Sequential Monte Carlo and Self Avoiding Walks

Nils Hallerfelt

February 2023

Introduction

In this report we will analyse self-avoiding walks (SAWs) in $\mathbb{Z}^d$. A SAW is a sequence of moves in a lattice that does not visit the same point more than once. In $\mathbb{Z}^d$, the set $\mathbf{S}_n(d) \in \mathbb{Z}^d$ will denote the set of all possible SAWs of length n that exists in $\mathbb{Z}^d$ and starts in the origin. To be more specific:

$$\mathbf{S}_n(d) = \{x : 0 : n \in \mathbb{Z}^{d(n+1)} : x_0 = \mathbf{0}, |x_k - x_{k-1}| = 1, x_\ell \neq_k, \forall 0 \leq \ell < k \leq n\} \tag{1}$$

The main objective of this report is to compute the cardinality of such sets for large n. We will usually denote this, the number of elements (SAWs) in such a set, as $c_n(d) = |\mathbf{S}_n(d)|$. However will sometimes use the $|\mathbf{S}_n(d)|$ notation when it is more illustrative.

Some Theory on SAWs

We here make an argument for why $c_{n+m}(d) \leq c_n(d) \cdot c_m(d)$. The set S_{n+m} has $|S_{n+m}|$ walks, each of which can be partitioned into walks of length n and of walks of length m. We can thus trivially construct $|S_{n+m}|$ ordered pairs from this. Furthermore, there is a total number of $|S_n \times S_m| = |S_n| \cdot |S_m|$ possible pairs. Of course, there exists a less or equal amount of ordered pairs as there are pairs. So:

$$|S_{n+m}(d)| \leq |S_n(d)| \cdot |S_m(d)| \tag{2}$$

We can now use the property expressed in eq 2 to prove that the limit $\mu_d = \lim_{n\to\infty} c_n^{1/n}(d)$ exists:

$$c_{n+m} \leq c_n c_m \implies \log c_{n+m} \leq \log c_n + \log c_m \tag{3}$$

So, the sequence $\log(c_{n+m})$ is subadditive, and we can apply Fekete's lemma.

$$\lim_{n\to\infty} \frac{\log c_n}{n} = \lim_{n\to\infty} \log c_n^{1/n} = \log \lim_{n\to\infty} c_n^{1/n} \tag{4}$$

As Fekete's lemma states that the leftmost expression exists, $\mu_d = \lim_{n\to\infty} c_n(d)^{1/n}$ must be positive and exist. This constant μ_d is called the connective constant.

Proposition 1. *The connectivity constant μ_d is positive and exists.*

Furthermore, it is conjectured that:

$$c_n(d) \sim \begin{cases} A_d \mu_d^n n^{\gamma_d - 1} & d = 1, 2, 3, d \geq 5 \\ A_d \mu_d^n \log(n)^{1/4} & d = 4 \end{cases} \quad \text{as } n \to \infty. \tag{5}$$

Here, some of the values of γ are known: $\gamma_1 = 1$, $\gamma_2 = 43/32$, $\gamma_d = 1$ for $d \geq 5$. We will use 5 to estimate the connectivity constant, using its relation to $c_n(d)$. We therefore now turn our focus on estimating $c_n(d)$ and how it depends on n.

We can cast this problem of calculating $c_n(d)$ to a stochastic one by representing the distribution of a SAW among random walks be the uniform distribution over $\mathbf{S}_n(d)$:

$$f(x_{0:n}; d) = \frac{\mathbf{1}_{\mathbf{S}_n(d)}(x_{0:n})}{c_n(d)} \tag{6}$$

where $\mathbf{1}_{\mathbf{S}_n(d)}$ is the indicator function. Thus c_n is the normalizing constant of the distribution f_n and we can use the framework of self-normalized importance sampling to get an estimate of $c_n(d)$:

$$c_n(d) = \mathbb{E}_{g_n}[\omega_n(X_{0:n})] \approx \frac{1}{N} \sum_{\ell}^{N} \omega_n(X_{0:n}^{\ell}) \tag{7}$$

A direct approach would be to use standard Monte Carlo importance sampling. However, this would have to be done on increasing dimensions as n increases, and it would overlook the key sequential nature of the problem. As n increases f_n (and c_n) evolves. Also, finding good envelopes g_n could be difficult. It would be better use previous information in the process, a sequential Monte Carlo strategy.

Some Theory on Sequential Importance Sampling

As we are interested in sequences $(c_n(d))_{n \geq 1}$ we adopt the sequential importance sampling technique. This way the high-dimensional problem is split into a sequence of simpler ones. Assume we have generated a particle $X_i^{0:n} = (X_i^0, ..., X_i^n)$ and calculated its weight $\omega_n^i = \omega_n(X_i^{0:n})$ using some standard importance sampling. The next generated sample $X_i^{0:n+1}$ and its weight ω_{n+1}^i should be easily generated without increasing complexity as n increases. This can be done using an envelope g satisfying the following:

$$g_{n+1}(x_{0:n+1}) = g_{n+1}(x_{n+1}|x_{0:n}) g_{n+1}(x_{0:n}) = g_{n+1}(x_{n+1}|x_{0:n}) g_n(x_{0:n}) \tag{8}$$

The conditional method can then be used, drawing one new sample X_i^{n+1} from $g_{n+1}(x_{n+1}|x_{0:n} = X_i^{0:n})$ and appending it to the already existing particle $X_i^{0:n}$ yielding $X_i^{0:n+1} = (X_i^{0:n}, X_i^{n+1})$. This allows us to sequentially update the weights:

$$\omega_{n+1}^i = \frac{z_{n+1}(X_i^{0:n+1})}{g_{n+1}(X_i^{0:n+1})} \tag{9}$$

$$= \frac{z_{n+1}(X_i^{0:n+1})}{z_n(X_i^{0:n}) g_{n+1}(X_i^{n+1}|X_i^{0:n})} \cdot \frac{z_n(X_i^{0:n})}{g_n(X_i^{0:n})} \tag{10}$$

$$= \frac{z_{n+1}(X_i^{0:n+1})}{z_n(X_i^{0:n}) g_{n+1}(X_i^{n+1}|X_i^{0:n})} \cdot \omega_n^i \tag{11}$$

where we in eq 10 have used the property of g described in eq 8 and multiplied both nominator and denominator with $z_n(X_i^{0:n})$.

Estimating $(c_n(2))_{n\geq 1}$

We now turn to a more practical point of view, using different techniques based on sequential importance sampling to estimate the $(c_n(2))_{n\geq 1}$ sequence. First we present two pure sequential importance sampling techniques with different envelopes, before using the more advanced technique *sequential importance sampling with resampling*.

In all the following examples $N = 1000$ (number of particles), $n_{max} = 499$ (the largest length), and $d = 2$ (dimension of $\mathbb{Z}$)

A Naive Approach

In a first naive approach we use SIS with instrumental distribution g_n being that of a random walk $(X_k)_0^n \in \mathbb{Z}^2$, where $X_0 = \mathbf{0}$ and each X_{k+1} is drawn uniformly among the four neighbours of X_k. We can show that this is the same as simulating a large number N of random walks in $\mathbb{Z}^2$, counting the number N_{SA} of self-avoiding ones, and estimate $c_n(d)$ using the observed ratio N_{SA}/N. From eq 7 we know that c_n can be estimated as the mean of the weights. Furthermore, we can write the weight for a particle the following way, using that g_n is uniform and that any point in $\mathbb{Z}^2$ have four neighbours:

$$\omega_n(X_i^{0:n}) = \frac{z_n(X_i^{0:n})}{g_n(X_i^{0:n})} = \frac{\mathbf{1}_{\mathbf{S}_n(d)}(X_i^{0:n})}{g_n(X_i^{0:n})} = \mathbf{1}_{\mathbf{S}_n(d)}(X_i^{0:n}) \cdot 4^n. \tag{12}$$

We plug this into eq 7:

$$c_n = \frac{4^n}{N} \sum_{i=1}^{N} \mathbf{1}_{\mathbf{S}_n(d)}(X_i^{0:n}) \tag{13}$$

The sum expression is equivalent with counting the number of self-avoiding walks among N random walks. So eq 13 can be simplified into:

$$c_n = \frac{N_{SA}}{N} \cdot 4^n \tag{14}$$

As there are 4^n random walks of length in in $\mathbb{Z}^2$, this is equivalent to estimating the ratio N_{SA}/N and multiply it with the total number of random walks. So in order to estimate c_n we simply run N random walks and count the number of self-avoiding ones and then multiply by 4^n. The results can be seen in table Figure (1) and Table (1).

n	1	2	3	4	5	6	7	8	9	...	21	22
$c_n(2)$	4	11.87	36.99	103.42	291.84	819.2	2.34e+3	6.49e+3	1.91e+4	...	8.80e+9	0

Table 1: Estimated number of SAW:s ($c_n(2)$) using the naive SIS. The table shows in the top row the length of the SAW:s, and in the second row the Estimated number of SAW:s ($c_n(2)$). We can see that after $n = 21$ the number of SAW:s are 0, which of course is not a realistic result. We only provide the first values and when it goes to zero for illustrative purposes.

We notice that the number of self avoiding walks first increases as n increases, an intuitive behaviour. However, after a relatively small amount of steps it suddenly reads to 0. This is of course not true. This is a problem inherited from weight degeneration. As the weights are updated at each step of the sequence, it is increasingly likely that the majority of the total weight will be concentrated to only a few particles, as the weights are calculated sequentially (se eq 11) by multiplication. In our case, this means that as soon a weight is calculated to be zero (the indicator function returns 0), all subsequent weights will be zero. The probability of a weight

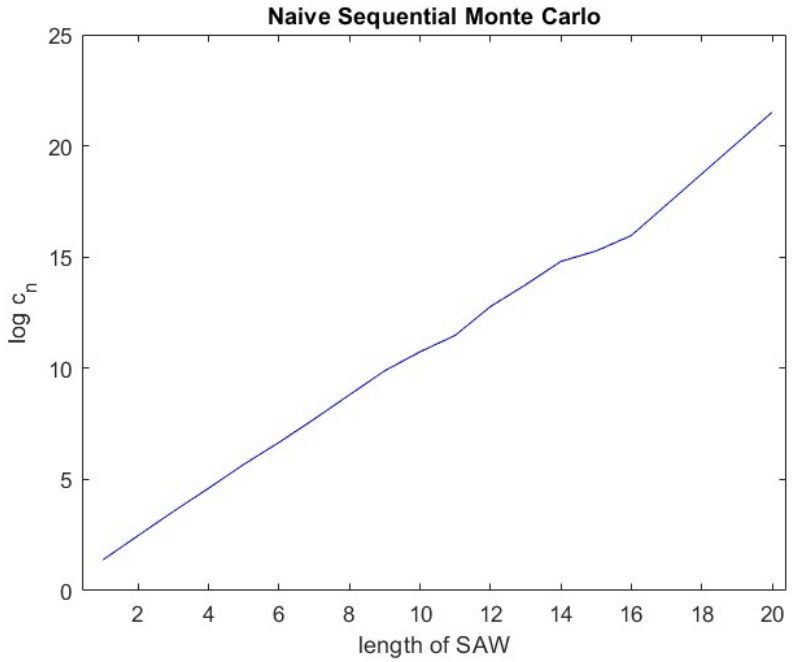

Figure 1: The logarithm of estimated number of SAW:s ($c_n(2)$) on the y-axis, using the naive SIS, plotted against the length of the SAW:s (n). Although the simulation was made up to $n = 499$, it was zero already after 21. Its plotted in logarithmic scale to show how far it goes before getting to zero, as the logarithm then goes undefined, and for later plots this is hard to see without the log scale.

subsequence to be set to zero increases as n gets larger as its more probable that a longer random walk "collides" with itself. So this whole sequence is killed from this step and forward. After a number of steps all particles can have been killed.

Now, we do not use the weights to calculate c_n directly here, however the equivalence between eq 13 and eq 14 ratio shows that the ratio N_{SA}/N will eventually be zero as n increases.

An Improved Approach

We now improve on the naive approach by picking another envelope g_n. Now g_n will be the distribution of a self-avoiding random walk $(X_k)_{k=0}^n$ in $\mathbb{Z}^2$ starting in the origin. This means that

1. $X_0 = \mathbf{0}$ and

2. given $X_{0:k}$, the next point X_{k+1} is drawn uniformly among the free neighbours $\mathbf{N}(X_{0:k})$ of $X_{0:k}$. If $\mathbf{N}(X_{0:k}) = \emptyset$, then X_{k+1} is set to X_k.

Intuitively this envelope should somewhat decrease the problem of weight degeneracy, as the sampling chooses not to collide with itself if possible, thus making a collision less likely to happen, and the weights are less likely to be set to zero as the indicator function will less often be zero. Also note that c_n must now be estimated using eq 7.

Using this envelope we ran the SIS algorithm with 10^3 particles. The results can be in figure (2) and Table (2).

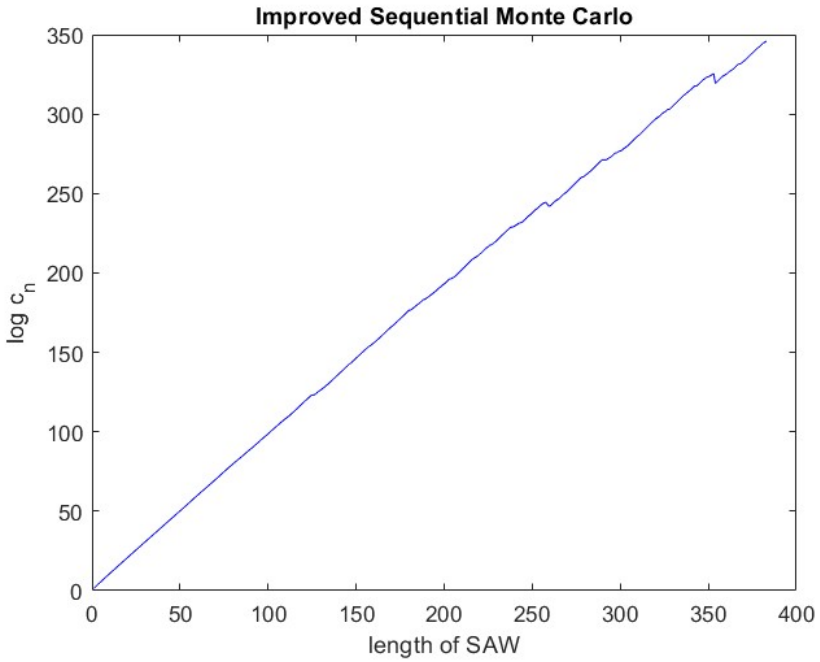

Figure 2: The logarithm of estimated number of SAW:s ($c_n(2)$) on the y-axis, using the improved envelope, plotted against the length of the SAW:s (n). Although the simulation was made up to $n = 499$, it was zero after $n = 384$. Its plotted in logarithmic scale to show how far it goes before getting to zero, as the logarithm then goes undefined (implying zero).

n	1	2	3	4	5	6	7	...	382	383	384
$c_n(2)$	4	12	36	99.47	282.53	781.06	2.18e+3	...	8.92e+149	1.78e+150	0

Table 2: Estimated number of SAW:s ($c_n(2)$) using SIS with the improved envelope. The table shows in the top row the length of the SAW:s, and in the second row the Estimated number of SAW:s ($c_n(2)$). We can see that after $n = 384$ the number of SAW:s are 0, which of course is not a realistic result. We only provide the first values and when it goes to zero for illustrative purposes.

We can see that c_n increases for longer now before its set to zero, an substantial improvement on the naive approach and on the problem with weight degeneracy.

Now a technique called sequential importance sampling with resampling will be used.

Sequential Importance Sampling With Resampling

Weight degeneration is a universal problem with the SIS method (Lecture 6, slide 27). This problem can be tackled by using resampling in the SIS method, resulting in the *sequential importance sampling with resampling* (SISR) method. This method is based on the idea (Gordon et al, 1993) of duplicating particles with large weights and killing particles with small weights. A natural question is how to select which particles to duplicate and which to kill. One way, and the way we will focus on, is to do multinomial resampling. With multinomial sampling we draw, with replacement, new particles $\tilde{X}_1^{0:n}, \ldots, \tilde{X}_N^{0:n}$ particles from the SIS produced particles

$X_1^{0:n}, \ldots, X_N^{0:n}$ with probabilities given by the normalized importance weights (Lecture 7, slide 19). So, for all $i = 1, \ldots, N$

$$\mathbb{P}\big(X_i^{0:n} = X_j^{0:n}\big) = \frac{\omega_n^j}{\sum_\ell \omega_n^\ell}.$$

This essentially means that particles with larger wights at step n have a larger probability of surviving the resampling and being duplicated, then those particles with low weights that then are more likely to be killed. An illustrative figure shows the resampling process, taken from Given's and Hoeting's "Computational Statistics, second edition" (page 173). See Figure (3).

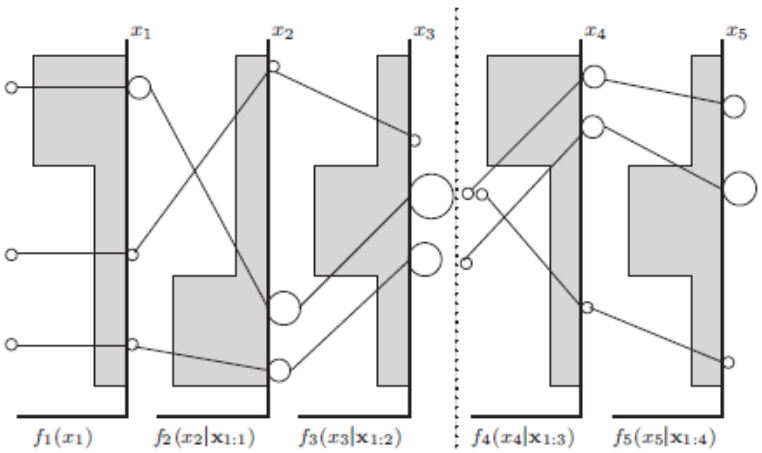

FIGURE 6.8 Illustration of sequential importance sampling. Time progresses from left to right. Samples are indicated by solid lines. Densities are shaded boxes. Points are shown with circles; the area of the circle is an indication of the sequence's weight at that time. The dashed line indicates an instance of weight regeneration through sequence resampling. See the text for a detailed discussion.

Figure 3: An illustrative figure describing the resampling process, credits to Given's and Hoeting's "Computational Statistics, second edition" (page 173). In the figure the resampling is done after the degeneration has reached a certain threshold, whereas we will instead do the resampling at each step/iteration.

After the selection we have a new group of particles $\tilde{X}_1^{0:n}, \ldots, \tilde{X}_N^{0:n}$. Each of these resampled particles are then assigned equal weights, $\tilde{\omega}_n^i = c$. We now proceed with regular SIS, setting for each i, $X_i^{0:n+1} = (\tilde{X}_i^{0:n}, X_{in+1}$, where $X_{in+1} \sim g_{n+1}(x_{n+1}|\tilde{X}_i^{0:n})$. The weights are updated in accordance with eq 10 with slight modification:

$$\frac{z_{n+1}(X_i^{0:n+1})}{z_n(\tilde{X}_i^{0:n})g_{n+1}(X_i^{n+1}|\tilde{X}_i^{0:n})} \cdot \frac{z_n(\tilde{X}_i^{0:n})}{g_n(\tilde{X}_i^{0:n})} = \frac{z_{n+1}(X_i^{0:n+1})}{z_n(\tilde{X}_i^{0:n})g_{n+1}(X_i^{n+1}|\tilde{X}_i^{0:n})} \cdot \tilde{\omega}_n^i \tag{15}$$

As $\tilde{\omega}_n^i$ is equal and arbitrary for every i, and the use of normalized importance weights in the selection procedure, it might as well be set to 1.

Running the SISR method on our problem produced the following results for $c_n(2)$ and can be seen in Figure (4) and table Table (3).

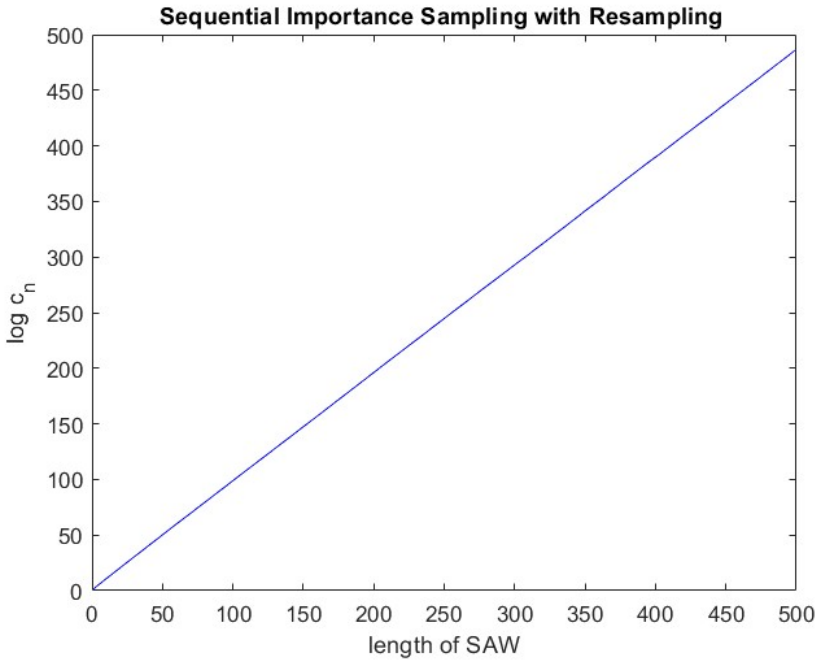

Figure 4: The logarithm of estimated number of SAW:s ($c_n(2)$) on the y-axis, using SISR with the previously used improved envelope, plotted against the length of the SAW:s (n). We note that the c_n never goes to zero for $n = 1, \ldots, 499$ now that the SISR method is used. Also note the linearity when using the log-scale, which will be used later on.

n	1	2	3	4	5	6	7	...	498	499
$c_n(2)$	4	12	36	100.40	285.35	779.29	2.18e+3	...	3.78e+210	9.97e+210

Table 3: Estimated number of SAW:s ($c_n(2)$) using SIS with the improved envelope. The table shows in the top row the length of the SAW:s, and in the second row the Estimated number of SAW:s ($c_n(2)$). We can see that after $n = 384$ the number of SAW:s are 0, which of course is not a realistic result. We only provide the first values and when it goes to zero for illustrative purposes.

Weight Degeneracy and Effective Sample Size - Experiment

To identify and measure weight degeneracy one can use the *effective sample size*. Degeneracy of the weights is related to their variability - as the weights on many particles goes to zero the remaining weights are concentrated to a few particles with increased variability as a consequence. A useful measure of variability in the weights is the squared coefficient of variation given by (Given, and Hoeting, "Computational Statistics, second edition", (page 172)) :

$$cv^2(\omega(X)) = \mathbf{E}(N\omega(X) - 1)^2. \tag{16}$$

If we suppose that N samples have normalized importance weights $\omega(x^i)$ for $i = 1, \ldots, N$ and that z of these weights are equal to zero. Then the then the estimation effectively relies on the $(n - z)$ particles with non-zero weights. Givens present the following equation for estimating the squared coefficient of variation:

$$\hat{cv}^2(\omega(X)) = \frac{1}{N}\sum_{i=1}^{N}(N\omega((x^i - 1)^2 = \frac{N}{N - z} - 1 \tag{17}$$

Using this they deduce the following measure of the effective sample size using standardized weights:

$$\hat{N}(g,f) = \frac{1}{\sum_{i=1}^{N} \omega(x^i)} \tag{18}$$

We can sequentially measure the effective samples size of our methods to see how the degeneracy evolves, where low values of effective sample size implies strong degeneracy in the weights. This has been done for the naive SIS, improved SIS, and SISR. The result can be seen in Figure (5), where $n_{max} = 499$ and 1000 particles are used (the same as for the prevously presented results):

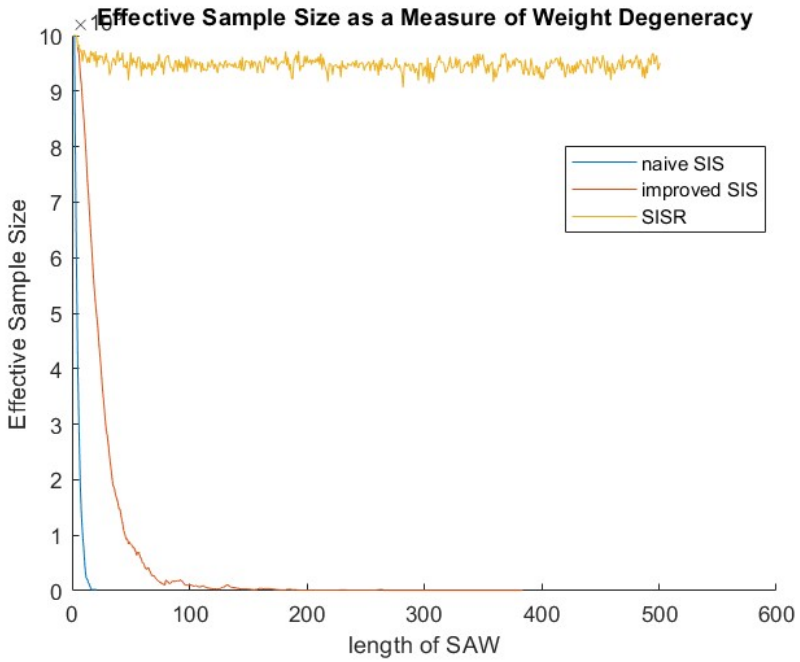

Figure 5: On the y-axis is the effective sample size, plotted against the length of the SAW:s. We can see the drastic improvement in effective sample of SISR compared to both SIS methods.

As we can see, the effective sample size decreases fast for the naive approach, decreases more slowly for the improved version but still approaching zero, and stays approximately constant for the SISR method. This shows the great advantage in using SISR as n grows large.

The Conjectured Relation

We will now use our estimates of $(c_n(2))_{n\geq 1}^{n_{max}}$ provided by the SISR method to obtain estimates of the parameters $A_2, \mu_d,$ and , γ_2 via the conjectured relation given by 5. This can be done by taking the logarithm of the relation in 5 for $d = 2$, yielding:

$$\log c_n(2) = \log A_2 + n \log \mu_2 + \gamma_2 \log n - \log n \tag{19}$$

$$\log(nc_n(2)) = \log A_2 + n \log \mu_2 + \gamma_2 \log n \tag{20}$$

$$\log(nc_n(2)) = \begin{bmatrix} 1 & n & \log n \end{bmatrix} \begin{bmatrix} \log A_2 \\ \log \mu_2 \\ \gamma_2 \end{bmatrix}. \tag{21}$$

We can identify a linear regression in these transformed parameters and use this to estimate $\log A_2, \log \mu_d,$ and $, \gamma_2$. This results in solving the following linear system of equations, where now $n_{max} = 199$:

$$\begin{pmatrix} 1 & 1 & \log 1 \\ 1 & 2 & \log 2 \\ & \vdots & \\ 1 & n_{max} & \ln n_{max} \end{pmatrix} \begin{pmatrix} \ln A_2 \\ \log \mu_2 \\ \gamma_2 \end{pmatrix} = \begin{pmatrix} \ln c_1(2) \\ \log\left(2c_2(2)\right) \\ \vdots \\ \log\left(100 c_{n_{max}}(2)\right) \end{pmatrix} \tag{22}$$

Solving this results in the parameters $\log A_2$, $\log \mu_2$ and γ_2, and this is done by multiplying both sides with the transpose of the leftmost matrix, and then finding its inverse (or pseudoinverse, depending on situation). We can then take the exponential of $\log A_2$, $\log \mu_2$ to get the original parameters. This has been done for 10 independent estimates of $c_n(2)$, and the result can be seen in figure 6.

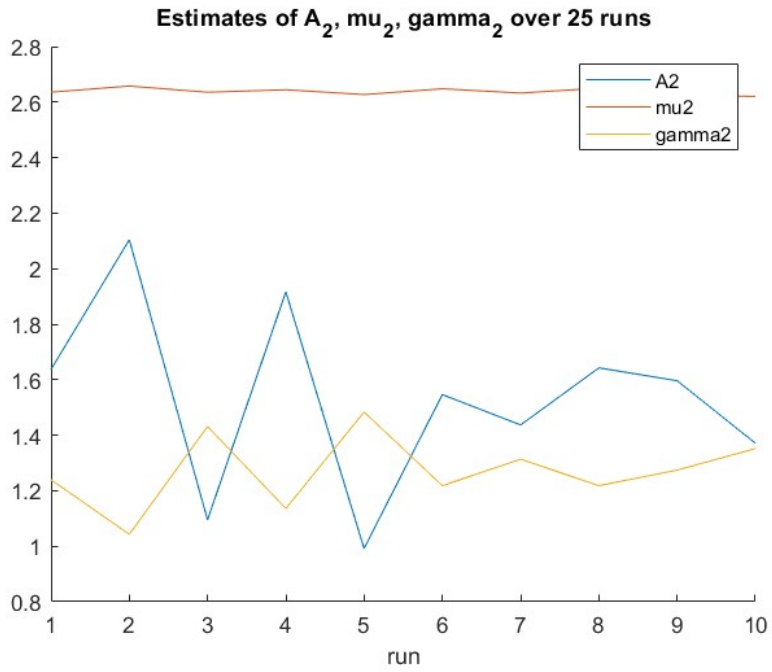

Figure 6: Estimates of A_2, μ_2, and γ_2 over 10 runs. Each run is with an iid SISR up to $n = 199$ with 1000 particles.

We now investigate which parameter is more easily estimated of the three. Lets rewrite eq 21 as:

$$\log(c_n(2)) = \begin{bmatrix} 1 & n & \log n \end{bmatrix} \begin{bmatrix} \log A_2 \\ \log \mu_2 \\ \gamma_2 \end{bmatrix} - \log(n) \tag{23}$$

If the coefficients have significantly different magnitudes, it means that one parameter has a stronger influence on $\log(c_n(2))$ than the others. In that case, it would be easier to predict the parameter with the stronger influence because it would have a clearer linear relationship with $\log(c_n(2))$. We can calculate the expected value among our 10 independent estimates, and get:

1. $\mathbf{E}[A_2] = 1.3298$
2. $\mathbf{E}[\mu_2] = 2.6385$

3. $\mathbf{E}[\gamma_2] = 1.3350$.

This implies that μ_2 is easier to estimate. We can also look at the variance, where a low variance is preferable.

1. $\mathbf{V}[A_2] = 0.0673$
2. $\mathbf{V}[\mu_2] = 5.7961e-05$
3. $\mathbf{V}[\gamma_2] = 0.0160$.

This makes us draw the conclusion that mu_2 is easier to estimate.

Parameter Bounds for general d

In this section we are able to deduce a few general bounds on the parameters.

The bound $d \leq \mu_d \leq 2d - 1$

We will here show that the connectivity constant is bounded by $d \leq \mu_d \leq 2d - 1$. We start with the upper bound. We can do this by finding an upper bound on the number of self-avoiding walks $c_n(d)$ and then use the definition of μ_d: $\mu_d = \lim_{n\to\infty} c_n(d)^{1/n}$.

An upper bound on $c_n(d)$ can be established with the following reasoning. Starting at the origin, there are are $2d$ neighbours, none of which have been visited. So there are $2d$ possible self avoiding walks of length $n = 1$. From now on, each lattice point reached by a self-avoiding walk (for any walk for that matter) have at most $2d - 1$ neighbours that have not been visited. So:

$$c_n(d) \leq 2d(2d-1)^{n-1} \tag{24}$$

Using the definition of μ_d:

$$\mu_d = \lim_{n\to\infty} c_n(d)^{1/n} \leq \lim_{n\to\infty} \left(2d(2d-1)^n\right)^{1/n} = (2d-1) \lim_{n\to\infty} (2d)^{1/n} = 2d - 1 \tag{25}$$

So, $\mu_d \leq 2d - 1$ which was to be shown.

For the lower bound, we can start from a lower bound on $c_n(d)$. In d dimensions we can be sure of obtaining a self avoiding walk if we restrict the walk to always go only in positive (or negative) directions along the dimension axes. This restriction results in a lower bound $c_n(d) \geq d^n$. We can once again use the definition for μ_d:

$$\mu_d = \lim_{n\to\infty} c_n(d)^{1/n} \geq \lim_{n\to\infty} (d^n)^{1/n} = d \tag{26}$$

So, $\mu_d \geq d$. Putt the lower and upper bound together to finalize:

$$d \geq \mu_d \geq 2d - 1 \tag{27}$$

The bound $A_d \geq 1$ for $d \geq 5$

We will here show that $A_d \geq 1$ for $d \geq 5$. This can easily be shown using previously introduced relations. Plugging eq 5 (for $d \geq 5$) into eq 2 yields (remember that $\gamma_d = 1$ for $d \geq 5$):

$$A_d \mu_d^{(n+m)} \leq A_d^2 \mu_d^n \mu_d^m = A_d^2 \mu_d^{(n+m)} \tag{28}$$

Using Proposition 1 we can safely deduce that $\mu_d^{(n+m)}$ is positive. We can also see that A_d must be positive from eq 5. We can then rewrite eq 28 by dividing by $\mu_d^{(n+m)}$ and A_d on both sides:

$$1 \leq A_d \quad \text{for } d \geq 5. \tag{29}$$

This hols for $n \to \infty$ (see eq 5), and any such sequence can be reindexed to a sequence of $n + m$ so that the above procedure can be applicable (for a sequence of length $k \geq 2$ we can set an arbitrary number of elements less or equal to $k - 1$ as the n first elements and the following as the m elements).

Comparing Theoretical Bounds with Practical Results

We will now compare practical estimates of the parameters for $d = 15$ with the bounds shown in eq 27 and eq 29. In both bounds we have used $n \to \infty$, so it is interesting to see how these hold when n is finite. We will also compare with an asymptotic bound μ_d for large d (Graham 2014):

$$\mu_d \sim 2d - 1 - \frac{1}{2d} - \frac{3}{(2d)^2} - \frac{16}{(2d)^3} + O(\frac{1}{d^4}) \tag{30}$$

We estimated the parameters A_d, μ_d and γ_d for $d = 15$ in 25 independent trials in the same way as these parameters was estimated previously, with the single change that $d = 15$ instead of 2. All the parameters in all trials respected their bounds, with $A_{15} \geq 1$, $15 \leq \mu_{15} \leq 29$, and $\gamma \approx 1$. We can also see that, for μ_{15} the upper bound is tight, however the lower bound is very loose. This is expected, as the lower bound on c_n that is used to motivate the lower bound is crude. In figure Figure (7) all parameters are plotted for each one of the 25 trials. The lines represent the bounds, where the uppermost line is the upper bound for μ_{15}, the middle line is the lower bound for μ_{15}, and the line at the bottom is the lower bound for A_{15}. As we could see that the μ values a re consistently close to their upper bound, we chose to calculate μ_d for $d = 1...15$ and compare it to the asymptotic bound given in eq 30. The result can be seen in Figure (8) and Figure (9). The calculated μ_d:s quickly get close to the asymptotic bound, however they overshoot it at times.

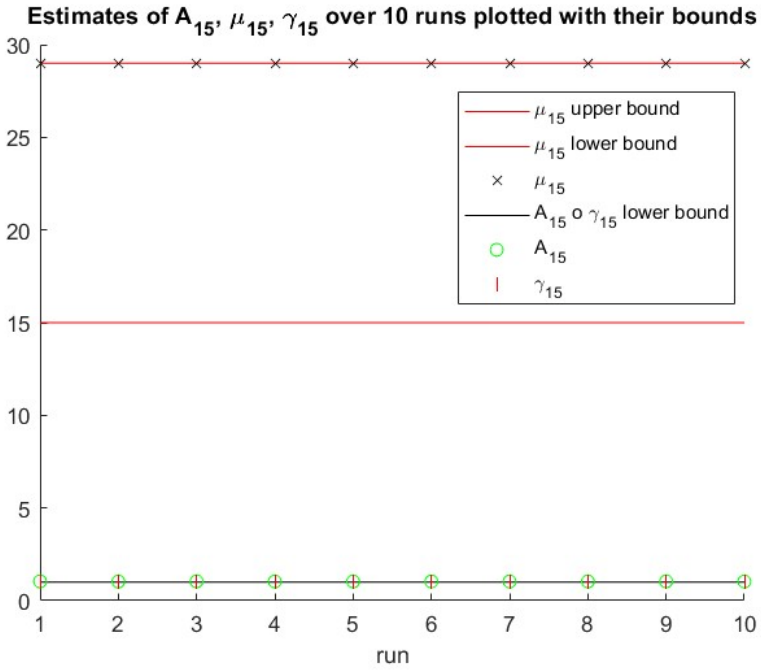

Figure 7: Estimates for A_{15}, μ_{15}, and γ_{15} and the closeness to their bounds over 10 runs. Each run is with an iid SISR up to $n = 199$ with 1000 particles.

Part 2 - Filter Estimation of Noisy Population Means

Let X_k be the relative population size in generation k of some organism. Relative population sizes are between zero and one where zero means extinction and one is the maximum size relative to the carrying capacity of the environment. However the closer we are to size one the more resources are consumed and then the next generation will be smaller. In a simplified relative population model we have the following dynamics.

$$X_{k+1} = B_{k+1}X_k(1 - X_k), B_{k+1} \in U(A, B), iid, k = 0, 1, \ldots$$

where B is the stochastic reproduction rate due to fluctuating environmental conditions. Assume that $X_0 \in U(C, D)$. Due to problems of measuring the exact relative population we get a measurement

$$Y_k|X_k = x \in U(Gx, Hx) \tag{31}$$

This can be seen as a hidden Markov model where the relative population size X is the hidden Markov chain, see Figure (10). We now want to estimate the filter expectation $\tau_k = E[X_k|Y_{0:k}]\text{for}k = 0, 1, 2, \ldots, 50$. Assume that the parameters are given as $A = 0.9$, $B = 3.9$, $C = 0.6$, $D = 0.99$, $G = 0.7$ and $H = 1.2$. We also have access to the measurements Y as well as the true values of X for comparison. We can here use the SISR method, with the uniform observation density given by 31, and initial distribution $U(C, D)$.

The Filter Expectation $\tau_k = \mathbb{E}[X_k|Y_{0:k}]$ for $k = 0, \ldots, 50$

As we can see in Figure (11) the estimates of τ behave as expected, as high values of τ_k are followed by lower ones in the next iteration, and vice versa.

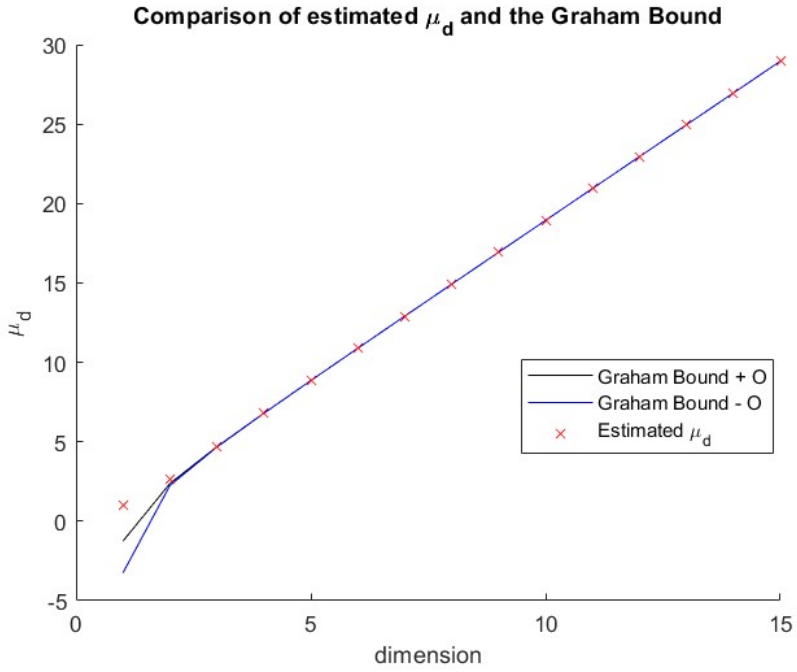

Figure 8: Estimates for $\mu_1, \ldots, \mu_{15}$. Each run is with an SISR up to $n = 199$ with 1000 particles. The graham bound is also plotted as an interval with, one for positive $O(\frac{1}{d^4})$ and one with negative. The μ:s are close to the bound but not inside it. See Figure (9).

Point wise Confidence Interval

A point wise confidence interval have also been constructed for X_k for $k = 1, \ldots 50$, see Figure (12). We can see that most of the true values are inside the point wise constructed 95% confidence interval, with a few outliers. This is clarified in Figure (13).

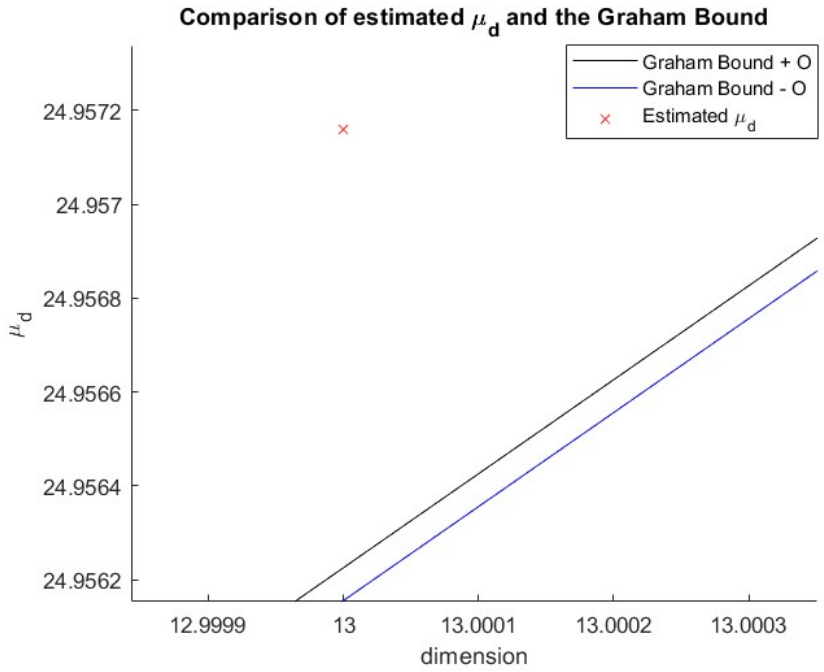

Figure 9: Estimate for $\mu_1 3$. The graham bound is the plotted lines, with $O = O(\frac{1}{d^4})$. The μ:s are close to the bound but not inside it.

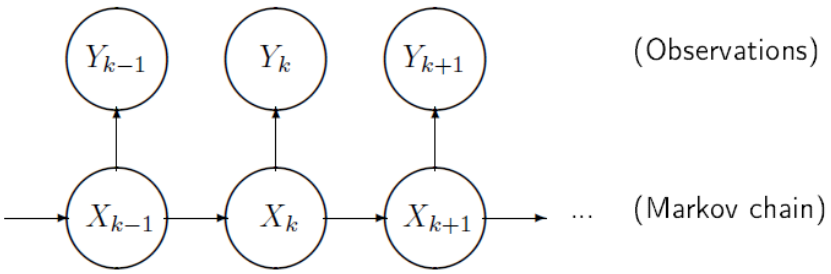

Figure 10: Illustration of an HMM, where X_k are the hidden actual values and Y_k are the measurements. (Lecture 7, slide 10)

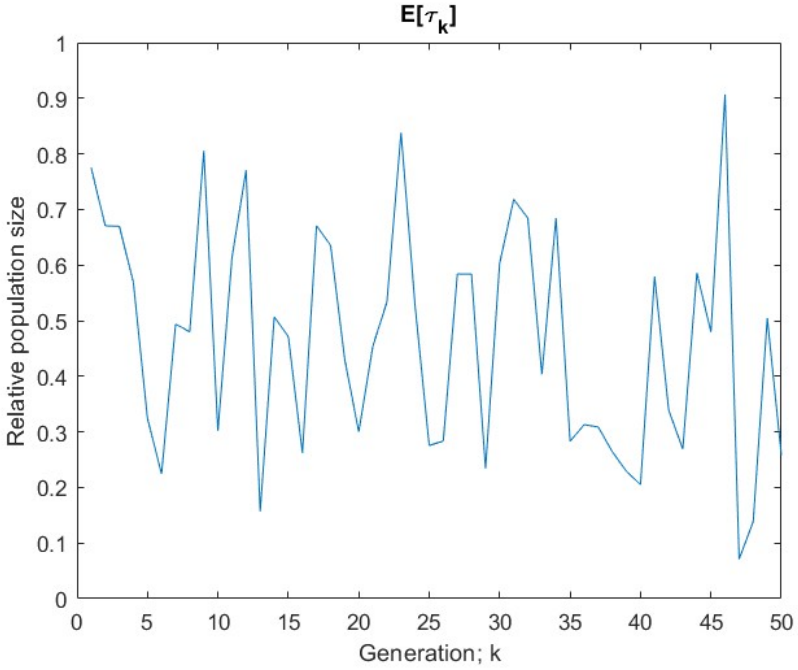

Figure 11: Expected value for τ as it evolves over generations from 0 to 50.

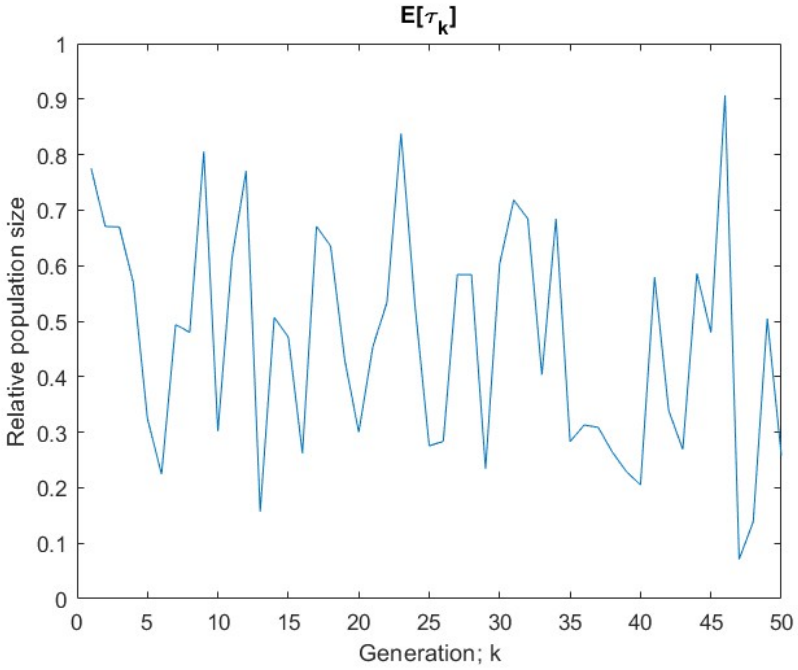

Figure 12: Upper and lower bounds on the relative population size and how these evolves with generations from 0 to 50. Also, the true X_k values are included, to show how these land with respect to the CI.

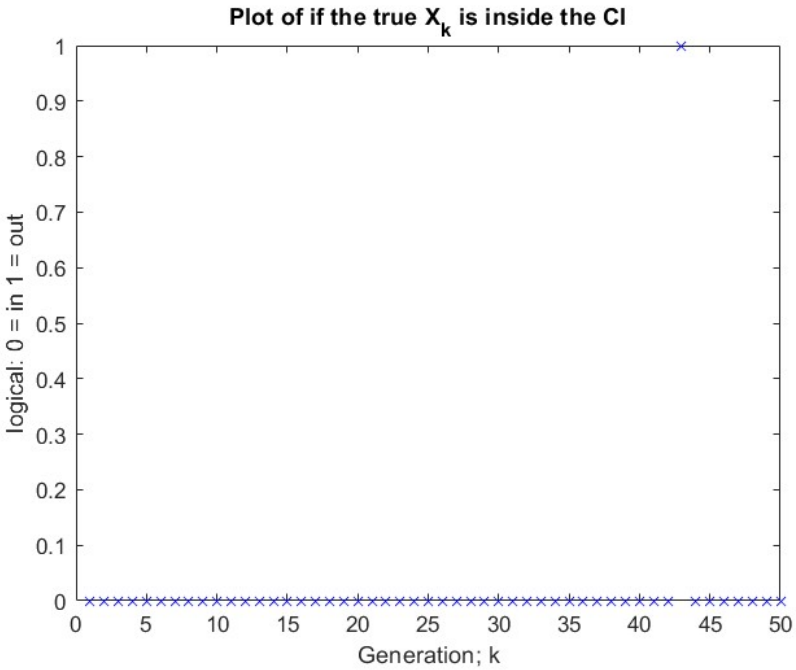

Figure 13: Logical plot of weather X_k lands inside its confidence interval for generations $1, \ldots, 50$. We can see that this only happens for one generation, k = 43.