta)\left(\frac{\delta}{y+\delta}+1\right) - \int_{y+\delta}^{1}\left(\frac{1}{t}+\frac{1}{t-y}\right)\varphi(t)\,dt \ge \frac{1}{100}.$$

Note. The method of its construction is probably more interesting than the proof itself. The formulas were separated into several pieces and tested numerically to check which approximations appeared viable.[2]

[2]The tool for this was Jupyter with Python. Cython was used to calculate the sums, because otherwise experimentation would have been too slow. The Taylor series trick came from the observation that two of the large terms usually came close to cancelling out, and another smaller term was nearly equal to the remainder.

Proof. First we consider the larger range $0 < y < 1$ and $0 < \delta \leq 1 - y$. We'll write the left side of the inequality above as $F(y, \delta) + G(y, \delta)$ where

$$F(y,\delta) := \varphi(y)\log(1/\delta) - \int_{y+\delta}^{1} \frac{1}{t-y}\varphi(t)\,dt - \varphi(y+\delta)$$

is chosen so that it will mostly cancel out, and G is the rest.

Let $a := y + \delta$ and $A := \log 1/a$. Then the part that's left over is

$$\begin{aligned} G(y,\delta) &:= -\varphi(y+\delta)\frac{\delta}{y+\delta} - \int_{y+\delta}^{1} \frac{\varphi(t)}{t}\,dt \\ &= -A^2 - \frac{1}{16} + \frac{y}{a}\left(A^2 + \frac{1}{16}\right) - \frac{A^3}{3} - \frac{A}{16}. \end{aligned}$$

We evaluate the integral $\int_{y+\delta}^{1} \varphi(t)/t\,dt = -\log(y+\delta)^3/3 - \log(y+\delta)/16$, write $\delta/(y+\delta) = 1 - y/a$, and expand everything out.

We put this part aside for now and focus on F. Differentiate F on the second parameter δ, and then write the result as an integral:

$$\begin{aligned} \frac{\partial F}{\partial \delta} &= -\frac{\varphi(y) - \varphi(y+\delta) - (-\delta)\varphi'(y+\delta)}{\delta} \\ &= -\frac{1}{\delta}\int_{y}^{y+\delta} (t-y)\varphi''(t)\,dt. \end{aligned}$$

In the second step we use $\varphi(y) - \varphi(a) - (y-a)\varphi'(x) = \int_a^y (y-t)\varphi''(t)\,dt$, the formula for the remainder of a first-order Taylor series, with $a = y + \delta$.

The next step is to write $F(y, \delta)$ as the sum of $F(y, 1-y)$ and the integral of the derivative, and then change the order of integration. In what follows we write $\theta(y) := F(y, 1-y)$.

$$\begin{aligned} F(y,\delta) &= F(y,1-y) + F(y,\delta) - F(y,1-y) \\ &= \theta(y) - \int_{\delta}^{1-y} \frac{\partial F}{\partial \delta}(y,s)\,ds \\ &= \theta(y) + \int_{\delta}^{1-y} \left[\frac{1}{s}\int_{y}^{y+s} (t-y)\varphi''(t)\,dt\right] ds \\ &= \theta(y) + \int_{y}^{1} (t-y)\varphi''(t)\left[\int_{\delta\vee(t-y)}^{1-y} \frac{1}{s}\,ds\right] dt \\ &= \theta(y) + \int_{y}^{1} (t-y)\varphi''(t)\log\frac{1-y}{\delta\vee(t-y)}\,dt. \end{aligned}$$

The double integral is over s, t with $\delta \leq s \leq 1 - y$ and $y \leq t \leq y + s$. This shape is a right triangle with one corner truncated. When those bounds are reversed, the bound on t becomes $y \leq t \leq 1$, and s now has to satisfy three inequalities: $\delta \leq s \leq 1 - y$ and $t - y \leq s$.

The integrand is nonnegative, because $\varphi''(t) = 2(1+\log(1/t))/t^2$, so shrinking the integral will make it smaller. Move the lower endpoint up to $y+\delta$. Then $t-y \geq \delta$ on the interval of integration, and $y \leq t \leq 1$, so we can bound the fraction in the logarithm:

$$\frac{1-y}{\delta \vee (t-y)} = \frac{1-y}{t-y} \geq \frac{1}{t}.$$

Then we get a simpler bound that has a closed form.

$$\begin{aligned} F(y,\delta) &\geq \theta(y) + \int_{y+\delta}^{1} (t-y)\varphi''(t)\log(1/t)\,dt \\ &= \theta(y) + 2y\left(1 - \frac{1}{a} + \frac{A}{a} - \frac{A^2}{a}\right) + A^2 + \frac{2A^3}{3}. \end{aligned}$$

The integral can be evaluated with the substitution $t = e^{-s}$.

Now, we look back to see what $G(y,\delta)$ is, and add it. The A^2 terms cancel out completely, the A^2/a terms partly cancel out, and we wind up with

$$F(y,\delta) + G(y,\delta) \geq \theta(y) + cy + d,$$

where c and d don't depend on y, only on the sum $a = y+\delta$:

$$c = 2 - \frac{2}{a} + \frac{2A}{a} - \frac{A^2}{a} + \frac{1}{16a}, \qquad d = -\frac{1}{16} + \frac{A^3}{3} - \frac{A}{16}.$$

This looks awful, and it is awful, but it will all work out. In fact, by the lemma after this one, $\theta(y) + cy + d \geq 1/100$ whenever $a/2 \leq y < a \leq 1$. Those inequalities hold by our assumption that $\delta \leq y \leq 1-\delta$. Therefore, $F+G$ is at least $1/100$, and the theorem is true. □

We have pushed our most tedious calculations into this next lemma, but it isn't all that bad. We just have to get out the calculator and check some bounds on $\theta(y)$, and then bound some polynomials.

Lemma 3.4. *If $a/2 \leq y < a \leq 1$, then*

$$\theta(y) + cy + d \geq 1/100,$$

where $\theta(y) := F(y, 1-y)$ is defined in Lemma 3.3 and

$$c = 2 - \frac{2}{a} + \frac{2A}{a} - \frac{A^2}{a} + \frac{1}{16a}, \qquad d = -\frac{1}{16} + \frac{A^3}{3} - \frac{A}{16}, \qquad A := \log 1/a.$$

Proof. First of all, we write out $\theta(y)$ explicitly. The upper and lower limits of the integral are the same, so it drops out and

$$\begin{aligned} \theta(y) = F(y, 1-y) &= \varphi(y)\log 1/(1-y) - \varphi(1) \\ &= \left(\log^2(1/y) + \frac{1}{16}\right)\log 1/(1-y) - \frac{1}{16}. \end{aligned}$$

We want to get lower bounds on this function, which we will do by brute force. If $y_0 \le y_1$, then we have the obvious lower bound from monotonicity

$$(\log^2(1/y_1) + \tfrac{1}{16})\log(1/(1-y_0)) - \tfrac{1}{16} \le \theta(y) \qquad \text{on } [y_0, y_1].$$

We use this on each of the intervals $[\frac{0}{20}, \frac{1}{20}], \dots, [\frac{19}{20}, \frac{20}{20}]$ to get lower bounds:

y_0	y_1	bound	y_0	y_1	bound
$^0/_{20}$	$^1/_{20}$	-0.0625	$^{10}/_{20}$	$^{11}/_{20}$	0.2285
$^1/_{20}$	$^2/_{20}$	0.2126	$^{11}/_{20}$	$^{12}/_{20}$	0.1957
$^2/_{20}$	$^3/_{20}$	0.3232	$^{12}/_{20}$	$^{13}/_{20}$	0.1648
$^3/_{20}$	$^4/_{20}$	0.3686	$^{13}/_{20}$	$^{14}/_{20}$	0.1366
$^4/_{20}$	$^5/_{20}$	0.3802	$^{14}/_{20}$	$^{15}/_{20}$	0.1123
$^5/_{20}$	$^6/_{20}$	0.3724	$^{15}/_{20}$	$^{16}/_{20}$	0.0931
$^6/_{20}$	$^7/_{20}$	0.3528	$^{16}/_{20}$	$^{17}/_{20}$	0.0805
$^7/_{20}$	$^8/_{20}$	0.3261	$^{17}/_{20}$	$^{18}/_{20}$	0.0771
$^8/_{20}$	$^9/_{20}$	0.2951	$^{18}/_{20}$	$^{19}/_{20}$	0.0874
$^9/_{20}$	$^{10}/_{20}$	0.2620	$^{19}/_{20}$	$^{20}/_{20}$	0.1247

The function is roughly N-shaped. The things to notice in this table are:

- $\theta(y)$ is always at least $-\frac{1}{16}$.
- $\theta(y) \ge 0.135 = \frac{1}{8} + \frac{1}{100}$ on $[1/2e^2, 1/e]$.
- $\theta(y) \ge 0.0725 = \frac{1}{16} + \frac{1}{100}$ on $[1/2e, 1]$.

Four places of precision are good enough to check this, if we also remember that $1/8 \le 1/e^2$ and $1/e \le 1/2$.

Now, $a/2 \le y < a$ by assumption, so $cy + d$ is at least the minimum of $ca + d$ and $ca/2 + d$. We can get a simple lower bound for $ca + d$ as follows:

$$\begin{aligned} ca + d &= \left(2a - 2 + 2A - A^2 + \frac{1}{16}\right) + \left(-\frac{1}{16} + \frac{A^3}{3} - \frac{A}{16}\right) \\ &= 2a - 2 + \frac{31}{16}A - A^2 + \frac{A^3}{3} \\ &\ge -\frac{A}{16}. \end{aligned}$$

The last step uses the inequality $a = e^{-A} \ge 1 - A + A^2/2 - A^3/6$, which is valid for all $A \ge 0$.[3]

[3]Integrate $e^{-x} \ge 0$ four times to get the sequence of inequalities $1 - e^{-x} \ge 0$, $x - 1 + e^{-x} \ge 0$, $x^2/2 - x + 1 - e^{-x} \ge 0$, $x^3/6 - x^2/2 + x - 1 + e^{-x} \ge 0$.

We will also set $P(A) := -2+\frac{31}{16}A - A^2 + A^3/3$, so $ca+d = 2a+P(A)$. For later, we record the fact that $P'(A) = \frac{31}{16} - 2A + A^2$ is at least $(A-1)^2$, which is always positive, so P is an increasing function of A.

Now we will show that $\theta(y)+ca+d \geq \frac{1}{100}$ everywhere by breaking the interval $[0,1]$ into three pieces, $[0,1/e^2] \cup [1/e^2, 1/e] \cup [1/e, 1]$.

Case 1a. Let $0 \leq a \leq 1/e^2$. Then $A \geq 2$, and $ca+d \geq P(A) \geq P(2) = 13/24$. Therefore, $\theta(y)+ca+d \geq \frac{13}{24} - \frac{1}{16} \geq \frac{1}{100}$.

Case 1b. Let $1/e^2 \leq a \leq 1/e$. Then $1/2e^2 \leq y \leq 1/e$, so $\theta(y) \geq \frac{1}{8} + \frac{1}{100}$. Therefore, $\theta(y)+ca+d \geq \frac{1}{8} + \frac{1}{100} - \frac{1}{16}A \geq \frac{1}{100}$.

Case 1c. Let $1/e \leq a \leq 1$. Then $1/2e \leq y \leq 1$, so $\theta(y) \geq \frac{1}{16} + \frac{1}{100}$. Therefore, $\theta(y)+ca+d \geq \frac{1}{16} + \frac{1}{100} - \frac{1}{16}A \geq \frac{1}{100}$.

We can do the same thing to get a lower bound for $ca/2+d$.

$$\begin{aligned} ca/2+d &= \left(a-1+A-\frac{A^2}{2}+\frac{1}{32}\right)+\left(-\frac{1}{16}+\frac{A^3}{3}-\frac{A}{16}\right) \\ &= a - \frac{33}{32} + \frac{15}{16}A - \frac{A^2}{2} + \frac{A^3}{3} \\ &\geq -\frac{1}{32} - \frac{A}{16} + \frac{A^3}{6}. \end{aligned}$$

Again, the last step uses the inequality $a = e^{-A} \geq 1-A+A^2/2-A^3/6$.

Let $Q(A)$ be the above polynomial, that is, $Q(A) := -\frac{1}{32} - \frac{1}{16}A + \frac{1}{6}A^3$. Then the derivative of Q on A is $(8A^2-1)/16$, which is positive when $A > 1/\sqrt{8}$ and negative when $0 \leq A < 1/\sqrt{8}$. Therefore, the minimum of Q is

$$\min_{A\geq 0} Q(A) = Q\left(\frac{1}{\sqrt{8}}\right) = -\frac{1}{24\sqrt{8}} - \frac{1}{32} \geq -\frac{1}{16}$$

and the minimum of Q on $[2,\infty)$ is $Q(2) = \frac{113}{96} \geq 1$.

We will show that $\theta(y)+ca/2+d$ is at least $\frac{1}{100}$ by breaking the interval into two pieces, $[0,1] = [0,1/e^2] \cup [1/e^2, 1]$.

Case 2a. Let $0 \leq a \leq 1/e^2$, or $A \geq 2$. Then $ca/2+d \geq Q(A) \geq Q(2) \geq 1$, and $\theta(y) \geq -\frac{1}{16}$, so the sum $\theta(y)+ca/2$ is certainly at least $\frac{1}{100}$.

Case 2b. Let $1/e^2 \leq a \leq 1$. Then $1/2e^2 \leq y \leq 1$, so $\theta(y)$ is at least $\frac{1}{16} + \frac{1}{100}$, and $Q(A) \geq Q(1/\sqrt{8}) \geq -\frac{1}{16}$, so $\theta(y)+ca/2+b$ is at least $\frac{1}{100}$.

We've gone case by case and proven that both of the bounds are always at least $\frac{1}{100}$, so $\theta(y)+cy+d \geq \frac{1}{100}$ and the lemma is true. □

We didn't get much conceptual insight from the proof of this lemma. All we've done is verify that a function of the form $\log^n 1/y + c$ really does satisfy the inequality in Lemma 3.3. But, having done this tedious work, we now cash it in to get the inequality that we promised earlier.

Theorem 3.5. *Suppose* $0 \le \varepsilon$, $\lambda \ge \frac{1+\varepsilon}{2}\log L$, $L \ge 8$. *Let* $f := 32\psi\frac{1+\varepsilon}{\varepsilon}$ *or* $f := 100\psi\log L$, *whichever is smaller. Then for all* $x = 0,\dots,L-1$,

$$f(x) \ge \frac{1}{q} + \frac{1}{2\lambda}\sum_{j=x+1}^{L-1}\left(\frac{1}{j} + \frac{1}{j-x}\right)f(j).$$

Proof. If $f = 100\psi\log L$, we start with the inequality from Theorem 3.2, multiply both sides of it by $100\log L$, and decrease the coefficient of the sum from $1/\log L$ to $1/2\lambda$. That gives us the above inequality immediately.

In the case where $f = 32\psi(1+\varepsilon)/\varepsilon$, we will exploit the hard-won fact that $\psi \ge \frac{1}{16}$. Let $t := 1 - \log(L)/2\lambda$, which by our assumption is at least $\varepsilon/(1+\varepsilon)$. Write $1 = t + (1-t)$ and use the inequality from Theorem 3.2.

$$\begin{aligned}\psi(x) &= t\psi(x) + (1-t)\psi(x)\\ &\ge t\psi(x) + (1-t)\frac{1}{\log L}\sum\left(\frac{1}{j} + \frac{1}{j-x+1}\right)\psi(j)\\ &= t\psi(x) + \frac{1}{2\lambda}\sum\left(\frac{1}{j} + \frac{1}{j-x+1}\right)\psi(j).\end{aligned}$$

Multiply this all by $32(1+\varepsilon)/\varepsilon \ge 32/t$, and recall that $\psi(x) \ge \frac{1}{16}$.

$$\begin{aligned}f(x) &\ge 2 + \frac{1}{2\lambda}\sum\left(\frac{1}{j} + \frac{1}{j-x+1}\right)f(j)\\ &\ge \frac{1}{q} + \frac{1}{2\lambda}\sum\left(\frac{1}{j} + \frac{1}{j-x+1}\right)f(j).\end{aligned}$$

In the last step we used the assumption that $L \ge 8$, so the sleep rate is at least $\frac{1}{2}\log L \ge 1$ and $1/q \le 2$. □

3.3.3. *An upper bound on the number of big steps.* We apply the analytical nonsense in the previous section to bound the expected number of big steps.

Theorem 3.6. *Start the activated random walk on the cycle of length L with sleep rate* $\lambda \ge \frac{1+\varepsilon}{2}\log L$ *and one chip on each vertex, all awake. The expected number of big steps before the walk sleeps is at most*

$$\min\left\{\frac{128(1+\varepsilon)}{\varepsilon}, 400\log L\right\}L.$$

Proof. Let f be as in the previous theorem, that is, $f = 32\psi(1+\varepsilon)/\varepsilon$ or $f = 100\psi\log L$, whichever is smaller.

Write $f(x;L)$ to make the dependence on L explicit. Let $h(x;L)$ be the expected number of big steps that it takes for the total process to reach zero from x, as in Section 3.3. We proved that the successive differences $\delta h(x;L) := h(x+1;L) - h(x;L)$ satisfy a recurrence:

$$\delta h(x;L) = \frac{1}{q} + \frac{1}{2\lambda} \sum_{j=x+1}^{L-1} \left(\frac{1}{j} + \frac{1}{j-x} \right) \delta h(j;L).$$

Suppose $f(j;L) \geq \delta h(j;L)$ for $j = x+1, \ldots, L-1$. By Theorem 3.5,

$$\begin{aligned} f(x;L) &\geq \frac{1}{q} + \frac{1}{2\lambda} \sum_{j=x+1}^{L-1} \left(\frac{1}{j} + \frac{1}{j-x} \right) f(j;L) \\ &\geq \frac{1}{q} + \frac{1}{2\lambda} \sum_{j=x+1}^{L-1} \left(\frac{1}{j} + \frac{1}{j-x} \right) \delta h(j;L) \\ &= \delta h(x;L), \end{aligned}$$

and by induction downward we have $f \geq \delta h$ for $x = 0, \ldots, L-1$. The base case is $x = L-1$, where $f(x;L) \geq 1/q$ and $\delta h(x;L) = 1/q$.

Therefore, the expected number of big steps that it takes to reach 0 from L is at most $f(0) + \cdots + f(L-1)$. We have an explicit formula for f, so we can estimate that. First, the sum of $\psi(0) + \cdots + \psi(L-1)$ is

$$\begin{aligned} \sum_{x=0}^{L-1} \psi(x;L) &= \psi(0;L) + \sum_{x=1}^{L-1} \psi(x;L) \\ &\leq \psi(0;L) + \int_0^L \log^2(L/t) + \frac{1}{16}\, dt. \\ &= \psi(0;L) + 33L/16 \\ &\leq 4L. \end{aligned}$$

Here the integral of $\int_0^L \log^2(L/t)\, dt$ is $2L$, and in the last step we use the inelegant bound $\psi(0;L) = 3\log^2 L + 3/16 \leq 31L/16$. This is clearly true for large L, and in fact it holds for all $L \geq 1$.[4]

If we multiply this by the minimum of $32(1+\varepsilon)/\varepsilon$ and $100 \log L$, we get the result. The expected time to reach 0 from L is at most

$$h(L;L) \leq \min \left\{ \frac{128(1+\varepsilon)}{\varepsilon}, 400 \log L \right\} L.$$

This is what we wanted. □

[4]The derivative $\frac{d}{dL}(31L/16 - 3\log^2 L - 3/16) = 31/16 - 6\log L/L$ is negative when $L \geq 8$, because $6\log L/L \leq 12/e^2 \leq 12/7$, and the bound holds for $L = 1, \ldots, 8$.

Of course, the concept of 'big steps' was something we introduced to simplify the situation. We're really trying to figure out how long the model takes to fall asleep, measured in terms of small steps. In the next section, we'll bound the expected number of small steps and estimate the tail.

3.4. **The number of small steps until sleep.** In this section, we get an upper bound on the expected sleep time, which proves the second part of Theorem 2.1. We will also estimate the tail, which turns out to be long.

Theorem 3.7. *The expected sleep time is at most* $128(1+\varepsilon)L^2/\varepsilon$*, and also at most* $400L^2 \log L$*.*

Proof. Let $T_q(x)$ be the time until a q-lazy random walk started at x leaves the set $\{1,\dots,L-1\}$. It's well-known that $\mathbb{E}[T_q(x)] = x(L-x)/(1-q)$.[5]

At every big step, we have probability q of falling asleep, which takes one small step. Otherwise, we take one step and then begin a q-lazy random walk on the cycle, stopping when we return to our original position. This is the same as a walk that's started at 1 and stops when it leaves $\{1,\dots,L-1\}$, so each big step takes $q+(1-q)(1+\mathbb{E}[T_q]) = L$ small steps on average.

Naively one would guess that the expected number of small steps is L times the expected number of big steps, and this turns out to be correct. However, the time taken is highly correlated with the state of the total process: the more time we take, the more chips we may wake up. We'll prove the claim that the expected number of small steps is L times the number of big steps to make sure that that intuition is not misleading.

Let X_n be the number of chips awake after the n-th big step, let S_n be the number of small steps taken in the first n big steps, and let T be the number of big steps it takes to hit zero for the first time, so that the number of small steps it takes to hit zero is S_T. Let $\mathscr{F}_n = \sigma(X_1,\dots,X_n,S_1,\dots,S_n)$ be the filtration containing the history of the walk. Let $\mathbb{E}_x$ be the expectation when the walk is started at x, so that for example $\mathbb{E}_0[T] = 0$ and $\mathbb{E}_x[T] = h(x)$.

If x is not zero, then we take at least one big step before stopping. By the tower rule, linearity, and the Markov property of the total process,

$$\mathbb{E}_x[S_T] = \mathbb{E}_x[S_1 + \mathbb{E}[S_T - S_1 \mid \mathscr{F}_1]] = L + \mathbb{E}_x[\mathbb{E}_{X_1}[S_T]]$$

when $x \neq 0$. The expectations are all clearly finite, so this is valid.

Let $m(x) := \mathbb{E}_x[S_T] - Lh(x)$. Then $m(x) = \sum_y p_{xy} m(y)$, including at zero, so m is a harmonic function on a finite irreducible Markov chain. It must be

[5]Here's a proof: let X_n be a lazy random walk started at x and T be the first time it leaves $\{1,\dots,L-1\}$. Then $M_n := X_n(L-X_n)/(1-q) - n$ is a martingale with bounded increments and T is a stopping time with $\mathbb{E}[T] < \infty$. By Doob's optional stopping theorem, $\mathbb{E}[M_T] = 0$, so $x(L-x)/(1-q) = \mathbb{E}[T]$.

a constant, and $m(0)=0$, so it's zero. Therefore, $\mathbb{E}_x[S_T]=Lh(x)$, and

$$\mathbb{E}_L[S_T]=Lh(L;L)\le\begin{cases}128(1+\varepsilon)L^2/\varepsilon\\400L^2\log L\end{cases}$$

which is what we want to prove. □

3.4.1. *The tail of the sleep time.* As we mentioned in the last section, when we take a big step, if the chip doesn't fall asleep, it begins a q-lazy random walk on the cycle, stopping when it returns to its original position.

Let's say that the walk 'gets stranded' if it reaches the midpoint of the cycle, which occurs with probability $2/L$ by the gambler's ruin estimate. If a walk does get stranded, then its return time decays roughly like an exponential random variable, with mean of order $L^2/(1-q)$.

In what follows, let $\sigma:=L^2/(1-q)\pi^2$.

Lemma 3.8. *Let a q-lazy random walk be started at $\lfloor L/2\rfloor$. Let T be the first time that it hits $\{0,L\}$. If L is sufficiently large and $q\ge 1/2$, then*

$$\mathbb{P}[T\ge t]\ge\mathbb{P}[U\ge t]=e^{-2t/\sigma},$$

where U is exponential with mean $\sigma/2=L^2/2(1-q)\pi^2$.

Proof. Let the lazy walk be X_t. We couple it with another q-lazy random walk as follows.

Let $\psi(y):=C\sin\pi y/L$, where C is chosen so that $\sum_y\psi(y)=1$. Let Y_t be a lazy random walk with starting point distributed according to ψ. Then,

$$q\psi(y)+\frac{1-q}{2}\psi(y+1)+\frac{1-q}{2}\psi(y-1)=\lambda\psi(y),$$

where $\lambda=(1-q)\cos\pi/L+q$, so at any time t, $\mathbb{P}[Y_t=y]=\lambda^t\psi(y)$ as long as $1\le y\le L-1$. Let U be the first time that this walk hits $\{0,L\}$. Then U has a simple distribution: $\mathbb{P}[U\ge t]=\sum_{y=1}^{L-1}\mathbb{P}[Y_{\lceil t\rceil-1}=y]=\lambda^{\lceil t\rceil-1}\ge\lambda^t$.

If L is large enough, then $\cos\pi/L\ge e^{-2\pi^2/L^2}$ by comparing power series, so $\lambda^t\ge(1-q)e^{-2t\pi^2/L^2}+qe^0\ge e^{-2t(1-q)\pi^2/L^2}$ by the convexity. In other words, we have $\mathbb{P}[U\ge t]\ge e^{-2t/\sigma}$.

If we could couple these walks in such a way that Y always hits the boundary of the interval before X, then $T\ge U$ and we would have the result. We do that in the following way:

- If $Y_t\in\{0,L\}$, then it stays there and X_t takes a q-lazy random step.
- If $Y_t\notin\{0,L\}$, then:
  - If $X_t\notin\{Y_t,L-Y_t\}$, we choose one of X,Y uniformly at random. That walk takes a $(1/2+q/2)$-lazy random step.
  - If $X_t=Y_t$, then $X_{t+1}=Y_{t+1}$, and both take a q-lazy random step.

– If $X_t = L - Y_t$ and $X_t \neq Y_t$, then $X_{t+1} = L - Y_{t-1}$ and both walks take a q-lazy random step in opposite directions.

Both walks considered independently are q-lazy random walks. If X hits the boundary, then it hits one of Y_t or $L - Y_t$ first and couples with it. So with this coupling, $T \geq U$, and we get the lower bound in the statement. □

We'll prove an upper bound as well, although this bound will be about walks that start at 1.

Lemma 3.9. *Let a q-lazy random walk be started at* 1. *Let T be the first time that it hits* $\{0, L\}$. *If L is sufficiently large and* $q \geq 1/2$, *then*

$$\mathbb{P}[T \geq t+1] \leq \mathbb{P}[U \geq t] = e^{-t/4\sigma},$$

where U is exponential with mean $4\sigma = 4L^2/(1-q)\pi^2$.

Proof. Let X_t be this walk. We repeat the proof of the last lemma, coupling to another walk Y_t, but in this case, $X_0 = 1$ and $Y_0 \in \{1, \dots, L-1\}$, so the hitting time of X is smaller than that of Y.

Recall from the last lemma that $\mathbb{P}[U \geq t] = \lambda^{\lceil t \rceil - 1}$, so

$$\mathbb{P}[T \geq 1 + tL^2/(1-q)] \leq \mathbb{P}[U \geq 1 + tL^2/(1-q)] \leq \lambda^{tL^2/(1-q)}.$$

In this case, we use the bound $\lambda = (1-q)\cos \pi/L + q \leq e^{-\pi^2(1-q)/4L^2}$, which holds for sufficiently large L again by comparing power series. We get the desired bound, $\mathbb{P}[T \geq 1 + tL^2/(1-q)] \leq e^{-\pi^2 t/4}$. □

With these upper and lower bounds for q-lazy random walk, we can show that activated random walk on the cycle can take a long time to go to sleep.

Lemma 3.10. *Start activated random walk on the cycle of length* $L \geq 8$ *with one chip per vertex and sleep rate* $\lambda \geq \frac{1}{2}\log L$. *If we start with every chip active, let T be the time to sleep. Then*

$$\frac{1}{2\lambda} e^{-2t\pi^2/L^2 \log L} \leq \mathbb{P}[T \geq t] \leq 3e^{-t/3600L^2 \log L}.$$

Proof. Let t_{sleep} be the expected number of small steps until sleep. This is at most $400L^2 \log L$ by Theorem 3.7. By Markov's inequality, we fall asleep with probability at least $\frac{1}{2}$ before $2t_{sleep}$ small steps.

If we haven't fallen asleep at that point, we start waiting for the end of the current big step. (This might take a lot of small steps: we're waiting for a q-lazy random walk to hit the boundary.) Once that's done, we start over: we wait another $2t_{sleep}$ steps and so on. Let N be the number of repetitions.

Let T_n be the time that we spend waiting for a big step at the end of the n-th repetition. The worst case is that it starts out stranded.

We couple this process to the simpler process that goes as follows. To start with, we wait $2t_{sleep}$ steps, then flip a fair coin. If the coin is heads, we wait for time $U_n + 1$, where U_n is an exponential random variable with mean 4σ, and then go back to the start. By Lemma 3.9, $\mathbb{P}[T_n \geq t] \leq \mathbb{P}[U_n + 1 \geq t]$.

Let M be the number of repetitions of this simpler process. We can couple these together so that $N \leq M$ and $T_n \leq U_n + 1$, and once we have done that, the sum $2t_{sleep}M + (U_1 + 1) + \cdots + (U_M + 1)$ is an upper bound on T.

We proceed to get a bound on the tail. Here $M \sim \text{Geom}(\frac{1}{2})$, so

$$\mathbb{P}[(2t_{sleep} + 1)M \geq t/2] \leq \mathbb{P}[M \geq t/6t_{sleep}] \leq 2^{1-t/6t_{sleep}} \leq 2e^{-t/9t_{sleep}}.$$

The rest of the sum, $U_1 + \cdots + U_M$, is the sum of a $\text{Geom}(\frac{1}{2})$ number of exponential random variables with mean 4σ, where $\sigma = L^2/\pi^2(1-q)$ as in the start of this section. Such a sum is itself exponentially distributed with mean 8σ,[6] so we get the manageable bound

$$\mathbb{P}[U_1 + \cdots + U_M \geq t/2] \leq e^{-t/16\sigma}.$$

Therefore,

$$\begin{aligned}\mathbb{P}[T \geq t] &\leq \mathbb{P}[2t_{sleep}M \geq t/2] + \mathbb{P}[U_1 + \cdots + U_M \geq t/2] \\ &\leq 2e^{-t/9t_{sleep}} + e^{-t/16\sigma} \\ &\leq 3e^{-t/7200L^2\lambda}\end{aligned}$$

because $9t_{sleep}$ and $16\sigma = 16L^2(1+\lambda)/\pi^2$ are both less than $7200L^2\lambda$. This gives us the upper bound on the tail.

To get the lower bound (and justify the scale of the upper bound), we estimate the probability that one of the big steps gets stranded.

We must make at least L big steps before the system falls asleep. At each big step, the coin comes up tails with probability $1 - q = 1/(1+\lambda)$, and the walk reaches the midpoint with probability $2/L$. The probability that one of the steps gets stranded is at least $1 - (1 - 1/L(1+\lambda))^L$, and we'll bound that below by $1/2\lambda$, which is valid for sufficiently large L.

If a step does get stranded, then by Lemma 3.8, the probability that it takes at least t steps to get back to the edge is at least $e^{-2t/\sigma}$. Our assumption is that $\lambda \geq \frac{1}{2}\log L$, so $\sigma = L^2(1+\lambda)/\pi^2 \geq L^2 \log L/2\pi^2$, and.

$$\mathbb{P}[T \geq t] \geq \frac{1}{2\lambda}e^{-2t/\sigma} \geq \frac{1}{2\lambda}e^{-2t\pi^2/L^2\log L}.$$

This is the lower bound that we claimed. □

Even when the sleep rate is very large, the lower bound on the tail probabilities still has the same scale: $\mathbb{P}[T \geq t] = \Omega(e^{-t/\sigma})$ for λ fixed and $t \to \infty$.

[6] Suppose we start an exponential clock at rate r. When it rings, we stop with probability p, and otherwise we restart the clock. The time until we stop is a sum of $\text{Geom}(p)$ exponentials of rate r, but it's also an exponential random variable with rate rp.

Increasing the sleep rate decreases the chance that one of the walks will get stranded, but that doesn't have much effect on the tail probabilities.

4. Higher dimensions

One higher-dimensional analogue of the cycle is the torus $T_d := \mathbb{Z}^d/(L\mathbb{Z}^d)$ with edge set $\{\{\mathbf{x}, \mathbf{x}+\mathbf{e}_j\} : \mathbf{x} \in T_d, 1 \le j \le d\}$. The situation is simpler on the torus. For the sleep time to be small, the sleep rate must be of order L^d, so high that all chips are likely to fall asleep before any of them move.

We again break the ARW process up into a series of 'big steps,' using the abelian property. At the start of every big step, each vertex has one chip on it. If all the chips are asleep, we're done. Otherwise, we pick some active chip, if there are any, and flip the coin. If it comes up heads, the chip falls asleep and the big step is over. Otherwise, it moves to a neighbour, and then it does a q-lazy random walk until it returns to its original position, waking up every chip it encounters. Once it returns, we have one chip on each vertex again, and we make our next big step. We aren't using the concept of 'sorted configurations' in higher dimensions.

If $x, x' \in \mathbb{Z}/L\mathbb{Z}$, denote the graph distance on the cycle by $d(x,x')$, and if we have two vertices $\mathbf{x}, \mathbf{x}' \in T_d$, let $d(\mathbf{x}, \mathbf{x}')$ be $\max\{d(x_1,x_1'), \ldots, d(x_d,x_d')\}$, the maximum distance between corresponding coordinates. The largest possible distance between two vertices is $\lfloor L/2 \rfloor$.

Then we have the lemma:

Lemma 4.1 (Half-visited lemma)**.** *If* $\mathbf{x}$ *is as far as possible from* $\mathbf{x}'$*, then with probability at least* 2^{-d-4}*, a simple random walk from* $\mathbf{x}$ *and stopped at* $\mathbf{x}'$ *visits at least half the vertices of the torus.*

We'll prove this later, in Section 4.3. For us, this means that if a single activated chip leaves its starting point and does a random walk, and that walk reaches a vertex at the maximum possible distance, then with constant probability it will wake up at least half the other chips before returning.

4.1. Bounds on the expected sleep time for dimension ≥ 3.

In this case, the sleep time must be of order L^d for the system to fall asleep quickly.

We'll first prove this lemma, which says that a single chip has probability of waking up the whole torus:

Lemma 4.2. *The probability that a random walk started at* 0 *visits at least half of the vertices in* T_d *before returning is at least* 2^{-d-5}.

Proof. It's well-known that the probability that a simple random walk on $\mathbb{Z}^3$ never returns to its starting point is greater than zero. In fact, the probability is $1/3I_3 = 0.659462+$, where I_3 is a certain triple integral defined in [3]. If $d > 3$, the probability is at least as large, so in any case it's at least $1/2$.

We can think of the walk on the torus as a walk on the infinite lattice $\mathbb{Z}^d$ modulo the larger lattice $L\mathbb{Z}^d$. If the walk on the infinite lattice never returns to its starting point, then the corresponding walk on the torus can only return to the start after it reaches distance $\lfloor L/2 \rfloor$ from the start.

By Lemma 4.1, if the walk does reach distance $\lfloor L/2 \rfloor$, then it visits at least half of the vertices with probability at least 2^{-d-4}. So, a walk on the torus visits at least half the chips with probability at least 2^{-d-5}.

Theorem 4.3. *Let T be the sleep time of ARW on the torus T_d, $d \geq 3$, with one chip on each vertex and sleep rate $\lambda \geq 1$. Let $L \geq 2$. Then*

$$\frac{1}{8}\exp\left(\frac{L^d}{2^{d+7}\lambda}\right) \leq \frac{\mathbb{E}[T]}{L^{2d}} \leq \exp\left(\frac{L^d+1}{\lambda}\right).$$

Proof. Upper bound. Let N be a counter, starting at zero. At the start of each big step, we add one to N if our coin comes up heads, and we reset it to zero if it's tails. (Even after all the chips have fallen asleep, we keep flipping coins, so that this process runs indefinitely.)

Eventually, we'll have $N = L^d$. At that point, the last L^d coins have all come up heads, so all the chips have fallen asleep without any of them moving and the process is over. Let T_1 be the first time when $N = L^d$. Then[7]

$$\mathbb{E}[T_1] = \frac{q^{-L^d}-1}{1-q} \leq (1+\lambda)\left(\frac{\lambda}{1+\lambda}\right)^{-L^d} \leq e^{(L^d+1)/\lambda}.$$

In the last step, $1+x \leq e^x$, so $x/(1+x) = 1/(1+1/x) \geq e^{-1/x}$. This is an upper bound on the expected big steps before sleep.

By Proposition 1.14(ii) in [1], the average return time of the q-lazy random walk on the torus is $1/\pi(x) = L^d$, where we are counting the one-step return that happens when the chip doesn't move at the start. Therefore, each big step takes L^d small steps in expectation, even conditional on the past.

Therefore, as in the proof of Theorem 3.7, the expected number of small steps is L^d times the expected number of big steps, and we get

$$\mathbb{E}[T] \leq L^d\mathbb{E}[T_1] \leq L^d e^{(L^d+1)/\lambda}.$$

Lower bound. By Lemma 4.2, every big step visits at least half the chips in the torus with probability $\geq 2^{-d-5}(1-q) = 2^{-d-5}/(1+\lambda)$.

Let's call a step that visits at least half of the chips an 'alarm.' If we start with $\lceil L/4 \rceil$ chips awake, the probability that they will all go to sleep without an alarm is less than $(1-2^{-d-5}/(1+\lambda))^{L^d/4} \leq \exp(2^{-d-7}L^d/(1+\lambda))$.

[7]Check that $f(n) = (q^{-L^d} - q^{-n})/(1-q)$ satisfies the difference equation $f(n) = qf(n+1) + (1-q)f(0)$ and the boundary condition $f(L^d) = 0$, so the expected number of big steps that it takes for the counter to reach L^d from n is $f(n)$.

If an alarm does occur, there will be at least $\lceil L/2 \rceil$ chips awake afterward, and it will take at least $L^d/4 - 1 \geq L^d/8$ big steps to get back into the original situation. Again, the expected number of small steps is L^d times the expected number of big steps, so:

$$\mathbb{E}[T] \geq \frac{L^{2d}}{8} + (1 - e^{-2^{-d-7}L^d/\lambda})\mathbb{E}[T],$$

or $\mathbb{E}[T] \geq L^{2d} \exp(2^{-d-7}L^d/\lambda)/8$. This is the lower bound. □

4.2. **Bounds for $d = 2$.** This case is almost the same, but the boundary between fast and slow is at $\lambda \sim L^2/\log L$.

Let $G(\mathbf{x})$ be the renormalized lattice Green's function for the discrete Laplacian on $\mathbb{Z}^d$. One way to define it is

$$\begin{aligned} G(\mathbf{x}) &= \lim_{n\to\infty} \mathbb{E}[\text{visits to } \mathbf{x} \text{ by time } n] - \mathbb{E}[\text{visits to } 0 \text{ by time } n] \\ &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{e^{i(x\theta_1 + y\theta_2)} - 1}{1 - \frac{1}{2}\cos\theta_1 - \frac{1}{2}\cos\theta_2} \, d\theta_1 \, d\theta_2. \end{aligned}$$

We use the following properties of that function:

- $G(\mathbf{x})$ is harmonic on $\mathbb{Z}^d \setminus \{0\}$,
- $G(\mathbf{0}) = 0$, and $G(1,0) = G(0,1) = G(-1,0) = G(0,-1) = -1$,
- $G(\mathbf{x}) = -\frac{2}{\pi} \log|\mathbf{x}| + O(1)$, where $|\mathbf{x}| = \sqrt{x_1^2 + x_2^2}$.

See for example Proposition 3.1 in [2], which gives the first asymptotic terms of G. Note that our function is four times their $\mathfrak{G}$.

This function is harmonic away from the origin. Let $\mathbf{X}_0, \ldots$ be a simple random walk on $\mathbb{Z}^2$ started at a neighbour of the origin. Let T be the first time that $\mathbf{X}_n = 0$ or $|\mathbf{X}_n| \geq r$. Then $G(\mathbf{X}_{n\wedge T})$ is a bounded martingale. Let A_r be the event that $|\mathbf{X}_n| \geq r$ before $X_n = 0$. Then

$$-1 = \mathbb{E}[G(\mathbf{X}_0)] = \mathbb{E}[G(\mathbf{X}_T)] = -\frac{2}{\pi}\mathbb{P}[A_r](\log r + O(1)),$$

and we get a sharp estimate of the probability of reaching distance r before hitting zero: $\mathbb{P}[A_r] = (\pi/2)/(\log r + O(1))$, which we will weaken slightly to $c/\log r \leq \mathbb{P}[A_r] \leq C/\log r$ for some constants $c, C > 0$ and $r \geq 2$.

Theorem 4.4. *Let T be the sleep time of ARW on the torus T_2 with one chip on each vertex and sleep rate $\lambda \geq 9L^{3/2}$. There are positive constants c_1, C_1 so that, for large enough L,*

$$\frac{1}{8} \exp\left(\frac{c_1 L^2/\log L}{1+\lambda}\right) \leq \frac{\mathbb{E}[T]}{L^4} \leq 8 \exp\left(\frac{C_1 L^2/\log L}{1+\lambda}\right).$$

Proof. If a simple random walk on $\mathbb{Z}^2$ is started at a neighbour of zero, then it reaches Euclidean distance $L/\sqrt{2}$ before hitting zero with probability at least $c/\log L$ for some constant $c_1 > 0$. If it does, then the corresponding random walk on the torus hits the set of 'farthest' sites, in the sense of Lemma 4.1. By that lemma, if that happens, then the walk has probability at least 2^{-7} of visiting at least half the torus before first hitting zero.

Say a big step that visits at least half of the vertices is an 'alarm,' as in the proof of Lemma 4.3. Then the probability that any single big step is an alarm is at least $2^{-7}c/((1+\lambda)\log L)$. If we start with $\lceil L^2/4 \rceil$ chips awake, the probability that they will all go to sleep without an alarm is less than

$$(1-2^{-7}c/((1+\lambda)\log L))^{L^2/4} \leq \exp(-c_1 L^d/((1+\lambda)\log L)),$$

where $c_1 = 2^{-9}c$. As before, if there is an alarm, then at least half the chips are awake after that step, and it takes at least another $L^d/4 - 1$ big steps until the number of chips awake is back down to $\lceil L^d/4 \rceil$, so by the same calculation as in the proof of Lemma 4.3, we get the lower bound we want:

$$\mathbb{E}[T] \geq \frac{L^4}{8} \exp(c_1 L^2/((1+\lambda)\log L)).$$

The upper bound for the time is similar. In this case, we will give names to two special kinds of step.

- If the coin comes up tails, and the walk reaches Euclidean distance $\sqrt{L}$ from the origin, that's a 'major step.'
- If the coin comes up tails but the walk doesn't reach that far, it's a 'minor step.'

Let p be the chance that a big step is major. Then $p = (1-q)\mathbb{P}[A_{\sqrt{L}}]$, which is at most $2C/(1+\lambda)\log L$, by the discussion before this theorem.

Suppose $p \leq 1/2$, which is true for all sufficiently large L. Then we can use the inequality $1-p \geq e^{-2p}$, so if we take L^2 big steps, the probability that there is no major step is $(1-p)^{L^d} \geq \exp(-4CL^2/((1+\lambda)\log L))$.

If there are no major steps in the first L^2 steps, then we have probability at least $1/4$ of falling asleep before $2L^2$ big steps have happened. To see this, we analyze the number of minor steps.

Let X be the number of minor steps in the first L^2 steps. Then $X \sim \mathrm{Bin}(L^2, 1-p-q)$. By Chebyshev's inequality, the probability that this is less than $3L^2(1-q)$ is at least $1/2$.[8] Each minor step stays within distance $\sqrt{L}$ of the origin, so it visits at most $9L \geq (2\sqrt{L}+1)^2$ vertices. Therefore, with probability at least $1/2$, there are at most $27L^3(1-q)$ chips awake after

[8] The mean of this random variable is $\mu = L^2(1-p-q) \leq L^2(1-q)$, and the variance is $L^2(1-p-q)(p+q) \leq L^2(1-q)$, so $\mathbb{E}[X - L^2(1-q) \geq 2L^2(1-q)] \leq 1/4$. Actually, the median of a binomial is between $\lfloor \mu \rfloor$ and $\lceil \mu \rceil$, but we don't need that much sharpness.

the first L^2 big steps, and they will all fall asleep immediately afterward with probability $(1-q)^{27L^3(1-q)}$, which is at least $1/2$, because $\lambda \geq 9L^{3/2}$:

$$(1-q)^{27L^3(1-q)} \geq e^{-27L^3/\lambda(1+\lambda)} \geq e^{-1/3} \geq \frac{1}{2}.$$

We've used the fact that $1-q \geq e^{-1/\lambda}$ from the proof of the last theorem. So, the probability that all the chips fall asleep in $2L^2$ big steps is at least

$$\frac{1}{4}(1-p)^{L^d} \geq \frac{1}{4}\exp(-4CL^2/((1+\lambda)\log L).$$

As before, if we fail, we try again, and the expected number of big steps until we succeed is at most $(2L^2)(4\exp(4CL^2/((1+\lambda)\log L))$.

The expected number of small steps is L^2 times larger, again by the reasoning in the proof of Lemma 3.7. Therefore,

$$\mathbb{E}[T] \leq 8L^4\exp(4CL^2/((1+\lambda)\log L)),$$

which is the upper bound we want, with $C_1 = 4C$. □

Because each big step may involve a q-lazy random walk on the torus, the tail probabilities of the sleep time of the higher-dimensional process also decay on the longer scale $\mathbb{E}[T]/(1-q)$, just as we saw in Theorem 3.10.

4.3. **The proof of the half-visited lemma.** We're going to prove that, if $\mathbf{x}, \mathbf{x}'$ are two vertices in the torus and $d(\mathbf{x},\mathbf{x})' = \max\{d(x_i, x_i')\} = \lfloor L/2 \rfloor$, then a simple random walk started at $\mathbf{x}$ has a constant probability of visiting more than half the torus when it hits $\mathbf{x}'$, including $\mathbf{x}$ and $\mathbf{x}'$.

We need two more lemmata first. Let $p(\mathbf{x})$ be the chance that a walk started at $\mathbf{x}$ visits at least $L^d/2$ vertices when it hits $\mathbf{0}$, including $\mathbf{x}$ and $\mathbf{x}'$.

Lemma 4.5. *The average value of p is at least* $1/2$.

Proof. By translation invariance and symmetry, $p(\mathbf{x})$ is equal to the chance that a walk from $\mathbf{0}$ has visited at least $L^d/2$ vertices before it reaches $\mathbf{x}$.

Let's start a simple random walk at $\mathbf{0}$, and run it forever. It almost certainly visits every site, and the first visits are in some order. Let $P(\mathbf{x};n)$ be the probability that the n-th site in the order is $\mathbf{x}$. Then $\sum_{\mathbf{x}} P(\mathbf{x};n) = 1$, so

$$\sum_{\mathbf{x}\in T_d} p(\mathbf{x}) = \sum_{\mathbf{x}\in T_d}\sum_{n\geq L/2}^{L} P(\mathbf{x};n) = \sum_{n\geq L/2}^{L} 1 = L - \lceil L/2 \rceil + 1 \geq L/2.$$

That means that the average of p is $L^{-d}\sum p(\mathbf{x}) \geq 1/2$. □

We also want to know something about the average of p on smaller sets. Let S_t be the set of vertices in the torus with first coordinate $t \in \mathbb{Z}/L\mathbb{Z}$, and let the average of p on S_t be denoted by $s(t) := L^{-d+1}\sum_{\mathbf{x}\in S_t} p(\mathbf{x})$.

The torus is symmetric under reflection, so $s(t) = s(-t) = s(L-t)$. The next lemma proves that s is monotone on the set $\{0, 1, \dots, \lfloor L/2 \rfloor\}$, and it's bounded below by $1/4$ as long as t is far enough away from zero.

Lemma 4.6. *$s(0) \le s(1) \le \cdots \le s(\lfloor L/2 \rfloor)$, and $s(\lceil L/3 \rceil) \ge 1/4$.*

Proof. The walk on the torus is nearest-neighbour, so if a walk starts at a point on S_t with $0 < t \le L/2$, it must hit either S_{t-1} or $S_{L-t+1} = S_{-t+1}$ before reaching the origin.

Choose a point uniformly from S_t, say $\mathbf{x}$. Let it walk until it hits one of the slabs S_{t-1} or S_{L-t+1} at a point $\mathbf{y}$, and then continue until it hits $\mathbf{0}$. The walk starting at $\mathbf{x}$ is a subset of the walk starting at $\mathbf{y}$, so $\mathbb{E}[p(\mathbf{x})] \ge \mathbb{E}[p(\mathbf{y})]$.

By translation invariance, the last $d-1$ coordinates of $\mathbf{y}$ are distributed uniformly in $\mathbb{Z}^{d-1}/L\mathbb{Z}^{d-1}$. The first coordinate is either $t-1$ or $L-t+1$, and the value of p is the same at $(y_1, y_2, \dots, y_d)$ and $(L-y_1, y_2, \dots, y_d)$ by reflection symmetry, so $\mathbb{E}[p(\mathbf{y})] = s(t-1)$. Therefore, $s(t) \ge s(t-1)$ for $0 < t \le L/2$. This gives us the monotonicity claimed in the statement.

Let r be the value in the statement of the theorem, $r := s(\lceil L/3 \rceil)$. By monotonicity and symmetry, $s(t) \le r$ when $t \in \{-\lceil L/3 \rceil, \dots, \lceil L/3 \rceil\}$. That's at least two-thirds of $\mathbb{Z}/L\mathbb{Z}$, and $s(t) \le 1$ for the other values, so

$$\frac{1}{2} \le \frac{s(0) + \cdots + s(L-1)}{L} \le \frac{2r}{3} + \frac{1}{3}.$$

Therefore, r must be at least $1/4$, which proves the lemma. □

Now we can prove the half-visited lemma. We recall the statement:

Lemma 4.1 (Half-visited lemma)**.** *If $\mathbf{x}$ is as far as possible from $\mathbf{x}'$, then with probability at least 2^{-d-4}, a simple random walk from $\mathbf{x}$ and stopped at $\mathbf{x}'$ visits at least half the vertices of the torus.*

Proof. The torus is transitive and symmetric under reflection and permutation of coordinates, so it's enough to prove the statement of the lemma when $\mathbf{x}' = 0$ and the first coordinate of $\mathbf{x}$ is $\lfloor L/2 \rfloor$.

Let $\mathbf{X}_t, \mathbf{Y}_t$ be two random walks on T_d coupled as follows. Each coordinate $i \in \{1, \dots, d\}$ has an exponential clock going at rate 2. When the clock for coordinate i rings:

- If $(X_i)_t \not\equiv (Y_i)_t$, then one chosen uniformly at random stays the same and the other changes by ± 1 uniformly.
- If $(X_i)_t \equiv (Y_i)_t$, then the walks stay together. With probability $\frac{1}{2}$, they both change by ± 1, and with probability $\frac{1}{2}$, they stay the same.

Each of the walks by itself looks like a simple random walk in continuous time with rate 1. When the walks start, $\mathbf{X}_0 = \mathbf{x}$ and $\mathbf{Y}_0$ is uniform on the farthest slab $S_{\lfloor L/2 \rfloor}$.

Let T_1 be the time that it takes for the walk on the first coordinate to hit zero. It's a continuous random variable with nonzero density on $(0,\infty)$, so we can define $t_{1/2}$ to be the unique time when $\mathbb{P}[T_1 \leq t] = 1/2$.

If $i = 2,\dots,d$, then at some point the two walks X_i and Y_i will collide with each other and lock together. Let $Z_i := X_i - Y_i \bmod L$. This is a simple random walk on the cycle in continuous time, started at a uniform vertex on the cycle, which runs twice as fast as the walk on the first coordinate.

Let $T_2,\dots,T_d$ be the times when $Z_2,\dots,Z_d$ first hit zero. These walks run faster and their starting distribution is closer to the origin, so $\mathbb{P}[T_i \leq t_{1/2}] \geq 1/2$ for each coordinate. All the walks are independent, so

$$\mathbb{P}[T_2,\dots,T_d \leq t_{1/2} \leq T_1] = \mathbb{P}[T_1 > t_{1/2}] \prod_{i=2}^{d} \mathbb{P}[T_i \leq t_{1/2}] \geq 2^{-d}.$$

The walk on the first coordinate starts at $\lfloor L/2 \rfloor$, and at any fixed time t, the probability that $(X_1)_t = x$ is largest when $x = \lfloor L/2 \rfloor$ and decreases as x gets farther away. Let A be the event that $X_1 \in \{\lceil L/3 \rceil,\dots,L - \lceil L/3 \rceil\}$. Positions near the starting point are more likely, so $\mathbb{P}[A] \geq 1/4$.

On the other hand, by the Markov property, once the walk hits zero, the situation is reversed: $W_t := (X_1)_{t+T_1}$ behaves like a random walk starting at zero, and $\mathbb{P}[W_t = x]$ is larger near zero. Let $B := \{T_1 \leq t_{1/2}\}$. Conditional on that event, W and T_1 are independent from each other, so

$$\begin{aligned}
\mathbb{P}[A \mid B] &= \mathbb{P}[(X_1)_{t_{1/2}} \in \{\lfloor L/2 \rfloor - m, \lfloor L/2 \rfloor + m\} \mid B] \\
&= \mathbb{P}[W_{t_{1/2} - T_1} \in \{\lfloor L/2 \rfloor - m, \lfloor L/2 \rfloor + m\} \mid B] \\
&\leq (2m+1)/L.
\end{aligned}$$

However, $\mathbb{P}[A]$ is the weighted average of $\mathbb{P}[A \mid B]$ and $\mathbb{P}[A \mid B^c]$, so $\mathbb{P}[A \mid B^c]$ is at least $(2m+1)/L$, and by our choice of m, that's at least $1/4$. Therefore,

$$\begin{aligned}
\mathbb{P}[A \cap B^c, \text{ and } T_2,\dots,T_d \leq t_{1/2}] &= \mathbb{P}[A \mid B^c]\, \mathbb{P}[T_2,\dots,T_d \leq t_{1/2} \leq T_1] \\
&\leq 2^{-d-2}.
\end{aligned}$$

If this event occurs, then we haven't hit zero by time $t_{1/2}$, and all the coordinates have locked, so $\mathbf{X}_{t_{1/2}} = \mathbf{Y}_{t_{1/2}}$. What's more, the first coordinate of the walks is between $\lceil L/3 \rceil$ and $L - \lceil L/3 \rceil$, so $s((Y_1)_{t_{1/2}}) \geq 1/4$.

The joint distribution of $(Y_2,\dots,Y_d)_t$ is uniform on $\mathbb{Z}^{d-1}/L\mathbb{Z}^{d-1}$ at any time t, so in particular it's uniform at $t_{1/2}$. Therefore,

$$\begin{aligned}
p(\mathbf{x}) &\geq \mathbb{E}[p(\mathbf{Y}_{t_{1/2}});\ A \text{ and } B^c \text{ both occur, and } T_2,\dots,T_d \leq t_{1/2}] \\
&\geq 2^{-d-2} \min\{s(\lceil L/3 \rceil),\dots,s(L - \lceil L/3 \rceil)\} \\
&\geq 2^{-d-4}.
\end{aligned}$$

This is the lower bound on p that we wanted. □

REFERENCES

[1] D.A. Levin, Y. Peres, and E.L. Wilmer. *Markov Chains and Mixing Times*. American Mathematical Soc.

[2] Per-Gunnar Martinsson and Gregory J. Rodin. Asymptotic expansions of lattice green's functions. *Proceedings: Mathematical, Physical and Engineering Sciences*, 458(2027):2609–2622, 2002.

[3] G. N. WATSON. THREE TRIPLE INTEGRALS. *The Quarterly Journal of Mathematics*, os-10(1):266–276, 01 1939.