Network Dynamics 3

Vidar Tobrand, vi5442to-s

May 2024

1 Introduction

The first part of this handin covers the simulation of return times and hitting times of a particle on a random walk graph. The second part covers graph coloring and network games

2 Single particle random walk

When analyzing random walks on graphs, also know as *Markoc chains* two interesting properties are the *return time* and *hitting time*.

2.1 Return time

The return time $E_a[\bar{T}_a^+]$ is the time it takes for a single particle to return to node a when starting in a. This return time was simulated in MATLAB. The theory behind the simulations starts with the transition rate matrix,

$$\Lambda = \begin{array}{c|c} & \begin{matrix} o & a & b & c & d \end{matrix} \\ \hline \begin{matrix} o \\ a \\ b \\ c \\ d \end{matrix} & \begin{pmatrix} 0 & \frac{2}{5} & \frac{1}{5} & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{3} & 0 & \frac{2}{3} \\ 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 \end{pmatrix} \end{array}$$

which describes the rate of the *Poisson clock* of each transition. The rate vector of the distribution,

$$\omega = \Lambda \mathbb{1}, \tag{1}$$

describes the rate of each node. The probability transition matrix is then computed as,

$$P = (diag(\omega))^{-1}\Lambda. \tag{2}$$

To simulate the random walks a random variable u was drawn from the set $U \in (0, 1)$ and compared to the cumulative transition matrix P_{sum} which accumulates the probabilities in the row of the transition probability matrix $\bar{P}$, so the last entry of a row summarizes to 1.

The theoretical hitting time was also computed and compared to the simulated values. Starting with the transition probability matrix,

$$\bar{P}_{ij} = \frac{\Lambda_{ij}}{w_*}, \quad w_* = \max_i w_i, \\ \bar{P}_{ii} = 1 - \sum_{j \neq i} \bar{P}_{ij}. \tag{3}$$

According to *theorem 7.2* in the lecture notes the expected return time is then,

$$E_a[\bar{T}_a^+] = \frac{1}{\omega_a \bar{\pi}_a}. \tag{4}$$

where for the invariant distribution it follows that,

$$\bar{P}_{a,0-j} \bar{\pi}_a = \bar{\pi}_a. \tag{5}$$

2.2 Hitting time

The average hitting time,

$$\bar{\tau}_i^S = E_i[T_S], \quad i \in \chi, \tag{6}$$

is according to *Theorem 7.2* in the lecture notes, the solution of the linear system

$$\bar{\tau}_s^S = 0, \quad s \in S, \qquad \bar{\tau}_i^S = \frac{1}{\omega_i} + \sum_{j \in \chi} P_{ij} \bar{\tau}_j^S. \tag{7}$$

This system can be solved by,

$$\bar{\tau}^S = (diag(w)P)^{-1} \frac{1}{w}. \tag{8}$$

where $P_{:,s} = 0$ & $\frac{1}{\omega_s} = 0$ The results are presented in the next section.

3 Results

The matrices and vectors described in the theory was computed to,

$$\omega = \begin{bmatrix} 0.60 \\ 1.00 \\ 1.00 \\ 1.00 \\ 0.67 \end{bmatrix}$$

,

$$P = \begin{bmatrix} 0.00 & 0.67 & 0.33 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.75 & 0.25 & 0.00 \\ 0.50 & 0.00 & 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.33 & 0.00 & 0.67 \\ 0.00 & 0.50 & 0.00 & 0.50 & 0.00 \end{bmatrix}$$

,

$$\bar{P} = \begin{bmatrix} 0.40 & 0.40 & 0.20 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.75 & 0.25 & 0.00 \\ 0.50 & 0.00 & 0.00 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.33 & 0.00 & 0.67 \\ 0.00 & 0.33 & 0.00 & 0.33 & 0.33 \end{bmatrix}$$

,

$$P_{sum} = \begin{bmatrix} 0.40 & 0.80 & 1.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.75 & 1.00 & 0.00 \\ 0.50 & 0.00 & 0.00 & 1.00 & 0.00 \\ 0.00 & 0.00 & 0.33 & 0.00 & 1.00 \\ 0.00 & 0.33 & 0.00 & 0.67 & 1.00 \end{bmatrix}$$

Using these matrices as described in the theory and running the simulations for a million iterations the average simulated- & theoretical return time, and the average simulated- & theoretical hitting time was calculated and are presented in table 1.

Table 1: Simulated and calculated hitting times

	$E_a[\bar{T}_a^+]$	$E_o[\bar{T}_d]$
Simulated	6.7 s	8.8 s
Theoretical	6.75 s	8.79 s

Here we can see that the difference between simulation and theoretical value differ on the second decimal. The reason for only using one decimal for the simulated values is fluctuation from 6.7-6.8 and 8.7-8.8.

4 Graph coloring and network games

Network games are games where each node in a graph represents a player. In this case we have 10 player represented in a line graph, see figure 1. The goal of the game is to color each node in the graph with different colors. In a line graph with no cycles the number of colors needed are two. This is the goal of the second part of the handin.

A network game can be represented as a potential game where each node $X(i) \in \nu$ is a player. The actions are represented by a action set $a \in \mathcal{A}$, in this case $\mathcal{A} = \{Red, Green\}$. To determine which action is best for the current player a cost function $c(a, x_{-i})$ is introduced. Which is defined as

$$c(a, X(-i)) = \sum_j W_{i,j} c(a, X(j)) \tag{9}$$

where the cost for a specific node is given by

$$c(s, X_j(t)) = \begin{cases} 1 & \text{if } X_j(t) = s \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

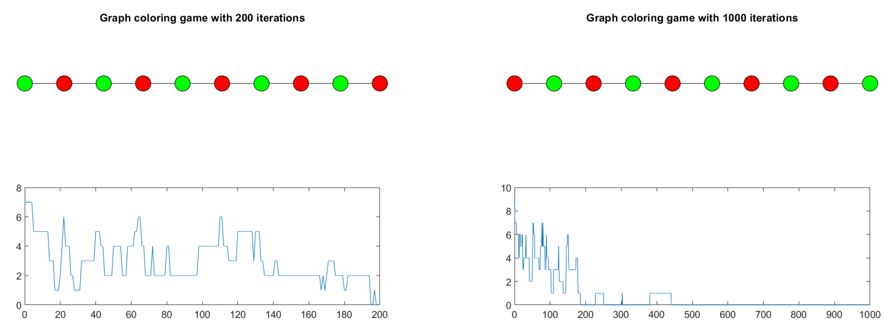
The probaility for each action follows a distribution given by

$$P(X_i(t+1) = a \mid X(t), I(t) = i) = \frac{e^{-\eta(t) \sum_j W_{ij} c(a, X_j(t))}}{\sum_{s \in \mathcal{C}} e^{-\eta(t) \sum_j W_{ij} c(s, X_j(t))}} \tag{11}$$

where $\eta(t)$ is the inverse of the noise, in this case $\eta(t) = t/100$. The potential function is the number of links between nodes with the same color. The potential function follows,

$$U(t) = \frac{1}{2} \sum_{i,j \in \nu} W_{ij} c(X_i(t), X_j(t)). \tag{12}$$

The simulations for the network game of the linegraph is shown in figure 1.



(a) 200 iterations (b) 1000 iterations

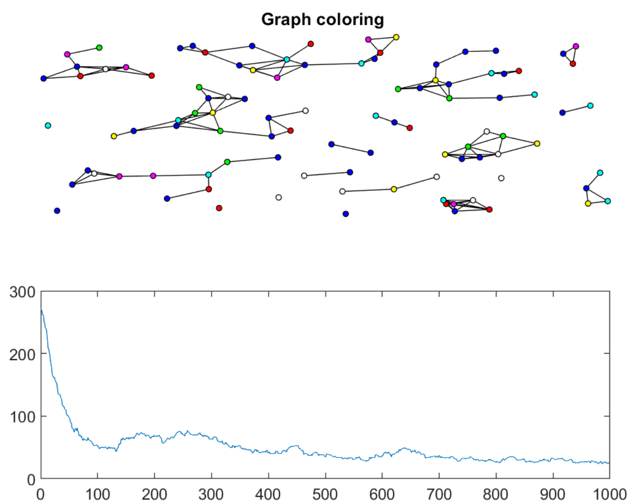
Figure 1: The state space at the last iteration and the potential function for each timestamp.

Here we see that the potential function does reach 0 when the game proceeds for 200 iterations but just barely. When iterating 1000 steps the majority of the state spaces have a potential function of 0 meaning that the graph coloring is optimal.

In the second part a larger graph is with 100 nodes and several nodes with higher degree than 2. The nodes represent a network were each node has a specific bandwidth, illustrated by a specific color. The action set $\mathcal{A} = \{red, green, blue, yellow, magenta, cyan, whiteandblack\}$. To minimize the interference between the nodes a cost function that punishes same nearby bandwidth for neighboring nodes are introduced. The cost function follows,

$$c(s, X_j(t)) = \begin{cases} 2 & \text{if } X_j(t) = s \\ 1 & \text{if } |X_j(t) - s| = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

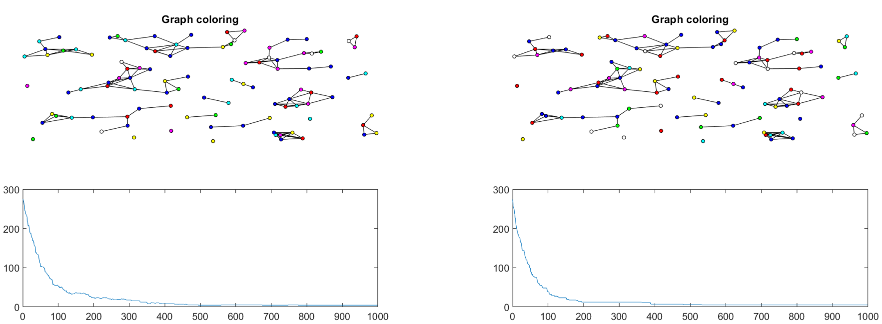
The probability distribution still follows equation 11. The results for 1000 iterations with $\eta = t/100$ is shown in figure 3.



$$\eta = \frac{t}{1000}$$

Figure 2: Graph coloring result and potential function over time.

Here we can see that the potential function does not converge. However in the plots in figure 3 we see that the higher the inverse of the noise, meaning less noise results in the graph converging to zero.



(a) $\eta = \frac{t}{100}$ (b) $\eta = log(t+1)$

Figure 3: Graph coloring result and potential function over time.