Stochastic Analysis of Networked Systems

by

Hairi

A Dissertation Presented in Partial Fulfillment
of the Requirement for the Degree
Doctor of Philosophy

Approved May 2020 by the
Graduate Supervisory Committee:

Lei Ying, Chair
Weina Wang
Junshan Zhang
Yanchao Zhang

ARIZONA STATE UNIVERSITY

August 2020

ABSTRACT

This dissertation presents a novel algorithm for recovering missing values of co-evolving time series with partial embedded network information. The idea is to connect two sources of data through a shared low dimensional latent space. The proposed algorithm, named **NetDyna**, is an Expectation-Maximization algorithm, and uses the Kalman filter and matrix factorization approaches to infer the missing values both in the time series and embedded network. The experimental results on real datasets, including a Motes dataset and a Motion Capture dataset, show that (1) **NetDyna** outperforms other state-of-the-art algorithms, especially with partially observed network information; (2) its computational complexity scales linearly with the time duration of time series; and (3) the algorithm recovers the embedded network in addition to missing time series values.

This dissertation also studies a load balancing algorithm, the so called power-of-two-choices(Po2), for many-server systems (with N servers) and focuses on the convergence of stationary distribution of Po2 in the both light and heavy traffic regimes to the solution of mean-field system such that the load of system is $\lambda = 1 - \frac{\gamma}{N^\alpha}$ for $\gamma > 0$ and $0 \leq \alpha < \frac{1}{18}$. The framework of Stein's method and state space collapse (SSC) are used to analyze both regimes.

In both regimes, the thesis first uses the argument of state space collapse to show that the probability of the state being far from the mean-field solution is small enough. By a simple Markov inequality, it is able to show that the probability is indeed very small with a proper choice of parameters.

Then, for the state space close to the solution of mean-field model, the thesis uses Stein's method to show that the stochastic system is close to a linear mean-field model. By characterizing the generator difference, it is able to characterize the dominant terms in both regimes. Note that for heavy traffic case, the lower and

upper bound analysis of a tridiagonal matrix, which arises from the linear mean-field model, is needed. From the dominant term, it allows to calculate the coefficient of the convergence rate.

In the end, comparisons between the theoretical predictions and numerical simulations are presented.

To my family.

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude towards my supervisor, Professor Lei Ying. His rigorous thinking in scientific research have deeply influenced me. Throughout the training of group meetings, every question he asked to my presentations made me think deeper about the topic and hence made me understand more. If I have to choose one thing that I'm most grateful for learning during my PhD time is the critical thinking ability. He has always been a great mentor who dedicated a great amount of time and lots of efforts in my doctoral study. I am eternally grateful for him as being my supervisor who changed my view towards scientific research.

I would like to thank Professors Weina Wang, Junshan Zhang and Yanchao Zhang for serving on my committee. I would also like to thank Professor Tong who served on my qualifying exam committee. Their their valuable insights and feedback on my research have made my thesis more complete.

The past several years would not have been as enjoyable without the company of terrific colleagues at INLAB and friends. The valuable and passionate discussions with Dr. Zhen Chen, Dr. Xin Liu, Mr. Dheeraj Narasimha, Mr. Yiqiu Liu, Mr. Pengfei Jiang, Mr. Honghao Wei and Mr. Weichang Wang. Thank you for all of help and happiness brought to me in this long journey.

Finally, I would like to thank my familiy, i.e. my parents and my brother for their unconditional support and love. And last but not the least, many thanks to my girlfriend Zona, who have been with me along this journey though thousands of miles apart. Thank you for believing in me all these years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure | Page

Chapter 1

INTRODUCTION

In this dissertation, we studied two parts of networked system. The first one is about sensor network, where we want to recover missing value problem. The second problem is to show the convergence rate coefficient analysis of queueing system.

Co-evolving time series are common in many real world applications such as temperature monitoring in smart buildings, mobile object tracking, and motion capturing for environmental monitoring. These co-evolving time series are generated from a collection of system components (such as sensors) over an extended time period. In many cases, because of reasons such as sensor duty cycling, packet losses in transmissions, and hardware malfunctions, we only observe incomplete time series with many missing values. Recovering missing values is important but challenging problem for many applications, e.g. for determining whether certain chemical level in the drinking water exceeds a threshold.

Large-scale and complex stochastic systems are ubiquitous in the real world, from the hospital beds to food deliver requests, to massive online platform like Zoom and Netflix. Moreover, in a global pandemic like Covid-19, the requests in such systems has been very high, where in queueing system we call a heavy traffic scenario. In this dissertation, we consider a queueing system called supermarket model, where jobs are allocated to servers according to a load balancing algorithm called power-of-the-two choices. The arrival process is a Poisson process and the service time is a random variable. Moreover, the power-of-the-two choices load balancing algorithm is also an algorithm in a randomized fashion in it. So our system is a stochastic system.

1.1 Background

A novel approach to recover missing values in co-evolving time series is to explore the correlation of different data sources (i.e. the co-evolution) instead of treating time series as independent processes. In particular, an embedded network may be constructed for co-evolving time series. For a wireless sensor network, the embedded network is a graph in which an edge exists between two nearby sensors to indicate the correlation of their measurements. In epilepsy study, measurements collected from different brain regions of a patient can be connected via a brain graph that describes the correlation of brain activities. While exploiting the underlying correlation of time series is proven to be very effective, learning the underlying embedded network itself can be a daunting task. Often, similar to the time series data, we only have partial information of the embedded network.

A good load balancer should have a critical feature of low latency. For example, in a call center, it's unpleasant for a customer to receive a service while have to wait a significant long time. However, in a large-scale server system (N servers), the communication overhead is very costly. In other words, it takes too much time to ask each server on how many jobs are waiting in line. Thus, in a large-scaled system, it's not ideal to apply the famous Joint the Shortest Queue algorithm. On the other hand, power-of-the-two choices algorithm has a relatively small overhead, yet it can effectively balance out the disparity in the system.

In order to characterize the latency, it's critical to characterize the system in the steady state. It's important to mention that the arrival traffic plays an important role in the analysis of such load balancing system. When the arrival rate is independent of the system size, it's relatively straightforward; However, when the arrival rate is dependent of system size, the system quantities are often entangled.

In this thesis, we consider when the arrival rate is $\lambda = 1 - \frac{\gamma}{N^\alpha}$ for some $0 < \gamma \leq 1$ and $0 \leq \alpha$. This form include both the constant arrival rate case and the system size dependent rate case.

1.2 Literature Review

Missing value recovery problem is closely related to the low-rank matrix factorization Koren *et al.* (2009), which has been extensively studied for recommendation systems. Mnih and Salakhutdinov (2008) proposed a probabilistic matrix factorization (PMF) method for user item rating matrix, which scales linearly with the number of observations. PMF performs well on large, sparse and imbalanced Netflix dataset. However, sequential dependence, which is the intrinsic difference of time series data from other data, is not encoded in the PMF. Srebro and Jaakkola (2003) proposed MSVD, a SVD based missing value recovery method, where missing values are initialized using a linear interpolation first. Ma *et al.* (2008) proposed probabilistic matrix factorization with the social network information. SoRec considered recovering missing values in user item matrix utilizing a social network, where it's been shown that social network information can help improve the recovery result.

Yi *et al.* (2000) proposed an online algorithm, which discovers the correlation among multiple time series and jointly recover the missing values. Papadimitriou *et al.* (2005); Li (2011) provided general frameworks for data mining tasks, including missing value recovery. For recovering missing values in time series, Papadimitriou *et al.* (2005) proposed Spirit, a PCA based learning model. Li *et al.* (2009) proposed DynaMMo, which learns the dynamics of latent variables. By filling missing values using linear interpolation or some other methods, DynaMMo uses Kalman filtering to estimate system parameters. However, there's no network data is used. Cai *et al.* (2015a) proposed DCMF, an algorithm that combines partially observed time series data with a fully observed embedded network. In Cai *et al.* (2015b), the authors have further developed DCMF into a higher dimension model by considering more than one type of measurements. However, both papers assume complete knowledge of the embedded network. Our work considers partially observed network information.

Neural network approach has been introduced to time series problem. Lipton *et al.* (2016); Che *et al.* (2018); Cao *et al.* (2018) proposed recurrent neural network based algorithms. However, they are looking at classification problems with missing values in time series. In this chapter, we studied the problem of recovering the missing value itself.

We note that Ma *et al.* (2008); Cai *et al.* (2015a,b) proposed to connect two different data resources via a shared latent feature space, which motivated our algorithm, NetDyna, which use a common latent feature space for both time series and the embedded network. Finally, Little and Rubin (2014) introduces different types of missing value patterns. In our work, we considers three different patterns including missing uniformly, missing as a block, and missing entirely. Our experimental results show that our algorithm outperforms other algorithms in most cases under these three different missing patterns.

On the other hand, performance analysis of load balancing algorithms in many-server systems is one of the most fundamental and widely-studied problems in queueing theory. The key to these analysis is to characterize the stationary distribution once the system reaches the steady state. If the stationary distribution is known, the expected queue length and the average waiting time can be easily calculated. However often times, for a large scaled stochastic system, it's either hard to precisely characterize it or computationally very much expensive. For example, in a queueing system with N homogeneous servers and b buffer size, the state space is in the order of $O(N^b)$. And for such a system, the transition rate is state dependent, thus it's hard to apply the elementary knowledge used in probability theory to calculate it. Therefore, an approximation to the stationary distribution is a reasonable and practical approach.

The stationary distribution of a stochastic system has been approximated by using the solution of the corresponding mean-field model. The convergence of distribution in finite time to the corresponding mean-field model has also been studied Kurtz (1971)Kurtz (1981). However, it differs from our focus of interest, which is the stationary distribution where the system has achieved the steady state. Besides, in order to have it applicable to the steady state situations, interchange of the limits has to be applied, where the convergence rate can't be specified. Especially the mean-field solution for Po2 algorithm have been addressed in Mitzenmacher (1996)Vvedenskaya *et al.* (1996). However, the study of convergence rate of mean-field solution to steady state distribution wasn't clear.

In order to tackle it, Stein's method, Stein (1972) Stein (1986), has been introduced into queueing systems to characterize the convergence rate. Ying (2016) has first shown the Po2 convergence results in light traffic for constant buffer size. Later Ying (2017) has derived the results to infinite buffer case including both light and heavy traffic regimes. Gast (2017) Gast and Van Houdt (2018) have refined the results by calculating the $\frac{1}{N}$ coefficient for general distance functions based on previous convergence results in light traffic regime. However, our work differs from this paper by considering heavy traffic regime and mean square distance function specifically.

Liu and Ying (2018) has applied a simple linear mean-field model in the neighborhood of the mean-field solution and the idea of state space collapse Bertsimas *et al.* (1998) for other regions to show that the zero waiting time results of many load balancing algorithms in the heavy traffic regime. The results in this work also share similar spirit with it. However, they considered a collection of load balancing algorithms, including power-of-d-choices, where d has to large enough.

1.3 Summary of Contributions

In Chapter 2 ,we formulate the missing value problem as maximum likelihood problem, which can be solved using the Expectation Maximization (EM) method for recovering time series and embedded network data. We propose an effective algorithm **NetDyna** for recovering missing values and evaluated the performance of **NetDyna** using real datasets under different missing settings. The experimental results showed the effectiveness and efficiency when comparing with the-state-of-the-art algorithms. In particular, **NetDyna** outperforms other algorithms for recovering missing time series when the embedded network is partially known.

In Chapter 3 through 5, we study power-of-2-choices load balancing systems (N servers) assuming exponential service time. Each server has a buffer of size $b-1$ i.e. a server can have at most one job in service and $b-1$ jobs in queue, where $b = O(\log N)$. Jobs are served in first-come-first-serve (FIFO) order.

In Chapter 3, we focus on the steady-state convergence of load balancing algorithms in the light traffic regime such that the load of system is λ is constant, where $0 < \lambda < 1$. We calculate the the coefficient term in front of convergent $1/N$ term. The proof of the main result is based on Stein's method and state space collapse.

In Chapter 4, we focus on the steady-state convergence of load balancing algorithms in the heavy traffic regime such that the load of system is $\lambda = 1 - \frac{\gamma}{N^\alpha}$, where $\gamma > 0$ is constant and $0 < \alpha < 0.25$. We establish a convergent bound. The proof of the main result is based on Stein's method and propagation of chaos.

In Chapter 5, we focus on the steady-state convergence of load balancing algorithms in the heavy traffic regime such that the load of system is $\lambda = 1 - \frac{\gamma}{N^\alpha}$, where

$\gamma > 0$ is constant and $0 < \alpha < 0.25$. We calculate the the coefficient term in front of the dominant convergent term. The proof of the main result is based on Stein's method and state space collapse.

We would emphasis the analysis framework in the dissertation combines two tools for steady-state analysis: Stein's method and state space collapse.

Chapter 2

MISSING VALUE RECOVERY

In this chapter, we first formulate the problem, present our algorithm and then present the experiment results.

2.1 Model and Problem Formulation

In this section, we present our model and formulate the problem of recovering coevolving time series under our model as a maximum likelihood problem.

2.1.1 Joint Embedding of Coevolving Time Series

We assume the coevolving time series are associated with a network where each node is the source of a time series and time series are correlated if the two sources are neighbors. Our goal is to recover missing values of coevolving time series with partial time series and partial network information.

Example: Consider a smart house with six rooms as shown in Figure 2.1, where sensors are placed in each room to track the temperature. A directed edge (i, j) represents the dependency of the temperature of room j on that of room i. The time series data are partially observed, as shown in Table 2.1. Our goal is to recover missing values in time series data and missing edges of the network. □

Consider coevolving time series with N sources and over T time slots. Each source creates one measurement at each time slot. The coevolving temperature data can be represented by an $N \times T$ matrix $\mathbf{X}$. The weighted adjacency matrix of the network is an $N \times N$ matrix, denoted by $\mathbf{S}$. Note that without imposing any structure on the time series data, missing measurements can be arbitrary values, so it is impossible

Table 2.1: Room Temperature Data of a Smart House

	t_1	t_2	t_3	t_4	t_5
room 1	72	73			74
room 2		74	73	72	
room 3	71	73			75
room 4	76		73		74
room 5	74	71			
room 6	75	73		72	

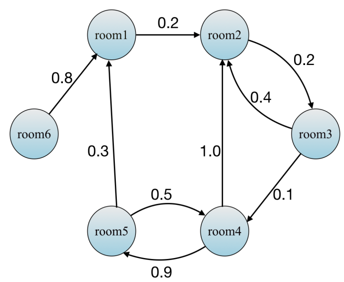


Figure 2.1: The Network Associated with Room Temperature Data

to recover these missing values. A widely observed structure in real-world datasets is that high-dimensional data can often be embedded into a low-dimensional space, which makes recovery of missing values possible. For example, matrix completion algorithms such as collaborative filtering are often based on this assumption. By leveraging this observation, we propose joint time series and network embedding. In particular, we assume each node (say node i) is associated with an L-dimensional latent vector $\mathbf{U}_i \in \mathbf{R}^L$. We define $\mathbf{U}$ to be an $L \times N$ matrix such that the ith column $\mathbf{U}_i \triangleq \mathbf{U}_{(:,i)}$ is the latent vector of object i. In addition, there is L-dimensional time series $\mathbf{Z}_t \in \mathbf{R}^L$ which evolves as a linear system such that

$$\mathbf{Z}_t = \mathbf{B}\mathbf{Z}_{t-1} + \mathbf{W}_t$$

where $\mathbf{B}$ is an $L \times L$ transition matrix, $\mathbf{Z}_1 \sim \mathcal{N}(z_0, \mathbf{\Psi}_0)$ and $\mathbf{W}_t \sim \mathcal{N}\left(0, \sigma_Z^2 \mathbf{I}\right)$. Note that $\mathbf{Z}_1$ is an L-dimensional Gaussian vector with a general distribution and $\mathbf{W}_t$ is zero mean Gaussian vector with i.i.d. components, representing i.i.d. Gaussian noise.

The time series data $\mathbf{X}_t$ (an N-dimensional vector) is generated based on $\mathbf{U}$ and $\mathbf{Z}_t$ such that

$$\mathbf{X}_t = \mathbf{U}'\mathbf{Z}_t + \epsilon_t,$$

where $\epsilon_t \sim \mathcal{N}\left(0, \sigma_X^2 \mathbf{I}\right)$ is a N-dimensional zero mean Gaussian noise.

The network matrix $\mathbf{S}$ is generated based on $\mathbf{U}$ such that

$$\mathbf{S} = \mathbf{V}'\mathbf{U} + \tau \tag{2.1}$$

where $\mathbf{V}$ is an $L \times N$ matrix such that the ith column $\mathbf{V}_i \triangleq \mathbf{V}_{(:,i)} \sim \mathcal{N}\left(\mathbf{U}_i, \sigma_V^2 \mathbf{I}\right)$ and $\tau_i \triangleq \tau_{(:,i)} \sim \mathcal{N}\left(0, \sigma_S^2 \mathbf{I}\right)$.

Figure 2.2 summarizes the joint embedding model, where $\mathbf{U}$ and $\mathbf{Z}$ are latent variables and $\mathbf{X}$ and $\mathbf{S}$ are observed variables.

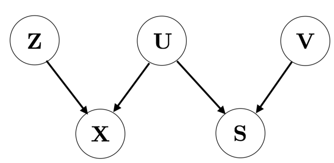


Figure 2.2: Time Series and Network Joint Embedding

2.1.2 Missing Values Recovery Problem and Maximum Likelihood Formulation

With the model and notation defined above, we introduce the maximum likelihood formulation for tackling the missing value recovery problem defined below. We define an $N \times T$ matrix $\mathbf{M}$ such that $\mathbf{M}_{nt} = 1$ when $\mathbf{X}_{nt}$ is observed and $\mathbf{M}_{nt} = 0$ if $\mathbf{X}_{nt}$ is missing; and define an $N \times N$ matrix $\tilde{\mathbf{M}}$ such that $\tilde{\mathbf{M}}_{ij} = 1$ if the edge weight (i, j)

is observed and $\tilde{\mathbf{M}}_{ij} = 0$ otherwise. These two matrices specify the entries of missing values.

Summarizing the model we have introduced, we have

$$\mathbf{Z}_t = \mathbf{B}\mathbf{Z}_{t-1} + \mathbf{W}_t \tag{2.2}$$

$$\mathbf{X}_t = \mathbf{M}_t \odot (\mathbf{U}'\mathbf{Z}_t + \epsilon_t) \tag{2.3}$$

$$\mathbf{S} = \tilde{\mathbf{M}} \odot (\mathbf{V}'\mathbf{U} + \tau), \tag{2.4}$$

where $\mathbf{M}_t = \mathbf{M}_{(:,t)}$ is the tth column of matrix $\mathbf{M}$.

Let $\hat{\mathbf{X}}$ and $\hat{\mathbf{S}}$ denote the observed time series and weighted adjacency matrix, where the missing values are set to be zero. The missing value recovery problem is defined below.

Missing Value Recovery:

Input: Partially observed time series from a networked data sources and a partially observed network, i.e. $\hat{\mathbf{X}}$, $\mathbf{M}$, $\hat{\mathbf{S}}$, $\tilde{\mathbf{M}}$.

Output: Complete time series data and network data $\mathbf{X}$ and $\mathbf{S}$ that match the observed data, i.e. $\mathbf{X} \odot \mathbf{M} = \hat{\mathbf{X}}$ and $\mathbf{S} \odot \tilde{\mathbf{M}} = \hat{\mathbf{S}}$.

Note that the co-evolving time series is a high-dimensional random process defined by (2.2)-(2.4), which is characterized by the parameter set

$$\theta = \{\mathbf{U}, \mathbf{B}, \mathbf{z}_0, \mathbf{\Psi}_0, \sigma_Z, \sigma_X, \sigma_V, \sigma_S\}.$$

We propose the following maximum likelihood problem for recovering the missing values:

$$\max_{\mathbf{X},\mathbf{S},\theta} p\left(\mathbf{X}, \mathbf{S} | \, \theta, \hat{\mathbf{X}}, \hat{\mathbf{S}}\right) \tag{2.5}$$

Note that the condition $\hat{\mathbf{X}}$ and $\hat{\mathbf{S}}$ mean

$$\mathbf{X} \odot \mathbf{M} = \hat{\mathbf{X}} \;\text{ and }\; \mathbf{S} \odot \tilde{\mathbf{M}} = \hat{\mathbf{S}}.$$

Table 2.2 summarizes the key notations to be used throughout the chapter.

2.2 Proposed Algorithm: NetDyna

Since directly maximizing likelihood (2.5) is intractable Goodfellow *et al.* (2016); Bishop (2006), we first use the expectation-maximization (EM) algorithm to maximize the evidence lower bound, defined below, to find $\boldsymbol{\theta}$ based on the observed data

$$E_{\mathbf{Z},\mathbf{V}|\theta,\hat{\mathbf{X}},\hat{\mathbf{S}}}[\ln p(\hat{\mathbf{X}}, \hat{\mathbf{S}}, Z, V|\theta)]. \tag{2.6}$$

After obtaining $\boldsymbol{\theta}$, and distributions of latent variables $\mathbf{Z}$ and $\mathbf{U}$, we then recover the missing time series and network data by solving the maximum likelihood problem.

The EM algorithm first initializes parameter set $\boldsymbol{\theta}$. In the expectation step, we compute the distributions of latent variables $\mathbf{Z}$ and $\mathbf{V}$ conditioned on the current parameter set $\boldsymbol{\theta}$ and observed data and then calculate (2.6). In the maximization step, we iteratively update the parameter set $\boldsymbol{\theta}$ based on the current distribution of latent variables. The algorithm terminates after the value of the evident lower bound converges.

2.2.1 The E-Step

In this step, parameter set $\boldsymbol{\theta}$ is fixed. According to Figure 2.2, time series data $\mathbf{X}$ and network data $\mathbf{S}$ are conditionally independent given $\mathbf{U}$, so can be inferred separately. In particular, we have To calculate expectation, the first step is to calculate the distributions of latent variables $\mathbf{Z}$ and $\mathbf{V}$ given the parameter set θ and observed data $\hat{\mathbf{X}}$ and $\hat{\mathbf{S}}$.

Distributions of Latent Variables for Time Series Data

Equations (2.2) and (2.3) show the dynamic of latent process $\{\mathbf{Z}_t\}_{t=1}^{T}$ and partially observed $\{\mathbf{X}_t\}_{t=1}^{T}$. To simplify the notation, for each time slot $t = 1, \cdots, T$, define

$$\mathbf{O}_t = \{i|\mathbf{M}_{it} > 0, i = 1, \cdots, N\}$$

$$\mathbf{X}_t^* = \mathbf{X}_t(\mathbf{O}_t)$$
$$\mathbf{H}_t = \mathbf{U}'(\mathbf{O}_t, :) \tag{2.7}$$

where $\mathbf{O}_t$ is the index set of observed entries at time t, $\mathbf{X}_t^*$ is the observed entries at time t and $\mathbf{H}_t$ is a compressed version of matrix $\mathbf{U}'$ at time t. For example, suppose

$$\mathbf{U}' = \begin{pmatrix} 1 & 0 \\ 0.5 & 0.5 \\ 0 & 1 \end{pmatrix}, \ \mathbf{Z}_t = \begin{pmatrix} 0.7 \\ 0.3 \end{pmatrix},$$

$$\mathbf{X}_t = \begin{pmatrix} 0.7 \\ 0.5 \\ 0 \end{pmatrix} \text{ and } \mathbf{M}_t = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}.$$

Then we have

$$\mathbf{O}_t = \{1, 2\}, \ \mathbf{X}_t^* = \begin{pmatrix} 0.7 \\ 0.5 \end{pmatrix}, \text{ and } \mathbf{H}_t = \begin{pmatrix} 1 & 0 \\ 0.5 & 0.5 \end{pmatrix}.$$

We can rewrite equation (2.3) as follows:

$$\mathbf{X}_t^* = \mathbf{H}_t \mathbf{Z}_t + \epsilon_t. \tag{2.8}$$

Since noises and the initial condition are Gaussian, so are the posteriors of latent variable $\{\mathbf{Z}_t\}_{t=1}^T$. Then, we apply the forward-backward algorithm Bishop (2006) or Kalman filter and smoother Gelb *et al.* (1974); Byron *et al.* (2004) to calculate the posteriors. Define the following two posterior distributions, one conditioned on observations up to time t and the other on all observations:

$$p(\mathbf{Z}_t | \mathbf{X}_1^*, \cdots, \mathbf{X}_t^*) = \mathcal{N}(\mathbf{Z}_t | \mu_t, \mathbf{\Psi}_t)$$
$$p(\mathbf{Z}_t | \mathbf{X}_1^*, \cdots, \mathbf{X}_T^*) = \mathcal{N}(\mathbf{Z}_t | \tilde{\mu}_t, \tilde{\mathbf{\Psi}}_t).$$

By applying the forward algorithm (or the Kalman filter) to estimate μ_t and $\mathbf{\Psi}_t$, we have

$$\begin{aligned}
\mathbf{P}_{t-1} &= \mathbf{B}\mathbf{\Psi}_{t-1}\mathbf{B}' + \sigma_Z^2\mathbf{I} \\
\mathbf{K}_t &= \mathbf{P}_{t-1}\mathbf{H}_t'(\mathbf{H}_t\mathbf{P}_{t-1}\mathbf{H}_t' + \sigma_X^2\mathbf{I})^{-1} \\
\mu_t &= \mathbf{B}\mu_{t-1} + \mathbf{K}_t(\mathbf{X}_t^* - \mathbf{H}_t\mathbf{B}\mu_{t-1}) \\
\mathbf{\Psi}_t &= (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\mathbf{P}_{t-1}
\end{aligned} \tag{2.9}$$

with initial conditions

$$\begin{aligned}
\mathbf{K}_1 &= \mathbf{\Psi}_0\mathbf{H}_1'(\mathbf{H}_1\mathbf{\Psi}_0\mathbf{H}_1' + \sigma_X^2\mathbf{I})^{-1} \\
\mu_1 &= \mathbf{z}_0 + \mathbf{K}_1(\mathbf{X}_1^* - \mathbf{H}_1\mathbf{B}\mathbf{z}_0) \\
\mathbf{\Psi}_1 &= (\mathbf{I} - \mathbf{K}_1\mathbf{H}_1)\mathbf{\Psi}_0.
\end{aligned} \tag{2.10}$$

Then, by applying the backward algorithm (or Kalman smoother), we have

$$\begin{aligned}
\mathbf{J}_t &= \mathbf{\Psi}_t\mathbf{B}'(\mathbf{P}_t)^{-1} \\
\tilde{\mu}_t &= \mu_t + \mathbf{J}_t(\tilde{\mu}_{t+1} - \mathbf{B}\mu_t) \\
\tilde{\mathbf{\Psi}}_t &= \mathbf{\Psi}_t + \mathbf{J}_t(\tilde{\mathbf{\Psi}}_t - \mathbf{P}_t)\mathbf{J}_t'.
\end{aligned} \tag{2.11}$$

The expectations are

$$\begin{aligned}
E[\mathbf{Z}_t] &= \tilde{\mu}_t \\
E[\mathbf{Z}_t\mathbf{Z}_{t-1}'] &= \tilde{\mathbf{\Psi}}_t\mathbf{J}_{t-1}' + \tilde{\mu}_t\tilde{\mu}_{t-1}' \\
E[\mathbf{Z}_t\mathbf{Z}_t'] &= \tilde{\mathbf{\Psi}}_t + \tilde{\mu}_t\tilde{\mu}_t'
\end{aligned} \tag{2.12}$$

which are needed for parameter update in maximization step.

Distributions of Latent Variables for Network Data

Now the posterior distributions of network latent variables can also be similarly derived. Rewrite (2.4)

$$\mathbf{S}' = \tilde{\mathbf{M}}' \odot (\mathbf{U}'\mathbf{V} + \tau), \tag{2.13}$$

and define

$$\begin{aligned} \tilde{\mathbf{O}}_j &= \{i | \tilde{\mathbf{M}}'_{ij} = \tilde{\mathbf{M}}_{ji} > 0, i = 1, \cdots, N\} \\ \mathbf{S}^*_j &= \mathbf{S}'_j(\tilde{\mathbf{O}}_j) \\ \mathbf{G}_j &= \mathbf{U}'(\tilde{\mathbf{O}}_j, :) \end{aligned} \tag{2.14}$$

where $\tilde{\mathbf{O}}_j$ is the index set of observed entries of column $\tilde{\mathbf{M}}'_j$(or row $\tilde{\mathbf{M}}(j, :)$), $\mathbf{S}^*_j$ is the observed entries of column vector $\mathbf{S}'_j$ and $\mathbf{G}_j$ is the compressed version of $\mathbf{U}'$. Further rewrite (2.4) as follows:

$$\mathbf{S}^*_j = \mathbf{G}_j\mathbf{V}_j + \tau_j$$

Noises for network data are also Gaussian, so are the posteriors of latent variables $\{\mathbf{V}_i\}_{i=1}^N$. Note that we assume that for $\mathbf{V}$, columns are independent of each other. Define

$$p(\mathbf{V}_j|\mathbf{S}^*_j) = \mathcal{N}(\mathbf{V}_j|\boldsymbol{\nu}_j, \boldsymbol{\gamma}_j)$$

By applying Bayes' theorem, we have

$$\begin{aligned} \boldsymbol{\gamma}_j &= (\sigma_V^{-2}\mathbf{I} + \sigma_S^{-2}\mathbf{G}'_j\mathbf{G}_j)^{-1} \\ \boldsymbol{\nu}_j &= \boldsymbol{\gamma}_j(\sigma_S^{-2}\mathbf{G}'_j\mathbf{S}^*_j + \sigma_V^{-2}\mathbf{U}_j). \end{aligned} \tag{2.15}$$

The expectations are

$$E[\mathbf{V}_j] = \nu_j$$

$$E[\mathbf{V}_j\mathbf{V}_j'] = \gamma_j + \nu_j\nu_j' \tag{2.16}$$

which are also used in the maximization step.

After obtaining the distributions of latent variables, we can obtain the following approximation for the evidence lower bound. The details are omitted due to the page limit. Note that we add a scalar λ into the approximation which balances the contributions of observed network data and time series data in inferring the latent variable $\mathbf{U}$, which turns out to be important as we will see in the numerical evaluation.

$$\begin{aligned} Q(\boldsymbol{\theta}) \triangleq& E_{\mathbf{Z},\mathbf{V}|\boldsymbol{\theta},\hat{\mathbf{X}},\hat{\mathbf{S}}}[\ln p(\hat{\mathbf{X}},\hat{\mathbf{S}},\mathbf{Z},\mathbf{V}|\boldsymbol{\theta})] \\ =&(1-\lambda)\Bigg(-\frac{1}{2}\operatorname{tr}(\boldsymbol{\Psi}_0^{-1}(E[\mathbf{Z}_1\mathbf{Z}_1'] - E[\mathbf{Z}_1]\mathbf{z}_0' - \mathbf{z}_0E[\mathbf{Z}_1'] \\ &+ \mathbf{z}_0\mathbf{z}_0')) + \frac{1}{2}\ln|\boldsymbol{\Psi}_0^{-1}| + \frac{\sigma_Z^{-2}}{2}\sum_{t=2}^{T}(-\operatorname{tr}(E[\mathbf{Z}_t\mathbf{Z}_t']) \\ &+ \operatorname{tr}(\mathbf{B}'E[\mathbf{Z}_t\mathbf{Z}_{t-1}'] + \operatorname{tr}(\mathbf{B}E[\mathbf{Z}_{t-1}\mathbf{Z}_t'])) \\ &- \operatorname{tr}(\mathbf{B}'\mathbf{B}E[\mathbf{Z}_{t-1}\mathbf{Z}_{t-1}'])) - \frac{L(T-1)}{2}\ln\sigma_Z^2 \\ &- \frac{1}{2\sigma_Z^2}\sum_{t=1}^{T}((\mathbf{X}_t^*)'\mathbf{X}_t^* - 2(\mathbf{X}_t^*)'\mathbf{H}_tE[\mathbf{Z}_t] \\ &+ \operatorname{tr}(\mathbf{H}_t'\mathbf{H}_tE[\mathbf{Z}_t\mathbf{Z}_t'])) - \frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{N}\mathbf{M}_{it}\ln\sigma_S^2\Bigg) \\ &+ \lambda\Bigg(\frac{\sigma_V^{-2}}{2}\sum_{i=1}^{N}(\operatorname{tr}(E[\mathbf{V}_i\mathbf{V}_i']) - 2\mathbf{U}_i'E[\mathbf{V}_i] + \mathbf{U}_i'\mathbf{U}_i) \\ &- \frac{NL}{2}\ln\sigma_V^2 - \frac{1}{2\sigma_S^2}\sum_{i=1}^{N}((\mathbf{S}_i^*)'\mathbf{S}_i^* - 2(\mathbf{S}_i^*)'\mathbf{G}_iE[\mathbf{V}_i] \\ &+ \operatorname{tr}(\mathbf{G}_i'\mathbf{G}_iE[\mathbf{V}_i\mathbf{V}_i'])) - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\tilde{\mathbf{M}}_{ji}\ln\sigma_S^2\Bigg) \\ &+ \text{const} \end{aligned} \tag{2.17}$$

2.2.2 *The M-Step*

In the M step, we iteratively update the parameter set

$$\boldsymbol{\theta} = \{\mathbf{U}, \mathbf{B}, \mathbf{z}_0, \boldsymbol{\Psi}_0, \sigma_Z, \sigma_X, \sigma_V, \sigma_S\}$$

by fixing the distributions of the latent variables. In particular, to update their values, we sequentially setting the derivative of the parameters in parameter set $\boldsymbol{\theta} = \{\mathbf{U}, \mathbf{B}, \mathbf{z}_0, \boldsymbol{\Psi}_0, \sigma_Z, \sigma_X, \sigma_V, \sigma_S\}$ to be 0.

In a summary, after the E-step, we update the parameter set $\boldsymbol{\theta}$ with the following update rules:

$$\begin{aligned}
\mathbf{z}_0^{\text{new}} &= E[\mathbf{Z}_1] \\
\boldsymbol{\Psi}_0^{\text{new}} &= E[\mathbf{Z}_1\mathbf{Z}_1'] - E[\mathbf{Z}_1]E[\mathbf{Z}_1'] \\
\mathbf{B}^{\text{new}} &= (\sum_{t=2}^{T} E[\mathbf{Z}_t\mathbf{Z}_{t-1}'])(\sum_{t=2}^{T} E[\mathbf{Z}_{t-1}\mathbf{Z}_{t-1}'])^{-1} \\
(\sigma_Z^2)^{\text{new}} &= \frac{1}{L(T-1)}\sum_{t=2}^{T} \text{tr}(E[\mathbf{Z}_t\mathbf{Z}_t'] - E[\mathbf{Z}_t\mathbf{Z}_{t-1}'](\mathbf{B}^{\text{new}})' \\
&\quad - \mathbf{B}^{\text{new}}E[\mathbf{Z}_t\mathbf{Z}_{t-1}'] + \mathbf{B}^{\text{new}}E[\mathbf{Z}_{t-1}\mathbf{Z}_{t-1}'](\mathbf{B}^{\text{new}})') \\
(\sigma_V^2)^{\text{new}} &= \frac{1}{NL}\sum_{i=1}^{N}(\text{tr}(E[\mathbf{V}_i\mathbf{V}_i']) - 2\mathbf{U}_i'E[\mathbf{V}_i] + \mathbf{U}_i'\mathbf{U}_i) \\
\mathbf{U}_i^{\text{new}} &= \mathbf{A}_1\mathbf{A}_2^{-1} \\
\mathbf{A}_1 &= \frac{\lambda}{\sigma_S^2}\sum_{j=1}^{N}\tilde{\mathbf{M}}_{ji}\mathbf{S}_{ji}E[\mathbf{V}_j]' + \frac{(1-\lambda)}{\sigma_X^2}\sum_{t=1}^{T}\mathbf{M}_{it}\mathbf{X}_{it}E[\mathbf{Z}_t]' \\
&\quad + \frac{\lambda}{(\sigma_V^2)^{\text{new}}}E[\mathbf{V}_i]' \\
\mathbf{A}_2 &= \frac{\lambda}{\sigma_S^2}\sum_{j=1}^{N}\tilde{\mathbf{M}}_{ji}E[\mathbf{V}_j\mathbf{V}_j'] + \frac{(1-\lambda)}{\sigma_X^2}\sum_{t=1}^{T}\mathbf{M}_{it}E[\mathbf{Z}_t\mathbf{Z}_t'] \\
&\quad + \frac{\lambda}{(\sigma_V^2)^{\text{new}}}\mathbf{I}
\end{aligned}$$

$$
\begin{aligned}
(\sigma_X^2)^{\text{new}} = & \frac{1}{\sum_{t=1}^{T}\sum_{i=1}^{N}\mathbf{M}_{it}}\sum_{t=1}^{T}((\mathbf{X}_t^*)'\mathbf{X}_t^* - 2(\mathbf{X}_t^*)'\mathbf{H}_t^{\text{new}}E[\mathbf{Z}_t] \\
& + \text{tr}(\mathbf{H}_t^{\text{new}}E[\mathbf{Z}_t\mathbf{Z}_t'](\mathbf{H}_t^{\text{new}})')) \\
(\sigma_S^2)^{\text{new}} = & \frac{1}{\sum_{j=1}^{N}\sum_{i=1}^{N}\tilde{\mathbf{M}}_{ij}}\sum_{j=1}^{N}((\mathbf{S}_j^*)'\mathbf{S}_j^* - 2(\mathbf{S}_j^*)'\mathbf{G}_j^{\text{new}}E[\mathbf{V}_j] \\
& + \text{tr}(\mathbf{G}_j^{\text{new}}E[\mathbf{V}_j\mathbf{V}_j'](\mathbf{G}_j^{\text{new}})'))
\end{aligned} \tag{2.18}
$$

The new parameter set

$$\boldsymbol{\theta} = \{\mathbf{U}^{\text{new}}, \mathbf{B}^{\text{new}}, \mathbf{z}_0^{\text{new}}, \mathbf{\Psi}_0^{\text{new}}, \sigma_Z^{\text{new}}, \sigma_X^{\text{new}}, \sigma_V^{\text{new}}, \sigma_S^{\text{new}}\}$$

will be used in the next E-step.

2.2.3 Recovering Missing Values

After the EM algorithm converges, we set $\mathbf{Z} = (\tilde{\mu}_1, \cdots, \tilde{\mu}_T)$ and $\mathbf{V} = (\nu_1, \cdots, \nu_N)$. Then, we recover the missing values by setting

$$
\begin{aligned}
\mathbf{X} =& \mathbf{M} \odot \hat{\mathbf{X}} + (\mathbb{1}_{N\times T} - \mathbf{M}) \odot \mathbf{U}'\mathbf{Z} \\
\mathbf{S} =& \tilde{\mathbf{M}} \odot \hat{\mathbf{S}} + (\mathbb{1}_{N\times N} - \tilde{\mathbf{M}}) \odot \mathbf{V}'\mathbf{U}.
\end{aligned}
$$

Note that these recovery rules are obtained based on the fact that the means are the maximum likelihood solutions given $\mathbf{Z}$, $\mathbf{V}$ and $\mathbf{U}$.

2.2.4 NetDyna

The proposed algorithm is summarized in Algorithm 1 named **NetDyna**. And the following two theorems summarize the time complexity and the memory complexity of the algorithm.

Theorem 1. *The time complexity of **NetDyna** is* $O(\#iteration \cdot (TL^3 + NL^3 + \sum_t(n_tL^2 + n_t^2L + n_t^3) + \sum_j(n_j^2L + n_jL^2))$.

Here, n_t denotes the number of observed entries in $\mathbf{X}_t$, i.e. cardinality of $\mathbf{O}_t$, and n_j denotes the number of observed entries in $\mathbf{S}(j,:)$, i.e. cardinality of $\tilde{\mathbf{O}}_j$.

Proof. The overall time complexity is composed of 4 parts, computing time series data in E step, computing network data in E step, updating parameter set in M step and computing the evidence lower bound.

For each iteration, the complexity for time series in E step is $O(L^3T + \sum_t L^2 n_t + Ln_t^2 + n_t^3)$, including computation for forward algorithm, backward algorithm and expectations; the complexity for network data is $O(L^3N + L^2\sum_j n_j)$, including computation for inferring and expectations; the complexity for updating parameter set is $O(TL^3 + NL^3 + L^2\sum_j n_j + L^2\sum_t n_t + L\sum_t n_t^2 + L\sum_j n_j^2)$; the complexity for computing evidence lower bound is $O(L^3 + L^2T + L\sum_t n_t + NL + L\sum_j n_j)$. So, for each iteration, the complexity is $O(TL^3 + NL^3 + \sum_t(n_tL^2 + n_t^2L + n_t^3) + \sum_j(n_j^2L + n_jL^2))$. Thus, the overall complexity is the result in the theorem. □

Theorem 2. *The memory complexity of **NetDyna** is* $O(NT + L^2T + N^2 + L^2N)$.

Proof. The space complexity is composed of 4 parts, storing input dataset, parameter set, intermediate values in E step and loglikelihood values. For input dataset, the space complexity is $O(NT + N^2)$; for parameter set, the complexity is $O(LN)$; for intermediate values in E step, the complexity is $O(L^2T + L^2N)$; for loglikelihood, the complexity is $O(1)$.

Thus, overall space complexity is $O(NT + L^2T + N^2 + L^2N)$. □

2.3 Experimental Results

This section presents a comprehensive experimental evaluation of **NetDyna**, in terms of its effectiveness, sensitivity and efficiency, with two real datasets; and its comparison with five existing algorithms.

2.3.1 Experimental Setup

To evaluate the reconstruction performance of **NetDyna**, we use the root mean squared error(RMSE) for both time series data and network data

$$\text{RMSE}_{\text{time}} = \sqrt{\frac{\sum_{i,t}(1 - \mathbf{M}_{it})(\mathbf{X}_{it} - \hat{\mathbf{X}}_{it})^2}{\sum_{i,t}(1 - \mathbf{M}_{it})}}$$

$$\text{RMSE}_{\text{net}} = \sqrt{\frac{\sum_{i,j}(1 - \tilde{\mathbf{M}}_{ij})(\mathbf{S}_{ij} - \hat{\mathbf{S}}_{ij})^2}{\sum_{i,j}(1 - \tilde{\mathbf{M}}_{ij})}}$$

where $\hat{\mathbf{X}}_{it}, \hat{\mathbf{S}}_{ij}$ are observed values and $\mathbf{X}_{it}, \mathbf{S}_{ij}$ are reconstructed values. Note that we will divide the dataset into training and test parts, the metrics are on test data.

Motes Dataset

The Motes dataset [1] consists of temperature measurements from 54 sensors deployed at the Intel Berkeley Research Lab over a month. The temperature measurements are the time series data. The dataset also contains locations of the sensors and the connectivity probabilities among sensors. So using these information, we define each entry of the network matrix as following:

$$\mathbf{S}_{ij} = \alpha \cdot (1 - \frac{d_{ij}}{\max_{i,j}(d_{ij})}) + (1 - \alpha) \cdot c_{ij}$$

where d_{ij} is the Euclidian distance between sensor i and sensor j, c_{ij} is the connectivity probability between sensor i and sensor j, and α is weight control of the two parts. In the experiment, we set $\alpha = 0.5$. In all the Motes dataset related simulations, we use data from all 54 sensors and time slots from 1 to 2880. Note that 2880 time slots is roughly duration of a whole day.

[1]http://db.csail.mit.edu/labdata/labdata.html

Motion Capture Dataset

Motion Capture dataset [2] consists of body movement measurements from markers placed on human body. We used the Mawashi Geri data, a ”spin kick” martial art movement, in which 41 markers have been used and there are 1472 frames. Each marker has 3-dimensional coordinates so we have 123 features. Using these data, we define each entry of network matrix as following:

$$\mathbf{S}_{ij} = \alpha \cdot (1 - \frac{d_{ij}}{\max_{i,j}(d_{ij})})$$

where d_{ij} is the Euclidian distance between two markers, while $\alpha = 1$ if i and j are the same coordinate, otherwise $\alpha = 0.5$.

Note that we have standardized, i.e. subtracted the mean and divided by the standard deviation, both datasets before applying **NetDyna** and other state-of-the-art algorithms.

We consider three types of synthetic missing value patterns for time series data: missing as a block fashion, missing uniformly and missing entirely. For the network data, we assume each entry is missing uniformly at random.

- Missing as a block, also called occlusion in Li *et al.* (2009): We randomly pick a sensor j and a starting time slot t for this marker. Then, we choose the duration of the missing block for the selected sensor and starting time as a Poisson distributed random variable according to the observed statistics. We repeat this step until the percentage of missing values reaches the threshold we set.

- Missing uniformly: For each observed value, we uniformly at random remove it according to a pre-selected probability.

[2] http://mocap.cs.cmu.edu

- Missing entirely: Randomly choose one of the time series and remove it entirely.

2.3.2 Effectiveness

For effectiveness, we compared **NetDyna** with the following algorithms.

1. Dynamical Contextual Matrix Factorization(DCMF) Cai *et al.* (2015a): It is a time series data mining algorithm with fully observed embedded network information. DCMF infers the missing values based on both observed values in time series data and complete network data. However, the fully observed network information is based only a zero mean prior, which is lack of accountability for network information.

2. Dynamical Matrix Factorization(DMF) Cai *et al.* (2015a): DMF is also a special case of DCMF, where missing values are inferred solely based on observed time series data. If we set $\lambda = 0$, **NetDyna** becomes DMF. We omit the result of that when comparing the algorithms.

3. DynaMMo Li *et al.* (2009): It is a time series data mining algorithm with no network information. DynaMMo utilizes only time series data $\mathbf{X}$, where missing values are filled by interpolation or other methods first. Then, learn the system parameter by maximizing the likelihood of time series data, which include both observed data and interpolated data. Note that interpolated data are also used in the objective; however, in a missing as a block setting, where missing values are evolving far from linear interpolation, the learning process will be misled.

4. Missing value Singular Value Decomposition(MSVD) Srebro and Jaakkola (2003): It combines the idea of SVD with interpolation. The method first uses linear interpolation to fill the missing values, then iteratively applies SVD to the time se-

ries matrix and updates the missing values accordingly. Again, like DynaMMo, MSVD will also be potentially misled by interpolated data.

5. Probabilistic Matrix Factorization(PMF) Mnih and Salakhutdinov (2008): PMF is a collaborative filtering algorithm, where it learns the latent variables by maximizing a posteriori of the observed values. However, PMF does not consider the dynamics of time series evolution.

Throughout the simulations, we set $L = 15$ for all algorithms for both Motes dataset and Motion Capture dataset.

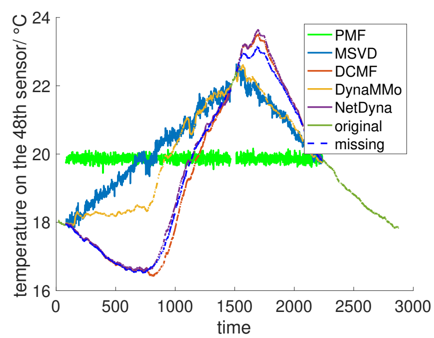


Figure 2.3: Quantative Results for Motes Dataset

Figure 2.3 shows the results of recovering missing time series values under the algorithms mentioned above and **NetDyna**. We selected the 48th sensor in the Motes dataset, two consecutive parts are missing, whose true values are denoted by blue dashed lines; three consecutive parts are observed, which are denoted by green lines. In this case, all algorithms are trying to recover the missing values and the algorithm that tracks blue dashed curve the closest is the best. PMF, MSVD, DCMF, DynaMMo and **NetDyna** are denoted by shiny green, blue, red, yellow and purple curves respectively. As we can see that in the first missing parts, **NetDyna** i.e. the purple curve tracks the missing values the best. Although in the second missing parts

DCMF is very close to **NetDyna**, the overall performance of **NetDyna** beats all state-of-the-art algorithms.

From Table 2.3 to Table 2.7, we will show the quantitative comparisons of time series data error of different algorithms with different missing patterns on both datasets.

The quantitative comparisons with the state-of-the-art algorithms are shown in Table 2.3 and Table 2.4 under different missing percentages for the Motes dataset. In the missing column, the first number is the missing percentage of the time series data and the second number is the missing percentage of network data. Note that the DCMF algorithm also considers network information, however it only allows complete network information. Therefore, for partially observed network information, we set network value to be 0, i.e. no connection, for the unobserved entries.

Under the missing as a block model with different missing percentage settings, **NetDyna** outperforms other algorithms almost in all cases. We note that with fully observed network information, DCMF performs closely to **NetDyna**; however, with partially observed network, **NetDyna** outperforms significantly. Also, under different percentages of partially observed network information, the RMSEs are similar to that of completely observed network information. Under the missing uniformly model, **NetDyna** slightly outperforms DCMF and outperforms other algorithms in almost all cases by a larger margin.

Table 2.5 and Table 2.6 summarize the results using the Motion Capture dataset. Under missing as a block setting, **NetDyna** outperforms all other algorithms up to missing 50% missing time series data. With sparse Motion Capture data such as when 90% time series data are missing, **NetDyna** is not as accurate as DynaMMo and MSVD. Under the missing uniformly model, **NetDyna** outperforms other algorithms.

Table 2.7 shows the results when one time series is completely missing and the number in the missing column denotes the percentage of missing network data. In this case, DynaMMo, MSVD and PMF algorithms are not defined. To have more comparisons, we introduced two more heuristic baselines.

- heuristic average: Recover the missing time series as the average of other time series whose sensors are connected with the missing sensor.
- heuristic weighted average: Recover the missing time series as the weighted average of other time series whose sensors are connected with the missing sensor.

As a result, **NetDyna** outperforms all the algorithms in the Motes dataset under all settings; with the Motion Capture dataset, **NetDyna** also outperforms in most of the cases, especially with partially observed network.

2.3.3 Sensitivity Results

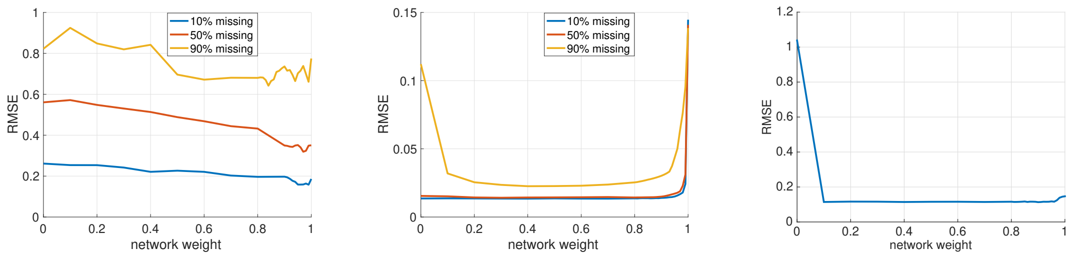
The experimental studies in section focus on the performance of **NetDyna** with different network weights and different levels of network sparsity.

Network Weight

We will first show that considering network data helps the recover missing time series data. Besides, the impact of network weight on **NetDyna** will be seen.

Figure 2.4 shows the performance with different network weights under the missing as a block model, missing uniformly model and missing entirely model, respectively, for the Motes dataset. Here we set network data to be fully observed for all these missing patterns. From the results, we can see that considering network information, i.e. $\lambda > 0$, in general improve the accuracy compared with only considering time series data, i.e. $\lambda = 0$.

Under the missing as a block mode, the best λ is between $(0.97, 0.99)$ in our experiments. Under the missing uniformly model, utilizing network information, i.e.



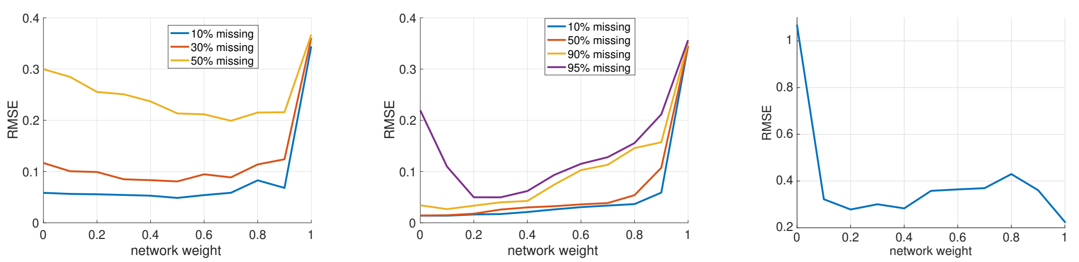
(a) under missing as a block fashion

(b) under missing uniformly fashion

(c) under missing entirely fashion

Figure 2.4: The Impact of Network Weight on the Performance for Motes Dataset

$\lambda \in (0, 1]$, helps improve the performance significantly when time series data is sparse, e.g. when 90% are missing as shown in Figure 2.4b. Under the missing entirely model, we can see that considering network data significantly improves the recovery results.We have similar results for Motion Capture dataset as well in Figure 2.5.



(a) under missing as a block fashion

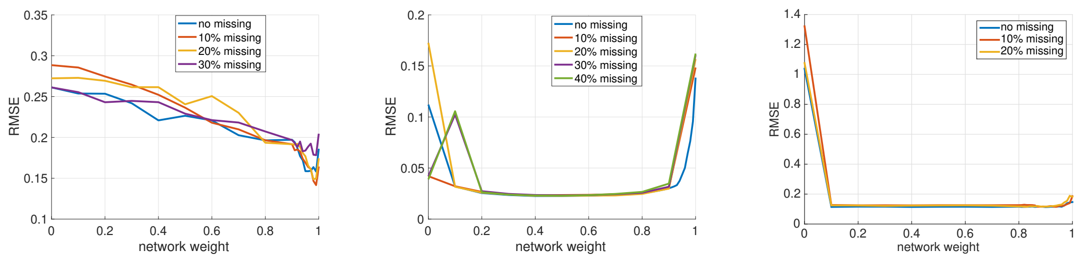
(b) under missing uniformly fashion

(c) under missing entirely fashion

Figure 2.5: The Impact of Network Weight on the Performance for Motion Capture Dataset

Network Sparsity

The experiment results focus on using partially observed network data for recovering missing values of time series data.



(a) under missing as a block fashion

(b) under missing uniformly fashion

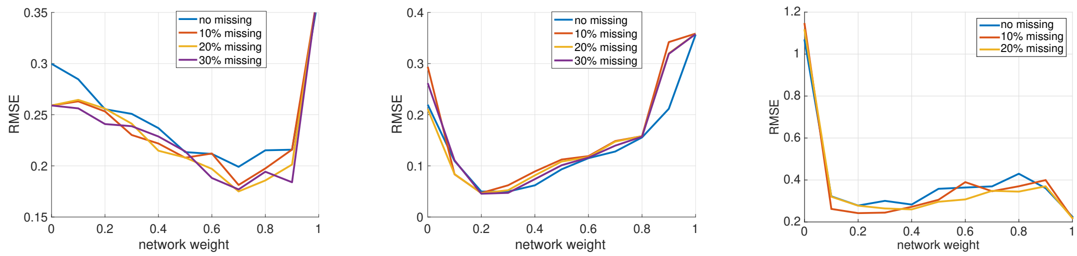
(c) under missing entirely fashion

Figure 2.6: The Impact of Sparsity of Network Information on the Performance for Motes Dataset

Figure 2.6 shows the performance with different amounts of network data under the missing as a block model, missing uniformly model and missing entirely model, respectively, for the Motes dataset.

From Figure 2.6a, we can see that even missing up to 30% of the network information, **NetDyna** performs closely to that of with fully observed network information. The best empirical network weights are in the interval $(0.9, 1)$.

For the uniformly missing case, we can observe from Figure 2.4b that network data is more helpful in the sparse time series data case, thus the result shown is when 90% time series data are missing. Similarly, with partially observed network information, the performance of **NetDyna** can also be enhanced. The best empirical network weight is in interval $(0.4, 0.6)$. For the entirely missing model, Figure 2.4c shows that with partially observed network information, the recovery results are as good as that of full network information. For the Motion Capture dataset, we have similar results as shown in Figure 2.7.



(a) under missing as a block fashion

(b) under missing uniformly fashion

(c) under missing entirely fashion

Figure 2.7: The Impact of Sparsity of Network Information on the Performance for Motion Capture Dataset

2.3.4 Network Data Recovery

In addition to recovering time series data, **NetDyna** can also recover the embedded network. In the presence of missing values in network information, it is inferred from observed network data and observed time series data. With 95% of time series data missing, we vary the sparsity of network data for Motes dataset. The reconstruction error for network data is shown in Figure 2.8. For network data recovery, the accuracy of recovering network data can increase when the network weight increases as shown in the figure.

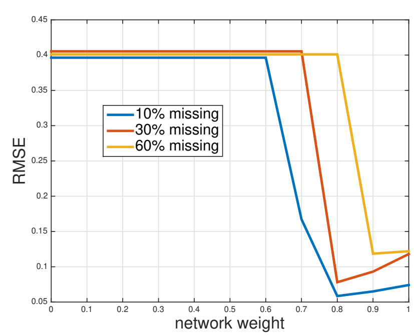


Figure 2.8: Network Data Sparsity on Network Data Recovery

2.3.5 *Efficiency Results*

As we discussed in Section 4 that the complexity of **NetDyna** is $O(\#\text{iteration} \cdot (TL^3 + NL^3 + \sum_t (n_t L^2 + n_t^2 L + n_t^3) + \sum_j (n_j^2 L + n_j L^2)))$. Figure 2.9 shows the running time of the algorithm on the Motes dataset versus the sequence length under missing as a block setting with different time series missing percentages and fully observed network information. We can see that the running time is almost linear to the sequence duration; and the algorithm is faster with sparser time series data.

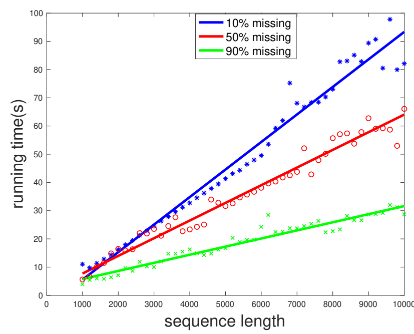


Figure 2.9: Running Time Versus the Sequence Length

Table 2.2: Symbols and Definitions

Symbol	Definitions and Descriptions
$\mathbf{A}$	matrix (bold upper case)
$\mathbf{A}_{ij}$	the element at row i and column j of matrix $\mathbf{A}$
$\mathbf{A}(:, j)$	the jth column of matrix $\mathbf{A}$
$\mathbf{A}(i, :)$	the ith row of matrix $\mathbf{A}$
$\mathbf{A}^T$	transpose of matrix $\mathbf{A}$
$\mathbf{a}$	column vector (bold lower case)
$\odot$	Hadamard product
$\mathbf{X}$	time series matrix
$\mathbf{M}$	indicator matrix for time series matrix
$\mathbf{S}$	embedded network matrix
$\tilde{\mathbf{M}}$	indicator matrix for embedded network matrix
$\mathbf{Z}$	time series latent matrix
$\mathbf{V}$	network latent matrix
$\mathbf{B}$	transition matrix
$\mathbf{U}$	object latent matrix
T	time duration
N	number of measurements in system
L	dimension of latent space
$\mathcal{N}(\mathbf{z}\|\mu, \mathbf{\Sigma})$	Gaussian distribution random variable
	with mean μ and covariance matrix $\mathbf{\Sigma}$

Algorithm 1: NetDyna

Input : Data set $\{\mathbf{X}, \mathbf{M}, \mathbf{S}, \tilde{\mathbf{M}}\}$, latent dimension L and network weight λ

Output: Parameter set $\boldsymbol{\theta}$ and estimated $\mathbf{X}, \mathbf{S}$

repeat
  for $t = 1 : T$ **do**
    Construct $\mathbf{H}_t$ based on Eq.(2.9);
    Estimate $\mu_t, \mathbf{\Psi}_t$ based on Eq.(2.11) and Eq.(2.12);
  end
  for $t = T : 1$ **do**
    Estimate $\tilde{\mu}_t, \tilde{\mathbf{\Psi}}_t$ based on Eq.(2.13);
    Estimate $E[\mathbf{Z}_t], E[\mathbf{Z}_t\mathbf{Z}'_{t-1}], E[\mathbf{Z}_t\mathbf{Z}'_t]$ based on Eq.(2.14);
  end
  for $j = 1 : N$ **do**
    Construct $\mathbf{G}_j$ based on Eq.(2.16);
    Estimate ν_j, γ_j based on Eq.(2.17);
    Estimate $E[\mathbf{v}_j], E[\mathbf{v}_j\mathbf{v}'_j]$ based on Eq.(2.18);
  end
  Update parameter set $\boldsymbol{\theta}$ based on Eq.(2.20);
until *converge*;
Set $\mathbf{Z} = (\tilde{\mu}_1, \cdots, \tilde{\mu}_T)$ and $\mathbf{V} = (\nu_1, \cdots, \nu_N)$;
Reconstruct $\mathbf{X} = \mathbf{M} \odot \hat{\mathbf{X}} + (\mathbb{1}_{N\times T} - \mathbf{M}) \odot \mathbf{U}'\mathbf{Z}$ and
$\mathbf{S} = \tilde{\mathbf{M}} \odot \hat{\mathbf{S}} + (\mathbb{1}_{N\times N} - \tilde{\mathbf{M}}) \odot \mathbf{V}'\mathbf{U}$;

Table 2.3: Root-Mean-Square-Error (RMSE) with the Motes Dataset under Missing as a Block

Missing	NetDyna	DCMF	DynaMMo	PMF	MSVD
(10%,0%)	0.1581	0.1604			
(10%,10%)	0.1628	0.2603	0.4190	0.9118	0.6000
(10%,20%)	0.1487	0.2696			
(50%,0%)	0.3191	0.3173			
(50%,10%)	0.3186	0.3765	0.6599	0.9736	0.7735
(50%,20%)	0.3082	0.4417			
(90%,0%)	0.6420	0.7238			
(90%,10%)	0.6198	0.6722	0.8990	1.0284	0.9834
(90%,20%)	0.6108	0.7239			

Table 2.4: Root-Mean-Square-Error (RMSE) with the Motes Dataset under Missing Uniformly Random

Missing	NetDyna	DCMF	DynaMMo	PMF	MSVD
(10%,0%)	0.0135	0.0133			
(10%,10%)	0.0134	0.0134	0.0140	0.0436	0.0257
(10%,20%)	0.0135	0.0133			
(50%,0%)	0.0141	0.0145			
(50%,10%)	0.0142	0.0148	0.0490	0.0614	0.0237
(50%,20%)	0.0142	0.0145			
(90%,0%)	0.0226	0.0243			
(90%,10%)	0.0234	0.0236	0.1684	0.4160	0.0214
(90%,20%)	0.0226	0.0237			

Table 2.5: Root-Mean-Square-Error (RMSE) with the Motion Capture Dataset under Missing as a Block

Missing	NetDyna	DCMF	DynaMMo	PMF	MSVD
(10%,0%)	0.0486	0.0487			
(10%,10%)	0.0476	0.0626	0.0676	0.8291	0.2350
(10%,20%)	0.0484	0.0838			
(30%,0%)	0.0808	0.1035			
(30%,10%)	0.0823	0.1206	0.1591	0.8443	0.2661
(30%,20%)	0.0867	0.1072			
(50%,0%)	0.1991	0.2520			
(50%,10%)	0.1813	0.2562	0.1854	0.8792	0.2797
(50%,20%)	0.1750	0.2654			
(90%,0%)	0.8720	0.9981			
(90%,10%)	0.7780	0.7241	0.3358	0.9618	0.3092

Table 2.6: Root-Mean-Square-Error (RMSE) with the Motion Capture Dataset under Missing Uniformly Random

Missing	NetDyna	DCMF	DynaMMo	PMF	MSVD
(10%,0%)	0.0139	0.0139			
(10%,10%)	0.0139	0.0139	0.0139	0.9995	0.0152
(10%,20%)	0.0140	0.0139			
(50%,0%)	0.0142	0.0142			
(50%,10%)	0.0142	0.0142	0.0145	0.9998	0.0177
(50%,20%)	0.0142	0.0142			
(90%,0%)	0.0267	0.0297			
(90%,10%)	0.0268	0.0299	0.1193	1.0010	0.0280
(90%,20%)	0.0260	0.0299			

Table 2.7: Root-Mean-Square-Error (RMSE) with the Motes and Motion Capture Datasets under One Time Series Is Entirely Missing

RMSE	NetDyna	DCMF	Average	Weighted Average
Motes(0%)	0.1133	0.1375	0.3603	0.3354
Motes(10%)	0.1154	0.1356	0.3637	0.3407
Motes(20%)	0.1155	0.2805	0.3627	0.3311
Motion(0%)	0.2248	0.2216	0.9287	0.7113
Motion(10%)	0.2185	0.2351	0.9247	0.7106
Motion(20%)	0.2183	0.3061	0.9386	0.7152

Chapter 3

LIGHT TRAFFIC CONVERGENT COEFFICIENT

In this chapter, we first introduce the original stochastic system that we are interested in, i.e. supermarket model with power-of-two-choices load balancing algorithm. The main interested of ours is that the stationary distribution of such a system in the steady state.

Then, we introduce the traditional mean-field model, of which the solution, i.e. the equilibrium point, is approximation of our interested quantity, the stationary distribution at steady state. Also, we will present the results from related literatures.

3.1 Load Balancing System

Consider a many-server system with N homogeneous servers, where job arrivals follows a Poisson process with rate λN and service times are i.i.d. exponential random variables with rate one. We consider $\lambda = 1 - \frac{\gamma}{N^\alpha}$ for some $0 < \gamma \leq 1$ and $0 \leq \alpha$. When $\alpha = 0$, λ is just a constant and we call it light traffic regime; When $\lambda > 0$, the arrival rate is depend on N and we call it heavy traffic regime.

We study a specific load balancing algorithm called power-of-two-choices algorithm.

Definition 1. *Power-of-Two-Choices(Po2): Po2 samples 2 servers uniformly at random and dispatches the job to the least loaded server among the 2 servers.*

Let $S_i(t)$ denote the fraction of servers with queue size at least i at time t. Under the finite buffer assumption with buffer size b, $S_i = 0, \forall i \geq b+1$. Define $\mathcal{S}$ to be

$$\mathcal{S} = \{s | 1 \geq s_1 \geq \cdots \geq s_b \geq 0\},$$

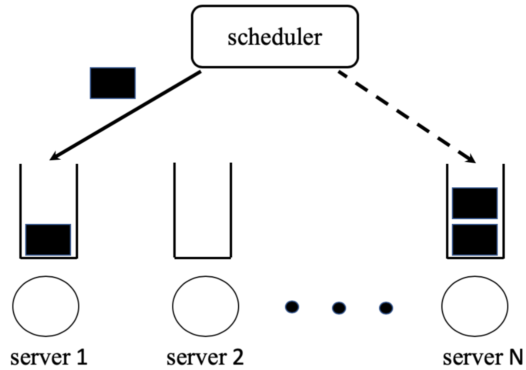


Figure 3.1: Po2 Load Balancing Algorithm

and $S(t) = [S_1(t), S_2(t), \cdots, S_b(t)]$. S is a continuous time Markov chain(CTMC) with the following transition rate

$$R_{ss'} = \begin{cases} N(s_k - s_{k+1}), & \text{if } s' = s - \frac{1_k}{N} \text{ and } 1 \leq k \leq b-1 \\ Ns_b, & \text{if } s' = s - \frac{1_b}{N} \\ \lambda N(s_{k-1}^2 - s_k^2), & \text{if } s' = s + \frac{1_k}{N} \\ \sum_{k=1}^{b} -\lambda N(s_{k-1}^2 - s_k^2) - N(s_k - s_{k+1}), & \text{if } s' = s \\ 0, & \text{otherwise} \end{cases}$$

where 1_k is a $b-$dimensional vector such that the kth element is 1 and the others are 0. Note that the first and second terms are a departure occurs with size k so s_k decreases by $\frac{1}{N}$. The third term is for the event that an arrival occurs and it is routed to a queue size $k-1$. Define a normalized transition rate

$$q_{ss'} = \frac{1}{N} R_{ss'}$$

Define the traditional mean-field model Ying (2017), we have

$$\dot{s} = f(s) = \sum_{s':s' \neq s} R_{ss'}(s' - s) = N \sum_{s':s' \neq s} q_{ss'}(s' - s)$$

more specifically

$$\dot{s}_k = f_k(s) = \begin{cases} \lambda(s_{k-1}^2 - s_k^2) - (s_k - s_{k+1}), & b-1 \leq k \leq 1 \\ \lambda(s_{b-1}^2 - s_b^2) - s_b, & k = b. \end{cases}$$

Denote the equilibrium point as s^* and it satisfies the conditions:

$$s_0^* = 1$$

$$\lambda[(s_{k-1}^*)^2 - (s_k^*)^2) - (s_k^* - s_{k+1}^*) = 0 \quad 1 \leq k \leq b-1$$

$$\lambda[(s_{b-1}^*)^2 - (s_b^*)^2] - s_b^* = 0$$

The existence and uniqueness of the equilibrium point has been proved in Mitzenmacher (1996). And define

$$g(s) = -\int_0^\infty d(s(t), s^*)dt, \quad s(0) = s.$$

where $d(s(t), s^*)$ is a distant function. Then the Stein's Equation holds

$$E[d(S(\infty), s^*)] = -E[\sum_{s'} R_{S(\infty),s'}\Gamma(S(\infty), s')]$$

where $\Gamma(s, s') = g(s') - g(s) - \nabla g(s) \cdot (s' - s)$. Also by the definition of $g(s)$, the Poisson equation holds

$$\nabla g(s) \cdot f(s) = d(s, s^*)$$

In the previous results Ying (2016, 2017), we only have order-wise convergence results of mean square error for the stationary distribution, i.e. $S(\infty)$ and mean-field solution, i.e. s^*. However, based on the order-wise results, we can actually calculate the convergence constant in front of the convergence rate.

3.2 Main Idea

The previous results tell us that as N increases, the state space will concentrate around the mean-field solution. Thus, we can divide the state space into two regions

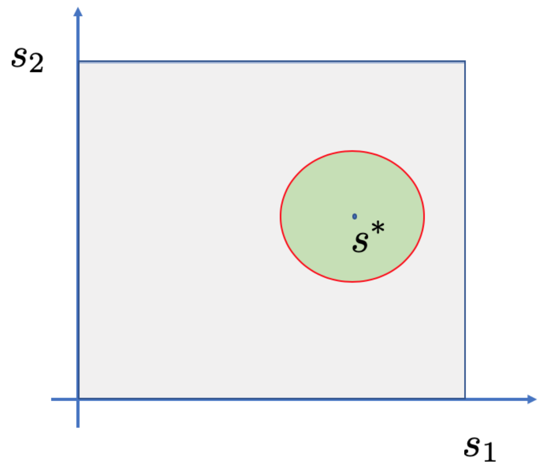


Figure 3.2: $b = 2$ Illustration of Inside Region and Outside Region

based on the closeness to the mean-field solution, i.e. the region close to the equilibrium point and the region far from the equilibrium point. And for the two regions, we apply different techniques. The ideas are following:

1)We first establish higher moments bound for mean field approximation. The purpose is to bound the probability of the states being far away from equilibrium point so that make the probability suitably small.

2)And then for states closer to equilibrium point, instead of the traditional mean-field model, we use a linearized mean-field model to approximate the generator of the original stochastic system. By carefully choosing the parameters, i.e. the order of higher moment and the size of ball, which defines the closeness from equilibrium point, we can look into the generator and calculate the coefficient in front of convergence rate.

For example in the case of $b = 2$, we have the state space to be $\{(\frac{p}{N}, \frac{q}{N})|p, q \in \{0, 1, \cdots, N\}\} \subset [0, 1]^2$. As in Figure, 3.2, we split the state space into two regions by the red circle. The grey area outside of red circle is the outside region. And the green area inside the red circle is the inside region. And the size of the circle is the parameter of our design.

Our main idea is that, in steady state, the probability of state being in outside region, i.e. grey area, is small; on the basis of that, we choose a simple linear function to approximate the generator of the stochastic system in the inside region, i.e. the green area. By doing so, we can more precisely quantify the stead state when N is sufficiently large.

3.3 Higher Moment Bounds

In this section, we assume the arrival rate λ is a constant and independent of system size N. First, we establish higher moment bounds, which will be utilized to show that the probability of being far from equilibrium point is small.

Let $g(s)$ be the solution to the Poisson equation, for $r \in \mathbb{N}$,

$$\nabla g(s) \cdot \dot{s} = \nabla g(s) \cdot f(s) = ||s - s^*||^{2r} \tag{3.1}$$

Then, the solution has the following form

$$\begin{aligned} g(s) &= -\int_0^\infty ||s(t) - s^*||^{2r} dt \\ &= -\int_0^\infty [\sum_{i=1}^{b} (s_i(t) - s_i^*)^2]^r dt \end{aligned}$$

We have for the stationary distribution of S

$$E[Gg(S)] = E[N \sum_{s' \neq S} R_{S,s'}(s)(g(s') - g(S))] = 0 \tag{3.2}$$

Then, we will have

$$\begin{aligned} &E[||S - s^*||^{2r}] \\ =&E[\nabla g(S) \cdot f(S) - N \sum_{s' \neq S} q_{S,s'}(g(s') - g(S))] \end{aligned}$$

$$=E[\nabla g(S) \cdot f(S) - \nabla g(S) \cdot \sum_{s' \neq S} q_{S,s'} N(s' - S) + \nabla g(S) \cdot \sum_{s' \neq S} q_{S,s'} N(s' - S)$$

$$- N \sum_{s' \neq S} q_{S,s'} (g(s') - g(S))]$$

$$=E[\nabla g(S) \cdot \left(f(S) - \sum_{s' \neq S} q_{S,s'} N(s' - S) \right) - \sum_{s' \neq S} q_{S,s'} N(g(s') - g(S) - \nabla g(S) \cdot (s' - S))]$$

$$=E[- \sum_{s' \neq S} q_{S,s'} N(g(s') - g(S) - \nabla g(S) \cdot (s' - S))]$$

where $g(s) = -\int_0^\infty [\sum_{i=1}^b (s_i(t) - s_i^*)^2]^r dt$.

We next focus on the following term

$$\begin{aligned}
& -(g(s') - g(s) - \nabla g(s) \cdot (s' - s)) \\
= & \int_0^\infty [\sum_{i=1}^b (s_i(t, s') - s_i^*)^2]^r dt - \int_0^\infty [\sum_{i=1}^b (s_i(t, s) - s_i^*)^2]^r dt \\
& - 2r(s' - s) \cdot \int_0^\infty [\sum_{i=1}^b (s_i(t, s) - s_i^*)^2]^{r-1} \sum_{i=1}^b (s_i(t, s) - s_i^*) \nabla s_i(t, s) dt \\
= & \int_0^\infty \Bigg([\sum_{i=1}^b (s_i(t, s') - s_i^*)^2]^r - [\sum_{i=1}^b (s_i(t, s) - s_i^*)^2]^r dt \\
& -2r(s' - s) \cdot [\sum_{i=1}^b (s_i(t, s) - s_i^*)^2]^{r-1} \sum_{i=1}^b (s_i(t, s) - s_i^*) \nabla s_i(t, s) \Bigg) dt \qquad (3.3)
\end{aligned}$$

Let

$$e_i(t) = s_i(t, s') - s_i(t, s) - \nabla s_i(t, s) \cdot (s' - s)$$

i.e.,

$$s_i(t, s') = e_i(t) + s_i(t, s) + \nabla s_i(t, s) \cdot (s' - s)$$

so

$$\begin{aligned}
& (s_i(t, s') - s_i^*)^2 \\
= & (e_i(t) + s_i(t, s) + \nabla s_i(t, s) \cdot (s' - s) - s_i^*)^2
\end{aligned}$$

$$
\begin{aligned}
=&(s_i(t,s)-s_i^*)^2+e_i^2(t)+[\nabla s_i(t,s)\cdot(s'-s)]^2\\
&+2e_i(t)(s_i(t,s)-s_i^*)+2(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)+2e_i(t)\nabla s_i(t,s)\cdot(s'-s)
\end{aligned}
$$

By summing over all i, we have

$$
\begin{aligned}
&\sum_{i=1}^{b}(s_i(t,s')-s_i^*)^2\\
=&\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2+\sum_{i=1}^{b}e_i^2(t)+\sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2+2\sum_{i=1}^{b}e_i(t)(s_i(t,s)-s_i^*)\\
&+2\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)+2\sum_{i=1}^{b}e_i(t)\nabla s_i(t,s)\cdot(s'-s)
\end{aligned}
$$

By raising to the rth order, we have

$$
\begin{aligned}
&[\sum_{i=1}^{b}(s_i(t,s')-s_i^*)^2]^r\\
=&[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2+\sum_{i=1}^{b}e_i^2(t)+\sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2+2\sum_{i=1}^{b}e_i(t)(s_i(t,s)-s_i^*)\\
&+2\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)+2\sum_{i=1}^{b}e_i(t)\nabla s_i(t,s)\cdot(s'-s)]^r\\
=&\sum_{\sum_{k=1}^{6}r_k=r,r_k\geq 0}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}[\sum_{i=1}^{b}e_i^2(t)]^{r_2}[\sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2]^{r_3}[2\sum_{i=1}^{b}e_i(t)(s_i(t,s)-s_i^*)]^{r_4}\\
&[2\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)]^{r_5}[2\sum_{i=1}^{b}e_i(t)\nabla s_i(t,s)\cdot(s'-s)]^{r_6}
\end{aligned}
$$

Note that when $r_1=r$ and $r_i=0$ for $i=2,\cdots,6$, the summand is $[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^r$, which is the inside term of second integral equation (3.3); and when $r_1=r-1, r_5=1$ and $r_i=0$ for $i=2,3,4,6$, the summand is $2r[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r-1}\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)$, which is the inside term of third integral equation (3.3). These terms will cancel out after taking integrals, thus we only consider the case $\{r_i\}_{1,\cdots,6}$ other than the two cases above. Let Σ be the collection of combination of all $\{r_i\}_{i=1,\cdots,6}$ that excludes above two cases, i.e.

$$
\begin{aligned}
\Sigma = &\{r_i, i = 1, \cdots, 6 | \sum_{i=1}^{6} = r\} \\
&\setminus \{\{r_1 = r, r_i = 0, \text{for } i = 2, \cdots, 6\}, \{r_1 = r-1, r_5 = 1, r_i = 0, \text{for } i = 2, 3, 4, 6\}\}.
\end{aligned}
$$

We have that, from Ying (2016),

$$|e_i(t)| \leq ||\mathbf{e}(t)|| = O(\frac{1}{N^2})$$

and the state transition condition

$$||s' - s|| = \frac{1}{N}$$

Also, we can show that

$$||\nabla s_i(t, s)|| \leq \tilde{c}$$

where $\tilde{c}$ is independent of system size N, the proofs are the same as the case in light traffic with constant buffer size.

Then, for the term inside integral, we have

$$
\begin{aligned}
&[\sum_{i=1}^{b} (s_i(t, s') - s_i^*)^2]^r - [\sum_{i=1}^{b} (s_i(t, s) - s_i^*)^2]^r \\
&- 2r(s' - s) \cdot [\sum_{i=1}^{b} (s_i(t, s) - s_i^*)^2]^{r-1} \cdot \sum_{i=1}^{b} (s_i(t, s) - s_i^*) \nabla s_i(t, s) \\
= &\sum_{\Sigma} \binom{r}{r_1} \binom{r - r_1}{r_2} \binom{r - r_1 - r_2}{r_3} \binom{r - r_1 - r_2 - r_3}{r_4} \binom{r - r_1 - r_2 - r_3 - r_4}{r_5} \\
&[\sum_{i=1}^{b} (s_i(t, s) - s_i^*)^2]^{r_1} [\sum_{i=1}^{b} e_i^2(t)]^{r_2} [\sum_{i=1}^{b} [\nabla s_i(t, s) \cdot (s' - s)]^2]^{r_3} [2 \sum_{i=1}^{b} e_i(t)(s_i(t, s) - s_i^*)]^{r_4} \\
&[2 \sum_{i=1}^{b} (s_i(t, s) - s_i^*) \nabla s_i(t, s) \cdot (s' - s)]^{r_5} [2 \sum_{i=1}^{b} e_i(t) \nabla s_i(t, s) \cdot (s' - s)]^{r_6} \\
\leq &\sum_{\Sigma} \binom{r}{r_1} \binom{r - r_1}{r_2} \binom{r - r_1 - r_2}{r_3} \binom{r - r_1 - r_2 - r_3}{r_4} \binom{r - r_1 - r_2 - r_3 - r_4}{r_5}
\end{aligned}
$$

$$
\begin{aligned}
&[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}[\sum_{i=1}^{b}e_i^2(t)]^{r_2}[\sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2]^{r_3}[2\sum_{i=1}^{b}|e_i(t)(s_i(t,s)-s_i^*)|]^{r_4} \\
&[2\sum_{i=1}^{b}|(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)|]^{r_5}[2\sum_{i=1}^{b}|e_i(t)\nabla s_i(t,s)\cdot(s'-s)|]^{r_6} \\
\leq&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5} \\
&[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}[\sum_{i=1}^{b}e_i^2(t)]^{r_2}[\sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2]^{r_3}[2\sum_{i=1}^{b}|e_i(t)|\cdot|s_i(t,s)-s_i^*|]^{r_4} \\
&[2\sum_{i=1}^{b}|s_i(t,s)-s_i^*|\cdot|\nabla s_i(t,s)\cdot(s'-s)|]^{r_5}[2\sum_{i=1}^{b}|e_i(t)|\cdot|\nabla s_i(t,s)\cdot(s'-s)|]^{r_6} \\
\leq&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5} \\
&[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2}})O(\frac{1}{N^{2r_3}})O(\frac{1}{N^{(2-2\alpha-4\xi)r_4}})[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_4} \\
&O(\frac{1}{N^{r_5}})[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_5}O(\frac{1}{N^{(3-2\alpha-4\xi)r_6}}) \\
\leq&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5} \\
&O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6}})[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_4+r_5}
\end{aligned}
\tag{3.4}
$$

Then, substitute into the following equation

$$
\begin{aligned}
&-(g(s')-g(s)-\nabla g(s)\cdot(s'-s)) \\
=&\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s')-s_i^*)^2]^r dt-\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^r dt \\
&\quad-2r(s'-s)\cdot\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r-1}\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)dt \\
\leq&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}
\end{aligned}
$$

$$
\begin{aligned}
&O(\frac{1}{N^{4r_2+2r_3+2r_4+r_5+3r_6}})\int_0^\infty [\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_4+r_5}dt\\
\leq&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{1}{N^{4r_2+2r_3+2r_4+r_5+3r_6}})\int_0^\infty (||s-s^*||\kappa e^{-\alpha t})^{2r_1}[b\kappa||s-s^*||e^{-\alpha t}]^{r_4+r_5}dt\\
=&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{1}{N^{4r_2+2r_3+2r_4+r_5+3r_6}})b^{r_4+r_5}k^{2r_1+r_4+r_5}\frac{1}{\alpha(2r_1+r_4+r_5)}||s-s^*||^{2r_1+r_4+r_5}\\
=&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{b^{r_4+r_5}}{N^{4r_2+2r_3+2r_4+r_5+3r_6}})||s-s^*||^{2r_1+r_4+r_5}.
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&E[||S-s^*||^{2r}]\\
=&E[-\sum_{s'\neq S}q_{S,s'}N(g(s')-g(S)-\nabla g(S)\cdot(s'-S))]\\
\leq&E[\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{b^{r_4+r_5}}{N^{4r_2+2r_3+2r_4+r_5+3r_6-1}})||S-s^*||^{2r_1+r_4+r_5}\sum_{s'\neq S}q_{S,s'}]\\
=&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{b^{r_4+r_5}}{N^{4r_2+2r_3+2r_4+r_5+3r_6-1}})E[||S-s^*||^{2(r_1+\frac{r_4+r_5}{2})}]\\
\leq&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{b^{r_4+r_5}}{N^{4r_2+2r_3+2r_4+r_5+3r_6-1}})O(\frac{1}{N^{r_1+\frac{r_4+r_5}{2}}})\\
=&\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}\\
&O(\frac{b^{r_4+r_5}}{N^{r_1+4r_2+2r_3+2.5r_4+1.5r_5+3r_6-1}})
\end{aligned}
$$

$$=\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$
$$O(\frac{b^{r_4+r_5}}{N^{r+3r_2+r_3+1.5r_4+0.5r_5+2r_6-1}})$$
$$=O(\frac{1}{N^{r(1-\xi)}})$$

where ξ can be chosen arbitrarily small for sufficiently large N. The last equality happens when $r_1 = r-1, r_3 = 1$ and the rest is 0, i.e. $r_i = 0$ for $i = 2,4,5,6$ and $r_1 = r-2, r_5 = 2$ and the rest is 0, i.e. $r_i = 0$ for $i = 2,3,4,6$. Note that we used the assumption that $\sum_{s'\neq s} q_{s,s'} < 2$. The second inequality, we assumed that $E[||S-s^*||^{2(r-1)}] \leq O(\frac{1}{N^{(r-1)(1-\xi)}})$ and by Lyapunov inequality we have $E[||S-s^*||^{2(r_1+\frac{r_4+r_5}{2})}] \leq (E[||S-s^*||^{2(r-1)}])^{\frac{2(r_1+\frac{r_4+r_5}{2})}{r-1}}$.

When $r_1 = r_4 = r_5 = 0$, we have similar analysis based on the fact that $\int_0^\infty ||e(t)||dt = O(\frac{1}{N^2})$.

3.4 State Space Collapse

Let ρ be a positive number. Hence, applying Markov inequality, we have

$$\begin{aligned}\mathbb{P}\{||S-s^*||^{2r} \geq \epsilon\} &\leq \frac{\mathbb{E}[||S-s^*||^{2r}]}{\rho}\\ &= \frac{1}{\rho}O(\frac{1}{N^{r(1-\xi)}})\end{aligned}$$

Let $\rho = \frac{1}{N^\epsilon}$, where $\epsilon > 0$. Then, the above inequality will become

$$\mathbb{P}\{||S-s^*||^{2r} \geq \frac{1}{N^\epsilon}\} \leq O(\frac{1}{N^{r(1-\xi)-\epsilon}}) \tag{3.5}$$

3.5 Linear Mean-Field Model

Define a set of states to be $\mathcal{B} = \{s \mid ||s-s^*||^{2r} \leq \frac{1}{N^\epsilon}\}$, which is close to equilibrium point. Let $d(s,s^*) = ||s-s^*||^2$ as the distance function. We consider a simple linear system

$$\dot{s} = l(s) = J(s^*)(s-s^*), \tag{3.6}$$

where $J(s^*)$ is the Jacobian matrix of $f(s)$ at the equilibrium point s^*.

The Jacobian matrix at a point s, is following

$$J(s) = \begin{bmatrix} -2\lambda s_1 - 1 & 1 & & 0 \\ 2\lambda s_1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 2\lambda s_{b-1} & -2\lambda s_b - 1 \end{bmatrix}$$

We first introduce a lemma that matrix $J(s^*)$ is invertible, i.e. $J(s^*)^{-1}$ exists.

Lemma 1 (Invertibility)**.** *For any $s \in \mathcal{S}$, the Jacobian matrix $J(s)$ is invertible.*

Proof. Since it's a tridiagonal matrix, we can write down the determinant in a recursive form

$$P_i = -(2\lambda s_i + 1)P_{i-1} - 2\lambda s_{i-1}P_{i-2}$$

where

$$P_i = \begin{vmatrix} -2\lambda s_1 - 1 & 1 & & 0 \\ 2\lambda s_1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 2\lambda s_{i-1} & -2\lambda s_i - 1 \end{vmatrix}$$

Furthermore, we can verify that in fact, P_i can be written in the following form

$$P_i = (-1)^i - 2\lambda s_i P_{i-1} \tag{3.7}$$

with $P_1 = -(2\lambda s_1 + 1)$. We can conclude 2 things from equation (5.5), for any $s \in \mathcal{S}$,

- the sign of P_i alternates: when i is odd, $P_i < 0$; when i is even, $P_i > 0$.
- the absolute value of P_i is no less than 1, i.e. $|P_i| \geq 1$

Because the determinant is nonzero, $J(s)$ is invertible. $\square$

Consider a function $g : \mathcal{S} \to \mathcal{S}$ such that it satisfies the following equation

$$Lg(s) \doteq \frac{dg(s)}{dt} = \nabla g(s) \cdot l(s) = ||s - s^*||^2. \tag{3.8}$$

According to the definition of the linear mean-field model in (3.6), we have

$$\nabla g(s) \cdot J(s^*)(s - s^*) = ||s - s^*||^2. \tag{3.9}$$

Lemma 2 (Solution of Poisson Equation)**.** *The solution of the Poisson equation* (3.9) *is*

$$\nabla g(s) = [J^T(s^*)]^{-1}(s - s^*) \tag{3.10}$$

and furthermore

$$\nabla^2 g(s) = [J^T(s^*)]^{-1}$$
$$\nabla^3 g(s) = 0$$

□

Proof. According to Poisson equation (3.9), we have

$$\nabla g(s)^T J(s^*)(s - s^*) = (s - s^*)^T(s - s^*).$$

It implies

$$[\nabla g(s)^T J(s^*) - (s - s^*)^T](s - s^*) = 0$$

holds for all s in the neighborhood of s^*. If we set

$$\nabla g(s)^T J(s^*) - (s - s^*)^T = 0$$

then, we can solve that

$$\nabla g(s) = [J^T(s^*)]^{-1}(s - s^*)$$

and the higher derivatives as in the lemma. □

3.6 Stein's Method

We have the generator difference as following

$$\mathbb{E}[d(S,s^*)\big|||S-s^*||^{2r}\leq\frac{1}{N^\epsilon}]=\mathbb{E}[Lg(S)-Gg(S)\big|||S-s^*||^{2r}\leq\frac{1}{N^\epsilon}]$$

We will expand the terms in generator equation and analyze the terms one by one.

3.6.1 Generator for the Stochastic System

Lemma 3. *The generator of function $g(s)$ is following*

$$Gg(s)=\nabla g(s)\cdot f(s)+\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g(s)_{ii}\tilde{f}_i(s) \tag{3.11}$$

where $\nabla^2 g(s)_{ii}$ is the i-th diagonal element of the Hessian matrix $\nabla^2 g(s)$ and $\tilde{f}_i(s)=\frac{1}{2}[\lambda(s_{i-1}^2-s_i^2)+(s_i-s_{i+1})]$.

Proof. Take a closer look at the generator by Taylor expansion around a state s

$$\begin{aligned}
&Gg(s)\\
&=\sum_{i=1}^{b}\lambda N(s_{i-1}^2-s_i^2)[g(s+e_i)-g(s)]+N(s_i-s_{i+1})[g(s-e_i)-g(s)]\\
&=\sum_{i=1}^{b}\lambda N(s_{i-1}^2-s_i^2)[\nabla g(s)\cdot e_i+\frac{1}{2}e_i^T\nabla^2 g(s)e_i]\\
&\quad+N(s_i-s_{i+1})[\nabla g(s)\cdot(-e_i)+\frac{1}{2}e_i^T\nabla^2 g(s)e_i]\\
&=\sum_{i=1}^{b}\nabla g(s)\cdot[\lambda(s_{i-1}^2-s_i^2)-(s_i-s_{i+1})]Ne_i\\
&\quad+\frac{1}{2}Ne_i^T\nabla^2 g(s)e_i[\lambda(s_{i-1}^2-s_i^2)+(s_i-s_{i+1})]\\
&=\nabla g(s)\cdot f(s)+\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g(s)_{ii}\tilde{f}_i(s)
\end{aligned}$$

Note that the second equality is because $\nabla^3 g(s)=0$ from Lemma 2. □

3.6.2 Function $\tilde{f}(s)$

Since when s is close to equilibrium point s^*, define $x_i = s_i - s_i^*$, we can further make a Taylor expansion at the equilibrium point s^* as following

$$\begin{aligned}
\tilde{f}_i(s) &= \frac{1}{2}[\lambda(s_{i-1}^2 - s_i^2) + (s_i - s_{i+1})] \\
&= \frac{\lambda}{2}[(s_{i-1}^* + x_{i-1})^2 - (s_i^* + x_i)^2] + \frac{1}{2}(s_i^* + x_i - s_{i+1}^* - x_{i+1}) \\
&= \frac{\lambda}{2}[(s_{i-1}^*)^2 + 2x_{i-1}s_{i-1}^* + x_{i-1}^2 - (s_i^*)^2 - 2x_i s_i^* - x_i^2] + \frac{1}{2}(s_i^* + x_i - s_{i+1}^* - x_{i+1}) \\
&= \frac{\lambda}{2}[(s_{i-1}^*)^2 - (s_i^*)^2] + \frac{1}{2}(s_i^* - s_{i+1}^*) + O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \\
&= \tilde{f}_i(s^*) + O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \qquad (3.12)
\end{aligned}$$

The last equality is because of the fact that $||s - s^*||^{2r} \leq \frac{1}{N^\epsilon}$, which implies $|x_i| \leq \frac{1}{N^{\frac{\epsilon}{2r}}}$.

3.6.3 Generator Difference

The mean square error close to the equilibrium point is

$$\begin{aligned}
&\mathbb{E}[d(S, s^*) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[Lg(S) - Gg(S) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[\nabla g(S) \cdot J(s^*)(S - s^*) - \nabla g(S) \cdot f(S) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii}\tilde{f}_i(S) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[\nabla g(S) \cdot (J(s^*)(S - s^*) - f(S)) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii}\left(\tilde{f}_i(s^*) + O(\frac{1}{N^{\frac{\epsilon}{2r}}})\right) \\
&\mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[\nabla g(S) \cdot (J(s^*)(S - s^*) - f(s)) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii}\tilde{f}_i(s^*) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \\
&\mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}].
\end{aligned}$$

According to Lemma 2, we have

$$\nabla g(s) = [J^T(s^*)]^{-1}(s - s^*),$$

$$\nabla^2 g(s) = [J^T(s^*)]^{-1}$$

to be the functions of $J(s^*)$, which is a function of N.

And in fact, since the traditional mean-field is a second-order system, we can write it down as

$$f(s) = f(s^*) + J(s^*)(s - s^*) + \frac{1}{2} < s - s^*, \nabla^2 f(s^*)(s - s^*) >$$

where $\nabla^2 f(s^*)$ is the Hessian of $f(s)$ at equilibrium point. For any $s \in \mathcal{S}$ and $i = 1, \cdots, b$, the Hessian has following form for f_i

$$\nabla^2 f_i(s)_{kj} = \frac{\partial^2 f_i(s)}{\partial s_j \partial s_k} = \begin{cases} -2\lambda, & \text{if } j = k = i \\ 2\lambda, & \text{if } j = k = i - 1 \\ 0, & \text{otherwise} \end{cases}$$

Substituting back to generator difference, we have

$$\begin{aligned} & \mathbb{E}[d(S, s^*) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\ =& \mathbb{E}[Lg(S) - Gg(S) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\ =& \mathbb{E}\big[[J^T(s^*)]^{-1}(S - s^*) \cdot (\frac{1}{2} < S - s^*, \nabla^2 f(s^*)(S - s^*) >) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} \tilde{f}_i(s^*) \\ & - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}\big]. \end{aligned} \tag{3.13}$$

There are 3 terms in equation (3.13). We will characterize the upper bounds of each term and a lower bound for the second term in equation (3.13).

Lemma 4 (Upper Bound on the First Term)**.** *For the first term in equation (3.13), we have an upper bound on its 2-norm, when* $||s - s^*||^{2r} \leq \frac{1}{N^\epsilon}$*,*

$$||[J^T(s^*)]^{-1}(s - s^*) \cdot < s - s^*, \nabla^2 f(s^*)(s - s^*) > || = O(\frac{1}{N^{\frac{3\epsilon}{2r}}}) \tag{3.14}$$

Proof. We have the 2-norm of the first term as following

$$
\begin{aligned}
& ||[J^T(s^*)]^{-1}(s-s^*)\cdot < s-s^*, \nabla^2 f(s^*)(s-s^*) > || \\
\leq\ & ||[J^T(s^*)]^{-1}(s-s^*)|| || < s-s^*, \nabla^2 f(s^*)(s-s^*) > || \\
\leq\ & ||[J^T(s^*)]^{-1}|| ||s-s^*|| || < s-s^*, \nabla^2 f(s^*)(s-s^*) > || \\
\leq\ & 2\sqrt{2}\lambda ||[J^T(s^*)]^{-1}|| ||s-s^*||^3
\end{aligned}
\tag{3.15}
$$

where the third inequality is because of followings

$$
\begin{aligned}
& || < s-s^*, \nabla^2 f(s^*)(s-s^*) > || \\
=& \sqrt{\sum_{i=1}^{b} [(s-s^*)\nabla^2 f_i(s^*)(s-s^*)]^2} \\
=& \sqrt{\sum_{i=1}^{b} \left(2\lambda[(s_{i-1}-s_{i-1}^*)^2 - (s_i - s_i^*)^2]\right)^2} \\
=& 2\lambda\sqrt{\sum_{i=1}^{b} [(s_{i-1}-s_{i-1}^*)^2 - (s_i - s_i^*)^2]^2} \\
\leq& 2\lambda\sqrt{\sum_{i=1}^{b} (s_{i-1}-s_{i-1}^*)^4 + (s_i - s_i^*)^4} \\
\leq& 2\sqrt{2}\lambda\sqrt{\sum_{i=1}^{b} (s_i - s_i^*)^4} \\
\leq& 2\sqrt{2}\lambda\sqrt{[\sum_{i=1}^{b} (s_i - s_i^*)^2]^2} \\
=& 2\sqrt{2}\lambda ||s-s^*||^2
\end{aligned}
$$

Note that for light traffic, the entries of the Jacobian matrix $J(s^*)$ are a constant that is independent of system size N. Since $||s-s^*||^{2r} \leq \frac{1}{N^\epsilon}$, combining with the inequality (3.15), we have

$$
||[J^T(s^*)]^{-1}(s-s^*)\cdot < s-s^*, \nabla^2 f(s^*)(s-s^*) > ||
$$

$$\leq 2\sqrt{2}\lambda \times O(1) \times \frac{1}{N^{\frac{3\epsilon}{2r}}} = O(\frac{1}{N^{\frac{3\epsilon}{2r}}})$$

□

Lemma 5 (Lower Bound on the Second Term)**.** *For the second term in equation (3.13), we have a lower bound as following*

$$-\frac{1}{N}\sum_{i=1}^{b} \nabla^2 g_{ii}(s)\tilde{f}_i(s^*) \geq \frac{\lambda(1-\lambda^2)}{3N} \tag{3.16}$$

Proof. It's easy to check, for $i = 1, \cdots, b$, $\tilde{f}_i(s^*) \geq 0$. From lemma 21, $\nabla^2 g(s)_{ii} < 0$, for $i = 1, \cdots, b$, so we have

$$-\nabla^2 g(s)_{ii}\tilde{f}_i(s^*) \geq 0$$

And we also have

$$\begin{aligned}\tilde{f}_i(s^*) &= \frac{1}{2}[\lambda((s^*_{i-1})^2 - (s^*_i)^2) + (s^*_i - s^*_{i+1})] \\ &= \lambda[(s^*_{i-1})^2 - (s^*_i)^2]\end{aligned}$$

the second equality is because s^* is the equilibrium point. Thus, for $i = 1$, we have

$$\tilde{f}_1(s^*) = \lambda[1 - (s^*_1)^2] \geq \lambda(1-\lambda^2)$$

So, we have

$$\frac{1}{N}\sum_{i=1}^{b} \nabla^2 g_{ii}(s)\tilde{f}_i(s^*) \geq -\frac{1}{N}J^{-1}_{11}(s^*)\tilde{f}_1(s^*) \geq \frac{\lambda(1-\lambda^2)}{3N}$$

□

Lemma 6 (Upper Bound on the Second Term)**.** *For the second term in equation (3.13), we have an upper bound as following*

$$-\frac{1}{N}\sum_{i=1}^{b} \nabla^2 g_{ii}(s)\tilde{f}_i(s^*) = O(\frac{1}{N^{1-\xi}}) \tag{3.17}$$

Proof. It's easy to check that $f_i(s^*) \leq 1$ for $i = 1, \cdots, b$. Therefore

$$-\frac{1}{N}\sum_{i=1}^{b} \nabla^2 g_{ii}(s)\tilde{f}_i(s^*) = \frac{b}{N}O(1) = O(\frac{1}{N^{1-\xi}})$$

The last equality is from the assumption $b = O(\log N)$. □

Lemma 7 (Upper Bound on the Third Term)**.** *For sufficiently large N, we have an upper bound for the third term in equation (3.13) as following*

$$|| - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(s)_{ii} O(\frac{1}{N^{\frac{\epsilon}{2r}}})|| = O(\frac{1}{N^{1+\frac{\epsilon}{2r}}}) \tag{3.18}$$

Proof. We have

$$|| - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(s)_{ii} O(\frac{1}{N^{\frac{\epsilon}{2r}}})|| \leq \frac{b}{N}O(1) \cdot O(\frac{1}{N^{\frac{\epsilon}{2r}}}) = O(\frac{1}{N^{1+\frac{\epsilon}{2r}}})$$

□

Based on these lemmas, we are now able to characterize the dominant term among the three in equation (3.13).

Lemma 8. *For sufficiently large N, we have for equation (3.13)*

$$E[d(S, s^*)\big| ||S - s^*||^{2r} \leq \frac{1}{N^{\epsilon}}] = -\frac{1}{N}\sum_{i=1}^{b} [J^T(s^*)]^{-1}_{ii} \tilde{f}_i(s^*) + o(\frac{1}{N}) \tag{3.19}$$

where the parameters satisfy following

$$r = 4 \quad , \quad \frac{8}{3} < \epsilon < 3$$

Proof. For the given condition of the parameters, it's easy to check that the upper bounds of the first and third term in equation (3.13) is order-wise smaller than the lower bounds of the second term, i.e.

$$\frac{3\epsilon}{2r} > 1$$

and

$$1 + \frac{\epsilon}{2r} > 1$$

which show that the lower bound of the second term is order-wise larger than upper bounds of the first and third terms. □

3.7 Main Results

In this section, we present our main results using 3 theorems. These are the results based on the observation that the second term in equation (3.13) is the dominant term.

Theorem 3 (Heavy Traffic Convergence)**.** *For sufficiently large N, we have that*

$$\mathbb{E}[||S - s^*||^2] = -\frac{1}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) + o(\frac{1}{N}) \tag{3.20}$$

where $J(s^*)$ is the Jacobian matrix of traditional mean-field model $f(s)$ at equilibrium point s^* and $\tilde{f}_i(s^*) = \frac{1}{2}[\lambda\left((s_{i-1}^*)^2 - (s_i^*)^2\right) + (s_i^* - s_{i+1}^*)]$ for $i = 1, 2, \cdots, b$.

Proof. Let parameters satisfy following conditions

$$r = 4 \qquad , \quad \epsilon = \frac{17}{6}$$

and $\xi > 0$ is arbitrarily small. Then, for sufficiently large N, the mean square distance is

$$\begin{aligned}
&E[||S - s^*||^2] \\
=&E[||S - s^*||^2 \Big| ||S - s^*||^{2r} \geq \frac{1}{N^\epsilon}] \cdot \mathbb{P}\{||S - s^*||^{2r} \geq \frac{1}{N^\epsilon}\} \\
&+ E[||S - s^*||^2 \Big| ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \cdot \mathbb{P}\{||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}\} \\
=&O(\log N) \cdot O(\frac{1}{N^{r(1-\xi)-\epsilon}}) + [-\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g(s)_{ii}\tilde{f}_i(s^*) + o(\frac{1}{N})] \cdot [1 - O(\frac{1}{N^{r(1-\xi)-\epsilon}})] \\
=&O(\frac{1}{N^{r(1-\xi)-\epsilon-\xi}}) - \frac{1}{N}\sum_{i=1}^{b}\nabla^2 g(s)_{ii}\tilde{f}_i(s^*) + O(\frac{1}{N^{1-\xi}}) \cdot O(\frac{1}{N^{r(1-\xi)-\epsilon}}) + o(\frac{1}{N})
\end{aligned}$$

$$= -\frac{1}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) + o(\frac{1}{N})$$

where the second equality is because $||s - s^*||^2 \leq b = O(\log N)$. We note that with the choice of parameters r, ϵ and the fact $\xi > 0$ is arbitrarily small, it's easy to check that non-dominant terms are strictly upper bounded by $\frac{1}{N}$ order. □

The theorem shows the mean square error $\mathbb{E}[||S - s^*||^2]$ consists of two terms: the first term $-\frac{1}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$ is the dominant term and is calculated explicitly; the second term $o(\frac{1}{N})$ is "small" when N is sufficiently large.

Theorem 4 (Convergence Upper Bound)**.** *For sufficiently large N, we have that*

$$\mathbb{E}[||S - s^*||^2] \leq -\frac{4}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) \tag{3.21}$$

Proof. From lemma 8 with the same parameter conditions, it's easy to check that for sufficiently large N, we have

$$E[d(S, s^*)\big|||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \leq -\frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$$

For sufficiently large N, it's easy to check,

$$\mathbb{P}\{||S - s^*||^{2r} \geq \frac{1}{N^\epsilon}\} \leq O(\frac{1}{N^{r(1-\xi)-\epsilon}}) \leq \frac{1}{N^{1+2\xi}}$$

Then from the above two inequalites, for sufficiently large N, the mean square distance is

$$\begin{aligned}
&E[||S - s^*||^2] \\
=&E[||S - s^*||^2\big|||S - s^*||^{2r} \geq \frac{1}{N^\epsilon}] \cdot \mathbb{P}\{||S - s^*||^{2r} \geq \frac{1}{N^\epsilon}\} \\
&+ E[||S - s^*||^2\big|||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \cdot \mathbb{P}\{||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}\} \\
\leq&\frac{b}{N^{1+2\xi}} - \frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)
\end{aligned}$$

$$\leq \frac{1}{N^{1+\xi}} - \frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$$

$$\leq -\frac{4}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$$

where the last inequality holds for sufficiently large N. □

This result tells us that we can have a calculable upper bound for light traffic convergence for mean square error.

3.8 Simulation Results

We provide the results of some simulations for various arrival rates and system sizes. The results are based on the average of 10 runs, where each run simulates 10^8 time steps. And the first 9×10^7 are ignored to let the system to reach steady state. We compared the results from the simulations to our predictions. The simulations are the results of empirical mean square error times the system size N. And our predictions and upper bounds are the results for the dominant term times system size, i.e. $-\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$ and calculable upper bound times system size $-4\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$.

Table 3.1, 3.2 and 3.3 show the comparisons of simulation results and predictions for arrival rates range from 0.1 to 0.9. For $\lambda = 0.1$, our prediction is close to the simulation results, especially when the system size increases. For $\lambda = 0.5$, our prediction result is not as close as that of in the case of $\lambda = 0.1$, but the trend that as the system size increases the simulation results are getting closer to our prediction result. For $\lambda = 0.9$, the simulation results and prediction result are off by a factor close to 2, but the trend is still as the system size increases, the discrepancy between simulation results and prediction decreases.

Note that as the arrival rate increases, the entries in matrix $J^{-1}(s^*)$ becomes

Table 3.1: $\lambda = 0.1, b = 10$

N	10	100	1000	10000
Simulation	0.1464	0.1041	0.0998	0.0998
Prediction	0.099			
Upper Bound	0.3960			

Table 3.2: $\lambda = 0.5, b = 40$

N	10	100	1000	10000
Simulation	0.5818	0.5414	0.5393	0.5369
Prediction	0.4341			
Upper Bound	1.7364			

Table 3.3: $\lambda = 0.9, b = 40$

N	10	100	1000	10000
Simulation	3.7527	2.9673	2.8632	2.9572
Prediction	1.4882			
Upper Bound	5.9528			

larger. So the first term in Eq.(3.13) becomes larger as a result, which causes larger discrepancy for larger arrival rates.

On the other hand, the upper bounds are enough for even moderately large system size like $N = 10$, which shows the effective of our refined analysis. And from a practical point of view, the upper bounds are calculable as compared to the convergence results in Ying (2016).

Chapter 4

HEAVY TRAFFIC FOR FINITE BUFFER SIZE

For finite buffer size, $b = O(\log N)$, the mean-field model is following

$$\dot{s}_k = \begin{cases} \lambda(s_{k-1}^2 - s_k^2) - (s_k - s_{k+1}), & b-1 \leq k \leq 1 \\ \lambda(s_{b-1}^2 - s_b^2) - s_b, & k = b. \end{cases}$$

where $s_0(t) = 1$ and $s_{b+1}(t) = 0$ for $t \geq 0$. There exists a unique equilibrium points $\{s_k^*\}_{k=1,\cdots,b}$. We next establish the upper bound on

$$E[||X(\infty)||^2]$$

where $X(\infty) = S(\infty) - s^*$ and the expectation is taken over stationary distribution.

Theorem 5. *For $\lambda = 1 - \frac{\gamma}{N^\alpha}$ and $\lambda > 0.75$, given $0 < \alpha < 0.25, 0 < \gamma \leq 1$, we have*

$$E[||S(\infty) - s^*||^2] \leq \frac{1}{N^{1-4\alpha-\xi'}}$$

where $\xi' > 0$ can be chosen arbitrarily small,

The proof and analysis follow from heavy traffic infinite buffer caseYing (2017).

Define $x = s - s^*$, so

$$\dot{x}_k = f_k(x) := \begin{cases} \lambda[(x_{k-1} + s_{k-1}^*)^2 - (x_k + s_k^*)^2)] - [(x_k + s_k^*) - (x_{k+1} + s_{k+1}^*)], \\ \qquad 1 \leq k \leq b-1 \\ \lambda[(x_{b-1} + s_{b-1}^*)^2 - (x_b + s_b^*)^2] - (x_b + s_b^*), \quad k = b \end{cases}$$

$$= \begin{cases} -\lambda(x_1^2 + 2s_1^* x_1) - (x_1 - x_2), & k = 1 \\ \lambda[(x_{k-1}^2 + 2s_{k-1}^* x_{k-1}) - (x_k^2 + 2s_k^* x_k)] - (x_k - x_{k+1}), & 2 \leq k \leq b-1 \\ \lambda[(x_{b-1}^2 + 2s_{b-1}^* x_{b-1}) - (x_b^2 + 2s_b^* x_b)] - x_b, & k = b. \end{cases} \tag{4.1}$$

The unique equilibrium point for the system is $x^* = 0$. Consider $d(x, x^*) = \sum_{i=1}^{b} x_k^2$. In this case,

$$g(x) = -\int_0^\infty \sum_{k=1}^{b} x_k^2(t, x)dt$$

where $x(t, x)$ is the solution for the dynamical system defined above. By combining the Poisson equation and the steady-state equation, we have

$$E[\sum_{k=1}^{b} X_k^2(\infty)] = E[\nabla g(X(\infty)) \cdot f(X(\infty)) - Gg(X(\infty))]$$

From the definitions of f_k and $R_{x,y}$, we have

$$f_k(x) = \sum_{y \neq x} R_{x,y}(y_k - x_k)$$

for $k = 1, \cdots, b$. So

$$\begin{aligned} \nabla g(x) \cdot f(x) &= \sum_{k=1}^{b} \frac{\partial g}{\partial x_k}[\sum_{y \neq x} R_{x,y}(y_k - x_k)] \\ &= \sum_{y \neq x} R_{x,y} \sum_{k=1}^{b} \frac{\partial g}{\partial x_k}(y_k - x_k) \\ &= \sum_{y \neq x} R_{x,y} \nabla g(x) \cdot (y - x) \end{aligned}$$

Therefore, we have the following equation

$$E[\sum_{k=1}^{b} X_k^2(\infty)] = E[-\sum_{y \neq X(\infty)} R_{X(\infty),y}\Gamma(X(\infty), y)] \tag{4.2}$$

where $\Gamma(X(\infty), y) = g(y) - g(X(\infty)) - \nabla g(X(\infty)) \cdot (y - x)$. We just need to establish a bound on $\Gamma(X(\infty), y)$.

Given an arbitrarily small $\xi > 0$, define

$$\tilde{k} = (\alpha + \xi) \log_2 N.$$

and WLOG, we assume $b \geq \tilde{k}$.

Lemma 9. *According to the definition of $\tilde{k}$, for sufficiently large N, we have for any $k \geq \tilde{k}$*

$$\lambda(s_k^* + 1) \leq \lambda(\lambda^{2^k - 1} + 1) \leq \sqrt{\lambda}.$$

Proof. For the equilibrium point finite buffer sized system, it satisfies the following equations

$$\lambda[(s_{k-1}^*)^2 - (s_k^*)^2] - (s_k^* - s_{k+1}^*) = 0, \quad 1 \leq k \leq b - 1$$
$$\lambda[(s_{b-1}^*)^2 - (s_b^*)^2] - s_b^* = 0, \quad k = b.$$

For any k that is $1 \leq k \leq b - 1$, by adding equation (k) to (b), we have

$$\lambda[(s_{k-1}^*)^2 - (s_b^*)^2] - s_k^* = 0$$

Thus, we have

$$s_k^* = \lambda[(s_{k-1}^*)^2 - (s_b^*)^2] \leq \lambda(s_{k-1}^*)^2.$$

For the case $k = b$, we consider the equation (b)

$$s_b^* + \lambda(s_b^*)^2 = \lambda(s_{b-1}^*)^2$$

For $k = b$, we also have inequality

$$s_k^* \leq \lambda(s_{k-1}^*)^2$$

So iteratively, given $s_0^* = 1$, for all $1 \leq k \leq b$,

$$s_k^* \leq \lambda^{2^k - 1}.$$

The first inequality follows as a result.

For the second inequality, note that

$$\lambda^{2^k} \leq \lambda^{2^{\tilde{k}}} = \lambda^{N^{\alpha+\xi}} = (1 - \frac{\gamma}{N^\alpha})^{N^{\alpha+\xi}},$$

so

$$\log \lambda^{2^k} \leq N^{\alpha+\xi} \log(1 - \frac{\gamma}{N^\alpha}) \leq_{(a)} -\gamma N^\xi = -\Theta(N^\xi),$$

where inequality (a) is a result of the Taylor expansion. Furthermore,

$$\log(\sqrt{\lambda} - \lambda) = \log\sqrt{\lambda} + \log(1 - \sqrt{\lambda}) = -\Theta(\alpha \log N).$$

Therefore, for sufficiently large N, we have

$$\lambda^{2^k} \leq \sqrt{\lambda} - \lambda,$$

and the lemma holds. □

Now define a sequence of $\{w_k\}$ such that for some $\epsilon > 0$

$$\begin{aligned}
&w_0 = 0 \\
&w_1 = 1 \\
&w_k = 1 + \frac{1}{2}\sum_{j=1}^{k} \frac{1}{(2\lambda+\epsilon)^{j-1}}, \quad 2 \leq k \leq \tilde{k} \\
&w_k = w_{\tilde{k}} + \frac{k - \tilde{k}}{2(2\lambda+\epsilon)^{\tilde{k}}}, \quad \tilde{k} < k \leq b.
\end{aligned}$$

We choose a constant ϵ independent of N such that

$$\min\{0.5, 2^{\frac{\alpha+2\xi}{\alpha+\xi}} - 2\lambda\} > \epsilon > 2 - 2\lambda.$$

Such an ϵ exists when $\lambda > 0.75$. We further define

$$\delta_0 = \frac{1 - \sqrt{\lambda}}{6(2\lambda+\epsilon)^{\tilde{k}}}$$

Lemma 10. *When N is sufficiently large, for any $1 \leq k \leq b$, $1 \leq w_k \leq 3$. And*

$$\delta_0 \geq \frac{1}{12N^{2\alpha+2\xi}}$$

Proof. To prove the result, we note that for $k \leq \tilde{k}$,

$$w_k \leq w_{\tilde{k}} \leq 1 + \frac{1}{2}\frac{1}{1-\frac{1}{2\lambda+\epsilon}} \leq 2$$

where the last inequality holds because $2\lambda + \epsilon > 2$. For $k > \tilde{k}$,

$$w_k \leq w_{\tilde{k}} + \frac{b}{2(2\lambda+\epsilon)^{\tilde{k}}} \leq 2 + 0.5 < 3,$$

where the second inequality holds because

$$(2\lambda+\epsilon)^{\tilde{k}} > 2^{\tilde{k}} = N^{\alpha+\xi}$$

which is larger than $b = O(\log N)$ for sufficiently large N.

For the second inequality, we have

$$\begin{aligned}
\delta_0 &= \frac{1-\sqrt{\lambda}}{6(2\lambda+\epsilon)^{\tilde{k}}} \\
&= \frac{1-\lambda}{6(1+\sqrt{\lambda})(2\lambda+\epsilon)^{\tilde{k}}} \\
&\geq \frac{1-\lambda}{6(1+\sqrt{\lambda})(2^{\frac{\alpha+2\xi}{\alpha+\xi}})^{\tilde{k}}} \\
&= \frac{\gamma}{6(1+\sqrt{\lambda})N^{\alpha}N^{\alpha+2\xi}} \\
&\geq \frac{\gamma}{12N^{2\alpha+2\xi}}
\end{aligned}$$

□

In the following analysis, we verify the condition in lemma 2.2 in Ying (2017). We define $V(x) = \sum_{k=1}^{b} w_k |x_k(t)|$.

Lemma 11. *(**Proof of C1**). For the dynamical system defined in (4.1), we have*

$$\dot{V}(x) \leq -\delta_0 V(x)$$

which implies that

$$|x(t)| \leq V(x(t)) \leq 3|x(0)|e^{-\delta_0 t}$$

Proof. Note that $V(x)$ is Lipschitz continuous function. We now consider regular points such that $\frac{d|x_k(t)|}{dt}$ exists for all k at time t. Define $\dot{V}(x) = \sum_{k=1}^{b} W_k(t)$ such that $W_k(t)$ includes all the terms involving $x_k(t)$. The lemma is proved by showing that

$$W_k(t) \leq -\delta_0 w_k |x_k(t)| \tag{4.3}$$

When $x_k(t) > 0$, we have

$$W_k(t) \leq w_{k+1}\lambda|x_k|(x_k + 2s_k^*) - w_k\lambda|x_k|(x_k + 2s_k^*) - w_k|x_k| + w_{k-1}|x_k|$$

The same inequality holds for $x_k(t) < 0$. So (4.3) holds if

$$w_{k+1}\lambda|x_k|(x_k + 2s_k^*) - w_k\lambda|x_k|(x_k + 2s_k^*) - w_k|x_k| + w_{k-1}|x_k| \leq -\delta_0 w_k|x_k|$$

in other words, if

$$w_{k+1} - w_k \leq \frac{(1-\delta_0)w_k - w_{k-1}}{\lambda(x_k + 2s_k^*)} \tag{4.4}$$

For $1 \leq k \leq \tilde{k}$, we have

$$w_{k+1} - w_k = \frac{1}{2(2\lambda+\epsilon)^k}$$

$$\frac{(1-\delta_0)w_k - w_{k-1}}{\lambda(x_k + 2s_k^*)} \geq \frac{w_k - w_{k-1} - \delta_0 w_k}{\lambda(1+\lambda)} \geq \frac{\frac{1}{2(2\lambda+\epsilon)^{k-1}} - \delta_0 w_k}{2\lambda}$$

So the inequality (4.4) holds if

$$2\lambda \le 2\lambda + \epsilon - 2\delta_0 w_k (2\lambda + \epsilon)^k$$

which can be established by proving

$$2\delta_0 w_b (2\lambda + \epsilon)^{\tilde{k}} \le \epsilon.$$

It can be verified that the inequality holds according to the definition of δ_0 and the fact that $\epsilon > 2 - 2\lambda \ge 1 - \sqrt{\lambda}$.

When $b \ge k \ge \tilde{k} + 1$, according to lemma 9,

$$\lambda(x_k + 2s_k^*) \le \lambda(1 + s_k^*) \le \sqrt{\lambda}.$$

Therefore, we have

$$w_{k+1} - w_k = \frac{1}{2(2\lambda + \epsilon)^{\tilde{k}}}$$

$$\frac{(1-\delta_0)w_k - w_{k-1}}{\lambda(x_k + 2s_k^*)} \ge \frac{w_k - w_{k-1} - \delta_0 w_k}{\sqrt{\lambda}} = \frac{\frac{1}{2(2\lambda+\epsilon)^{\tilde{k}}} - \delta_0 w_k}{\sqrt{\lambda}}.$$

So inequality (4.4) holds if

$$\sqrt{\lambda} \le 1 - 2\delta_0 w_k (2\lambda + \epsilon)^{\tilde{k}},$$

in other words, if

$$w_k \le \frac{1 - \sqrt{\lambda}}{2\delta_0 (2\lambda + \epsilon)^{\tilde{k}}},$$

which holds because $w_k \le 3$ according to lemma 10 and $\frac{1-\sqrt{\lambda}}{2\delta_0(2\lambda+\epsilon)^{\tilde{k}}} = 3$ according to the definition of δ_0.

From the discussion above, we conclude that

$$\dot{V}(t) \le -\sum_{k=1}^{b} \delta_0 w_k |x_k(t)| = -\delta_0 V(t)$$

. □

We further have the following first-order system for system (4.1):

$$\dot{x}_k^{(1)} = g_k(x^{(1)}) = 2\lambda(x_{k-1} + s_{k-1}^*)x_{k-1}^{(1)} - 2\lambda(x_k + s_k^*)x_k^{(1)} - x_k^{(1)} + x_{k+1}^{(1)}, \tag{4.5}$$

where $0 < k \leq b$ and we define $x_0^{(1)} = x_{b+1}^{(1)} \equiv 0$.

Lemma 12. ***(Proof of C2)****. Under the dynamical system defined by (4.5), we have*

$$|x^{(1)}(t)| \leq |x^{(1)}(0)| = 1.$$

Proof. First recall that $s_k(t) = x_k(t) + s_k^*(t) \geq 0$ for any $t \geq 0$ and k. Define

$$V(t) = \sum_{k=1}^{b} |x_k^{(1)}(t)|.$$

Note that

$$\frac{d|x^{(1)}(t)|}{dt} \leq 2\lambda(x_{k-1} + s_{k-1}^*)|x_{k-1}^{(1)}| - 2\lambda(x_k + s_k^*)|x_k^{(1)}| - |x_k^{(1)}| + |x_{k+1}^{(1)}|$$

So

$$\dot{V}(t) \leq -2\lambda(x_b + s_b^*)|x_b^{(1)}| - |x_1^{(1)}| \leq 0$$

Also

$$|x^{(1)}(0)| = |N(z-y)| = |N1_k| = 1$$

where 1_k is the b dimensional vector with kth element being $\frac{1}{N}$ the rest is 0. Because we are only interested in transition where $R_{xy} \neq 0$, from Stein's equation (4.2). Hence the lemma holds. □

Define

$$\tilde{\delta} = \frac{\epsilon}{6(2\lambda + \epsilon)^{\tilde{k}}}$$

Lemma 13. *For sufficiently large N, we have*

$$\tilde{\delta} \geq \frac{\gamma}{3N^{2\alpha+2\xi}}$$

Proof.

$$\begin{aligned}
\log \tilde{\delta} &= \log \epsilon - \log 6 - \tilde{k} \log(2\lambda + \epsilon) \\
&\geq \log(2 - 2\lambda) - (\alpha + \xi) \log N \log(2\lambda + \epsilon) - \log 6 \\
&\geq \log \gamma - \alpha \log N - (\alpha + \xi) \log N \cdot \frac{\alpha + 2\xi}{\alpha + \xi} - \log 3 \\
&= -(2\alpha + 2\xi) \log N + \log \frac{\gamma}{3}.
\end{aligned}$$

So, $\tilde{\delta} \geq \frac{\gamma}{3N^{2\alpha+2\xi}}$. □

Lemma 14. *(**Proof of C3**). For sufficiently large N and for all x, we have*

$$\begin{aligned}
&|x^{(1)}| \leq V(x^{(1)}) \leq 3|x^{(1)}| \\
&\dot{V}(x^{(1)}(t)) \leq -\tilde{\delta} V(x^{(1)}(t)), \quad \textit{if} \quad |x(t)| \leq \frac{1}{8}
\end{aligned}$$

Proof. Define the Lyapunov function

$$V(x^{(1)}) = \sum_{k=1}^{b} w_k |x_k^{(1)}|$$

Following the proof of lemma 11, we obtain that

$$\dot{V}(x^{(1)}) \leq \sum_{k=1}^{b} -[2w_k \lambda(x_k + s_k^*) + w_k - 2\lambda w_{k+1}(x_k + s_k^*) - w_{k-1}]|x_k^{(1)}|$$

So the lemma holds by proving

$$-[2w_k \lambda(x_k + s_k^*) + w_k - 2\lambda w_{k+1}(x_k + s_k^*) - w_{k-1}] \leq -\tilde{\delta} w_k$$

i.e. by proving

$$w_{k+1} - w_k \leq \frac{w_k - w_{k-1} - \tilde{\delta} w_k}{2\lambda(x_k + s_k^*)} \tag{4.6}$$

For $1 \leq k \leq \tilde{k}$, we have

$$w_{k+1} - w_k = \frac{1}{2(2\lambda+\epsilon)^k}$$
$$\frac{w_k - w_{k-1} - \tilde{\delta} w_k}{2\lambda(x_k + s_k^*)} \geq \frac{\frac{1}{2(2\lambda+\epsilon)^{k-1}} - \tilde{\delta} w_k}{2\lambda}$$

so inequality (4.6) holds if

$$2\lambda \leq 2\lambda + \epsilon - \tilde{\delta} 2 w_k (2\lambda+\epsilon)^{\tilde{k}},$$

which holds according to the definition of $\tilde{\delta}$ and the fact $1 \leq w_k \leq 3$.

When $b \geq k \geq \tilde{k}+1$, according to the definition of $\tilde{k}$,

$$s_k^* \leq s_{\tilde{k}}^* \leq \lambda^{N^{\alpha+\xi}-1}.$$

If $\lambda \geq \frac{64}{81}$, then

$$s_k^* \leq \frac{1}{\sqrt{\lambda}} - 1 \leq \frac{1}{8}$$

according to lemma 9; otherwise, we can find a sufficiently large N such that

$$s_k^* \leq \lambda^{N^{\alpha+\xi}-1} \leq \frac{1}{8}.$$

Now given $|x_k| \leq |x| \leq \frac{1}{8}$, we have

$$w_{k+1} - w_k = \frac{1}{2(2\lambda+\epsilon)^{\tilde{k}}}$$
$$\frac{(1-\tilde{\delta})w_k - w_{k-1}}{2\lambda(|x_k| + s_k^*)} \geq \frac{w_k - w_{k-1} - \tilde{\delta} w_k}{\frac{\lambda}{2}} = \frac{\frac{1}{2(2\lambda+\epsilon)^{\tilde{k}}} - \tilde{\delta} w_k}{\frac{\lambda}{2}}$$

So inequality (4.6) holds if

$$\frac{\lambda}{2} \leq 1 - \tilde{\delta} 2 w_k (2\lambda+\epsilon)^{\tilde{k}}.$$

Note that according to the definition of $\tilde{\delta}$,

$$\tilde{\delta} 2 w_k (2\lambda+\epsilon)^{\tilde{k}} \leq 6\tilde{\delta}(2\lambda+\epsilon)^{\tilde{k}} = \epsilon.$$

So the inequality holds because $\epsilon < 0.5$ from its definition. From the above, we conclude $\dot{V}(t) \le -\tilde{\delta} V(t)$ when $|x(t)| \le \frac{1}{8}$. $\square$

Lemma 15. *Given $|e(t)| \le \frac{1}{N}$, we have*

$$\frac{d|e(t)|}{dt} \le 4(\lambda+4)\frac{1}{N^2}$$

Proof. We first have for $1 < k < b$,

$$\begin{aligned}
\dot{e}_k(t) =& f_k\left(x(t) + \frac{1}{N}x^{(1)}(t) + e(t)\right) - f_k(x(t)) - \frac{1}{N}\sum_{j=1}^{b}\frac{\partial f_k}{\partial x_j}(x(t))x_j^{(1)}(t) \\
=& \lambda\left((x_{k-1}(t) + \frac{1}{N}x_{k-1}^{(1)}(t) + e_{k-1}(t))^2 + 2s_{k-1}^*(x_{k-1}(t) + \frac{1}{N}x_{k-1}^{(1)}(t) + e_{k-1}(t))\right) \\
&- \lambda\left((x_k(t) + \frac{1}{N}x_k^{(1)}(t) + e_k(t))^2 + 2s_k^*(x_k(t) + \frac{1}{N}x_k^{(1)}(t) + e_k(t))\right) \\
&- \left(x_k(t) + \frac{1}{N}x^{(1)}(t) + e_k(t)\right) + \left(x_{k+1}(t) + \frac{1}{N}x_{k+1}^{(1)}(t) + e_{k+1}(t)\right) \\
&- \lambda(x_{k-1}^2(t) + 2s_{k-1}^*x_{k-1}(t)) + \lambda(x_k^2(t) + 2s_k^*x_k(t)) + (x_k(t) - x_{k+1}(t)) \\
&- \frac{2}{N}\lambda(x_{k-1} + s_{k-1}^*)x_{k-1}^{(1)} + \frac{2}{N}\lambda(x_k + s_k^*)x_k^{(1)} + \frac{1}{N}x_k^{(1)} - \frac{1}{N}x_{k+1}^{(1)} \\
=& \lambda\left(e_{k-1}^2 + 2(x_{k-1} + s_{k-1}^* + \frac{1}{N}x_{k-1}^{(1)})e_{k-1} - e_k^2 - 2(x_k + s_k^* + \frac{1}{N}x_k^{(1)})e_k\right) \\
&- (e_k - e_{k+1}) + \lambda\frac{1}{N^2}\left((x_{k-1}^{(1)})^2 - (x_k^{(1)})^2\right) \\
=& 2\lambda(x_{k-1} + s_{k-1}^*)e_{k-1} - 2\lambda(x_k + s_k^*)e_k - (e_k - e_{k+1}) \\
&+ \lambda\left(e_{k-1}^2 + 2\frac{1}{N}x_{k-1}^{(1)}e_{k-1} - e_k^2 - 2\frac{1}{N}x_k^{(1)}e_k\right) + \lambda\frac{1}{N^2}\left((x_{k-1}^{(1)})^2 - (x_k^{(1)})^2\right) \\
=& g_k(e) + \lambda\left(e_{k-1}^2 + 2\frac{1}{N}x_{k-1}^{(1)}e_{k-1} - e_k^2 - 2\frac{1}{N}x_k^{(1)}e_k\right) + \lambda\frac{1}{N^2}\left((x_{k-1}^{(1)})^2 - (x_k^{(1)})^2\right)
\end{aligned}$$

where the last equality holds according to the definition of $g_k(\cdot)$. The same equation holds for $k = 1$ and $k = b$. From the equality from above and following the proof lemma 12, we can further obtain

$$\frac{d|e(t)|}{dt} \le \sum_{k=1}^{b} 2\lambda(e_k^2 + 2\frac{1}{N}|x_k^{(1)}||e_k|) + 2\lambda\frac{1}{N^2}(x_k^{(1)})^2 \le 2\lambda|e(t)|^2 + \frac{4}{N}|e(t)| + \frac{2\lambda}{N^2} \tag{4.7}$$

Given $|e(t)| \leq \frac{1}{N}$, we conclude

$$\frac{d|e(t)|}{dt} \leq (4\lambda + 4)\frac{1}{N^2} \tag{4.8}$$

□

Lemma 16. *For Lyapunov function* $V(e(t)) = \sum_{k=1}^{b} w_k |e_k(t)|$, *we have*

$$\dot{V}(e(t)) \leq \tilde{\delta} V(e(t)) + \frac{6\lambda}{N^2}(|x^{(1)}(t)|)^2$$

Proof. Recall that

$$\begin{aligned}\dot{e}_k(t) =& \lambda\left(2(x_{k-1} + s_{k-1}^* + \frac{e_{k-1}}{2} + \frac{1}{N}x_{k-1}^{(1)})e_{k-1} - 2(x_k + s_k^* + \frac{e_k}{2} + \frac{1}{N}x_k^{(1)})e_k\right) \\ &- (e_k - e_{k+1}) + \lambda\frac{1}{N^2}\left((x_{k-1}^{(1)})^2 - (x_k^{(1)})^2\right)\end{aligned}$$

Again consider

$$\dot{V}(e(t)) = \sum_{k=1}^{b} W_k(t) + W(t)$$

where $W_k(t)$ includes all the terms involving $e_k(t)$ and $W(t)$ includes all the remaining terms.

$$\begin{aligned}W_k(t) \leq& w_{k+1}[2\lambda(x_k + s_k^*)|e_k| + \lambda|e_k|^2 + \frac{2\lambda}{N}|x_k^{(1)}| \cdot |e_k|] \\ &- w_k[2\lambda(x_k + s_k^*)|e_k| + |e_k| - \lambda|e_k|^2 - \frac{2\lambda}{N}|x_k^{(1)}| \cdot |e_k|] + w_{k-1}|e_k| \leq -\tilde{\delta} w_k |e_k|\end{aligned}$$

i.e., just need to prove that

$$\begin{aligned}& w_{k+1}[2\lambda(x_k + s_k^*) + \lambda|e_k| + \frac{2\lambda}{N}|x_k^{(1)}|] - w_k[2\lambda(x_k + s_k^*) + 1 - \lambda|e_k| - \frac{2\lambda}{N}|x^{(1)}|] + w_{k-1} \\ =& w_{k+1}2\lambda(x_k + s_k^*) - w_k 2\lambda(x_k + s_k^*) + w_{k-1} + \lambda|e_k|(w_{k-1} + w_k) + \frac{2\lambda}{N}|x_k^{(1)}|(w_{k-1} + w_k) \\ &- w_k \leq -\tilde{\delta} w_k\end{aligned}$$

Note that $|e_k(t)|$ and $\frac{1}{N}|x_k^{(1)}|$ can be made arbitrarily small by choosing sufficiently large N. Thus, for a sufficiently large N, following analysis of lemma 5, we have

$$\sum_{k=1}^{b} W_k(t) \leq -\tilde{\delta} V(t) \tag{4.9}$$

Since $1 \leq w_k \leq 3$ for all $k \geq 1$,

$$\dot{V}(e(t)) \leq -\tilde{\delta}V(e(t)) + \max_k w_k \frac{2\lambda}{N^2}||x^{(1)}||^2 \leq -\tilde{\delta}V(e(t)) + \frac{6\lambda}{N^2}|x^{(1)}|^2$$

□

The analysis above verifies conditions C1-C5 in Lemma 2.2 Ying (2017) with $c_1 = c_{l1} = c_{le} = 1, c_{u1} = c_{ue} = 3, d_1 = d_e = \frac{1}{8}, \delta_1 = \delta_e = \tilde{\delta}$ and $c_e = 4\lambda + 4$. Furthermore, $|x^{(1)}(0)| = |N(z-y)| = 1 =: b_u$, and $\tilde{t}_{d,z} = \frac{1}{\delta_0}\max\{0, \ln 24|x(0)|\}$ according to lemma 11. Parameter $\alpha = 2$ in condition 5, according to lemma 8. Therefore, both C6 and C7 hold. Hence, we conclude that there exists a constant κ such that when N is sufficiently large, the following two inequalities hold

$$\begin{aligned}
\int_0^\infty |x^{(1)}(t)|^2 dt &\leq \kappa(\tilde{t}_{d,z} + \frac{1}{2\delta_1}) \\
&= \kappa(\frac{\max\{0, \ln 24|x(0)|\}}{\delta_0} + \frac{1}{2\delta_1}) \\
&\leq \kappa(\frac{12}{\gamma}N^{2\alpha+2\xi}\max\{0, \ln 24|x(0)|\} + \frac{6}{\gamma}N^{2\alpha+2\xi}) \\
\int_0^\infty |e(t)|dt &\leq \kappa(\frac{\tilde{t}_{d,z}^2}{N^2} + \frac{\tilde{t}_{d,z}}{\delta_e}\frac{1}{N^2} + \frac{1}{\delta_1\delta_e}\frac{1}{N^2}) \\
&\leq \kappa(\frac{\max^2\{0, \ln 24|x(0)|\}}{\delta_0^2 N^2} + \frac{\max\{0, \ln 24|x(0)|\}}{\delta_0\delta_e}\frac{1}{N^2} + \frac{1}{\delta_1\delta_e}\frac{1}{N^2}) \\
&\leq \kappa(\frac{144}{\gamma^2}\frac{\max^2\{0, \ln 24|x(0)|\}}{N^{2-4\alpha-4\xi}} + \frac{36}{\gamma^2}\frac{\max\{0, \ln 24|x(0)|\}}{N^{2-4\alpha-4\xi}} + \frac{36}{\gamma^2}\frac{1}{N^{2-4\alpha-4\xi}})
\end{aligned} \tag{4.10}$$

Therefore, we will have bound on 2-norm as following

$$\begin{aligned}
|\Gamma(z,y)| &\leq \int_0^\infty (3x_{\max}|e(t)| + \frac{1}{N^2}|x^{(1)}|^2)dt \\
&\leq \frac{3\kappa}{\gamma^2}(144\frac{\max^2\{0, \ln 24|x(0)|\}}{N^{2-4\alpha-4\xi}} + 36\frac{\max\{0, \ln 24|x(0)|\}}{N^{2-4\alpha-4\xi}} + 36\frac{1}{N^{2-4\alpha-4\xi}}) \\
&+ \frac{\kappa}{\gamma}(12N^{2\alpha+2\xi}\max\{0, \ln 24|x(0)|\} + 6N^{2\alpha+2\xi})\frac{1}{N^2} \\
&\leq \frac{3\kappa}{\gamma^2}(144\max{}^2\{0, \ln 24|x(0)|\} + 36\max\{0, \ln 24|x(0)|\} + 36)\frac{1}{N^{2-4\alpha-4\xi}}
\end{aligned}$$

$$+\frac{\kappa}{\gamma}(12\max\{0,\ln 24|x(0)|\}+6)\frac{1}{N^{2-2\alpha-2\xi}}$$

$$\leq \max{}^2\{0,\ln 24|x(0)|\}\frac{1}{N^{2-4\alpha-5\xi}}+\frac{1}{N^{2-4\alpha-5\xi}} \tag{4.11}$$

where the last inequality holds for sufficiently large N. Therefore, by equation (4.2), we have

$$E[\sum_{k=1}^{b}X_k^2(\infty)]\leq E[(\max{}^2\{0,\ln 24|x(0)|\}\frac{1}{N^{2-4\alpha-5\xi}}+\frac{1}{N^{2-4\alpha-5\xi}})(\sum_{y\neq X(\infty)}R_{X(\infty),y})]$$

By choosing the initial condition to be the stationary distribution, and rewriting the above equation, we have

$$E[\sum_{k=1}^{b}X_k^2(\infty)]\leq E[(\max{}^2\{0,\ln 24|X(\infty)|\}\frac{1}{N^{2-4\alpha-5\xi}}+\frac{1}{N^{2-4\alpha-5\xi}})(\sum_{y\neq X(\infty)}R_{X(\infty),y})]$$

Note that

$$\sqrt{b}||X(\infty)||=\sqrt{b}\sqrt{\sum_{k=1}^{b}X_k^2(\infty)}\geq\sum_{k=1}^{b}|X_k(\infty)|=|X(\infty)|$$

which implies that

$$\max{}^2\{0,\ln 24|X(\infty)|\}\leq(24|X(\infty)|)^2\leq 576b||X(\infty)||^2$$

Recall that $b=O(\log N)$ and $\sum_{y\neq X(\infty)}R_{X(\infty),y}\leq 2N$, then Therefore, in the end, we have

$$E[\sum_{k=1}^{b}X_k^2(\infty)]\leq\frac{1152b}{N^{1-4\alpha-5\xi}}E[\sum_{k=1}^{b}X_k^2(\infty)]+\frac{2}{N^{1-4\alpha-5\xi}}$$

$$\leq\frac{1}{N^{1-4\alpha-6\xi}}E[\sum_{k=1}^{b}X_k^2(\infty)]+\frac{1}{N^{1-4\alpha-6\xi}}$$

the second inequality holds for a sufficiently large N. By moving the first term to the left-hand-side and then dividing both sides by $1-\frac{1}{N^{1-4\alpha-6\xi}}$, we get

$$E[\sum_{k=1}^{b}X_k^2(\infty)]\leq\frac{1}{1-\frac{1}{N^{1-4\alpha-6\xi}}}\frac{1}{N^{1-4\alpha-6\xi}}$$

$$\leq \frac{1}{N^{1-4\alpha-7\xi}} \tag{4.12}$$

Recall that $0 < \alpha < 0.25$, so $\frac{1}{N^{1-4\alpha-6\xi}}$ can be made arbitrarily small when choosing sufficiently large N. So when N is sufficiently large and by choosing $\xi^{'} = 7\xi$.

Lemma 17. *For $t \geq 0$ and sufficiently large N, we have following bound on term $e(t)$*

$$|e_t| \leq \frac{1}{N^{2-2\alpha-4\xi}} \tag{4.13}$$

Proof. According to C4 and C6, and the fact $e(0) = 0$, we have that for $t \leq \tilde{t}_{d,z}$,

$$\begin{aligned}
|e(t)| &\leq \frac{c_e}{N^2} t \\
&\leq \frac{c_e}{N^2} t_{d,z} \\
&\leq \frac{c_e}{N^2} \frac{1}{\delta_0} \max\{0, \log 24|x(0)|\} \\
&\leq \frac{c_e}{N^2} \cdot 12N^{2\alpha+2\xi} \cdot 24|x(0)| \\
&\leq \frac{288c_e}{N^{2-2\alpha-2\xi}} b \\
&\leq \frac{1}{N^{2-2\alpha-3\xi}}
\end{aligned}$$

The last inequality is because $b = O(\log N)$.

For $t \geq \tilde{t}_{d,z}$, from C5, based on comparison principle, we obtain

$$\begin{aligned}
|e_t| &\leq \frac{1}{c_{le}} V_e(e(t)) \leq \frac{1}{c_{le}} V_e(e(t)) \\
&\leq V_e(e(\tilde{t}_{d,z})) e^{-\delta_e(t-\tilde{t}_{d,z})} + \frac{c}{N^2} e^{-\delta_e(t-\tilde{t}_{d,z})} \frac{1}{\alpha\delta_1 - \delta_e} (1 - \exp(-(\alpha\delta_1 - \delta_e)(t - \tilde{t}_{d,z})))
\end{aligned}$$

where $\tilde{t}_{d,z} = \frac{1}{\delta_0} \max\{0, \log 24|x(0)|\}$, $\delta_1 = \delta_e = \tilde{\delta}$, $\alpha = 2$(in lemma 2.2) and $c = c_{er}(\frac{c_{u1}c_1}{c_{l1}})^\alpha$, so

$$|e_t| \leq V_e(e(\tilde{t}_{d,z})) + \frac{c}{N^2} \frac{1}{\tilde{\delta}}$$

$$
\begin{aligned}
&\leq c_{ue}|e(\tilde{t}_{d,z})| + \frac{c}{N^2}\frac{1}{\tilde{\delta}} \\
&\leq c_{ue}\frac{1}{N^{2-2\alpha-3\xi}} + \frac{c}{N^2}3N^{2\alpha+2\xi} \\
&\leq \frac{1}{N^{2-2\alpha-4\xi}} \qquad (4.14)
\end{aligned}
$$

where the last inequality holds for sufficiently large N. □

We note that with the assumption of $b = O(\log N)$, we can actually achieve a larger range of $\alpha \in (0, 0.25)$ compared to that of Ying (2017), which is $\alpha \in (0, 0.2)$ in heavy traffic.

Chapter 5

HEAVY TRAFFIC CONVERGENCE RATE COEFFICIENT

In this chapter, we characterize the convergence dominant term in the heavy traffic case. Similar to the procedures in chapter 3, we first show the probability of being far from the solution of the mean-field model is small and use a linear mean-field model to characterize for the state space close to the solution of mean-field model.

Since the arrival rate is a function of N, the solution of mean-field model is also an N dependent value. Thus, we need a more sophisticated analysis of the tridiagonal matrix for the linear meal-field model in order to characterize the dominant term.

5.1 Higher Moment Bounds

Previously, for any $0 < \alpha < 0.25$ and a sufficiently large N, we have

$$E[||S - s^*||^2] \leq \frac{1}{N^{1-4\alpha-7\xi}}$$

Also

$$||e(t)|| \leq |e(t)| \leq \frac{1}{N^{2-2\alpha-4\xi}}$$

and

$$\begin{aligned}\int_0^\infty |e(t)|dt &\leq \frac{\kappa}{\gamma^2}(144\frac{\max^2\{0, \ln 24|x(0)|\}}{N^{2-4\alpha-4\xi}} + 36\frac{\max\{0, \ln 24|x(0)|\}}{N^{2-4\alpha-4\xi}} + 36\frac{1}{N^{2-4\alpha-4\xi}}) \\ &\leq \frac{1}{N^{2-4\alpha-5\xi}}\end{aligned}$$

Then, we consider the higher moment $E[||S - s^*||^{2r}]$ for $r \in \mathbb{N}$

$$E[||S - s^*||^{2r}] \leq \frac{1}{N^{r(1-4\alpha-\xi')}} \tag{5.1}$$

where $\xi^{'} = 7\xi$.

We continue to analyze the term in equation (3.3). Similar to light traffic case, by raising to the rth order, we have

$$
\begin{aligned}
&[\sum_{i=1}^{b}(s_i(t,s') - s_i^*)^2]^r \\
=&[\sum_{i=1}^{b}(s_i(t,x) - s_i^*)^2 + \sum_{i=1}^{b} e_i^2(t) + \sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2 + 2\sum_{i=1}^{b} e_i(t)(s_i(t,s) - s_i^*) \\
&+ 2\sum_{i=1}^{b}(s_i(t,s) - s_i^*)\nabla s_i(t,s)\cdot(s'-s) + 2\sum_{i=1}^{b} e_i(t)\nabla s_i(t,s)\cdot(s'-s)]^r \\
=&\sum_{\sum_{k=1}^{6} r_k = r, r_k \geq 0} \binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5} \\
&[\sum_{i=1}^{b}(s_i(t,s) - s_i^*)^2]^{r_1}[\sum_{i=1}^{b} e_i^2(t)]^{r_2}[\sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2]^{r_3}[2\sum_{i=1}^{b} e_i(t)(s_i(t,s) - s_i^*)]^{r_4} \\
&[2\sum_{i=1}^{b}(s_i(t,s) - s_i^*)\nabla s_i(t,s)\cdot(s'-s)]^{r_5}[2\sum_{i=1}^{b} e_i(t)\nabla s_i(t,s)\cdot(s'-s)]^{r_6}
\end{aligned}
$$

We have that

$$|e_i(t)| \leq |e(t)| \leq \frac{1}{N^{2-2\alpha-4\xi}}$$

and the state transition condition

$$||s - s'|| = \frac{1}{N}$$

Also

$$s^{(1)}(t) = \nabla s(t,s)\cdot N(s'-s)$$

So, we have

$$\nabla s(t,s)\cdot(s'-s) = \frac{1}{N}s^{(1)}(t)$$

therefore, $|\nabla s(t,s)\cdot(s'-s)| = \frac{1}{N}|s^{(1)}(t)| \leq \frac{1}{N}|s^{(1)}(0)| = \frac{1}{N}$, by lemma 4 in Chaper 4. Also

$$||\nabla s(t,s)\cdot(s'-s)||^2 = \sum_{i=1}^{b}[\nabla s_i(t,s)\cdot(s'-s)]^2$$

$$= \sum_{i=1}^{b} |\frac{1}{N} s_i^{(1)}(t)|^2$$
$$= \frac{1}{N^2} \sum_{i=1}^{b} |s_i^{(1)}(t)|^2$$
$$\leq \frac{1}{N^2} \sum_{i=1}^{b} |s_i^{(1)}(t)|$$
$$\leq \frac{1}{N^2} |s^{(1)}(t)| \leq \frac{1}{N^2} |s^{(1)}(0)| = \frac{1}{N^2}$$

The first inequality is because $|s^{(1)}(t)| \leq |s^{(1)}(0)| = 1$. So, for any $i \in \{1, \cdots, b\}$,we have

$$|\nabla s_i(t,s) \cdot (s' - s)| \leq ||\nabla s(t,s) \cdot (s' - s)|| \leq \frac{1}{N}$$

$$[\sum_{i=1}^{b} (s_i(t,s') - s_i^*)^2]^r$$
$$= \sum_{\sum_{k=1}^{6} r_k = r, r_k \geq 0} \binom{r}{r_1} \binom{r - r_1}{r_2} \binom{r - r_1 - r_2}{r_3} \binom{r - r_1 - r_2 - r_3}{r_4} \binom{r - r_1 - r_2 - r_3 - r_4}{r_5}$$
$$[\sum_{i=1}^{b} (s_i(t,s) - s_i^*)^2]^{r_1} [\sum_{i=1}^{b} e_i^2(t)]^{r_2} [\sum_{i=1}^{b} [\nabla s_i(t,s) \cdot (s' - s)]^2]^{r_3} [2 \sum_{i=1}^{b} e_i(t)(s_i(t,s) - s_i^*)]^{r_4}$$
$$[2 \sum_{i=1}^{b} (s_i(t,s) - s_i^*) \nabla s_i(t,s) \cdot (s' - s)]^{r_5} [2 \sum_{i=1}^{b} e_i(t) \nabla s_i(t,s) \cdot (s' - s)]^{r_6}$$
$$\leq \sum_{\sum_{k=1}^{6} r_k = r, r_k \geq 0} \binom{r}{r_1} \binom{r - r_1}{r_2} \binom{r - r_1 - r_2}{r_3} \binom{r - r_1 - r_2 - r_3}{r_4} \binom{r - r_1 - r_2 - r_3 - r_4}{r_5}$$
$$[\sum_{i=1}^{b} (s_i(t,s) - s_i^*)^2]^{r_1} [\sum_{i=1}^{b} e_i^2(t)]^{r_2} [\sum_{i=1}^{b} [\nabla s_i(t,s) \cdot (s' - s)]^2]^{r_3} [2 \sum_{i=1}^{b} |e_i(t)(s_i(t,s) - s_i^*)|]^{r_4}$$
$$[2 \sum_{i=1}^{b} |(s_i(t,s) - s_i^*) \nabla s_i(t,s) \cdot (s' - s)|]^{r_5} [2 \sum_{i=1}^{b} |e_i(t) \nabla s_i(t,s) \cdot (s' - s)|]^{r_6}$$
$$\leq \sum_{\sum_{k=1}^{6} r_k = r, r_k \geq 0} \binom{r}{r_1} \binom{r - r_1}{r_2} \binom{r - r_1 - r_2}{r_3} \binom{r - r_1 - r_2 - r_3}{r_4} \binom{r - r_1 - r_2 - r_3 - r_4}{r_5}$$
$$[\sum_{i=1}^{b} (s_i(t,s) - s_i^*)^2]^{r_1} [\sum_{i=1}^{b} e_i^2(t)]^{r_2} [\sum_{i=1}^{b} [\nabla s_i(t,s) \cdot (s' - s)]^2]^{r_3} [2 \sum_{i=1}^{b} |e_i(t)| \cdot |s_i(t,s) - s_i^*|]^{r_4}$$

$$[2\sum_{i=1}^{b}|s_i(t,s)-s_i^*|\cdot|\nabla s_i(t,s)\cdot(s'-s)|]^{r_5}[2\sum_{i=1}^{b}|e_i(t)|\cdot|\nabla s_i(t,s)\cdot(s'-s)|]^{r_6}$$

$$\leq \sum_{\sum_{k=1}^{6}r_k=r, r_k\geq 0}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2}})O(\frac{1}{N^{2r_3}})O(\frac{1}{N^{(2-2\alpha-4\xi)r_4}})[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_4}$$

$$O(\frac{1}{N^{r_5}})[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_5}O(\frac{1}{N^{(3-2\alpha-4\xi)r_6}})$$

$$\leq \sum_{\sum_{k=1}^{6}r_k=r, r_k\geq 0}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6}})[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}$$

$$[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_4+r_5} \tag{5.2}$$

When $r_1 = r$ and $r_i = 0$ for $i = 2, \cdots, 6$, the summand is $[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^r$; and when $r_1 = r-1, r_5 = 1$ and $r_i = 0$ for $i = 2,3,4,6$, the summand is $2r[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r-1}\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)\cdot(s'-s)$. Let Σ be the collection of combination of all $\{r_i\}_{i=1,\cdots,6}$ that excludes above two cases, i.e.

$$\Sigma = \{r_i, i=1,\cdots,6|\sum_{i=1}^{6}=r\}$$
$$\setminus\{\{r_1=r, r_i=0, \text{for} \quad i=2,\cdots,6\},\{r_1=r-1, r_5=1, r_i=0, \text{for} \quad i=2,3,4,6\}\}.$$

Then, substitute into the following equation

$$-(g(s')-g(s)-\nabla g(s)\cdot(s'-s))$$
$$=\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s')-s_i^*)^2]^r dt-\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^r dt$$
$$-2r(s'-s)\cdot\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r-1}\sum_{i=1}^{b}(s_i(t,s)-s_i^*)\nabla s_i(t,s)dt$$
$$\leq\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6}})\int_0^\infty[\sum_{i=1}^{b}(s_i(t,s)-s_i^*)^2]^{r_1}$$

$$[\sum_{i=1}^{b}|s_i(t,s)-s_i^*|]^{r_4+r_5}dt$$

$$\leq\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6}})\int_0^\infty(||s-s^*||\kappa e^{-\alpha t})^{2r_1}$$

$$[b\kappa||s-s^*||e^{-\alpha t}]^{r_4+r_5}dt$$

$$=\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{1}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6}})b^{r_4+r_5}k^{2r_1+r_4+r_5}\frac{1}{\alpha(2r_1+r_4+r_5)}$$

$$||s-s^*||^{2r_1+r_4+r_5}$$

$$=\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{b^{r_4+r_5}}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6}})||s-s^*||^{2r_1+r_4+r_5}$$

Therefore,

$$E[||S-s^*||^{2r}]$$

$$=E[-\sum_{s'\neq S}q_{S,s'}N(g(s')-g(S)-\nabla g(S)\cdot(s'-S))]$$

$$\leq E[\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{b^{r_4+r_5}}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6-1}})||S-s^*||^{2r_1+r_4+r_5}\sum_{s'\neq s}q_{s,s'}]$$

$$=\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{b^{r_4+r_5}}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6-1}})E[||S-s^*||^{2(r_1+\frac{r_4+r_5}{2})}]$$

$$\leq\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{b^{r_4+r_5}}{N^{2(2-2\alpha-4\xi)r_2+2r_3+(2-2\alpha-4\xi)r_4+r_5+(3-2\alpha-4\xi)r_6-1}})O(\frac{1}{N^{(r_1+\frac{r_4+r_5}{2})(1-4\alpha-\xi')}})$$

$$=\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{b^{r_4+r_5}}{N^{(1-4\alpha-\xi')r_1+2(2-2\alpha-4\xi)r_2+2r_3+(2.5-4\alpha-4\xi-0.5\xi')r_4+(1.5-2\alpha-0.5\xi')r_5+(3-2\alpha-4\xi)r_6-1}})$$

$$=\sum_{\Sigma}\binom{r}{r_1}\binom{r-r_1}{r_2}\binom{r-r_1-r_2}{r_3}\binom{r-r_1-r_2-r_3}{r_4}\binom{r-r_1-r_2-r_3-r_4}{r_5}$$

$$O(\frac{b^{r_4+r_5}}{N^{(1-4\alpha-\xi')r+(3-8\xi+\xi')r_2+(1+4\alpha+\xi')r_3+(1.5-4\xi+0.5\xi')r_4+(0.5+2\alpha+0.5\xi')r_5+(2+2\alpha-4\xi-\xi')r_6-1}})$$

$$\leq O(\frac{\log^2 N}{N^{(1-4\alpha-\xi')r+4\alpha+\xi'}}) \leq \frac{1}{N^{(1-4\alpha-\xi')r}}$$

where in the first inequality we used the fact that $\sum_{s'\neq s} q_{s,s'} \leq 2$; the second inequality, we assumed that $E[||S-s^*||^{2(r-1)}] \leq O(\frac{1}{N^{(1-4\alpha-\xi')(r-1)}})$ and by Lyapunov inequality we have $E[||S-s^*||^{2(r_1+\frac{r_4+r_5}{2})}] \leq (E[||S-s^*||^{2(r-1)}])^{\frac{2(r_1+\frac{r_4+r_5}{2})}{r-1}}$; the second from the last inequality is because order-wise it's the smallest when $r_1 = r-2, r_5 = 2$ and $r_i = 0$ for $i = 2,3,4,6$. And the last inequality holds for sufficiently large N.

When $r_1 = r_4 = r_5 = 0$, we have similar analysis based on the fact that $\int_0^\infty |e(t)|dt = O(\frac{1}{N^{2-4\alpha-5\xi}})$.

Therefore, Eq.(25) holds for all $r \in \mathbb{N}$, by mathematical induction.

5.2 State Space Collapse

Let ϵ be a positive number. Applying Markov inequality we have

$$\mathbb{P}\{||S-s^*||^{2r} \geq \epsilon\} \leq \frac{\mathbb{E}[||S-s^*||^{2r}]}{\epsilon} = \frac{1}{\epsilon}\frac{1}{N^{r(1-4\alpha-\xi')}}$$

Let $\epsilon = \frac{1}{N^\beta}$, where $\beta > 0$. Then, the above inequality will become

$$\mathbb{P}\{||S-s^*||^{2r} \geq \frac{1}{N^\beta}\} \leq \frac{1}{N^{r(1-4\alpha-\xi')-\beta}} \quad (5.3)$$

Define a set of states to be $\mathcal{B} = \{s \mid ||s - s^*||^{2r} \leq \frac{1}{N^\epsilon}\}$, which is close to equilibrium point. Let $d(s, s^*) = ||s - s^*||^2$ as the distance function. We consider a simple linear system

$$\dot{s} = l(s) = J(s^*)(s - s^*), \tag{5.4}$$

where $J(s^*)$ is the Jacobian matrix of $f(s)$ at the equilibrium point s^*. In the heavy traffic, the entries of $J(s^*)$ is generally also a function of N as is s^* itself.

The Jacobian matrix at a point s, is following

$$J(s) = \begin{bmatrix} -2\lambda s_1 - 1 & 1 & & 0 \\ 2\lambda s_1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 2\lambda s_{b-1} & -2\lambda s_b - 1 \end{bmatrix}$$

We first introduce a lemma that matrix $J(s^*)$ is invertible, i.e. $J(s^*)^{-1}$ exists.

Lemma 18 (Invertibility). *For any $s \in \mathcal{S}$, the Jacobian matrix $J(s)$ is invertible.*

Proof. Since it's a tridiagonal matrix, we can write down the determinant in a recursive form

$$P_i = -(2\lambda s_i + 1)P_{i-1} - 2\lambda s_{i-1} P_{i-2}$$

where

$$P_i = \begin{vmatrix} -2\lambda s_1 - 1 & 1 & & 0 \\ 2\lambda s_1 & \ddots & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 2\lambda s_{i-1} & -2\lambda s_i - 1 \end{vmatrix}$$

Furthermore, we can verify that in fact, P_i can be written in the following form

$$P_i = (-1)^i - 2\lambda s_i P_{i-1} \tag{5.5}$$

with $P_1 = -(2\lambda s_1 + 1)$. We can conclude 2 things from equation (5.5), for any $s \in \mathcal{S}$,

- the sign of P_i alternates: when i is odd, $P_i < 0$; when i is even, $P_i > 0$.
- the absolute value of P_i is no less than 1, i.e. $|P_i| \geq 1$

Because the determinant is nonzero, $J(s)$ is invertible. □

Consider a function $g : \mathcal{S} \to \mathcal{S}$ such that it satisfies the following equation

$$Lg(s) \doteq \frac{dg(s)}{dt} = \nabla g(s) \cdot l(s) = ||s - s^*||^2. \tag{5.6}$$

According to the definition of the linear mean-field model in (5.4), we have

$$\nabla g(s) \cdot J(s^*)(s - s^*) = ||s - s^*||^2. \tag{5.7}$$

Lemma 19 (Solution of Poisson Equation)**.** *The solution of the Poisson equation* (5.7) *is*

$$\nabla g(s) = [J^T(s^*)]^{-1}(s - s^*) \tag{5.8}$$

and furthermore

$$\nabla^2 g(s) = [J^T(s^*)]^{-1}$$
$$\nabla^3 g(s) = 0$$

□

Proof. According to Poisson equation (5.7), we have

$$\nabla g(s)^T J(s^*)(s - s^*) = (s - s^*)^T (s - s^*).$$

It implies

$$[\nabla g(s)^T J(s^*) - (s - s^*)^T](s - s^*) = 0$$

holds for all s in the neighborhood of s^*. If we set

$$\nabla g(s)^T J(s^*) - (s - s^*)^T = 0$$

then, we can solve that

$$\nabla g(s) = [J^T(s^*)]^{-1}(s - s^*)$$

and the higher derivatives as in the lemma. □

5.4 Upper Bound on Hessian Matrix $\nabla^2 g(s)$

In the following two sections, we will characterize both Hessian matrix $\nabla^2 g(s)$ and function $\tilde{f}(s)$ for states s that are close to equilibrium point s^*.

Lemma 20 (Upper Bound on Elements of Matrix $J(s^*)$)**.** *For all $i, j = 1, \cdots, b$ and sufficiently large N, we have*

$$|[J(s^*)]^{-1}_{ij}| \leq \frac{12}{\gamma} N^{2\alpha+2\xi}$$

□

Proof. First, we show that for any $\Phi \in R^b \setminus \{0\}$, we have

$$\frac{||J(s^*)\Phi||}{||\Phi||} \geq \delta_0$$

where δ_0 is the absolute value of the negative drift of the traditional mean-field model and $\delta_0 = \frac{12}{\gamma} N^{2\alpha+2\xi}$. For details, please refer to the technical report.

Since $J(s^*)$ is a tridiagonal matrix that satisfies $J(s^*)_{i,i+1} J(s^*)_{i+1,i} > 0$ for all i, we know that $J(s^*)$ can be diagonalized and the eigenvalues are all real. Also, we know eigenvalues are negative from the fact $J(s^*)$ is a Hurwitz matrix.

Define following Lyapunov functions

$$L_2(s) = \sqrt{\sum_{k=1}^{b}(s_k - s_k^*)^2}$$
$$L_w(s) = \sum_{k=1}^{b} w_k|s_k - s_k^*|$$

where $w_k \geq 1, k = 1, \cdots, b$ are defined in technical report. We have following inequalities

$$L_2(s) \leq L_w(s)$$

And we have the immediate result from the exponential convergence, for $x(t)$ of linear dynamical system $\dot{s}(t) = J(s^*)(s(t) - s^*)$, that

$$L_2(s(t)) = \sqrt{\sum_{k=1}^{b}(s_k(t) - s_k^*)^2} \leq \kappa \exp(-\delta_0 t)$$

for some $\kappa > 0$ and $t \geq 0$. The proof for exponential convergence of the linear system to the equilibrium point is very similar to that of traditional mean-field system.

Since $J(s^*)$ is diagonalizable, then any vector in an b-dimensional space can be represented by a linear combination of the orthonormal eigenvectors $r_k, \quad k = 1, \cdots, b$ of the matrix $J(s^*)$. Suppose the eigenvalues are $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_b < 0$.

$$s_0 - s^* = x_0 = \sum_{i=1}^{b} \alpha_i r_i$$

for some $\alpha_i \in R$ and $i = 1, \cdots, b$. Therefore, the general solution, for $s(t)$ of linear dynamical system $\dot{s}(t) = J(s^*)(s(t) - s^*)$, is a linear combination of the individual solutions for the eigenvectors

$$s(t) - s^* = \sum_{i=1}^{b} \alpha_i r_i \exp(\mu_i t).$$

So

$$L_2(s(t)) = ||\sum_{i=1}^{b} \alpha_i r_i \exp(\mu_i t)|| \leq \kappa \exp(-\delta_0 t)$$

Since this is true for all $x_0 \in \mathbb{R}^b$, we can choose an initial condition such that $\alpha_i = 0$ for $i = 1, \cdots, b-1$ such that for all $t \geq 0$

$$L_2(s(t)) = ||\alpha_b \exp(\mu_b t)|| \leq \kappa \exp(-\delta_0 t)$$

Thus we conclude

$$\mu_b \leq -\delta_0.$$

As a result, for any $\Phi \in R^b \setminus \{0\}$, we have,for some $\beta_i \in R$ and $i = 1, \cdots, b$,

$$\begin{aligned}
\Phi &= \beta_1 r_1 + \beta_2 r_2 + \cdots + \beta_b r_b \\
J(s^*)\Phi &= \beta_1 J(s^*) r_1 + \beta_2 J(s^*) r_2 + \cdots + \beta_b J(s^*) r_b \\
&= \beta_1 \mu_1 r_1 + \beta_2 \mu_2 r_2 + \cdots + \beta_b \mu_b r_b
\end{aligned}$$

so

$$\frac{||J(s^*)\Phi||}{||\Phi||} = \frac{\sqrt{\sum_{i=1}^{b} \beta_i^2 \mu_i^2}}{\sqrt{\sum_{i=1}^{b} \beta_i^2}} \geq \frac{\sqrt{\mu_b^2 \sum_{i=1}^{b} \beta_i^2}}{\sqrt{\sum_{i=1}^{b} \beta_i^2}} = |\mu_b| \geq \delta_0$$

Based on the paper of Robinson and Wathen Robinson and Wathen (1992), let $x = y = 0$ for both diagonal element Eq.(4.5) and non-diagonal elements Eq.(4.7). We will have upper bound for any $i, j = 1, \cdots, b$

$$|[J(s^*)]_{ij}^{-1}| \leq \frac{1}{\delta_0} \leq \frac{12}{\gamma} N^{2\alpha+2\xi}$$

□

Lemma 21 (Lower Bounds on Diagonal Elements of Matrix $J^{-1}(s^*)$)**.** *For tridiagonal matrix $J^{-1}(s^*)$, we have that*

$$|J^{-1}_{11}(s^*)| \geq \frac{1}{3} \tag{5.9}$$

and for all $i = 1, \cdots, b$, we have $J^{-1}_{ii}(s^*) < 0$. □

Proof. Given an $n \times n$ tridiagonal matrix G_n with entries denoted as following

$$G_n = \begin{bmatrix} x_1 & y_1 & & 0 \\ z_1 & x_2 & \ddots & \\ & \ddots & \ddots & y_{n-1} \\ 0 & & z_{n-1} & x_n \end{bmatrix}.$$

According to the paper of Kılıç (2008), define a backward continued fraction C_n by the entries of G_n as follows:

$$\begin{aligned} C_n &= [x_1 + \frac{-y_1 z_1}{x_2+} \frac{-y_2 z_2}{x_3+} \cdots \frac{-y_{n-1} z_{n-1}}{x_n}] \\ &= x_n + \cfrac{-y_{n-1} z_{n-1}}{x_{n-1} + \cfrac{-y_{n-2} z_{n-2}}{\ddots_{x_2 + \frac{-y_1 z_1}{x_0}}}} \end{aligned}$$

Let the sequence $\{P_n\}$ be for $1 \leq k \leq n-1$

$$P_{k+1} = x_{k+1} P_k - y_k z_k P_{k-1}$$

where $P_0 = 1$, $P_1 = x_1$. From the proof of Lemma , we know the sequence is also the iterative equation to calculate the determinant of $J(s^*)$. We introduce the following theorems from Kılıç (2008) to apply to our case.

Theorem 6. *Let the $n \times n$ tridiagonal matrix G_n have the form above. Let $G_n^{-1} = [w_{ij}]$ denote the inverse of G_n. Then*

$$w_{ii} = \frac{1}{C_i} + \sum_{k=i+1}^{n} \left(\frac{1}{C_k} \prod_{t=i}^{k-1} \frac{y_t z_t}{(C_t)^2}\right)$$

Theorem 7. *Let the matrix G_n be as above. Then for $n \geq 1$*

$$detG_n = P_n$$

Theorem 8. *Given a general backward continued function $A = [a_0 + \frac{b_1}{a_1+} \frac{b_2}{a_2+} \cdots \frac{b_n}{a_n}]$. If $0 \leq k \leq n$ and C_k is the kth backward convergent to A, that is $C_k = [a_0 + \frac{b_1}{a_1+} \frac{b_2}{a_2+} \cdots \frac{b_k}{a_k}]$, then $C_k = \frac{P_k}{P_{k-1}}$.*

Thus some of the convergents of C_n are

$$C_1 = [x_1] = \frac{P_1}{P_0} = x_1,$$
$$C_2 = [x_1 + \frac{-y_1 z_1}{x_2}] = \frac{P_2}{P_1} = \frac{x_1 x_2 - y_1 z_1}{x_1},$$

So in our case, we have that for $i = 1, \cdots, b-1$

$$y_i = 1$$

$$z_i = 2\lambda s_i^*$$

and for $i = 1, \cdots, b$

$$x_i = -2\lambda s_i^* - 1$$

Thus, we have

$$C_1 = x_1 = -2\lambda s_1^* - 1,$$
$$C_2 = \frac{x_1 x_2 - y_1 z_1}{x_1} = -2\lambda s_2^* - 1 + \frac{2\lambda s_1^*}{2\lambda s_1^* + 1} = -2\lambda s_2^* - \frac{1}{2\lambda s_1^* + 1},$$

Note that the sequence $\{P_n\}$ is the $n \times n$ size of Jacobian matrix's determinant, and we know that the sign of P_n alternates, thus $C_k = \frac{P_k}{P_{k-1}} < 0$ for all $k = 1, \cdots, n$. Furthermore, from theorem 4 and 6, we know that $J^{-1}(s^*)_{ii} < 0$ for all $i = 1, \cdots, b$.

Therefore, we have

$$J^{-1}(s^*)_{11} = \frac{1}{C_1} + \sum_{k=2}^{n} (\frac{1}{C_k} \prod_{t=1}^{k-1} \frac{2\lambda s_t^*}{(C_t)^2}) < 0$$

and

$$|J^{-1}(s^*)_{11}| \geq \frac{1}{|C_1|} \geq \frac{1}{3}$$

the last inequality is because $0 \leq s_1^* \leq 1$.

Therefore, $J^{-1}(s^*)_{11} \geq \frac{1}{3}$. □

5.6 Stein's Method

We have the generator difference as following

$$\mathbb{E}[d(S,s^*)\Big|||S-s^*||^{2r} \leq \frac{1}{N^\epsilon}] = \mathbb{E}[Lg(S) - Gg(S)\Big|||S-s^*||^{2r} \leq \frac{1}{N^\epsilon}]$$

We will expand the terms in generator equation and analyze the terms one by one.

5.6.1 Generator for the Stochastic System

Lemma 22. *The generator of function $g(s)$ is following*

$$Gg(s) = \nabla g(s) \cdot f(s) + \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(s)_{ii}\tilde{f}_i(s) \tag{5.10}$$

where $\nabla^2 g(s)_{ii}$ is the i-th diagonal element of the Hessian matrix $\nabla^2 g(s)$ and $\tilde{f}_i(s) = \frac{1}{2}[\lambda(s_{i-1}^2 - s_i^2) + (s_i - s_{i+1})]$.

Proof. Take a closer look at the generator by Taylor expansion around a state s

$$\begin{aligned}
&Gg(s)\\
&= \sum_{i=1}^{b} \lambda N(s_{i-1}^2 - s_i^2)[g(s+e_i) - g(s)] + N(s_i - s_{i+1})[g(s-e_i) - g(s)]\\
&= \sum_{i=1}^{b} \lambda N(s_{i-1}^2 - s_i^2)[\nabla g(s)\cdot e_i + \frac{1}{2}e_i^T\nabla^2 g(s)e_i]\\
&\quad + N(s_i - s_{i+1})[\nabla g(s)\cdot(-e_i) + \frac{1}{2}e_i^T\nabla^2 g(s)e_i]
\end{aligned}$$

$$= \sum_{i=1}^{b} \nabla g(s) \cdot [\lambda(s_{i-1}^2 - s_i^2) - (s_i - s_{i+1})] N e_i$$
$$+ \frac{1}{2} N e_i^T \nabla^2 g(s) e_i [\lambda(s_{i-1}^2 - s_i^2) + (s_i - s_{i+1})]$$
$$= \nabla g(s) \cdot f(s) + \frac{1}{N} \sum_{i=1}^{b} \nabla^2 g(s)_{ii} \tilde{f}_i(s)$$

Note that the second equality is because $\nabla^3 g(s) = 0$ from Lemma 19. □

5.6.2 Function $\tilde{f}(s)$

Since when s is close to equilibrium point s^*, define $x_i = s_i - s_i^*$, we can further make a Taylor expansion at the equilibrium point s^* as following

$$\begin{aligned}
\tilde{f}_i(s) &= \frac{1}{2}[\lambda(s_{i-1}^2 - s_i^2) + (s_i - s_{i+1})] \\
&= \frac{\lambda}{2}[(s_{i-1}^* + x_{i-1})^2 - (s_i^* + x_i)^2] + \frac{1}{2}(s_i^* + x_i - s_{i+1}^* - x_{i+1}) \\
&= \frac{\lambda}{2}[(s_{i-1}^*)^2 + 2x_{i-1}s_{i-1}^* + x_{i-1}^2 - (s_i^*)^2 - 2x_i s_i^* - x_i^2] + \frac{1}{2}(s_i^* + x_i - s_{i+1}^* - x_{i+1}) \\
&= \frac{\lambda}{2}[(s_{i-1}^*)^2 - (s_i^*)^2] + \frac{1}{2}(s_i^* - s_{i+1}^*) + O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \\
&= \tilde{f}_i(s^*) + O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \qquad (5.11)
\end{aligned}$$

The last equality is because of the fact that $||s - s^*||^{2r} \leq \frac{1}{N^\epsilon}$, which implies $|x_i| \leq \frac{1}{N^{\frac{\epsilon}{2r}}}$.

5.6.3 Generator Difference

The mean square error close to the equilibrium point is

$$\begin{aligned}
&\mathbb{E}[d(S, s^*) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[Lg(S) - Gg(S) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[\nabla g(S) \cdot J(s^*)(S - s^*) - \nabla g(S) \cdot f(S) - \frac{1}{N} \sum_{i=1}^{b} \nabla^2 g(S)_{ii} \tilde{f}_i(S) \mid ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[\nabla g(S) \cdot (J(s^*)(S - s^*) - f(S)) - \frac{1}{N} \sum_{i=1}^{b} \nabla^2 g(S)_{ii} \left(\tilde{f}_i(s^*) + O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \right)
\end{aligned}$$

$$
\begin{aligned}
&\Big| \, ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[\nabla g(S) \cdot (J(s^*)(S - s^*) - f(S)) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} \tilde{f}_i(s^*) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \\
&\Big| \, ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}].
\end{aligned}
$$

According to Lemma 19, we have

$$
\begin{aligned}
\nabla g(s) &= [J^T(s^*)]^{-1}(s - s^*), \\
\nabla^2 g(s) &= [J^T(s^*)]^{-1}
\end{aligned}
$$

to be the functions of $J(s^*)$, which is a function of N.

And in fact, since the traditional mean-field is a second-order system, we can write it down as

$$
f(s) = f(s^*) + J(s^*)(s - s^*) + \frac{1}{2} < s - s^*, \nabla^2 f(s^*)(s - s^*) >
$$

where $\nabla^2 f(s^*)$ is the Hessian of $f(s)$ at equilibrium point. For any $s \in \mathcal{S}$ and $i = 1, \cdots, b$, the Hessian has following form for f_i

$$
\nabla^2 f_i(s)_{kj} = \frac{\partial^2 f_i(s)}{\partial s_j \partial s_k} = \begin{cases} -2\lambda, & \text{if } j = k = i \\ 2\lambda, & \text{if } j = k = i - 1 \\ 0, & \text{otherwise} \end{cases}
$$

Substituting back to generator difference, we have

$$
\begin{aligned}
&\mathbb{E}[d(S, s^*) \, \Big| \, ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}[Lg(S) - Gg(S) \, \Big| \, ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}] \\
=&\mathbb{E}\big[[J^T(s^*)]^{-1}(S - s^*) \cdot (\frac{1}{2} < S - s^*, \nabla^2 f(s^*)(S - s^*) >) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} \tilde{f}_i(s^*) \\
&- \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(S)_{ii} O(\frac{1}{N^{\frac{\epsilon}{2r}}}) \, \Big| \, ||S - s^*||^{2r} \leq \frac{1}{N^\epsilon}\big].
\end{aligned} \tag{5.12}
$$

There are 3 terms in equation (5.12). Based on the two lemmas 20 and 21, we will characterize the upper bounds of each term and a lower bound for the second term in equation (5.12).

Lemma 23 (Upper Bound on the First Term)**.** *For the first term in equation (5.12), we have an upper bound on its 2-norm, when* $||s - s^*||^{2r} \leq \frac{1}{N^\epsilon}$*,*

$$||[J^T(s^*)]^{-1}(s - s^*) \cdot < s - s^*, \nabla^2 f(s^*)(s - s^*) > || = O(\frac{1}{N^{\frac{3\epsilon}{2r} - 2\alpha - 3\xi}}) \tag{5.13}$$

Proof. We have the 2-norm of the first term as following

$$\begin{aligned}
&||[J^T(s^*)]^{-1}(s - s^*) \cdot < s - s^*, \nabla^2 f(s^*)(s - s^*) > || \\
&\leq ||[J^T(s^*)]^{-1}(s - s^*)|| || < s - s^*, \nabla^2 f(s^*)(s - s^*) > || \\
&\leq ||[J^T(s^*)]^{-1}|| ||s - s^*|| || < s - s^*, \nabla^2 f(s^*)(s - s^*) > || \\
&\leq 2\sqrt{2}\lambda ||[J^T(s^*)]^{-1}|| ||s - s^*||^3
\end{aligned} \tag{5.14}$$

where the third inequality is because of followings

$$\begin{aligned}
& || < s - s^*, \nabla^2 f(s^*)(s - s^*) > || \\
=& \sqrt{\sum_{i=1}^{b} [(s - s^*) \nabla^2 f_i(s^*)(s - s^*)]^2} \\
=& \sqrt{\sum_{i=1}^{b} \left(2\lambda[(s_{i-1} - s_{i-1}^*)^2 - (s_i - s_i^*)^2]\right)^2} \\
=& 2\lambda \sqrt{\sum_{i=1}^{b} [(s_{i-1} - s_{i-1}^*)^2 - (s_i - s_i^*)^2]^2} \\
\leq& 2\lambda \sqrt{\sum_{i=1}^{b} (s_{i-1} - s_{i-1}^*)^4 + (s_i - s_i^*)^4} \\
\leq& 2\sqrt{2}\lambda \sqrt{\sum_{i=1}^{b} (s_i - s_i^*)^4} \\
\leq& 2\sqrt{2}\lambda \sqrt{[\sum_{i=1}^{b} (s_i - s_i^*)^2]^2} \\
=& 2\sqrt{2}\lambda ||s - s^*||^2
\end{aligned}$$

And from Lemma 20, for sufficiently large N, we have that

$$\begin{aligned}
||[J^T(s^*)]^{-1}|| = ||[J(s^*)]^{-1}|| \leq & \max_{ij} |[J(s^*)]_{ij}^{-1}| \times b^2 \\
=& O(N^{2\alpha+2\xi}) \times O(\log^2 N) \\
=& O(N^{2\alpha+3\xi}) \qquad (5.15)
\end{aligned}$$

Since $||s - s^*||^{2r} \leq \frac{1}{N^\epsilon}$, combining inequalities (5.14) and (5.15), we have

$$\begin{aligned}
& ||[J^T(s^*)]^{-1}(s - s^*) \cdot < s - s^*, \nabla^2 f(s^*)(s - s^*) > || \\
& \leq 2\sqrt{2}\lambda \times O(N^{2\alpha+3\xi}) \times \frac{1}{N^{\frac{3\epsilon}{2r}}} = O(\frac{1}{N^{\frac{3\epsilon}{2r} - 2\alpha - 3\xi}})
\end{aligned}$$

$\square$

Lemma 24 (Lower Bound on the Second Term)**.** *For the second term in equation (5.12), we have a lower bound as follwing*

$$-\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g_{ii}(s)\tilde{f}_i(s^*) \geq \frac{\lambda\gamma}{3N^{1+\alpha}} \tag{5.16}$$

Proof. Recall that $\nabla^2 g(s) = [J^T(s^*)]^{-1}$ and for $i = 1, \cdots, b$, $J^{-1}(s^*)_{ii} < 0$, by lemma 21. And it's easy to check, for $i = 1, \cdots, b$, $\tilde{f}_i(s^*) \geq 0$.Therefore, for $i = 1, \cdots, b$, we have

$$-\nabla^2 g(s)_{ii}\tilde{f}_i(s^*) \geq 0$$

And we also have

$$\begin{aligned}\tilde{f}_i(s^*) &= \frac{1}{2}[\lambda((s^*_{i-1})^2 - (s^*_i)^2) + (s^*_i - s^*_{i+1})] \\ &= \lambda[(s^*_{i-1})^2 - (s^*_i)^2]\end{aligned}$$

the second equality is because s^* is the equilibrium point. Thus, for $i = 1$, we have

$$\begin{aligned}\tilde{f}_1(s^*) = \lambda[1 - (s^*_1)^2] &\geq \lambda(1 - \lambda^2) \\ &= \lambda(1-\lambda)(1+\lambda) \geq \frac{\lambda\gamma}{N^\alpha}\end{aligned}$$

So, we have

$$\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g_{ii}(s)\tilde{f}_i(s^*) \geq -\frac{1}{N}J^{-1}_{11}(s^*)\tilde{f}_1(s^*) \geq \frac{\lambda\gamma}{3N^{1+\alpha}}$$

□

Lemma 25 (Upper Bound on the Second Term)**.** *For the second term in equation (5.12), we have an upper bound as following*

$$-\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g_{ii}(s)\tilde{f}_i(s^*) = O(\frac{1}{N^{1-2\alpha-3\xi}}) \tag{5.17}$$

Proof. It's easy to check that $f_i(s^*) \leq 1$ for $i = 1, \cdots, b$. And recall that $|\nabla^2 g(s)_{ii}| \leq O(N^{2\alpha+2\xi})$. Therefore

$$-\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g_{ii}(s)\tilde{f}_i(s^*) = \frac{b}{N}O(N^{2\alpha+2\xi}) = O(\frac{1}{N^{1-2\alpha-3\xi}})$$

□

Lemma 26 (Upper Bound on the Third Term)**.** *For sufficiently large N, we have an upper bound for the third term in equation (5.12) as following*

$$||-\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g(s)_{ii}O(\frac{1}{N^{\frac{\epsilon}{2r}}})|| = O(\frac{1}{N^{1+\frac{\epsilon}{2r}-2\alpha-3\xi}}) \tag{5.18}$$

Proof. Recall that $|\nabla^2 g(s)_{ii}| = O(N^{2\alpha+2\xi})$ for $i = 1, \cdots, b$. Thus, we have

$$||-\frac{1}{N}\sum_{i=1}^{b}\nabla^2 g(s)_{ii}O(\frac{1}{N^{\frac{\epsilon}{2r}}})|| \leq \frac{b}{N}O(N^{2\alpha+2\xi})\cdot O(\frac{1}{N^{\frac{\epsilon}{2r}}}) = O(\frac{1}{N^{1+\frac{\epsilon}{2r}-2\alpha-3\xi}})$$

□

Based on these lemmas, we are now able to characterize the dominant term among the three in equation (5.12).

Lemma 27. *For $0 < \alpha < \frac{1}{18}$ and sufficiently large N, we have for equation (5.12)*

$$E[d(S,s^*)\big|||S-s^*||^{2r} \leq \frac{1}{N^{\epsilon}}] = -\frac{1}{N}\sum_{i=1}^{b}[J^T(s^*)]^{-1}_{ii}\tilde{f}_i(s^*) + o(\frac{1}{N^{1+\alpha}}) \tag{5.19}$$

where the parameters satisfy following

$$\frac{3(1+\alpha)}{1-18\alpha-27\xi} < r$$
$$\frac{2r(1+3\alpha+3\xi)}{3} < \epsilon < r(1-4\alpha-7\xi) - 1 - \alpha - \xi$$

Proof. For the given condition of the parameters, it's easy to check that the upper bounds of the first and third term in equation (5.12) is order-wise smaller than the lower bounds of the second term, i.e.

$$\frac{3\epsilon}{2r} - 2\alpha - 3\xi > 1 + 3\alpha + 3\xi - 2\alpha - 3\xi = 1 + \alpha$$

and

$$\begin{aligned} 1 + \frac{\epsilon}{2r} - 2\alpha - 3\xi &> 1 + \frac{1}{3} + \alpha + \xi - 2\alpha - 3\xi \\ &> (1 + \alpha) + (\frac{2}{9} - 2\xi) \end{aligned}$$

where the last inequality is by the fact $0 < \alpha < \frac{1}{18}$. □

5.7 Main Results

In this section, we present our main results using 3 theorems. These are the results based on the observation that the second term in equation (5.12) is the dominant term.

Theorem 9 (Heavy Traffic Convergence)**.** *For $0 < \alpha < \frac{1}{18}$ and sufficiently large N, we have that*

$$\mathbb{E}[||S - s^*||^2] = -\frac{1}{N} \sum_{i=1}^{b} [J^T(s^*)]_{ii}^{-1} \tilde{f}_i(s^*) + o(\frac{1}{N^{1+\alpha}}) \tag{5.20}$$

where $J(s^*)$ is the Jacobian matrix of traditional mean-field model $f(s)$ at equilibrium point s^* and $\tilde{f}_i(s^*) = \frac{1}{2}[\lambda \left((s^*_{i-1})^2 - (s^*_i)^2\right) + (s^*_i - s^*_{i+1})]$ for $i = 1, 2, \cdots, b$.

Proof. Let parameters satisfy following conditions

$$\begin{aligned} \frac{3(1+\alpha)}{1 - 18\alpha - 27\xi} &< r \\ \frac{2r(1 + 3\alpha + 3\xi)}{3} &< \epsilon < r(1 - 4\alpha - 7\xi) - 1 - \alpha - \xi \end{aligned}$$

and $\xi > 0$ is arbitrarily small. Then, for sufficiently large N, the mean square distance is

$$
\begin{aligned}
&E[||S-s^*||^2] \\
=&E[||S-s^*||^2 \big| ||S-s^*||^{2r} \geq \frac{1}{N^{\epsilon}}] \cdot \mathbb{P}\{||S-s^*||^{2r} \geq \frac{1}{N^{\epsilon}}\} \\
&+ E[||S-s^*||^2 \big| ||S-s^*||^{2r} \leq \frac{1}{N^{\epsilon}}] \cdot \mathbb{P}\{||S-s^*||^{2r} \leq \frac{1}{N^{\epsilon}}\} \\
=&O(\log N) \cdot \frac{1}{N^{r(1-4\alpha-7\xi)-\epsilon}} + [-\frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(s)_{ii} \tilde{f}_i(s^*) \\
&+ O(\frac{1}{N^{2+\epsilon-\alpha-3\xi}}) + o(\frac{1}{N^{1+\alpha}})] \cdot (1 - \frac{1}{N^{r(1-4\alpha-7\xi)-\epsilon}}) \\
=&O(\frac{1}{N^{r(1-4\alpha-7\xi)-\epsilon-\xi}}) - \frac{1}{N}\sum_{i=1}^{b} \nabla^2 g(s)_{ii} \tilde{f}_i(s^*) \\
&+ O(\frac{1}{N^{1-2\alpha-3\xi}}) \cdot \frac{1}{N^{r(1-4\alpha-7\xi)-\epsilon}} + o(\frac{1}{N^{1+\alpha}}) \\
=&-\frac{1}{N}\sum_{i=1}^{b} [J^T(s^*)]^{-1}_{ii} \tilde{f}_i(s^*) + o(\frac{1}{N^{1+\alpha}})
\end{aligned}
$$

where the second equality is because $||s-s^*||^2 \leq b = O(\log N)$. We note that with the choice of parameters r, ϵ and the fact $0 < \alpha < \frac{1}{18}$, the lower bound of the term $-\frac{1}{N}\sum_{i=1}^{b}[J^T(s^*)]^{-1}_{ii}\tilde{f}_i(s^*) + o(\frac{1}{N^{1+\alpha}})$ is $O(\frac{1}{N^{1+\alpha}})$, while the other terms are strictly upper bounded by this order for sufficiently large N. □

The theorem shows the mean square error $\mathbb{E}[||S-s^*||^2]$ consists of two terms: the first term $-\frac{1}{N}\sum_{i=1}^{b}[J^T(s^*)]^{-1}_{ii}\tilde{f}_i(s^*)$ is the dominant term and is calculated explicitly; the second term $o(\frac{1}{N^{1+\alpha}})$ is "small" when N is sufficiently large.

Theorem 10 (Convergence Upper Bound)**.** *For $0 < \alpha < \frac{1}{18}$ and sufficiently large N, we have that*

$$\mathbb{E}[||S-s^*||^2] \leq -\frac{4}{N}\sum_{i=1}^{b}[J^T(s^*)]^{-1}_{ii}\tilde{f}_i(s^*) \tag{5.21}$$

Proof. From lemma 27 with the same parameter conditions, it's easy to check that for sufficiently large N, we have

$$E[d(S,s^*)\big| ||S-s^*||^{2r} \leq \frac{1}{N^{\epsilon}}] \leq -\frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$$

Also, it's easy to check

$$\mathbb{P}\{||S-s^*||^{2r} \geq \frac{1}{N^{\epsilon}}\} \leq \frac{1}{N^{r(1-4\alpha-\xi')-\epsilon}} \leq \frac{1}{N^{1+\alpha+\xi}}$$

Then from the above two inequalites, for sufficiently large N, the mean square distance is

$$\begin{aligned}
&E[||S-s^*||^2] \\
=&E[||S-s^*||^2\big| ||S-s^*||^{2r} \geq \frac{1}{N^{\epsilon}}]\cdot\mathbb{P}\{||S-s^*||^{2r} \geq \frac{1}{N^{\epsilon}}\} \\
&+E[||S-s^*||^2\big| ||S-s^*||^{2r} \leq \frac{1}{N^{\epsilon}}]\cdot\mathbb{P}\{||S-s^*||^{2r} \leq \frac{1}{N^{\epsilon}}\} \\
\leq&\frac{b}{N^{1+\alpha+\xi}} - \frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) \\
\leq&\frac{1}{N^{1+\alpha}} - \frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) \\
\leq&-\frac{4}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)
\end{aligned}$$

where the second from the last inequality is because the first term is larger than RHS of the inequality (5.16). □

This result tells us that we can have a calculable upper bound for heavy traffic convergence for mean square error.

Theorem 11. *For $0 < \alpha < \frac{1}{12}$ and sufficiently large N, we have that*

$$\mathbb{E}[||S-s^*||^2] = O(\frac{1}{N^{1-2\alpha-3\xi}})$$

where $\xi > 0$ is arbitrarily small.

Proof. Let parameters satisfy following conditions

$$\frac{3(1-2\alpha-2\xi)}{1-12\alpha-21\xi} < r$$
$$\frac{2r}{3} < \epsilon < r(1-4\alpha-7\xi) - 1 + 2\alpha + 2\xi$$

Then for sufficiently large N, we have

$$\begin{aligned}
&E[||S-s^*||^2] \\
=&E[||S-s^*||^2 \big| ||S-s^*||^{2r} \geq \frac{1}{N^\epsilon}] \cdot \mathbb{P}\{||S-s^*||^{2r} \geq \frac{1}{N^\epsilon}\} \\
&+ E[||S-s^*||^2 \big| ||S-s^*||^{2r} \leq \frac{1}{N^\epsilon}] \cdot \mathbb{P}\{||S-s^*||^{2r} \leq \frac{1}{N^\epsilon}\} \\
\leq&\frac{b}{N^{1-2\alpha-2\xi}} - \frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) \\
\leq&\frac{1}{N^{1-2\alpha-3\xi}} - \frac{3}{N}\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*) \\
=&O(\frac{1}{N^{1-2\alpha-3\xi}})
\end{aligned}$$

where the last inequality is by inequality (5.17). □

This result tells us an order-wise upper bound on mean square error for a slightly larger range of α.

5.8 Simulation Results

We provide the results of some simulations for various γ and system sizes when $\alpha = 0.05$. The results are based on the average of 10 runs, where each run simulates 10^8 time steps. And the first 9×10^7 are ignored to give the system to reach steady state. We compared the results from the simulations to our dominant terms and upper bounds. The simulations are the results of empirical mean square error times the system size N. And our dominant terms and upper bounds are multiplied by system size, i.e. $-\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$ and $-4\sum_{i=1}^{b}[J^T(s^*)]_{ii}^{-1}\tilde{f}_i(s^*)$ respectively.

Table 5.1: $\gamma = 0.1, \alpha = 0.05$

N	10	100	1000	10000
λ	0.9109	0.9206	0.9292	0.9369
Prediction	1.6366	1.80	1.9886	2.1970
Upper Bound	6.5464	7.20	7.9544	8.7880
Simulation	4.2687	3.6443	4.0438	4.2011

Table 5.2: $\gamma = 0.01, \alpha = 0.05$

N	10	100	1000	10000
λ	0.9911	0.9921	0.9929	0.9937
Prediction	13.7883	15.4351	17.2826	19.3546
Upper Bound	55.1532	61.7404	69.1304	77.4184
Simulation	30.4452	48.8215	36.0871	35.5646

Table 5.1 and 5.2 show the comparisons between predictions and simulation results for $\gamma = 0.1, \alpha = 0.05$ and $\gamma = 0.01, \alpha = 0.05$ for different system sizes. The arrival rate is not only the function of system size and approaching 1 as N gets larger. As N increases, the simulation results are in the same order with the dominant terms and are bounded by the upper bounds. As we can see, the upper bound is enough for even moderately large system size, e.g. $N = 10$, which shows the effectiveness our refined upper bound. And from a practical point of view, the upper bound is calculable as compared to the convergence results in Ying (2017), where the upper bound contains an arbitrarily small number.

Chapter 6

CONCLUSION

In this dissertation, we studied the missing value recovery problems in time series data with the present of partially observed network information. Also, we studied the stationary distribution of the power-of-two-choices load balancing for many-server systems (N servers) to the solution of corresponding mean-field model in both light and heavy traffic regimes. We developed Stein's method and state space collapse framework to analyze the systems in steady state and characterized the convergence dominant terms and corresponding coefficients in both light and heavy traffic regimes.

Chapter 2 studied the missing value recovery problem in time series data. We first defined and formulated problem into a likelihood maximization problem. Then, using Kalman filtering and matrix factorization methods to tackle time series data and network data respectively, as a result proposed **NetDyna** algorithm. In the end, we experimented on real world datasets to show the efficiency of our algorithm in comparison with the state-of-the-art algorithms.

Chapter 3 studied the convergence rate coefficient of the Po2 load balancing algorithm to the solution of mean-field model in light traffic regime. Stein's method and state space collapse (SSC) are introduced and applied in this chapter and demonstrated to be a potential framework in steady-state analysis of such system. By using higher moment bounds based state space collapse argument, we are able to show that the state space concentrates around the solution of the mean-field model. And then using a linear mean-field model, we are able to characterize the convergence term mean square error distance. And the simulation result shows the theoretical prediction is close for light traffic case.

Chapter 4 studied the stochastic system converges to the mean-field model under the finite buffer size assumption where $b = O(\log N)$ in heavy traffic regime. A convergence result is derived using stein's method and propagation of chaos arguments. This is a convergence result needed for the next chapter in order to characterize the convergence dominant term.

Chapter 5 studied convergent rate coefficient in the heavy traffic regime. We used the similar Stein's method and state space collapse argument. And with a more sophisticated tridiagonal matrix analysis, we are still able to characterize the dominant term in the mean square error. And we are able to characterize the corresponding coefficient. The simulation result shows that the predictions are within 2 factor simulation results and as N becomes larger, the prediction is closer.

Bertsimas, D., D. Gamarnik and J. N. Tsitsiklis, "Geometric bounds for stationary distributions of infinite markov chains via lyapunov functions", (1998).

Bishop, C. M., *Pattern recognition and machine learning* (springer, 2006).

Byron, M. Y., K. V. Shenoy and M. Sahani, "Derivation of kalman filtering and smoothing equations", (2004).

Cai, Y., H. Tong, W. Fan and P. Ji, "Fast mining of a network of coevolving time series", in "Proceedings of the 2015 SIAM International Conference on Data Mining", pp. 298–306 (SIAM, 2015a).

Cai, Y., H. Tong, W. Fan, P. Ji and Q. He, "Facets: Fast comprehensive mining of coevolving high-order time series", in "Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", pp. 79–88 (ACM, 2015b).

Cao, W., D. Wang, J. Li, H. Zhou, L. Li and Y. Li, "Brits: Bidirectional recurrent imputation for time series", in "Advances in Neural Information Processing Systems", pp. 6775–6785 (2018).

Che, Z., S. Purushotham, K. Cho, D. Sontag and Y. Liu, "Recurrent neural networks for multivariate time series with missing values", Scientific reports **8**, 1, 6085 (2018).

Gast, N., "Expected values estimated via mean-field approximation are 1/n-accurate", Proc. ACM Meas. Anal. Comput. Syst. **1**, 1, 17:1–17:26 (2017).

Gast, N. and B. Van Houdt, "A refined mean field approximation", in "Proc. Ann. ACM SIGMETRICS Conf.", (Irvien, CA, 2018).

Gelb, A., A. S. C. T. Staff and A. S. Corporation, *Applied Optimal Estimation*, Applied Optimal Estimation (MIT Press, 1974).

Goodfellow, I., Y. Bengio and A. Courville, *Deep learning*, vol. 1 (MIT press Cambridge, 2016).

Kılıç, E., "Explicit formula for the inverse of a tridiagonal matrix by backward continued fractions", Applied Mathematics and Computation **197**, 1, 345–357 (2008).

Koren, Y., R. Bell and C. Volinsky, "Matrix factorization techniques for recommender systems", Computer **42**, 8 (2009).

Kurtz, T. G., "Limit theorems for sequences of jump markov processes approximating ordinary differential processes", J. Appl. Probab. **8**, 2, 344–356 (1971).

Kurtz, T. G., *Approximation of population processes*, vol. 36 (SIAM, 1981).

Li, L., *Fast algorithms for mining co-evolving time series*, Ph.D. thesis, Carnegie Mellon University (2011).

Li, L., J. McCann, N. S. Pollard and C. Faloutsos, "Dynammo: Mining and summarization of coevolving sequences with missing values", in "Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 507–516 (ACM, 2009).

Lipton, Z. C., D. C. Kale and R. Wetzel, "Modeling missing data in clinical time series with rnns", Machine Learning for Healthcare (2016).

Little, R. J. and D. B. Rubin, *Statistical analysis with missing data*, vol. 333 (John Wiley & Sons, 2014).

Liu, X. and L. Ying, "On achieving zero delay with power-of-d-choices load balancing", in "Proc. IEEE Int. Conf. Computer Communications (INFOCOM)", (Honolulu,Hawaii, 2018).

Ma, H., H. Yang, M. R. Lyu and I. King, "Sorec: social recommendation using probabilistic matrix factorization", in "Proceedings of the 17th ACM conference on Information and knowledge management", pp. 931–940 (ACM, 2008).

Mitzenmacher, M., *The Power of Two Choices in Randomized Load Balancing*, Ph.D. thesis, University of California at Berkeley (1996).

Mnih, A. and R. R. Salakhutdinov, "Probabilistic matrix factorization", in "Advances in neural information processing systems", pp. 1257–1264 (2008).

Papadimitriou, S., J. Sun and C. Faloutsos, "Streaming pattern discovery in multiple time-series", in "Proceedings of the 31st international conference on Very large data bases", pp. 697–708 (VLDB Endowment, 2005).

Robinson, P. D. and A. J. Wathen, "Variational bounds on the entries of the inverse of a matrix", IMA journal of numerical analysis **12**, 4, 463–486 (1992).

Srebro, N. and T. Jaakkola, "Weighted low-rank approximations", (ICML, 2003).

Stein, C., "A bound for the error in the normal approximation to the distribution of a sum of dependent random variables", in "Proc. Sixth Berkeley Symp. Math. Stat. Prob.", pp. 583–602 (1972).

Stein, C., "Approximate computation of expectations", Lecture Notes-Monograph Series **7**, i–164 (1986).

Vvedenskaya, N. D., R. L. Dobrushin and F. I. Karpelevich, "Queueing system with selection of the shortest of two queues: An asymptotic approach", Problemy Peredachi Informatsii **32**, 1, 20–34 (1996).

Yi, B.-K., N. D. Sidiropoulos, T. Johnson, H. Jagadish, C. Faloutsos and A. Biliris, "Online data mining for co-evolving time sequences", in "Data Engineering, 2000. Proceedings. 16th International Conference on", pp. 13–22 (IEEE, 2000).

Ying, L., "On the approximation error of mean-field models", in "Proc. Ann. ACM SIGMETRICS Conf.", (Antibes Juan-les-Pins, France, 2016).

Ying, L., "Stein's method for mean field approximations in light and heavy traffic regimes", Proc. ACM Meas. Anal. Comput. Syst. **1**, 1, 12:1–12:27 (2017).