ROBUST NON-LINEAR LYAPUNOV DEEP LEARNING CONTROL DESIGN FOR CHAOTIC SYSTEMS

by

AMR SALAH MAHMOUD

A dissertation submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY IN ELECTRICAL AND COMPUTER ENGINEERING

2022

Oakland University
Rochester, Michigan

Doctoral Advisory Committee:

Mohamed A. Zohdy, Ph.D., Chair
Brian Dean, Ph.D.
Darrell Schmidt, Ph.D.
Richard Olawoyin, Ph.D.

الحمدّ والشكر لك ربي لما وفقتني إليه، والحمد لك جلّ جلالك

I dedicate the final product of this dissertation to my parents who have supported and inspired me throughout this journey.

ACKNOWLEDGMENTS

I would like to express my special thanks and gratitude to my advisor and mentor Dr. Mohamed Zohdy (Coach) as for without him the work presented in this dissertation wouldn’t have been possible without the mentoring, guidance, support, and encouragement. Besides my advisor I would like to express my deep and sincere gratitude to the members of the advisory committee Dr. Richard Olawoyin, Dr. Brian Dean and Dr. Darrell Schmid for their time and valuable input.

I would also like to thank my fellow lab mates and teaching assistant colleagues who through their discussions I was inspired to bring the ideas in this dissertation forth. Finally, I would like to thank my parents and sister for without their love, encouragement, support, and inspiration to begin and continue this journey this wouldn’t have been possible.

Amr Salah Mahmoud

ABSTRACT

ROBUST NON-LINEAR LYAPUNOV DEEP LEARNING CONTROL DESIGN FOR CHAOTIC SYSTEMS

by

AMR SALAH MAHMOUD

Adviser: Mohamed Zohdy, Ph.D.

Despite their operational success, machine learning controllers lack theoretical guarantees in terms of system stability. In contrast, classic model-based controller design uses principled approaches such as Linear Quadratic Regulator (LQR) to synthesize stable controllers with verifiable proofs. In addition, deep learning controllers encounter feedback timing bottlenecks that increase exponentially with the system complexity. Deep learning is also dependent on the quality and diversity of the dataset to produce unbiased findings; therefore, the prediction of deep learning is not guaranteed. As a result, in this research, we develop and implement a guaranteed stability solution for safety critical and chaotic systems through the integration of Lyapunov Stability theory and deep machine learning. Three control methods are researched, leading to the development of the Deep Lyapunov-stable controller: the deep learning methodology, the Lyapunov control function, and controller parameters. In this research, we provide a generic method for synthesizing a Deep Lyapunov-stable control and a way to simultaneously confirm its stability. A unique Lyapunov control function is devised and shown to be effective in managing Duffing, Van der Pol, and Zohdy-Harb nonlinear

systems, but with restrictions on the system's oscillation frequency, initial conditions and disturbances. Subsequently, Dynamic Lyapunov Deep Learning is introduced to alleviate the Lyapunov control's shortcomings. Developing a deep learning architecture in combination with a customized Lyapunov control resolves the temporal delay and Lyapunov parameters calibration concern. Different datasets are also presented before establishing the one with the best accuracy. In addition to the dataset, the architecture of the deep learning model has a significant effect on the model's accuracy. A process for relearning is intended to accommodate the introduction of new system dynamics. Based on the correlation study, we also designed an optimization technique to improve the integration of the deep learning layer and controller layer. The proposed integration of Deep Learning and Lyapunov Control, referred to as Lyapunov Deep Learning (LDL) control, is applied in MATLAB / SIMULINK to the magnetic levitation chaotic non-linear system to demonstrate its effectiveness in addressing sudden changes in system behavior, the environment, and demands in comparison to other methods of control.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

In this research, we focus on advancing the field of nonlinear controllers. Over the past decade, researchers have concentrated on the linearization of nonlinear systems and the use of linear controllers such as PID with the hope that it will perform well enough for pole placement, control, root-locus, or other linear techniques to work. This might be attributed to the ease of implementation and the low time requirement of linear control. Controllers that require linearization, such as PID or LQR techniques lose some of the best properties of the system while linearizing. On the other hand, nonlinear controllers overcome this disadvantage by using the nonlinear model. New technologies have been increasing in complexity in an exponential manner in response to the market's overwhelming need for greater functionality, performance, and bandwidth there, for as much functionality as possible is integrated in new designs. As such, linear controllers will no longer be able to provide the desired outcome as complexities increase and demand for efficiency and lossless design increase.

1.1 Fundamentals of Complex Non-Linear Systems

In order to understand the best method to control a system, we should begin by understanding what modern complex nonlinear systems require. While it is simple to define linear functions, the term nonlinear encompasses anything else. As Stanislaw Ulam, a famous scientist in the field of mathematics and nuclear physics once explained, “Using a term like nonlinear science is like referring to the bulk of zoology as the study of non-elephant animals."[1]. As Figure 1 shows, most Dynamic Systems are non-linear in nature and most Nonlinear systems are Chaotic in nature.

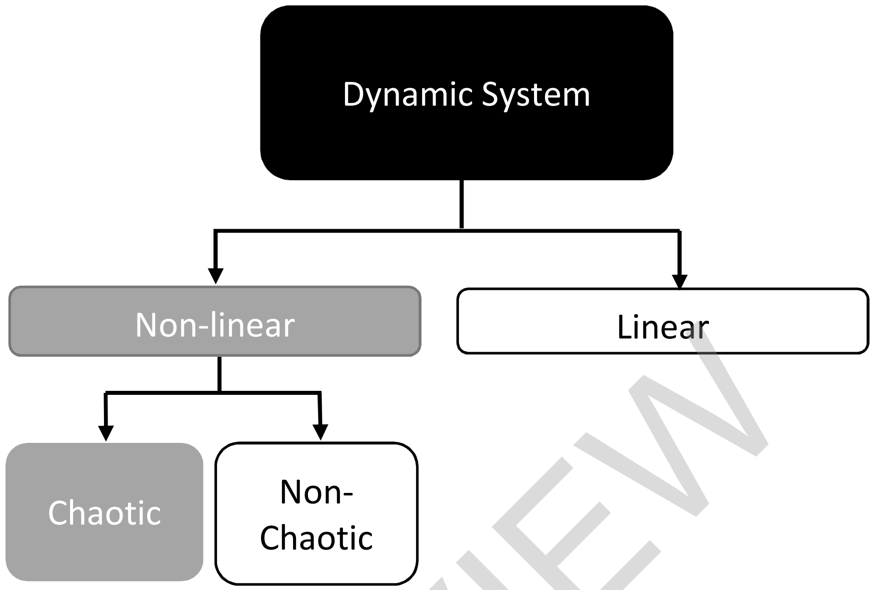

Figure 1 Relationship between Dynamic Systems, Nonlinear and Chaotic systems

Nonlinearity frequently emerges from the collective behavior of even the simplest systems, it is insufficient to combine the effects of the components merely linearly. Emergent phenomena, including chaos, solitons, fractals, and meta/multi-stability, are produced by the interactions between the components. Even if the underlying physics is deterministic, the ensuing dynamics can be very unpredictable and result in the formation of non-equilibrium patterns.

Methods of solution or analysis for problems involving nonlinear differential equations are situation specific. Lotka–Volterra, Navier–Stokes, Duffing and Van Der Pol equations are examples of nonlinear differential equations. One of the most challenging aspects of nonlinear issues is that it is typically impossible to combine existing solutions to create new ones. A family of linearly independent solutions may be utilized through the superposition principle to derive generic solutions for linear problems, for instance. This is shown by one-dimensional heat transfer with Dirichlet boundary conditions, whose solution is represented as a time-dependent linear combination of sinusoids with varying frequencies; this makes solutions extremely versatile. Identifying several exact solutions to nonlinear equations is feasible, but the absence of a superposition principle precludes the creation of additional solutions. A further distinction between linear and nonlinear systems is that nonlinear dynamics can only be solved by using computers and simulating the dynamics. Nonlinear systems dynamics are several, but we present a few examples below, some of which will be utilized later in this research.

1.1.1 Duffing System

For the purpose of simulating damped and driven oscillators, a nonlinear second-order differential equation known as the Duffing equation is used. The Duffing system is named after Georg Duffing (1861–1944) [2]. In addition to being an example of a highly complex chaotic system, the frequency response of the Duffing system also exhibits the phenomena of jump resonance, which is a form of frequency hysteresis behavior. The Duffing equation describes the nonlinear oscillations of a mass connected to a nonlinear spring and a linear damper. Duffing Dynamic Differential Equation is presented in 1.

$$\ddot{x} + \varphi\dot{x} + \delta x + \gamma x^3 = cos\, t + u \tag{1}$$

1.1.2 Van der pol System

The Van der Pol oscillator was devised by Balthasar van der Pol, a Dutch electrical engineer and scientist at Philips. Van der Pol discovered stable oscillations, which he subsequently termed relaxation oscillations. In addition, Van der Pol and his colleague, van der Mark, reported in the September 1927 edition of Nature that, at particular driving frequencies, an irregular noise could be heard, which was subsequently determined to be the outcome of deterministic chaos. Van der pol Dynamic Differential Equation is presented in 2.

$$\ddot{x} + \varphi(1 - x^2)\dot{x} + x = \cos t + u \quad (2)$$

1.1.3 Navier–Stokes Equations in Fluid Dynamics

The motion of viscous fluid substances may be understood via the use of a set of partial differential equations known as the Navier–Stokes equations. Claude-Louis Navier, a French engineer and scientist, and George Gabriel Stokes, a mathematician, both contributed to the naming of this phenomenon. Over the course of many decades, beginning in 1822 and continuing through 1842–1850, the ideas were gradually evolved [2, 3]. For Newtonian fluids, the Navier–Stokes equations quantitatively express momentum and mass conservation. Occasionally, they are accompanied with a state equation that links pressure, temperature, and density. The Navier–Stokes equations are valuable because they describe the physics of numerous phenomena that are of importance to science and engineering. They can be utilized to represent weather, ocean currents, water flow in a pipe, and air flow around a wing. The full and simplified Navier–Stokes equations aid in the design of aircraft and automobiles, the study of blood flow, the design of power plants, and the analysis of pollution, among several other applications. They can be used to research and model magnetohydrodynamics when combined with Maxwell's equations.

1.1.4 Lotka–Volterra Equations

In 1910, Alfred J. Lotka was the first person to describe the Lotka–Volterra predator–prey model as a component of the concept of autocatalytic chemical processes. This model was developed by Lotka and Volterra [4]. It is common practice to utilize first-order nonlinear differential equations when attempting to describe the dynamics of biological systems involving the interaction of two species, one of which acts as a predator while the other acts as prey. These equations are sometimes referred to as the predator–prey equations since they are generally recognized by that name. When attempting to represent the dynamics of natural populations of predators and prey, many models, including the Lotka–Volterra model and the Rosenzweig–MacArthur model, have been used.

Concerning the reliability of models that are dependent on prey or ratios, there has been a great deal of disagreement. It is generally accepted that Richard Goodwin carried out the first application of the Lotka–Volterra Equations 3 in either 1965 or 1967 [4]. In the hypothetical system, predators thrive so long as there is an ample supply of prey, but they run out of their food supply and finally die out. The number of animals that are hunted will eventually increase since there will be fewer predators. These activities continue in a cycle of population growth followed by population decrease.

$$\begin{aligned} \frac{dx}{dt} &= \alpha x - \beta xy \\ \frac{dy}{dt} &= \delta xy - \gamma y \end{aligned} \tag{3}$$

1.1.5 Lorenz Chaotic System

Edward Lorenz, a mathematician, and meteorologist initially explored the Lorenz system, a set of ordinary differential equations. The model is known for having chaotic solutions for parameter values and beginning circumstances. An accumulation of chaotic solutions to the Lorenz system is what is known as the Lorenz attractor. The "butterfly effect" originates from the real-world implications of the Lorenz attractor, which state that in a chaotic physical system, in the absence of perfect knowledge of the initial conditions, even a disturbance in the air caused by a butterfly flapping its wings, our ability to predict its future course will always fail[5, 6]. This idea has made its way into popular culture, where it is used to describe the inability to accurately forecast the behavior of a system. This illustrates that physical systems may be completely predictable while yet keeping the unpredictability that is fundamental to their nature. When plotted in phase space, the shape of the Lorenz attractor itself looks like a butterfly. This resemblance is most apparent when looking at the attractor in its entirety. In its current form, the model may be represented by a set of three ordinary differential equations that are collectively referred to as the Lorenz equations.

1.2 An Overview of Nonlinear Control

Two frequent characteristics of novel control challenges are that the system's attractive operating range is not always close to equilibrium, necessitating explicit consideration of nonlinear effects in order to build a good controller. Even though physical modeling enables the precise identification of well-defined nonlinear systems the controller must contend with a large degree of uncertainty, owing mostly to due to a lack of familiarity with the system's specifications and an inability to measure the status of the entire system. This issue demonstrates the critical requirement for the development of controller tools that take on unpredictable nonlinear system behavior. When considered conceptually, they may be generally categorized into Analytically and computationally oriented. An analytical model of the system, and controller design is the result of a methodical procedure that ensures a desired behavior. Because stability is a necessary but not sufficient requirement for this technique, it is commonly referred to as robust stabilization. It encompasses Lyapunov-based approaches, gain-assignment methods, and conventional robust and adaptive tools. On the other hand, computationally focused approaches do not require an analytical model and may be built based on a numerical model of the system to be controlled—for example, produced by the collection of vast quantities of data to approximate its behavior. The most visible examples of this school include neural network-based control, fuzzy control, and intelligent control. Recently, a second class of computationally focused methodologies has gained prominence, which is based on analytical models of the system. To attempt to replicate the evolution of linear systems. To account for nonlinear effects in theory, piecewise linear models are offered. Typically, an optimal control objective is defined, and the controller design challenge is to demonstrate that the optimization is possible for the given numerical values of the system model, e.g., that it can be translated into linear matrix inequalities and a control signal can be numerically produced. Two disadvantages exist with the optimum control strategy. To begin, the solutions are vulnerable to plant uncertainty, such as a lack of complete state measurement and parametric uncertainty, which are prevalent concerns in the majority, if not all, actual applications. Second, calculation of the optimal control law is only achievable for low-dimensional systems, casting doubt on the method's application to nonlinear systems. Additionally, there is not necessarily a compelling rationale, other than mathematical convenience, for expressing the intended behavior of a dynamical system in terms of an optimization scalar criteria. While computationally oriented techniques benefit from rapidly developing computer technology, they focus on providing answers to specific issues rather than on explaining why, how, and when these solutions work. Therefor in this research we aim to comprehend the underlying process by which the system operates. The information is contained in the dynamics of the nonlinear system and disclosed by a thorough nonlinear analysis.

1.2.1 Gain Scheduling

Gain scheduling is a typical method for regulating nonlinear systems whose dynamics vary across operating conditions. Gain scheduling is utilized when a single set of controller gains does not offer the necessary performance and stability throughout the whole range of plant operating circumstances.

1.2.2 Adaptive Control

Adaptive control is an active field in the design of control systems to account for uncertainty. The major distinction between adaptive controllers and linear controllers is the adaptive controller's capacity to change itself to deal with unforeseen model uncertainties. Direct and indirect adaptive control are the two primary classifications. Indirect approaches estimate the plant's parameters and then utilize the predicted model data to calibrate the controller. Direct techniques are those in which the estimated parameters are utilized directly by the adaptive controller.

1.2.3 Model Predictive Control

MPC models anticipate the change in the system's dependent variables that will result from changes in the independent variables. The setpoints of regulatory PID controllers (pressure, flow, temperature, etc.) or the final control element are often controller-adjustable independent variables (valves, dampers, etc.). We make use of independent variables that are not subject to the influence of the controller here in the role of disturbances. The dependent variables in these processes are additional measurements that either represent control goals or process constraints.

Model predictive control may be broken down into many subtypes, one of which is known as nonlinear model predictive control, or NMPC for short. NMPC makes use of nonlinear system models for prediction. The iterative solution of optimal control problems with a limited prediction horizon is required in NMPC, just as it is in linear MPC. In linear MPC, these problems have a convex solution, however in nonlinear MPC, the convexity of these problems is not guaranteed. Both the theoretical framework of NMPC stability and the numerical solution face challenges as a result of this[7].

Typically, the numerical solution of NMPC optimum control problems is based on direct optimal control techniques employing Newton-type optimization procedures in one of the following variants: direct single shooting, direct multiple shooting, or direct collocation.

NMPC algorithms often make use of the similarity between successive optimum control problems. This allows for an efficient initialization of the Newton-type solution approach by a properly shifted estimate from the previously calculated optimum solution. As a result, a significant amount of computation time may be saved as a result of this. Path following algorithms are algorithms that never attempt to iterate any optimization problem to the point where it converges, but instead only take a few iterations towards the solution of the most recent NMPC problem, before proceeding to the next one, which is suitably initialized; see, exploit the similarity of subsequent problems even further[8].

NMPC is increasingly being applied to applications with high sampling rates, such as in the automotive industry, or even when the states are spread in space, thanks to the breakthroughs that have been made in controller hardware and computational algorithms, such as preconditioning. In the past, NMPC applications were predominantly used in the process and chemical industries, which had relatively slow sampling rates (Distributed parameter systems).

Recent aerospace applications of NMPC include tracking optimum terrain-following/avoidance trajectories in real time[9].

Model predictive control algorithm utilizes the following functions:

- An optimization algorithm
- A cost function J
- A dynamic model of the process
- Sliding mode control

Sliding Mode Control is an approach to nonlinear control that modifies the dynamics of a nonlinear system by applying a discontinuous control signal [10]. This control signal causes the system to slide over a cross-section of the system's normal behavior, which in turn modifies the dynamics of the system. Legislation to govern the input received from the state is not a time-continuous function. Instead, it can transition from one continuous structure to another in accordance with the position it now occupies in the state space. Control using a sliding mode is thus an example of control using a variable structure. The sliding-mode-control rule toggles between states according to the