THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF MATHEMATICS

MODEL REDUCTION THROUGH MORI-ZWANZIG FORMALISM

ZHENYU TIAN
SPRING 2019

A thesis
submitted in partial fulfillment
of the requirements
for baccalaureate degrees
in Mathematics and Statistics
with honors in Mathematics

Reviewed and approved* by the following:

John Harlim
Professor of Mathematics
Thesis Supervisor

Aissa Wade
rofessor of Mathematics
Honors Adviser

*Signatures are on file in the Schreyer Honors College.

Abstract

In this thesis, we investigate theoretical and numerical methods of Mori-Zwanzig Formalism, a formulation that allows us to estimate the solutions of nonlinear time-dependent problems when the full dynamics are too complex to be fully resolved or when part of the dynamics are unknown. We began by introducing the general goal and background knowledge for this thesis. At the heart of this thesis, we investigate the solutions of the Mori-Zwanzig projection using various choices of bases of L2 space, including the analytical basis functions that can be derived in special circumstances and the data-driven basis functions obtained via the diffusion map algorithm. We found that the quality of the solutions are sensitive to the choice of basis functions.

Table of Contents

List of Figures

Acknowledgements

First and foremost, I would like to thank Dr. John Harlim for his continuous support through the past year. Without his guidance, I may never have the opportunity to learn this great method. I also want to thank him for pushing me working on this project, which helped me to improve my coding and writing skills.

Also, I would like to thank my honors advisor Dr. Aissa Wade for her support in the past two years. Dr. Aissa's advices kept me on track in this rigorous honors program. Without her, I may not be able to graduate with honors title.

Lastly, I want to thank my math modeling class instructor, Dr. Conway, who introduced me to Dr. Harlim and supported me applying to Schreyer Honors College. Without her, this thesis would not been exist.

1 Introduction

This chapter provides the general goal and background knowledge for this thesis. We start by introducing the outline of goals for this project. Then, we will review the basics of Hamiltonian Systems introduced in Chorin and Hald's, *Stochastic Tools in Mathematics and Science* [1]. Finally, we will discuss an example that will be tested throughout this project.

1.1 Outline of Goals

Consider an ODE system:

$$\frac{d}{dt}\phi(t) = R(\phi(t)), \quad \phi(0) = x \tag{1.1}$$

where ϕ is an n-dimensional vector and R is an n-dimensional vector field. We denote $R = (\hat{R}, \tilde{R})$. In many applications, we are only interested in predicting the evolution of some statistics of ϕ in time. This problem can be challenging if the unresolved vector field $\tilde{R}$ is not available. The goal of this project is to reconstruct the missing dynamics from a full time series of the solutions of the model using the Mori-Zwanzig Formalsim.

Suppose the vector $\hat{\phi} = (\phi_1, \phi_2, ..., \phi_m)$ is the variable of interests, and $\hat{x} = (x_1, x_2, ..., x_m)$ is the initial data for $\hat{\phi}$. The initial condition for other components will be determined by some known pdf. The initial condition is chosen in the following way: $\hat{x}$ is chosen once, then the rest of the initial conditions are sampled from the known pdf. Our goal is to estimate the evolution of $\hat{\phi}$ in such congifuration. In the following section, we will review a Hamiltonian system, which will be used to realize the goal above.

1.2 Hamiltonian System

In this section, we will explain the concepts in section 1.1 with two-particle Hamiltonian system. Before that, we will first introduce the definition of Hamiltonian System.

Hamiltonian System is a dynamical system governed by Hamiltonian's equation $H(p, q, t)$, defined in $\Omega \times I$, where $(p, q) \in \Omega$ and $t \in I$. Here, Ω is the phase space, and $\Omega \times I$ is the extended

phase space. The evolution equation is given by:

$$\begin{aligned} \frac{\partial p}{\partial t} &= -\frac{\partial H}{\partial q} \\ \frac{\partial q}{\partial t} &= +\frac{\partial H}{\partial p} \end{aligned} \tag{1.2}$$

Given the initial condition $(p(0), q(0))$, the solution is defined by:

$$(p_i(t), q_i(t)) = \Phi_H(t, t_0)(p_i(0), q_i(0))$$

Here, $\Phi_H(t, t_0)$ is a transformation mapping the phase space into itself such that (p_i, q_i) is the value at time t [2]. p_i and q_i are the position and momenta of the ith particle, respectively.
Now, consider the following two-particle Hamiltonian system with Hamiltonian $H = \frac{1}{2}(q_1^2 + q_2^2 + q_1^2 q_2^2 + p_1^2 + p_2^2)$, where q_i and p_i, $i = 1, 2$. Using equation (1.2), the equations of motion are:

$$\begin{aligned} \dot{q_1} &= p_1 \\ \dot{p_1} &= -q_1(1 + q_2^2) \\ \dot{q_2} &= p_2 \\ \dot{p_2} &= -q_2(1 + q_1^2) \end{aligned} \tag{1.3}$$

Throughout this thesis, we will use the above system as a test problem. Suppose we are interested in predicting the state of the first particle, and we have the initial values $q_1(0) = 1$ and $p_1(0) = 0$. Assume that $q_2(0)$,$p_2(0)$ are sampled from the pdf $W = e^{-H(q,p)}/Z$ (canonical density with temperature 1). We want to calculate $E[q_1(t)|q_1(0), p_1(0)]$, $E[p_1(t)|q_1(0), p_1(0)]$, the expected values of $q_1(t)$ and $p_1(t)$ given their initial values. To solve this, one easy way is to sample $q_2(0)$,$p_2(0)$ from W and solve the full system. Then, one should repeat the step for each sample and average the values of $q_1(t)$ and $p_1(t)$. The expected values will tend to zero because the randomness of $q_2(0)$,$p_2(0)$ increases uncertainty as time grows. As a result, q_1 and p_1 will converge to the best estimate in absence of information, which is zero. The result is shown in the following graphs.

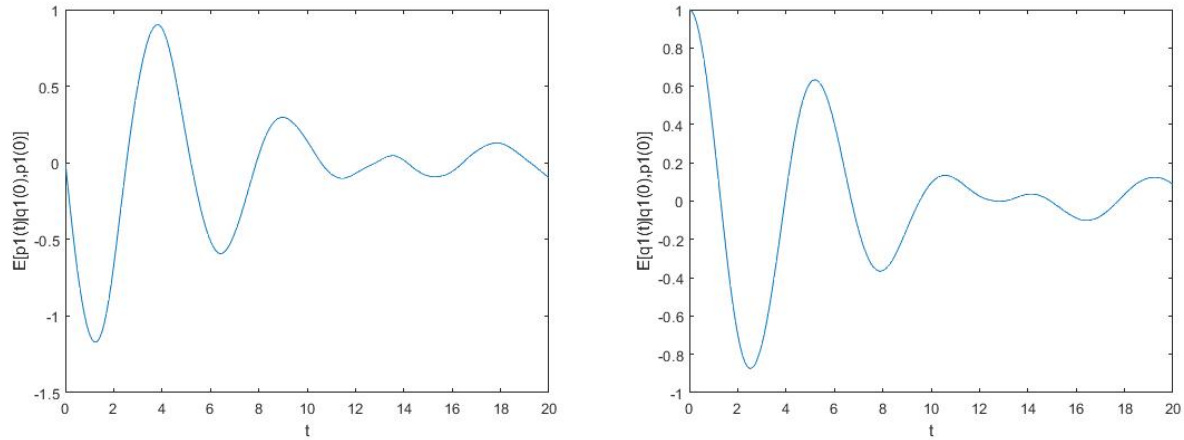

Figure 1.1: The figure above shows the expected value of $p_1(t)$ and $q_1(t)$ using Monte-Carlo method. The left figure corresponds to the expected value of $p_1(t)$. The right figure corresponds to the expected value of $q_1(t)$.

Our goal is to recover these statistics through the Mori-Zwanzig Formalism.

2 | Mori-Zwanzig Formalism

This chapter provides analytic methods of Mori-Zwanzig Formulation introduced in Chorin, Hald, and Kupferman's *Optimal prediction with memory*[3]. We begin by introducing Liouville equation. Then, we will project functions of L^2 onto the space $\hat{L}^2$ of functions of vector $\hat{x}$ using the basis we will define in Chapter 3. Finally, we will follow Mori-Zwanzig procedure and estimate the noise term.

Suppose the full dynamics are give on a set of ordinary differential equations, which is

$$\frac{d}{dt}\phi(t) = R(\phi(t)), \quad \phi(0) = x.$$

Now, we let $L = \sum_{i=1}^{n} R_i(x)\frac{\partial}{\partial_{x_i}}$ and consider the following partial differential equation

$$\frac{\partial}{\partial_t}u(x,t) = Lu(x,t), \quad u(x,0) = g(x). \tag{2.1}$$

This is the Liouville equation. The solution of this equation is $u(x,t) = g(\phi(x,t))$. To be specific, if $g(x) = x_i$, the solution is $u(x,t) = \phi_i(x,t)$. Moreover, $u(x,t)$ can be written in semigroup notation such that $u(x,t) = (e^{tL}g)(x) = g(\phi(x,t))$. Note that $e^{tL}L = Le^{tL}$, therefore, equation (2.1) becomes

$$\frac{\partial}{\partial_t}e^{tL}g = Le^{tL}g = e^{tL}Lg.$$

Now, define P as the projection of functions from space L^2 to space $\hat{L}^2$, the space formed by square integratable functions of x and $\hat{x}$, respectively. P is defined as:

$$Pf(\hat{x}) = \sum_{\mu \in I}(f, h^\mu)h^\mu(\hat{x}) \approx E[f|\hat{x}]$$

where h^μ forms an orthonormal basis of $\hat{L}^2$.

Next, we focus on a specific equation in the system $\phi_j(x,t) = e^{tL}x_j$ and split it into two terms:

$$\frac{\partial}{\partial t}e^{tL}x_j = e^{tL}Lx_j = e^{tL}PLx_j + e^{tL}QLx_j \tag{2.2}$$

where $Q = I - P$.

The first term in equation (2.2) can be elaborated as:

$$e^{tL}PLx_j = e^{tL}PR_j = e^{tL}\sum_{\mu}(R_j, h^{\mu})h^{\mu}(\hat{x}) = \sum_{\mu}(R_j, h^{\mu})h^{\mu}(\hat{\phi}(x,t)).$$

For the example in equation (1.3) we can deduce this term explicitly:

$$\begin{aligned}
e^{tL}PLx_1 &= E[R_1(\hat{\phi}(x,t))|\hat{x}] \\
&= \frac{\int R_1(\hat{\phi}(x,t))e^{-H(x)}d\tilde{x}}{\int e^{-H(x)}d\tilde{x}} \\
&= \frac{\int \phi_2(x,t)e^{-H(x)}d\tilde{x}}{\int e^{-H(x)}d\tilde{x}} \\
&= \phi_2(x,t) \\
e^{tL}PLx_2 &= E[R_2(\hat{\phi}(x,t))|\hat{x}] \\
&= \frac{\int R_2(\hat{\phi}(x,t))e^{-H(x)}d\tilde{x}}{\int e^{-H(x)}d\tilde{x}} \\
&= \frac{\int -\phi_1(x,t)(1+\phi_3^2(x,t))e^{-H(x)}d\tilde{x}}{\int e^{-H(x)}d\tilde{x}} \\
&= \frac{\int -\phi_1(x,t)e^{-H(x)}d\tilde{x}}{\int e^{-H(x)}d\tilde{x}} - \frac{\int \phi_1(x,t)\phi_3^2(x,t)e^{-H(x)}d\tilde{x}}{\int e^{-H(x)}d\tilde{x}} \\
&= -\phi_1(x,t) - \frac{\phi_1(x,t)}{1+\phi_1^2(x,t)}
\end{aligned} \tag{2.3}$$

The evolution operators e^{tL} and e^{tQL} satisfy the Dyson formula:

$$e^{tL} = e^{tQL} + \int_0^t e^{(t-s)L}PLe^{sQL}ds. \tag{2.4}$$

Plug this to the second term in equation (2.2), we can get

$$e^{tL}QLx_j = e^{tQL}QLx_j + \int_0^t e^{(t-s)L}PLe^{sQL}QLx_j ds. \tag{2.5}$$

Let $F_j(x,t) = e^{tQL}QLx_j$, $K_j(\hat{x},t) = PLF_j(x,t)$, then equation (2.2) can be rewritten as:

$$\begin{aligned}
\frac{\partial}{\partial t}e^{tL}x_j &= e^{tL}PLx_j + \int_0^t e^{(t-s)L}K_j(\hat{x},s)ds + F_j(x,t) \\
&= e^{tL}PLx_j + \int_0^t K_j(\hat{\phi}(x,t-s),s)ds + F_j(x,t).
\end{aligned} \tag{2.6}$$

We are interested in finding:

$$\frac{\partial}{\partial t}Pe^{tL}x_j = Pe^{tL}PLx_j + P\int_0^t e^{(t-s)}K_j(\hat{x},s)ds. \tag{2.7}$$

The last term in equation (2.6) is dropped because $PF_j = 0$

The third term in equation (2.6) is the noise term, which depends on full knowledge of the initial condition x. [3] Note that $F_j(x,t)$ is the solution of the following orthogonal dynamic system:

$$\begin{aligned} \frac{\partial}{\partial t}F_j(x,t) &= QLF_j(x,t) = LF_j(x,t) - PLF_j(x,t) \\ F_j(x,0) &= QLx_j = R_j(x) - (PR_j)(\hat{x}). \end{aligned} \tag{2.8}$$

Formally, equation (2.8) is equivalent to:

$$F_j(x,t) = e^{tL}F_j(x,0) - \int_0^t e^{(t-s)L}PLF_j(x,s)ds. \tag{2.9}$$

Let $a_j^v(s) = (LF_j(\cdot,s), h^v)$, then $K_j(\hat{x},s)$ can be rewritten as:

$$K_j(\hat{x},s) = PLF_j(x,s) = \sum_{\nu\in I} a_j^\nu(s)h^\nu(\hat{x}). \tag{2.10}$$

As a result, the second term of $F_j(x,t)$ in equation (2.9) becomes:

$$\int_0^t e^{(t-s)L}PLF_j(x,s)ds = \int_0^t \sum_{\nu\in I} a_j^\nu(s)h^\nu(\hat{\phi}(x,t-s))ds.$$

Now, multiply both sides of equation (2.9) by L and take the inner product with h^μ, we will get:

$$(LF_j(t), h^\mu) = (Le^{tL}F_j(0), h^\mu) - \int_0^t \sum_{\nu\in I} a_j^v(s)(Le^{(t-s)L}h^v, h^\mu)ds.$$

If we denote $f_j^\mu(t) = (Le^{tL}F_j(0), h^\mu)$, $g^{\nu\mu}(t) = (Le^{tL}h^v, h^\mu)$, then the above equation can be rewritten as:

$$a_j^\mu(t) = f_j^\mu(t) - \int_0^t \sum_{\nu\in I} a_j^v(s)g^{\nu\mu}(t-s)ds. \tag{2.11}$$

If we denote $\gamma^{\nu\mu} = (e^{tL}h^\nu, h^\mu)$, then the memory term in equation (2.7) is given as:

$$P\int_0^t e^{(t-s)}PLF_j(x,s)ds = \int_0^t \sum_{\nu,\mu\in I} a_j^v(s)\gamma^{\nu\mu}(t-s)h^\mu(\hat{x})ds.$$

Now, we introduces matrices A, F, G, and Γ such that $A_{j\mu} = a_j^\mu$, $F_{j\mu} = f_j^\mu$, $G_{\nu\mu} = g^{\nu\mu}$, $\Gamma_{\nu\mu} = \gamma^{\nu\mu}$. Then, equation (2.7) can be represented using matrix multiplications:

$$\frac{\partial}{\partial t}Pe^{tL}x_j = Pe^{tL}PLx_j + \int_0^t A(s)\Gamma(t-s)h^\mu(\hat{x})ds \tag{2.12}$$

where

$$Pe^{tL}PLx_j = \sum_{\nu,\mu\in I}(R_j, h^\mu)(e^{tL}h^\mu, h^\nu)h^\nu(\hat{x}).$$

In the following chapters, we will discuss choices of basis functions and discuss how to estimate A, F, G, and Γ.

3 | Basis of L^2 Function

This chapter introduces two different bases of L^2 function. Namely, the adjusted Hermite Polynomial basis [3], and the data-driven basis introduced in [4]. These bases will be used to solve the Voltera equation in equation (2.11). Note that we will continue using equation (1.3) as our example. The variables will be modified to x_i's to keep consistent with the definitions in this chapter, that is:

$$\begin{aligned} \dot{x_1} &= x_2 \\ \dot{x_2} &= -x_1(1+x_3^2) \\ \dot{x_3} &= x_4 \\ \dot{x_4} &= -x_3(1+x_1^2) \end{aligned} \tag{3.1}$$

where we denoted the Hamiltonian equation as,

$$H(x) = \frac{1}{2}(x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1^2 x_3^2).$$

3.1 Hermite Polynomial Basis

First, we define our inner product in the following way:

$$(f,g) = \frac{1}{Z}\int_{R^4} f(x)g(x)e^{-H(x)}dx \tag{3.2}$$

In this section, we assume that if f and g only depend on $\hat{x} = (x_1, x_2)$, this inner product can be further simplified to:

$$\begin{aligned} (f,g) &= \frac{1}{Z}\int_{R^4} f(x)g(x)e^{-\frac{1}{2}(x_1^2+x_2^2+x_3^2+x_4^2+x_1^2x_3^2)}dx \\ &= \frac{2\pi}{Z}\int_{R^2} f(x_1,x_2)g(x_1,x_2)\frac{e^{-\frac{1}{2}x_1^2-\frac{1}{2}x_2^2}}{\sqrt{1+x_1^2}}dx_1dx_2. \end{aligned} \tag{3.3}$$

Now, the question is left to calculating Z. And this can be easily done by setting $f = g = 1$ and using Monte Carlo integration:

$$\begin{aligned} Z &= 2\pi\int_{R^2}\frac{e^{-\frac{1}{2}x_1^2-\frac{1}{2}x_2^2}}{\sqrt{1+x_1^2}}dx_1dx_2 \\ &\approx \frac{2\pi}{N}\sum_{k=1}^{N}\frac{e^{-\frac{1}{2}x_{1,k}^2-\frac{1}{2}x_{2,k}^2}}{\sqrt{1+x_{1,k}^2}}/q(\hat{x}) \end{aligned} \tag{3.4}$$

where $q(\hat{x})$ is the distribution function of x_1 and x_2. To obtain $q(\hat{x})$ from a sample, we can compute the multivariate kernel density estimate of $x_{1,k}$ and $x_{2,k}$.

Now, we can redefine the inner product as:

$$\begin{aligned}(f,g) &= \int_{R^2} f(x_1,x_2)g(x_1,x_2)w(x_1,x_2)dx_1dx_2 \\ &\approx \frac{1}{N}\sum_{k=1}^{N} f(\hat{x_k})g(\hat{x_k})\hat{w}(\hat{x_k})\end{aligned} \tag{3.5}$$

where

$$\begin{aligned}w(\hat{x}) &= \frac{e^{-H(x)}}{Z} \\ \hat{w}(\hat{x}) &= \frac{w(\hat{x})}{q(\hat{x})}.\end{aligned}$$

Since

$$\begin{aligned}\int e^{-\frac{x_4^2}{2}}dx_4 &= \sqrt{2\pi} \\ \int e^{-\frac{(1+x_1^2)x_3^2}{2}}dx_3 &= \sqrt{\frac{2\pi}{1+x_1^2}}\end{aligned} \tag{3.6}$$

we can simplify $w(\hat{x})$ to:

$$w(\hat{x}) = \frac{2\pi}{Z}\frac{e^{-\frac{1}{2}x_1^2-\frac{1}{2}x_2^2}}{\sqrt{1+x_1^2}}.$$

Now, we want to introduce the Hermite polynomials $H_k(x)$, which are defined through the following recursion relation:

$$H_0(x) = 1, \quad H_1(x) = x, \quad H_k(x) = \frac{1}{\sqrt{k}}xH_{k-1}(x) - \sqrt{\frac{k-1}{k}}H_{k-2}(x).$$

The Hermite polynomials generated from the recursion relation above are orthonormal with respect to $\frac{1}{\sqrt{2\pi}}e^{-x^2/2}$. Therefore,

$$\int H_k(x)H_l(x)\frac{e^{-x^2/2}}{\sqrt{2\pi}} = \delta_{kl}. \tag{3.7}$$

Based on the Hermite polynomials, we then define

$$\tilde{H}_k(x) = (1+2\alpha)^{1/4}H_k(\sqrt{1+2\alpha}x)e^{-\alpha x^2/2}.$$

If we let $x = \sqrt{1+2\alpha}\,y$ and plug it into equation (3.7), then

$$\begin{aligned}\delta_{kl} &= \int H_k(\sqrt{1+2\alpha}\,y)H_l(\sqrt{1+2\alpha}\,y)\frac{e^{-(1+2\alpha)y^2/2}}{\sqrt{2\pi}}\sqrt{1+2\alpha}\,dy \\ &= \int (1+2\alpha)^{1/4}H_k(\sqrt{1+2\alpha}\,y)e^{-\alpha y^2/2}(1+2\alpha)^{1/4}H_l(\sqrt{1+2\alpha}\,y)e^{-\alpha y^2/2}\frac{e^{-y^2/2}}{\sqrt{2\pi}}dy \\ &= \int \tilde{H}_k(y)\tilde{H}_l(y)\frac{e^{-y^2/2}}{\sqrt{2\pi}}dy.\end{aligned} \tag{3.8}$$

Therefore, $\tilde{H}_k$ also has the orthogonal property.

Now, we let

$$h^{\mu}(\hat{x}) = \tilde{H}_{\mu_1}(x_1)\tilde{H}_{\mu_2}(x_2)(1+x_1^2)^{\frac{1}{4}}$$

To get an orthonormal basis, we need to introduce the following constant:

$$\begin{aligned} c^{\mu} &= \int_{R^2} (h^{\mu}(\hat{x}))^2 w(\hat{x}) dx_1 dx_2 \\ &\approx \frac{1}{N}\sum_{k=1}^{N} (h^{\mu}(\hat{x_k}))^2 \hat{w}(\hat{x_k}). \end{aligned}$$

For our example, c^{μ} is:

$$\begin{aligned} c^{\mu} &= \int_{R^2} (h^{\mu}(\hat{x}))^2 w(\hat{x}) dx_1 dx_2 \\ &= \int\int \tilde{H}^2_{\mu_1}(x_1)\tilde{H}^2_{\mu_2}(x_2)(1+x_1^2)^{\frac{1}{2}} w(\hat{x}) dx_1 dx_2 \\ &= \frac{2\pi}{Z}\int\int \tilde{H}^2_{\mu_1}(x_1)\tilde{H}^2_{\mu_2}(x_2)(1+x_1^2)^{\frac{1}{2}} \frac{e^{-\frac{1}{2}x_1^2-\frac{1}{2}x_2^2}}{\sqrt{1+x_1^2}} dx_1 dx_2 \\ &= \frac{2\pi}{Z}\int\int \tilde{H}^2_{\mu_1}(x_1)\tilde{H}^2_{\mu_2}(x_2) e^{-\frac{1}{2}x_1^2-\frac{1}{2}x_2^2} dx_1 dx_2 \\ &= \frac{2\pi}{Z}\int \tilde{H}^2_{\mu_1}(x_1) e^{-\frac{1}{2}x_1^2} dx_1 \int \tilde{H}^2_{\mu_2}(x_2) e^{-\frac{1}{2}x_2^2} dx_2 \\ &= \frac{4\pi^2}{Z}. \end{aligned} \tag{3.9}$$

As a result, the final Hermite Polynomial basis is:

$$h^{\mu}(\hat{x}) = \frac{\tilde{H}(\mu_1, x_1)\tilde{H}(\mu_2, x_2)(1+x_1^2)^{\frac{1}{4}}}{\sqrt{c_{\mu}}}. \tag{3.10}$$

Note that this basis is used only when the functions are based on $\hat{x}$. However, some of the functions depend on x in the Mori-Zwanzig process. Therefore, we need to compute the weight carefully.

Recall that our inner product is defined in equation (3.2). Since we assumed f and g only depend on $\hat{x}$ in previous section, we simplified it to equation (3.5). However, this simplification does not work if f and g depend on x. Now, we need to compute the basis with respect to $e^{-H(x)}$. In fact, the general steps are the same as previous section. The only difference is the weight used in Monte Carlo integration.

First, we let

$$\begin{aligned} v(x_i) &= \frac{1}{Z} e^{-\frac{1}{2}(x_1^2+x_2^2+x_3^2+x_4^2+x_1^2x_3^2)} \\ \hat{v}(x_i) &= v(x_i)/q(x) \end{aligned}$$

where Z is defined in equation (3.4) and $x_k = (x_{1,k}, x_{2,k}, x_{3,k}, x_{4,k}) \sim q(x)$.

Then, the inner product estimation can be expressed as:

$$\begin{aligned}(f,g) &= \int_{R^4} f(x)g(x)v(x)dx \\ &\approx \frac{1}{N}\sum_{k=1}^{N} f(x_k)g(x_k)\hat{v}(x_k).\end{aligned} \tag{3.11}$$

Given this new weight, the standardized constant c^ν will also be influenced:

$$\begin{aligned}c^\nu &= \int_{R^2} (h^\nu(\hat{x}))^2 v(\hat{x}) \\ &\approx \frac{1}{N}\sum_{k=1}^{N} (h^\nu(\hat{x_k}))^2 \hat{v}(\hat{x_k}).\end{aligned}$$

With this standardized constant, our final basis with respect to $e^{-H(x)}$ will be:

$$h^\nu(\hat{x}) = \frac{\tilde{H}(\nu_1, x_1)\tilde{H}(\nu_2, x_2)(1+x_1^2)^{\frac{1}{4}}}{\sqrt{c^\nu}}. \tag{3.12}$$

3.2 Data-driven Basis

In the previous sections, we introduced the analytic bases given weight v or w. However, both cases need the information about the distribution function, which is not always explicitly available. Therefore, we will consider another basis that does not need this information.

In particular, we will consider basis functions obtained from the diffusion maps algorithm [4]. In this algorithm, the basis functions are obtained from solving an eigenvalue problem of a symmetric negative definite operation that takes functions defined on a smooth manifold $M \subseteq \mathbb{R}^n$ where the data lie on or close to.

Basically, given $x_i \sim q(x)$ with sampling density q, the diffusion maps algorithm construct a matrix L_ϵ that approximates

$$L_\epsilon f(x) = q^{-1} div(q \nabla f(x)) + O(\epsilon)$$

at each point $x \in M$ in probability. A discrete approximation to the eigenfunctions $\varphi(x)$ can be obtained by solving the eigenvalue problem:

$$L_\epsilon \vec{\varphi_k} = \lambda_k \vec{\varphi_k}$$

where $\vec{\varphi_k} \in \mathbb{R}^n$ and the ith component $(\vec{\varphi_k})_i \approx \varphi_k(x_i)$.

In our application, given $x_i \sim q(x)$, where $q(x)$ is the distribution of the initial conditions, we compute these discrete approximates to their basis functions. Numerically, we use Hamiltonian Monte-Carlo to sample $x_i \sim q$ where $q(x) \propto e^{-H(x)}$, where $H(x)$ is known in our example. In general, we may be given the examples x_i rather than $H(x)$. We should also mention that we need to evaluate $\vec{\varphi_k}$ at a new point $y \neq x_i$. We will dealt with the estimation of $\varphi_k(y)$ using a Nystrom extension, as described in [5].

Recall that we can compute the conditional expectation analytically as:

$$E[\phi_j(x,t)|\hat{x}] = \sum_{\nu} (\phi_j(t), h^{\nu}) h^{\nu}(\hat{x}). \tag{3.13}$$

We can compare the expectation based on different bases:

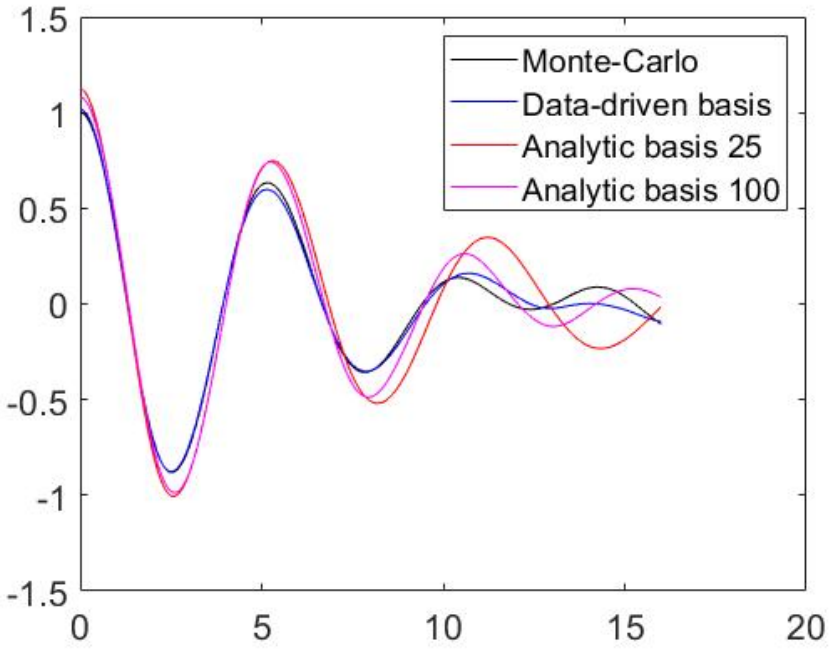

Figure 3.1: This figure computes $E[\phi_j(x,t)|\hat{x}]$ based on different bases. The black line is the real solution estimated by Monte-Carlo method; the blue line represent the conditional expectation computed based on 100 data-driven basis; the red line represent the expectation computed based on 25 analytic basis; the magenta line represent the expectation computed based on 100 analytic basis.

As we can see from figure 3.1, the expectation based on 100 data-driven basis functions is close to the real solution, while the expectation based on 25 and 100 analytic basis functions are slightly off the real solution.

4 | Numerical Methods

In this chapter, we will discuss the numerical methods for Mori-Zwanzig Formalism. We start by introducing the numerical methods to solve the matrices A, F, G, and Γ for our example. Then, we will present the results using different basis.

Our first step is to find matrix F and G, where the elements are given by the following equations:

$$\begin{aligned} f_j^\mu(t) &= (Le^{tL}F_j(\cdot,0), h^\mu) \\ g^{\nu\mu}(t) &= (Le^{tL}h^v, h^\mu). \end{aligned} \tag{4.1}$$

From the equations above, we noticed that both of them have the form $(L..., h^\mu)$. Although we can solve these inner products in their current forms, it might be time consuming since we need to compute $Le^{tL}F_j(0)$ and $Le^{tL}h^v$ separately and store them in two variables as we write the code. A simpler solution is to first transform these two inner products in the following way:

$$\begin{aligned} f_j^\mu(t) &= (Le^{tL}F_j(\cdot,0), h^\mu) = -(e^{tL}F_j(\cdot,0), Lh^\mu) \\ g^{\nu\mu}(t) &= (Le^{tL}h^v, h^\mu) = -(e^{tL}h^v, Lh^\mu). \end{aligned} \tag{4.2}$$

By doing so, we only need to compute Lh^μ once and use it to compute both of the matrices.
To compute the integrals,we will use Monte-Carlo methods. Therefore, G can be calculated in the following way:

$$\begin{aligned} g^{\nu\mu}(t) &= (Le^{tL}h^\nu, h^\mu) \\ &= -(e^{tL}h^\nu, Lh^\mu) \\ &= -\int e^{tL}h^\nu(\hat{x})Lh^\mu(\hat{x})v(x)dx \\ &= -\int h^\nu(\hat{\phi}(x,t))R\nabla_x h^\mu(\hat{x})v(x)dx \\ &= -\int h^\nu(\hat{\phi}(x,t))(R_1\frac{dh^\mu}{dx_1} + R_2\frac{dh^\mu}{dx_2})v(x)dx \\ &\approx -\frac{1}{N}\sum_{i=1}^{N} h^\nu(\hat{\phi}(x_i,t))(R_1\frac{dh^\mu(\hat{x_i})}{dx_1} + R_2\frac{dh^\mu(\hat{x_i})}{dx_2})\hat{v}(x_i). \end{aligned} \tag{4.3}$$

Similarly, F can be calculated in the following way:

$$\begin{aligned} f_j^\mu(t) &= (Le^{tL}F_j(\cdot,0), h^\mu) \\ &= -(e^{tL}F_j(\cdot,0), Lh^\mu) \\ &= -\int e^{tL}F_j(\cdot,0)Lh^\mu(\hat{x})v(x)dx \\ &\approx -\frac{1}{N}\sum_{i=1}^{N} F_j(\phi(x_i,t),0)(R_1\frac{dh^v(\hat{x_i})}{dx_1} + R_2\frac{dh^v(\hat{x_i})}{dx_2})\hat{v}(x_i). \end{aligned} \tag{4.4}$$

Given F and G, we can compute A using the composite trapezoidal rule. Then, equation (2.11) becomes:

$$a_j^\mu(n\Delta t) = f_j^\mu(n\Delta t) - \sum_{i=0}^{n}\sum_{v\in I} a_j^v(i\Delta t)g^{v\mu}((n-i)\Delta t)\Delta t\hat{w}i. \tag{4.5}$$

Using matrix notation, we have:

$$\begin{aligned} &A(n\Delta t) = F(n\Delta t) - (G'(n\Delta t)A(0)\Delta t + ... + G'(0)A(n)\Delta t) \\ &A(n\Delta t) + \Delta tG'(0)A(n\Delta t) = F(n\Delta t) - \Delta t(G'(n\Delta t)A(0) + ... + G'(\Delta t)A((n-1)\Delta t)) \\ &A(n\Delta t) = (I + \Delta tG_0)^{-1}[F(n\Delta t) - \Delta t(G'(n\Delta t)A(0) + ... + G'(\Delta t)A((n-1)\Delta t))]. \end{aligned} \tag{4.6}$$

Finally, Γ can be computed as:

$$\begin{aligned} \gamma^{\nu\mu}(t) &= (e^{tL}h^\nu, h^\mu) \\ &= \int e^{tL}h^\nu(\hat{x})h^\mu(\hat{x})v(x)dx \\ &= \int h^\nu(\hat{\phi}(x,t))h^\mu(\hat{x})v(x)dx \\ &\approx \frac{1}{N}\sum_{i=1}^{N} h^\nu(\hat{\phi}(x_i,t))h^\mu(\hat{x_i})\hat{v}(x_i). \end{aligned} \tag{4.7}$$

Our final step is to estimate the solution of the system by solving:

$$\begin{aligned} \frac{\partial}{\partial t}\Phi(t) &= e^{tL}PLx_j + \int_0^t A(s)\Gamma(t-s)h(\hat{x})ds \\ &= \sum_\mu (R_j, h^\mu)h^\mu(e^{tL}\hat{x}) + \int_0^t A(s)\Gamma(t-s)h(\hat{x})ds, \quad \Phi(0) = \hat{x}. \end{aligned} \tag{4.8}$$

To estimate equation (4.8), we can use Runge-Kutta 4 method to solve the ODE system and trapezoidal rule to solve the integrals.

For trapezoidal rule, suppose we want to solve $\int_a^b f(x)dx$, we have the following approximation:

$$\begin{aligned} \int_a^b f(x)dx &\approx \sum_{k=1}^{n} \frac{f(x_{k-1}) + f(k)}{2}\Delta t \\ &= \frac{\Delta t}{2}(f(x_0) + 2f(x_1) + 2f(x_2) + ... + 2f(x_{n-1}) + f(x_n)). \end{aligned}$$

Numerically, we set $\Delta t = 0.01$ for our example.

For Runge-Kutta 4 method,to solve $\frac{dx}{dt} = f(x) + \int_0^t g(s)ds$, we can follow the following procedures:

1. Let τ be the time step

2. Compute

$$\begin{aligned}
k_1 &= f(x, i\tau) + \int_0^{i\tau} g(s)ds \\
&\approx f(x, i\tau) + \sum_{k=0}^{i} g(k\tau)w_k \\
k_2 &= f(x + \frac{k_1}{2}, (i + \frac{1}{2})\tau) + \int_0^{(i+\frac{1}{2})\tau} g(s)ds \\
&\approx f(x + \frac{k_1}{2}, (i + \frac{1}{2})\tau) + \sum_{k=0}^{(i+\frac{1}{2})} g(k\tau)w_k \\
k_3 &= f(x + \frac{k_2}{2}h, (i + \frac{1}{2})\tau) + \int_0^{(i+\frac{1}{2})\tau} g(s)ds \\
&\approx f(x + \frac{k_2}{2}h, (i + \frac{1}{2})\tau) + \sum_{k=0}^{(i+\frac{1}{2})} g(k\tau)w_k \\
k_4 &= f(x + k_3, (i + 1)\tau) + \int_0^{(i+1)\tau} g(s)ds \\
&\approx f(x + k_3, (i + 1)\tau) + \sum_{k=0}^{i+1} g(k\tau)w_k \\
i &= 0, 2, ..., n - 1
\end{aligned} \tag{4.9}$$

   where

$$w_k = \begin{cases} \frac{\tau}{2} & \text{, } k \text{ is the end point} \\ \tau & \text{, otherwise} \end{cases}$$

3. Estimate $f(x, (i + 1)\tau)$ using:

$$f(x, (i + 1)\tau) = f(x, \tau) + \frac{\tau}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

Since Runge-Kutta 4 method needs to compute integrals at $\frac{1}{2}\tau$, we set $\tau = 2\Delta t = 0.02$.

The results for our example are shown in the following plots:

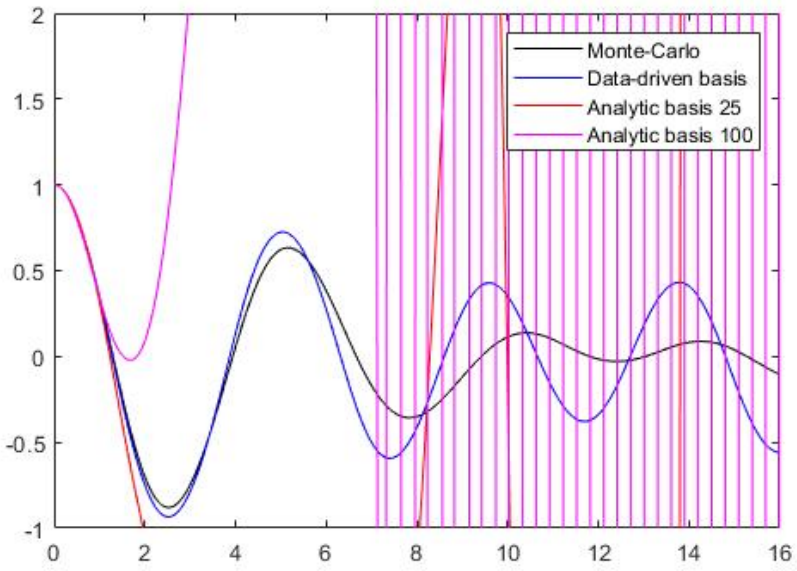

Figure 4.1: This figure shows the Mori-Zwanzig prediction of $\Phi_1(t)$ using analytic basis. The black line represents the solution of true solution estimated by Monte-Carlo integration. The blue line shows the Mori-Zwanzig prediction of $\Phi_1(t)$ using 100 data driven bases.The red line represents the solution by solving equation (4.8) using 25 analytic bases. The magenta line represents the solution using 100 analytic bases.

As we can see from the results, Mori-Zwanzig Formalism is very sensitive to the choice of basis functions. The results using both analytic bases is good in a very short period and start losing information as time increases. On the other hand, the data driven basis does provide a much better approximation. The trade off for this better estimation is the huge amount of time used to generate the data driven basis.

5 | Conclusion

In this paper, we applied several basis functions to solve reduced dynamics based on Mori-Zwanzig Formalism. We defined Hamiltonian systems and presented an example, which is used to test the Mori-Zwanzig procedure. We discussed two bases that can be used in the example. We also discussed Liouville equation and its application in Mori-Zwanzig Formalism. Lastly, we discussed Mori-Zwanzig Formalism both analytically and numerically. Based on the final results, we found that Mori-Zwanzig is very sensitive to the basis functions. In order to get a good result, we need to be careful about the basis choices.

Moreover, all of our conclusions are drawn from the Hamiltonian example. The feasibility and performance of this method on other systems are not checked at this point. Therefore, we want to test Mori-Zwanzig formalism on other problems in the future. Another important task is to find a way to get more precise estimations with relatively short running time. Our current method does not have a good balance between precision and running time. Further investigation is necessary to overcome this problem.

Bibliography

[1] Alexandre J. Chorin and Ole H Hald. *Stochastic Tools in Mathematics and Science*. Springer New York, 2013.

[2] S. Dimova and Ts. Bazlyankov. Numerical methods for hamiltonian systems: Implementation and comparison. AIP Publishing LLC, 2015.

[3] Alexandre J. Chorin, Ole H. Hald, and Raz Kupferman. Optimal prediction with memory. *Physica D: Nonlinear Phenomena*, 166(3):239 – 257, 2002.

[4] John Harlim. *Data-Driven Computational Methods: Parameter and Operator Estimations*. Cambridge University Press, 2018.

[5] John Harlim and Haizhao Yang. Diffusion forecasting model with basis functions from qr-decomposition. *Journal of Nonlinear Science*, 28(3):847–872, Jun 2018.

Academic Vitae

Zhenyu Tian
tianzhenyv@gmail.com

EDUCATION

PENNSYLVANIA STATE UNIVERSITY **University Park, PA**
B.S. in Applied and Industrial Mathematics; B.S. in Applied Statistics; Minor in Information Science and Technology
May 2019

- Schreyer Honors College
- Eberly College of Science, Mathematics; Eberly College of Science, Statistics
- Dean's List every semester
- Specialized course work includes mathematical modeling, calculus, ordinary differential equations, partial differential equations, linear algebra, abstract algebra, classical analysis, numerical analysis, python, databases, probability, coding theory, stochastic modeling, ANOVA, and regression courses

EXPERIENCE

Undergraduate Honors Thesis **University Park, PA**
Supervisor: Dr. John Harlim, Professor of Mathematics at the Pennsylvania University *August 2018-April 2019*

- Researched on the Mori-Zwanzig formalism
- Learned Hermite polynomials and Hamiltonian systems

Applied Mathematics REU **University Park, PA**
Participant *May 2018- June 2018*

- Researched on the win rate of video game DOTA2
- Learned Multidimensional Scaling and simulation in MATLAB

Mathematics Advanced Study Semesters **University Park, PA**
Part time Participant *August 2017- December 2017*

- Attended program for advanced mathematical studies
- Participated in colloquiums with mathematicians
- Conducted research on Fabricius-Berre Theorem

COMPUTER SKILLS

- Proficiency in Microsoft Office Suite
- Learned Python, java, MATLAB, R, SAS, Minitab, LaTeX, SQL

ADDITIONAL SKILLS

- Mathematical models: regressions, Markov chains, linear compartment models, dimensional analysis, epidemic models, predator-prey models, spatial models, cellular automata, etc.
- Bilingual in Mandarin Chinese and English

ACHIEVEMENTS

Bayard D. and Ethel M. Kunkle Scholarship *Fall 2018*
Mary Lister McCammon Award *April 2018*

- Outstanding achievement in mathematics during the 2017-2018 academic year

Matthew Rosenshine Fund *March 2018*

- Excellence in statistics for the spring semester of the 2017-2018 academic year