A MODIFIED ARNOLDI ITERATION FOR TRANSITION PROBABILITY MATRICES OF REVERSIBLE MARKOV CHAINS

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE UNIVERSITY OF HAWAII AT MĀNOA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

August 2018

BY

JOSEPH CHONG

Thesis Committee:

June Zhang, Chairperson

Bjørn Kjos-Hanssen

Galen Sasaki

Abstract

Reversible Markov chains are used for modeling many physical and network phenomena. The second largest eigenvalue magnitude of the transition probability matrix gives a upper bound on the mixing time of a reversible Markov chain, but is incalculable for large transition probability matrices using typical eigenvalue algorithms. We present the Modified Arnoldi iteration - a modification of the Arnoldi iteration for reversible Markov chains that utilizes sample estimates where matrix operations may be infeasible, thereby being a possible option when usual algorithms are nonviable.

Acknowledgments

I would like to thank my advisor Dr. Zhang for the opportunity to work on this project. I would also like to thank Dr. Kjos-Hanssen and Dr. Sasaki for graciously taking time out of their busy schedules to be on my thesis committee. Finally, I would like to thank my family and friends for their support in my day-to-day life.

Contents

List of Figures

Chapter 1

Introduction

Reversible Markov chains find application in many settings. One famous application of reversible Markov chains is the Metropolis-Hastings algorithm [7], where a reversible Markov chain is constructed to converge to a distribution that is difficult to sample from directly. Other applications of reversible Markov chains include Birth-death processes [11], where a state variable increments or decrements its value with certain rates, and the scaled SIS process [19], which models the spread of epidemics on a graph.

The second largest eigenvalue magnitude of the transition probability matrix is of interest because it gives a upper bound on the mixing time of the Markov chain [13]. The knowledge provided by the second largest eigenvalue magnitude will be used by the user of the Metropolis-Hastings algorithm to decide when to begin recording samples, or the user of the scaled SIS process to find the network states with highest probability at equilibrium. The challenge in calculating the second largest eigenvalue magnitude lies in the fact that we may not be able to use typical eigenvalue algorithms, as the transition probability matrix can be too large to load into memory. For example, the size of the state space for scaled SIS is 2^M, where M is the number of nodes on a graph. Practical networks are composed of hundreds to thousands of nodes, but the largest vector a typical computer can load may be length 2^{15}.

The Arnoldi iteration, first given in [1], is a commonly used technique in solving eigenvalue problems. Suppose we wish to estimate the eigenvalues of P. Let $\boldsymbol{f}$ be

some trial vector, n some positive integer, and $K_n[\boldsymbol{f}]$ be the subspace defined as

$$K_n[\boldsymbol{f}] = \text{span}\{\boldsymbol{f}, P\boldsymbol{f}, \ldots, P^{n-1}\boldsymbol{f}\}.$$

The Arnoldi iteration approximates the eigenvalues of P by identifying the vector $\tilde{\boldsymbol{v}} \in K_n[\boldsymbol{f}]$, and scalar $\tilde{\lambda} \in \mathbb{C}$ that best approximates the condition

$$P\tilde{\boldsymbol{v}} \approx \tilde{\lambda}\tilde{\boldsymbol{v}}.$$

The steps of the Arnoldi iteration involve computations that are infeasible when the size of the state space is sufficiently large. We develop in this thesis modifications to the Arnoldi iteration specifically for finding the second largest eigenvalue magnitude of transition probability matrices of reversible Markov chains, utilizing sample estimates in places where the algorithm calls for computations on prohibitively large matrices and vectors, thus opening options when typical matrix operations are infeasible.

1.1 Previous Works

Work that is pertinent to our topic include analytical methods of bounding the second largest eigenvalue - Cheeger bounds and Poincaire bounds [5] are commonly used. We develop numerical options for cases where analytical methods are not easily applicable. An alternative to calculating analytical bounds is to utilize the samples generated by the Markov chain to create estimates. Past works utilizing samples generated from a Markov chain to estimate the second largest eigenvalue include [14], [9] [6].

The authors of [14] consider the problem of using sample data generated by a continuous time reversible Markov chain to estimate the second largest eigenvalue when the transition rate matrix has the form

$$Q = \sum_{k=1}^{r} \alpha_k Q_k.$$

where Q_k are known, but α_k are not. A method to calculate the parameter esti-

mates $\hat{\alpha}_k$ using samples generated over the time interval $[0, t]$ is introduced, and the transition rate matrix is estimated to be

$$\widehat{Q} = \sum_{k=1}^{r} \hat{\alpha}_k Q_k.$$

The authors proceed to prove several conditions on the asymptotic behavior of several estimators as $t \to \infty$, including $\hat{\lambda}_2$, the second largest eigenvalue of $\widehat{Q}$.

The authors of [9] consider the problem of bounding the mixing time of a discrete time reversible Markov chain when the transition probabilities are unknown. A method of approximating the transition probabilities using the relative frequencies of observed state transitions from generated samples is introduced, which allows for the construction of a approximate transition probability matrix $\widehat{P}$. The second largest eigenvalue magnitude is then estimated by solving for the eigenvalues of $\widehat{P}$. Subsequent steps are developed to calculate confidence intervals for the estimate.

Our work draws upon Gade's 2007 thesis [6], which addresses the problem of estimating the second largest eigenvalue magnitude of transition probability matrices of reversible Markov chains (see Appendix A.1 for a detailed discussion of [6]). The method given in [6] utilizes the Krylov subspace along with the autocovariance of the Markov chain to calculate estimates, and requires solving a ill-conditioned problem. The method presented in this thesis circumvents the step of solving a ill-conditioned problem.

1.2 Organization

This thesis is organized as follows. Chapter 2 gives the background concepts on reversible Markov chains and the Arnoldi iteration. Chapter 3 presents our Modified Arnoldi iteration. Chapter 4 shows the results we've obtained on model problems. Chapter 5 is the conclusion to this thesis.

Chapter 2

Background

The eigenvectors and eigenvalues of a $N \times N$ matrix P are the vectors $\boldsymbol{v}$ and scalars λ that satisfy

$$P\boldsymbol{v} = \lambda\boldsymbol{v}.$$

If P is the transition probability matrix of a reversible Markov chain, it holds several properties that are pertinent to solving the eigenvalue problem. We present in this chapter some properties of reversible Markov chains and numerical techniques used in the study of eigenvalue problems.

2.1 Markov Chains

The Markov chains analyzed in this thesis have the property of being *ergodic*: irreducible and aperiodic. We will limit our discussion to Markov chains with finite state space. We will use χ to denote the state space, and N to denote the size of the state space. The $N \times N$ transition probability matrix P contains the transition probabilities from state i to j, p_{ij}, where $i, j \in \chi$. The equilibrium distribution π for an ergodic Markov chain is known to be unique [3]. It is also the dominant left eigenvector of P with strictly positive entries summing to one. The dominant eigenvalue is 1, and the corresponding right eigenvector is $\mathbf{1}$:

$$\pi P = \pi$$

$$P\mathbf{1} = \mathbf{1}.$$

2.1.1 Reversible Markov Chains

The definitions and theorems in this section are from [3]. A reversible Markov chain is defined as follows.

Definition 2.1.1. *A Markov chain with transition probability matrix P is reversible if its equilibrium distribution satisfies*

$$\pi_i P(i,j) = \pi_j P(j,i) \tag{2.1}$$

for all i, j in χ, where π is the equilibrium distribution and $P(i,j)$ is the transition probability from i to j.

We next introduce the inner product space $\ell^2(\pi)$, given in [3], which has properties that are useful when studying the eigenvalues of P. Let π be a strictly positive distribution. The inner product space $\ell^2(\pi)$ is the vector space $\mathbb{R}^N$ with the inner product

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle_\pi = \sum_{i \in \chi} x_i y_i \pi_i, \qquad \forall \boldsymbol{x}, \boldsymbol{y} \in \ell^2(\pi) \tag{2.2}$$

and norm

$$\|\boldsymbol{x}\|_\pi = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle_\pi} = \sqrt{\sum_{i \in \chi} x_i^2 \pi_i} < \infty. \tag{2.3}$$

Given the state of a Markov chain, it may be easier or advantageous to work with some function of the state, rather than to work with the state itself. The observable vector $\boldsymbol{f}$ is the member of $\ell^2(\pi)$ obtained by mapping each Markov state, $i \in \chi$, to $\mathbb{R}$, $f : \chi \to \mathbb{R}$:

$$\boldsymbol{f} = \begin{bmatrix} f(i_1) \\ f(i_2) \\ \vdots \\ f(i_N) \end{bmatrix}. \tag{2.4}$$

Let X_t be a sample from a Markov chain at time t. Using the equilibrium distribution

as the initial distribution, the mean and variance of $f(X_t)$ is

$$\mathbb{E}_\pi [f(X_t)] = \langle \boldsymbol{f} \rangle_\pi = \sum_{i \in \chi} f(i)\pi_i \tag{2.5}$$

$$\mathrm{var}_\pi (f(X_t)) = \|\boldsymbol{f}\|_\pi^2 - \langle \boldsymbol{f} \rangle_\pi^2. \tag{2.6}$$

The following example illustrates the aforementioned calculations.

Example 2.1.1. *Suppose there is a Markov chain describing the evolution of a epidemic on a two node network. Each node is either infected (1) or susceptible (0). The state space of the Markov chain is:*

$$i_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad i_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad i_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad i_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

If we select $f(i)$ to be the number of infected nodes in state i, then $\boldsymbol{f}$ is

$$\boldsymbol{f} = \begin{bmatrix} f(i_0) \\ f(i_1) \\ f(i_2) \\ f(i_3) \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \end{bmatrix}.$$

Suppose the equilibrium distribution is

$$\pi = \begin{bmatrix} \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & \frac{1}{3} \end{bmatrix}.$$

The mean of $f(X_t)$ with respect to π is

$$\langle \boldsymbol{f} \rangle_\pi = \sum_{i \in \chi} f(i)\pi_i = 0 \times \frac{1}{3} + 1 \times \frac{1}{6} + 1 \times \frac{1}{6} + 2 \times \frac{1}{3} = 1.$$

The norm of $\boldsymbol{f}$ with respect to π is

$$\|\boldsymbol{f}\|_\pi = \sqrt{\sum_{i \in \chi} f(i)^2 \pi_i} = \sqrt{\frac{1}{6} + \frac{1}{6} + 4 \times \frac{1}{3}} = \sqrt{\frac{5}{3}}.$$

The variance of $f(X_t)$ with respect to π is

$$\|\boldsymbol{f}\|_\pi^2 - \langle \boldsymbol{f} \rangle_\pi^2 = \frac{5}{3} - 1.$$

If the Markov chain has reached equilibrium by time t, then the probability of observing $X_t = i, X_{t+s} = j$ is $\pi_i P^s(i,j)$. We can then define the autocovariance of $f(X_t)$ as

$$\begin{aligned} C_f(s) &= \mathbb{E}_\pi\left[f(X_t) f(X_{t+s})\right] - \mathbb{E}_\pi\left[f(X_t)\right]^2 \\ &= \sum_{i\in\chi}\sum_{j\in\chi} P^s(i,j) f(i) f(j) \pi_i - \left(\sum_{i\in\chi} f(i)\pi_i\right)^2 \\ &= \langle \boldsymbol{f}, P^s \boldsymbol{f} \rangle_\pi - \langle \boldsymbol{f} \rangle_\pi^2. \end{aligned} \tag{2.7}$$

Note that when starting from the equilibrium distribution, reversible Markov chains are *wide sense stationary* - the mean is independent of time, and the autocovariance depends not on two points in time, but the difference between them.

The following theorem from [3] will be utilized on multiple occasions throughout the course of this thesis.

Theorem 2.1.1. *A Markov chain with transition probability matrix P is reversible if and only if*

$$\langle \boldsymbol{x}, P\boldsymbol{y} \rangle_\pi = \langle \boldsymbol{y}, P\boldsymbol{x} \rangle_\pi \tag{2.8}$$

for all $\boldsymbol{x}, \boldsymbol{y}$ in $\ell^2(\pi)$.

Definition 2.1.2. *The transition probability matrix P is **self-adjoint** in $\ell^2(\pi)$ if*

$$\langle \boldsymbol{x}, P\boldsymbol{y} \rangle_\pi = \langle \boldsymbol{y}, P\boldsymbol{x} \rangle_\pi \tag{2.9}$$

for all $\boldsymbol{x}, \boldsymbol{y}$ in $\ell^2(\pi)$.

The next theorem characterizes the eigenvalues and eigenvectors of reversible Markov chains.

Theorem 2.1.2. *The transition matrix P of a reversible Markov chain with equilibrium distribution π satisfies the following:*

1. *P has real eigenvalues.*

2. *The right eigenvectors of P form a orthonormal basis of $\ell^2(\pi)$.*

3. *P is diagonalizable.*

2.2 Rayleigh-Ritz and the Arnoldi Iteration

The Rayleigh-Ritz procedure is a method of obtaining a eigenvalue estimate using vectors from a n dimensional subspace. As we will see, the main advantage in seeking the best eigenvalue approximations from a subspace instead of solving directly the original $N \times N$ eigenvalue problem is in the dramatic simplification to a much more manageable $n \times n$ eigenvalue problem, where $n \ll N$. The following construction of the Rayleigh-Ritz procedure is from [2]. We begin detailing the Rayleigh-Ritz procedure by introducing the best approximation theorem.

Theorem 2.2.1 (The Best Approximation Theorem [12])**.** *Let W be a subspace of $\mathbb{R}^N$, $\boldsymbol{y}$ any vector of $\mathbb{R}^N$, and $\tilde{\boldsymbol{y}}$ the orthogonal projection of $\boldsymbol{y}$ onto W. Then $\tilde{\boldsymbol{y}}$ is the closest point in W to $\boldsymbol{y}$, in the sense that*

$$||\boldsymbol{y} - \tilde{\boldsymbol{y}}|| \leq ||\boldsymbol{y} - \boldsymbol{x}|| \tag{2.10}$$

for all $\boldsymbol{x}$ in W.

Let P be the matrix whose eigenvalues we wish to approximate, and we do so by finding $\tilde{\boldsymbol{v}} \in W$, along with $\tilde{\lambda} \in \mathbb{C}$ that gives the closest approximation to

$$P\tilde{\boldsymbol{v}} \approx \tilde{\lambda}\tilde{\boldsymbol{v}} \tag{2.11}$$

($\tilde{\boldsymbol{v}}$ and $\tilde{\lambda}$ are eigenvector and eigenvalue appoximations). From the Best Approxima-

tion Theorem, $\tilde{\lambda}\tilde{\boldsymbol{v}}$ will be the orthogonal projection of $P\tilde{\boldsymbol{v}}$ onto W. As a result,

$$P\tilde{\boldsymbol{v}} - \tilde{\lambda}\tilde{\boldsymbol{v}} \perp W.$$

Equivalently,

$$\boldsymbol{q}^T(P\tilde{\boldsymbol{v}} - \tilde{\lambda}\tilde{\boldsymbol{v}}) = 0 \tag{2.12}$$

for all $\boldsymbol{q} \in W$.

Let $Q = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_n]$ be a $N \times n$ matrix with columns forming a orthonormal basis for W (ideally, $n \ll N$). From Equation (2.12) we have

$$Q^T(P\tilde{\boldsymbol{v}} - \tilde{\lambda}\tilde{\boldsymbol{v}}) = \mathbf{0}. \tag{2.13}$$

Since $\tilde{\boldsymbol{v}} \in W$, $\tilde{\boldsymbol{v}}$ will be a linear combination of the columns of Q, so $\tilde{\boldsymbol{v}} = Q\boldsymbol{z}$ for some $\boldsymbol{z} \in \mathbb{R}^n$. Inserting into Equation (2.13) gives

$$Q^T(PQ\boldsymbol{z} - \tilde{\lambda}Q\boldsymbol{z}) = \mathbf{0}. \tag{2.14}$$

Rearranging and making the substitution $H_n = Q^TPQ$ gives

$$H_n\boldsymbol{z} = \tilde{\lambda}\boldsymbol{z}. \tag{2.15}$$

Therefore the eigenvalues of the H_n are estimates of the eigenvalues of P. The problem of finding the eigenvalues of the $N \times N$ matrix P has been reduced to finding the eigenvalues of the $n \times n$ matrix H_n. The Rayleigh-Ritz method is summarized in Algorithm (2.1).

Algorithm 2.1 Rayleigh-Ritz Procedure [2]

Input: Size $N \times N$ matrix P and n dimensional subspace W.

1: Calculate a orthonormal basis for W, and store the result as the columns of the $N \times n$ matrix Q.
2: Calculate $H_n = Q^TPQ$.
3: Find the eigenvalues of the $n \times n$ matrix H_n.

2.2.1 The Arnoldi Iteration

The dimension n Krylov subspace is defined as

$$K_n[\boldsymbol{f}] = \text{span}\{\boldsymbol{f}, P\boldsymbol{f}, \ldots, P^{n-1}\boldsymbol{f}\}$$

where $\boldsymbol{f}$ is a trial vector. We will refer to the vectors $\boldsymbol{f}, P\boldsymbol{f}, \ldots, P^{n-1}\boldsymbol{f}$ as the power iterates. The Arnoldi iteration carries out the Rayleigh-Ritz procedure using $K_n[\boldsymbol{f}]$ as the choice for W. Algorithm (2.2), from [18], gives the steps of the Arnoldi iteration. The result of the Arnoldi iteration is the $(n+1) \times n$ matrix H; truncating the last row

Algorithm 2.2 Arnoldi Iteration [18]

Input:
Matrix P
Trial vector $\boldsymbol{f}$
Number of dimensions n

Output:
$(n+1) \times n$ matrix H

1: $H \leftarrow (n+1) \times n$ zero matrix
2: $\boldsymbol{q}_1 \leftarrow \frac{1}{\sqrt{\langle \boldsymbol{f}, \boldsymbol{f} \rangle}} \boldsymbol{f}$
3: **for** $k = 1, \ldots, n$ **do**
4: $\quad \boldsymbol{u} \leftarrow P\boldsymbol{q}_k$
5: $\quad$ **for** $j = 1, \ldots, k$ **do**
6: $\quad\quad h_{j,k} \leftarrow \langle \boldsymbol{u}, \boldsymbol{q}_j \rangle$
7: $\quad\quad \boldsymbol{u} \leftarrow \boldsymbol{u} - h_{jk}\boldsymbol{q}_j$
8: $\quad$ **end for**
9: $\quad h_{k+1,k} \leftarrow \sqrt{\langle \boldsymbol{u}, \boldsymbol{u} \rangle}$
10: $\quad \boldsymbol{q}_{k+1} \leftarrow \boldsymbol{u}/h_{k+1,k}$
11: **end for**

gives H_n. Solving for the eigenvalues of the $n \times n$ matrix H_n gives the approximations to n eigenvalues of P. Typically, these approximate the n eigenvalues of P largest in magnitude.

The Arnoldi iteration calls for a computation using a matrix of size $N \times N$ in line 4 ($O(N^2)$ computations), and vectors of length N in lines 2, 6, 7, 9, and 10 ($O(N)$ computations for each line). Such computations become infeasible for sufficiently large N. We propose a modification of the Arnoldi Iteration using properties of reversible Markov chains to estimate the second largest eigenvalue of P without computations

on $N \times N$ matrices or length N vectors.

Chapter 3

Modified Arnoldi Iteration

Given a reversible Markov chain with a state space of size N and equilibrium distribution π, we wish to find the second largest eigenvalue magnitude of the corresponding $N \times N$ transition probability matrix P, as the second largest eigenvalue magnitude is known to give a upper bound on the mixing time of the Markov chain [13]. The Arnoldi iteration gives $\tilde{\lambda}_i$, the best approximation to the ith eigenvalue of P using the dimension n Krylov subspace $K_n[\boldsymbol{f}]$. However, the steps of the Arnoldi iteration consists of computations with matrices and vectors of size N, which may be infeasible for sufficiently large N. This chapter introduces modifications to the Arnoldi iteration that are possible when the Markov chain is reversible. We will use λ_* and $\boldsymbol{v}_*$ to refer to the second largest eigenvalue magnitude and the corresponding right eigenvector of P.

3.1 Modified Arnoldi Iteration

The approximation of eigenvalues using the Arnoldi iteration is accomplished by minimizing the Euclidean norm of the error:

$$\min ||P\tilde{\boldsymbol{v}} - \tilde{\lambda}\tilde{\boldsymbol{v}}||. \tag{3.1}$$

To incorporate the properties of reversible Markov chains into our analysis, consider, instead, minimizing the norm of the error with respect to π:

$$\min ||P\tilde{\boldsymbol{v}} - \tilde{\lambda}\tilde{\boldsymbol{v}}||_{\pi}, \tag{3.2}$$

which has the equivalent interpretation of minimizing the average squared error:

$$\min \mathbb{E}_{\pi}\left(P\tilde{\boldsymbol{v}} - \tilde{\lambda}\tilde{\boldsymbol{v}}\right)^2.$$

The resulting Arnoldi iteration is shown in Algorithm (3.1). Algorithm (3.1) as

Algorithm 3.1 Arnoldi Iteration on $\ell^2(\pi)$

Input:
Matrix P
Trial vector $\boldsymbol{f}$
Number of dimensions n

Output:
$(n+1) \times n$ matrix H

1: $H \leftarrow (n+1) \times n$ zero matrix
2: $\boldsymbol{q}_1 \leftarrow \frac{1}{\sqrt{\langle \boldsymbol{f}, \boldsymbol{f} \rangle_{\pi}}} \boldsymbol{f}$
3: **for** $k = 1, \ldots, n$ **do**
4: $\quad \boldsymbol{u} \leftarrow P\boldsymbol{q}_k$
5: $\quad$ **for** $j = 1, \ldots, k$ **do**
6: $\quad\quad h_{j,k} \leftarrow \langle \boldsymbol{u}, \boldsymbol{q}_j \rangle_{\pi}$
7: $\quad\quad \boldsymbol{u} \leftarrow \boldsymbol{u} - h_{jk}\boldsymbol{q}_j$
8: $\quad$ **end for**
9: $\quad h_{k+1,k} \leftarrow \sqrt{\langle \boldsymbol{u}, \boldsymbol{u} \rangle_{\pi}}$
10: $\quad \boldsymbol{q}_{k+1} \leftarrow \boldsymbol{u}/h_{k+1,k}$
11: **end for**

written still requires $O(N^2)$ computations in line 4, and $O(N)$ computations (in addition to knowledge of π) in lines 2, 6, and 9. It was shown in [6] that inner products with respect to π between vectors in $K_n[\boldsymbol{f}]$ are equal to linear combinations of the autocovariance of $\boldsymbol{f}$. With this observation in mind, we now construct a method that carries out the Arnoldi iteration utilizing autocovariances, instead of directly calculating inner products in $\ell^2(\pi)$. Assuming $\mathbb{E}_{\pi}\boldsymbol{f} = 0$, the autocovariance

of $\boldsymbol{f}$ is

$$\begin{aligned} C_f(s) &= \langle \boldsymbol{f}, P^s \boldsymbol{f} \rangle_\pi - \langle \boldsymbol{f} \rangle_\pi \\ &= \langle \boldsymbol{f}, P^s \boldsymbol{f} \rangle_\pi. \end{aligned}$$

The sample autocovariance is

$$\widehat{C}_f(s) = \frac{1}{T-|s|} \sum_{t=0}^{T-|s|-1} \left(f(X_t) - \bar{f} \right) \left(f(X_{t+s}) - \bar{f} \right), \tag{3.3}$$

where $\bar{f}$ is the sample mean. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be the following two vectors in the Krylov subspace $K_n[\boldsymbol{f}]$:

$$\begin{aligned} \boldsymbol{x} &= \xi_1 \boldsymbol{f} + \xi_2 P \boldsymbol{f} + \cdots + \xi_n P^{n-1} \boldsymbol{f} \\ \boldsymbol{y} &= \phi_1 \boldsymbol{f} + \phi_2 P \boldsymbol{f} + \cdots + \phi_n P^{n-1} \boldsymbol{f}. \end{aligned}$$

The inner product of $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\ell^2(\pi)$ is

$$\begin{aligned} \langle \boldsymbol{x}, \boldsymbol{y} \rangle_\pi &= \sum_{i=1}^{n} \sum_{j=1}^{n} \xi_i \phi_j \langle P^{i-1} \boldsymbol{f}, P^{j-1} \boldsymbol{f} \rangle_\pi \\ &= \sum_{i,j} \xi_i \phi_j \langle \boldsymbol{f}, P^{i+j-2} \boldsymbol{f} \rangle_\pi \\ &= \sum_{i,j} \xi_i \phi_j C_f(i+j-2). \end{aligned} \tag{3.4}$$

If we have good estimates for the autocovariance of $\boldsymbol{f}$, we can estimate the inner products using Equation (3.4), thereby circumventing computations over N numbers. To illustrate, consider the Arnoldi iteration for $n = 1$:

- $\boldsymbol{q}_1 \leftarrow \frac{1}{\sqrt{\langle \boldsymbol{f}, \boldsymbol{f} \rangle_\pi}} \boldsymbol{f} = \frac{1}{\sqrt{C_f(0)}} \boldsymbol{f}$
- $\boldsymbol{u} \leftarrow P \boldsymbol{q}_1 = \frac{1}{\sqrt{C_f(0)}} P \boldsymbol{f}$
- $h_{1,1} \leftarrow \langle \boldsymbol{u}, \boldsymbol{q}_1 \rangle_\pi \quad = \langle \frac{1}{\sqrt{C_f(0)}} P \boldsymbol{f}, \frac{1}{\sqrt{C_f(0)}} \boldsymbol{f} \rangle_\pi$
  $= \frac{C_f(1)}{C_f(0)}$

- $\boldsymbol{u} \leftarrow \boldsymbol{u} - h_{11}\boldsymbol{q}_1 = \frac{1}{\sqrt{C_f(0)}} P\boldsymbol{f} - h_{11}\frac{1}{\sqrt{C_f(0)}}\boldsymbol{f}$
- $h_{2,1} \leftarrow \sqrt{\langle \boldsymbol{u}, \boldsymbol{u} \rangle_\pi} = \left[\frac{C_f(2)}{C_f(0)} - 2h_{11}^2 + h_{11}^2\right]^{1/2}$
- $\boldsymbol{q}_2 \leftarrow \boldsymbol{u}/h_{2,1} = \frac{1}{h_{2,1}}\frac{1}{\sqrt{C_f(0)}} P\boldsymbol{f}$

While it is no longer necessary to multiply vectors in $\mathbb{R}^N$ when utilizing autocovariance, converting the steps of Algorithm (3.1) to equivalent calculations utilizing autocovariance can become cumbersome. In the Modified Arnoldi Iteration we propose, the aforementioned calculations are systematized.

Let X be the $(n+1)\times(n+1)$ matrix with column i containing the power iterates coefficients for $\boldsymbol{q}_i$. Let $\boldsymbol{y}$ be the $(n+1)\times 1$ column vector containing the power iterates coefficients for $\boldsymbol{u}$. Let $\hat{B}$ be the sample autocovariance matrix given by

$$\widehat{B} = \begin{bmatrix} \widehat{C}_f(0) & \widehat{C}_f(1) & \cdots & \widehat{C}_f(n) \\ \widehat{C}_f(1) & \widehat{C}_f(2) & \cdots & \widehat{C}_f(n+1) \\ \vdots & \vdots & \cdots & \vdots \\ \widehat{C}_f(n) & \widehat{C}_f(n+1) & \cdots & \widehat{C}_f(2n) \end{bmatrix}. \tag{3.5}$$

We will use the notation $X(:,i)$ to denote the ith column of X. Algorithm (3.2) is the Modified Arnoldi iteration. Upon constructing H and removing the last row to obtain H_n, solving for the n eigenvalues of H_n gives n values from $\mathbb{C}$, one of which we will select as our approximation to λ_*. To select the appropriate candidate, we:

- Discard the eigenvalues having magnitude greater than 1.
- Discard ill-conditioned eigenvalues [6]. The condition number of a eigenvalue is a measure of how sensitive the eigenvalue is to perturbations in the matrix [16], and is equal to

  $$\text{Cond}(\lambda) = \frac{\|\boldsymbol{u}_{H_n}\|\|\boldsymbol{v}_{H_n}\|}{|\langle \boldsymbol{u}_{H_n}, \boldsymbol{v}_{H_n}\rangle|},$$

  where $\boldsymbol{u}_{H_n}$ and $\boldsymbol{v}_{H_n}$ are the left and right eigenvectors of H_n corresponding to λ. A perfectly conditioned eigenvalue has a condition number of 1, and the

Algorithm 3.2 Modified Arnoldi Iteration

Input:

Number of dimensions n

Sample autocovariance matrix $\widehat{B}$

Output:

$(n+1) \times n$ matrix H

$H \leftarrow (n+1) \times n$ zero matrix
$X \leftarrow (n+1) \times (n+1)$ zero matrix
$X(1,1) \leftarrow \frac{1}{\sqrt{\widehat{C}_f(0)}}$
for $k = 1, \ldots, n$ **do**
    $\boldsymbol{y} \leftarrow \text{shift}(X(:,k))$
    for $j = 1, \ldots, k$ **do**
        $h_{j,k} \leftarrow \langle \boldsymbol{y}, \widehat{B} X(:,j) \rangle$
        $\boldsymbol{y} \leftarrow \boldsymbol{y} - h_{j,k} X(:,j)$
    end for
    $h_{k+1,k} \leftarrow \sqrt{\langle \boldsymbol{y}, \widehat{B}\boldsymbol{y} \rangle}$
    $X(:,k+1) \leftarrow \boldsymbol{y}/h_{k+1,k}$
end for

condition number goes to infinity as $\boldsymbol{u}_{H_n}$ and $\boldsymbol{v}_{H_n}$ become orthogonal. We will discard all eigenvalues having a condition number greater than 2.

- Select the largest magnitude from the remaining candidates as the approximation to λ_*. This selection rule is justified by invoking Rayleigh's theorem [3]:

  Theorem 3.1.1 (Rayleigh's Theorem)**.** *If P is the transition matrix for a reversible Markov chain with equilibrium distribution π, then*

$$\lambda_* = \max_{\boldsymbol{f} \in \ell^2(\pi)} \left| \frac{\langle \boldsymbol{f}, P\boldsymbol{f} \rangle_\pi}{\langle \boldsymbol{f}, \boldsymbol{f} \rangle_\pi} \right| \quad s.t.\ \mathbb{E}_\pi \boldsymbol{f} = 0. \tag{3.6}$$

  The eigenvalue estimates can be interpreted as the following estimates:

$$\tilde{\lambda} \approx \frac{\langle \tilde{\boldsymbol{v}}, P\tilde{\boldsymbol{v}} \rangle_\pi}{\langle \tilde{\boldsymbol{v}}, \tilde{\boldsymbol{v}} \rangle_\pi}$$

  In addition, assuming $\boldsymbol{f}$ has zero mean has the effect of making the power

iterates orthogonal to the dominant right eigenvector, making $K_n[\boldsymbol{f}]$ orthogonal to $\mathbf{1}$:

$$\begin{aligned}\langle P^k \boldsymbol{f}, \mathbf{1}\rangle_\pi &= \langle \boldsymbol{f}, P^k \mathbf{1}\rangle_\pi \\ &= \langle \boldsymbol{f}, \mathbf{1}\rangle_\pi \\ &= 0.\end{aligned}$$

Using the selection rules above, it is possible for all candidates to be discarded. In such cases, we output no estimate and instead use a different set of samples to estimate λ_*. The use of multiple sets of sample runs will be discussed in Section 3.1.2.

Although we have assumed $\boldsymbol{f}$ to have zero mean, in practice, it is not necessary to subtract the mean from $\boldsymbol{f}$. The equation for autocovariance is

$$C_f(s) = \langle \boldsymbol{f}, P^s \boldsymbol{f}\rangle_\pi - \langle \boldsymbol{f}\rangle_\pi^2.$$

Evaluating the autocovariance for $\boldsymbol{f}$ and $\boldsymbol{f}' = \boldsymbol{f} - \mathbb{E}_\pi \boldsymbol{f}$ will show $C_f(s) = C_{f'}(s)$, and $\widehat{B}$ will remain unchanged.

How should n, the number of Krylov subspace dimensions, be chosen? If the actual autocovariance values are known, there is no harm in choosing a large value for n — the only price to be paid is in the additional computations needed to form H_n. However when sample autocovariance values are used, incrementing n comes with the risk of introducing errors. Each sample autocovariance estimate can be seen as the sum of the actual autocovariance and a error term:

$$\widehat{C}_f(s) = C_f(s) + \varepsilon_s$$

Recall that the inner product of two vectors in $K_n[\boldsymbol{f}]$ is a linear combination of $2n-1$ autocovariance values:

$$\langle \boldsymbol{x}, \boldsymbol{y}\rangle = \sum_{i=1,j=1}^{n} \xi_i \phi_j C_f(i+j-2)$$

As n increases, the estimates for inner products collect more error terms, affecting the estimates for $h_{j,k}$. It is therefore preferable to keep n as small as possible when working with sample autocovariances. We introduce next the lag parameter r, which will allow us to generate eigenvalue estimates using modest value for n.

3.1.1 Lags

Subsampling is a technique used in Monte Carlo Markov Chain (MCMC) calculations to reduce the autocorrelation between samples [15], where autocorrelation is defined as

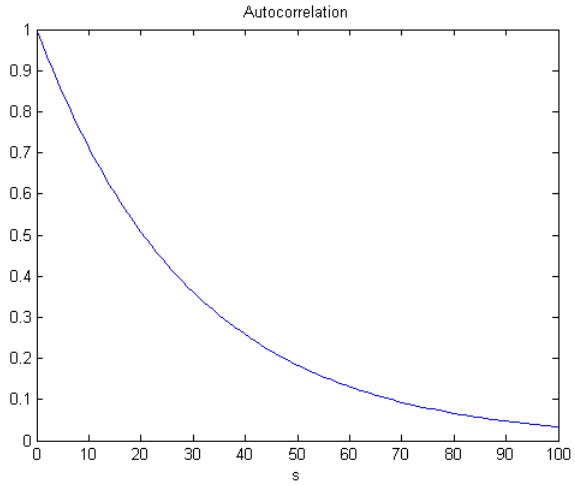
$$\rho(s) = C_f(s)/C_f(0).$$

The use of subsampling in the estimation of eigenvalues of transition probability matrices is discussed in [6], and we adapt the technique here. Instead of sampling the Markov chain at every step, consider instead sampling the Markov chain at every rth step, where $r \geq 1$. Equivalently, we sample from the transition probability matrix P^r instead of P. In a typical setting, the autocorrelation decreases exponentially to zero. Sampling the Markov chain at every r steps results in a autocorrelation function that decays faster — Figure 3-1 shows the plots of the autocorrelation of a test Markov chain using lags $r = 1, 2$, and 20. The use of the lag parameter therefore can be seen as having the effect of "squeezing" the autocorrelation to the left.

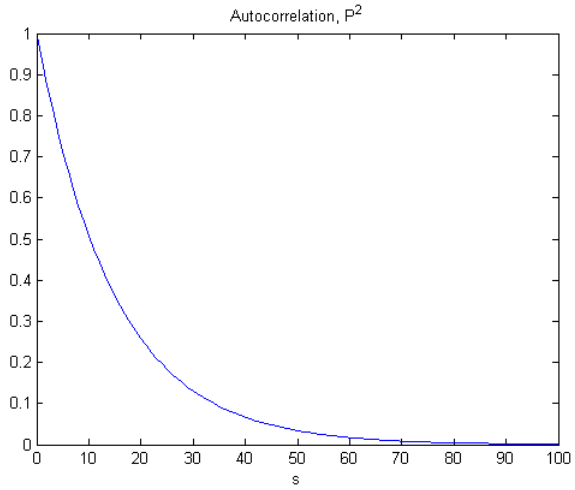
Having introduced the lag parameter r, we now give a interpretation of how r works in conjunction with the number of Krylov subspace dimensions n in the estimation of λ_*. Let $\tilde{\lambda}$ be the best approximation to λ_* using vectors in $K_n[\boldsymbol{f}]$, and $\tilde{\boldsymbol{v}}$ be the corresponding eigenvector estimate. Then $\tilde{\lambda}$ can be written as

$$\begin{aligned}\tilde{\lambda} &= \frac{\langle \tilde{\boldsymbol{v}}, P\tilde{\boldsymbol{v}} \rangle_\pi}{\langle \tilde{\boldsymbol{v}}, \tilde{\boldsymbol{v}} \rangle_\pi} \\ &= \frac{b_0 C_f(1) + b_1 C_f(2) + \cdots + b_{2n-2} C_f(2n-1)}{b_0 C_f(0) + b_1 C_f(1) + \cdots + b_{2n-2} C_f(2n-2)} \\ &= \frac{b_0 \rho(1) + b_1 \rho(2) + \cdots + b_{2n-2} \rho(2n-1)}{b_0 \rho(0) + b_1 \rho(1) + \cdots + b_{2n-2} \rho(2n-2)}.\end{aligned} \tag{3.7}$$

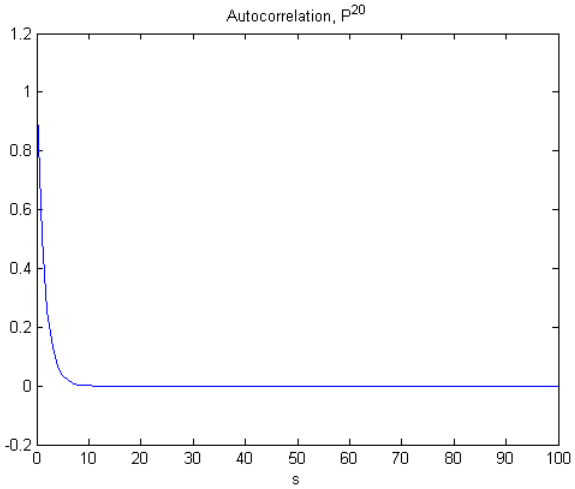
From Equation (3.7), we see that the choice for n can be interpreted as the choice



(a) $r = 1$



(b) $r = 2$



(c) $r = 20$

Figure 3-1: Plots of $\rho(s)$ for different values of s of a test Markov chain using lags $r = 1, 2$, and 20.

for the number of autocorrelation terms used to approximate λ_*. Let us call the set $[0, 1, \ldots, 2n-1]$ the *evaluation region* — if s is inside the evaluation region, then $\rho(s)$ is used in the approximation in Equation (3.7). For a fixed n, increasing r has the effect of "squeezing" the autocorrelation into the evaluation region. We demonstrate with a example.

Let $n = 7$. The evaluation region is $[0, 1, \ldots, 13]$. If f has the autocorrelation in Figure 3-1(a) or 3-1(b), which corresponds to using lags $r = 1$ or $r = 2$, then $\rho(s)$ has nonzero values for $s > 13$, so the evaluation region does not capture the entire nonzero content of $\rho(s)$. As a result, we can expect our approximation to change if we increment n. However, if f has the autocorrelation in Figure 3-1(c), which corresponds to using lag $r = 20$, then $\rho(s) \approx 0$ for $s > 10$, so the evaluation region captures the entire nonzero content of $\rho(s)$, and everything outside of the evaluation region is negligible. We can expect our approximation to change negligibly by increasing n, as doing so would amount to adding negligibly small terms to the numerator and denominator of Equation (3.7). Increasing the r parameter therefore has the effect of "squeezing" the nonzero content of $\rho(s)$ into the evaluation region.

What happens as $r \to \infty$? As $s \to \infty$, $C_f(s) \to 0$. However, the same is not the case for $\widehat{C}_f(s)$, which takes on random values. The difference in magnitude between $C_f(s)$ and $\widehat{C}_f(s)$ as $s \to \infty$ can be significant [6][17], so increasing r to too large a value comes with the risk of introducing errors in the calculation of H. Estimating λ_* therefore includes the task of finding a good choice for n and r. In Chapter 4, we will experiment using lags between $r = 1$ and $r = 50$.

The sample covariance matrix resulting from the use of lag r is

$$\widehat{B}_r = \begin{bmatrix} \widehat{C}_f(0) & \widehat{C}_f(r) & \cdots & \widehat{C}_f(nr) \\ \widehat{C}_f(r) & \widehat{C}_f(2r) & \cdots & \widehat{C}_f((n+1)r) \\ \vdots & \vdots & \cdots & \vdots \\ \widehat{C}_f(nr) & \widehat{C}_f((n+1)r) & \cdots & \widehat{C}_f(2nr) \end{bmatrix}. \tag{3.8}$$

Let α be the Modified Arnoldi estimate obtained using $\widehat{B}_r$. We will use $\hat{\lambda}_*$ to denote

our approximation to λ_*, which is

$$\hat{\lambda}_* = \alpha^{1/r}.$$

Figure 3-2 illustrates the Modified Arnoldi calculations when evaluating a Markov chain with a slowly decaying autocorrelation (actual autocovariance values were used - not sample estimates). The choice of $r = 20$ produces accurate estimates with $n = 6$, while the choice of $r = 1$ will require a larger value for n to achieve accurate estimates. Algorithm 3.3 gives the steps to calculating $\hat{\lambda}_*$.

Algorithm 3.3 $\hat{\lambda}_*$ Calculation Procedure

Input:

Samples $X_0, \ldots, X_{T-1}$

Choice for f, n, and r

Output:

$\hat{\lambda}_*$: estimate to λ_*

1: Form the sample covariance matrix in Equation (3.8) using the estimator

$$\widehat{C}_f(s) = \frac{1}{T-|s|} \sum_{t=0}^{T-|s|+1} \left(f(X_t) - \bar{f}\right)\left(f(X_{t+s}) - \bar{f}\right)$$

2: Run Algorithm 3.2 to generate the $(n+1) \times n$ matrix H

3: Truncate the last row of H to obtain H_n

4: Solve for the eigenvalues of H_n - these are candidate estimates.

5: Discard the candidates that have magnitude greater than 1.

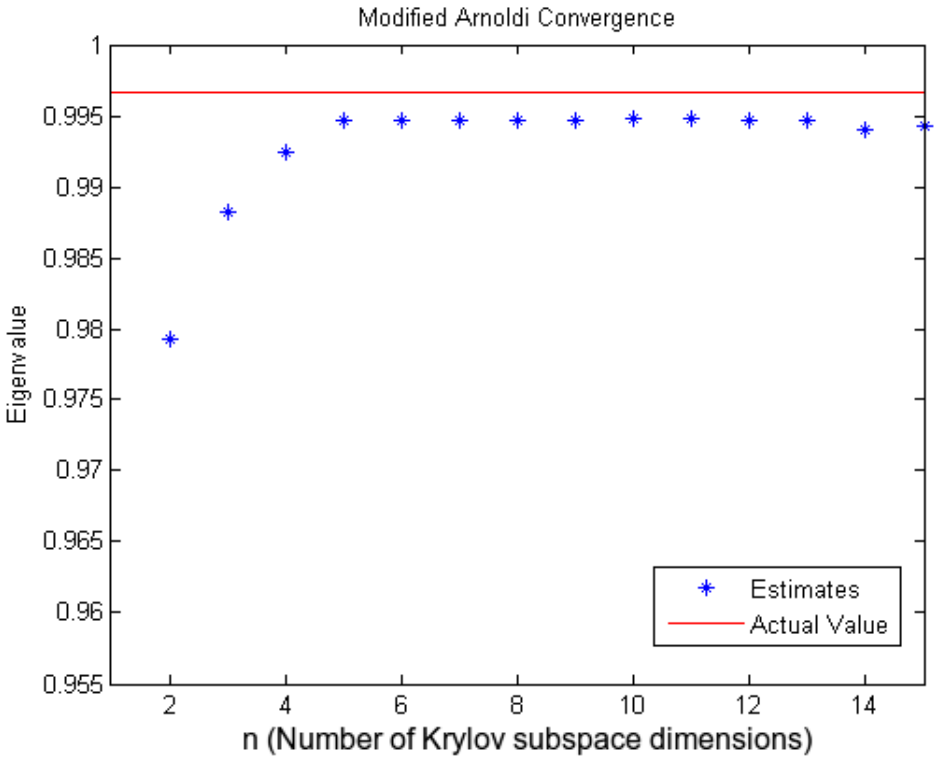
6: Discard the candidates that have a condition number greater than 2.

7: $\alpha \leftarrow$ Maximum magnitude over the remaining candidates.

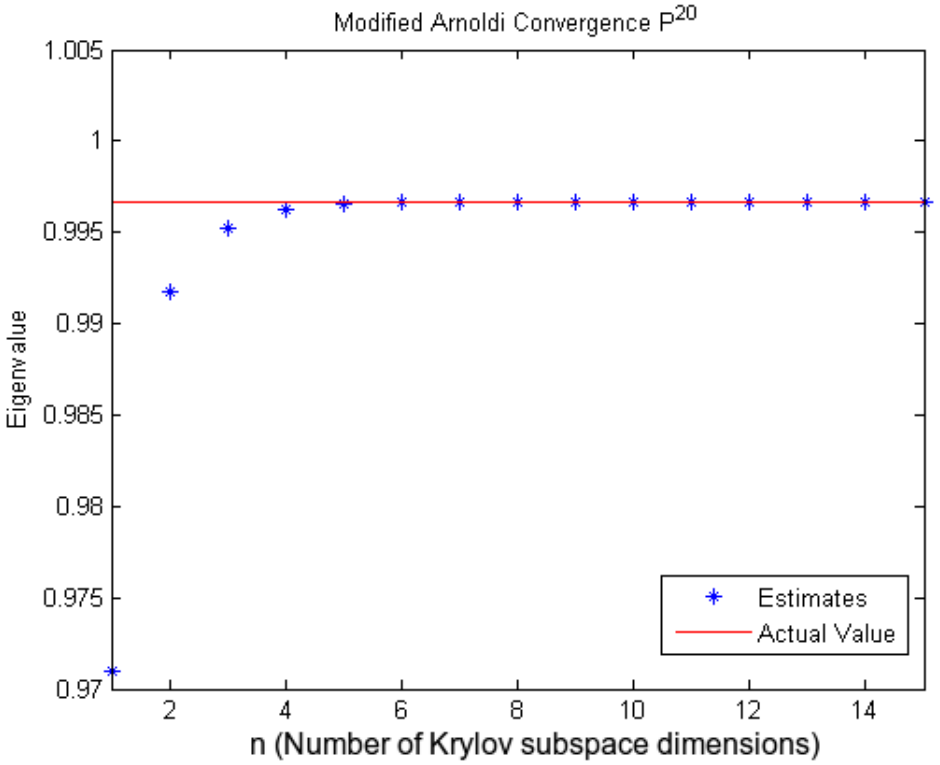
8: $\hat{\lambda}_* \leftarrow \alpha^{1/r}$

3.1.2 Batch Estimates

Since different sets of samples will yield different estimates for λ_*, an estimate aggregated over multiple batches of samples is preferable to one generated from a single batch. Figure 3-3 illustrates a hypothetical situation where 10 batches of samples were used to estimate λ_*. Six of the estimates come close to the actual answer, two batches resulted in overestimations, and two batches failed to produce a result. The



(a) $r = 1$



(b) $r = 20$

Figure 3-2: Modified Arnoldi estimates using $r = 1$ and $r = 20$. Actual autocovariance values were used.

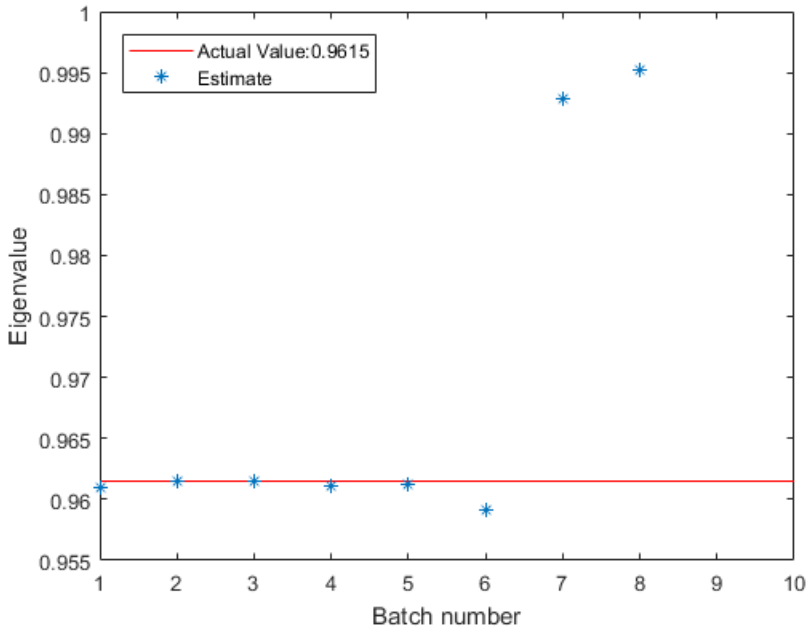


Figure 3-3: Estimates of λ_* generated from 10 sample runs

overestimates are the result of the errors in the sample autocovariance — we will refer to the overestimates as "noisy" eigenvalues. Since the noisy eigenvalues will skew the sample mean, the median may be closer to the actual eigenvalue. In Chapter 4, we will report both the sample mean and median.

Chapter 4

Results

This chapter gives the results produced by Modified Arnoldi on the following reversible Markov Chains:

- Ehrenfest Urn Model
- Scaled SIS Process

4.1 Ehrenfest Urn Model

For the continuous time Ehrenfest urn process,the eigenvalues and right eigenvectors of the transition probability matrix are given in [10]. We will use the discrete time embedded Markov chain given in [6]. The embedded Markov chain is constructed as follows: Start with M balls split between urn 1 and urn 2. At each time step, select a ball from one of the urns and place it in urn 1 with probability p, and in urn 2 with probability $1-p$. Let i be the number of balls in urn 1. The transition probabilities for the Ehrenfest urn model are

$$P(i,j) = \begin{cases} \frac{M-i}{M}p & \text{if } j = i+1 \\ \frac{i}{M}(1-p) & \text{if } j = i-1 \\ 1 - \frac{M-i}{M}p - \frac{i}{M}(1-p) & \text{if } j = i \\ 0 & \text{otherwise.} \end{cases}$$

The second largest eigenvalue magnitude is $\lambda_* = 1 - \frac{1}{M}$.

4.1.1 Results

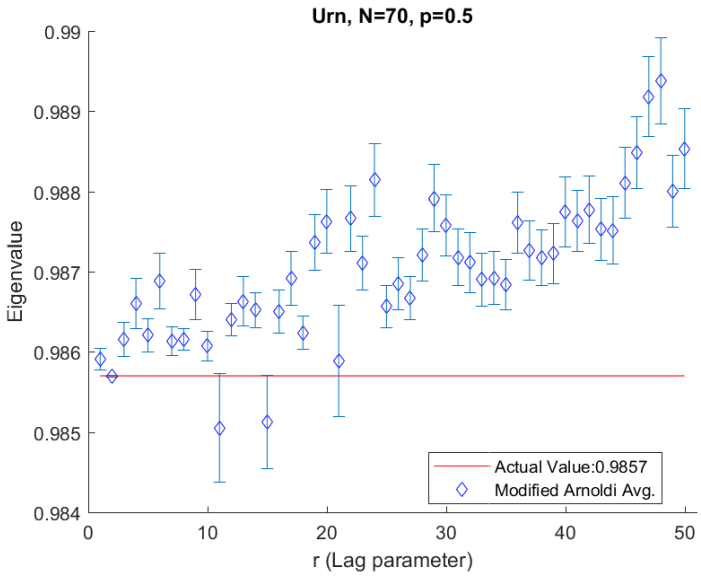
The following procedure was used to estimate λ_* for $M = 70, p = 0.5$:

- Generate 100 batches of $T = 10^7$ samples $X_0, X_1, \ldots, X_{T-1}$, where X_0 is recorded after a burn in period of 50,000 samples following a random starting position.
- Apply the observable $f(X_t) = i$, the number of balls in urn 1 for state X_t.
- Execute Algorithm 3.3 using $n = 5$ and varying r from 1 to 50. For each value of r, calculate the average and median estimates.

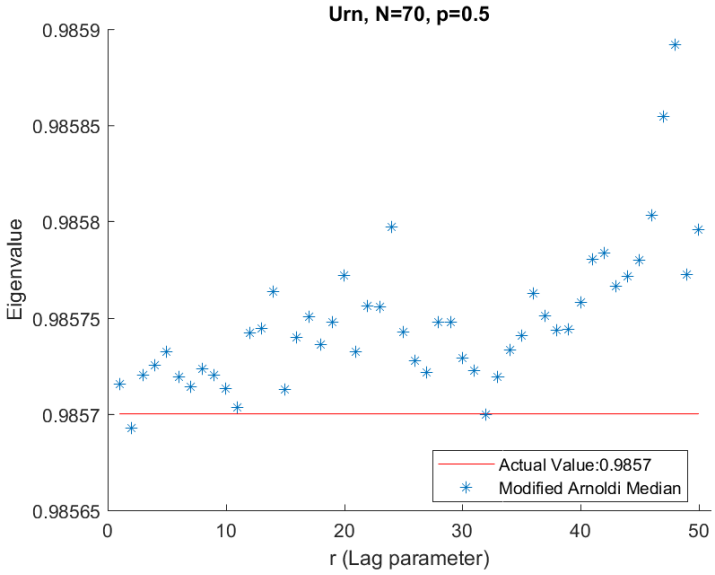
Figure 4-1 shows the results. Figure 4-1(a) is generated using the calculation $\hat{\lambda}_{*b} \pm \sqrt{\frac{\widehat{\text{var}}\ \hat{\lambda}_*}{L}}$, where $\hat{\lambda}_{*b}$ is the average estimate of λ_*, and L is the sample size. The results show that the average estimate is skewed by "noisy" eigenvalues, but the median is more robust to the overestimates and closer to the actual value.

4.2 Scaled SIS process

The scaled SIS process describes the spread of a virus on a M node network characterized by the adjacency matrix A. The continuous time Markov chain is given in [19]. We will use the embedded discrete time Markov chain given in Appendix A.2. The parameters ζ, γ describe the rates at which a susceptible node becomes infected, and the parameter μ describes the rate at which a infected node becomes healed. The state of the network is given by the length M column vector $\boldsymbol{x}$, where $x_k = 1$ if node k is infected, and $x_k = 0$ if node k is susceptible. The size of the state space is 2^M. At each time increment, either a single node changes its state or no change occurs in the network. Using $\boldsymbol{x}$ to denote the network state before a time step, and $\boldsymbol{x}'$ to denote the network state after the time step, the state transitions are illustrated as follows:



(a) Average



(b) Median

Figure 4-1: Modified Arnoldi estimates for the Ehrenfest Urn Model.

- Susceptible node k becomes infected:

$$\boldsymbol{x} = \Big[x_1, x_2, \ldots, x_k = 0, \ldots, x_M\Big]^T$$
$$\boldsymbol{x}' = \Big[x_1, x_2, \ldots, x_k = 1, \ldots, x_M\Big]^T$$

- Infected node k becomes healed:

$$\boldsymbol{x} = \Big[x_1, x_2, \ldots, x_k = 1, \ldots, x_M\Big]^T$$
$$\boldsymbol{x}' = \Big[x_1, x_2, \ldots, x_k = 0, \ldots, x_M\Big]^T$$

- No change in the network:

$$\boldsymbol{x} = \boldsymbol{x}'$$

Let d_k be the number of infected neighbors to node k, which can be calculated by multiplying the kth row of A to $\boldsymbol{x}$:

$$d_k = A\left(k, :\right)\boldsymbol{x}$$

The transition probabilities of the Markov chain are

$$P\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \begin{cases} \frac{1}{\alpha}\zeta\gamma^{d_k} & \text{if susceptible node } k \text{ turns infected} \\ \frac{1}{\alpha}\mu & \text{if infected node } k \text{ becomes healed} \\ 1 - \left[\sum\limits_{\text{infected nodes}} \frac{1}{\alpha}\mu + \sum\limits_{\text{susceptible nodes}} \frac{1}{\alpha}\zeta\gamma^{d_k}\right] & \text{if } \boldsymbol{x} = \boldsymbol{x}' \\ 0 & \text{otherwise.} \end{cases}$$

where α is the normalizing constant given by

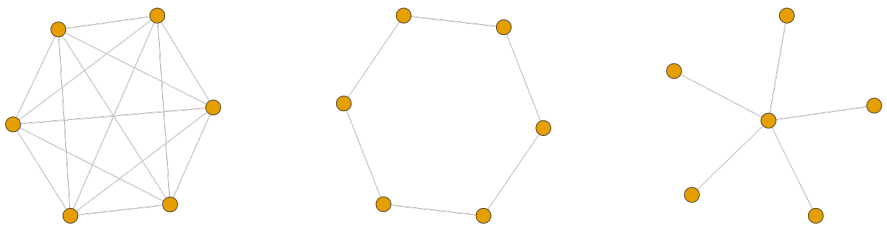
$$\alpha = \begin{cases} M\left(\zeta + \mu\right) & \text{if } \gamma \leq 1 \\ \zeta\left(\mathbf{1}, \gamma^{A\mathbf{1}}\right) + M\mu & \text{if } \gamma > 1. \end{cases}$$

4.2.1 Results

We will investigate the Scaled SIS process on structured graphs. The observables we will use are:

- **Number of Infected Nodes.** This number can be calculated by evaluating $f(\boldsymbol{x}) = \langle \mathbf{1}, \boldsymbol{x} \rangle$.
- **Number of Infected Edges.** An *infected edge* is a edge connecting two infected nodes. The number of infected edges can be calculated by evaluating $f(\boldsymbol{x}) = \frac{1}{2}\langle \boldsymbol{x}, A\boldsymbol{x} \rangle$.

4.2.2 Structured Graphs - Complete, Ring, and Star Graphs



(a) Complete (b) Ring (c) Star

Figure 4-2: 6 node Complete, Ring, and Star graphs

This section is the results of our experiments using Complete, Ring, and Star graphs comprised of 8 or 14 nodes, and Scaled SIS parameters $\zeta = 2, \gamma = 2, \mu = 10$. The following procedure was used to estimate λ_* for each graph:

1. Generate 100 batches of $T = 10^7$ samples $X_0, X_1, \ldots, X_{T-1}$, where X_0 is recorded after a burn in period of 50,000 samples following a random starting position.
2. Apply the observable $f(X_t)$.
3. Execute Algorithm 3.3 using $n = 5$ and varying r from 1 to 50. For each value of r, calculate the average and median estimates.

Complete graph, 8 nodes

The size of the state space for the Scaled SIS process on 8 node graphs is 2^8. Figure 4-3 shows the Modified Arnoldi iteration estimates using $\boldsymbol{f}$ = number of infected nodes. To illustrate the effect of using different observables, Figure 4-4 gives the results using $\boldsymbol{f}$ = number of infected edges, and Figure 4-5 gives the results using $\boldsymbol{f} = \boldsymbol{v}_*$, the right eigenvector corresponding to λ_*. The plots show $\boldsymbol{v}_*$ as the observable giving the best estimates.

How should one choose $\boldsymbol{f}$? Let $\boldsymbol{v}_*, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ be the right eigenvectors of P. It is noted in [6] that optimal observables have the form

$$\boldsymbol{f} = a_1\boldsymbol{v}_1 + a_*\boldsymbol{v}_* + a_3\boldsymbol{v}_3 + \cdots + a_m\boldsymbol{v}_m, \tag{4.1}$$

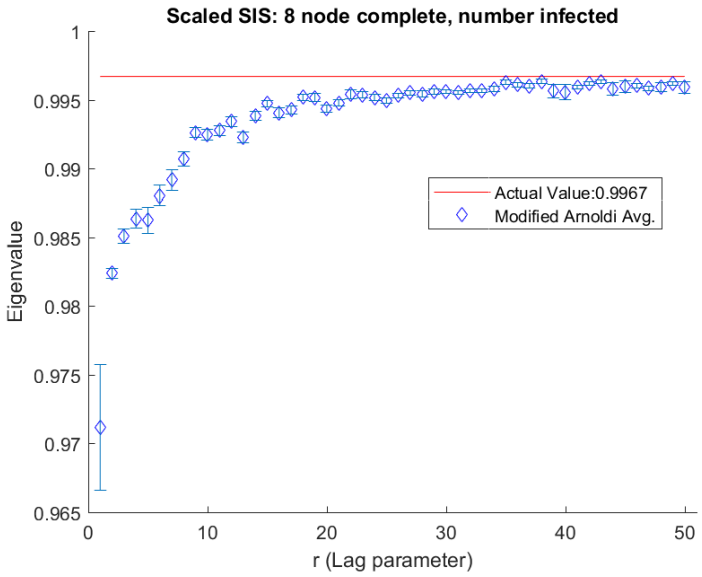
where $m \ll N$, and a_* has the maximum magnitude over $a_*, a_3, a_4, \ldots, a_m$. We were unable to find optimal observables for the Scaled SIS process — finding optimal observables for use with Modified Arnoldi is left as a topic for future research.

Ring and Star graphs, 8 nodes

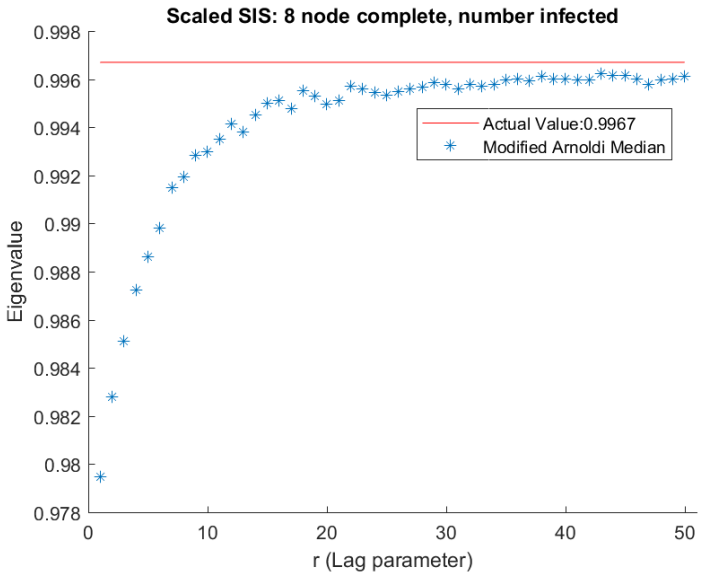
Figures 4-6 and 4-7 gives the results for the Scaled SIS process on the 8 node ring and star graphs, using the observable $\boldsymbol{f}$ = number of infected nodes. The average estimates are skewed by "noisy" eigenvalues, leading the median to be closer to the actual value.

Ring and Star graphs, 14 nodes

We now turn our attention to Markov chains with larger state spaces. The size of the state space for the Scaled SIS process on 14 node graphs is 2^{14}. Figures 4-8 and 4-9 gives the results for the Scaled SIS process on 14 node ring and star graphs, using the observable $\boldsymbol{f}$ = number of infected nodes. For both Markov chains, the median is close to the actual value, while the average estimates for the 14 node ring are erratic due to skewing by "noisy" eigenvalues.

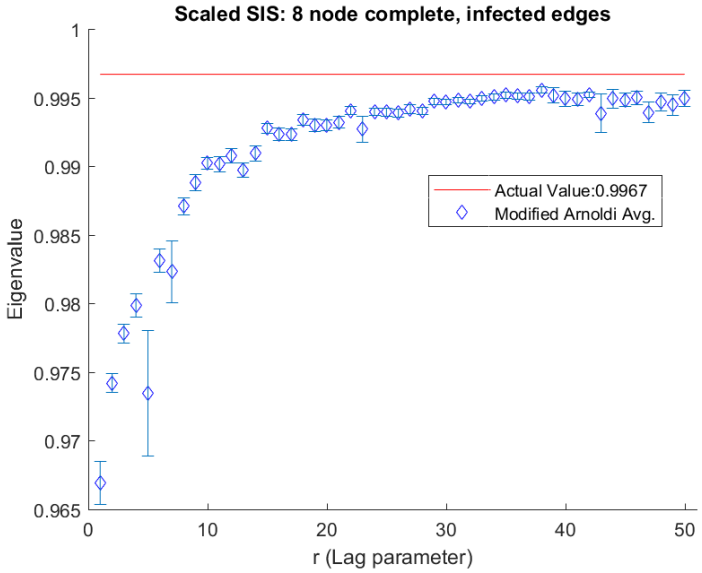


(a) Average

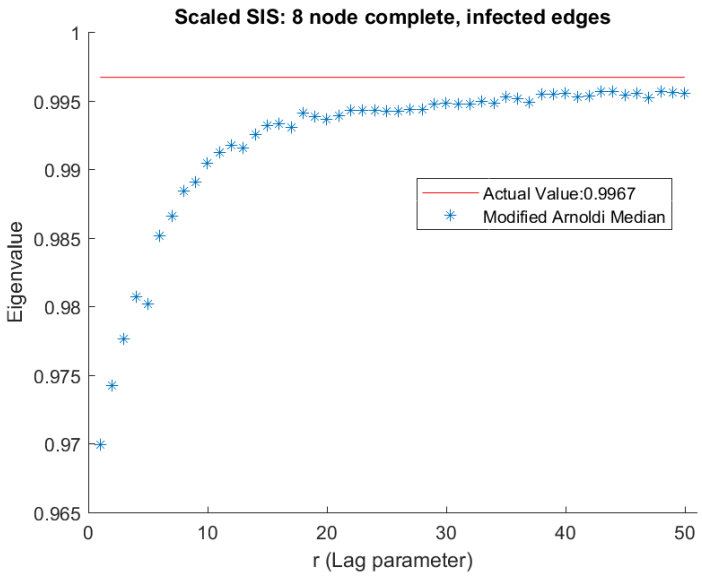


(b) Median

Figure 4-3: Scaled SIS on 8 node Complete graph. Results using $f(\boldsymbol{x}) = \langle \mathbf{1}, \boldsymbol{x} \rangle$

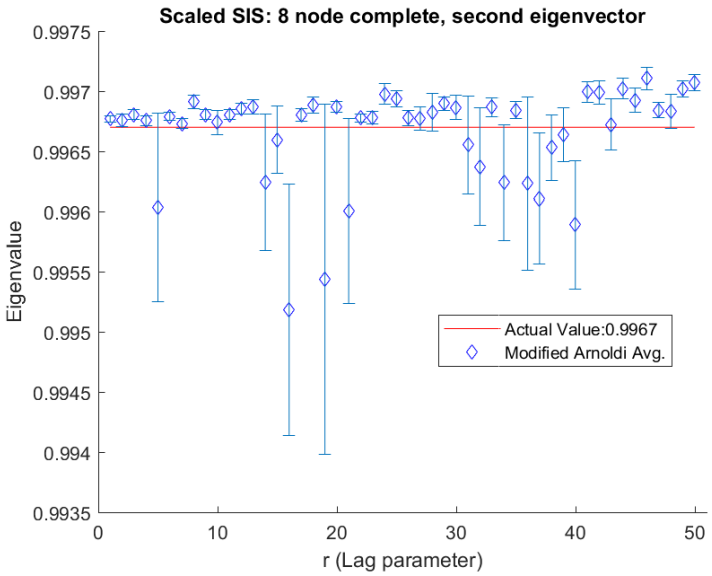


(a) Average

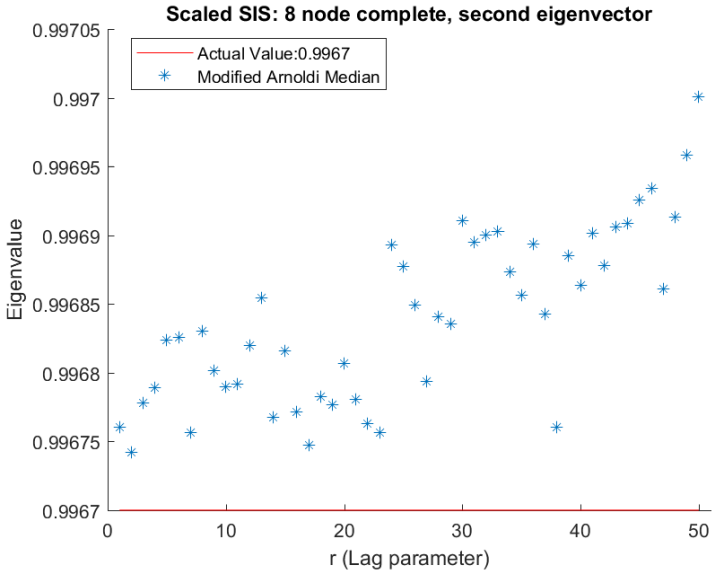


(b) Median

Figure 4-4: Scaled SIS on 8 node Complete graph. Results using $f(\boldsymbol{x}) = \frac{1}{2}\langle \boldsymbol{x}, A\boldsymbol{x} \rangle$

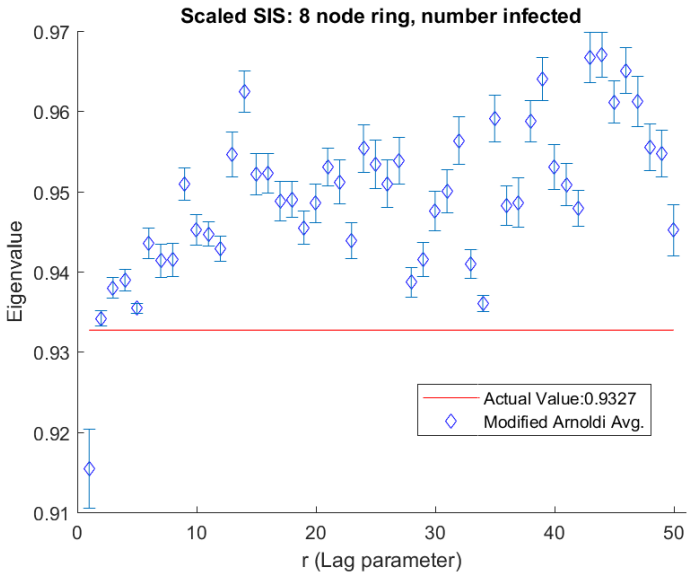


(a) Average

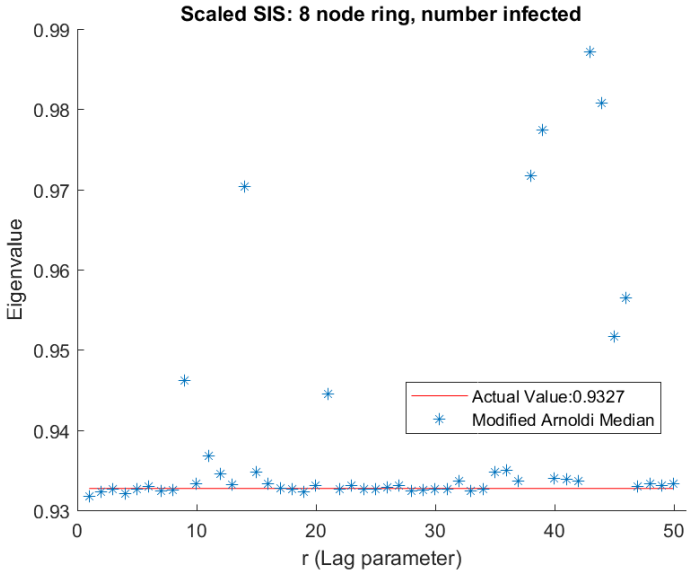


(b) Median

Figure 4-5: Scaled SIS on 8 node Complete graph. Results using $\boldsymbol{f} = \boldsymbol{v}_*$

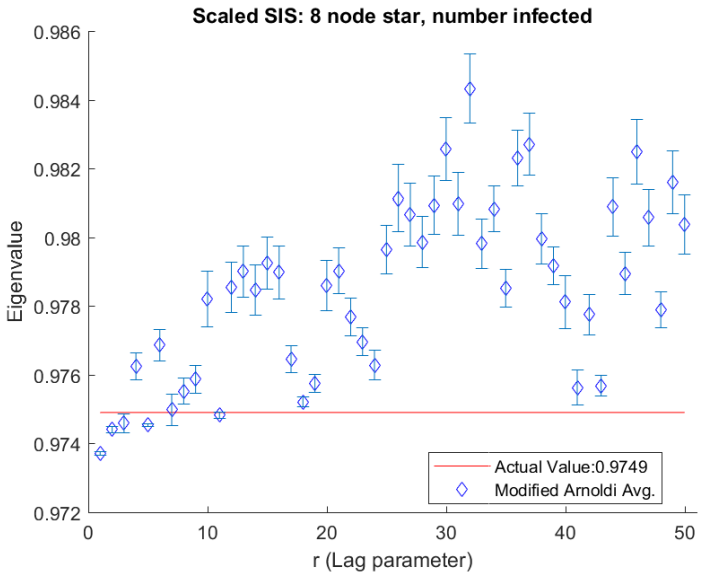


(a) Average

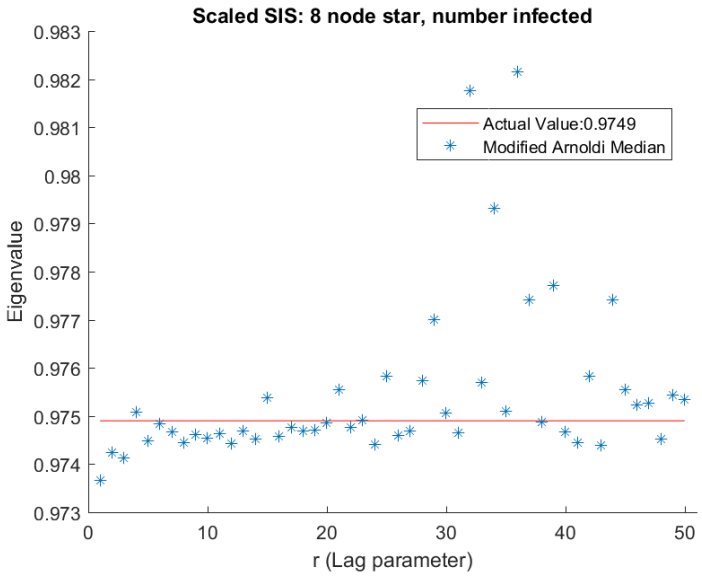


(b) Median

Figure 4-6: Scaled SIS on 8 node Ring. Results using $f(\boldsymbol{x}) = \langle \mathbf{1}, \boldsymbol{x} \rangle$

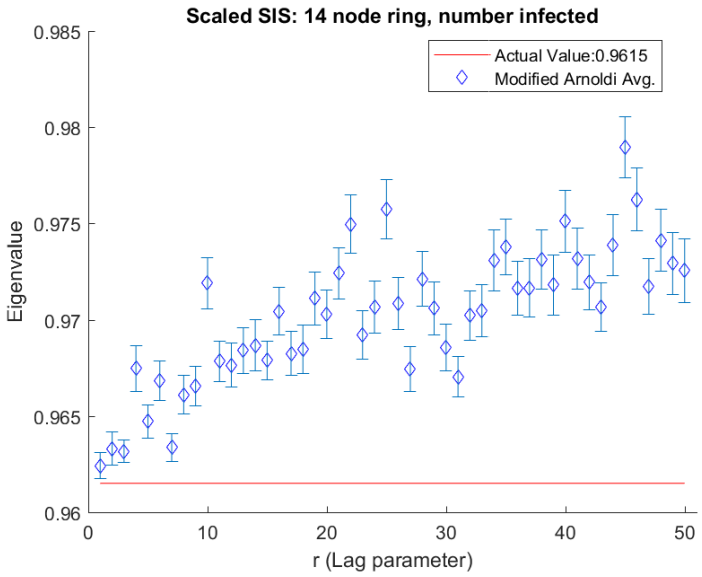


(a) Average

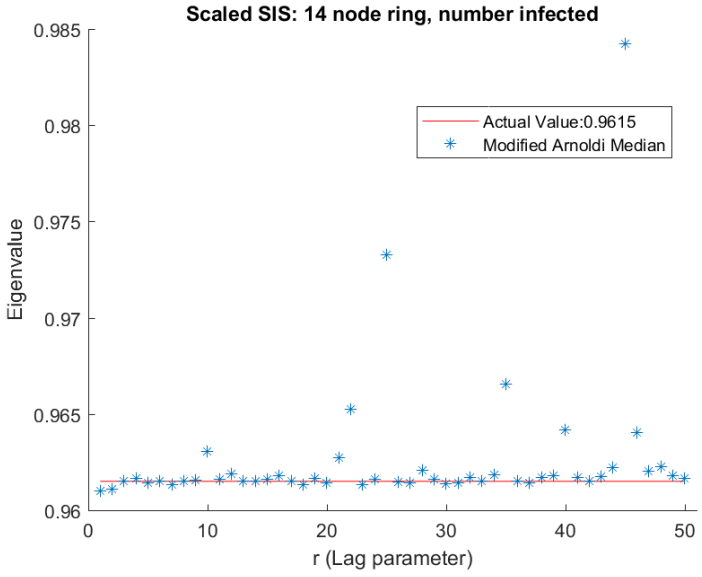


(b) Median

Figure 4-7: Scaled SIS on 8 node Star. Results using $f(\boldsymbol{x}) = \langle \mathbf{1}, \boldsymbol{x} \rangle$

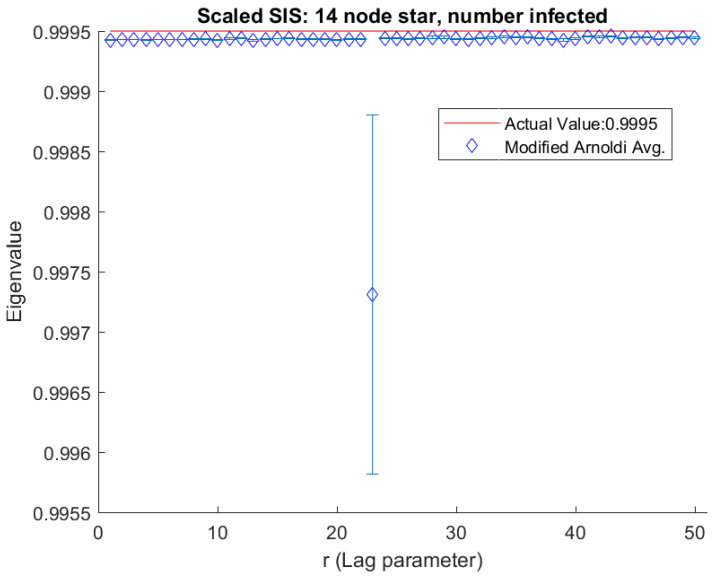


(a) Average

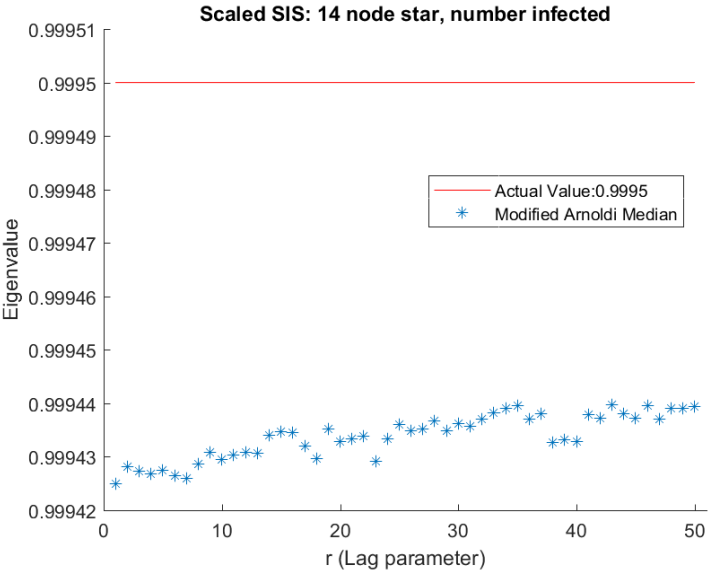


(b) Median

Figure 4-8: Scaled SIS on 14 node Ring. Results using $f(\boldsymbol{x}) = \langle \mathbf{1}, \boldsymbol{x} \rangle$



(a) Average



(b) Median

Figure 4-9: Scaled SIS on 14 node Star. Results using $f(\boldsymbol{x}) = \langle \mathbf{1}, \boldsymbol{x} \rangle$

Chapter 5

Conclusion

We demonstrated the modification of the Arnoldi iteration for the purpose of estimating the second largest eigenvalue magnitude of reversible Markov chains. To conclude this thesis, we outline additional topics that may strengthen the utility of the Modified Arnoldi iteration.

- **Robust Estimates.** The results show that errors in the sample autocorrelation may lead to overestimates of λ_*. As a result, the sample median was closer to the true value than the average in most instances. We would like to investigate methods to detect and remove the "noisy" eigenvalues that arise as a result of poor sample autocorrelation estimates.

- **Incorporating estimates of the integrated autocorrelation time.** For large s, the following upper bound on $C_f(s)$ is given in [17]:

$$C_f(s) \leq \kappa \exp\left(-\frac{s}{\tau_{exp}}\right)$$

where κ is some constant and τ_{exp} is the *exponential autocorrelation time:*

$$\tau_{exp} = \frac{1}{\ln \lambda_*}$$

The integrated autocorrelation time is given as

$$\tau_{int} = \sum_{s=0}^{\infty} \frac{C_f(s)}{C_f(0)}.$$

For $\tau_{exp} \gg 1$, τ_{exp} can be approximated by

$$\tau_{exp} \approx \frac{1}{2}\tau_{int}$$

There are various methods of estimating τ_{int} - a good estimate can be used to inform our choice for n and r. For example, the following value for r is recommended in [6]:

$$r = \left\lfloor \frac{4\tau_{int}}{2n-1} \right\rfloor \tag{5.1}$$

The integrated autocorrelation time may also lead to insights on how lengths of sample runs are related to estimation errors.

- **Which regions of χ produce good estimates?** Some batches produced better estimates than others. It is therefore worth investigating whether sampling from certain portions of the state space yield better eigenvalue estimates than others.

Appendix A

Background Information

A.1 The Krylov Subspace Pencil

We detail here the Krylov Subspace Pencil (KSP) approximation given by [6]. Let P be the $N \times N$ transition matrix of a reversible Markov chain. Let $\boldsymbol{f}$ be a observable in $\ell^2(\pi)$ with zero mean, and let $\boldsymbol{v}_i$ be the right eigenvectors of P. Suppose $\boldsymbol{f}$ has insignificant components along all but a few of the right eigenvectors, with the component largest in magnitude in the direction of $\boldsymbol{v}_2$, the eigenvector corresponding to λ_*.

$$\begin{aligned}\boldsymbol{f} &= \psi_1\boldsymbol{v}_1 + \psi_2\boldsymbol{v}_2 + \cdots + \psi_N\boldsymbol{v}_N \\ &\approx \psi_2\boldsymbol{v}_2 + \cdots \psi_m\boldsymbol{v}_m,\end{aligned}$$

where $m \ll N$. In such cases, using the Krylov Subspace allows us to estimate λ_* using the sample autocovariance of $\boldsymbol{f}$. The dimension n Krylov subspace generated by $\boldsymbol{f}$ is defined as

$$K_n[\boldsymbol{f}] = \text{span}\{\boldsymbol{f}, P\boldsymbol{f}, P^2\boldsymbol{f}, \ldots, P^{n-1}\boldsymbol{f}\}$$

Let $\boldsymbol{u}$ be any vector in $K_n[\boldsymbol{f}]$. Then for some $\xi_1, \ldots, \xi_n \in \mathbb{R}$,

$$\boldsymbol{u} = \sum_{j=1}^{n} \xi_j P^{j-1}\boldsymbol{f}. \tag{A.1}$$

The **variational quotient**[1] with $\boldsymbol{u}$ is defined as

$$\frac{\langle \boldsymbol{u}, P\boldsymbol{u} \rangle_\pi}{\langle \boldsymbol{u}, \boldsymbol{u} \rangle_\pi} \tag{A.2}$$

Using Equation (A.1) allows the variational quotient to be expanded to

$$\frac{\langle \boldsymbol{u}, P\boldsymbol{u} \rangle_\pi}{\langle \boldsymbol{u}, \boldsymbol{u} \rangle_\pi} = \frac{\sum\limits_{i,j} \xi_i \xi_j \langle P^{i-1}\boldsymbol{f}, P^j \boldsymbol{f} \rangle_\pi}{\sum\limits_{i,j} \xi_i \xi_j \langle P^{i-1}\boldsymbol{f}, P^{j-1} \boldsymbol{f} \rangle_\pi}. \tag{A.3}$$

Let A and B be the following $n \times n$ autocovariance matrices:

$$\begin{aligned} A &= \begin{bmatrix} C_f(1) & C_f(2) & \dots & C_f(n) \\ C_f(2) & C_f(3) & \dots & C_f(n+1) \\ \vdots & \vdots & \dots & \vdots \\ C_f(n) & C_f(n+1) & \dots & C_f(2n-1) \end{bmatrix} \\ B &= \begin{bmatrix} C_f(0) & C_f(1) & \dots & C_f(n-1) \\ C_f(1) & C_f(2) & \dots & C_f(n) \\ \vdots & \vdots & \dots & \vdots \\ C_f(n-1) & C_f(n) & \dots & C_f(2n-2) \end{bmatrix}. \end{aligned} \tag{A.4}$$

Equivalently,

$$\begin{aligned} A\,(i,j) &= C_f\,(i+j-1) \\ B\,(i,j) &= C_f\,(i+j-2)\,. \end{aligned}$$

The following result is from Gade 2007.

Theorem A.1.1. *The best approximation to λ_* using variational quotients formed by vectors in $K_n[\boldsymbol{f}]$ is the largest magnitude over the generalized eigenvalues satisfying*

[1] This quantity is known as the **Rayleigh quotient** for the case where P is Hermitian and Euclidean inner products are used.

the generalized eigenvalue problem

$$A\boldsymbol{\xi} = \lambda B\boldsymbol{\xi}. \tag{A.5}$$

Proof. Consider first the numerator of Equation (A.3). Since P is self adjoint with respect to π,

$$\begin{aligned}
\langle \boldsymbol{u}, P\boldsymbol{u}\rangle_\pi &= \sum_{i,j} \xi_i \xi_j \langle P^{i-1}\boldsymbol{f}, P^j \boldsymbol{f}\rangle_\pi \\
&= \sum_{i,j} \xi_i \xi_j \langle \boldsymbol{f}, P^{i+j-1}\boldsymbol{f}\rangle_\pi \\
&= \sum_{i,j} \xi_i \xi_j C_f\left(i+j-1\right).
\end{aligned}$$

Similarly, the denominator can be written as

$$\begin{aligned}
\langle \boldsymbol{u}, \boldsymbol{u}\rangle_\pi &= \sum_{i,j} \xi_i \xi_j \langle P^{i-1}\boldsymbol{f}, P^{j-1} \boldsymbol{f}\rangle_\pi \\
&= \sum_{i,j} \xi_i \xi_j \langle \boldsymbol{f}, P^{i+j-2}\boldsymbol{f}\rangle_\pi \\
&= \sum_{i,j} \xi_i \xi_j C_f\left(i+j-2\right).
\end{aligned}$$

Then the variational quotient becomes

$$\frac{\langle \boldsymbol{u}, P\boldsymbol{u}\rangle_\pi}{\langle \boldsymbol{u}, \boldsymbol{u}\rangle_\pi} = \frac{\langle \boldsymbol{\xi}, A\boldsymbol{\xi}\rangle}{\langle \boldsymbol{\xi}, B\boldsymbol{\xi}\rangle}. \tag{A.6}$$

From Rayleigh's theorem, the largest generalized eigenvalue satisfying Equation (A.5) is also the largest variational quotient in $K_n\left[\boldsymbol{f}\right]$ and the best estimate to λ_* using vectors in $K_n\left[\boldsymbol{f}\right]$. □

The problem given in Equation (A.5) is highly ill conditioned [6], and requires the Generalized Upper Triangle algorithm [4] to solve.

The Modified Arnoldi iteration differs from the KSP in that the Modified Arnoldi

iteration approximates the condition

$$P\tilde{\boldsymbol{v}} \approx \tilde{\lambda}\tilde{\boldsymbol{v}}, \tag{A.7}$$

while the KSP method maximizes the expression

$$\max_{\boldsymbol{u}\in K_n[\boldsymbol{f}]} \left| \frac{\langle \boldsymbol{u}, P\boldsymbol{u}\rangle_\pi}{\langle \boldsymbol{u}, \boldsymbol{u}\rangle_\pi} \right|. \tag{A.8}$$

The Modified Arnoldi iteration circumvents the step of solving a ill-conditioned problem.

A.2 The Scaled SIS Process

Given a M node network characterised by a undirected graph with adjacency matrix A, the status of each node is either 1 (infected) or 0 (susceptible). The network states are given as M bit column vectors, where the ith row indicates the status of node i. The evolution of the network is composed of state transitions involving either a single susceptible node turning infected, a single infected node turning susceptible, or all nodes retaining their current state. The rates corresponding to each transition are given as follows:

- **Healing**: The time it takes for infected agents to transition to the susceptible state is exponentially distributed with rate μ. The average time it takes to leave the infected state is then equal to $1/\mu$. We will refer to μ as the *healing rate*.

  Consider the case where infected node k becomes susceptible. The network configurations before and after the transition are

$$\boldsymbol{x} = \Big[x_1, \ldots, x_k = 1, \ldots, x_M\Big]^T$$
$$\boldsymbol{x}' = \Big[x_1, \ldots, x_k = 0, \ldots, x_M\Big]^T$$

The corresponding entry in the transition rate matrix is

$$Q\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \mu$$

- **Infection**: The time it takes for a susceptible agent to transition to the infected state is exponentially distributed with rate $\zeta\gamma^d$ where d is the number of infected neighbors adjacent to the susceptible agent. The average time it takes to leave the susceptible state is then equal to $\frac{1}{\zeta\gamma^d}$. This is in contrast to works using earlier forms of the SIS process, which utilize a infection rate of the form $\zeta + \gamma^d$. The name *scaled SIS* therefore emphasizes the choice of infection rate $\zeta\gamma^d$, which is more amenable to analysis.

  Infection in scaled SIS can be interpreted as the resultant of two components: *exogenous* infection and *endogenous* infection. Exogenous infection characterizes the scenario where a susceptible becomes infected due to a source external to the network, and occurs with rate ζ. Endogenous infection characterizes the scenario where a susceptible becomes infected due to influence from infected neighbors, and occurs with rate γ^d.

  Consider the case where susceptible node k becomes infected. The network configurations before and after the transition are

$$\boldsymbol{x} = \Big[x_1, \ldots, x_k = 0, \ldots, x_M\Big]^T$$
$$\boldsymbol{x}' = \Big[x_1, \ldots, x_k = 1, \ldots, x_M\Big]^T$$

  The corresponding entry in the transition rate matrix is

$$Q\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \zeta\gamma^{d_k}$$

  where d_k is the number of infected neighbors of node k, and can be calculated

by multiplying $\boldsymbol{x}$ by the kth row of A:

$$d_k = A(k,:)\boldsymbol{x}$$

- **More than one change in the network**: Since each network transition changes the status of only a single node, the entry in the transition rate matrix when $\boldsymbol{x}$ and $\boldsymbol{x}'$ differ by more than one bit is

$$\begin{aligned} Q(\boldsymbol{x}, \boldsymbol{x}') &= 0 \\ &= Q(\boldsymbol{x}', \boldsymbol{x}) \end{aligned}$$

- **No change in the network**: The corresponding entry in the transition rate matrix is

$$Q(\boldsymbol{x}, \boldsymbol{x}) = -\sum_{\boldsymbol{x} \neq \boldsymbol{x}'} Q(\boldsymbol{x}, \boldsymbol{x}')$$

The following result is from [19].

Theorem A.2.1. *The scaled SIS process is a reversible Markov process with the equilibrium distribution given by*

$$\pi(\boldsymbol{x}) = \frac{1}{Z} \left(\frac{\zeta}{\mu}\right)^{1^T \boldsymbol{x}} \gamma^{\frac{1}{2}\boldsymbol{x}^T A \boldsymbol{x}}, \qquad \forall \boldsymbol{x} \in \chi \tag{A.9}$$

where Z is the partition function given by

$$Z = \sum_{\boldsymbol{x} \in \chi} \left(\frac{\zeta}{\mu}\right)^{1^T \boldsymbol{x}} \gamma^{\frac{1}{2}\boldsymbol{x}^T A \boldsymbol{x}}$$

Since the state space grows exponentially with the number of nodes in the network, calculating Z is infeasible for many practical purposes. Therefore, it is assumed we have no knowledge of Z and can only know the entries of π up to a normalizing constant.

A.2.1 Scaled SIS Process Embedded Markov Chain

The eigenvalue algorithms discussed in this thesis apply only to discrete time Markov chains (DTMC). The scaled SIS process can be transformed to a DTMC by applying the following operation on Q:

$$\begin{aligned} P &= I + \frac{1}{\alpha}Q, \\ \alpha &\geq \max_i |Q(i,i)| \end{aligned} \tag{A.10}$$

The resulting Markov chain has the following properties:

- Reversible and ergodic.
- The eigenvectors of P and Q are the same, and hence the equilibrium distribution of both Markov chains are the same.
- The eigenvalues of P and Q map to each other by the relation

$$\lambda^{(P)} = 1 + \frac{1}{\alpha}\lambda^{(Q)} \tag{A.11}$$

Choosing α to be larger than the spectral radius of Q has the effect of setting all eigenvalues of P greater than 0. In addition, the second eigenvalue of P can be mapped to the second eigenvalue of Q using Equation (A.11) - if α is less than the spectral radius, such may not be the case. The following theorem provides a means to upper bound the largest eigenvalue magnitude of Q.

Theorem A.2.2 (Gershgorin Circle Theorem [8])**.** *Let Q be a real matrix with real eigenvalues, and define R_i to be the sum over the off diagonal entries of row i:*

$$R_i = \sum_{k \neq i} Q(i,k)$$

Then every eigenvalue is in at least one of the intervals:

$$D_i = \{t : |t - Q(i,i)| \leq R_i\} \tag{A.12}$$

Let $C_{\boldsymbol{x}}$ be the sum over the off diagonal entries of the column corresponding to $\boldsymbol{x}$:

$$C_{\boldsymbol{x}} = \sum_{k \neq \boldsymbol{x}} Q\left(\boldsymbol{x}, k\right)$$

A corollary of Gershgorin's circle theorem is the following upper bound on the eigenvalue magnitudes:

$$|\lambda| \leq \max_{\boldsymbol{x} \in \chi} |Q\left(\boldsymbol{x}, \boldsymbol{x}\right)| + C_{\boldsymbol{x}} \tag{A.13}$$

The quantity $|Q\left(\boldsymbol{x}, \boldsymbol{x}\right)| + C_{\boldsymbol{x}}$ is the rate of exit added to the rate of entrance into state $\boldsymbol{x}$, and is equal to

$$\zeta \langle \mathbf{1}, \gamma^{A\boldsymbol{x}} \rangle + M\mu$$

We then obtain the following upper bound to the largest eigenvalue magnitude of Q

$$|\lambda| \leq \begin{cases} M\left(\zeta + \mu\right) & \text{if } \gamma \leq 1 \\ \zeta \langle \mathbf{1}, \gamma^{A\mathbf{1}} \rangle + M\mu & \text{if } \gamma > 1 \end{cases} \tag{A.14}$$

Our choice for α is given in Equation (A.14).

Bibliography

[1] W. E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9:17–29, 1951.

[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. Vorst. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, 2000.

[3] P. Bremaud. *Markov Chains Gibbs Fields, Monte Carlo Simulation, and Queues*. Springer, 1999.

[4] J. Demmel and B. Kagstrom. The generalized schur decomposition of an arbitrary pencil $a - \lambda b$: robust software with error bounds and applications. *ACM Transactions on Mathematical Software*, 19:160–174, 1993.

[5] P. Diaconis and D. Stroock. Geometric bounds for eigenvalues of markov chains. *The Annals of APplied Probability*, 1:36–61, 1991.

[6] K. K. Gade. *Numerical Estimation of the Second Largest Eigenvalue of a Reversible Markov Transition Matrix*. PhD thesis, New York University, 2009.

[7] W.K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, page 97–109, 1970.

[8] R. Horn and C. Johnsons. *Matrix Analysis*. Cambridge University Press, second edition, 2012.

[9] D. Hsu, A. Kontorovich, and C. Szepesvári. Mixing time estimation in reversible markov chains from a single sample path. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1459–1467, Cambridge, MA, USA, 2015. MIT Press.

[10] S. Karlin and J. McGregor. Ehrenfest urn models. *Journal of Applied Probability*, 2:352–376, 1965.

[11] D.G. Kendall. On the generalized birth-and-death process. *The Annals of Mathematical Statistics*, 19:1–15, 1948.

[12] D. Lay. *Linear Algebra and Its Applications*. Pearson, fourth edition, 2011.

[13] D.A. Levin, Y. Peres, and E. L. Wilmer. *Markov Chains and Mixing Times*. American Mathematical Society, 2009.

[14] G. Pritchard and D. J. Scott. Empirical convergence rates for continuous-time markov chains. *Journal of Applied Probability*, 38(1):262–269, 2001.

[15] C. Robert and G. Casella. *Introducing Monte Carlo Methods with R*. Springer, 2010.

[16] Y. Saad. *Numerical Methods of Large Eigenvalue Problems*. Society for Industrial and Applied Mathematics, 2 edition, 2011.

[17] A. Sokal. Monte carlo methods in statistical mechanics: Foundations and new algorithms. Technical report, New York University, 1996.

[18] L. N. Trefethen and D. Bau. *Numerical Linear Algebra*. SIAM, 1997.

[19] J. Zhang and J. Moura. Diffusion in social networks as sis epidemics: Beyond full mixing and complete graphs. *IEEE Journal of Selected Topics in Signal Processing*, 8:17–29, 2014.