Advancements in Autonomous Ship Trajectory Tracking: A Comparative Study of Mechanistic and Neural Network Models with NMPC

By Tanjil Islam

A dissertation Submitted to the School of Graduate Studies in partial fulfillment of
the requirements for the degree of

Master of Engineering

Faculty of Engineering and Applied Science

Memorial University of Newfoundland

St. John's, Newfoundland, Canada

October 2024

Abstract

The evolution of autonomous ships marks a significant stride in maritime operations, promising applications across a wide range of industries. These innovations enhance shipping and marine operations by improving safety through the reduction of human error and by enhancing the quality of life for mariners by alleviating tedious or difficult workloads. Whether in commercial shipping, passenger transport, or scientific research, autonomous ships are set to revolutionize the way we navigate the seas, making maritime activities more efficient and safer. Central to advancing this domain is the precise trajectory tracking of Autonomous Surface Vessels (ASVs), which is vital for their safe and efficient navigation. It is simultaneously required to operate within specified timeframes while adhering to maritime regulations and safely maneuvering amidst dynamic marine conditions such as waves, currents, and winds, which pose formidable technical challenges to autonomous trajectory tracking.

Presently, both model-based and data-driven controllers are pivotal in navigating Autonomous Vessels, emphasizing the critical need for accuracy and reliability in trajectory following. However, achieving precise trajectory tracking under real-world conditions remains intricate due to varying ship dynamics and environmental disturbances, necessitating tailored controller designs.

In this study, our principal aim is to develop a controller that comprehensively addresses these challenges while upholding safety constraints. We leverage Nonlinear Model Predictive Control (NMPC) for its suitability in handling nonlinear ship models, accommodating unmodeled dynamics, managing diverse constraints, and ensuring course stability amidst multivariable systems. An Unscented Kalman Filter (UKF) is integrated with NMPC to mitigate wave-induced disturbances and enhance robustness.

Our NMPC controller with UKF, implemented with mechanistic and Neural Network (NN) ship models, is evaluated through trajectory tracking simulations and experimental trials using the Magne Viking ship model at the National Research Council (NRC) in Canada. Incorporating an Artificial Neural Network captures intricate ship dynamics, exhibiting promising results in simulations and practical experiments. We compare the performances of mechanistic and NN models to validate their efficacy, proposing further enhancements through deep neural network training with natural data.

Integrating NMPC with neural network structures represents a core aspect of this research, aiming to advance autonomous ship trajectory tracking capabilities in real-world scenarios.

Acknowledgements

I am profoundly thankful for the grace of God, whose strength and patience have guided me to complete this thesis. His unwavering support has been the foundation of my journey.

I extend my sincere gratitude to my esteemed supervisors, Dr. Syed Imtiaz, Dr. Salim Ahmed, and Dr. Faisal Khan, for their invaluable guidance, generous assistance, and meticulous supervision. Their encouragement, insightful advice, and constant support were pivotal in the successful completion of this work.

I would like to acknowledge the National Research Council (NRC), Canada, for their valuable suggestions and support in providing experimental data, as well as for the financial assistance that made this research possible.

My heartfelt thanks go to my research colleagues, team members, and friends at Memorial University for their knowledge, expertise, and assistance throughout this academic endeavor. Special appreciation goes to Colleen Dalton, Tina Dwyer, and the staff of the Faculty of Engineering and Applied Science for their continuous support during my master's degree.

I am grateful to the Engineering Computing Service, Faculty of Engineering and Applied Science, for their substantial assistance during my academic journey, which significantly contributed to the success of this thesis.

Lastly, I want to express my deepest gratitude to my family and friends in St. John's and Bangladesh. I dedicate this thesis to the memory of my late father, Md. Abdul Majid, and I am thankful to my mother, Farida Majid, and my wife Tanjena Ferdous for their unwavering encouragement and support in all aspects of my life.

Table of contents

List of tables ………………………………………………………………….. vii

List of figures …………………………………………………………………. ix

1.0 Introduction …………………………………………………………………. 1

1.1 Background and Motivation ………………………………………….... 1

1.2 Aim and Objective ………………………………………………………… 4

1.3 Thesis Structure ……………………………………………………………. 5

2.0 Literature Review ……………………………………………………………. 8

2.1 Trajectory tracking of Autonomous surface vessels ………………………….. 8

2.2 Controllers for Trajectory Tracking …………………………………………… 9

2.3 Models for NMPC ………………………………………………………... 14

2.4 Use of Filters in Ship Control ………………………………………………... 18

3.0 NMPC-Based Trajectory Tracking with Mechanistic Model for Autonomous Maritime Navigation in Presence of Disturbances. ……………………………..... 22

3.1 Introduction ………………………………………………………………. 23

3.2 Methodology ………………………………………………………………… 30

3.3 Results and Discussion …………………………………………………… 49

3.4 Conclusion ………………………………………………………………... 75

4.0 Comparative Study of Physics-Based and Neural Network NMPC Models for the Autonomous Ship Trajectory Tracking…………………………………… 71

4.1 Introduction ……………………………………………………………………... 72

4.2 Methodology ………………………………………………………………. 76

4.3 Results and Discussion …………………………………………………………… 97

4.4 Conclusion ………………………………………………………………….....146

5.0 Conclusion and Recommendations ……………………………………….......... 148

5.1 Conclusion …………………………….……………………………………....... 148

5.2 Recommendations ………………………………………………………............. 150

References ……………………………………………………………………........... 151

List of Tables

Table 3-1: Controller Comparison Matrix: A Comprehensive Overview of Studies ……… 26

Table 3-2: Summary of ship motion variables for maneuvering applications ……………... 35

Table 3-3: Magne Viking Vessel Identified Parameters…………………………………….. 39

Table: 3-4 UKF algorithm for state estimation …………………………………………….. 43

Table 3-5: Error comparisons for simulation vs experimental results for the Circular Trajectory ………………………………………………………………………………………. 65

Table 3-6: Error comparisons for simulation vs experimental results for the Figure 8 Trajectory

………………………………………………………………………………………. 65

Table 3-7: comparison of the forces for different prediction horizon for the two trajectories

……………………………………………………………………………………... 72

Table 4-1: Summary of ship motion variables for maneuvering applications ……………..... 80

Table 4-2: Magne Viking model Vessel Parameters …………………………………...... 80

Table 4-3: Neural Network Training Data 83

Table 4-4: Neural Network Testing Data arrangement (Only firs steps are shown) 86

Table 4-5: The steps involved in Unscented Kalman Filter implementation.......................... 87

Table 4-6: Training with MATLAB Neural Network toolbox 92

Table 4-7 : Summary of the Neural Network Training Algorithm 99

Table 4-8: Error comparisons for circular trajectory for all the models ....

132

Table 4-9: comparison of the forces for different prediction horizon for the circular trajectory 133

Table 4-10: Error comparisons for Figure 8 trajectory for all the models 142

Table 4-11: comparison of the forces for different prediction horizon for the Figure 8 trajectory 142

Table 4-12 : Summary of the Deep Neural Network Training Algorithm 146

List of Figures

Figure 1-1: Conceptual overview of an Autonomous Ship's Core Systems 2

Figure 2-1: Workflow Diagram of Artificial Neural Network Algorithm 16

Figure 3-1: Full scale Magne Viking ship 34

Figure 3-2: Motion Variables of marine vessel 35

Figure 3-3: Magne Viking model ship (NRC, St. John’s, NL, Canada) 38

Figure 3-4 Tracking performance with UKF disabled 47

Figure 3-5 Tracking performance with UKF disabled... 47

Figure 3-6: Sample Trajectory Generation 49

Figure 3-7: Trajectory Tracking of two different trajectories in the presence of disturbances. 51

Figure 3-8: Wave model used for simulating disturbance..52

Figure 3-9: Comparison of X and Y positions for the circular trajectory 52

Figure 3-10: Comparison of X and Y positions for the Figure-eight trajectory 52

Figure 3-11: Comparison of heading angles for the circular trajectory................................. 53

Figure 3-12: Comparison of heading angles for the Figure-eight trajectory.......................... 53

Figure 3-13: The error for positions the different prediction horizons for circular and figure eight trajectory... 54

Figure 3-14: The heading error for different prediction horizons for circular and figure eight trajectory... 55

Figure 3-15(A): Comparison of velocity components for the circular trajectory for different N. 56

Figure 3-15(B): Comparison of velocity components for the Figure-eight trajectory for different N.. 56

Figure 3-16A: The error for surge velocities for the different prediction horizons for circular and figure eight trajectory. 57

Figure 3-16B: The error for sway velocities for the different prediction horizons for circular and figure eight trajectory... 57

Figure 3-16C: The error for yaw rates for the different prediction horizons for circular and figure eight trajectory.. 58

Figure 3-17A: Comparison of Forces for the circular trajectory for different N..................... 60

Figure 3-17B: Comparison of Forces for the Figure 8 trajectory for different N.................... 61

Figure 3-18: Trajectory Tracking of two different trajectories in the testing bay (NRC, Canada) 62

Figure 3-19: Comparison of X and Y positions for the circular trajectory.............................. 66

Figure 3-20: Comparison of X and Y positions for the Figure-eight trajectory...................... 66

Figure 3-21: Comparison of heading angles for the circular trajectory.................................. 67

Figure 3-22: Comparison of heading angles for the Figure-eight trajectory........................... 67

Figure 3-23: The error for positions the different prediction horizons for circular and figure eight trajectory. 67

Figure 3-24: The heading error of the different prediction horizons for circular and figure eight trajectory. 68

Figure 3-25(A): Comparison of velocity components for the circular trajectory for different N. 70

Figure 3-25(B): Comparison of velocity components for the Figure-eight trajectory for different N. 70

Figure 3-26A: The error for surge velocities for the different prediction horizons for circular and figure eight trajectory. 70

Figure 3-26B: The error for sway velocities for the different prediction horizons for circular and figure eight trajectory. 71

Figure 3-26C: The error for yaw rates for the different prediction horizons for circular and figure eight trajectory. 71

Figure 3-27A: Comparison of Forces for the circular trajectory for different N.................. 73

Figure 3-27B: Comparison of Forces for the Figure 8 trajectory for different N................. 74

Figure 4-1: Magne Viking model ship (NRC, St. John's, NL, Canada) 88

Figure 4-2: Feedforward Neural Network framework for training the dataset for Magne Viking. 89

Figure 4-3: Tracking performance with UKF-OFF 100

Figure 4-4: Tracking performance with UKF-ON 101

Figure 4-5: Figure 8 shaped Trajectory Generation.. 102

Figure 4-6: Mean Squared Error for NN training.. 104

Figure 4-7: Gradient, Learning rate, ssX and validation for NN.. 105

Figure 4-8: Error Histogram in NN training.. 107

Figure 4-9: Neural Network model testing with a data set unseen to the trained model...... 108

Figure 4-10: Wave model used in simulation.. 109

Figure 4-11: Trajectory Tracking of circular path in the presence of simulated disturbance. 110

Figure 4-12: Comparison of X and Y positions for the circular trajectory........................... 110

Figure 4-13: Comparison of heading angles for the circular trajectory................................ 112

Figure 4-14: Position error at different prediction horizons for circular trajectory.............. 112

Figure 4-15: Heading angle error at different prediction horizons for circular.................... 113

Figure 4-16: Comparison of X-velocity component $v_x$for the circular trajectory................ 114

Figure 4-17: The error in v_x at different prediction horizons for circular trajectory............ 114

Figure 4-18: Comparison of Y-velocity components v_y for the circular trajectory............. 115

Figure 4-19: The error in v_y at the different prediction horizons for circular...................... 115

Figure 4-20: Comparison of yaw rate v_ψ for the circular trajectory.................................... 116

Figure 4-21: The error in v_ψ at different prediction horizons for circular.......................... 116

Figure 4-22: Forces applied for the circular trajectory.. 118

Figure 4-23: Trajectory Tracking of Figure 8 in the presence of simulated disturbance...... 118

Figure 4-24: Comparison of X and Y positions for the Figure 8 trajectory......................... 119

Figure 4-25: Comparison of heading angles for the Figure 8 trajectory.............................. 121

Figure 4-26: Position error at different prediction horizons for Figure 8............................ 122

Figure 4-27: Heading angle error at different prediction horizons for Figure 8.................. 123

Figure 4-28: Comparison of X-velocity component $v_x$for the Figure 8 trajectory.............. 123

Figure 4-29: The error in v_x at different prediction horizons for Figure 8 trajectory........... 123

Figure 4-30: Comparison of Y-velocity components v_y for the Figure 8 trajectory............ 124

Figure 4-31: The error in v_y at different prediction horizons for Figure 8.......................... 124

Figure 4-32: Comparison of yaw rate v_ψ for the Figure 8 trajectory.................................. 125

Figure 4-33: The error in v_ψ at different prediction horizons for Figure 8 trajectory......... 125

Figure 4-34: Forces applied for the Figure 8 Trajectory.. 126

Figure 4-35: Magne Viking Ship model at NRC, Canada.. 128

Figure 4-36: Results of trajectory tracking of a circle at the experimental setup at NRC, Canada... 129

Figure 4-37: Comparison of X and Y positions for the Figure 8 trajectory......................... 130

Figure 4-38: Comparison of heading angles for the circular trajectory................................ 132

Figure 4-39: Position error at different prediction horizons for circular trajectory............... 132

Figure 4-40: Heading angle error at different prediction horizons for circular trajectory...... 133

Figure 4-41: Comparison of X-velocity component $v_x$for the circular trajectory................. 134

Figure 4-42: The error in v_x at different prediction horizons for circular trajectory............. 134

Figure 4-43: Comparison of Y-velocity components v_y for the circular trajectory............... 135

Figure 4-44: The error in v_y at different prediction horizons for circular trajectory............. 135

Figure 4-45: Comparison of yaw rate v_ψ for the circular trajectory...................................... 136

Figure 4-46: The error in v_ψ at different prediction horizons for circular trajectory............ 136

Figure 4-47: Forces applied for the circular trajectory 139

Figure 4-48: Trajectory Tracking of Figure 8 at experimental setup at NRC, Canada 140

Figure 4-49: Comparison of X and Y positions for the Figure 8 trajectory 141

Figure 4-50: Comparison of heading angles for the Figure 8 trajectory 142

Figure 4-51: Position error at different prediction horizons for Figure 8 143

Figure 4-52: Heading angle error at different prediction horizons for Figure 8 143

Figure 4-53: Comparison of X-velocity component $v_x$for the Figure 8 trajectory 144

Figure 4-54: The error in v_x at different prediction horizons for Figure 8 trajectory 144

Figure 4-55: Comparison of Y-velocity components v_y for the Figure 8 trajectory............. 145

Figure 4-56: The error in v_y at different prediction horizons for Figure 8........................ 145

Figure 4-57: Comparison of yaw rate v_{ψ} for the Figure 8 trajectory................................ 146

Figure 4-58: The error in v_{ψ} at different prediction horizons for Figure 8 trajectory........ 146

Figure 4-59: Forces applied for the Figure 8 Trajectory.. 147

Figure: 4-60 Initial validation results by training the neural network with natural data.......151

Figure: 4-61 Improved model by training with deep neural network.................................. 152

Figure 4-62: Deep Neural Network structure for training with the real data set................. 152

Chapter 1

1.0 Introduction

1.1 Background and motivation

An autonomous ship, also known as an un-crewed ship or autonomous vessel, is a type of watercraft that can at least partially operate and navigate without human intervention or with minimal human supervision. These vessels use various technologies, including sensors, cameras, radar, GPS, artificial intelligence, and advanced control systems to perform the tasks usually carried out by the ship crews. The concept of autonomous ships is similar to that of more established autonomous vehicles, such as self-driving cars and aerial drones, but applied to the maritime industry. Autonomous ships have the potential to revolutionize various aspects of maritime transportation and operations, offering benefits such as increased safety, reduced operational costs, improved efficiency, and the ability to operate in challenging or hazardous conditions without risking human lives. The maritime industry is highly interested into the potential integration of Autonomous Surface Vessels (ASVs) into its operation. As an example, Rolls-Royce stated in their 2017 announcement that they will be introducing a fully autonomous surface vessel within 2035 [1].

Ensuring that a ship follows a predefined trajectory is essential for safe navigation. The ship can avoid collisions with other vessels, obstacles, and navigational hazards by adhering to a planned path. This is especially critical in busy waterways and areas with heavy maritime traffic. Depending on the ship's mission or objectives, it may need to follow specific trajectories for scientific research, data collection, military operations, or other specialized tasks. Trajectory tracking allows the ship to meet these mission-specific requirements.

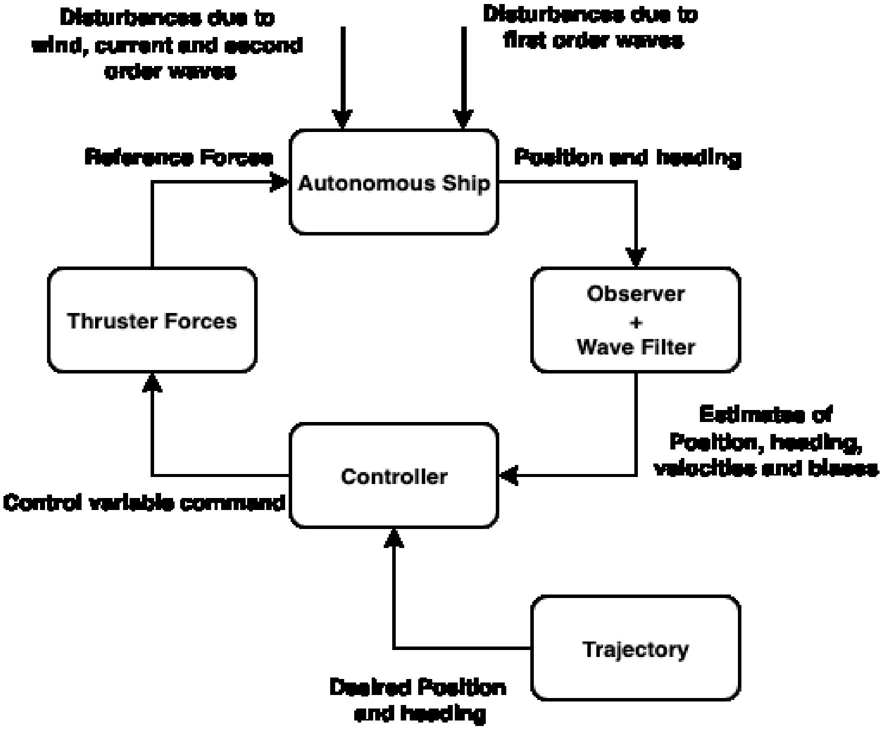

Disturbances due to wind, current and second order waves
Disturbances due to first order waves
Reference Forces
Autonomous Ship
Position and heading
Thruster Forces
Observer + Wave Filter
Estimates of Position, heading, velocities and biases
Controller
Control variable command
Trajectory
Desired Position and heading

Figure 1-1: Conceptual overview of an Autonomous Ship's Core Systems

Advanced control systems are responsible for the autonomous operation of the ship, including making decisions about navigation, route planning, collision avoidance, and responding to various situations and environmental conditions. The goal is to allow these vessels to navigate, make decisions, and respond to the environment with a high degree of autonomy and safety.

Trajectory tracking involves following a predefined path or trajectory with precision. Advanced control systems use sophisticated algorithms to calculate the optimal control inputs, such as steering and throttle, needed to maintain the ship on the desired trajectory. The marine environment is dynamic, with factors like wind, waves, currents, and tides affecting a ship's movement. An advanced control system can continuously analyze sensor data and make real-time adjustments to ensure that the ship stays on course, accounting for these environmental variations.

In addition to staying on course, advanced control systems can optimize the ship's trajectory for various factors, such as fuel efficiency, speed, or avoiding adverse weather conditions. Several hurdles must be surmounted before fully operational autonomous ships become a reality. These challenges include automatic path planning, navigation and trajectory tracking, cooperation with other vessels, power and energy management, fault detection, isolation, and reconfiguration. To tackle these obstacles, academia and the maritime industry have been actively conducting research efforts, with ongoing endeavors to find viable solutions. In [2], [3] an adaptive approach is proposed to govern an autonomous vessel's movement and path tracking. This method incorporates neural networks to estimate the propeller dynamics and manage uncertainties related to hydrodynamics. The [4], [5], [3] introduces a Nonlinear Model Predictive Control (NMPC) approach designed to precisely track position and velocity in surface vessels.

Our research focussed on building an advanced controller for the autonomous ship maneuvering. Advanced control systems ensure that autonomous ships can accurately follow predefined trajectories while adapting to changing environmental conditions and maintaining safety. These systems rely on sophisticated algorithms, sensor fusion, and real-time data processing for precise trajectory tracking in complex maritime environments. In our study, we employed NMPC using two distinct models—Mechanistic and Neural Network of the Magne Viking Ship model. These controllers' trajectory tracking performances were evaluated in both simulations, and a controlled experimental setup at the National Research Council Canada.

Controlling vessel motion has been a challenging task due to the intricate and nonlinear dynamics of vessels, along with the influence of changing environmental disturbances. As a result, it has drawn significant attention in the field. Despite numerous research endeavors, implementing adaptive control strategies within the maritime sector remains in its early stages of development. Most research was limited to the simulation works and focused on the one working model. Thus, comparing the performance of the different working models of the ship needed to be clarified. The previous research works were neither implemented on the complex trajectories nor on the trajectories required to follow tight radius. Including the intricacies of a complex trajectory in the simulation and the experiments are necessary. Previous works did not meet the challenges posed by the tight radius trajectories which are characterized by demanding turning paths. It requires a high level of control precision to get a successful tracking performance in the experiments for these trajectories. Also as mentioned above, majority of the studies were predominantly centered around simulation studies, for example, a mathematical model integrated with a model predictive controller. This limitations in studies have led to a gap in understanding the efficiency of different control strategies for maneuvering.

1.2 Aim and Objective

The primary goal of this thesis is to develop a Nonlinear Model Predictive Controller (NMPC) for Autonomous Surface Vehicles (ASVs). This controller should possess the capability to adapt to various model structures effectively, ensuring accurate trajectory predictions. To achieve this, we utilized both mechanistic models and Neural Network models that integrate with the NMPC.

The research objectives for this work are outlined as follows:

- Develop a NMPC controller for tracking ship trajectories, capable of adapting to any ship model. The controller should provide precise trajectory tracking and consistently meet the desired accuracy requirements.

- Create a Feedforward Neural Network-based ship model capable of capturing unmodeled ship dynamics in a mechanistic ship model.
- Assess the NMPC controller's performance by testing it with complex trajectories using mechanistic, neural network ship models in both simulation and real-life experimental setup at the NRC Canada facilities.
- Evaluate the controller's applicability in real-life scenarios through experimentation, with a focus on its potential implementation in actual maritime operations.

1.3 Thesis Structure

In this thesis, the research focuses on the application of NMPC for the trajectory tracking of ships. The investigation is carried out through simulation and real-life experiments conducted at the National Research Council Canada (NRC). To achieve the research objectives, several critical research tools and methodologies have been employed, primarily based on MATLAB.

Chapter 1 of the thesis sets ground for the research providing the background and the underlying motivations and delineates the research objectives. This chapter further provides an overview of the research issue that forms the focal point of this thesis.

Chapter 2 of the thesis provides a comprehensive literature review. This phase involves an in-depth analysis of existing literature, case studies, and scholarly work related to ship trajectory tracking and NMPC. The insights gained from this review inform the selection of appropriate models, methodologies, and tools for the subsequent chapters.

In Chapter 3, the primary focus is on the development and evaluation of the NMPC controller and the use of mechanistic model. The research tools and methodologies employed in this chapter include:

NMPC Optimization: MATLAB-based fmincon is utilized for NMPC optimization. This tool enables the fine-tuning of the controller parameters to achieve precise trajectory tracking.

Mechanistic Model: A mechanistic model is developed to describe the ship's dynamics. This model forms the basis for the control system and trajectory prediction.

Simulation and real-life experiments: The expected outcomes from Chapter 3 include simulation results that demonstrate the performance of the NMPC controller in the presence of wave as an environmental disturbance. For states estimations and wave filtering an Unscented Kalman Filter (UKF) is employed. The real-life experiment results at NRC Canada using the mechanistic model are also presented here in this chapter. The performance of the controller in simulation and real-life environment is also analyzed.

Chapter 4 represents the culmination of the research efforts, where the NMPC controller is tested in a real-life experimental setup at NRC Canada with different ship models. This phase involves the use of the established tools and models in a practical maritime environment. The primary research tools used in this chapter include:

Feedforward Neural Network Model: Utilizing MATLAB's deep learning toolbox, a Feedforward Neural Network (FFNN) model is constructed. This neural network captures some of the difficult to model system dynamics that are neglected in the mechanistic model.

Real-Life Experiments: Actual vessel experiments are conducted to validate the NMPC controller's performance under real-world conditions with the three different ship models, providing insights into the effectiveness of each model in trajectory tracking.

Data Collection and Analysis: Data collected during experiments are analyzed to assess the accuracy of trajectory tracking and the controller's adaptability to environmental factors.

Evaluating the NMPC controller's effectiveness through simulations and a practical maritime scenario demonstrates its potential for real-world applications.

Chapter 5 provides a brief recapitulation of the thesis's main attributes. Additionally, within this chapter, a set of recommendations for potential future research endeavors is presented.

In summary, this thesis employs a combination of MATLAB-based optimization, deep learning tools, and real-life experimentation to investigate the application of NMPC in ship trajectory tracking. The outcomes include improved trajectory tracking performance and insights into the adaptability and effectiveness of various ship models in both simulated and real-life settings.

Chapter 2

2.0 Literature Review

Trajectory tracking is a fundamental and critical aspect of autonomous maritime operations, as it directly influences the vessel's ability to navigate accurately, efficiently, and safely. Trajectory tracking involves precisely following a predefined path or trajectory by a vessel, ensuring it adheres to its intended route. In this context, a path refers to a series of waypoints or coordinates that outline the desired route the vessel should follow. At the same time, a trajectory includes not only the path but also the dynamic aspects of navigation, such as speed and timing at each waypoint. Trajectory tracking involves precisely following this predefined trajectory by a vessel, ensuring it adheres to its intended route in both spatial and temporal dimensions. This capability is essential for maritime applications, such as cargo transportation, scientific research, offshore operations, and defense. Accurate trajectory tracking is pivotal for safe navigation and avoiding collisions with other vessels or obstacles, especially in congested maritime environments. The importance of an advanced control system for the trajectory tracking of a ship cannot be overstated, and it is crucial for various reasons. The foremost importance of advanced control systems in trajectory tracking lies in their role in ensuring the safety of ships and the maritime environment. These systems are instrumental in making real-time adjustments to the ship's course and speed to prevent collisions with other vessels, obstacles, and navigational hazards, reducing the risk of accidents and protecting lives and property.

2.1 Trajectory Tracking of Autonomous Surface Vessels

Autonomous Surface Vessels (ASVs) have become a focal point of interest within the realm of control theory and engineering due to the immense commercial and military importance. The commercial applications of ASVs are evident in their contributions to oceanographic data collection, resource exploration, construction, hydrographic and environmental surveys. ASVs also play a crucial role in shipping and marine operations, including offshore transfers, ice clearing, and refueling. Meanwhile, their military roles encompass intelligence, surveillance, reconnaissance, mine countermeasures, and antisubmarine warfare [6]. Summarized in [7], the functions of ASVs are primarily categorized into three key areas: Dynamic Positioning (DP), path-following, and trajectory tracking. A DP system, in its essence, manages the control of an ASV during fully actuated, low-speed operations. DP system's primary objective is maintaining the ASV's position and heading at a fixed point or a designated waypoint along a track despite of disturbances. Notable applications of DP include pipeline laying and oil/gas exploration. Recent DP research endeavors have addressed model uncertainties and disturbances intrinsic to ASV systems, aiming to enhance control performance [8], [9]. The pursuit of path-following control revolves around guiding the ASV along a predefined route with a suitable speed profile. In scenarios where ASVs operate at high speeds, most lack direct actuation in the sway direction [1]. Consequently, the foremost challenge in path-following control pertains to the underactuation of ASVs. In this configuration the degrees of freedom were higher than the control vector's dimensions. This challenge has been effectively met by employing methodologies such as backstepping and Lyapunov's direct approach where the lateral position of the ship (sway direction) cannot be directly controlled because of the under actuation. [10]. The backstepping method was used to stabilize the nonlinear systems by decomposing the system in a series of interconnected small subsystems. Trajectory tracking involves an ASV's precise following of a spatial and temporal trajectory, with strict timing requirements. This is particularly vital in offshore applications, ensuring safe navigation, emission reduction, and energy conservation [11]. Distinct from path-following, trajectory tracking necessitates incorporating a specific speed control law, alongside the steering control law.

2.2 Controllers for Trajectory Tracking

Robust control methods ([12],[13],[14]) aim to make a ship's position error very close to zero where they have used composite nonlinear feedback path following controllers and Lyapunov method and finite time leader follower formation. However, they cannot precisely determine how close it gets, which affects how accurately the ship follows its path. Some methods achieve a specific level of accuracy or get very close to zero, but they take a very long time to do so using adaptive finite time fault tolerant controller [15]. Others are quicker ([16], [17]) where motion planning was used. But in this case, they needed to be more precise to predict how long it will take to converge. Also, they depend on initial conditions and unknown factors like control limits or uncertainties in the ship's model. In simple terms, the challenge of making a ship follow its path very accurately within a fixed and known time frame is still a problem that still needs to be fully solved, even though it is crucial in marine control. Advanced control systems are pivotal in ensuring the safety, precision, and efficiency of trajectory tracking in ships and the maritime environment. They are instrumental in making real-time adjustments to prevent collisions, protect lives and property, and optimize trajectories for cost savings and reduced environmental impact. These systems can adapt to dynamic environmental conditions, ensuring ships stay on course while offering redundancy and fail-safes for continued operation during system failures. Additionally, they help vessels comply with maritime regulations, promote predictability for safer navigation, and can be customized to meet specific mission requirements, whether in research, military operations, or other specialized tasks.

2.2.1 Most used control methods for trajectory tracking

It is imperative to employ a suitable tracking controller alongside specific control methodologies to ensure both robust tracking performance and vehicle stability based on the chosen model. Over the years, extensive research has yielded many control methods tailored for the trajectory tracking of autonomous vehicles. This section presents some of the most widely employed techniques, complemented by cutting-edge research findings. Furthermore, we delve into various optimization strategies to mitigate the identified weaknesses and further enhance performance [59].

PID controller

The PID controller, a linear feedback control system, is extensively used in autonomous vehicles for its straightforward design. Notably, it offers ease of operation and adjustment while maintaining a degree of robustness. A significant advantage lies in its independence from precise knowledge of mathematical models or the specific plant, setting it apart from other controllers. This feature enhances its broad applicability across diverse applications, as highlighted in the work of [18]. However, vehicles equipped with a PID (Proportional-Integral-Derivative) controller tend to exhibit subpar performance when they commence their operations from varying initial conditions, as observed by [19]. In response to this issue, [20] introduced a fuzzy PID controller designed to address this challenge. This novel controller not only adapts effectively to diverse starting conditions but also offers the benefit of faster convergence without the presence of steady-state tracking errors. However, the inherent linearity of traditional PID controllers introduces certain complexities when dynamically adjusting PID control gains in real-time.

Fuzzy Logic Controller

To address this issue, a proposed controller based on fuzzy logic principles offers a promising solution. Fuzzy controllers excel in scenarios where a complete mathematical model is unavailable, thanks to their adaptability and approximate reasoning capabilities. Furthermore, fuzzy logic is adept at handling imprecise and uncertain circumstances, making it a suitable choice. Fuzzy logic controllers exhibit excellent convergence to reference paths, ensuring safe and smooth autonomous driving. Additionally, these controllers offer real-time operation, enhancing computational efficiency, as emphasized by [21]. Consequently, the application of fuzzy-logic-based steering control in autonomous vehicles benefits from these advantages. We can find relevant fuzzy controllers in the studies by [22]. However, it's important to note some limitations of the fuzzy-logic-based controller. One such limitation is an increased tendency for steering command oscillations, as observed by [23]. Moreover, establishing the stability and performance analysis of these controllers can be challenging.

Sliding Mode Controller

Sliding mode control (SMC) techniques are widely employed in the field of nonlinear control systems due to their distinctive attributes. SMC offers remarkable performance against parameter uncertainties, disturbances arising from pronounced nonlinearity, external disruptions, and the intricate operating conditions encountered by autonomous vehicles, as demonstrated in the works in [24]. Furthermore, the SMC controller facilitates swift convergence of the vehicle to its intended path, as evidenced in the work of [25].

SMC boasts several advantages, including rapid response, insensitivity to variations in parameters and disturbances, and straightforward physical implementation, among others. Moreover, SMC can be effectively combined with other control algorithms to address each other's limitations, as highlighted by [26]. But SMC's practical utilization in vehicles presents certain challenges. The need for continuous linearization to maintain controlled variables close to the reference path, as discussed by [27], poses a significant hurdle. Additionally, when employing the SMC approach for trajectory tracking, it can result in substantial lateral acceleration, particularly in scenarios involving rapid changes in the curvature of the reference path, such as in a severe double lane change scenario, as pointed out in [28].

Model Predictive Controller

Model predictive control (MPC) is a powerful state feedback controller. It leverages the mathematical model of vehicles to anticipate their future behavior within a limited forecast horizon, MPC demonstrates exceptional aptitude in addressing control issues involving uncertain and nonlinear systems, largely due to its inherent capabilities. Given the significance of constraints in autonomous vehicle operation, MPC plays a crucial role in maintaining vital limits, such as rollover prevention, slip control, and lateral stability. Its constraint-handling capacity has made MPC the preferred algorithm for managing scenarios that push the boundaries of vehicle dynamics. For instance, [29] successfully resolved the challenge of adhering to tire-road friction limits using an MPC controller.

Furthermore, MPC excels at managing multi-input and multi-output (MIMO) systems with intricate input-output interactions, a task that often proves challenging for PID controllers.

Additionally, MPC is regarded as an evolution of optimal control strategies. It solves an optimization problem, involving cost functions, to determine inputs that meet desired objectives and constraints over a specified time horizon at each time step, as outlined in the works of [30] and [31]. Given these advantages, MPC emerges as a natural choice for trajectory tracking control in autonomous vehicles, capable of computing optimal solutions while accommodating both soft and hard constraints.

In a broader context, MPC can be categorized into linear MPC (LMPC), nonlinear MPC (NMPC), adaptive MPC, and robust MPC. LMPC excels at tracking fixed operating points attainable with a linear model, as described by [30]. However, LMPC relies on predictions based on a linear time-invariant (LTI) model, making it more sensitive to prediction errors. As a result, its tracking performance deteriorates in the presence of strong nonlinearity, given the degradation in LTI prediction accuracy, a point emphasized by [32]. On the other hand, NMPC employs cost functions and nonlinear mathematical plant models to compute optimized inputs while adhering to specified constraints, making it especially well-suited for handling the high degree of nonlinearity inherent in dynamic vehicle models. Despite its advantages, a significant limitation of MPC, in general, is its difficulty in implementation. Solving the optimization problems in MPC often requires significant computational effort and the solution of complex functions with numerous local minima. This computational complexity can make real-time implementation challenging, particularly for systems with fast dynamics or limited computational resources.

2.3 Models for NMPC

To enclose a broader spectrum of working models with NMPC for the complex trajectories we have used the mechanistic, Feed Forward Neural Network (FNN) and hybrid model. The FNN is discussed in this chapter. The latter two models will be discussed in the Chapters 3 and 4.

Artificial neural networks are robust when solving complex problems that involve non-linear relationships and the manipulation of multiple parameters. Their ability to learn from data and adapt to various tasks makes them a versatile and widely used technology in machine learning and artificial intelligence. An artificial neural network (ANN), often referred to as a neural network (NN), employs interconnected mathematical nodes or neurons to construct a network capable of representing intricate functional relationships. This technique is especially well-suited for addressing problems that entail the manipulation of numerous parameters and non-linear interpolation. Consequently, it offers an effective solution for scenarios where traditional theoretical and mathematical methods may fall short [33]. To create a high-performing neural network, having a substantial amount of experimental data is crucial. Throughout the training and testing phases, it is essential to fine-tune the network's structure, learning algorithm, and various parameters to align with the specific problem being studied. Once the neural network is finely tuned and trained on this data, it becomes capable of yielding favorable results when provided with new input data, even if it hasn't encountered it before. The Figure 2-1 shows the fundamental concept of Neural Network [34]. Artificial Neural Networks (ANNs) are the models inspired by the functioning of human neural networks. They aim to process and make sense of vast and intricate datasets. ANNs accomplish this by leveraging mathematical connections between their constituent processing units, collectively forming the network's architecture. Through this interconnection, ANNs have the capability to categorize data by assigning numerical weight values to each input, enabling the analysis and classification of cases [35].

ANNs excel at recognizing and learning complex patterns in data. In the context of autonomous ships, various factors like

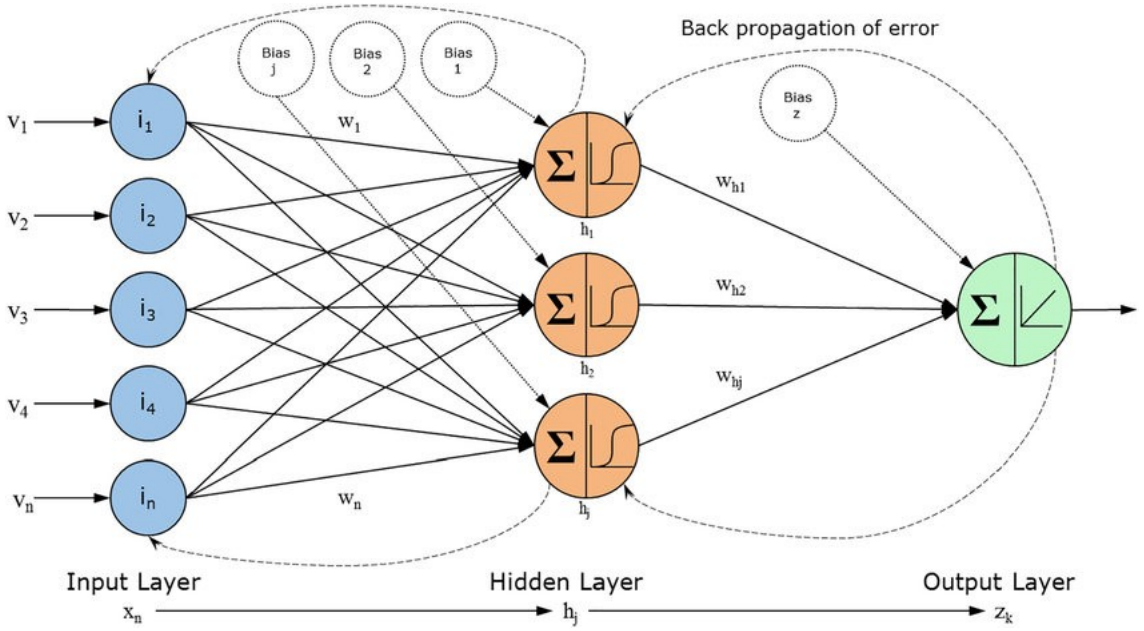


Figure 2-1: Workflow Diagram of Artificial Neural Network Algorithm

ocean currents, weather conditions, other vessels, and navigation regulations contribute to trajectory planning. ANNs can analyze historical data and adapt to these intricate patterns.

2.3.1Feedforward neural network model for NMPC

Artificial Neural Networks (ANNs) combined with NMPC can offer a powerful and flexible solution for various application. Potentially it can outperform the control strategies based on mechanistic models. ANNs can approximate complex, nonlinear systems more accurately than traditional linear models. By using ANNs to represent the dynamic system, NMPC can account for intricate and time-varying system behaviors, making it a suitable choice for systems where linear models may be inadequate. ANNs can enhance the predictive accuracy of NMPC by capturing system dynamics and disturbances effectively. This leads to more precise future state predictions, enabling better control decisions. As the system evolves or faces unmodeled variations, ANNs can adjust their predictions, allowing NMPC to maintain high performance and robustness. The flexibility of Neural Network based models allows for encoding and handling various constraints, including state bounds, input constraints, and safety constraints, which is essential in many control applications. It can also reduce the computation cost associated with the NMPC optimization problem solving. The combination of Artificial Neural Network and NMPC can lead to suitable operation in real time environment where fast control responses are necessary, such as robotics, autonomous vehicles, and process controls. By handling the system uncertainty and measurement noise the neural network model can create a robust NMPC controller. The neural network model can handle multi-objective optimization for NMPC, such as tracking the reference trajectory as well as minimizing the energy consumption. In some cases, the neural network can continuously the update the model based on the recent data which in turns improves the control performance. In recent research, learning based solutions for modelling the non-linear behaviour of autonomous vehicles have been proposed ([36] ,[37],[38]). Artificial Neural Networks (ANNs) models are very useful in the field of ship control due to their ability to model and provide effective solutions for various nonlinear systems in predictable situations [39].

The early research involved using ship position, course, and speed as inputs in the neural network and the controller provided the following outputs, rudder angle and propeller speed. A minimum-time ship maneuvering method utilizing a neural network controller and model predictive compensator was introduced by. An artificial neural network was used for real-time control by interpolating pre-calculated minimum time solutions [40]. These solutions, however, did not consider environmental disturbances and they did not maintain consistent training data. The controller proposed in [42], adaptive backstepping based controller was proposed which can move slowly at wind disturbances. But at this point the longitudinal velocity of the ship was considered zero. A study in [43], used a feedforward neural network along Proportional - Derivative (PD) controller. It was difficult work with PD controllers for multivariate systems. While exploring the literature, applications of neural networks (NN) for vessel control in the maritime domain appeared to be a relatively under-explored area. So, to overcome these problems a precisely trained neural network is used along with NMPC in this paper's chapter 4.

2.4 Use of Filters in Ship Control

Wave filtering is a vital component in trajectory tracking control systems, directly impacting the performance of the final control system. Effectively controlling a vessel's position and heading presents a unique challenge in dynamic sea states. While intuitively, reacting to every wave disturbance might seem desirable, there are compelling reasons why this approach could be more practical. Continuously adjusting course corrections to counter high-frequency wave motions would be incredibly energy-intensive. Ships, huge ones, have inherent inertia. They can't react instantaneously to high-frequency wave disturbances. Trying to do so would result in jerky, unstable motions, potentially compromising safety and maneuverability. This is where wave filtering comes in. It acts like an intelligent separator, isolating the high-frequency wave disturbances from other, lower-frequency control signals. The control system focuses on these slower signals, allowing the ship to respond to critical factors like currents, wind, and the desired course.

The Extended Kalman Filter (EKF) is a widely adopted method in numerous nonlinear estimation and machine learning scenarios. Its applications perform tasks such as estimating the state of nonlinear dynamic systems, determining parameters for identifying nonlinear systems, and engaging in dual estimation (e.g., using the Expectation Maximization (EM) algorithm) where both states and parameters are simultaneously estimated.

In [62] researchers use EKF and Particle Filter-based sensor fusion algorithms to address dynamic ship positioning. The proposed method combines onboard sensor measurements of a ship's position and heading with distance data from coastal sensors (e.g., radar) to accurately estimate the ship's state vector. The offshore autonomous industry widely uses Kalman Filter for velocity estimation and wave filtering. The article in [57] explores vessel response models for Kalman filter design, touching on parameter and noise covariance estimation for filter tuning. The provided case study demonstrates the application of Kalman filters in ship autopilots, aligning with contemporary offshore industry practices. The EKF was also utilized in Autonomous Underwater Vessel's tracking [63]. In this proposed method the EKF is employed to handle uncertainties in localization and ocean currents. The Extended Kalman Filter is suggested as an adaptive algorithm in [64] for estimating position, velocity, and acceleration. This estimation is crucial for predicting the trajectory of maneuvering ocean vessels. This paper in [66] introduces globally asymptotically stable time-varying kinematic filters for estimating linear motion quantities (position, linear velocity, and acceleration) in three dimensions for mobile platforms. The filters, derived from Kalman and optimal filters for linear time-varying systems, provide explicit optimal filtering solutions through coordinate transformations.

The EKF linearizing the system locally and uses the linear system for predicting the system states. However, this simplification can lead to significant errors in the predictions, resulting in less accurate results and, in some cases, filter failures. The EKF leverages the well-established Kalman Filter framework, which operates efficiently with linear systems. The problem arises when dealing with highly nonlinear systems. In these cases, the actual system behavior can deviate significantly from the linearized model, mainly when operating away from the chosen linearization point. In extreme scenarios, the linearization error can accumulate over time. These accumulating errors can cause the EKF's state estimates to diverge significantly from the true system state.

The Unscented Kalman Filter (UKF) offers a solution to this problem by using a different approach. It approximates the distribution of the states by choosing a small number of specific points within that distribution. These carefully selected points, called sigma points, are passed through the nonlinear system, they provide more accurate predictions up to the third order of the distribution. In contrast, the EKF only provides predictions up to the first-order accuracy. Despite its enhanced accuracy, the UKF doesn't significantly increase computational complexity compared to the EKF, making it a practical choice for many applications. The UKF provides a more accurate approximation of the nonlinear probability distribution for the solution versus the propagation function. This generally results in better convergence and a more precise capture of nonlinear dynamics. Additionally, the UKF eliminates the need to calculate the Jacobian.

In [46], The significant improvements in performance achieved by the Unscented Kalman Filter (UKF) in the estimation of state for nonlinear control was shown. In linear systems, the finite dimensional Kalman filter can advance using Gaussian distributions. In contrast, for nonlinear systems, approximations of probability distributions must be made at each step of the process [47]. In this research we have utilized the Unscented Kalman Filter to filter out the waves while the trajectory tracking of the ship. In the study [65], a novel Adaptive Unscented Kalman Filter

(AUKF) is applied to the ships' dynamic positioning (DP) system. The AUKF addresses model uncertainties, encompassing time-varying noise statistics, model mismatch, and slow-varying drift forces. The adaptive algorithm is designed to dynamically adjust real-time process and measurement noise covariances, utilizing the critical principle of covariance matching. The study in [67] suggests a tracking controller for underactuated nonlinear autonomous ships to navigate along a reference path amidst constant disturbances from waves, wind, and ocean currents. The controller, developed using an unscented Kalman filter (UKF) and backstepping techniques, employs UKF to update uncertain parameters dynamically online. This approach prevents parameter drift caused by time-varying added mass matrices. The UKF in collaboration with neural networks also provided great results for autonomous vessels. The outcomes in [68] demonstrated that the autopilot, based on the UKF and Radial Basis Function Neural Network (RBFNN), effectively fulfilled course keeping, course changing, and trajectory tracking. The study in [69] a comparison between EKF and UKF was demonstrated in the field of underwater robotics. The study shows the better performance of UKF using the same sensors and processes for the UAV for the two filtering methods. The computational load was well handled by the UKF and provided more accuracy for better navigation of the UAV.

Chapter 3

3.0 NMPC-Based Trajectory Tracking with Mechanistic Model for Autonomous Maritime Navigation in Presence of Disturbances.

Abstract

Autonomous ships represent the future of maritime operations, with applications ranging from military missions, search and rescue operations, oil and gas exploration, and various other fields. Ensuring the safe and efficient trajectory tracking of Autonomous Surface Vessels (ASVs) is a paramount concern for advancing this industry. Precisely following paths while adhering to maritime rules and regulations within specific time constraints play a crucial role in autonomous ship operations. This challenge becomes even more significant in the marine environment, given unavoidable factors such as waves, currents, wind, and other environmental disturbances. Currently, both model-based and data-driven controllers are equally used for navigating Autonomous Vessels. An indispensable aspect of these controllers is their ability to keep autonomous ships on the intended trajectory accurately and reliably. Achieving precise trajectory following under real-world conditions remains a complex challenge. The varying dynamics of different ships and environmental conditions hinder a one-size-fits-all controller design. The controller's role becomes even more intricate when dealing with tight and complex trajectories.

In this research, our primary objective is to design a controller with a comprehensive consideration of these factors while adhering to essential safety constraints. We employed Nonlinear Model Predictive Control (NMPC) to address these concerns, particularly well-suited. NMPC allowed us to work with nonlinear ship models, account for unmodeled dynamics, satisfy diverse constraints, and maintain the ship's course, all while managing multivariable systems effectively. Additionally, we integrated an Unscented Kalman Filter (UKF) alongside NMPC to filter out wave-induced disturbances, enhancing the controller's robustness. This combined NMPC controller with UKF, implemented with three distinct models, was employed to conduct trajectory tracking operations in simulation and experimental setups at the National Research Council (NRC) in Canada with the Magne Viking ship model.

3.1 Introduction

Various aspects of marine transportation and operations has been revolutionized by the emergence of Autonomous Surface Vessels (ASVs). The significant advantages of ASVs include safer operation, decreased operational expenses, enhanced efficiency, and the capacity to undertake tasks in demanding or perilous environments while safeguarding human lives. The marine industry is very much interested to add the Autonomous Ships in their operations. By adapting and following a planned path precisely the applicability of ASVs can move one step further. Ships are employed for different activities, like scientific research, gathering data, or military operations. Controller for trajectory tracking is one of the key elements of ASV. Route planning, proper navigations, collision avoidance and maintaining stability in various difficult environmental conditions are the main objective of designing a controller for autonomous ship.

Over the years, many controllers are designed for the trajectory following of the ASVs. It is very crucial function of the controller to ensure the robust tracking performance while taking the environmental conditions into account. In Proportional Integral Derivative (PID) controllers there are advantages because it does not require precise knowledge of the mathematical model of the system. We can see the widespread use of the PID controllers [18]. Varying initial conditions can provide mediocre performance for the trajectory tracking problems for the vehicles [19]. Also, PID requires substantial adjustments of its parameters if any changes occur while the system is operating whereas NMPC is highly robust when multiple changes occur in the system [48]. The fuzzy PID controllers provides faster convergence and effective functionality in different initial conditions [20]. But the inherent linearity of PID controllers produce complexity when the dynamic control of gains is required in real time. Some Fuzzy Logic Controllers can be used as a good controller for autonomous vehicle, but they have limitations in the steering command because they can produce substantial amount of oscillation [23]. Sliding Mode Controller (SMC) is well suited for the nonlinear control systems. It offers noticeable performance while facing external disruptions and critical operating conditions encountered by the ASVs [24]. But they need to maintain continuous linearization for getting appropriate control variables to keep the vehicle in its course [27]. Robust control methods in ([12],[13],[14]) tries that a ship's position tracking error gets very close to zero where they implemented Lyapunov method, composite nonlinear feedback path following controllers and finite time leader following formations. But ultimately the convergence was slow and could not determine precisely how close it can go. Methods in [15] using finite fault tolerant controller produced good accuracy but the convergence rates were slow. The quicker solutions presented in [16] and

[17] using motion planning but were dependent on the initial conditions, control limits and uncertainties in the ship models.

The [49] shows good results for trajectory tracking of the ship with a NMPC controller integrated with genetic algorithm. But in this case the simulation did not involve any disturbances. Model Predictive Control (MPC) inherently offers a solution for managing constrained systems, showcasing its ability to explicitly handle system constraints. MPC has found widespread use in industrial processes [50]. Lately, The MPC has been applied to control the path of surface vessels [51] and heading control [52]. In [53] MPC is utilized for vessel path following with Line of Sight (LOS) guidance algorithm where a linear model of the ship is utilized showing the improvement of the performance in the simulations. However, this Lyapunov- based design techniques it is difficult to incorporate constraints. Moreover, the applications of MPC in vessel systems mentioned above are primarily limited to path following problems whereas the trajectory tracking has a wide range of applications with the applications of advanced methodology.

In the literatures [3], [4], [5] a Nonlinear Model Predictive Control (NMPC) algorithm was used to track the position and velocity of the ships. Vessel motion controlling is a challenging task due to the nonlinear dynamics of the vessel and the presence of the environmental conditions like waves, currents, and wind. Implementing adaptive control in the marine environment was proposed in [2], [3]. Even though the research work is done extensively the implementation of adaptive approach is still in rudimentary level.

It is also important that the controller is tested with different trajectories and conditions to make it applicable for the real time testing. Most of the works were confined within the simulations.

The previous research works did not capitalize on the ship following the complex trajectories with tight radius. This thesis aims to develop a Nonlinear Model Predictive Controller (NMPC) for Autonomous Surface Vessels (ASVs). The objective is to create a controller that can effectively adjust its features for different ship models, providing precise predictions for ship trajectories. To accomplish this, we employed a combination of mechanistic model and Unscented Kalman Filter (UKF) models seamlessly integrated into the NMPC. Apparently, within a fixed time frame making a ship to follow its trajectory accurately and efficiently is still very challenging. The advanced control is pivotal and essential for safety, precision, prevention of collision in a marine environment. In this section the following table shows the comprehensive comparisons of the different controllers used in different studies along with their advantages and disadvantages.

Table 3-1: Controller Comparison Matrix: A Comprehensive Overview of Studies

Controller Type	Study Reference	Application Domain	Advantages	Disadvantages
PID	Lu´cia Moreira , Thor I. Fossen , C. Guedes Soares, Path following control system for a tanker ship model	Automatic path following control system for an autonomous marine surface vessel.	• Easier implementation of the controller. • LOS guidance methods were used which can be later incorporated with weather routing, collision and obstacle avoidance.	• Only simulation results are presented. • Path following was the main focus not trajectory tracking
	Dinh Due Vo et al. Designing a	Ship autopilot system for trajectory	• Easy to implement controller with	• Simulation and experimental

	PID Controller for Ship Autopilot System	tracking	less dependence on the ship model. • Experimental results were presented.	results did not show any involvement of the disturbances in the system. • The experiment was done in a swimming pool which is might not be a good experimental setup. • The experimental results for path tracking and heading control were far away from the calculated path and headings.
	Adinath Jain et al. PD CONTROLLER BASED UNMANNED SHIP NAVIGATION	Autonomous Ship Navigation	• Simple formulation and application of the controller. • Six degrees of freedom ship dynamics is used for formulating the problem	• Linearized model is used which may not be able to capture all the characteristics of the ship. • Only simulation was the means to test the performance of the controller and disturbance model was introduced in the simulation with Gaussian random process.

				- It is assumed that the waves are generated due the wind only.
Fuzzy PID	Sin-Der Lee; Ching-Yaw Tzeng; Ber-Jin Chen, Design and experiment of a fuzzy PID track-keeping ship autopilot.	Track keeping of the autonomous ships	- Simulations and Experiments were conducted to show the effectiveness of the controller. The test was conducted in open water.	-Tuning of the PID and Fuzzy membership functions are complex. - Simple straight forward trajectory was used. It is not clear how it will perform in the complex trajectories.
	Minh-Duc Le et al, A new and effective fuzzy PID autopilot for ships		- Easy to construct controller. - Better performance than PID.	- Not efficient in handling the nonlinear systems. - Only simulation was done as a controller's performance evaluation criteria. - Doesn't mention much about the speed control, thrust allocation. - Not much knowledge about the trajectories used in the simulation.
NMPC	Bruno J. Guerreiro et al. Tracking Nonlinear Model	Trajectory tracking control for autonomous surface craft (ASC) in the	- The simulations were done with modeled constant currents.	- Use of a wave filter would have been a better addition. There was no filter used.

	Predictive Control for Autonomous Surface Craft	presence of ocean currents		
	Mohamed Abdelaal et al. NMPC-based trajectory tracking and collision avoidance of unmanned surface vessels with rule-based colregs confinement.	Trajectory tracking and collision avoidance of the ASVs	• Collision avoidance was also shown along with the trajectory tracking problems. • The ACADO toolbox in MATLAB was utilized to solve the NMPC problem which makes the iteration time much faster. The execution time of the algorithm was less than 20ms.	• What kind of disturbances were simulated that was not clearly mentioned but a disturbance observer is used. • Only simulation results were added to show the study for the trajectory tracking and collision avoidance. • The optimum prediction horizon was not mentioned for trajectory tracking. • Though the multiple shooting optimization was used there was only two thrusters (surge and yaw moment) used for the model generation. No sway force was mentioned. • Instead of a straight-line trajectory a complex or irregular shaped trajectory could be used to show

				the versatility of the controller.
	Martin Kosch et al. Hardware-in-the-Loop Trajectory Tracking and Collision Avoidance of Automated Inland Vessels Using Model Predictive Control	Trajectory tracking and collision avoidance of the ASVs	• Hardware used in this study takes the influence of the communication delays. • CaSadi toolbox utilization provides a better computational time. • Collision avoidance and trajectory tracking is combined.	• Artificial white noise was introduced as a disturbance in the simulation which is not a good method to replicate waves. • EKF was used to filter the disturbances. EKF involve linearization of the system which may produce inaccurate results. • In the model of the ship no Coriolis component were added which can make the model less accurate.

3.2 Methodology

The trajectory tracking of autonomous ships involves tracking their position, analyzing heading angles, observing velocity, and allocating thruster forces. A robust and precise controller must handle all these elements effectively, ensuring they stay within acceptable limits for both simulation and experimental setups. In this section we will discuss about the methods and techniques used for the trajectory tracking of the Autonomous Surface Vessel. Our developed controller was implemented on the Magne Viking Ship simulated model as well as on scaled physical model. The following steps will be discussed in this part of the study:

- Design of Nonlinear Model Predictive Controller.

- Vessel Model for Magne Viking.
- Disturbance Models.
- Implementation of Unscented Kalman Filter.

3.2.1 Design of Nonlinear Model Predictive Controller

An NMPC scheme is used for trajectory tracking of the surface vessel Magne Viking. A 3 – DOF model is used with three control variables Surge force, Sway force and Yaw moment. In ship maneuvering control, it's typical to represent a ship's motion with a 3-degree-of-freedom (DOF) model, combining surge, sway, and yaw movements, while disregarding heave, roll, and pitch motions [54]. We used a quadratic cost function. A real time efficient MATLAB code is generated using the Optimization Toolbox and fmincon solver. Nonlinear Model Predictive Control (NMPC) utilizes the vessel's nonlinear dynamics model directly as the prediction model. It iteratively solves a nonlinear optimization problem online at each time step [55]. The fundamental components of an NMPC include the cost function, prediction model, state constraints, and input constraints. In this section, we elaborate on the design of each of these elements. The NMPC utilizes an output feedback control architecture and implements the offset-free formulation as proposed in [56]. The state space model f represents the nominal six state model of the trajectory tracking. The predicted states can be descried by the equation (3.1).

$$x(k+T) = x(k) + \int_{k}^{k+T} f\big(x(\tau)\big)d\tau, \tag{3.1}$$

Where x (k) is the current state and T is the sampling time.

In this equation:

$x(k)$ is the state vector at the current time step k,

$x(k+T)$is the state vector at the next time step $k+T$

T is the time increment,

$f\big(x(\tau)\big)$ represents the dynamic model of the system, describing how the state vector evolves over time,

τ is the variable of integration over the interval from k to $k+T$

This equation models the continuous-time dynamics of the ship, integrating the dynamic function $f\big(x(\tau)\big)$ over the time interval to update the state vector from the current time step to the next.

In this study, the state vector $x(t)$) is defined as:

$$x(t) = [x, y, \psi, v_x, v_y, v_\psi]$$

where:

$x,$ and y represent the position coordinates,

ψ denotes the yaw angle,

v_x, v_y are the surge and sway velocities, respectively,

v_ψ is the yaw rate.

To create an offset free NMPC, we use the findings from the [56] and [57]. As a starting point, we introduce a disturbance model and a disturbance integrator into the prediction model, forming an augmented model denoted as $\mathrm{f}_{\mathrm{aug}}$. This augmented state space model combines the states with the disturbance model d as expressed in Equation (3.2). During state prediction, the initial estimate of the disturbance remains constant, like a step disturbance, as outlined in Equation

(3.3). The prediction model is numerically integrated using the explicit Euler's Method. The forecasted state is determined by Equation (3.2).

$$x(k+T) = x(k) + \int_{k}^{k+T} f_{aug}(x(\tau), d(k))d\tau, \tag{3.2}$$

$$d(k+T) = d(k), \tag{3.3}$$

The predicted output is given by equation (3.4) where g_{aug} is the output model.

$$y(k+T) = g_{aug}(x(\tau), d(k)) \tag{3.4}$$

3.2.2 Optimization Scheme

The objective of the cost function is to minimize the discrepancy between the desired equilibrium state targets $\bar{x}$ and the actual system states $x(k)$. This includes aligning the equilibrium input target $\bar{u}$ with the current input $u(k)$ as well to ensure precise tracking of the reference signal, denoted as $r(k)$ which is the desired trajectory properties for the ship. The goal is to achieve offset-free tracking, meaning that the system maintains alignment with the desired reference over time.

$$J = \frac{min}{u} \sum_{\kappa=k}^{k+m}(\hat{x}(\kappa) - \bar{x})^T \lambda_1 (\hat{x}(\kappa) - \bar{x}) + \lambda_2 (u(\kappa) - \bar{u})^2) \tag{3.5}$$

$$\bar{x} = f_{aug}(\bar{x}, \bar{u}, \hat{d}(k)) \tag{3.6}$$

$$\bar{r} = g_{aug}(\bar{x}, \hat{d}(k)) \tag{3.7}$$

Here the $\lambda_1\, and\, \lambda_2$ are the cost function weights and m is the prediction horizon. The optimization problem is solved within defined constraints for the different trajectories. The cost function calculates the cost of the optimal control problem using the running and the terminal cost over the prediction horizon. In this research fmincon function was used to solve constrained optimization problem and we get the new states from the Euler integration method. Fmincon finds the minimum of constrained nonlinear multivariate function. It has active set method, interior point method and trust region reflective method for optimization. In our study we used the interior point method for optimization. Equation (3.8) shows the state and input constraints. The states and input variable sets will be discussed in the vessel model section.

$$x_{min} \leq x \leq x_{max}; u_{min} \leq u \leq u_{max} \tag{3.8}$$

3.2.3 Vessel Model

For our simulation and experiments we have used the Magne Viking ship model. MAGNE VIKING (IMO: 9423839) is a Tug/Supply Vessel sailing under the flag of Norway. The vessel has an overall length (LOA) of 85.2 meters and a width of 22.84 meters. The full-scale vessel is shown in Figure 3-1. .

Figure 3-1: Full scale Magne Viking ship

The ship model [59] is represented as follows and 3 DoF motion is considered here:

$$\dot{\eta}(t) = R\big(\eta(t)\big)V(t) \tag{3.9}$$

$$M\dot{V}(t) + C\big(V(t)\big)V(t) + D\big(V(t)\big)V(t) = \tau \tag{3.10}$$

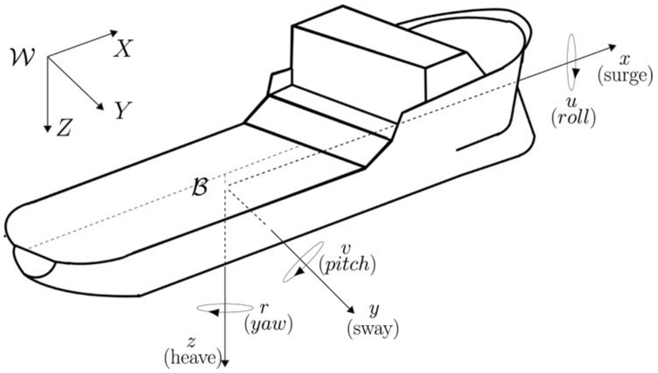


Figure 3-2: Motion Variables of marine vessel [74]

where $\eta(t) = [x(t), y(t), \psi(t)]^T$ represents the position and the orientation of the ship at time *t*, and $V(t) = \left[v_x(t), v_y(t), v_\psi(t)\right]^T$ is the speed for the vessel, τ is the vector that denotes the forces applied to the ship center of gravity. The elements of the position-orientation vector η are the north-east positions $((x(t), y(t))$ are relative to local geographical frame and the yaw rate ψ is relative to the north. The components of the velocity vectors are the surge and sway velocities and the yaw moment or the yaw rate. $v_x\ and\ v_y$ are the body fixed velocities. Similarly, the yaw rate is a component of the angular velocity of the body fixed frame with respect to local geographical frame. Due to the assumption that the translational motion is confined only to the horizontal plane, the angular velocity only has one component.

Table 3-2: Summary of ship motion variables for maneuvering applications

Variable	Name	Frame	Units
x	North Position	Earth fixed	m
y	East Position	Earth fixed	m
ψ	Heading or Yaw angle	Body fixed	rad
v_x	Surge Speed	Body fixed	m/s
v_y	Sway Speed	Body fixed	m/s
v_ψ	Yaw rate	Body fixed	rad/s
τ_x	Surge Force	Body fixed	N
τ_x	Sway Force	Body fixed	N
τ_x	Yaw moment	Body fixed	N-m
$\eta(t) = [x(t), y(t), \psi(t)]^T$	Generalized position		
$V(t) = \left[v_x(t), v_y(t), v_\psi(t)\right]^T$	Generalized velocity		
$\tau = \left[\tau_x, \tau_y, \tau_\psi\right]^T$	Generalized force		

Here the M represents the inertial Mass Matrix consisting of two components Rigid Body Matrix (M_{RB}) and Added Mass Matrix (M_A).

$$M = M_{RB} + M_A \tag{3.11}$$

Where

$$M_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix}; \qquad M_A = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix} \qquad (3.12)$$

Here m is the mass of the ship, x_g represents the distance from the vessel's center of gravity to the center of the body-fixed coordinate frame.

C represents matrices for Coriolis and Centrifugal effects, incorporating both rigid-body components and additional contributions from Coriolis and centripetal forces.

$$C(V) = C_{RB}(V) + C_A(V) \qquad (3.13)$$

Where,

$$C_{RB}\,(V) = \begin{bmatrix} 0 & 0 & -m(x_g v_\psi + v_y) \\ 0 & 0 & mv_x \\ m(x_g v_\psi + v_y) & -mv_x & 0 \end{bmatrix} \qquad (3.13)$$

$$C_A(V) = \begin{bmatrix} 0 & 0 & c_{13}(V) \\ 0 & 0 & c_{23}(V) \\ -c_{13}(V) & -c_{23}(V) & 0 \end{bmatrix},$$

Here $c_{13}(V) = Y_{\dot{v}} v_x + \frac{1}{2}(N_{\dot{V}} + Y_{\dot{r}})$ and $c_{23}(V) = -X_{\dot{u}} v_x$

The Damping matrix is formed by combining two matrices, one linear and the other nonlinear.

$$D(V) = D_L + D_{NL}(V)$$

Where,

$$D_L = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix}$$

$$D_{NL}(V) = \begin{bmatrix} -d_{11}(V) & 0 & 0 \\ 0 & -d_{22}(V) & -d_{23}(V) \\ 0 & -d_{32}(V) & -d_{33}(V) \end{bmatrix}$$

With $d_{11}(V) = X_{|u|u}|v_x| + X_{uuu}v_x^2$, $d_{22}(V) = Y_{|v|v} \left|v_y\right| + Y_{r|v|} \left|v_\psi\right|$,

$d_{23}(V) = Y_{|v|r}|v_Y| + Y_{r|r|}\left|v_\psi\right|$, $d_{32}(V) = N_{v|v|}\left|v_y\right| + N_{r|v|}\left|v_\psi\right|$ and

$$d_{33}(V) = N_{v|r|}\left|v_y\right| + N_{r|r|} \left|v_\psi\right|$$

More details on models and parameters can be found in the literatures [54] and [58].

The matrix $R(\eta)$ is a rotation matrix responsible for converting the vessel's velocity from body-fixed coordinates to inertial velocities. It is defined as:

$$R(\eta) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.14}$$

The ψ is the ship heading and τ is the vector of forces applied to the ship.

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_\psi \end{bmatrix} \tag{3.15}$$

Where τ_x and τ_y are the surge and sway forces and τ_ψ is the yaw moment.

The proposed controller's performance was evaluated using mathematical model and physical model of the Magne Viking ship. The physical model is a 1:14.7 scale replica of the Magne Viking. The Figure 3-3 below shows the Magne Viking ship model used for the trajectory tracking experiment in NRC testing facility. Essential vessel parameters as reported in Table 3-2 were obtained through various experiments conducted by NRC for system identification [74].These identified parameters form the basis for constructing an accurate mechanistic model essential for simulating the ship's behavior and conducting trajectory tracking experiments.

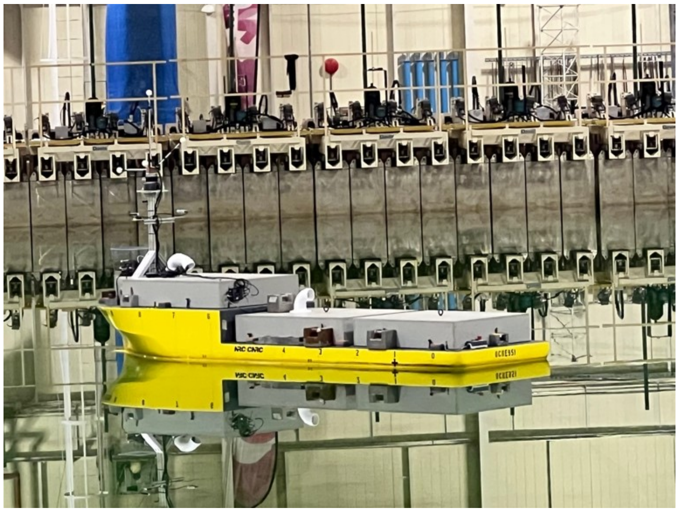


Figure 3-3: Magne Viking model ship (NRC, St. John's, NL, Canada)

Table 3-3: Magne Viking Vessel Identified Parameters

Parameters	Value	Parameters	Value
m	1290.45	$Y_{\|r\|v}$	0
x_g	0.0184	$N_{\|r\|v}$	0
I_z	1.57765x10^3	N_r	26
$X_{\dot{u}}$	158.23	$Y_{\|v\|r}$	0
$Y_{\dot{v}}$	1087.3	$N_{\|v\|r}$	0
$Y_{\dot{r}}$	0	$Y_{\|r\|r}$	0
$N_{\dot{v}}$	0	$N_{\|r\|r}$	3913.5
$N_{\dot{r}}$	$1.55992x10^3$	N_v	0
$X_{\|u\|u}$	194	X_{uuu}	-112
$Y_{\|v\|v}$	0	X_u	41.788
$N_{\|v\|v}$	0	Y_v	336.055

Y_r	0		

Here the hydrodynamic coefficients $\{X_{(.)},\ Y_{(.),}\ N_{(.)}\}$ are refer to as hydrodynamic derivatives. These are the partial derivatives of forces moments with respect to corresponding velocities.

3.2.4 Disturbance Model

This section discusses how to control the path of marine surface vessels when facing unpredictable changes in the environment. To address the issue of unknown disturbances, an observer is created to estimate these disturbances. This estimation is then used to design a new and strong controller for accurately following a desired trajectory. The disturbances induced by the wind, waves and currents are main obstacles for the ship to be remain in its course. From equation (3.10), a model for vessel dynamics can be expressed as

$$M\dot{V}(t) + C\big(V(t)\big)V(t) + D\big(V(t)\big)V(t) + d(V_{rc}\gamma_c) = \tau_{control} + \tau_{wind} + \tau_{waves} \qquad (3.16)$$

The term $d(V_{rc}\gamma_c)$ represents the current forces which reflects the transfer of the energy from the vessel to the water. It depends on the speed and direction of the current with respect to the vessel. Here,

$$V_{rc} = \sqrt{{v_x}_{rc}^{2} + {v_y}_{rc}^{2}} = \sqrt{\left(v_x - v_{x_c}\right)^2 + \left(v_y - v_{y_c}\right)^2} \qquad (3.17)$$

$$\gamma_{rc} = -atan2(v_{y_{rc}}, v_{x_{rc}}) \qquad (3.18)$$

The v_{x_c} and v_{y_c} are the current velocity measured in the vessel body fixed frame and the angle of the current γ_{rc} is measured relative to the bow of the vessel. It is very common to express the current forces as a function of non-dimensional current coefficients in the directions of surge, sway, and yaw, which is expressed as –

$$d(V_{rc}\gamma_c) = \frac{1}{2}\rho V_{rc}^2 \begin{bmatrix} A_{Fc} & C_{Xc}(\gamma_{rc}) \\ A_{Lc} & C_{Yc}(\gamma_{rc}) \\ A_{Lc}L_{oa} & C_{Nc}(\gamma_{rc}) \end{bmatrix} \tag{3.19}$$

Here ρ is the water density. The formula includes variables like frontal area (A_{Fc}), lateral area (A_{Lc}) of the underwater part of the hull, and ship length (L_{oa}). For dynamically positioned vessels, typical current coefficients are determined through experiments or computational fluid dynamic models [60]. Estimating current coefficients (C_{Xc}, C_{yc}, C_{Nc}) accurately is challenging unless there's an extensive hydrodynamic analysis and scale-model testing for the specific vessel. In such cases, it's common to simplify the model (as in equation 3.19) by using a linear damping term and a bias term [61] and can be expressed as-

$$d(V_{rc}\gamma_c) \approx Dv - R^T(\psi)b \tag{3.20}$$

So the vessel model from (3.9) and (3.10) becomes

$$M\dot{V}(t) + C\big(V(t)\big)V(t) + D\big(V(t)\big)V(t) = R^T\big(\eta(t)\big)b + \tau_{control} + \tau_{wind} + \tau_{wave} \qquad ()$$

The bias is constant in earth fixed coordinates when it is assumed that the currents are slowly varying. Thus, to include the bias term in the equation of the ship motion (3.21) it must be rotated so that it captures the effects of the current forces change along with the heading change of the ship. The bias is estimated using the Unscented Kalman Filter (UKF).

Similarly, the wind forces can be expressed by means of nondimensional force coefficients which are shown in the following equation [58],

$$\tau_{wind} = \frac{1}{2}\rho_a V_{rw}^2 \begin{bmatrix} A_{Fw} & C_{Xw}(\gamma_{rw}) \\ A_{Lw} & C_{Yw}(\gamma_{rw}) \\ A_{Lw}L_{oa} & C_{Nw}(\gamma_{rw}) \end{bmatrix} \tag{3.21}$$

Here, ρ_a is the air density.

States and Optimization Variable Weights A_{F_w} and A_{L_w} are the frontal and lateral projected wind areas respectively. The L_{o_a} is the ship length. The wind speed and directions V_{rw} and γ_{rw} with respect to the vessel is given by the following equations where β_w is the wind direction in earth fixed co ordinates.

$$V_{rc} = \sqrt{v_{x_{rw}}^2 + v_{y_{rw}}^2} \tag{3.22}$$

$$\gamma_{rw} = -atan2(v_{y_{rw}}, v_{x_{rw}}) \tag{3.23}$$

$$v_{x_{rw}} = v_x - V_w cos\beta_w \tag{3.24}$$

$$v_{y_{rw}} = v_y - V_w sin\beta_w \tag{3.25}$$

However, the wind coefficients in (3.21) can be determined from the computational fluid dynamics or the model testing or from the scaling coefficients from the similar vessel. But from the control design perspective the wind speed and the direction are generally used for approximate feedforward compensation. The errors associated because of this compensation are modelled in the bias term in (3.21). Therefore, the bias takes into account the simplified current as well as the wind forces.

The wave forces are modelled as a sum of nonlinear and linear wave components.

$$\boldsymbol{\tau_{wave}} = \tau_{wave}^{lin} + \tau_{wave}^{nlin} \tag{3.26}$$

The linear and the low frequency nonlinear components are very much relevant to the ships motion control. The high-frequency nonlinear wave forces are generally considered as input disturbances and modeled as bias term. So, the bias represents combination of nonlinear wave drift and current. However, the linear wave forces usually behave like an equivalent output disturbance. In this study environmental forces waves, wind, and currents are external factors that can disrupt the motion-control system of a vessel. These forces are categorized in wave and low-frequency components. Waves create pressure changes on the hull, generating forces with both oscillatory (wave-frequency) and nonlinear components. The oscillatory forces align with the wave frequency, while nonlinear forces result from the quadratic relationship between pressure and fluid-particle velocity induced by wave motion. Nonlinear forces encompass frequencies both lower and higher than the waves. Mean wave forces induce vessel drift, and forces at the difference of wave frequencies may lead to resonance in vessel motion [58].

3.2.5 Unscented Kalman Filter

The Unscented Kalman Filter (UKF) employs a deterministic sampling method to calculate mean and covariance estimates using a minimal set of sigma points. In comparison to the Extended Kalman Filter (EKF), the UKF demonstrates greater effectiveness in handling nonlinear systems across diverse applications like railways, ships, aircraft, solar probes, and other fields [62]. The UKF involves the prediction and update steps. The steps for filtering the waves and to estimate the states from the UKF are described in this table below.

Table: 3-4 UKF algorithm for state estimation

Step	Description
Initialization	Set initial state estimate x$_0$ and covariance P$_0$. Set process noise covariance Q$_0$ and measurement noise covariance R$_0$.
Prediction Step	Generate Sigma Points: $(\chi_{k-1})_0 = \hat{x}_{k-1}$ $(\chi_{k-1})_i = \hat{x}_{k-1} + \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i$ $(\chi_{k-1})_i = \hat{x}_{k-1} - \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i,$ i = 1, …., n
	Propagate Sigma Points through Dynamic Model: $(\hat{\chi}_k)_i = f((\chi_{k-1})_i), i = 0, \dots\dots, 2n$
	Calculate Predicted Mean: $\hat{x}_k{}^- = \sum_{i=0}^{2n} W_i^m (\hat{\chi}_k)_i$
	Calculate Predicted Covariance: $P_k = \sum_{i=0}^{2n} W_i^c ((\hat{\chi}_k)_i - \hat{x}_k{}^-)((\hat{\chi}_k)_i - \hat{x}_k{}^-)^T + Q_{k-1}$
Update Step	Generate Sigma points: $(\chi_{k-1})_0 = \hat{x}_{k-1}$
	$(\chi_{k-1})_i = \hat{x}_{k-1} + \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i$
	$(\chi_{k-1})_i = \hat{x}_{k-1} - \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i,$ i = 1, …., n
	Calculate Predicted Measurement Mean:

	$$\mu_k = \sum_{i=0}^{2n} W_i^m \left(\hat{y}_k^i - \mu_k\right)\left(\hat{y}_k^i - \mu_k\right)^T + R_k$$
	Calculate Predicted Measurement Covariance: $$S_k = \sum_{i=0}^{2n} W_i^c \left(\hat{y}_k^i - \mu_k\right)\left(\hat{y}_k^i - \mu_k\right)^T + R_k$$
	Calculate Cross-Covariance: $$C_k = \sum_{i=0}^{2n} W_i^c ((\chi_k^-)_i - \hat{x}_k{}^-)((\chi_k^-)_i - \hat{x}_k{}^-)^T$$
	Calculate Kalman Gain: $$K_k = C_k S_k^{-1}$$
	Update State Estimate: $$x_k = x_k^- + K_k(y_k - \mu_k)$$
	Update Covariance: $$P_k = P_k^- - K_k S_k K_k^T$$
Iterate	Repeat Prediction and Update steps for each time step.

Where, $\lambda = \alpha^2(n + k) - n$ is a scaling parameter and the parameters $\alpha \: and \: k$ define the distribution or extent of the sigma points around the mean. In this case, the λ is chosen as 3. The covariances are determined as follows:

P_k is chosen as a 15x15 identity matrix as the number of states for UKF was 15 mentioned in equation 3.37. For determining Q_k for different matrices were chosen. They are:

Q1 = 1*diag ([1,1,1,10,10,10])

Q2 = 0.001*eye (3)

Q3 = 0.01*diag([40,50,40])

Q4 = 0.005*diag([10,10,10])

And Q_k is block diagonal matrix consisting of Q1, Q2, Q3 and Q4.

And the is chosen as $R_k = \begin{bmatrix} 0.005 & 0 & 0 \\ 0 & 0.005 & 0 \\ 0 & 0 & 0.0005 \end{bmatrix}$

The matrices were selected through a tuning process based on sample data to achieve the desired response of the system. Q1 is chosen as a diagonal matrix with relatively high values (e.g., 1) on the diagonal. This indicates a higher uncertainty in the position estimates due to factors like sensor noise or external disturbances. Q2 is a small diagonal matrix (e.g., 0.001*eye (3)) with a non-zero value only in the element corresponding to yaw (ψ). This suggests a relatively low uncertainty in the orientation measurement compared to the position. Q3 (with higher diagonal elements compared to Q2) and Q4 (with lower diagonal elements) represent the uncertainties in the velocity estimates.

For the estimation of the vessel's state using the UKF the state vector consists of the waves, positions, orientations velocities and the biases. The states vector x_s represented as follows:

$$x_s(t) = \left[F_{w1}, F_{w2}, F_{w3}, F_{w4}, F_{w5}, F_{w6}, x, y, \psi, v_x, v_y, v_\psi, b_1, b_2, b_3\right]^T \tag{3.37}$$

So,

$$\dot{\hat{x}}_s(t) = f(\hat{x}, U) + w \tag{3.38}$$

$$\hat{y}(t) = h(\hat{x}, U) + v \tag{3.39}$$

$\dot{\hat{x}}_s(t)$ and $\hat{y}(t)$ are the estimated states. In these equations w and v are respectively the process and the measurement noises. So $Q(t) = E(ww)^T$ and $R(t) = E(vv)^T$ are respectively the process-noise covariance matrix and measurement-noise covariance matrix. The UKF uses the Magne Viking Ship model, detailed in Section 3.2.3, to estimate the states. This model incorporates the dynamics of the ship, including surge, sway, yaw velocities, and the corresponding forces. By accurately representing the physical behavior of the Magne Viking, the UKF can provide precise state estimations critical for effective control and navigation.

The figures 3-4 and 3-5 below demonstrates the impact of the Unscented Kalman Filter (UKF) on trajectory tracking. A simulation using the same wave model is run with the UKF disabled (UKF-OFF) and enabled (UKF-ON). The results show significant differences in tracking performance between the two scenarios.

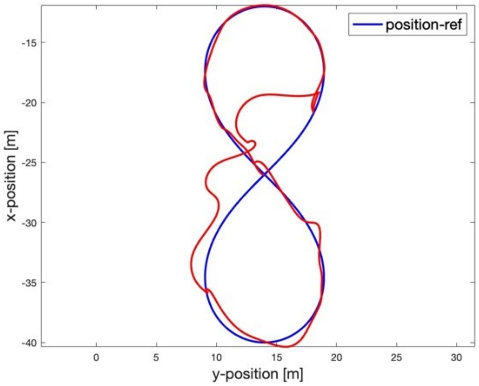


Figure 3-4: Tracking performance with UKF disabled

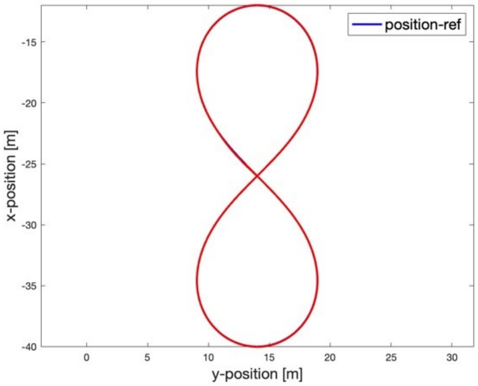


Figure 3-5: Tracking performance with UKF enabled

3.2.6 Trajectory Generation and Implementation

Various trajectories were generated using different schemes for the initial testing and validation of the controller's feasibility. These trajectories were designed to explore the controller's performance under various conditions and to assess its robustness. Key factors considered during this phase included constraints imposed on the trajectories and the discrete points along the paths.

Exploring trajectories with diverse characteristics was crucial in refining the NMPC. We gained valuable insights into adaptability and responsiveness by subjecting the controller to trajectories of varying complexities.

A significant aspect of this process involved:

- Establishing constraints on the trajectories.
- Defining specific points along the paths.
- Observing the controller's behavior in response to these variations.

The goal was to fine-tune the controller's parameters and determine optimal weights for the cost function in the NMPC.

For trajectory generation we kept the average velocity constant at 0.2 m/s which is a recommended value for the ship model we used. And for the oval shaped trajectory the surge velocity 0.2 m/s was maintained by using the proper constraints for the velocity. The heading angle was calculated from the trajectories with discretization of the equations expressing the x and y. So,

$$dx = x_2 - x_1$$

$$dy = y_2 - y_1$$

$$\text{And } \psi = atan2\frac{dy}{dx}$$

Later, for the comparison of performance of the controller with regular shaped and complex shape we adapted the circular trajectory and complex shaped figure eight trajectory which were then tested at NRC. Here's a sample trajectory that we used for the simulation and experiment.

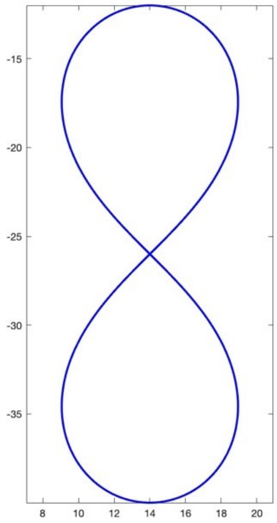


Figure 3-6: Sample Trajectory Generation

3.3 Results and Discussion

In addressing the critical challenge of precise trajectory tracking for Autonomous Surface Vessels (ASVs), this research contributes significantly to this aspect of autonomous ship operations. The complexity of maritime environments, characterized by unpredictable factors such as waves, currents, and wind, necessitates the development of a robust controller capable of navigating diverse conditions. While model-based and data-driven controllers have shown promise, existing literature reveals a persistent need for improvements, especially in respect of efficiency and improved performance in terms of accuracy tolerance. This study focuses on designing and implementing a NMPC for ASVs, leveraging its capability to accommodate nonlinear ship models, handle unmodeled dynamics, and effectively manage multivariable systems. Integrating an UKF further enhances the controller's robustness by filtering out wave-induced disturbances. The comprehensive testing of this combined NMPC controller with UKF involved three distinct models, with simulations and experimental setups conducted at the National Research Council (NRC) in Canada, utilizing the Magne Viking ship model.

In the subsequent Results and Discussion section, we present and analyze the outcomes of our experimentation, shedding light on the controller's performance under different conditions. This analysis includes examining the controller's ability to precisely follow trajectories, mitigate effect of disturbances, and adhere to safety constraints. Through this exploration, we aim to contribute valuable insights to the ongoing discourse on advancing autonomous ship technologies, emphasizing the practical applicability of the developed controller in real-world scenarios. To analyze the performance of our trajectory tracking controller we have used the circular trajectory and the figure-eight trajectory. A circular trajectory might test the model's ability to maintain a consistent path, while a figure-eight trajectory introduces more complexity and challenges the model's ability to handle changes in direction and speed. Also using two distinct trajectories enables a comprehensive comparison of the physics-based model with other models which will be discussed in the 4th chapter of this thesis. It allows us to demonstrate which model performs better under uniform motion (circular trajectory) versus a motion with varying dynamics (figure-eight trajectory).

3.3.1 Simulation Results

A circular trajectory with a 5-meter radius and a figure-eight shape trajectory with a 7-meter radius were employed for the ship's navigation. The figure-eight trajectory, being more complex, better simulates real-world navigation scenarios where a ship must make frequent course adjustments. This complexity helps demonstrate the robustness and adaptability of the control models under study. The circular trajectory is more suited for steady state tracking performance, while the figure-eight could be more suited for evaluating transient response and precision in more dynamic conditions. For this simulation part we have used the physics based mode described in Section 3.2 as the system. In addition to the ship kinematics this model includes the external forces like the waves. Below the tracking performance of the controller under the effect of the wave forces is shown. We have used DM to express the Dynamic Model in the figures. DM5, DM10 and DM20 respectively shows the performance of the controller with prediction horizons 5, 10 and 20.

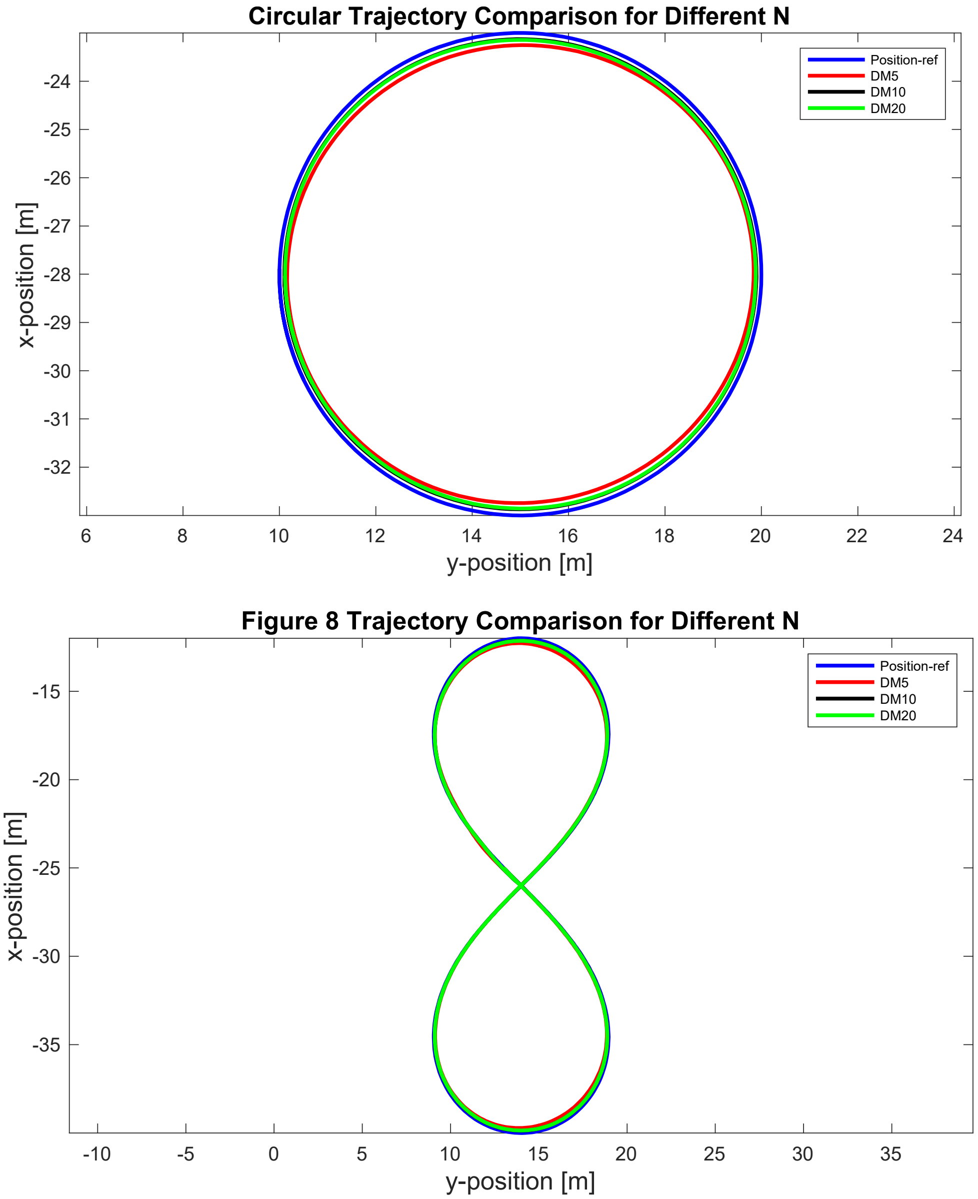


Figure 3-7: Trajectory Tracking of two different trajectories in the presence of disturbances.

The wave disturbance that is used in the simulation for all the trajectories are shown in Figure 3-8

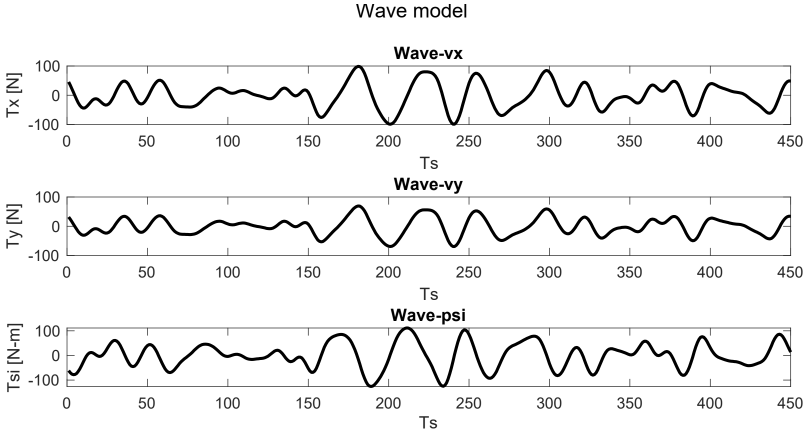


Figure 3-8: Wave model used for simulating disturbance

The wave induced forces plays a significant role in affecting the velocities of an autonomous ship during trajectory tracking. These disturbances can introduce variations and uncertainties that the ship's control system must adapt to maintain the desired trajectory and velocity.

3.3.1.1 Trajectory Tracking Analysis for Simulation Results

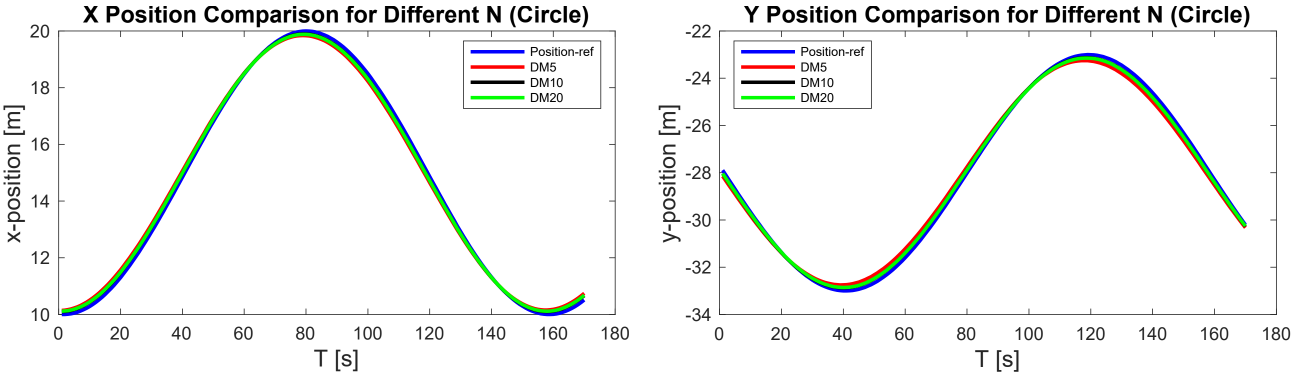


Figure 3-9: Comparison of X and Y positions for the circular trajectory

For both circular and figure-eight trajectory the position error is lowest at a prediction horizon of 10 and slightly increases at 20 (Figure 3-9 & Figure 3-13). This suggests that the control models require a moderate prediction horizon to optimize position accuracy, especially in more complex trajectories like the figure-eight. The position error doesn’t follow a simple increasing or decreasing trend with the prediction horizon for both trajectories. This suggests that the relationship between prediction horizon and position error is non-linear and may depend on other factors like model configuration or external disturbances. A shorter prediction horizon (N = 5) might not provide sufficient time for the model to anticipate and adjust for upcoming changes in the trajectory, while a longer horizon (N = 20) slows down the controller and could introduce too much uncertainty, leading to less accurate position control.

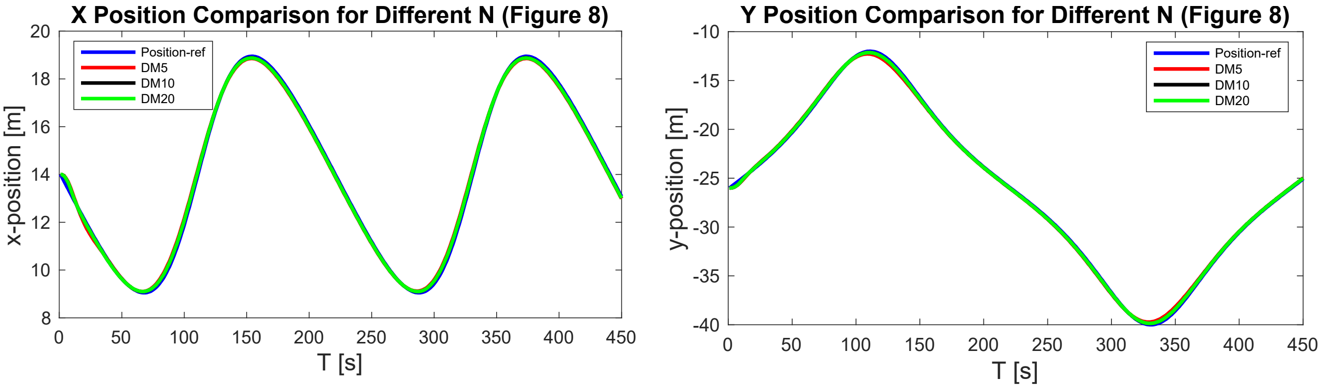
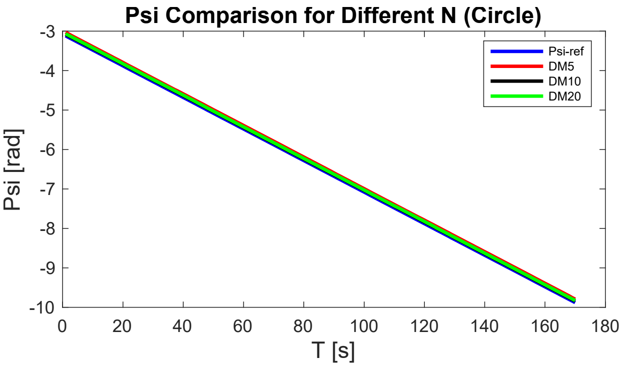


Figure 3-10: Comparison of X and Y positions for the Figure-eight trajectory



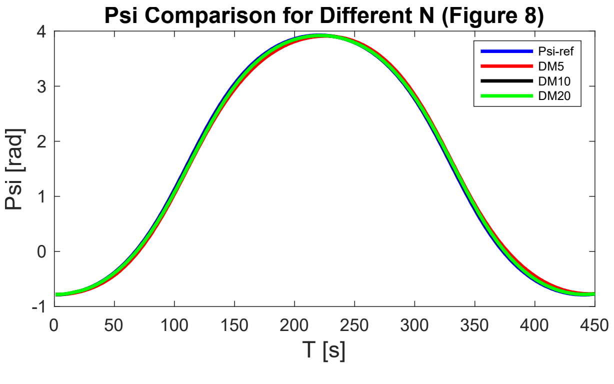


Figure 3-11: Comparison of heading angles for the circular trajectory

Figure 3-12: Comparison of heading angles for the Figure-eight trajectory

From the Figures 3-8 and 3-9 it is observed that the heading angle error being lowest at N = 5 and highest at N = 20 indicates that the control models are more effective at maintaining the desired heading in the short term. This could be due to the immediate responsiveness required for heading control, which is better addressed with a shorter prediction horizon. At N = 20, the increased heading angle error could result from the models' difficulty in accurately predicting and adjusting for the heading far into the future, where uncertainties and external disturbances (such as wave forces) have a more significant impact. The Figures 3-11 and 3-12 show a comparison of the heading angle tracking for all the three prediction horizons for the two trajectories.

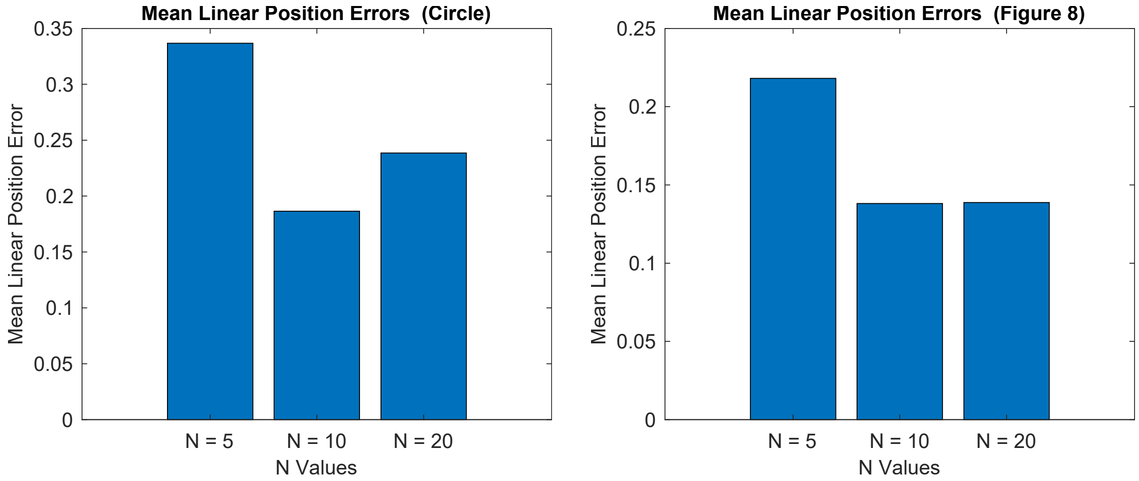
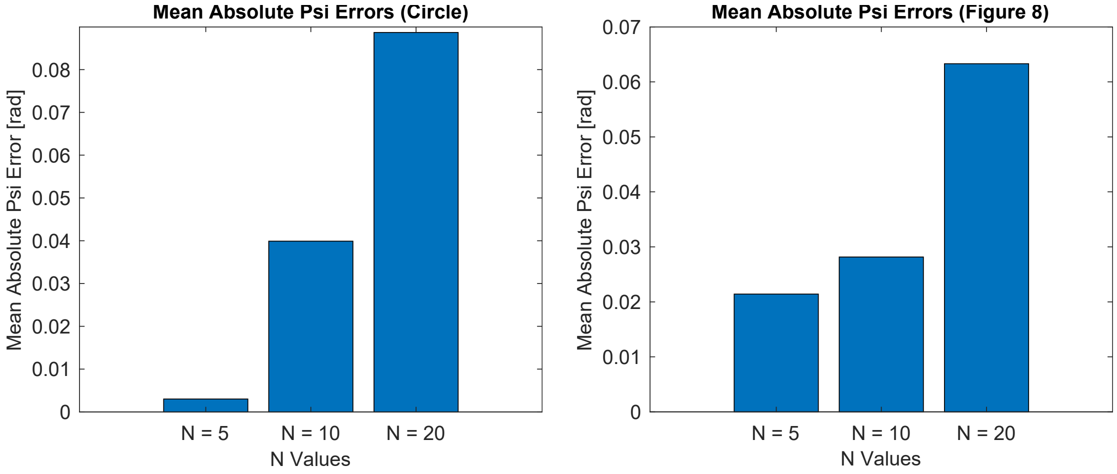


Figure 3-13: The error for positions the different prediction horizons for circular and figure eight trajectory.

Over a longer horizon, even minor modeling errors or deviations from the predicted path can accumulate, leading to a significant overall error. This is particularly true for complex trajectories, where small heading or position errors can quickly lead to a substantial deviation from the intended path. From the results produced it is apparent that the controller can allow accurate trajectory tracking even in the presence of the disturbance like wave forces. The deviation from the desired trajectory for the position and orientation for all the prediction horizon is summarized in table 3-6 which shows that lowest error for position and heading angle is 0.1381 meters and 0.0214 radians respectively. The Figure 3-14 shows that shorter prediction horizon has lowest heading error. With the shorter prediction horizon there is less accumulation of model inaccuracies and disturbances. In shorter horizons, the predictions are more accurate as they are less susceptible to the compounding effects of dynamic uncertainties and external factors. Additionally, the control algorithm can make more frequent corrections, maintaining closer adherence to the desired trajectory and heading.


Mean Absolute Psi Errors (Circle)

Mean Absolute Psi Errors (Figure 8)

Figure 3-14: The heading error for different prediction horizons for circular and figure eight trajectory.

The simulation studies for the Magne Viking ship were carried out for a constant speed of 0.2 m/s. The analysis of the velocity error shows that velocity distribution for both the trajectories remain within the limits for the given disturbance model in the form of wave affecting the surge, sway, and yaw velocities. The disturbance effects on the ship's velocities are shown in (Figure 3-15) for the two trajectories studied (circle and figure-eight) and correlating them with the error values for velocities (Figure 3-16).

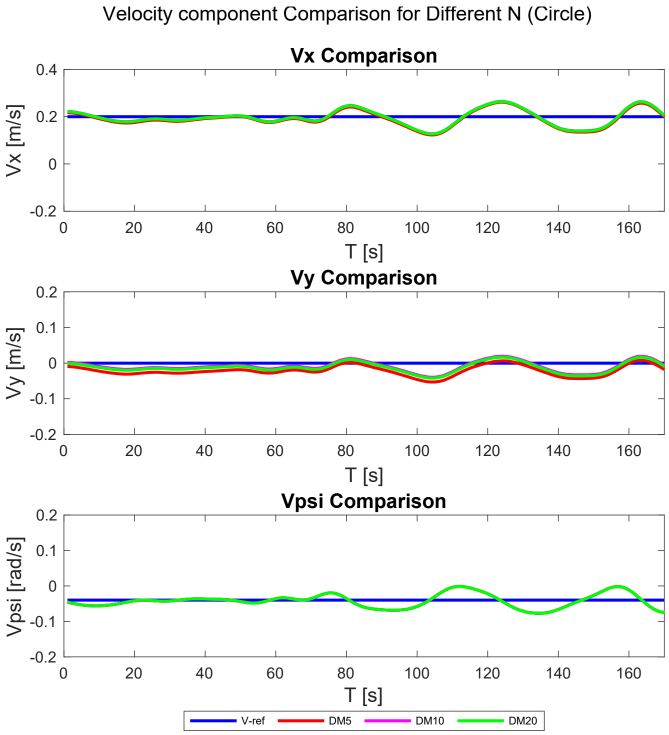


Figure 3-15(A): Comparison of velocity components for the circular trajectory for different N.

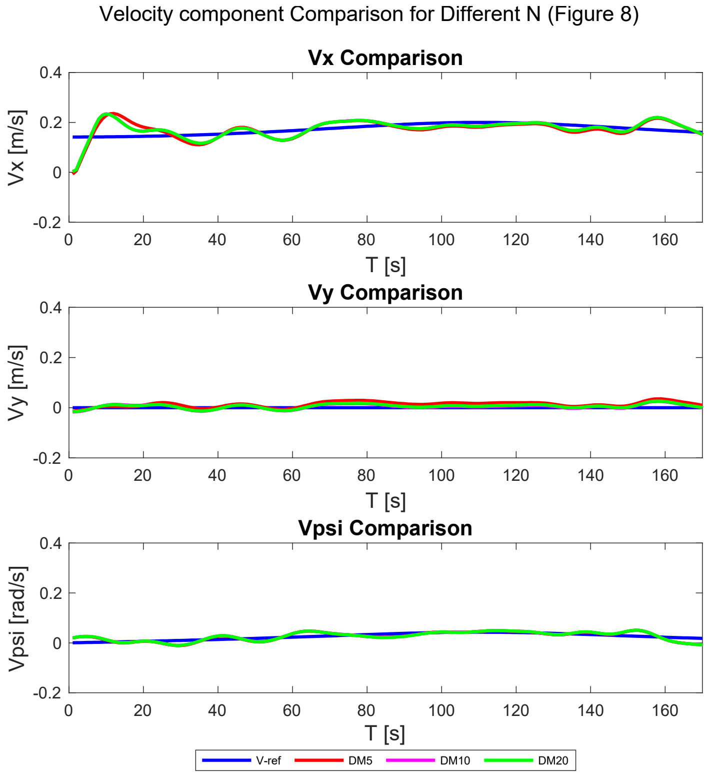


Figure 3-15(B): Comparison of velocity components for the Figure-eight trajectory for different N.

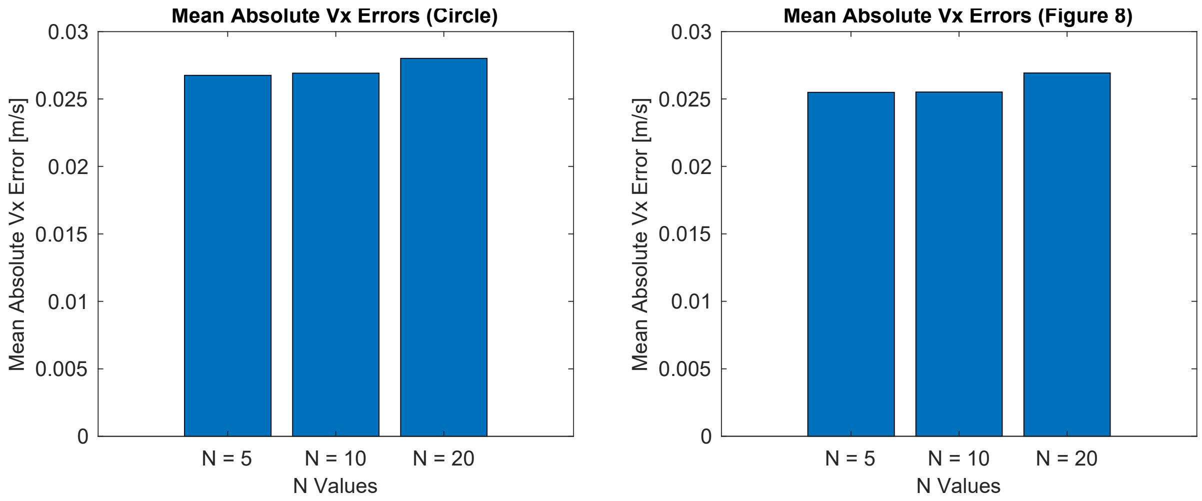


Figure 3-16A: The error for surge velocities for the different prediction horizons for circular and figure eight trajectory.

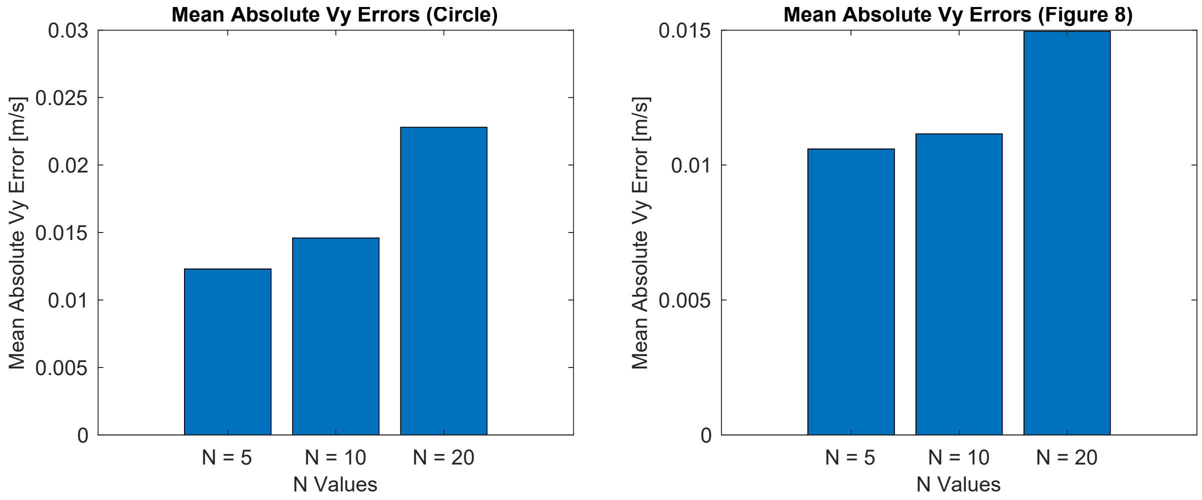


Figure 3-16B: The error for sway velocities for the different prediction horizons for circular and figure eight trajectory.

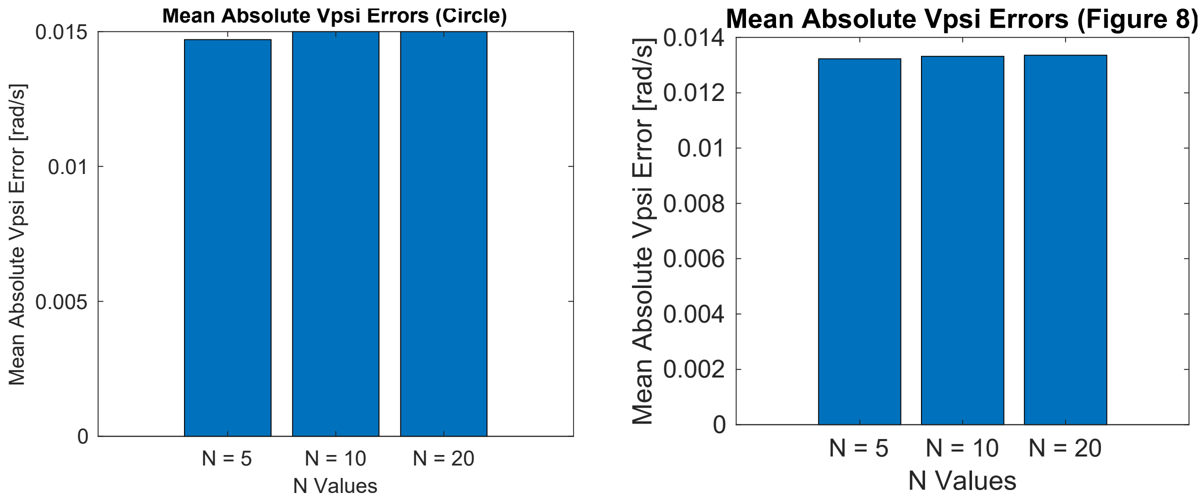


Figure 3-16C: The error for yaw rates for the different prediction horizons for circular and figure eight trajectory.

For all prediction horizons (N = 5, 10, 20), the actual velocities maintained by the NMPC seem to closely follow the reference, suggesting good performance in maintaining the desired surge velocity despite the wave disturbances (Figure 3-15). The wave disturbances, as modeled by the disturbance model, would typically cause fluctuations in the surge velocity due to the forces exerted on the ship. However, the close tracking performance indicates that the NMPC controllers are effectively compensating for these disturbances. The prediction horizon's impact on the velocity control can be assessed by looking at the tightness of the tracking around the reference velocity. When comparing the figure-eight to the circular trajectory, one would expect more variability in the velocity for the figure-eight due to its more complex shape. However, the consistent tracking across the figure-eight suggests that the NMPC model with UKF is adequately accounting for this complexity. The surge velocity control demonstrated here suggests that the NMPC model is robust against the wave disturbances simulated in this scenario. For practical applications, it would be important to validate this robustness across a wider range of conditions, including varying wave intensities and directions. The error for the velocities for the two trajectories shows the variation of error with the different prediction horizons. But the errors are significantly low in this case as shown in Table 3-5 and 3-6. For circular trajectory maximum surge, sway and yaw velocity errors are 0.028 m/s, 0.0228 m/s and 0.015 rad/s respectively. For eight shaped trajectory maximum surge, sway and yaw velocity errors values are 0.02269 m/s, 0.015 m/s and 0.0134 rad/s respectively. These low error values suggest the high precision of the control algorithm. The minimal errors could be attributed to the effectiveness of the UKF in accurately estimating the ship's states and the robust design of the control algorithm, which may efficiently compensate for model inaccuracies and external disturbances. However, these possibilities need to be explored further to confirm their impact on error reduction.

For the simulation, all the tested prediction horizons showed low errors, indicating effective performance. However, future work could explore the impact of using shorter or longer prediction horizons to determine if further improvements in accuracy can be achieved.

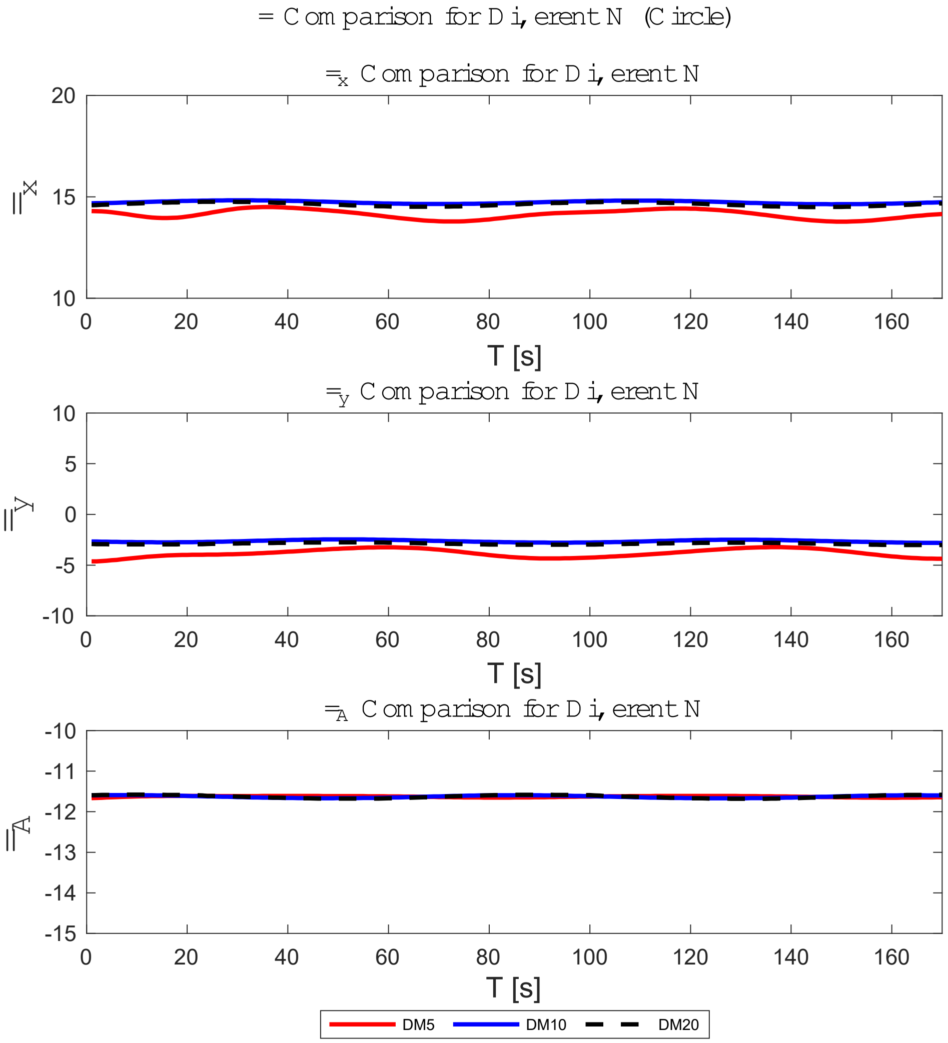


Figure 3-17A: Comparison of Forces for the circular trajectory for different N.

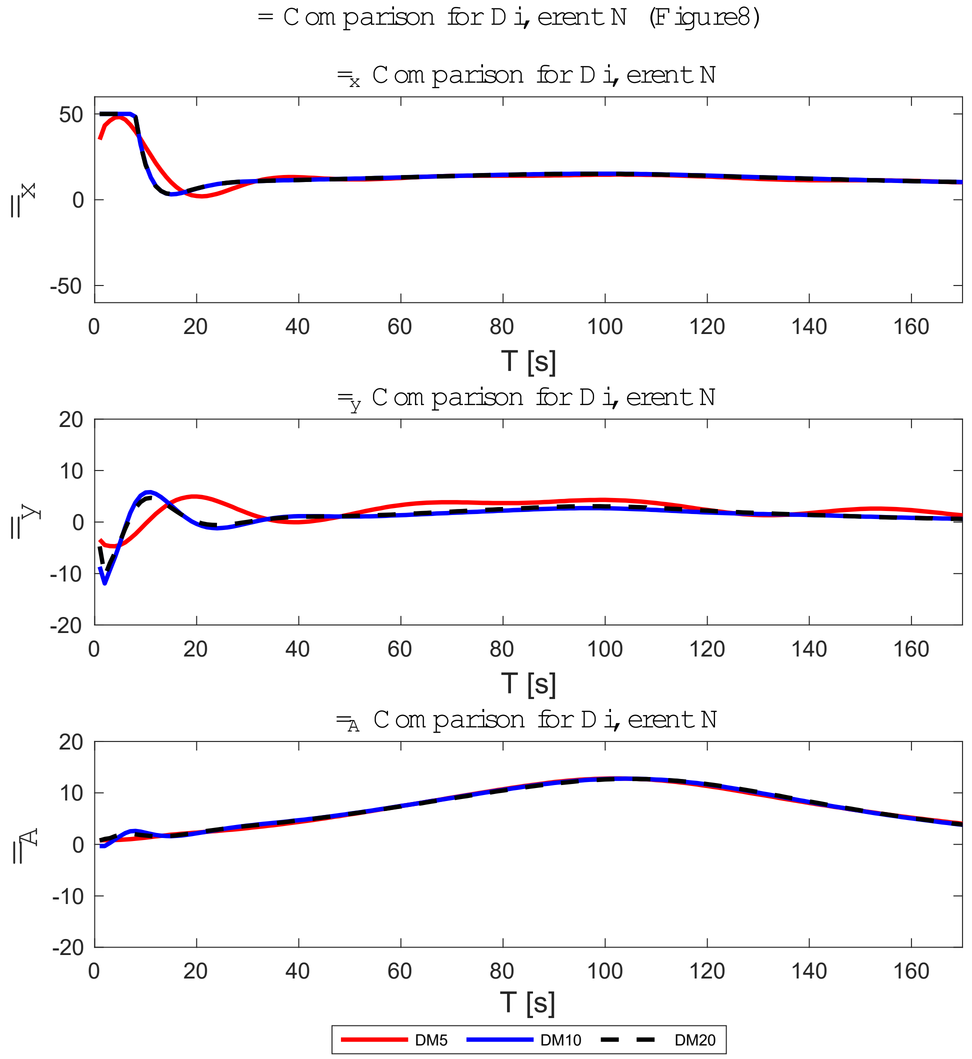


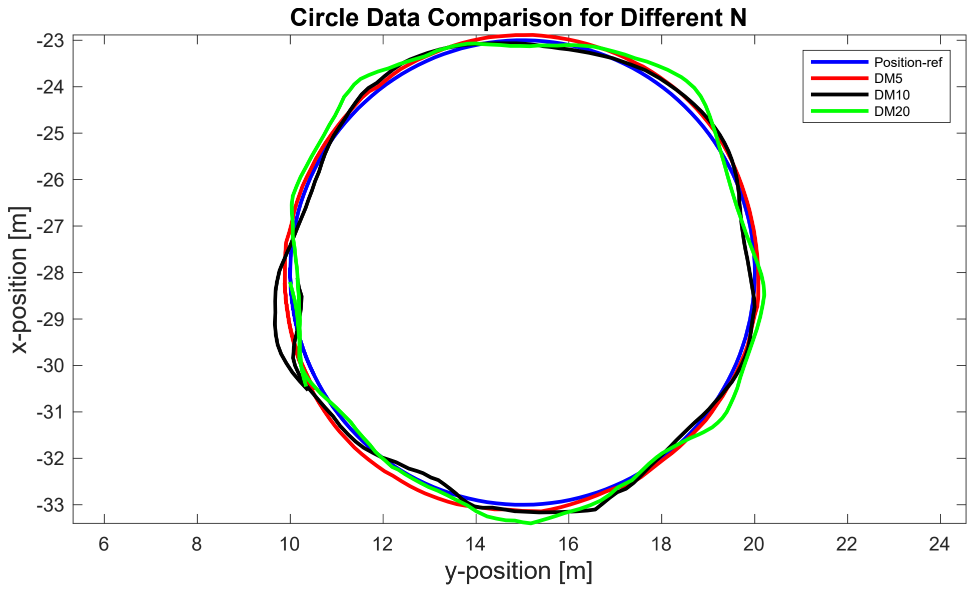
Figure 3-17B: Comparison of Forces for the Figure 8 trajectory for different N.

The forces applied by the thrusters are shown in Figure 3-17. For both trajectories, the charts show the corrective forces and moments that are needed over time (T) for different prediction horizons (N = 5, 10, 20). These forces directly respond to the disturbances and the ship's dynamics.

In the circular trajectory, the forces are relatively stable and consistent across different prediction horizons, indicating that the NMPC controller effectively counters the disturbances with minimal variation in control effort. Conversely, for the figure-eight trajectory, more fluctuation in control forces, especially in τ_x, which indicates the NMPC controller is adapting to the more complex trajectory requirements.

3.3.2 Experimental Results

This section presents the experimental findings from the tests done to understand the effectiveness of the controller developed.


Circle Data Comparison for Different N

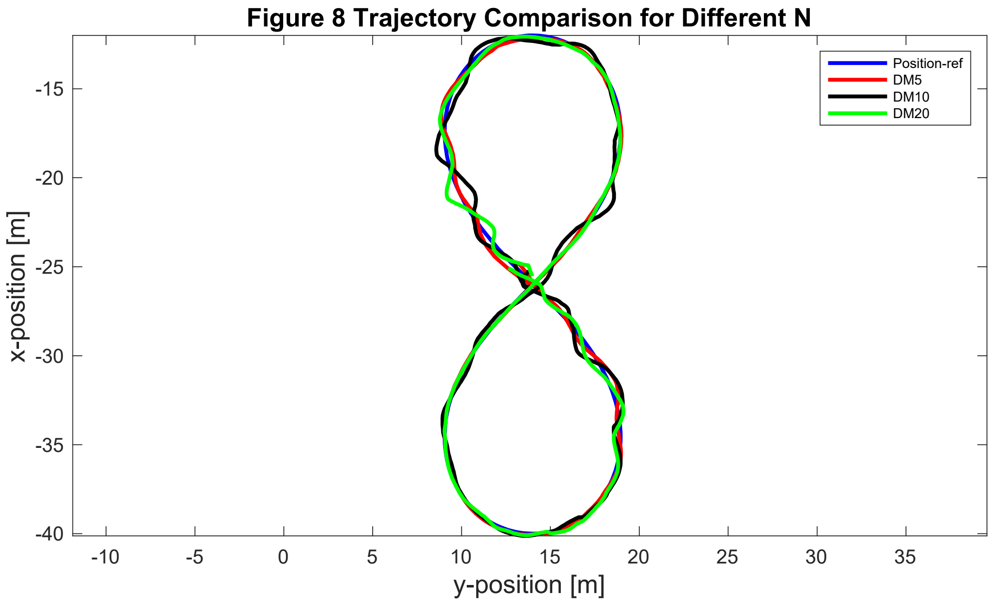


Figure 8 Trajectory Comparison for Different N

Figure 3-18: Trajectory Tracking of two different trajectories in the testing bay (NRC, Canada)

To validate the accuracy of the NMPC in trajectory tracking, experimental tests were conducted at the NRCs wave tank. Similar to the simulations, we examined the controller's performance through two trajectories: a circular path and a more intricate figure-eight shape. Figure 3-18 shows the performance of the controller for the trajectory tracking of the Magne Viking in the testing facility. These tests offered valuable insights into the controller's ability to adjust to different trajectory patterns and adapt to changes in the desired paths. The comparison of two-trajectory tracking performances, as illustrated in the figure 3-18, reveals a noticeable difference between simulation results and real-life testing. While the simulations showed more precise tracking, the actual implementation of the model exhibited deviations, underscoring the disparities between simulated predictions and real-world outcomes.

The analysis clearly shows a divergence between the simulated and experimental results. Factors that seemed insignificant during the simulations emerged as impactful during the actual experiments, significantly influencing the outcomes.

Comparison of simulation and experimental results

There are significant deviations from the projected path which is visible in Figure 3-19 and Figure 3-20. The error analysis is also presented in Table 3-5 and Table 3-6 which shows the clear comparisons between the simulation and the experimental values. A comparison of the errors for the experimental results are also presented in the Figures 3-23. The reasons for the deviation from the simulated results can be conglomerated. NMPC highly relies on model-based prediction. Simulation relies on mathematical models to represent the real-world environment. It is only possible to capture some of the complex dynamics for the model; hence, some dynamics are omitted, resulting in differences when the controller is tested in real environments.

Environment variability also affects the performance. However, the real-world environment cannot be fully captured in the simulations. One of the most critical factors considered for the deviation is the sensor and actuator imperfections. The testing facility uses the Qualisys motion capture system. Even though this system is known for its high accuracy and precision, different tests may require calibration for different applications each time. The calibration of this equipment is not easy and requires a lot of work and time. Testing a controller within a time slot and calibrating the equipment with high precision was challenging. For this type of optical tracking technology, the calibration and positioning of the camera are essential. Tuning the tracking system to achieve uniform coverage across the entire basin is challenging and not always feasible. As a result, certain orientations of the model and areas within the tank may exhibit lower tracking performance or occasional dropouts, which can significantly impact feedback and overall tracking performance.

The system operates primarily in a digital mode. Noise is minimal, and mechanical issues have not posed significant problems. However, inherent lag and delays arising from multiple PC-based software programs within the control loop represent a more substantial source of unmodeled error compared to the simulation environment. The OPC communication with the ship was achieved through the Wi-Fi system from window PC. The wireless connectivity always has some lag and interference from other devices. But delays caused by software timing, both within the Windows operating system and the control software, are more significant than delays due to WIFI.

The real-world setup's scale and complexity are unmatchable with the simulations. Finally, the idealized version of the simulation differs from the testing environment, where wear and tear, manufacturing tolerances and flexibility, human interaction, and intervention significantly influence.

A comparison of the error for the simulation and the experimental results for the different parameters significant for the trajectory control is presented in Table 3-5 and 3-6.

Table 3-5: Error comparisons for simulation vs experimental results for the Circular Trajectory

Parameters	**Simulation**			**Experiment**		
Prediction Horizon, N	5	10	20	5	10	20
Sum of root mean squared position error, e_{xy} (m)	0.3367	**0.1865**	0.2386	0.28	**0.2548**	0.3152
Heading Angle error, e_{ψ} (rad)	**0.003**	0.0399	0.0883	**0.005**	0.036	0.0691
Surge velocity error, e_{v_x} (m/s)	**0.0266**	0.0267	0.028	**0.0221**	0.0275	0.0318
Sway Velocity error, e_{v_y} (m/s)	**0.0123**	0.0146	0.0228	**0.0228**	0.0382	0.0428
Yaw rate error, $e_{v_{\psi}}$(rad/s)	**0.0147**	0.015	0.015	**0.002**	0.0164	0.0307
Average Execution Time per iteration,T_s(s)	**0.1204**	0.3348	0.8552	**0.2182**	0.4985	0.8272

Table 3-6: Error comparisons for simulation vs experimental results for the Figure 8 Trajectory

Parameters	**Simulation**			**Experiment**		
Prediction Horizon, N	5	10	20	5	10	20
Sum of root mean squared position error, e_{xy} (m)	0.2181	**0.1381**	0.1387	**0.2248**	0.3005	0.1862
Heading Angle error, e_{ψ} (rad)	**0.0214**	0.0282	0.0633	**0.0532**	0.0601	0.0813
Surge velocity error, e_{v_x} (m/s)	**0.0255**	0.0255	0.0269	**0.0214**	0.0257	0.0274

Sway Velocity error, e_{v_y} (m/s)	**0.0106**	0.0112	0.015	**0.0219**	0.0303	0.0322
Yaw rate error, e_{v_ψ}(rad/s)	**0.0132**	0.0133	0.0134	**0.0079**	0.0111	0.0146
Average Execution Time per iteration,T_s(s)	**0.1019**	0.3862	0.8418	**0.0918**	0.3899	0.8339

. In the experimental analysis, both the circular and figure-eight trajectories showed improved performance at a lower prediction horizon of N = 5. Position error deviations for the circular path were measured at 0.28 m, 0.2548 m, and 0.3152 m for prediction horizons of 5, 10, and 20, respectively. The figure-eight trajectory, which is more complex, had position error deviations of 0.2248 m, 0.3005 m, and 0.1862 m for the same respective prediction horizons.

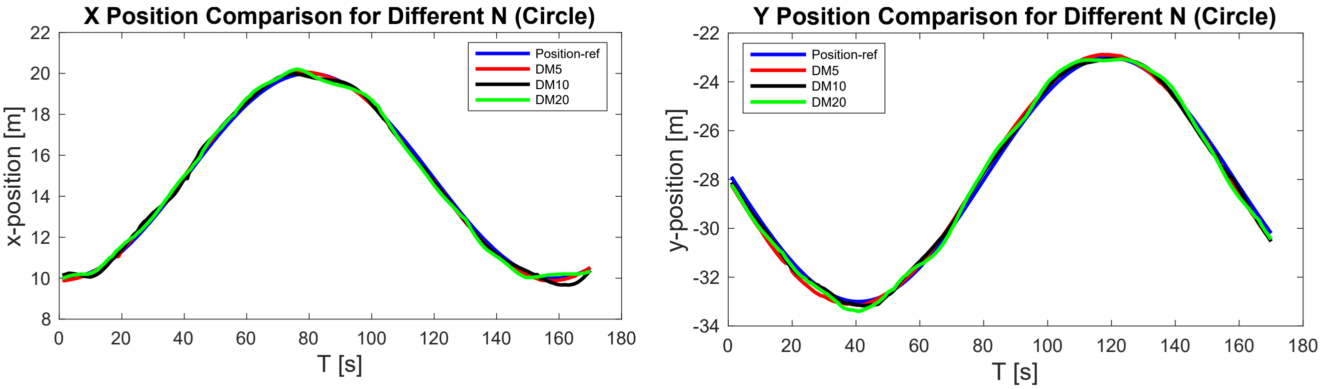


Figure 3-19: Comparison of X and Y positions for the circular trajectory

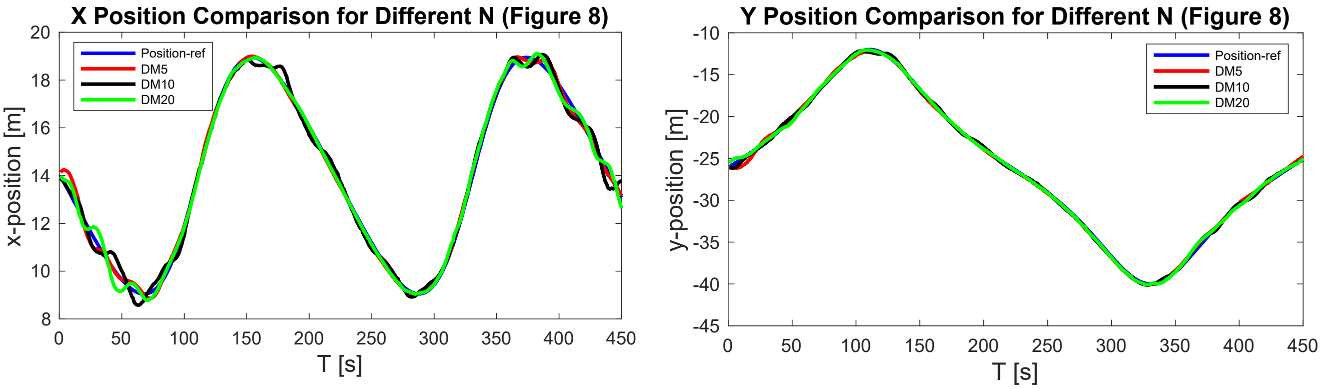
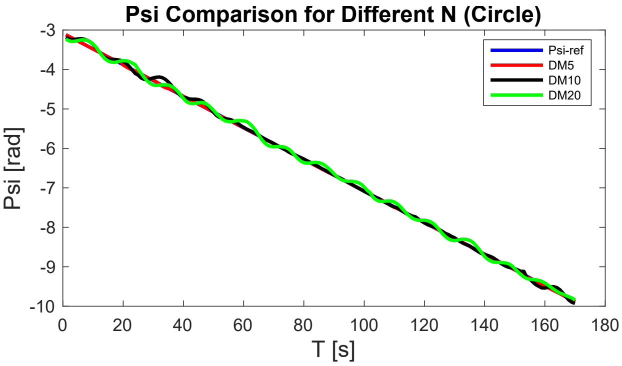


Figure 3-20: Comparison of X and Y positions for the Figure-eight trajectory



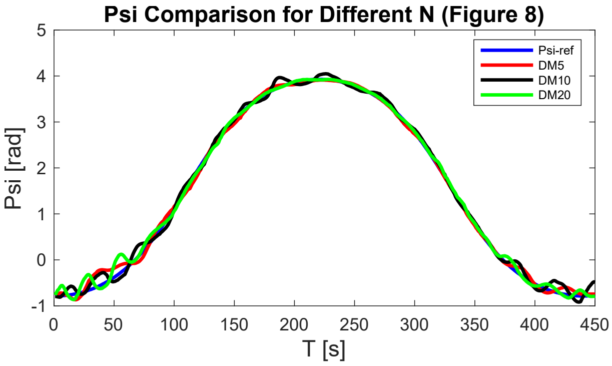
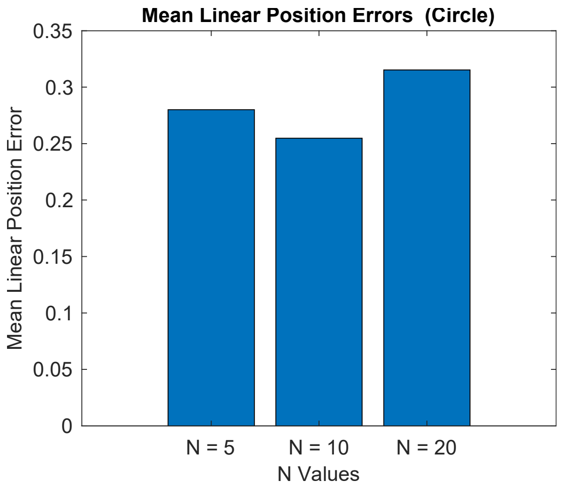


Figure 3-21: Comparison of heading angles for the circular trajectory

Figure 3-22: Comparison of heading angles for the Figure-eight trajectory



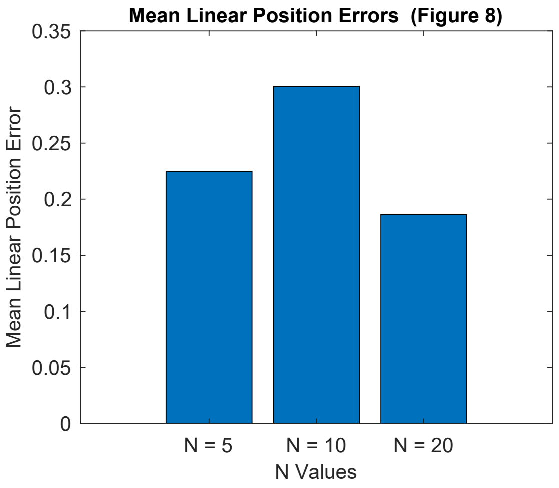


Figure 3-23: The error for positions the different prediction horizons for circular and figure eight trajectory.

In the comparative evaluation of heading angle errors (Figure 3-24), a prediction horizon of N = 5 yielded the yaw angles 0.005 rad for the circular trajectory and 0.0532 rad for the figure-eight shaped trajectory. Although longer prediction horizons displayed better positional accuracy, the orientation accuracy, indicated by yaw angle errors, was compromised. The study noted that while positional error reductions at higher prediction horizons were minimal, significant yaw angle deviations had a detrimental impact on the ship's ability to maintain its course, an undesirable outcome for autonomous navigation. When the prediction horizon is extended, small inaccuracies in modeling or minor deviations can build up over time, resulting in considerable errors. This effect is more pronounced in complex trajectories, where even negligible errors in heading or position can amplify into large deviations from the planned route. The higher prediction horizon takes high computational times which is not desired in the trajectory tracking problem. For trajectory tracking problem a smaller prediction horizon is well suited.

An observation worth noting was the asymmetry in the error along one side of the figure-eight trajectory. This discrepancy stemmed from a calibration error in the feedback camera. Another potential contributing factor to this disparity could be the inconsistent feedback received from the sensors. The calibration of the mechanical structure supporting the sensor mount may also have played a role in causing this deviation or error.

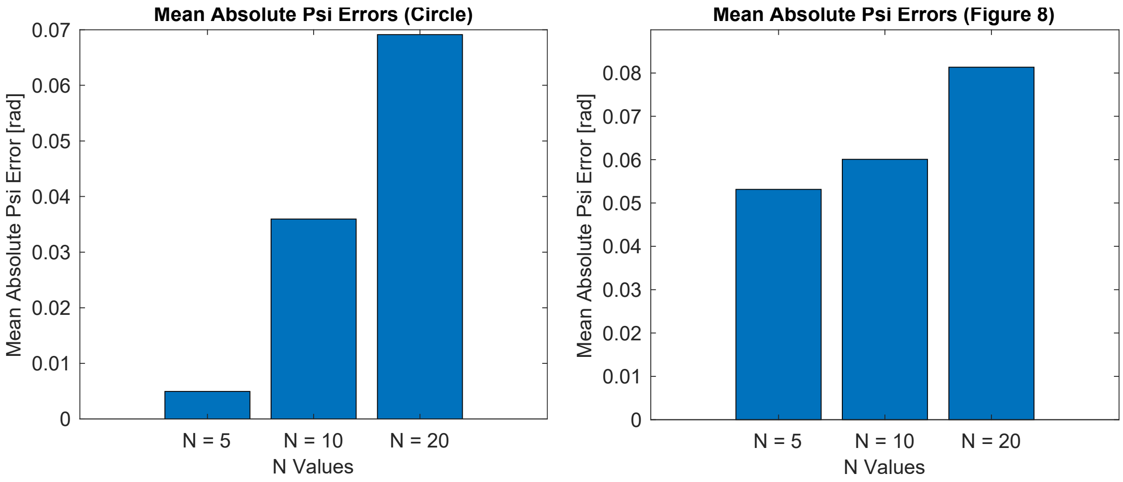


Figure 3-24: The heading error of the different prediction horizons for circular and figure eight trajectory.

Based on the above figures and the figures below for the velocity comparisons (Figures 3-25 and 3-26) for both the cases of trajectory tracking N = 5 provides the best results. The heading angle error increases with the prediction horizon in both simulation and experiment, with N = 5 having the smallest error. Surge and sway velocity errors increase with the prediction horizon in the experiments. Yaw rate error is least at N = 5 in the experiment, which is consistent with the better heading control at this horizon. The average execution time per iteration increases with the prediction horizon, which is expected due to the higher computational load for longer horizons. In both trajectories, the experimental results generally favor a prediction horizon of N = 5 for most error metrics, indicating that a shorter horizon allows the NMPC controller to better cope with the complexities of the real world, which is a dynamically changing environment. This could be due to the quicker adaptability and responsiveness of the NMPC controller at a shorter horizon, which is critical for dealing with rapidly changing wave patterns. The longer computation times for larger horizons could lead to a delay in response, which may not be ideal in an environment where quick adjustments are necessary.

When optimizing NMPC for autonomous ships, it's important to balance prediction accuracy with computational efficiency and responsiveness to disturbances. The data suggests that for the conditions tested, a smaller prediction horizon (N = 5) is more effective for managing the ship's trajectory and orientation.

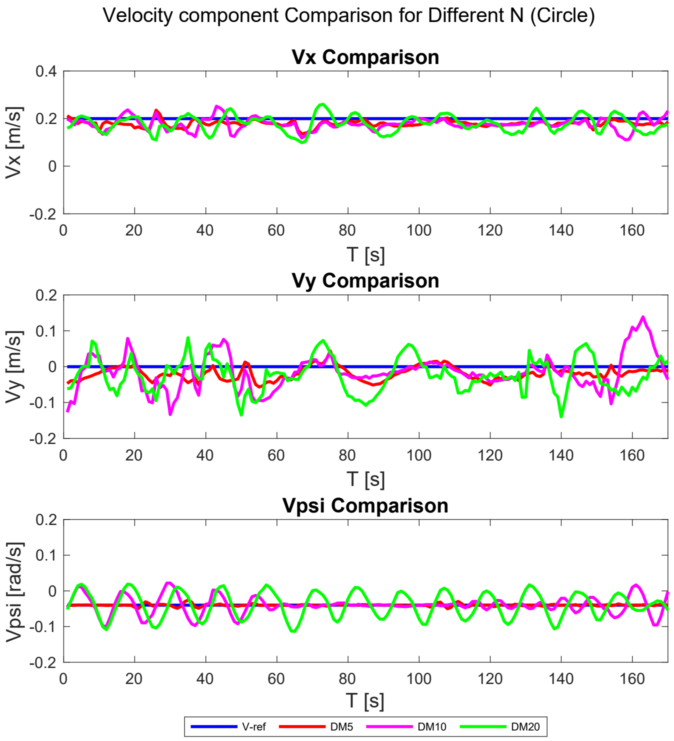


Figure 3-25(A): Comparison of velocity components for the circular trajectory for different N.

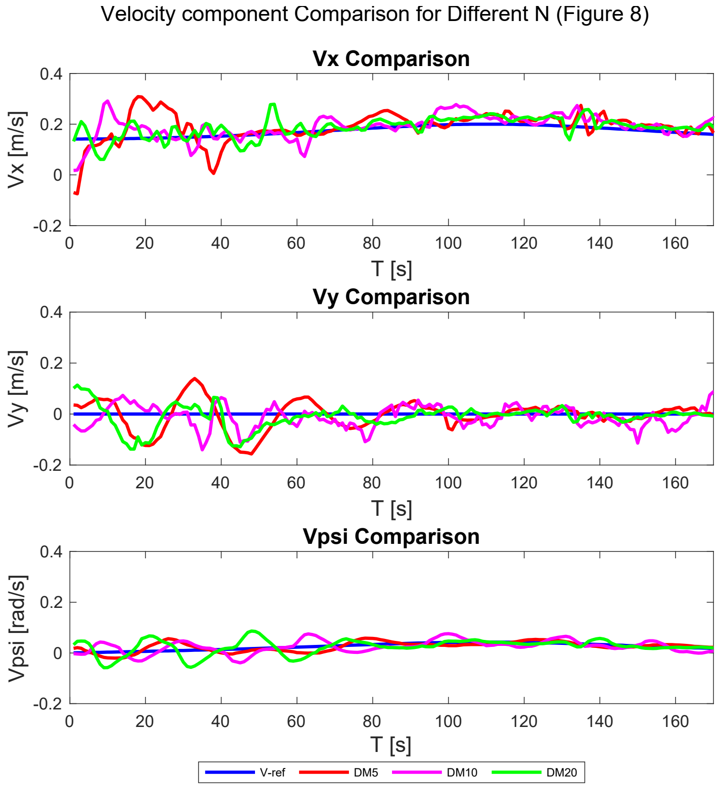


Figure 3-25(B): Comparison of velocity components for the Figure-eight trajectory for different N.

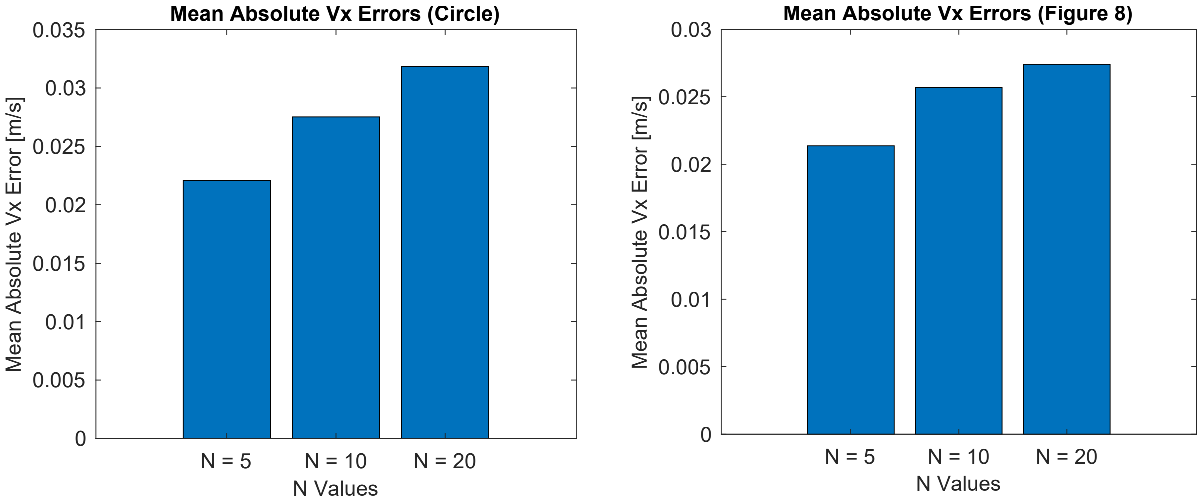


Figure 3-26A: The error for surge velocities for the different prediction horizons for circular and figure eight trajectory.

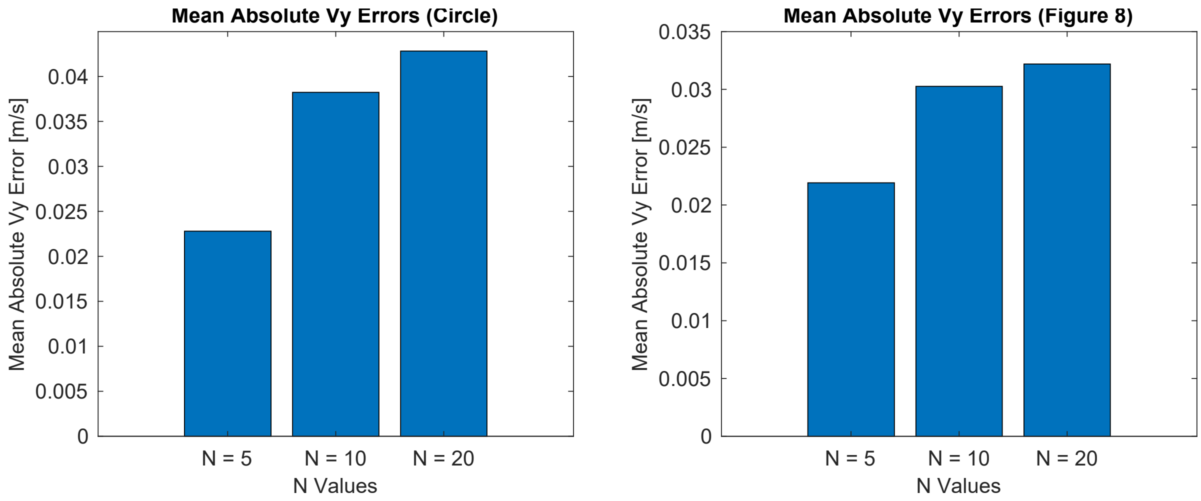


Figure 3-26B: The error for sway velocities for the different prediction horizons for circular and figure eight trajectory.

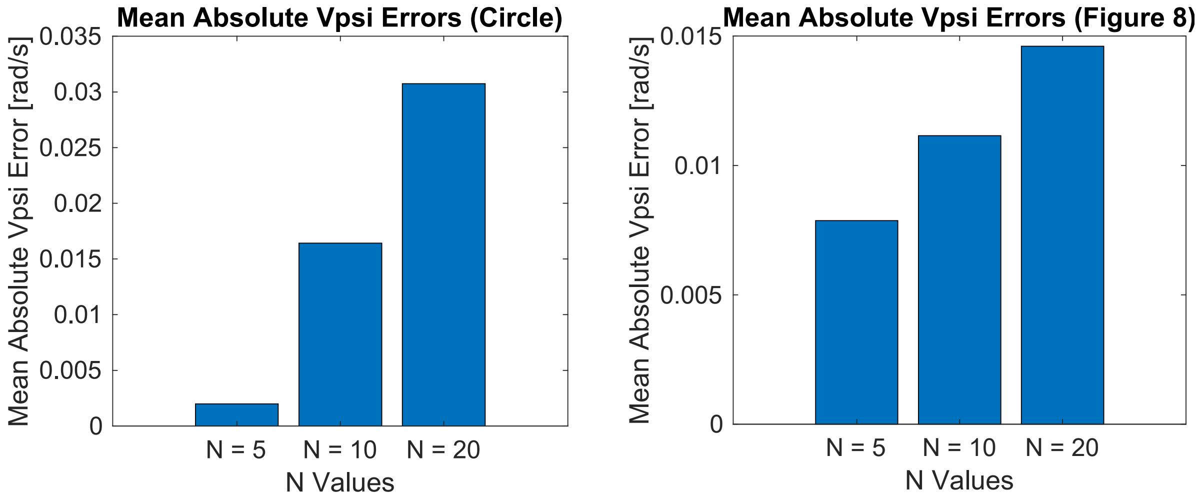


Figure 3-26C: The error for yaw rates for the different prediction horizons for circular and figure eight trajectory.

From the analysis of the forces applied as the control actions to the ship we get this data set for the experimental set up in Table 3-7. From this table and Figure 3-27 we can tell that The NMPC controller appears to exert more control effort for the figure-eight trajectory than for the circular one, which is expected due to the figure-eight's complexity. For both trajectories, a prediction horizon of N = 5 tends to result in a more conservative range of control efforts, suggesting a balance between responsiveness and overcorrection. Prediction horizons of N = 10 and N = 20, exhibit more aggressive ranges of control forces and moments that could be less efficient and potentially lead to increased wear and energy consumption. The optimal prediction horizon is N = 5 for maintaining efficiency in control efforts, especially when handling complex trajectories and dynamic environmental disturbances. This horizon allows for a responsive yet not overly aggressive control strategy, which is advantageous for operational efficiency and system durability. Exploring even shorter prediction horizons could provide insights into whether further improvements in responsiveness or efficiency can be achieved in the control strategy.

Table 3-7: comparison of the forces for different prediction horizon for the two trajectories

Trajectory		**Surge force (τ_x) range**			**Surge force (τ_y) range**			**Yaw moment (τ_ψ) range**		
		N =5	**N =10**	**N=20**	**N =5**	**N =10**	**N=20**	**N =5**	**N =10**	**N=20**
Circular	min	-10.19	-46.69	-23.24	-20.38	-50	-49.99	-53.68	-85	-85
	max	20.12	50	49.99	-6.59	50	32.35	26.34	100	99.99
Figure eight	min	-36.55	-50	-49.99	-50	-50	-34.26	-10.72	-61.60	-77.45
	max	50	50	50	33.48	50	50	13.42	33.47	45.36

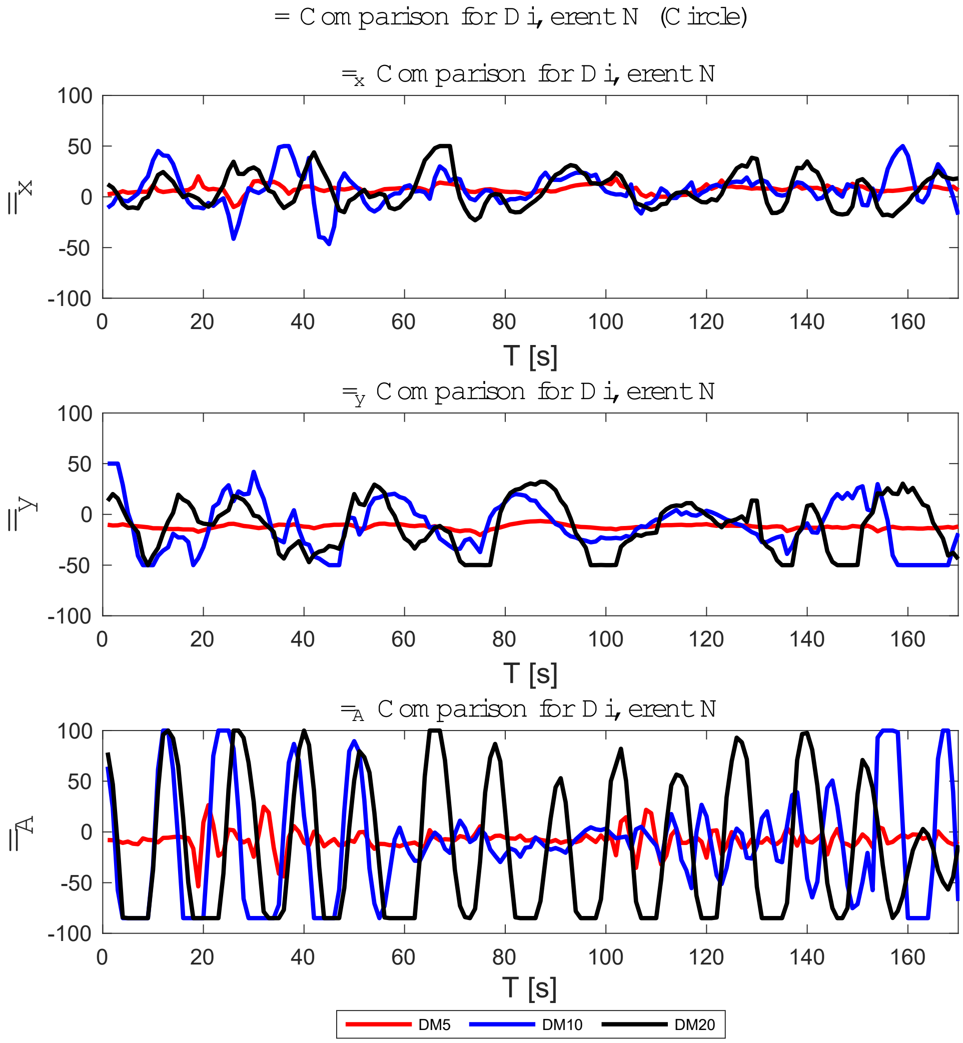


Figure 3-27A: Comparison of Forces for the circular trajectory for different N.

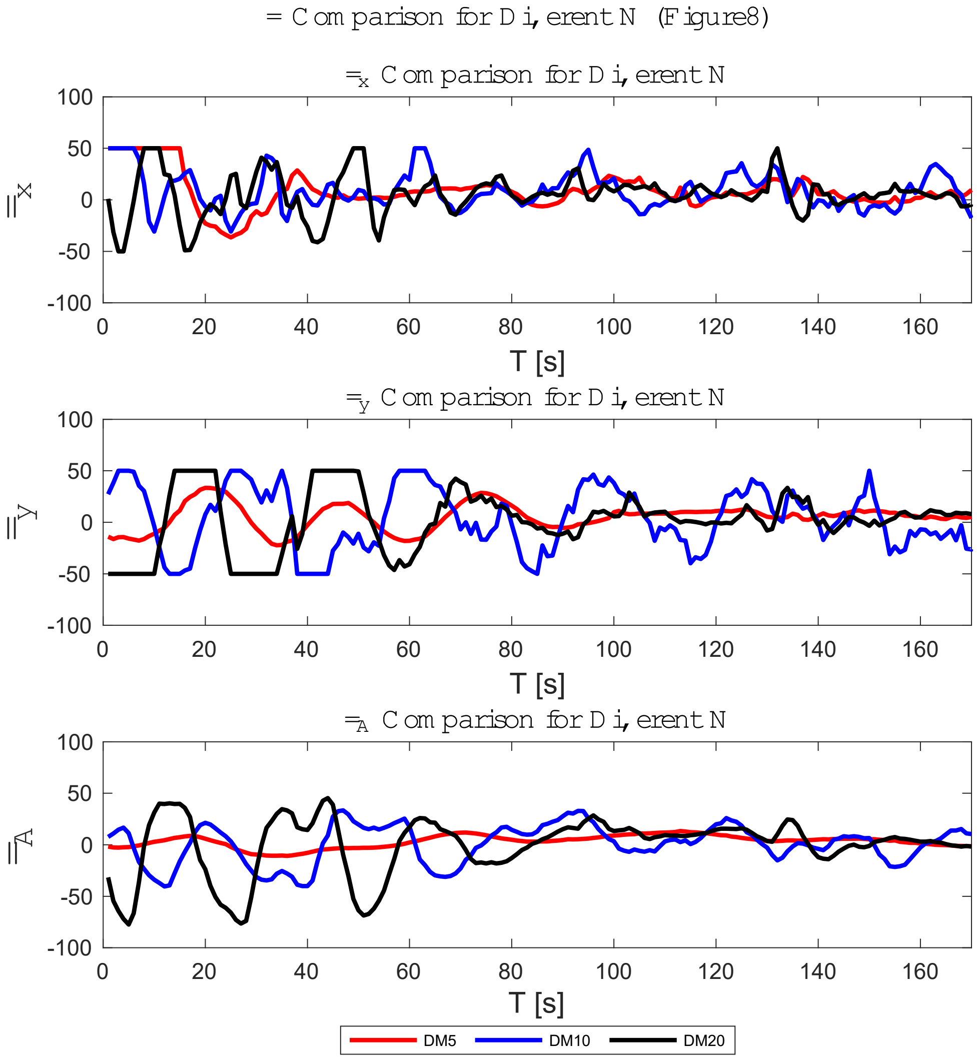


Figure 3-27B: Comparison of Forces for the Figure 8 trajectory for different N.

After observing all the results, it can be concluded that with a shorter prediction horizon the controller works better because it allows the controller to be more responsive to immediate changes, making quick adjustments as needed. It also required less computational time, allowing for more frequent updates and potentially more accurate real-time control. The interior-point method is effective at handling constraints, which are critical in maintaining control performance at lower speeds like 0.2 m/s. Tighter control over these constraints helps prevent the ship from violating physical and operational limits. But the large prediction horizons were violating the operational limits frequently which was observed in the Figure 3-23. The overall performance of the controller in a real-life environment was pretty good. The trajectory was followed very well and the errors were numerically very small.

3.4 Conclusion

The proposed NMPC was evaluated in both simulation and experimental settings. Simulation results demonstrate significant potential for the controller's effectiveness, particularly under harsh environmental conditions. However, experimental results were conducted in calm water due to the unavailability of the wave generator at the NRC testing facility.

From the simulation findings, a moderate control horizon was found to provide favorable results. Conversely, experimental studies indicated that a smaller prediction horizon (N=5) yielded the best performance. This is attributed to the slower controller response with larger control horizons, resulting in deviations from the trajectory and occasional oscillatory behavior. Moreover, larger control horizons were observed to lead to accumulation of errors and model inaccuracies over time.

The controller aims to achieve two objectives: maintaining trajectory and constant speed. There appears to be an interaction between response speed and tracking accuracy, highlighting the need for tuning in ship navigation to optimize performance.

Additionally, challenges such as drop-offs in feedback, delays from Windows PC and control software, and associated errors were identified as significant factors affecting system performance. Addressing these issues is crucial for enhancing overall control effectiveness and reliability in practical maritime operations.

Chapter 4

4.0 Comparative Study of Physics-Based and Neural Network NMPC Models for the Autonomous Ship Trajectory Tracking.

Abstract

Autonomous Surface Vessels (ASVs) are rapidly emerging as a cutting-edge technology in maritime operations. Their safe and efficient trajectory-tracking capability is essential for the growing autonomous ship industry. Adhering to stringent maritime regulations especially in the unpredictable marine environment, poses considerable challenges. Environmental elements such as waves, currents, and winds introduce complexities that demand a robust control system for ASVs to navigate effectively. This research aimed to conceive a controller that addresses these multifaceted challenges while conforming to all safety standards. Nonlinear Model Predictive Control (NMPC) emerged as the optimal solution, adept at managing the nonlinear dynamics of ships, accounting for unmodeled behavior, and adhering to multiple constraints. The NMPC controller, augmented by an Unscented Kalman Filter (UKF) to counteract wave effects, demonstrated its robustness in simulated and real-world trials conducted at the National Research Council (NRC) of Canada. Incorporating Machine Learning (ML) with NMPC has significantly advanced our trajectory-tracking objectives. Achieving precise trajectory following in real-world maritime conditions continues to be an intricate endeavor, with the complexity further amplified by diverse ship dynamics and environmental settings. An Artificial Neural Network (ANN) could capture intricate ship dynamics, showing impressive results in simulations and hands-on experiments. The performances of mechanistic and neural network models were meticulously compared in both simulated environments and experimental basins to validate their effectiveness

and practicality. Finally, an improvement in the model is suggested using a deep neural network trained with natural data. The implementation of NMPC with neural network structures was one of the core objectives of the thesis.

4.1 Introduction

The marine industry is keenly interested in integrating ASVs into their operations, recognizing their capacity to operate autonomously. These vessels serve diverse purposes, including scientific research, data collection, and military operations, necessitating accurate trajectory tracking to fulfill their specific missions. Robust control methodologies detailed in [12], [13], [14] aim to minimize a ship's tracking error, employing techniques like the Lyapunov method, composite nonlinear feedback controllers, and finite-time leader-following formations. Despite these efforts, the convergence rate towards minimal error remains slow, and the exact limits of error reduction are indeterminate. Research in [15] utilizing a finite fault-tolerant controller achieved commendable accuracy but similarly suffered from sluggish convergence rates. More rapid solutions discussed in [16] and [17], which rely on motion planning, are often contingent on specific initial conditions, the precise calibration of control variables, and the uncertainties inherent in ship models. Some of the research papers worked extensively for the implementation of the adaptive control approach for the autonomous ships. In the presence of the environmental disturbances, it is very difficult to control the motion of the vessel due to the nonlinear dynamics of the ship. Proposals for implementing adaptive control within such a fluctuating marine environment have been put forth in [2], [3], yet implementing it in the practical level is to be confirmed. The functions of the autonomous vessels are primarily consisted of the three main tasks: dynamic positioning, path following and trajectory tracking [7]. By far path planning and trajectory tracking requires the most robust controller. It is very significant in the offshore applications where safe navigation is the most important concern. The trajectory tracking involves following a defined path with the suitable speed profile. To enhance the control performances the ship model addressing the uncertainties and disturbances is important [8], [9]. Under actuation of the ships play a vital role because at high-speed operations the autonomous surface vessels struggle to provide direct actuation in the sway direction. The backstepping and Lyapunov's approach has been proved useful in correcting this issue [10].

Artificial Neural Networks (ANNs) are strong machine learning technique to deal with complex problems with strong nonlinearities. It can identify the relationships between the multiple parameters to manipulate them. In autonomous shipping, ANNs have the potential to significantly enhance trajectory planning by learning from past data on ocean currents, weather patterns, maritime traffic, and navigational regulations, thereby adapting to the multifaceted variables affecting a vessel's path. Integrating ANNs with Nonlinear Model Predictive Control (NMPC) forms a potent and adaptable control framework that has the potential to surpass traditional mechanistic model-based strategies. ANNs are adept at modeling complex and nonlinear relationships, thus empowering NMPC to handle unpredictable and evolving system behaviors where linear models might fall short. By embedding difficult to model system dynamics within ANNs, NMPC gains enhanced predictive capabilities, leading to more accurate anticipation of future states and informed control actions. ANN-based models have the capacity to incorporate a variety of constraints, from input limitations to state and safety parameters, which are crucial across numerous control applications. Additionally, they can streamline the computational demands of NMPC's optimization process. This combination of Artificial Neural Networks (ANN) and Nonlinear Model Predictive Control (NMPC) works well for tasks that require quick responses, like in robotics, autonomous vehicles, and process control systems. ANNs contribute to NMPC's robustness by managing uncertainties and noise in system measurements. Moreover, they enable NMPC to tackle multi-objective optimizations, balancing trajectory adherence with other objectives like energy efficiency. In specific scenarios, reinforcement learning has been used to refine the control model in real-time using fresh data, thereby continuously enhancing control quality.

Recent explorations in the realm of autonomous vehicle control have unveiled learning-based methodologies for capturing nonlinear behaviors ([36], [37], [38]). ANNs have shown promise in maritime control due to their ability in modelling nonlinear systems under predictable conditions [39]. Initial research utilized ship position, heading, and velocity as neural network inputs to determine control outputs like rudder angle and propeller speed. In [40] a minimum-time maneuvering approach employing an ANN controller and predictive compensator was devised, interpolating pre-computed minimum-time solutions for real-time control. In another instance, ANNs were deployed to simultaneously manage rudders, bow thrusters, and tugs [41]. However, these approaches did not account for environmental factors and lacked robust training datasets. The control mechanism in [42] introduced an adaptive backstepping controller that performed adequately under wind disturbances, albeit assuming the ship's longitudinal velocity was nil. Further, a study in [43] implemented a feedforward neural network alongside a Proportional-Derivative (PD) controller, which proved challenging for multivariable system management. Vehicles operating with a Proportional-Integral-Derivative (PID) controller often display less-than-ideal performance when initiating tasks from different starting points, a limitation noted in [19]. To mitigate this shortcoming, [20] proposed an enhanced version of the fuzzy PID controller, which is engineered to adapt more fluidly to a range of initial conditions. However, one must consider that the fundamental linearity of standard PID controllers can complicate the dynamic tuning of control gains during live operation. Fuzzy logic controllers are commended for their swift alignment with reference paths, promoting safe and efficient autonomous navigation, as detailed in [21]. These controllers are also known for their real-time responsiveness, which optimizes computational tasks. Despite their effectiveness, as documented in [22], they are not without drawbacks. A notable concern is their propensity to induce oscillations in steering commands, a phenomenon highlighted in [23], which can complicate stability and performance evaluations. Sliding Mode Control (SMC) is a robust technique for managing non-linear control systems. It is particularly effective in handling uncertainties and external disturbances in autonomous vehicles, as evidenced by researchers in [24]. It enables vehicles to quickly align with their designated paths with a high degree of reliability, a benefit confirmed by [25]. Advantages of SMC include its quick reaction time, robustness to system variations, and ease of implementation. Additionally, it can be integrated with other controls for enhanced performance [26]. However, challenges arise in its application, such as the necessity for ongoing linearization [27] and the potential for high lateral acceleration during abrupt trajectory changes [28]. Model Predictive Control (MPC) is highly effective for managing uncertain and nonlinear systems, particularly in autonomous vehicle operations where constraint adherence, such as rollover prevention and lateral stability, is crucial. Its proficiency in constraint management is highlighted in [29], where MPC adeptly maintained tire-road friction limits. MPC's strength lies in handling multi-input and multi-output (MIMO) systems. As an advanced optimal control strategy, it continually solves optimization problems, balancing objectives, and constraints over a defined time horizon, as detailed in [30] and [31]. These attributes make MPC an ideal solution for trajectory tracking in autonomous vehicles, adept at navigating complex constraints with an Artificial Neural Network or mechanistic model. The study in [70] shows that the NMPC controller with disturbances in the form of current ocean currents provided promising results. But the simulation-based results are not enough to understand the adaptability and versatility of the NMPC controller. Also, incorporation of wave disturbances is important. The research in [71] uses NMPC to solve trajectory tracking and collision avoidance problems. They utilized disturbance filter to showcase the simulation results. But the study does not involve the experimental results. The trajectory used for tracking was a straight line. Irregular or complex trajectories in an real experimental setup fully shows the utility of the controller. An improved Nonlinear Disturbance Observer (INDO) based NMPC was utilized in [72]. The research showed good results in Dynamic Positioning (DP) of the ship and is expected to track the trajectories with almost zero offset. But in this work the DP of the ship was shown in simulations and no trajectory tracking task was done. In the paper in [73] a backstepping controller is compared with the MPC based controller for comparing the tracking results. Apparently the MPC showed much better tracking results. But still, simulation results without disturbance modelling cannot express the model's full potential.

This thesis seeks to develop a robust NMPC for Autonomous Surface Vessels capable of adapting to different ship models and environmental conditions, ensuring precise and efficient trajectory tracking.

4.2 Methodology

The key areas of focus in our study include:

- Development and Design of the Nonlinear Model Predictive Controller.
- Development of the Magne Viking Vessel model using Neural Network technique.
- Disturbance models and Unscented Kalman Filter methods.

4.2.1 Nonlinear Model Predictive Controller (NMPC)

In this section, we focus on the trajectory tracking of the Magne Viking surface vessel using a Nonlinear Model Predictive Control (NMPC) scheme. The ship's motion is modeled using a 3-degree-of-freedom (DOF) framework, emphasizing surge, sway, and yaw movements, while excluding heave, roll, and pitch [54]. The NMPC, leveraging the vessel's nonlinear dynamics as its prediction model, operates in real-time by iteratively solving nonlinear optimization problems at each time step, as detailed in [55]. Key elements of NMPC include a quadratic cost function, prediction model, state and input constraints, and output feedback control architecture incorporating an offset-free formulation [56]. The state space model, denoted as f, captures the nominal six-state model essential for accurate trajectory tracking, expressed through equation (4.1).

$$x(k+T) = x(k) + \int_{k}^{k+T} f\big(x(\tau)\big)d\tau, \tag{4.1}$$

Where x (k) represents the current state and T is the sampling time.

$$x(k+T) = x(k) + \int_{k}^{k+T} f_{aug}(x(\tau), d(k)d\tau, \tag{4.2}$$

$$d(k+T) = d(k), \tag{4.3}$$

Equation (4.4) gives the predicted output, where g_{aug} is the output model.

$$y(k+T) = g_{aug}(x(\tau), d(k)) \tag{4.4}$$

Utilizing insights from [56] and [57], we construct an offset-free NMPC by integrating a disturbance model and integrator into the prediction model. This results in an augmented model, labeled as 'f', which merges the original state space (faug) with a disturbance model 'd', as formulated in Equation (4.2). For state predictions, the initial disturbance estimate is treated as constant, resembling a step disturbance, in line with the methodology described in Equation (4.3). The prediction model's numerical integration is executed using explicit Euler's Method, with the predicted state outcomes defined by Equation (4.2).

4.2.2 Cost Function

The cost function minimizes the gap between the desired equilibrium state targets and the actual system states It focuses on aligning the desired equilibrium state targets $\bar{x}$ and the current system states $x(k)$, as well as synchronizing the equilibrium input target $\bar{u}$ with the present input $u(k)$. This alignment is key for accurately tracking the reference signal $r(k)$ which outlines the ship's desired trajectory characteristics. The overarching goal is to ensure offset-free tracking, maintaining consistent alignment with the desired reference trajectory over time.

$$J = \underset{u}{min} \sum_{\kappa=k}^{k+m}(\hat{x}(\kappa) - \bar{x})^T \lambda_1 (\hat{x}(\kappa) - \bar{x}) + \lambda_2 (u(\kappa) - \bar{u})^2) \tag{4.5}$$

$$\bar{x} = f_{aug}(\bar{x}, \bar{u}, \hat{d}(k)) \tag{4.6}$$

$$\bar{r} = g_{aug}(\bar{x}, \hat{d}(k)) \tag{4.7}$$

The cost function, influenced by weights λ_1 *and* $\lambda_2$and operating over a prediction horizon (m), computes the cost of optimal control by considering both running and terminal costs. The optimization, constrained by the trajectories, is addressed using the fmincon function for solving constrained optimization problems, as shown in Equation (4.8). The fmincon found the minimum of the constrained nonlinear multivariate function. The optimization scheme involved the active set method, the interior point method and trust region reflective method for optimization. Our study involved the interior point optimization techniques because of its robustness in handling the nonlinear constraints and solving large scale problems. After getting each states the new state is obtained through Euler's Integration. Details on the states and input variables will be further explored in the vessel model section.

$$x_{min} \leq x \leq x_{max}; u_{min} \leq u \leq u_{max} \tag{4.8}$$

4.2.3 Vessel Model for NMPC

The proposed NMPC was implemented on the simulated model and physical model of the Magne Viking ship. The Magne Viking model ship is a 1:14.7 scale replica of the actual ship. Key vessel parameters were determined through a series of experiments conducted by the National Research Council (NRC).

Mechanistic Model

$$\dot{\eta}(t) = R\big(\eta(t)\big)V(t) \tag{4.9}$$

$$M\dot{V}(t) + C\big(V(t)\big)V(t) + D\big(V(t)\big)V(t) = \tau \tag{4.10}$$

Here,

$\eta(t) = [x(t), y(t), \psi(t)]$ represents the position and orientation.

$V(t) = \left[v_x(t), v_y(t), v_\psi(t)\right]^T$ denotes the vessel's velocity.

τ symbolizes the forces exerted on the ship's center of gravity. For $\eta(t) = [x(t), y(t), \psi(t)]$, the components $x(t), y(t)$ indicate the ship's position in the north-east direction relative to the local geographical frame, while $\psi(t)]$ represents the yaw rate aligned with the north. Given our model's focus on horizontal plane motion, the angular velocity is described by a singular component. The variables related to ship motions and their corresponding measurement frames are concisely presented in the following table.

Table 4-1: Summary of ship motion variables for maneuvering applications

Variable	Name	Frame	Units
x	North Position	Earth fixed	m
y	East Position	Earth fixed	m
ψ	Heading or Yaw angle	Body fixed	rad
v_x	Surge Speed	Body fixed	m/s
v_y	Sway Speed	Body fixed	m/s
v_ψ	Yaw rate	Body fixed	rad/s
τ_x	Surge Force	Body fixed	N
τ_x	Sway Force	Body fixed	N
τ_x	Yaw moment	Body fixed	N-m
$\eta(t) = [x(t), y(t), \psi(t)]^T$	Generalized position		
$V(t) = \left[v_x(t), v_y(t), v_\psi(t)\right]^T$	Generalized velocity		
$\tau = \left[\tau_x, \tau_y, \tau_\psi\right]^T$	Generalized force		

Here in the equation (4.9) The symbol M denotes the Inertial Mass Matrix, which is composed of two distinct elements: the Rigid Body Matrix M_{RB} and the Added Mass Matrix M_A.

$$M = M_{RB} + M_A \tag{4.11}$$

Where,

$$M_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix}; \qquad M_A = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix} \tag{4.12}$$

$$C(V) = C_{RB}(V) + C_A(V) \tag{4.13}$$

In this context, ' m ' refers to the mass of the ship, while x_g indicates the distance between the ship's center of gravity and the center of the body-fixed coordinate frame. ' 'represents matrices that account for Coriolis and Centrifugal forces, encompassing effects from both the rigid body dynamics and additional forces due to Coriolis and centripetal actions.

Where,

$$C_{RB}\,(V) = \begin{bmatrix} 0 & 0 & -m\left(x_g v_\psi + v_y\right) \\ 0 & 0 & m v_x \\ m\left(x_g v_\psi + v_y\right) & -m v_x & 0 \end{bmatrix} \tag{4.14}$$

$$C_A(V) = \begin{bmatrix} 0 & 0 & c_{13}(V) \\ 0 & 0 & c_{23}(V) \\ -c_{13}(V) & -c_{23}(V) & 0 \end{bmatrix},$$

Here $c_{13}(V) = Y_{\dot{v}} v_x + \frac{1}{2}(N_{\dot{V}} + Y_{\dot{r}})$ and $c_{23}(V) = \; -X_{\dot{u}} v_x$

The Damping matrix is a combination of two matrices having linear and non-linear damping parameters.

$$D(V) = D_L + D\_NL(V)$$

Where,

$$D_L = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix}$$

$$D_{NL}(V) = \begin{bmatrix} -d_{11}(V) & 0 & 0 \\ 0 & -d_{22}(V) & -d_{23}(V) \\ 0 & -d_{32}(V) & -d_{33}(V) \end{bmatrix}$$

With $d_{11}(V) = X_{|u|u}|v_x| + X_{uuu} v_x^2$, $d_{22}(V) = Y_{|v|v}\,|v_y| + Y_{r|v|}\,|v_\psi|$,

$d_{23}(V) = Y_{|v|r}|v_Y| + Y_{r|r|}|v_\psi|$, $d_{32}(V) = N_{v|v|}|v_y| \; + N_{r|v|}|v_\psi|$ and

$$d_{33}(V) = N_{v|r|}\left|v_y\right| + N_{r|r|}\left|v_\psi\right|$$

The matrix $R(\eta)$ serves as rotation matrix, crucial for transforming the vessel's velocity from body-fixed coordinates to inertial velocities. It is defined as:

$$R(\eta) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

In this matrix, ψ represents the ship's heading, and τ is the vector denoting the forces applied to the ship.

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_\psi \end{bmatrix} \tag{4.16}$$

Where τ_x and τ_y are the surge and sway forces (N) and τ_ψ is the moment in N-m.

For our performance assessment in both simulations and experiments, we chose the Magne Viking model ship as the Autonomous Surface Vessel (ASV). This model is a scaled-down version of the actual vessel. Critical parameters of the vessel were ascertained through a series of tests conducted by the National Research Council (NRC). These identified parameters are crucial for the creation of a precise mechanistic model, pivotal for accurately simulating the ship's dynamics and conducting trajectory tracking experiments. Figure 4-1 below depicts the Magne Viking ship model that was employed in the trajectory tracking experiment at the NRC's testing facility.

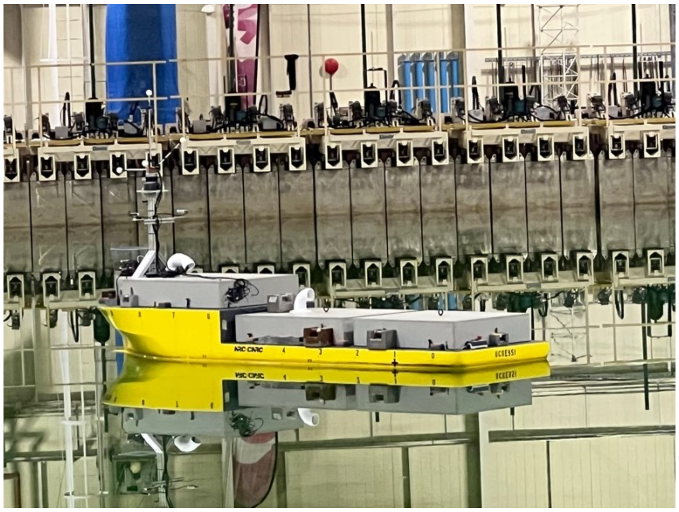


Figure 4-1: Magne Viking model ship (NRC, St. John’s, NL, Canada)

Table 4-2: Magne Viking model Vessel Parameters

Parameters	Value	Parameters	Value	Parameters	Value
m	1290.45	$Y_{\|r\|v}$	0	$N_{\|v\|v}$	0
x_g	0.0184	$N_{\|r\|v}$	0	Y_r	0
I_z	$1.57765\mathrm{x}10^{3}$	N_r	26	Y_v	336.055
$X_{\dot{u}}$	158.23	$Y_{\|v\|r}$	0	$N_{\dot{r}}$	$1.55992x10^{3}$
$Y_{\dot{v}}$	1087.3	$N_{\|v\|r}$	0	$X_{\|u\|u}$	194
$Y_{\dot{r}}$	0	$Y_{\|r\|r}$	0	$Y_{\|v\|v}$	0
$N_{\dot{v}}$	0	$N_{\|r\|r}$	3913.5	N_v	0
X_{uuu}	-112	X_u	41.788		

Neural Network Model Development

The Neural Network model for predicting the ship velocities was developed using MATLAB deep learning toolbox was utilized. Training data was generated based on the available mechanistic model of the ship. This data is crucial as it captures the ship's behavior under various conditions and scenarios. The Magne Viking ship model was used to generate data sets for Neural Network. The generated data set helped the Neural Network to generalize and act according to the trajectory to be followed. Randomly generated forces provided the data set from the Magne Viking model which were then used to train the NN model.

Neural Network Architecture

A feedforward Neural Network (ANN) was chosen to create the NN model. This type of network is well-suited for pattern recognition tasks like predicting ship movements based on historical data. The network was trained using a dataset consisting of 40,000 data points which were generated from the ship's existing model. These data points were carefully selected to represent a wide range of possible scenarios and conditions the ship might encounter. It was tested with a completely different data set to ensure the model's accuracy and generalizability. The testing data was completely unseen to the neural network and hence it prevented information leakage to the trained model. This step is critical to validate that the model performs well with new, unseen data, indicating its reliability in real-world applications.

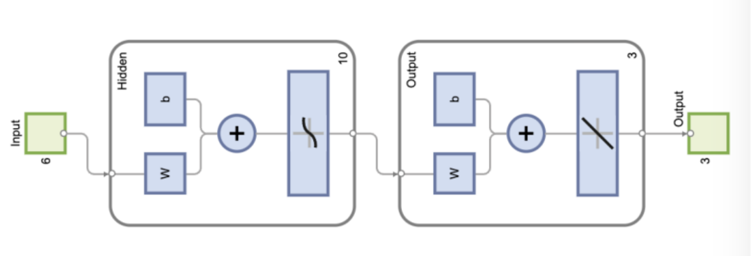


Figure 4-2: Feedforward Neural Network framework for training the dataset for Magne Viking.

In the training process of the neural network using Bayesian regularization, the dataset was strategically divided training and testing sets.

Training Data (80%): A significant majority, 80% of the data, was allocated for training the network. This substantial portion allows the model to learn and adapt to a different scenario, ensuring a thorough understanding of the dynamics involved in ship motion.

Testing Data (10%): Another 10% of the dataset was used for testing. This phase is crucial for assessing the model's performance on data it hasn't seen during training, which is critical for evaluating its predictive capabilities.

Validation Data (10%): The remaining 10% of the data was reserved for validation. This step is essential to fine-tune the model, check for overfitting, and confirm that it generalizes well to new data. Validation helps in making any necessary adjustments to the model before its final deployment.

To accurately capture the dynamics of the ship, the neural network is designed to process specific input and output data. The neural network receives a total of 6 input variables. These inputs consist of the velocity and forces of the ship from the previous time step (t-1). This information includes components such as surge (v_x), sway (v_y), yaw velocities (v_ψ), and the corresponding forces τ_x, τ_y, τ_ψ acting on the ship at that moment. Based on these inputs, the neural network predicts three output variables: the ship's velocities at the current time step (t).

Table 4-3: Neural Network Training Data

Inputs for training NN						Outputs		
${v_x}_{t-1}$	${v_y}_{t-1}$	${v_\psi}_{t-1}$	${\tau_x}_{t-1}$	${\tau_y}_{t-1}$	${\tau_\psi}_{t-1}$	${v_x}_t$	${v_y}_t$	${v_\psi}_t$

These output velocities provide critical information about the ship's surge, sway, and yaw movement. During the training phase, the NN understands the relationship between the forces and velocities at the one-time step and how they influence the ship's velocities in the subsequent time step. This learning process involves adjusting the network's internal parameters to minimize prediction errors. The trained neural network becomes a predictive tool that can estimate the future state of the ship's velocities based on the current and past states. This capability is crucial for the dynamic modeling of the vessel, as it provides a more accurate representation of how the ship will move and react to various forces. By incorporating these inputs and outputs, the neural network forms an integral part of the dynamic modeling process, offering enhanced predictive capabilities essential for effective navigation and control of the ship.

For the testing phase of the neural network, a specific methodology was employed to evaluate its predictive accuracy: The output velocity predicted by the neural network for the current time step (t) was utilized as part of the input for the next time step (t+1). This approach simulates a continuous prediction scenario, where the network's output feeds directly into its subsequent input, mirroring real-world conditions. Along with the output velocity used as input, the forces calculated for the next time step were also included in the input dataset. These forces, crucial in determining the ship's movement, ensure that the network considers both the immediate past state and the expected external influences in its prediction. This process was repeated sequentially, creating a chain of predictions where each step's output becomes the following step's input. This testing approach closely mirrors how the neural network would function in real-time operation, providing insights into its effectiveness in dynamic environments.

Table 4-4: Neural Network Testing Data arrangement (Only first steps are shown)

Steps	Inputs for testing NN						Outputs		
1	$v_{x_{t-1}}$	$v_{y_{t-1}}$	$v_{\psi_{t-1}}$	$\tau_{x_{t-1}}$	$\tau_{y_{t-1}}$	$\tau_{\psi_{t-1}}$	v_{xo_t}	v_{yo_t}	$v_{\psi_{ot}}$
2	v_{xo_t}	v_{yo_t}	$v_{\psi_{ot}}$	τ_{x_t}	τ_{y_t}	τ_{ψ_t}	$v_{xo_{t+1}}$	$v_{yo_{t+1}}$	$v_{\psi o_{t+1}}$

The key focus of this testing was to assess how accurately the neural network could predict future velocities based on past data and immediate inputs. The accuracy of these predictions is critical for the successful implementation of the network in real-world maritime navigation and control systems. This testing methodology was essential in validating the neural network's capability to effectively predict ship dynamics, an integral part of its role in enhancing maritime navigation and control. So, after testing the trained NN model with the newly generated different set of data from the ship model it was ready to be used with the NMPC. For testing the trained model only require the initial velocities and the forces from the data set. From the next time steps the velocities are replaced by the generated output velocities from the trained model.

4.2.3.1 Training algorithm of Neural Network

The Bayesian regularization method was employed to train the neural network. Bayesian regularization prevents overfitting, ensuring the neural network generalizes well to new, unseen data. This is crucial for a model used in dynamic environments like maritime navigation, where conditions vary significantly. It works by adjusting the network weights, finding a balance between fitting the training data accurately and keeping the weights small. This balance is critical to creating a robust model that performs reliably under different scenarios. Bayesian regularization inherently controls the complexity of the neural network. It does this by considering the probability of the data in the model, which helps determine the most probable weights. This method results in models that generalize better, making them more reliable when applied to real-world situations, such as predicting the dynamics of a ship based on various inputs. Using Bayesian regularization for training thus ensures that the neural network is accurate with the training data and capable of performing well with new, real-world data in the context of ship dynamics and control.

The trained ANN model was then integrated into the NMPC framework. This integration allows the NMPC to utilize the ANN model's predictions for efficient and accurate trajectory tracking of the ship. The final model, a combination of ANN and NMPC, is designed explicitly for trajectory tracking. It allows for enhanced control and prediction capabilities, enabling the ship to navigate effectively while accounting for various dynamic environmental factors. This approach demonstrates a sophisticated application of artificial intelligence in maritime navigation, enhancing the capabilities of traditional control systems with the predictive power of neural networks.

4.2.4 Disturbance Model

This research delves into managing the navigation of marine surface vessels amidst unforeseen environmental variations. The employed ship movement model incorporates elements such as the Coriolis and centripetal matrix, alongside nonlinear damping factors. To tackle the challenge of undetected disturbances, we implement an observer to approximate these disruptions. This estimated data aids in devising a robust controller, optimized for precise trajectory adherence. Key challenges in maintaining the vessel's course stem from disturbances caused by wind, waves, and currents. As depicted in Equation (4.10), a dynamic model for the vessel is formulated to address these factors.

$$M\dot{V}(t) + C\big(V(t)\big)V(t) + D\big(V(t)\big)V(t) \ + d(V_{rc}\gamma_c) = \ \tau_{control} + \tau_{wind} + \tau_{waves} \qquad (4.17)$$

The term $d(V_{rc}\gamma_c)$ shows the current forces. It depends on the speed and direction of the current with respect to the vessel. It shows the transfer of the energy from the vessel to the water. Here,

$$V_{rc} = \sqrt{{v_x}_{rc}^2 + {v_y}_{rc}^2} = \sqrt{\left(v_x - v_{x_c}\right)^2 + \left(v_y - v_{y_c}\right)^2} \tag{4.18}$$

$$\gamma_{rc} = -atan2({v_y}_{rc}, {v_x}_{rc}) \tag{4.19}$$

The variables v_{x_c} and v_{y_c} represent the current velocity as measured in the vessel's body-fixed frame, while γ_{rc} denotes the angle of the current, measured in relation to the vessel's bow. Typically, the forces exerted by the current are expressed through nondimensional coefficients, which correspond to the surge, sway, and yaw directions of the vessel.

Which is expressed as –

$$d(V_{rc}\gamma_c) = \frac{1}{2}\rho V_{rc}^2 \begin{bmatrix} A_{Fc} & C_{Xc}(\gamma_{rc}) \\ A_{Lc} & C_{Yc}(\gamma_{rc}) \\ A_{Lc}L_{oa} & C_{Nc}(\gamma_{rc}) \end{bmatrix} \tag{4.20}$$

In this context, ρ represents the density of water. The formula we use incorporates factors such as the frontal area (A_{Fc}), and lateral area (A_{Lc}) of the vessel's submerged hull, as well as the overall length of the ship (L_{oa}). For vessels equipped with dynamic positioning systems, typical current coefficients are usually derived from experimental data or computational fluid dynamics models, as referenced in [60]. Accurately estimating the current coefficients (C_{Xc}, C_{yc}, C_{Nc}) is a complex task, often requiring in-depth hydrodynamic analysis and scale-model tests tailored to the specific vessel. In certain scenarios, as mentioned in [61], it's common to simplify the model (as per Equation 4.19) by incorporating a linear damping term alongside a bias term. This can be expressed as follows:

$$d(V_{rc}\gamma_c) \approx Dv - R^T(\psi)b \tag{4.21}$$

So the vessel model from (4.9) and (4.10) becomes

$$M\dot{V}(t) + C\big(V(t)\big)V(t) + D\big(V(t)\big)V(t) = R^T\big(\eta(t)\big)b + \tau_{control} + \tau_{wind} + \tau_{wave} \tag{4.22}$$

If currents change gradually, the bias term remains constant in earth-fixed coordinates. To accurately represent this in the ship motion equation (4.22), the bias term is rotated to reflect how current forces vary with changes in the ship's heading. The estimation of the bias involves the use of the Unscented Kalman Filter (UKF). Likewise, the impact of wind forces on the vessel is characterized using nondimensional force coefficients, as detailed in the subsequent equation.

$$\tau_{wind} = \frac{1}{2}\rho_a V_{rw}^2 \begin{bmatrix} A_{Fw} & C_{Xw}(\gamma_{rw}) \\ A_{Lw} & C_{Yw}(\gamma_{rw}) \\ A_{Lw}L_{oa} & C_{Nw}(\gamma_{rw}) \end{bmatrix} \tag{4.23}$$

In this scenario, ρ_a denotes the density of air. The variables A_{F_w} and A_{L_w} represent the frontal and lateral areas of the ship exposed to the wind, respectively. The term L_{o_a}refers to the length of the ship. The wind speed and direction relative to the vessel are indicated by V_{rw} and γ_{rw}, respectively.

$$V_{rc} = \sqrt{v_{x_{rw}}^2 + v_{y_{rw}}^2} \tag{4.23}$$

$$\gamma_{rw} = -atan2(v_{y_{rw}}, v_{x_{rw}}) \tag{4.24}$$

$$v_{x_{rw}} = v_x - V_w cos\beta_w \tag{4.25}$$

$$v_{y_{rw}} = v_y - V_w sin\beta_w \tag{4.26}$$

The wind coefficients in Equation (4.23) can be determined through computational fluid dynamics, model testing, or by applying scaling coefficients from similar vessels. However, from a control design standpoint, wind speed and direction are typically utilized for approximate feedforward compensation. The inaccuracies stemming from this compensation are factored into the bias term of Equation (4.22). Therefore, this bias accounts for both the simplified representations of current and wind forces.

The wave forces are determined as a sum of nonlinear and linear wave components.

$$\boldsymbol{\tau_{wave}} = \tau_{wave}^{lin} + \tau_{wave}^{nlin} \tag{4.27}$$

Linear and low-frequency nonlinear components are crucial in the motion control of ships. The low-frequency nonlinear wave forces are typically treated as input disturbances and are incorporated into the model as a bias term, representing a combination of nonlinear wave and current effects. On the other hand, linear wave forces often act as equivalent output disturbances. In this research, environmental forces such as waves, wind, and currents are identified as external elements that can impact a vessel's motion control system. These forces are divided into wave and low-frequency components. Waves exert pressure on the ship's hull, leading to forces that have both oscillatory (aligned with wave frequency) and nonlinear aspects. The oscillatory forces correspond to the wave frequency, while the nonlinear forces emerge from the quadratic relationship between pressure and fluid particle velocity caused by wave motion. These nonlinear forces cover a spectrum of frequencies, both above and below the wave frequencies. Mean wave forces can cause vessel drift, and forces at differing wave frequencies might induce resonance in the vessel's motion. While high-frequency wave-induced forces are generally too significant for direct ship motion control, they can contribute to hull vibrations.

4.2.5 Unscented Kalman Filter

The Unscented Kalman Filter (UKF) utilizes a deterministic sampling technique to obtain mean and covariance estimations with a limited number of sigma points. This approach is particularly effective in nonlinear systems, outperforming the Extended Kalman Filter (EKF) in a variety of applications, including railways, ships, aircraft, solar probes, and more, as noted in [62]. The UKF operates through two primary phases: prediction and update. These steps in the table 4-5 were employed for filtering wave effects and for state estimation using the UKF.

Table 4-5: The steps involved in Unscented Kalman Filter implementation.

Step	**Description**
Initialization	Set initial state estimate x_0 and covariance P_0 . Set process noise covariance Q_0 and measurement noise covariance R_0.
Prediction Step	Generate Sigma Points: $(\chi_{k-1})_0 = \hat{x}_{k-1}$ $(\chi_{k-1})_i = \hat{x}_{k-1} + \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i$ $(\chi_{k-1})_i = \hat{x}_{k-1} - \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i,$ i = 1, …., n
	Propagate Sigma Points through Dynamic Model: $(\hat{\chi}_k)_i = f((\chi_{k-1})_i), i = 0, \ldots\ldots, 2n$
	Calculate Predicted Mean:

	$\hat{x}_k{}^- = \sum_{i=0}^{2n} W_i^m (\hat{\chi}_k)_i$
	Calculate Predicted Covariance: $P_k = \sum_{i=0}^{2n} W_i^c ((\hat{\chi}_k)_i - \hat{x}_k{}^-)((\hat{\chi}_k)_i - \hat{x}_k{}^-)^T + Q_{k-1}$
Update Step	Generate Sigma points: $(\chi_{k-1})_0 = \hat{x}_{k-1}$
	$(\chi_{k-1})_i = \hat{x}_{k-1} + \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i$
	$(\chi_{k-1})_i = \hat{x}_{k-1} - \sqrt{n+\lambda}\left(\sqrt{P_{k-1}}\right)_i,$ i = 1, …., n
	Calculate Predicted Measurement Mean: $\mu_k = \sum_{i=0}^{2n} W_i^m (\hat{y}_k^i - \mu_k)(\hat{y}_k^i - \mu_k)^T + R_k$
	Calculate Predicted Measurement Covariance: $S_k = \sum_{i=0}^{2n} W_i^c (\hat{y}_k^i - \mu_k)(\hat{y}_k^i - \mu_k)^T + R_k$
	Calculate Cross-Covariance: $C_k = \sum_{i=0}^{2n} W_i^c ((\chi_k^-)_i - \hat{x}_k{}^-)((\chi_k^-)_i - \hat{x}_k{}^-)^T$
	Calculate Kalman Gain: $K_k = C_k S_k^{-1}$
	Update State Estimate:

	$x_k = x_k^- + K_k(y_k - \mu_k)$
	Update Covariance: $P_k = P_k^- - K_k S_k K_k^T$
Iterate	Repeat Prediction and Update steps for each time step.

The scaling parameter, λ, in the Unscented Kalman Filter (UKF) is defined as , $\lambda = \alpha^2(n+k) - n$, where α and k determine the distribution or extent of sigma points around the mean. In this case, λ is set to 3. The covariances are computed as follows:

P_k is a 15x15 identity matrix, reflecting the 15 states for the UKF as mentioned in the table 4-5. For determining Q_k, different matrices are chosen:

Q1 = 1*diag ([1,1,1,10,10,10])

Q2 = 0.001*eye (3)

Q3 = 0.01*diag([40,50,40])

Q4 = 0.005*diag([10,10,10])

And Q_k is block diagonal matrix consisting of Q1, Q2, Q3 and Q4.

And the is chosen as $R_k = \begin{bmatrix} 0.005 & 0 & 0 \\ 0 & 0.005 & 0 \\ 0 & 0 & 0.0005 \end{bmatrix}$

For the state estimation using the UKF the following state vector is used.

$$x_s(t) = \left[F_{w1}, F_{w2}, F_{w3}, F_{w4}, F_{w5}, F_{w6}, x, y, \psi, v_x, v_y, v_\psi, b_1, b_2, b_3\right]^T \tag{4.28}$$

So, the dynamic equation for the state estimation is given by:

$$\dot{\hat{x}}_s(t) = f(\hat{x}, U) + w \tag{4.29}$$

And the measurement equation is:

$$\hat{y}(t) = h(\hat{x}, U) + v \tag{4.30}$$

Here, $\dot{\hat{x}}_s(t)$ and $\hat{y}(t)$ represent the estimated states. The terms w and v denote the process and measurement noises, respectively. The covariance matrices for process noise $Q(t) = E(ww)^T$ and measurement noise $(t) = E(vv)^T$are utilized in the UKF. The Magne Viking Ship model is employed for state estimation in the UKF. Diagrams in Figure 4-3 and Figire 4-4 showing the trajectory tracking with and without UKF is demonstrated in the Figure. Here in the simulation with the same wave model UKF-OFF and UKF-ON shows significant changes in the tracking.

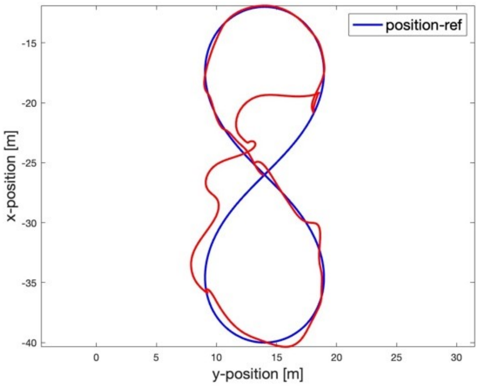


Figure 4-3: Tracking performance with UKF-OFF

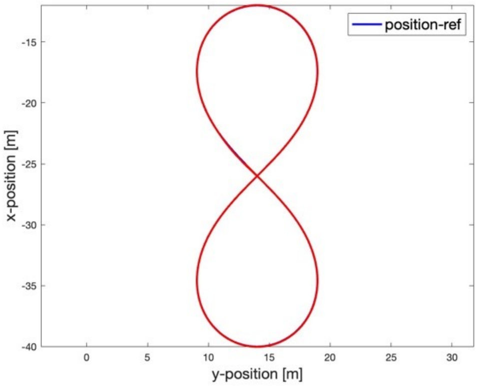


Figure 4-4: Tracking performance with UKF-ON

4.2.6 Planning the Trajectory

To test the controller's feasibility the diverse schemes were utilized to generate the different trajectories. The robustness of the controller was assessed by evaluating the controller's performance under different conditions for different trajectories. The discrete points of the trajectories were to be followed by the controller along the path. The specified points of the trajectory were to be followed while remaining within the defined constrained. And the behaviour of the controller was observed in response to these variations. While following the different trajectories the response from the NMPC helped to refine the controller's performance. Throughout this phase, the goal was to fine-tune the controller's parameters and determine optimal weights for the cost function in the NMPC.

For trajectory generation, a constant average velocity of 0.2 m/s was maintained, aligning with the recommended value for the ship model used. In the case of the oval-shaped trajectory, surge velocity was constrained appropriately to ensure a consistent 0.2 m/s. The heading angle was calculated based on the trajectories, involving the discretization of equations expressing the changes in x and y coordinates:

$$dx = x_2 - x_1$$

$$dy = y_2 - y_1$$

And $\psi = atan2\frac{dy}{dx}$

Subsequently, for comparing the controller's performance with regular and complex shapes, circular and figure-eight trajectories were adapted. These trajectories were then tested at the NRC (National Research Council). A sample trajectory used for simulation and experimentation is provided for reference.

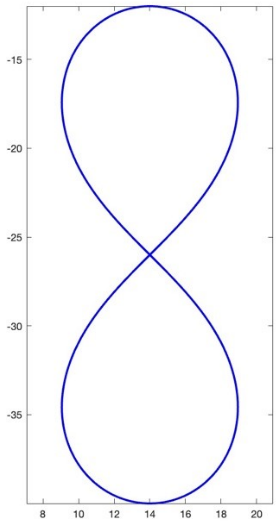


Figure 4-5: Final Generated Trajectory in a Figure-8 Shape

4.3 Results and Discussion

This section focuses on evaluating the performance of the Nonlinear Model Predictive Controller (NMPC) when integrated with two distinct models: mechanical and neural network. The assessment encompasses both simulation and experimental settings. The controllers' effectiveness in maintaining consistent paths and adapting to changing dynamics were evaluated for a circular trajectory and figure 8-shaped trajectory.

4.3.1 Training the Neural Network Model

A significant part of this study was preparing the neural network model for the NMPC which was then used for the simulation and experiments for the Magne Viking ship. The Training progress, data division, Mean Squared Errors and other aspects of the training is described in this section.

Table 4-6: Training with MATLAB Neural Network toolbox

Unit	Initial Value	Stopped Value	Target Value
Epoch	0	1000	1000
Elapsed Time	-	00:2:33	-
Performance	0.115	4.3e-11	0
Gradient	0.238	1.23e-07	1e+10
Mu	0.005	5e+04	1e+10
Effective# Pattern	103	103	0
Sum Squared Pattern	69.2	249	0

The table 4-6 summarizes the training progress for a neural network. The training stopped after 1000 epochs, which took 2 minute and 33 seconds, indicating a relatively swift process. The mean squared error decreased progressively decreased from 0.115 to 4.3e-11 showing the convergence of the algorithm as shown in Figure 4-6.

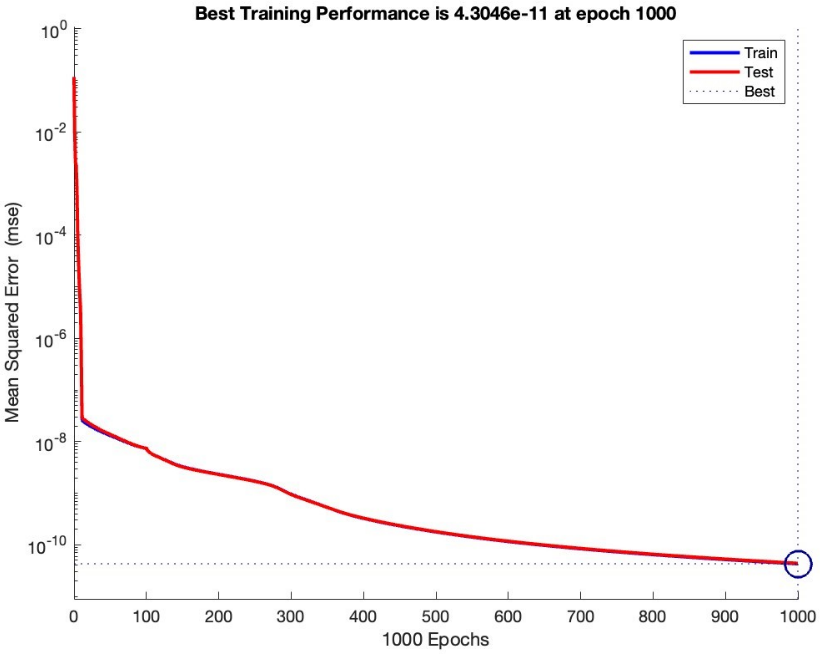


Figure 4-6: Mean Squared Error for NN training.

The gradient, which reflects the steepest descent in the loss landscape, began at 0.238 and decreased to 1.23e-07. The target gradient value is 1e-07, and the stopping value is very close to this target. This suggests that the training likely stopped because the gradient reached a value indicating minimal loss improvement with further training steps.

The sum of squared parameters increased from 69.2 to 249, which may suggest an increase in the magnitude of the weights, possibly leading to a more complex model with a greater risk of overfitting. However, with further context on the acceptable range for this metric, it's easier to draw a definitive conclusion. The summary of convergence results are shown in Figure 4-6.

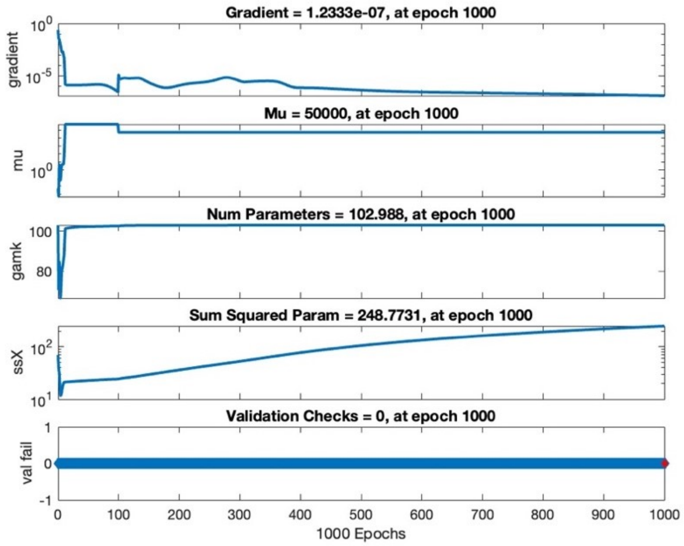


Figure 4-7: Gradient, Learning rate, ssX and validation for NN.

Table 4-7 shows the data division, training algorithm and performance of the training session.

Table 4-7 : Summary of the Neural Network Training Algorithm

Data			
Predictors – 39999 observations with 6 features			
Responses – 39999 observations with 3 features			
Training set - 80%			
Testing and validation – 20%			
Algorithm			
Data division – Random			
Training Algorithm – Bayessian Regularization			
Performance – Mean Squared Error			
Training Results			
Layer size – 10			
	Observations	MSE	R
Training	35999	4.3046×10^{-11}	1.0
Testing	4000	4.3961×10^{-11}	1.0

The neural network was trained successfully, with a final performance indicating an excellent fit to the training data. The training stopped due to the gradient reaching a plateau, a common stopping criterion to prevent overfitting and unnecessary computations. The performance of this well-trained model is further noticed in the testing with unseen new data sets and the simulation and experimental results with NMPC.

Error Histogram Analysis:

The histogram shows the distribution of errors (the difference between the targets and the neural network outputs) for training and test datasets. Most errors cluster around zero, indicated by the vertical 'Zero Error' line in figure 4-8, suggesting that the network predictions are generally accurate. The distribution of errors is slightly skewed, which might indicate a bias in prediction or an asymmetry in the dataset. However, the skewness is minimal, and the network performs consistently across training and testing.

Regression Plot Analysis:

Combining the training progress table analysis, we can conclude that the neural network has been trained effectively with a high degree of accuracy. The network's performance on the test data suggests that it can generalize well to new data, and the errors are mostly centered around zero with a minimal spread, indicating high precision in predictions. This is consistent with the low-performance value and gradient at the stopping point, demonstrating successful training and good potential for practical application. This distribution of data into training, testing, and validation sets ensures a robust and reliable model. It allows the neural network to learn effectively, be tested against unfamiliar data, and be validated for overall performance, which is key for its application in precise maritime navigation and control.

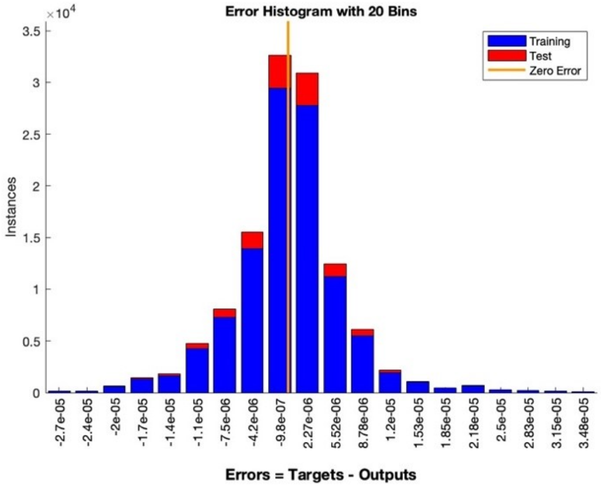


Figure 4-8: Error Histogram in NN training.

Testing Neural Network Model with Unseen Data set

The previously trained neural network model underwent further evaluation using a completely new dataset derived from a randomly selected trajectory. This dataset encompassed both the necessary velocities and forces for the ship to adhere to the chosen path. Upon testing, the neural network model demonstrated impressive performance, indicated by a low error margin. Figures 4-9 illustrate the efficacy of the neural network model in accurately predicting the ship's dynamics. Similarly, when using with NMPC the model will predict the velocities and from the Euler’s integration the position and the heading angles can be determined.

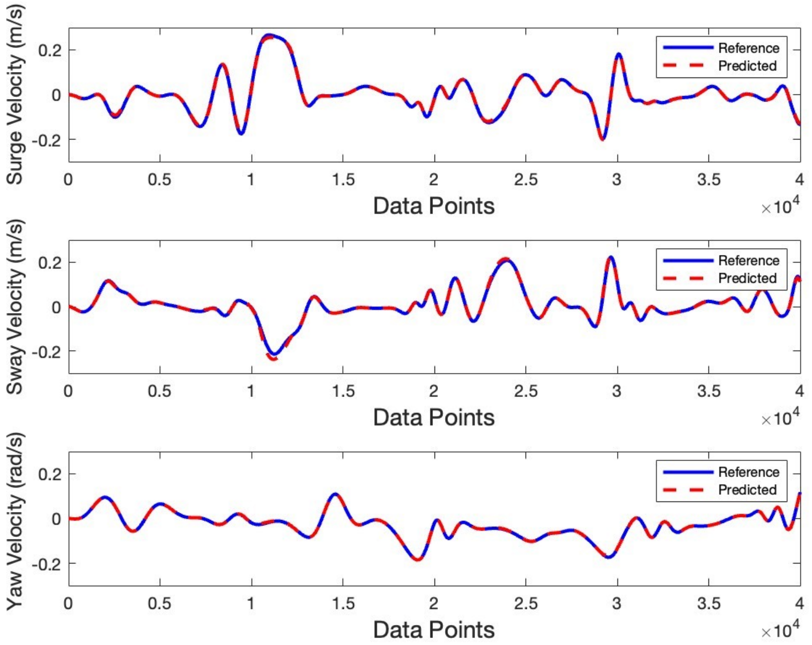


Figure 4-9: Neural Network model testing with a data set unseen to the trained model

The predicted and measured velocities are closely aligned most of the time, demonstrating the neural network's ability to capture the dynamics of the ship's motion. There are instances where the predicted values diverge from the measurements, but these deviations are relatively brief, and the model quickly returns to close alignment with the measurements. The model performs similarly well across all three variables, maintaining a tight correspondence with the measured valued, indicative of a well-trained neural network. The close alignment of predicted and reference values confirms the model's reliability and potential for practical application in Autonomous Surface Vessel Magne Viking trajectory tracking.

4.3.2 Simulation Results for Trajectory Tracking

The simulation featured two trajectories for ship navigation: a circular path with a 5-meter radius and a more complex figure-eight trajectory with a 7-meter radius. The figure-eight trajectory's complexity closely mirrors real-world maritime conditions, requiring frequent navigational adjustments and thus serving as a robust test for the controller's adaptability. Conversely, the circular trajectory is ideal for assessing steady-state performance. Employing NMPC, the study demonstrated the controller's effectiveness across different prediction horizons and in the presence of wave disturbances. This part of the simulation utilized a traditional physics-based model and a neural network model. This section will analyze the controller's performance in relation to critical parameters for ship trajectory tracking.

The wave disturbance that is used in the simulation for all the trajectories are shown in Figure4-10

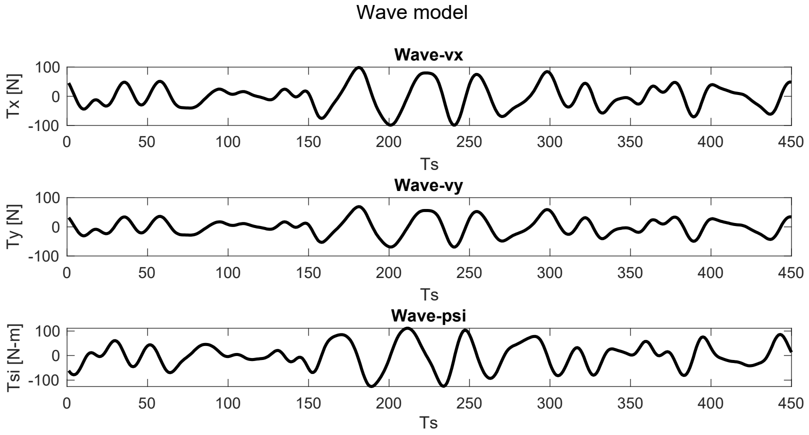


Figure 4-10: Wave model used in simulation.

The forces induced by waves play a crucial role in influencing the velocities of an autonomous ship during trajectory tracking. These disturbances introduce variations and uncertainties that necessitate adaptation from the ship's control system. The control system must dynamically adjust to these external influences to effectively maintain the desired trajectory and velocity of the ship.

4.3.2.1 Circular Trajectory Results analysis

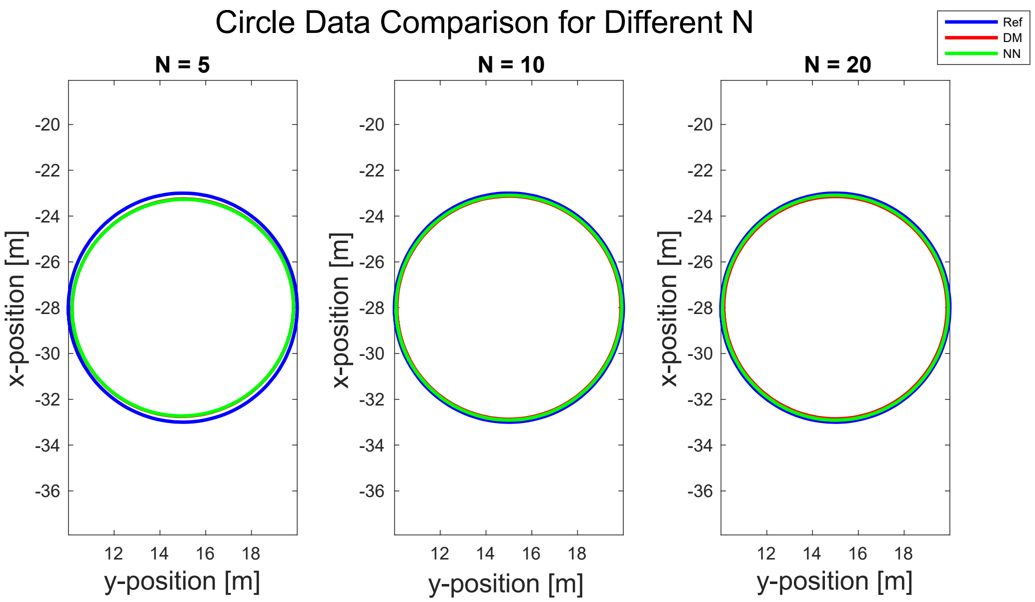


Figure 4-11: Trajectory Tracking of circular path in the presence of simulated disturbance.

The Figure 4-11 demonstrates the trajectory tracking performance of the mechanistic model and Neural Network model. In the following sections the position, velocity, and the consumed power for all the three models will be discussed.

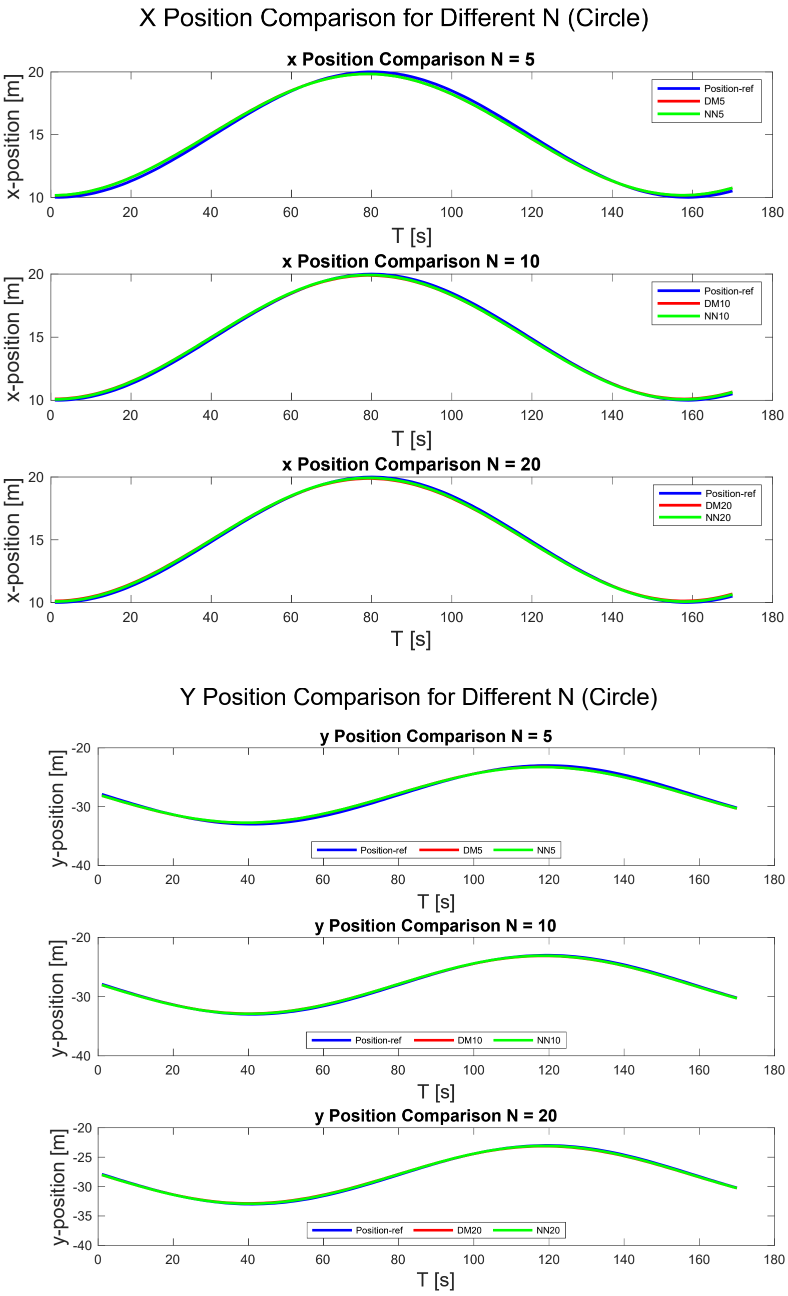


Figure 4-12: Comparison of X and Y positions for the circular trajectory

From the comparison of the position in the Figure 4-12 and from the position error diagram in Figure 4-14 it is visible that the Neural Network model has the lowest error amongst the two model in the prediction horizon of 20. For the trajectory tracking analysis the position and the heading angle management are the two most important factors. Hence, from the Figure 4-13 and the 4-15 we can observe that the neural network model provides the best heading following for the circular trajectory in the prediction horizon of 10. In this simulation result the position and orientation error are high in the lower prediction horizon but gets reduced in the N =10. But there is an increased error in the N = 20 prediction horizon and the neural network model works best in a higher prediction horizon than the mechanistic model. So, a moderate control horizon was able to capture the immediate responses required for the trajectory to be followed. In the meantime, the higher prediction horizon could present higher uncertainty which can lead to less accurate calculation. The relationship between the prediction horizon and the error of the position is nonlinear and mainly depends on the model configuration and the disturbances. It is also evident that the higher prediction horizon can accumulate more errors over the period. It is observed that the mechanistic model did not perform a well for position and orientation at prediction horizon 20 due to higher computational cost, higher computational time, and more uncertainty.

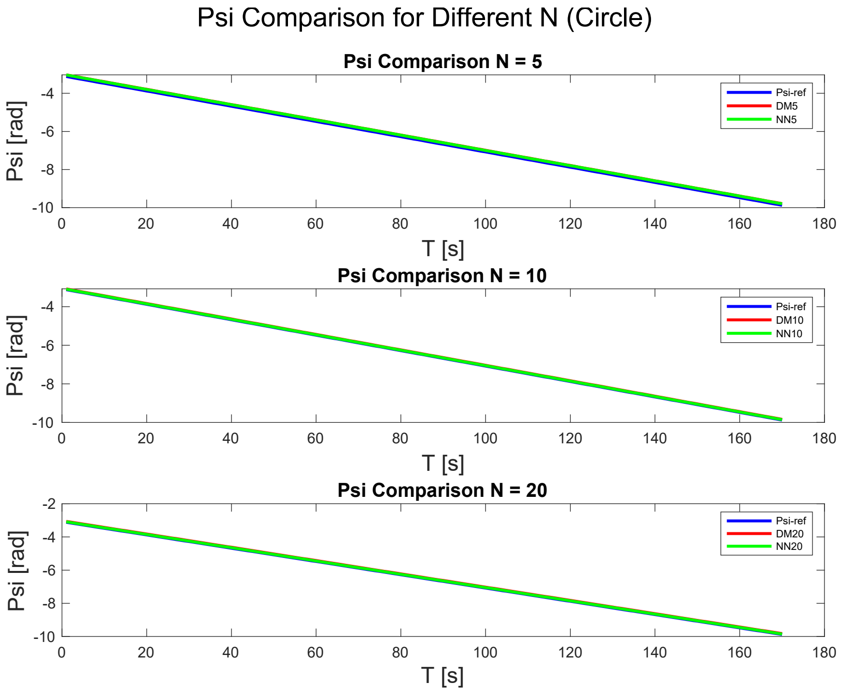


Figure 4-13: Comparison of heading angles for the circular trajectory

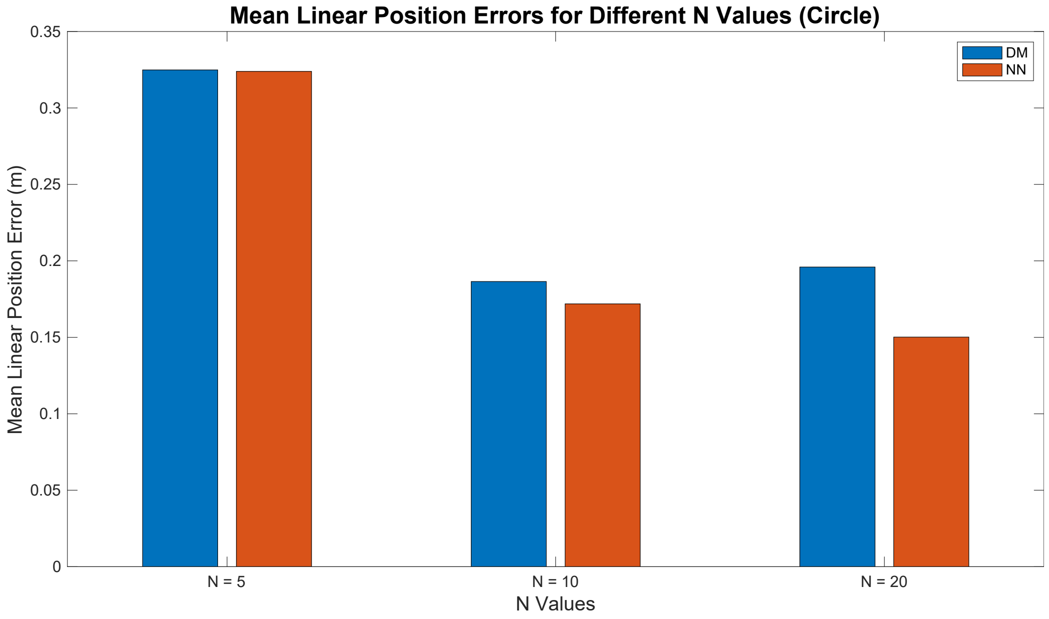


Figure 4-14: Position error at different prediction horizons for circular trajectory

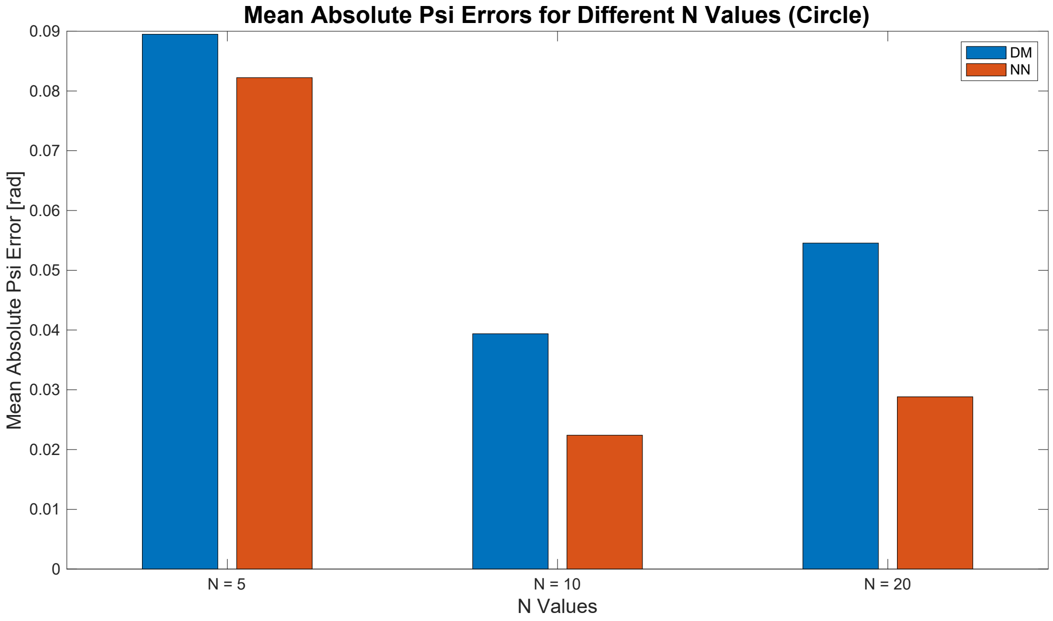


Figure 4-15: Heading angle error at different prediction horizons for circular.

From Figure 4-16 to 4-21, we observe the velocity distribution and the error calculated in this part. The ship moving at 0.2 m/s with induced disturbances can be seen in these figures. The velocity distribution shows that the velocity remained within the limits even with the effects of the wave disturbances affecting the surge, sway, and yaw velocities. For all the prediction horizons, the three models achieved the velocities required to track the path on time. The NMPC controller was able to compensate for the disturbances effectively. The consistency in the velocity tracking also suggests the effectiveness of the UKF applied.

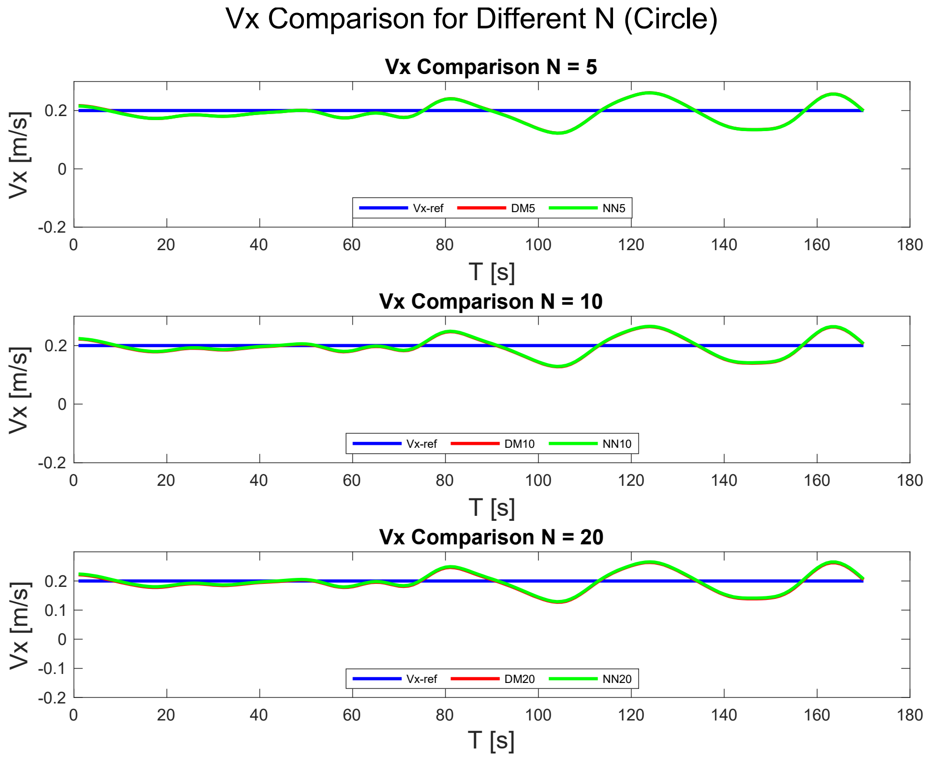


Figure 4-16: Comparison of X-velocity component $v_x$for the circular trajectory

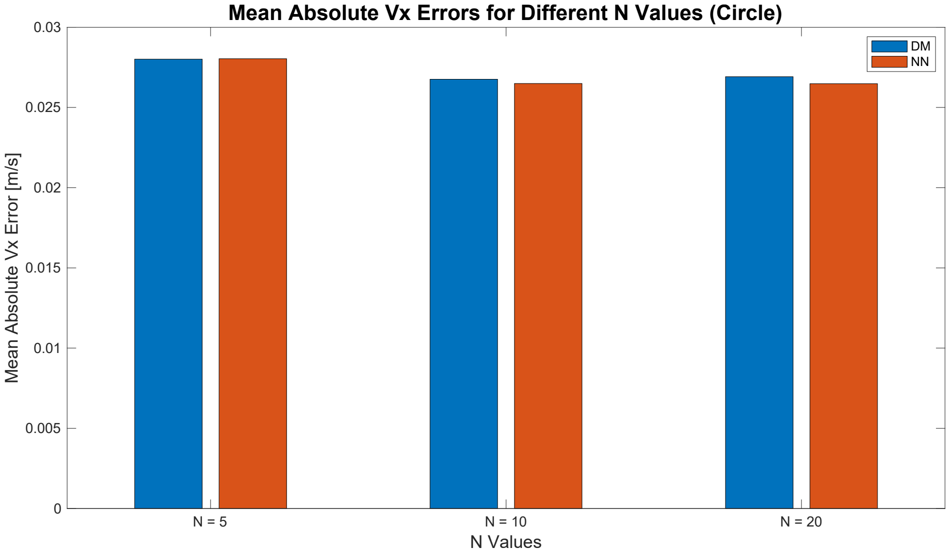


Figure 4-17: The error in v_x at different prediction horizons for circular trajectory

From the Figures 4-17 and 4-21 it is seen that the difference between performance for the velocity tracking is almost similar for all the models. But for tracking the v_y the hybrid model produced a better outcome which can be concluded from the figure 4-19.

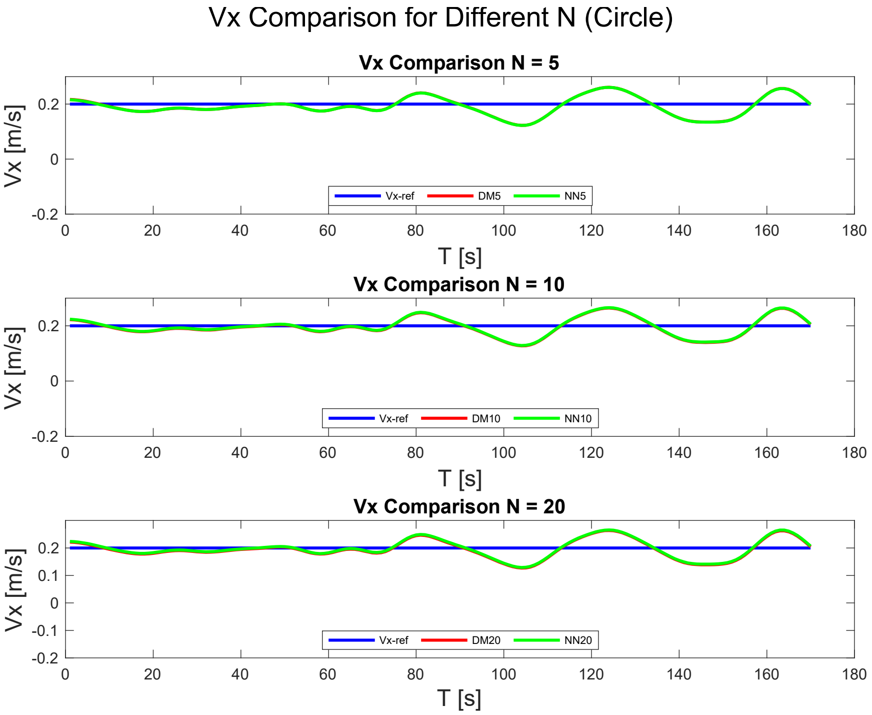


Figure 4-18: Comparison of Y-velocity components v_y for the circular trajectory

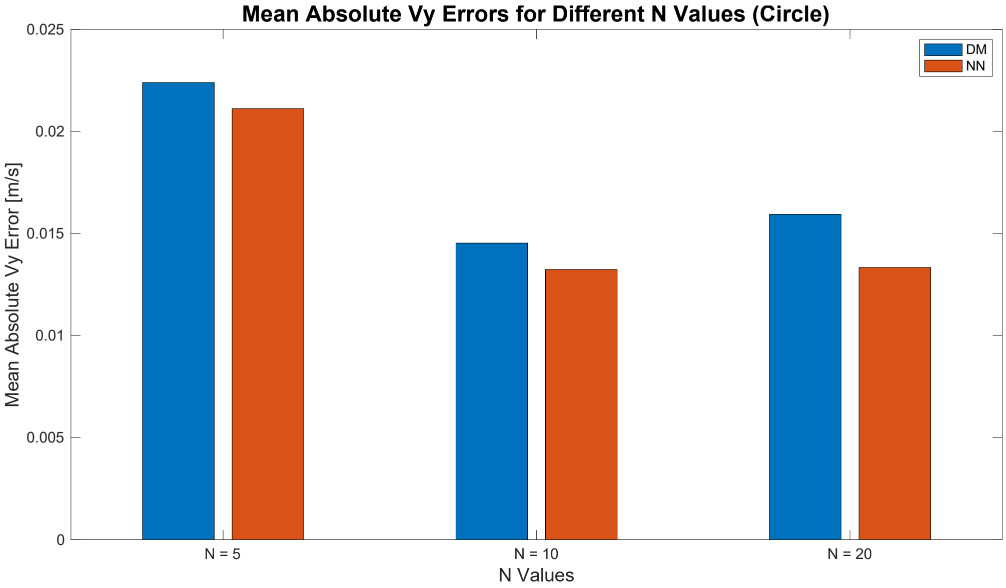


Figure 4-19: The error in v_y at the different prediction horizons for circular

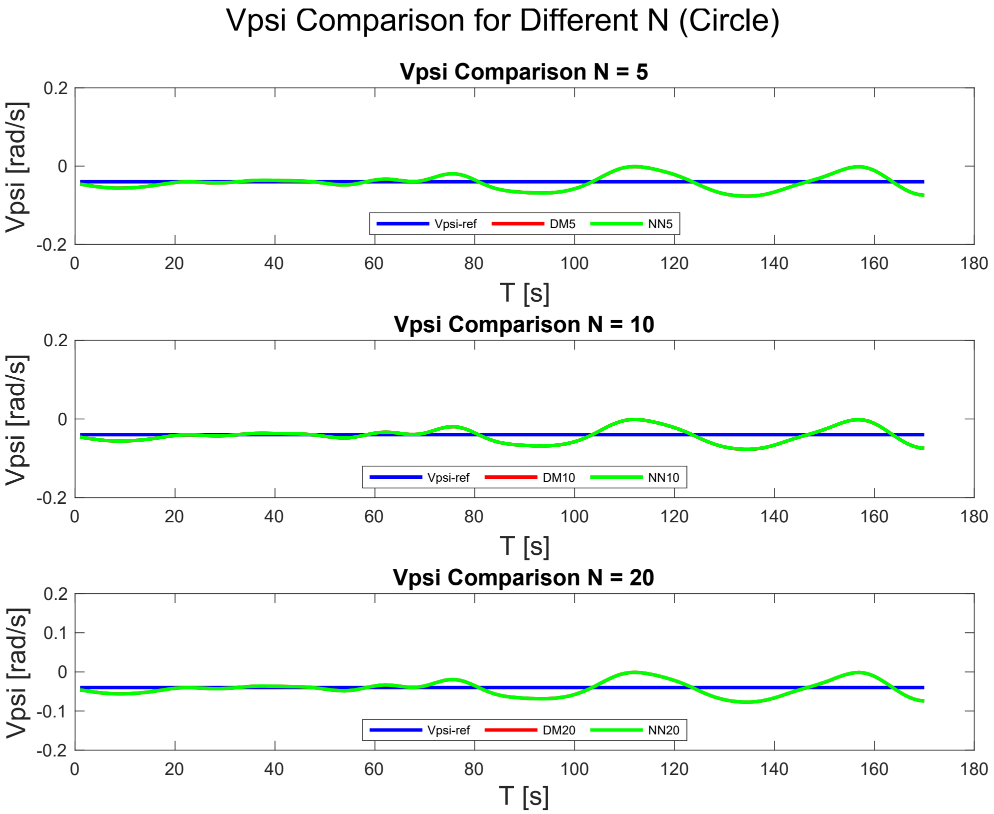


Figure 4-20: Comparison of yaw rate v_ψ for the circular trajectory

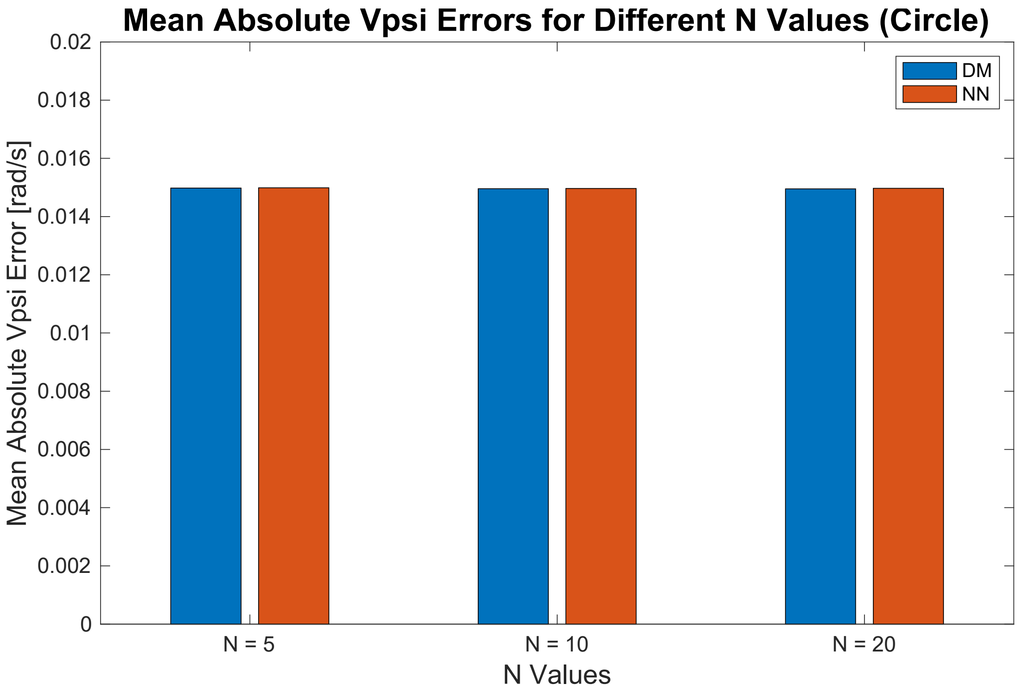
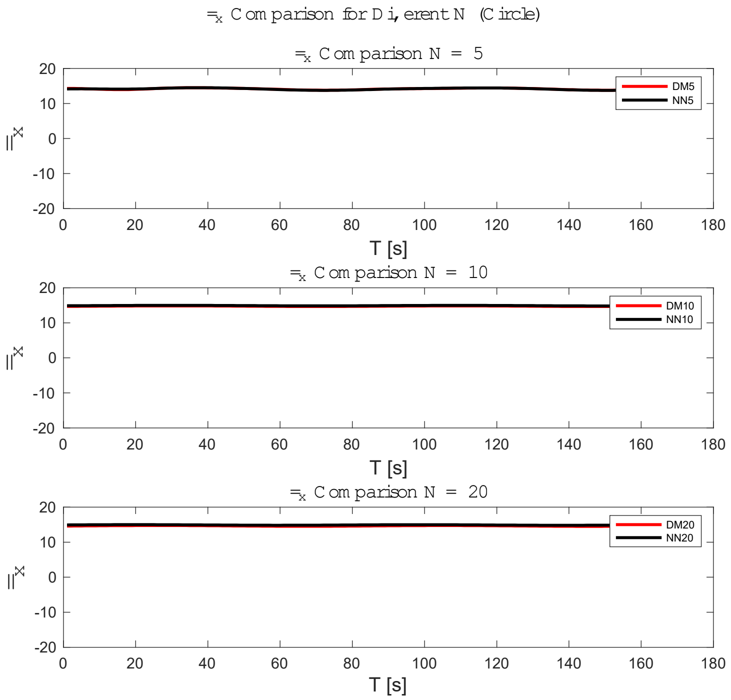


Figure 4-21: The error in v_ψ at different prediction horizons for circular

In this circular trajectory the forces are stable (Figure 4-22) which represents that the control actions applied by the controller with the different models were good enough to keep the ship in the desired path for all the prediction horizons. The forces incorporated with different models for N = 5, 10 and 20 is showed in Figure 4-22.



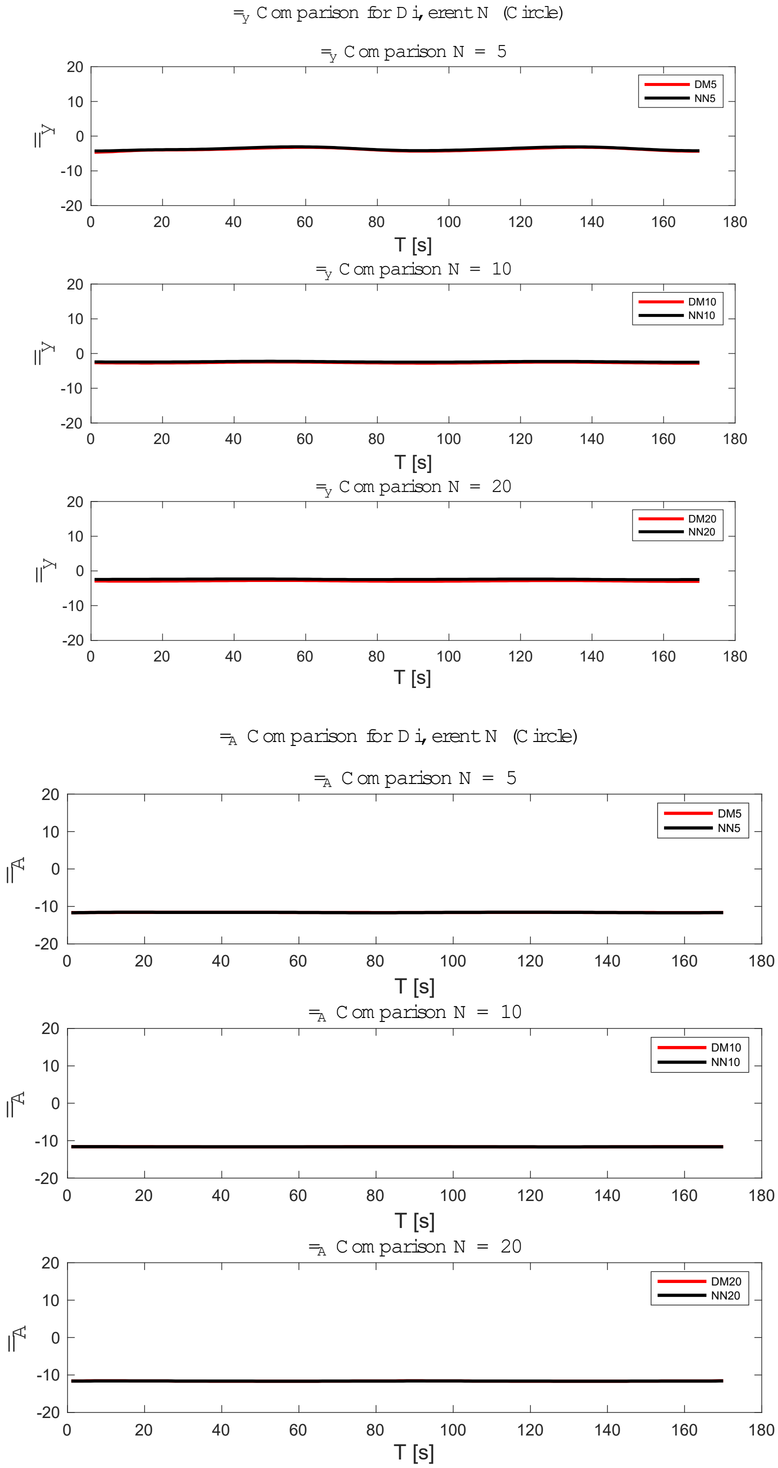


Figure 4-22: Forces applied for the circular trajectory.

4.3.2.2 Figure 8 Trajectory Result Analysis

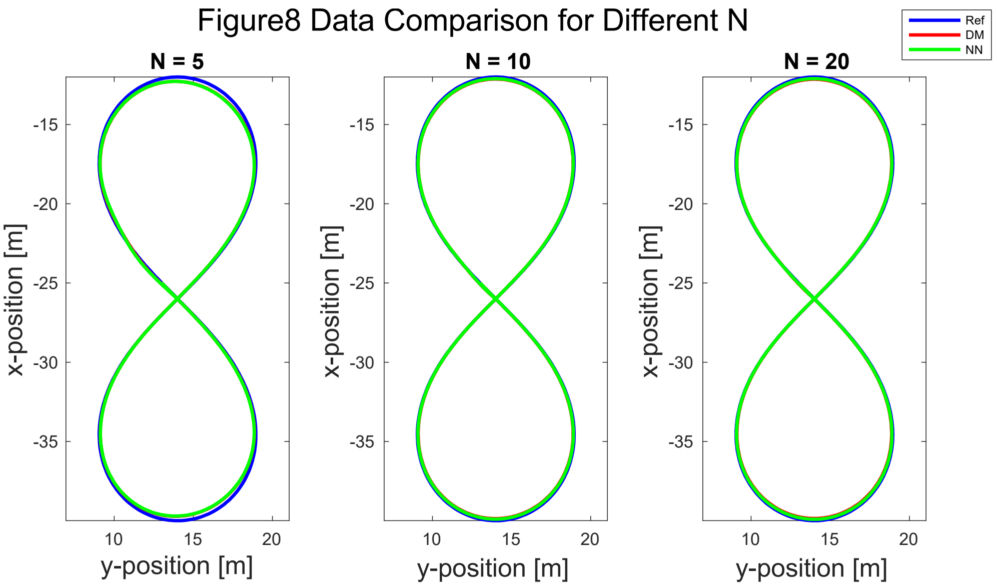
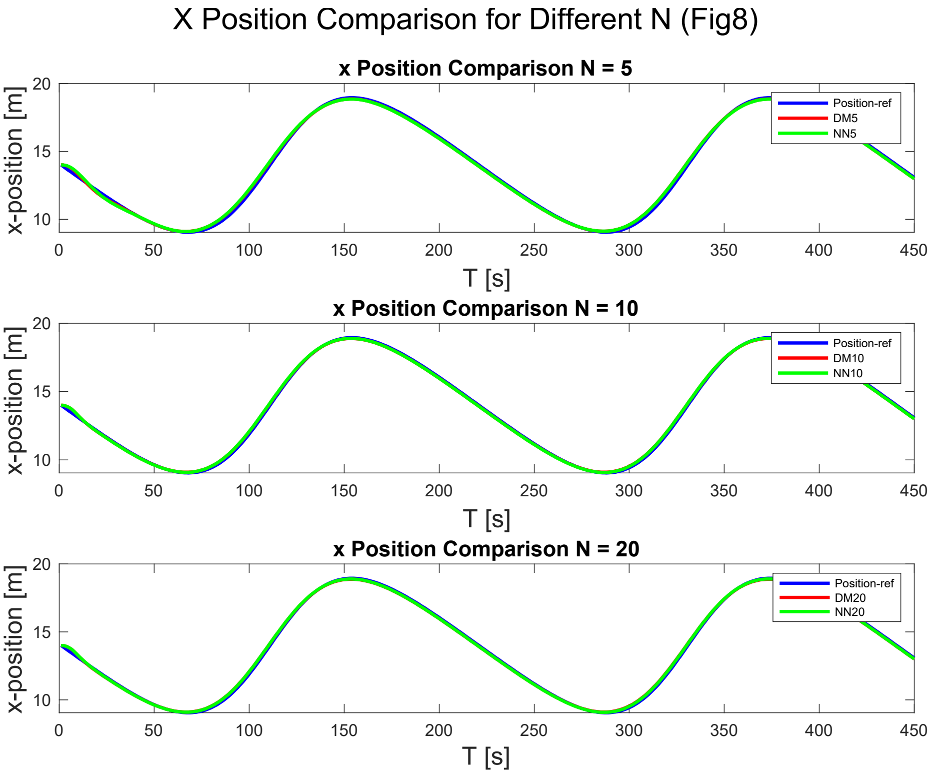


Figure 4-23: Trajectory Tracking of Figure 8 in the presence of simulated disturbance.

Figure 4-23 depicts the trajectory tracking performance of the Figure 8 trajectory.



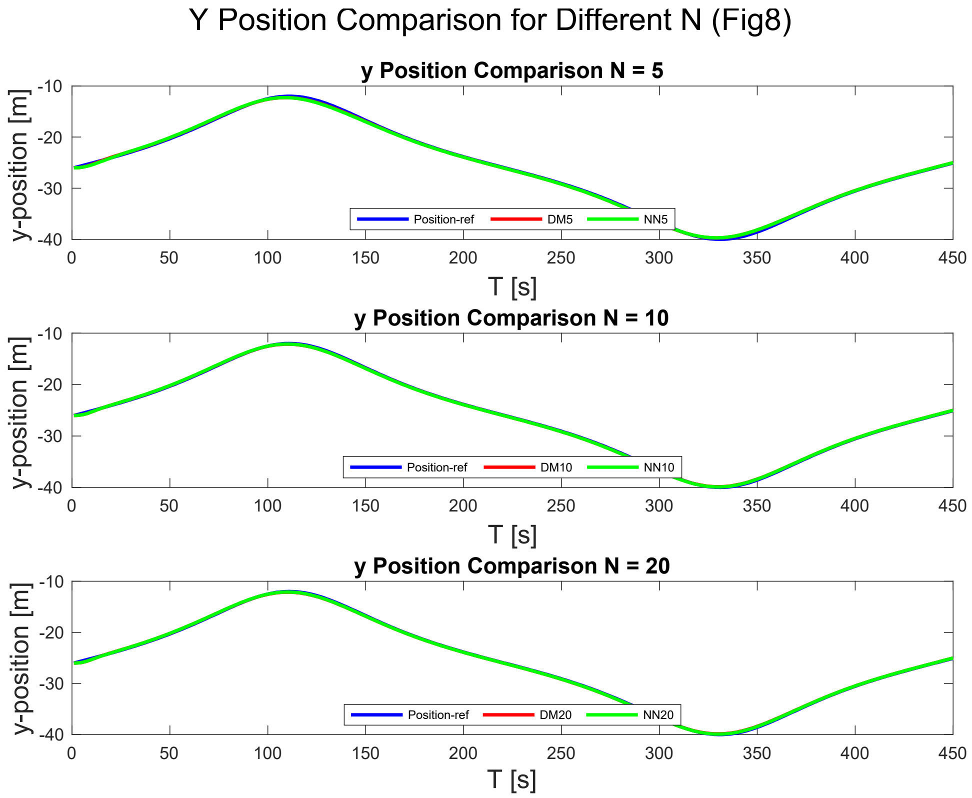


Figure 4-24: Comparison of X and Y positions for the Figure 8 trajectory

Figure 4-24 shows the X and Y position comparison for the two models with different prediction horizon for the shape-8 trajectory. From figures 4-25 to 4-27, we get a clear idea about the performance of the controller for this trajectory. In simulation, the higher prediction horizon works better for the 8 shaped trajectory. For prediction horizon N = 20, the neural network model provided results that are better than the mechanistic model. In comparing the numbers, the neural network model with NMPC works better for this trajectory. In Figures 4-26 and 4-27, the neural network model has the lowest position and heading angle errors. From the visual data, it is evident that the neural network (NN) model is outperforming other models in most aspects of trajectory tracking for an autonomous ship. Specifically, the NN model yields the lowest mean position error at a prediction horizon of N = 20. The NN model again shows superior performance for the heading angle psi error, particularly at the prediction horizon of N = 10. Selecting the best model for ship trajectory tracking depends on the specific performance criteria deemed most critical. If minimizing position error is the priority, especially over longer prediction horizons, the NN is the most effective model. Meanwhile, the NN model at N = 10 would be the optimal choice if heading accuracy is more crucial at mid-range prediction horizons.

Given that the NN model demonstrates strong performance in both position and heading error metrics, it would generally be considered a good candidate for ship trajectory tracking, assuming it also performs well across other essential parameters not depicted in the images. Considering the overall system requirements, including computational efficiency, response time, and robustness to disturbances, is essential before finalizing the controller choice. Selecting a controller requires careful consideration of response time and computational expense. If trajectory tracking is the sole objective, a lower prediction horizon paired with an efficient model may suffice. However, a longer prediction horizon could be necessary for functionalities like obstacle avoidance.

The increased prediction horizon potentially offering a broader scope for adjustments like the purpose of obstacle avoidance, but it can also introduce greater uncertainty as we have seen in the circular trajectory results. A longer prediction horizon can aggregate more errors over time. After looking at the figure eight trajectory other than the mechanistic model, all the other models managed to perform well for position and heading angle effectively. Here, the data driven approach like neural networks offered more flexibility and better handling of the intricacies of a figure eight trajectory. It adapted more successfully to its demand while the performance of the other models was also acceptable.

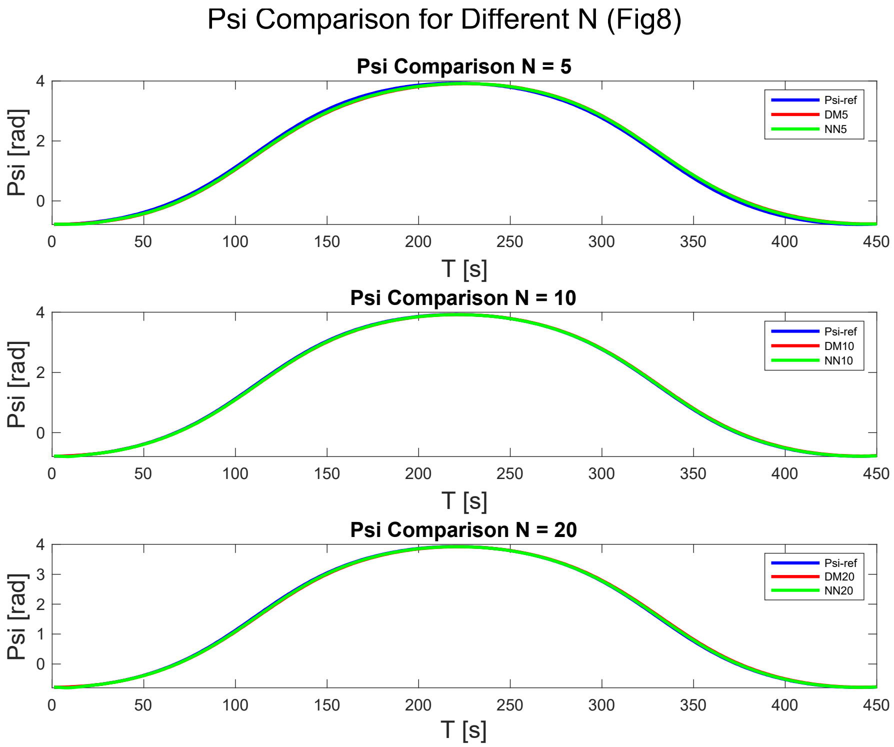


Figure 4-25: Comparison of heading angles for the Figure 8 trajectory

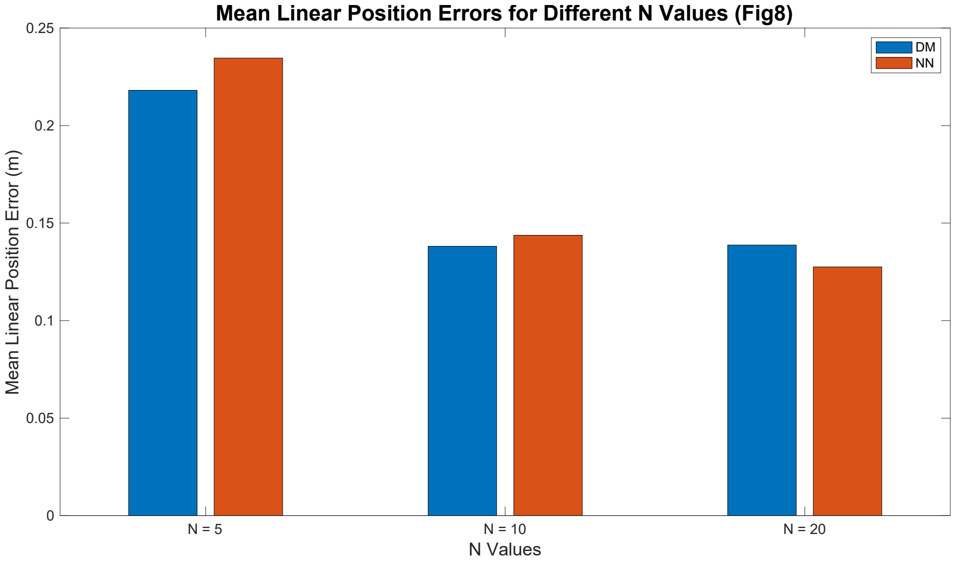


Figure 4-26: Position error at different prediction horizons for Figure 8

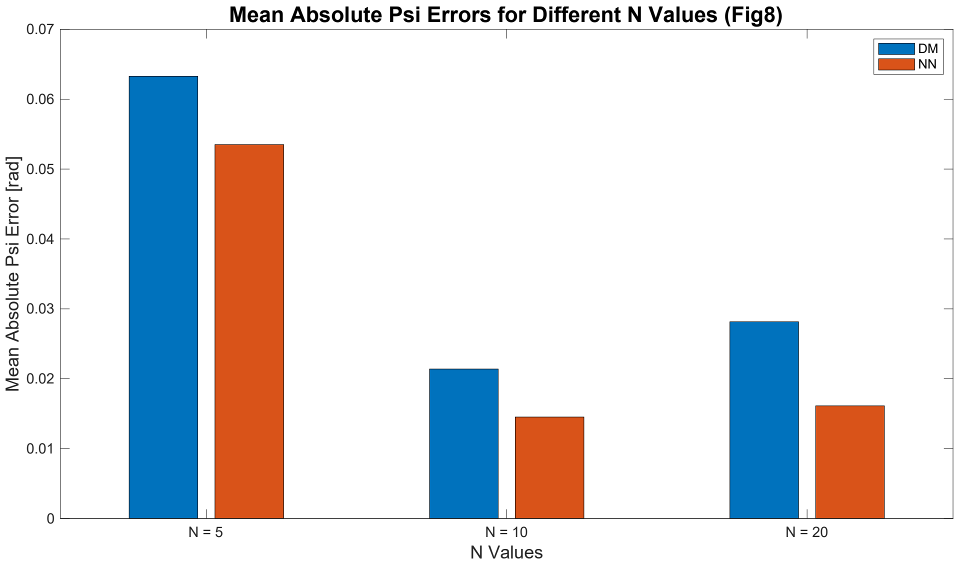


Figure 4-27: Heading angle error at different prediction horizons for Figure 8

We observe the velocity distribution from Figure 4-28 to 4-33. The errors were also calculated here. The surge, sway, and yaw rate remained within the desired range for this simulation—all the models for three different prediction horizons performed well with the inclusion of wave disturbances. The Unscented Kalman Filter was effective in filtering the wave.

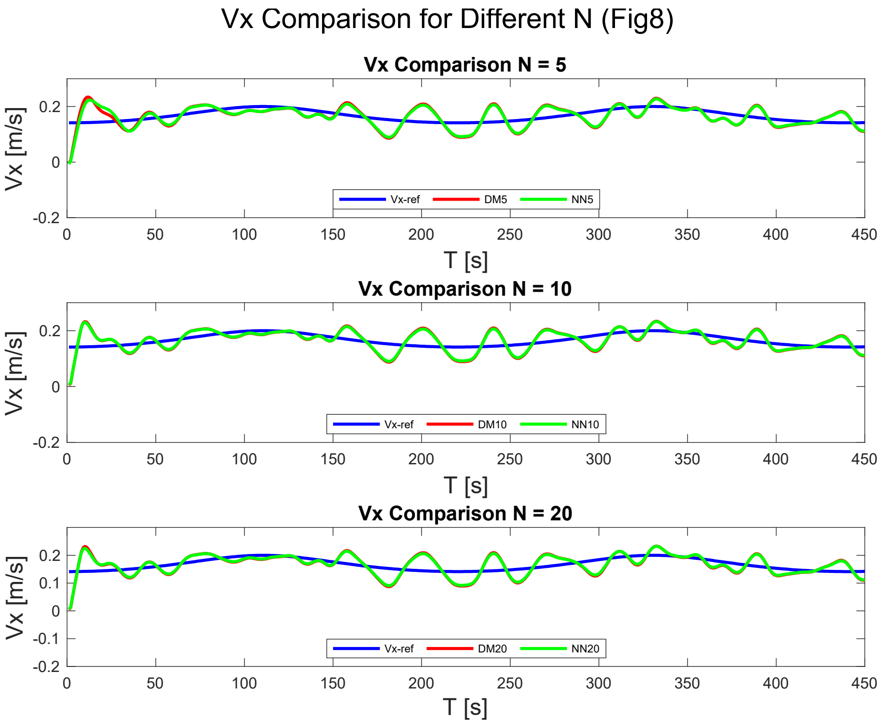


Figure 4-28: Comparison of X-velocity component $v_x$for the Figure 8 trajectory

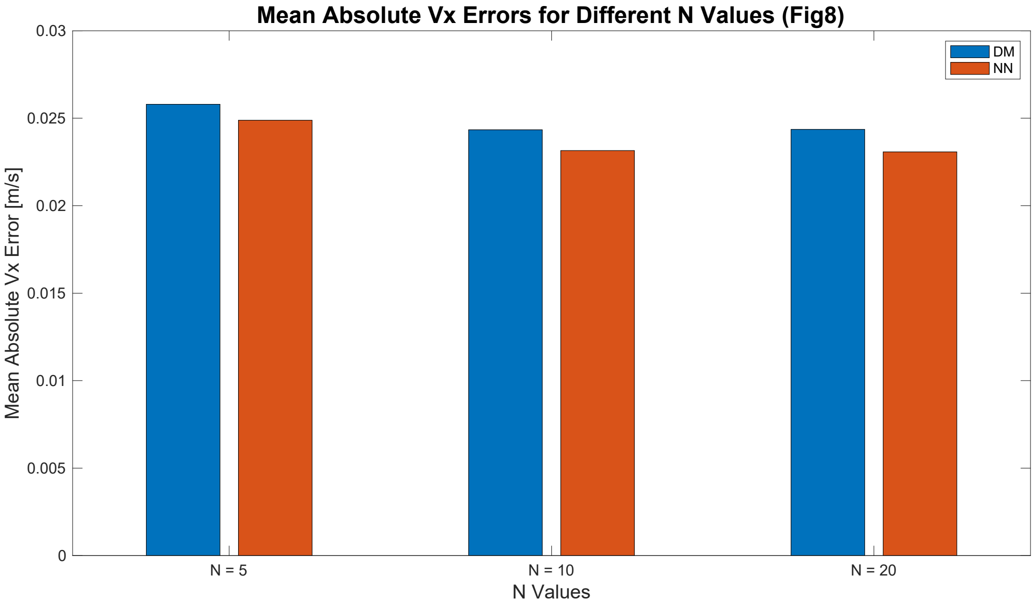


Figure 4-29: The error in v_x at different prediction horizons for Figure 8 trajectory

From the Figures 4-28 and 4-33 it is seen that the difference between performance for the velocity tracking is following a trend. For all the cases v_x, v_y, v_ψ are showing less error for the neural network model at a prediction horizon of N =10.

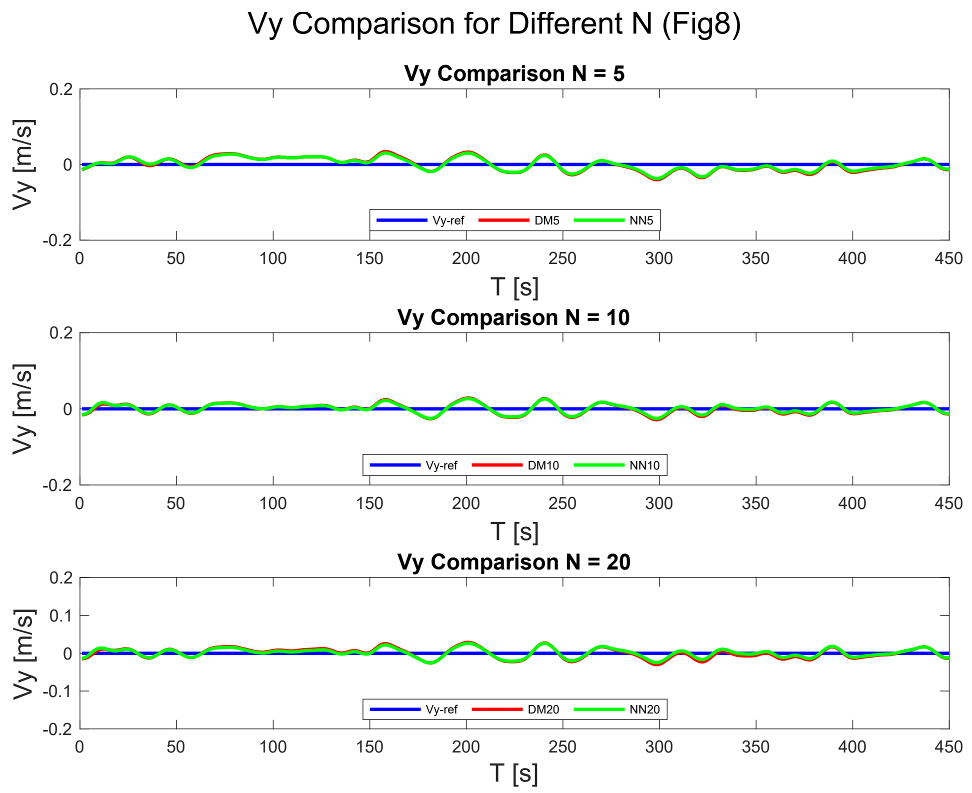


Figure 4-30: Comparison of Y-velocity components v_y for the Figure 8 trajectory

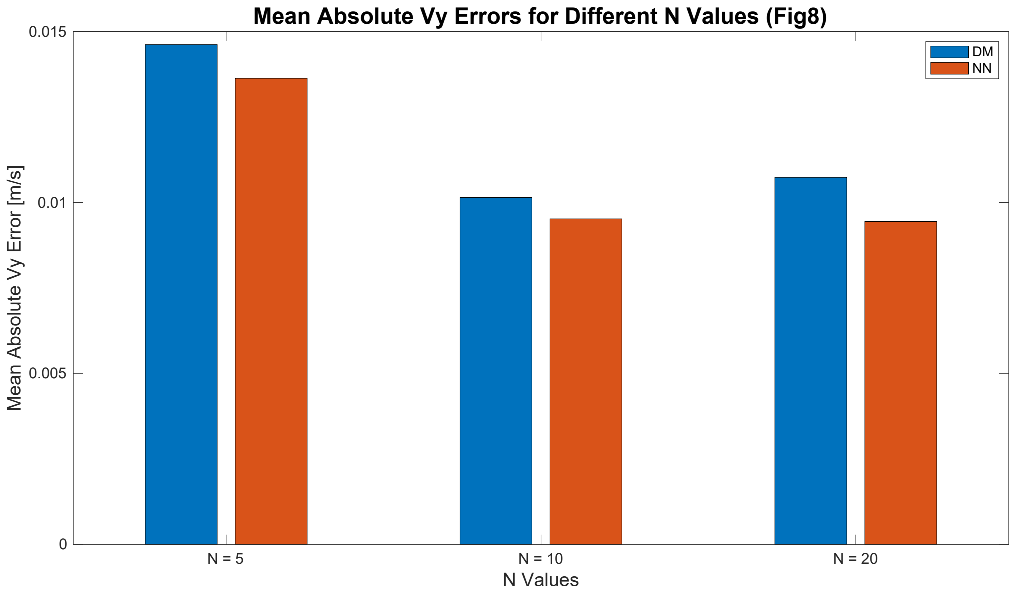


Figure 4-31: The error in v_y at different prediction horizons for Figure 8

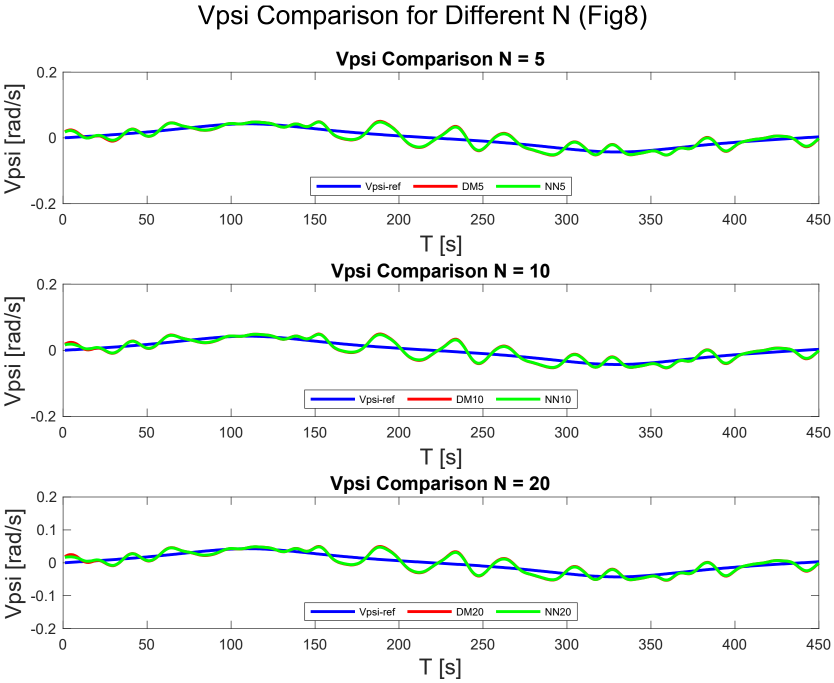


Figure 4-32: Comparison of yaw rate v_ψ for the Figure 8 trajectory

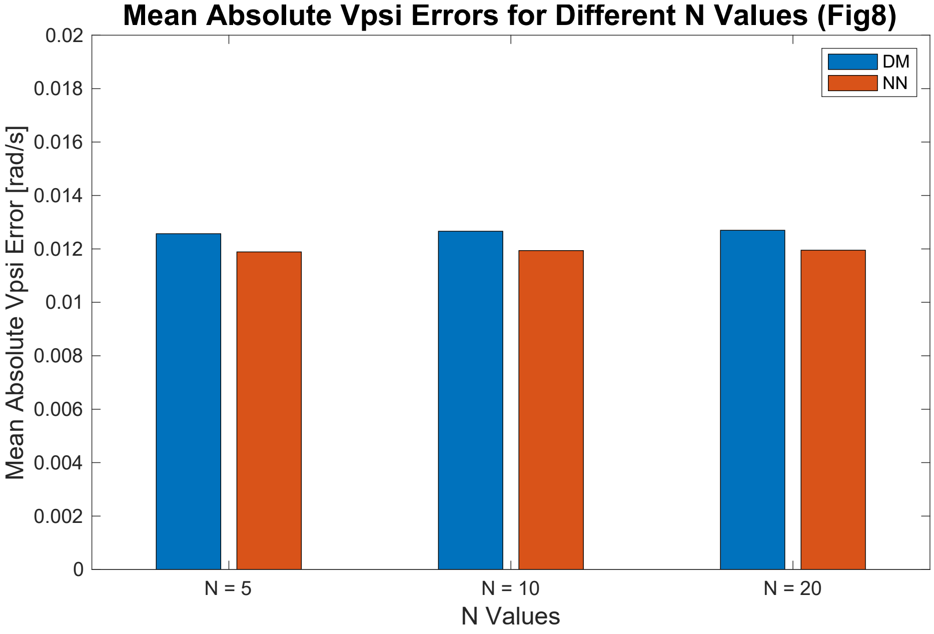
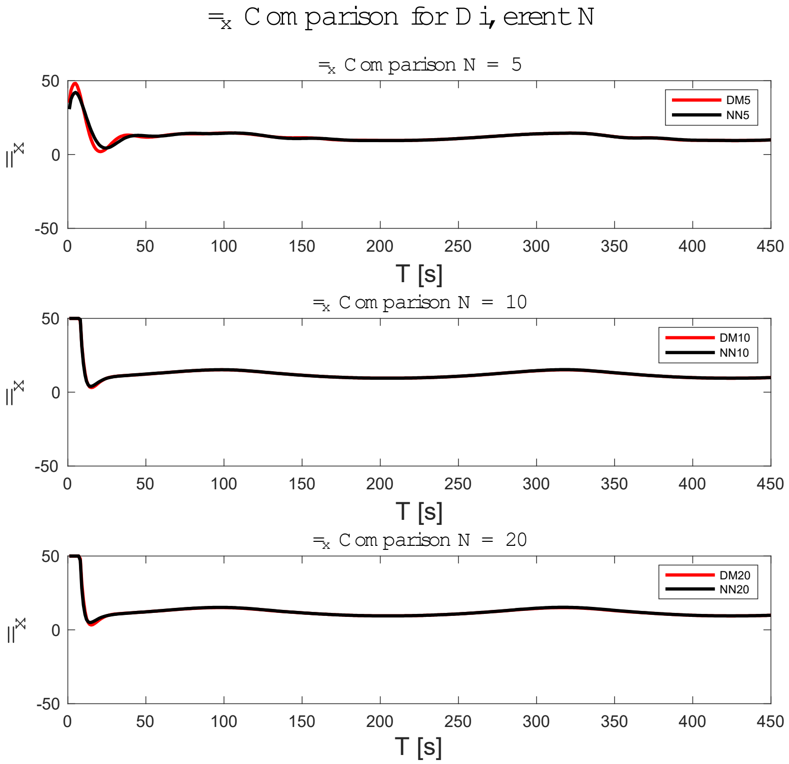


Figure 4-33: The error in v_ψ at different prediction horizons for Figure 8 trajectory

In this trajectory the forces are stable (Figure 4-34) for most of the part for all the models for all the prediction horizons. The τ_x and τ_y shows some peaks at the starting which may be the result of the complexity involved in the tracking of figure eight path. It is also to be mentioned that initial alignment of the ship and the starting from stationary position can likely represent the initial control efforts required to align the ship in the trajectory. Moreover, the control strategy is to converge with the trajectory, and it is programmed to prioritize the quick convergence which may result in higher initial forces to overcome the less accurate predictions at the beginning.



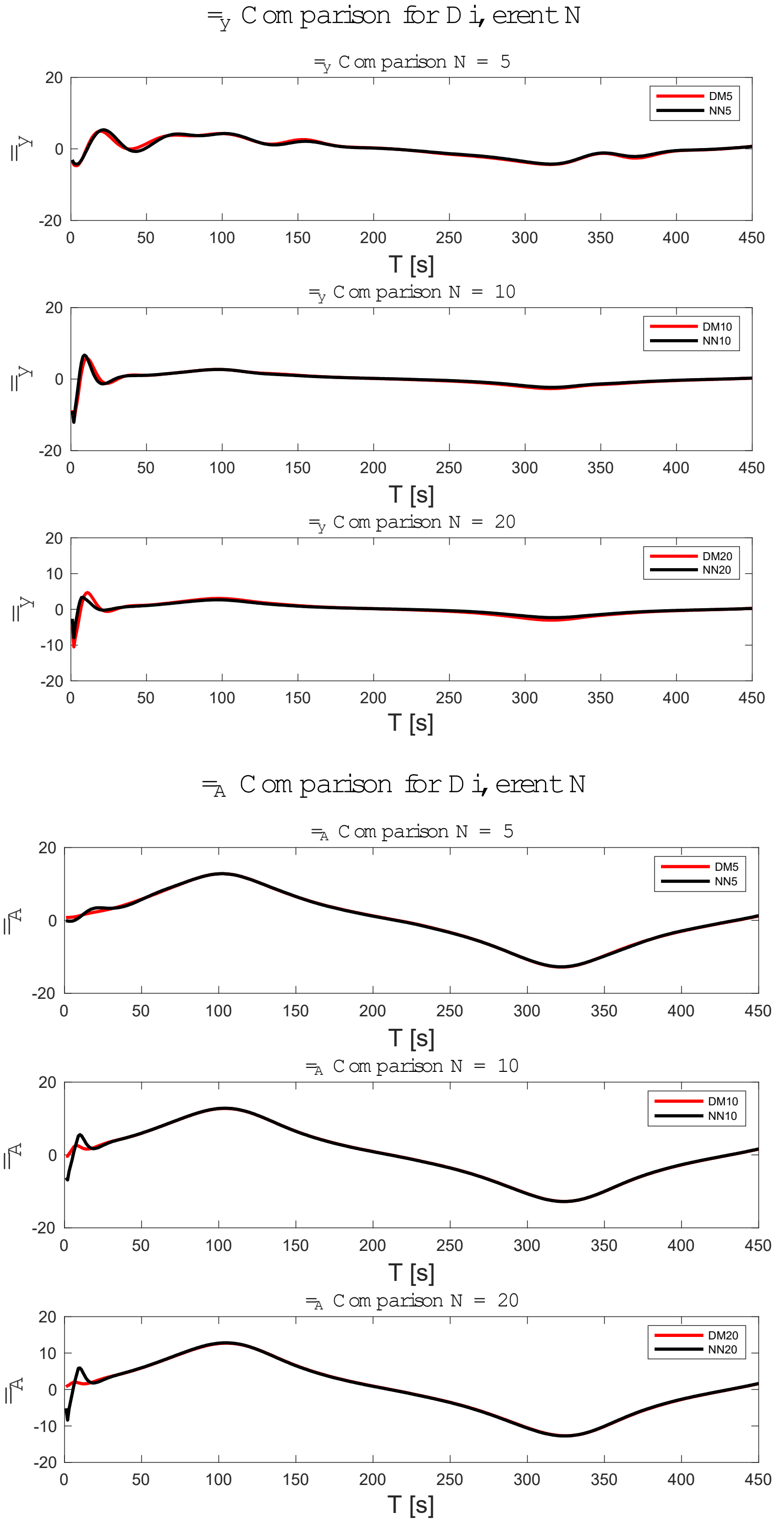


Figure 4-34: Forces applied for the Figure 8 Trajectory.

4.3.3 Experimental Results

For evaluating the NMPC controller the Magne Viking ship model at the National Research Council (NRC), Canada was used as the testing ground. Similar to the simulation studies, the performance of the NMPC with different models was evaluated using the circular trajectory of 5 m radius and a figure 8 shaped trajectory of 7m radius. The experiments were instrumental in assessing controller’s ability to work in real world scenarios. The Figure 4-35 show the experimental setup and the Magne Viking ship model. Here is summary of the testing conditions at NRC:

Ship Model: Magne Viking (1:14 scaled ship model)
Controller Used: NMPC
Software Running Environment: MATLAB 2022
NRC PC used: AMD Ryzen 9 7950X 16 core Processor (4.50 GHz) with 128 GB memory running in Windows 11.
The feedback system used to track the performance of the controller is Qualisys Camera feedback. The test was performed in calm water in the testing basin inside the NRC facility.

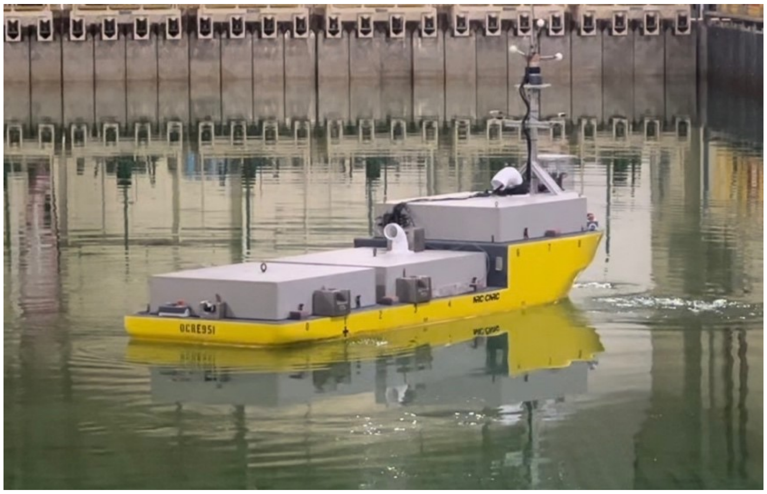


Figure 4-35: Magne Viking Ship model at NRC, Canada

4.3.3.1 Experiment with Circular Trajectory

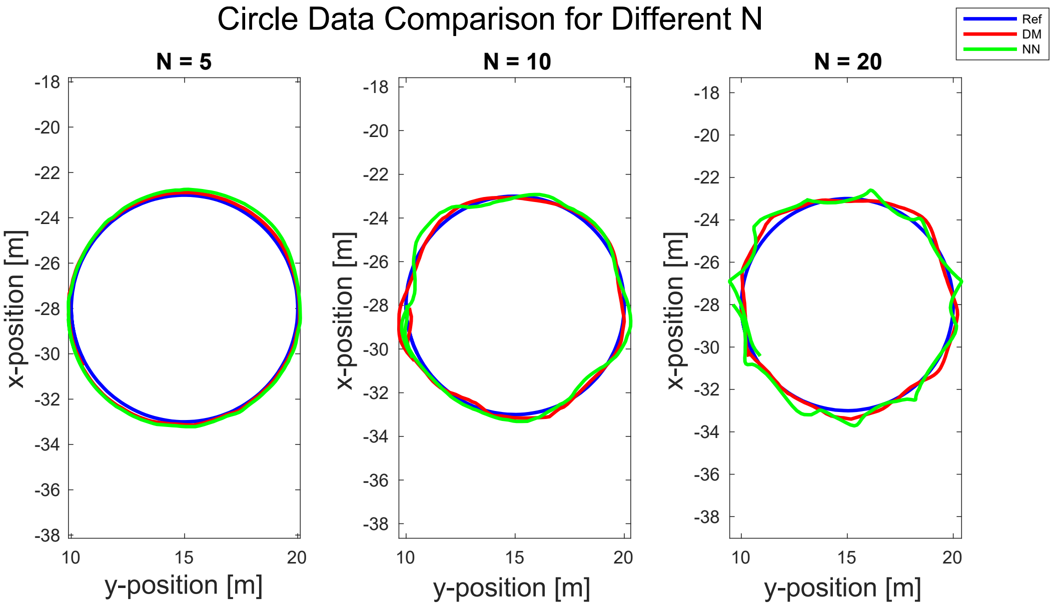
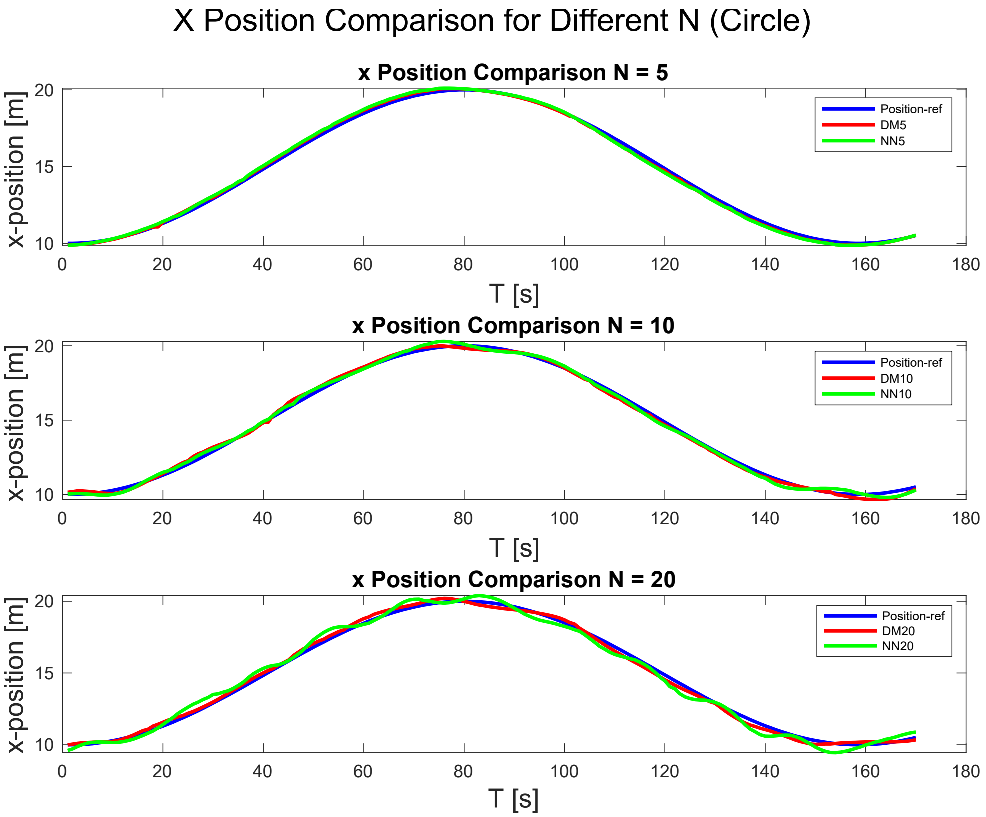


Figure 4-36: Results of trajectory tracking of a circle at the experimental setup at NRC, Canada.

Figure 4-36 and 4-37 shows the results of the trajectory tracking with all the NMPC models incorporated. There are visible differences between the simulation and the experimental results which is expected to happen when the controller is tested in a real-world scenario. The model accuracy, sensor noise and accuracy, computational limitations and many uncertainties can cause these differences. The decreasing performance with an increasing prediction horizon may be due to a poor ship model.


X Position Comparison for Different N (Circle)
x Position Comparison N = 5
x Position Comparison N = 10
x Position Comparison N = 20
x-position [m]
T [s]
10
15
20
0
20
40
60
80
100
120
140
160
180
Position-ref
DM5
NN5
DM10
NN10
DM20
NN20

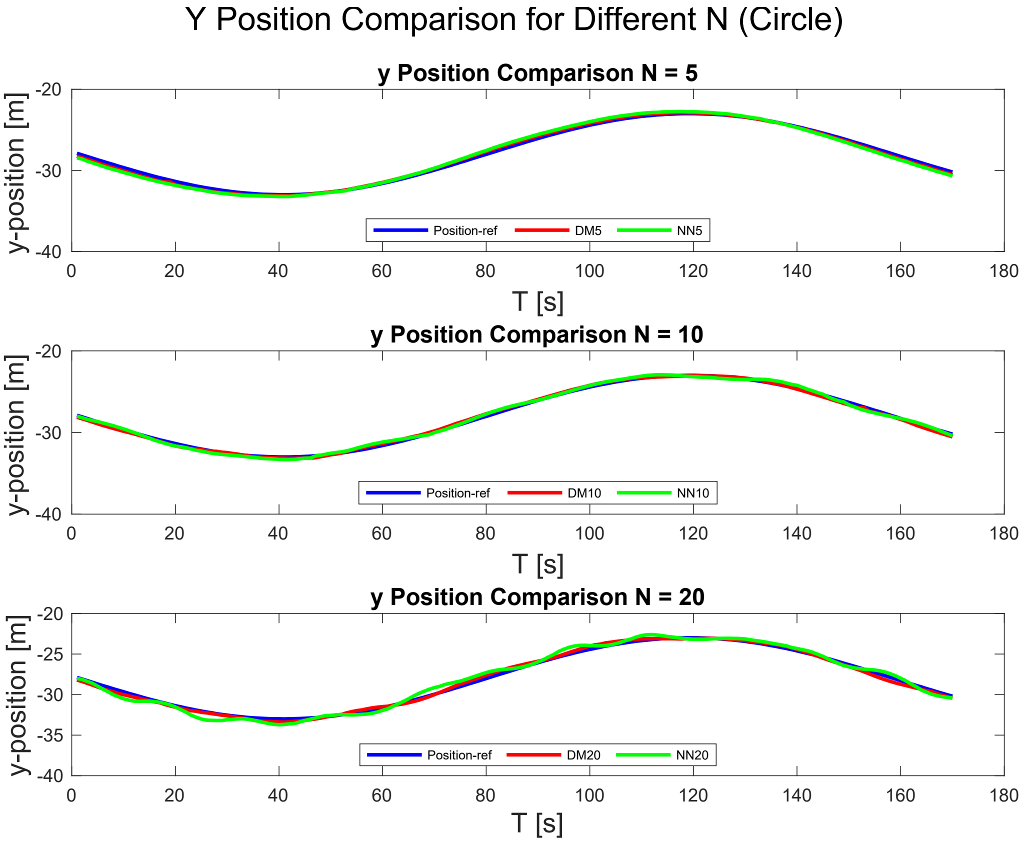

Y Position Comparison for Different N (Circle)
y Position Comparison N = 5
y Position Comparison N = 10
y Position Comparison N = 20
y-position [m]
T [s]
-20
-25
-30
-35
-40
0
20
40
60
80
100
120
140
160
180
Position-ref
DM5
NN5
DM10
NN10
DM20
NN20

Figure 4-37: Comparison of X and Y positions for the Figure 8 trajectory

Figure 4-37 shows the X and Y position comparison for all two models with different prediction horizons for the circular trajectory. From the Figures 4-36 to 4-40 it is observed that the models worked well for the low and moderate prediction horizons. For prediction horizon N = 5 and N = 10 all the models showed good results. Amongst them the mechanistic model worked better for minimizing the position errors (Figure 4-39). For the heading angle comparison, it is noticeable that for both the prediction horizons (5 and 10) the mechanistic model was showing better heading control than the other two models. The higher prediction horizons were neither suitable for position nor the orientation. So, for low to moderate prediction horizons any of the models can work with the Magne Viking ship according to the specific goal to be achieved by the ship. The mechanistic model and the neural network model worked well with the lower prediction horizon which means that the NMPC controller was able to properly utilize the model to get a good prediction. It may be mentioned that when the prediction horizon is increased small inaccuracies in the modeling build up over time. This phenomenon is more dominant in the case of experimental setup. In the simulation the errors were identical for both mechanistic and neural network model. But in the experiment the neural network was showing poor performance.

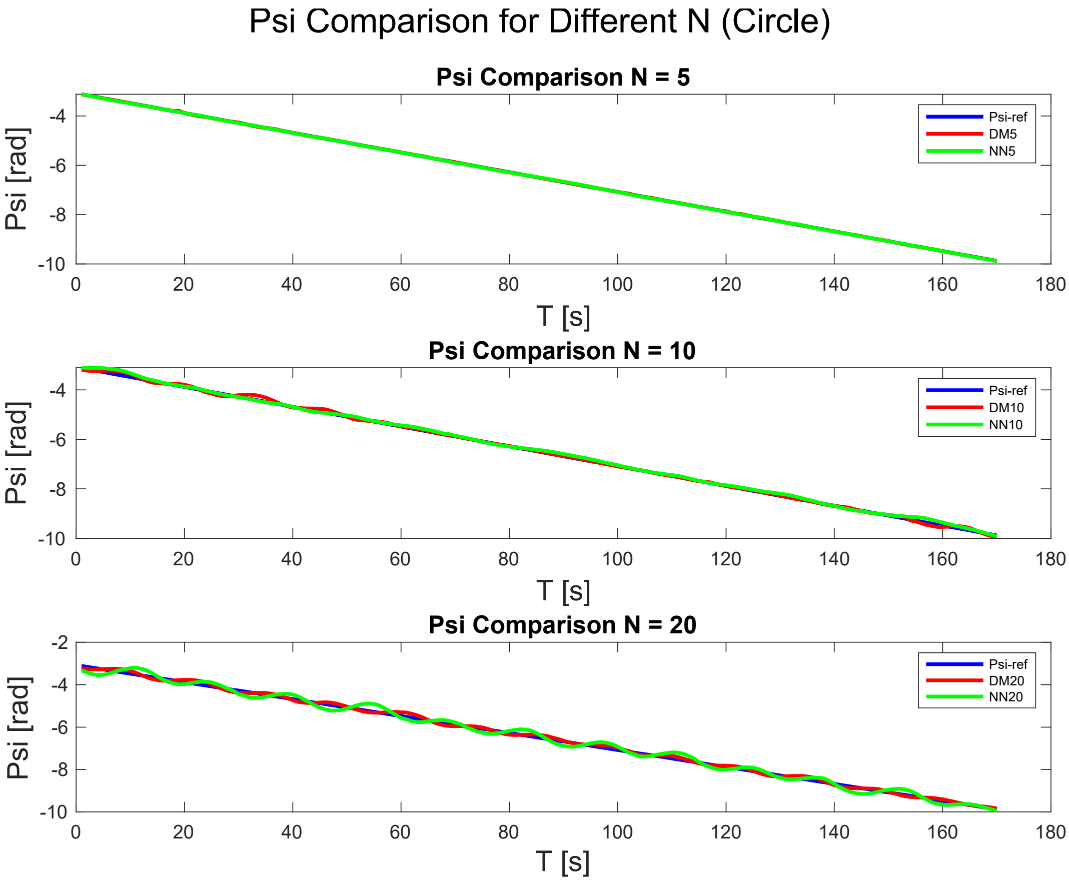


Figure 4-38: Comparison of heading angles for the circular trajectory

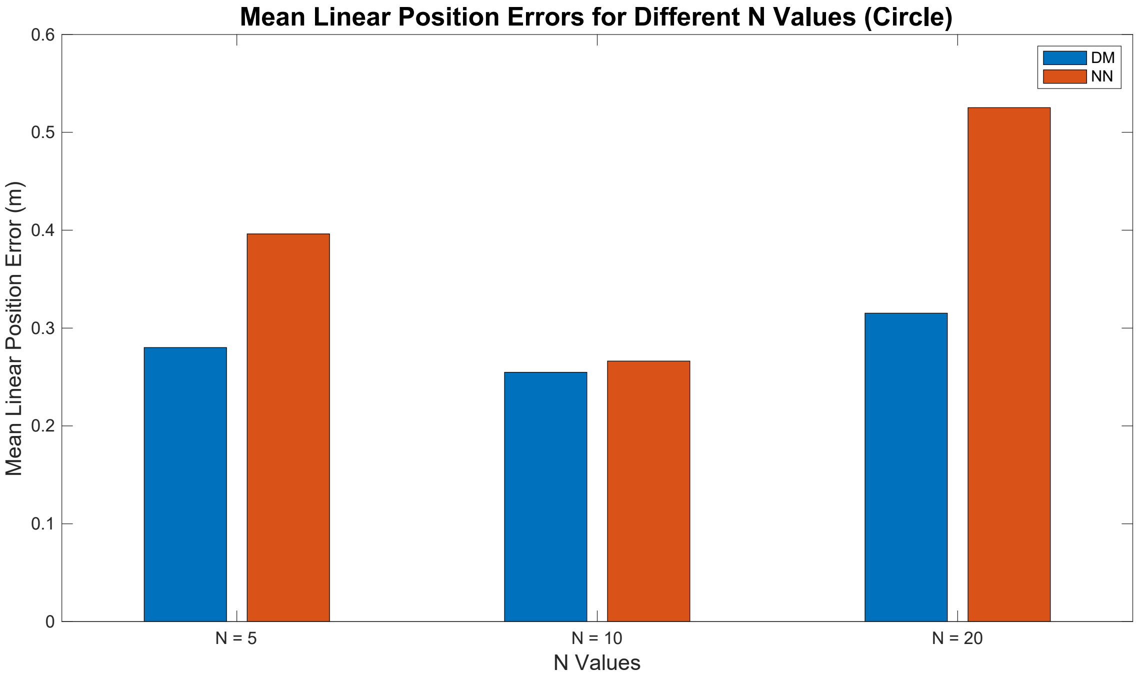


Figure 4-39: Position error at different prediction horizons for circular trajectory

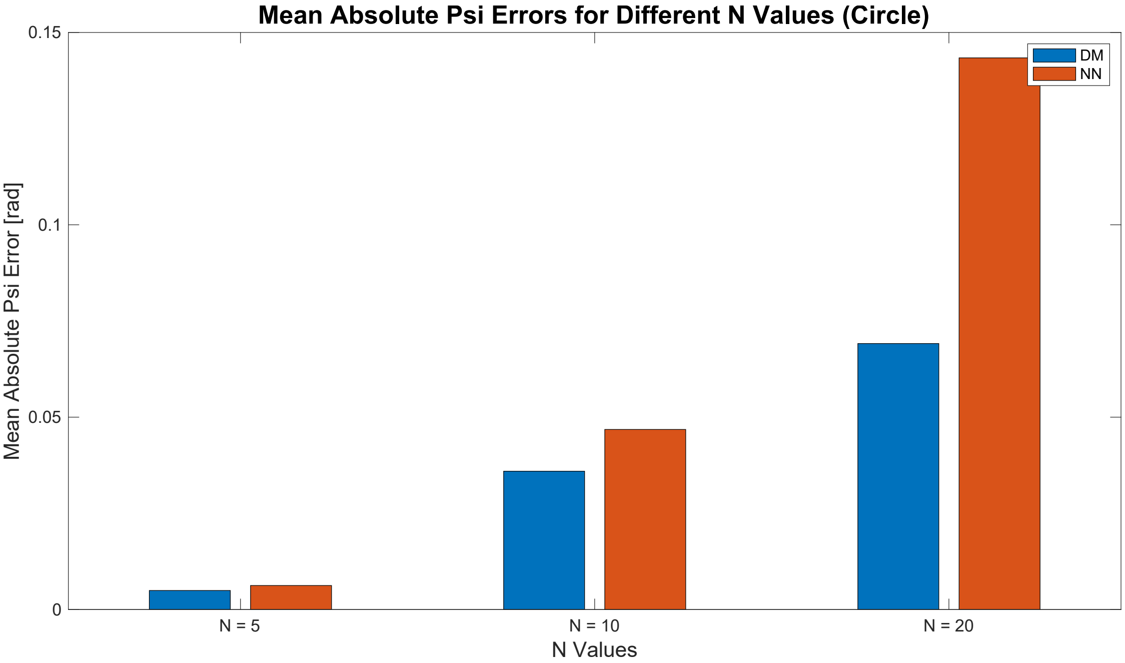


Figure 4-40: Heading angle error at different prediction horizons for circular trajectory

From the Figures 4-41 to 4-45 we have some ideas about the velocity distribution while the experimental setup was running. The errors were also calculated which is shown in the figures 4-42 to 4-46. All the velocities surge, sway and yaw rate have small error level. But for all the cases the lower prediction horizon was working better for all the models. For example, in the figure 4-41 and 4-42 the v_x error from the reference values were deviated much in the prediction horizon 20 than the prediction horizon 5 or 10. From the other figures representing the v_y and v_ψ the neural network and mechanistic both were working well for the lower prediction horizon (5 and 10).

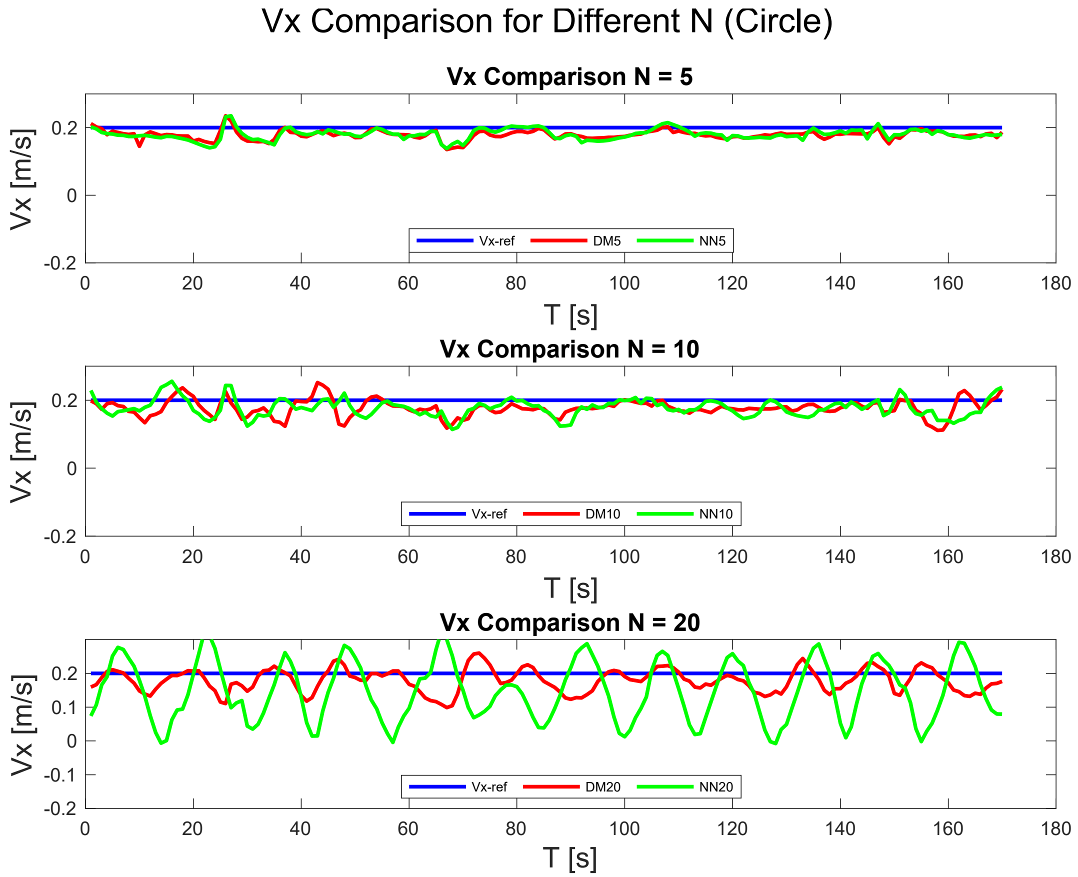


Figure 4-41: Comparison of X-velocity component $v_x$for the circular trajectory

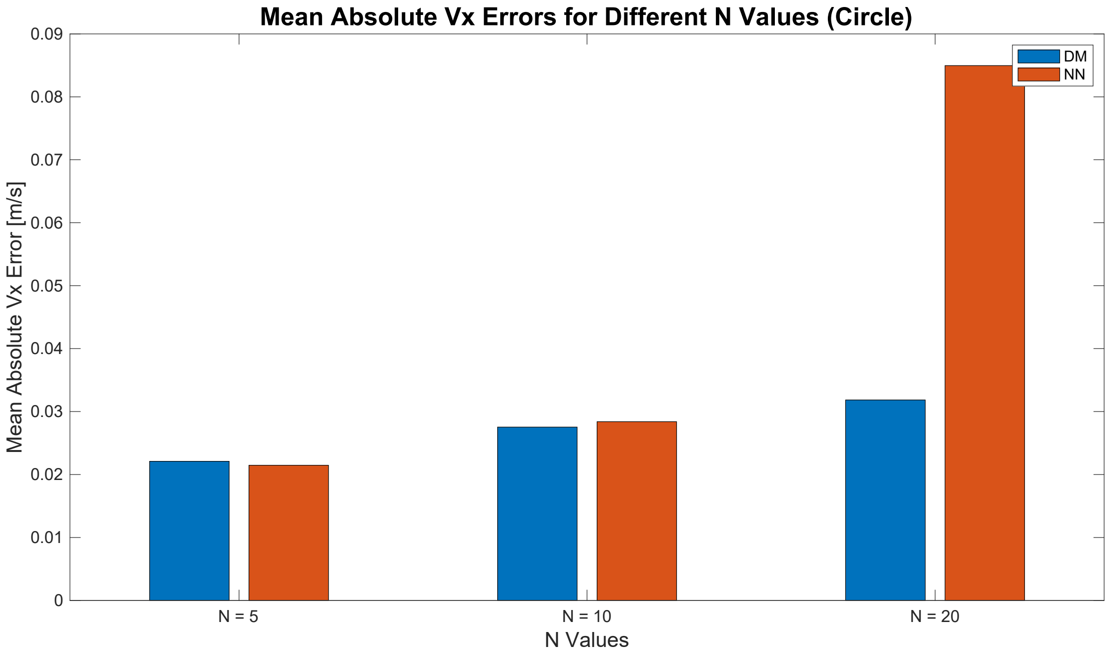


Figure 4-42: The error in v_x at different prediction horizons for circular trajectory

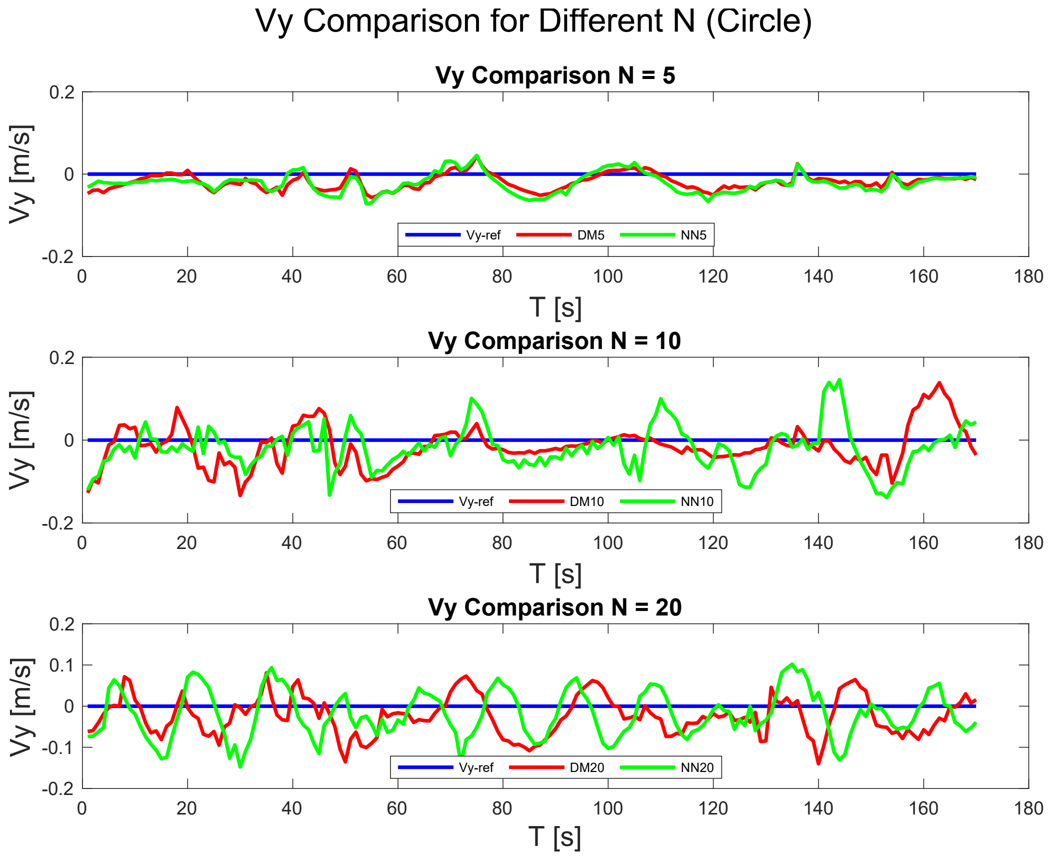


Figure 4-43: Comparison of Y-velocity components v_y for the circular trajectory.

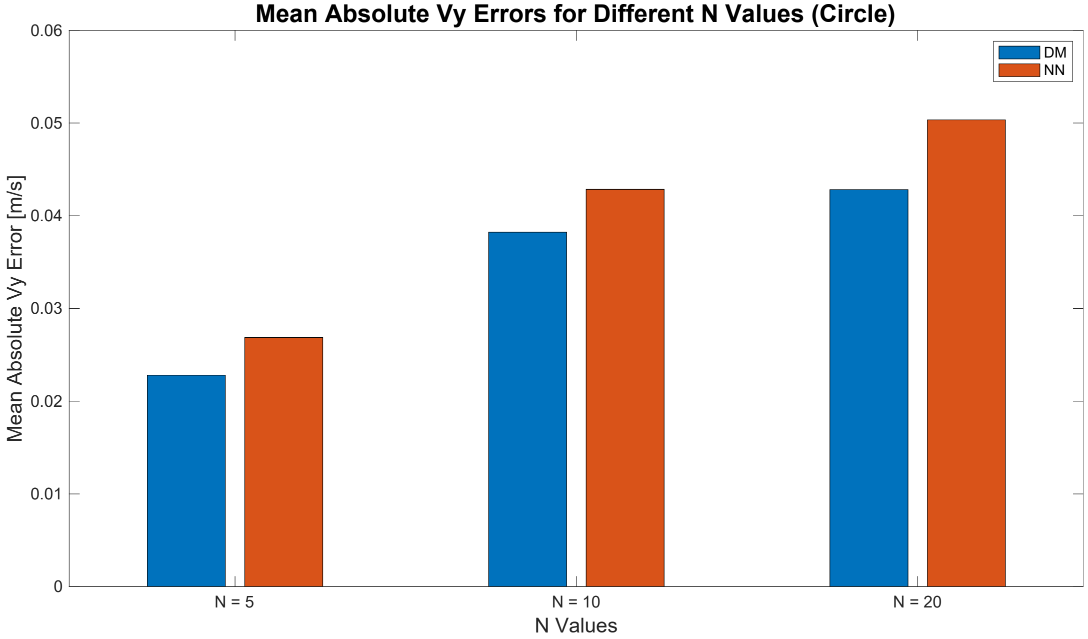


Figure 4-44: The error in v_y at different prediction horizons for circular trajectory.

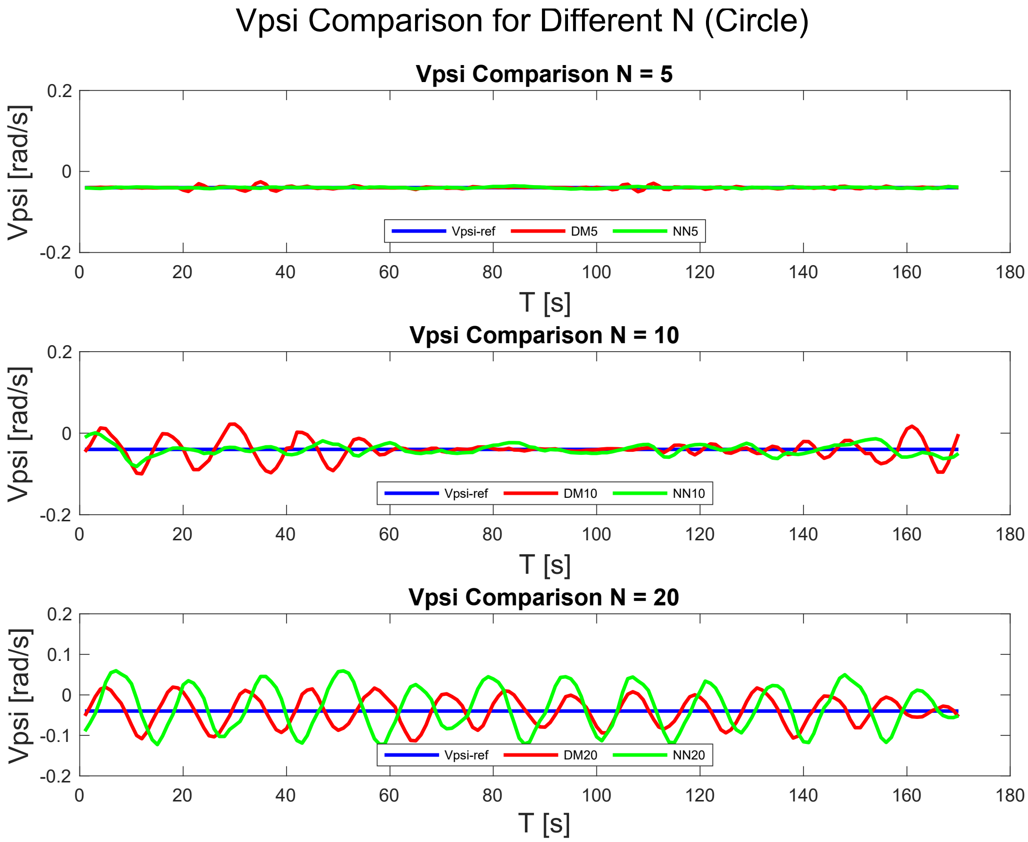


Figure 4-45: Comparison of yaw rate v_ψ for the circular trajectory

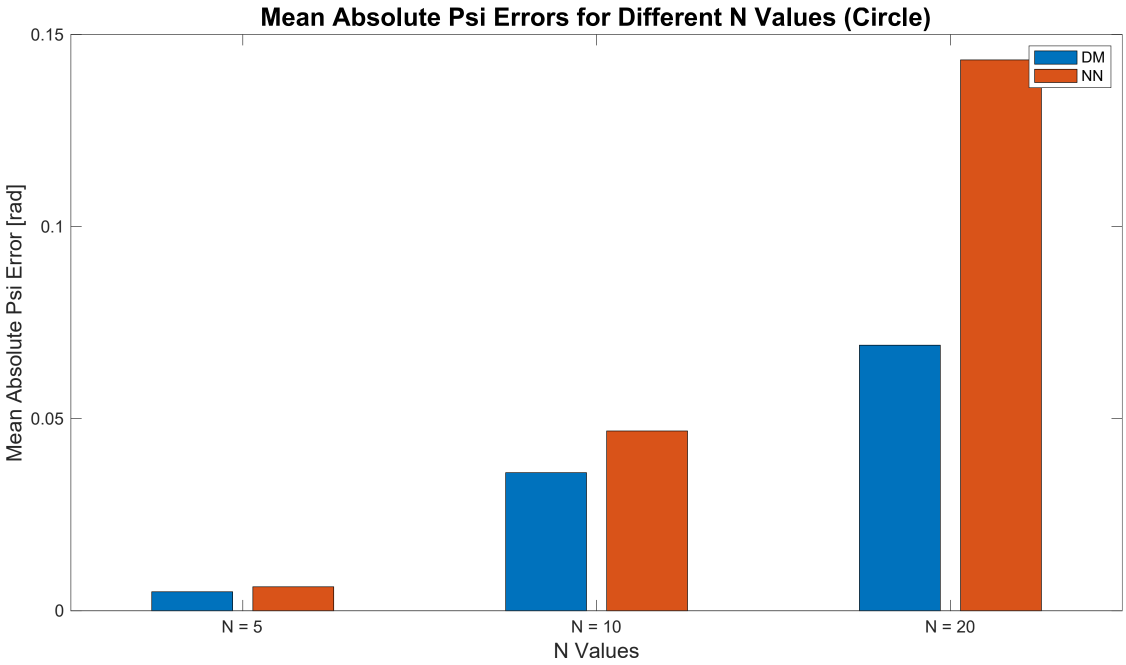
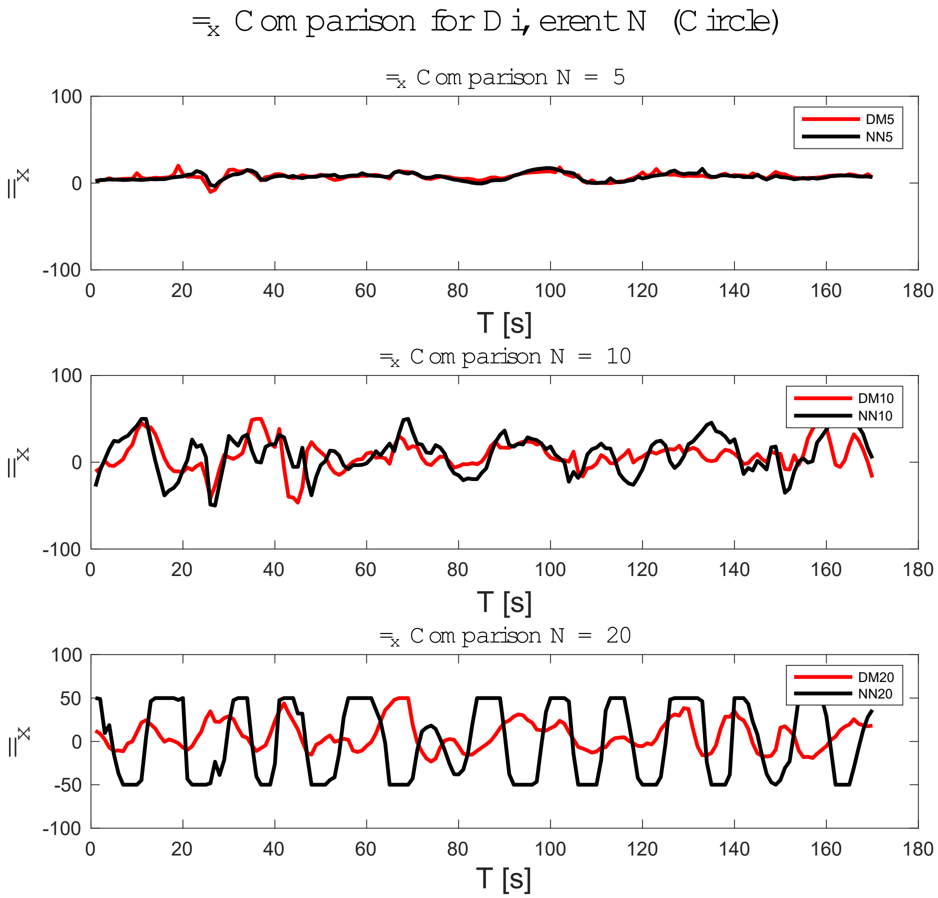
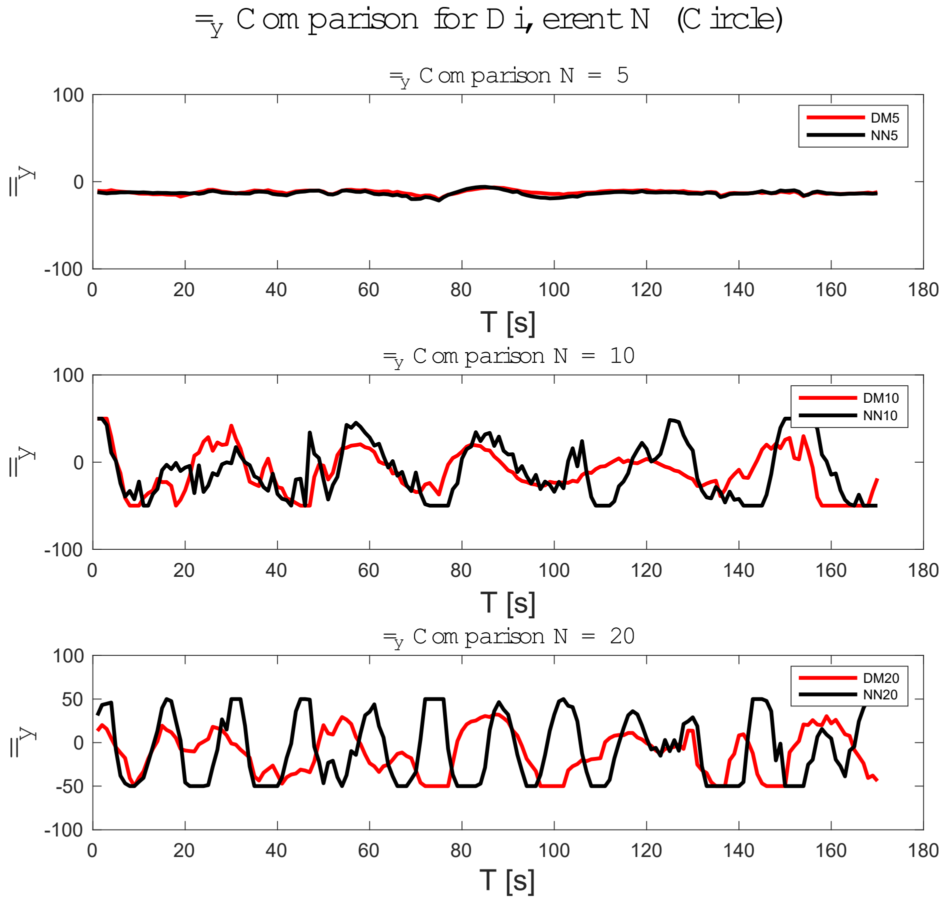


Figure 4-46: The error in v_ψ at different prediction horizons for circular trajectory

In this trajectory the forces are more stable (Figure 4-47) for the lower prediction horizons. When the higher prediction horizon is used the thruster forces were saturated.




=$_y$ Comparison for Different N (Circle)
=$_y$ Comparison N = 5
100
0
-100
=$_y$
0
20
40
60
80
100
120
140
160
180
T [s]
DM5
NN5
=$_y$ Comparison N = 10
100
0
-100
=$_y$
0
20
40
60
80
100
120
140
160
180
T [s]
DM10
NN10
=$_y$ Comparison N = 20
100
50
0
-50
-100
=$_y$
0
20
40
60
80
100
120
140
160
180
T [s]
DM20
NN20

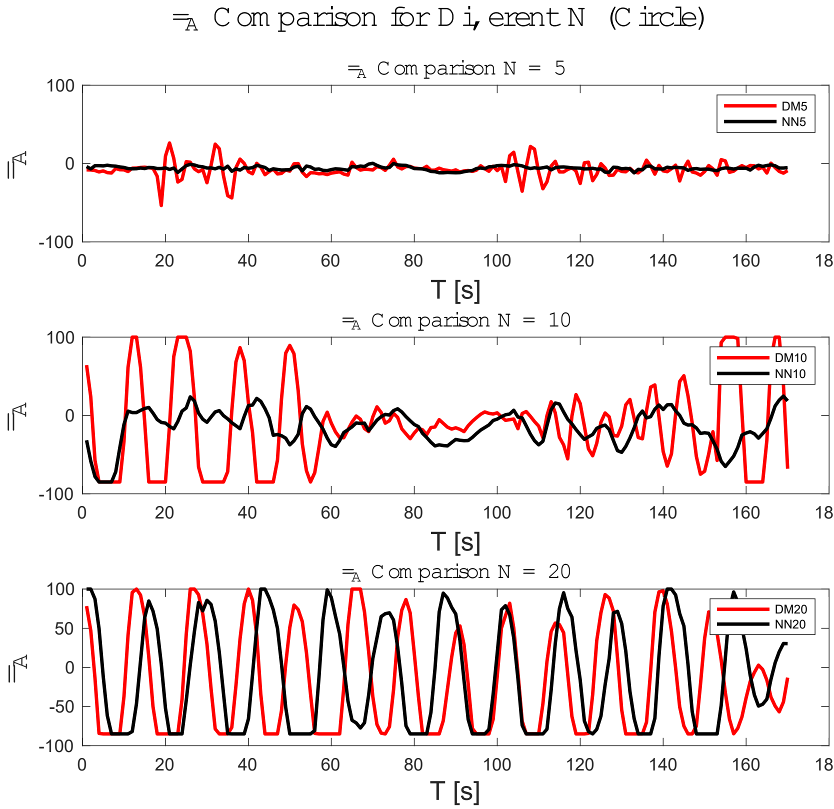


Figure 4-47: Forces applied for the circular trajectory.

Table 4-8 shows the summary of the errors for the different models for the different ship maneuvering parameters. Table 4-8 shows the range of the forces required to keep the ship on its desired trajectory.

Table 4-8: Error comparisons for circular trajectory for all the models

Parameters	**Mechanistic model**			**Neural Network model**		
Prediction Horizon, N	5	10	20	5	10	20
Sum of root mean squared position error, e_{xy} (m)	0.28	0.2548	0.3152	0.3963	0.2663	0.5252
Heading Angle error, e_{ψ} (rad)	0.05	0.036	0.0691	0.0062	0.0468	0.1434
Surge velocity error, e_{v_x} (m/s)	0.0221	0.0275	0.0318	0.0215	0.0284	0.0850
Sway Velocity	0.0228	0.0382	0.0428	0.0269	0.0428	0.0503

error, e_{v_y} (m/s)						
Yaw rate error, e_{v_ψ}(rad/s)	0.0020	0.0164	0.0307	0.0011	0.0104	0.0475
Average Execution Time per iteration,T_s(s)	0.2182	0.4985	0.8272	0.2568	1.0471	2.0262

Table 4-9: comparison of the forces for different prediction horizon for the circular trajectory

Models used with NMPC		**Surge force (τ_x) range (N)**			**Surge force (τ_y) range (N)**			**Yaw moment (τ_ψ) range (N-m)**		
		N =5	**N =10**	**N=20**	**N =5**	**N =10**	**N=20**	**N =5**	**N =10**	**N=20**
Mechanistic Model	min	-10.19	-46.69	-23.24	-20.38	-50	-49.99	-53.68	-85	-85
	max	20.12	50	49.99	-6.59	50	32.35	26.34	100	99.99
Neural Network Model	min	-3.4727	-49.99	-49.99	-21.55	-50	-49.99	-11.89	-85	-85
	max	17.21	50.00	50.00	-5.87	50.00	50.00	0.3819	28.85	99.99

4.3.3.2 Experiment with Figure 8 Trajectory

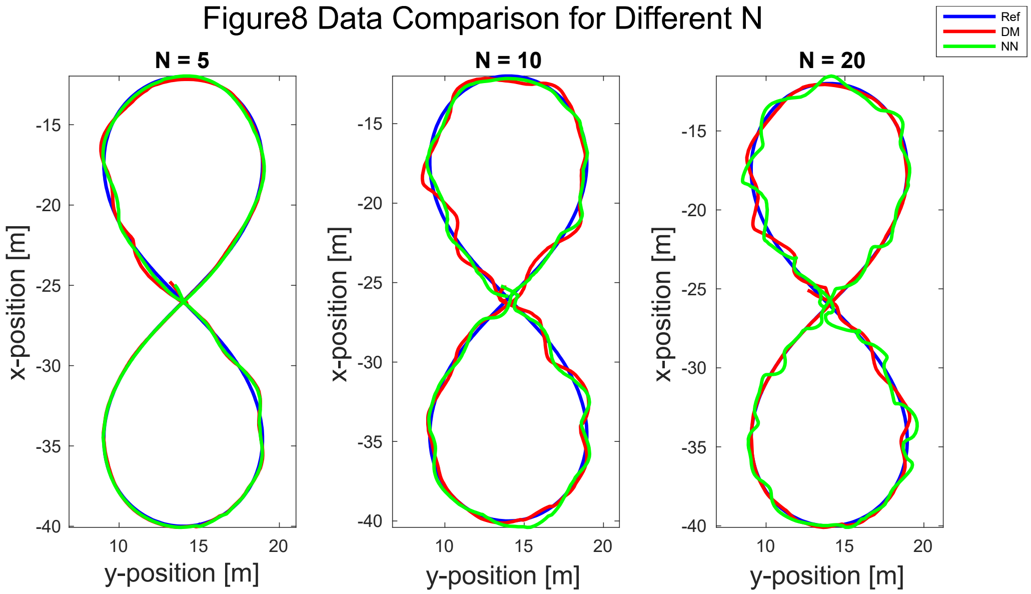


Figure 4-48: Trajectory Tracking of Figure 8 at experimental setup at NRC, Canada.

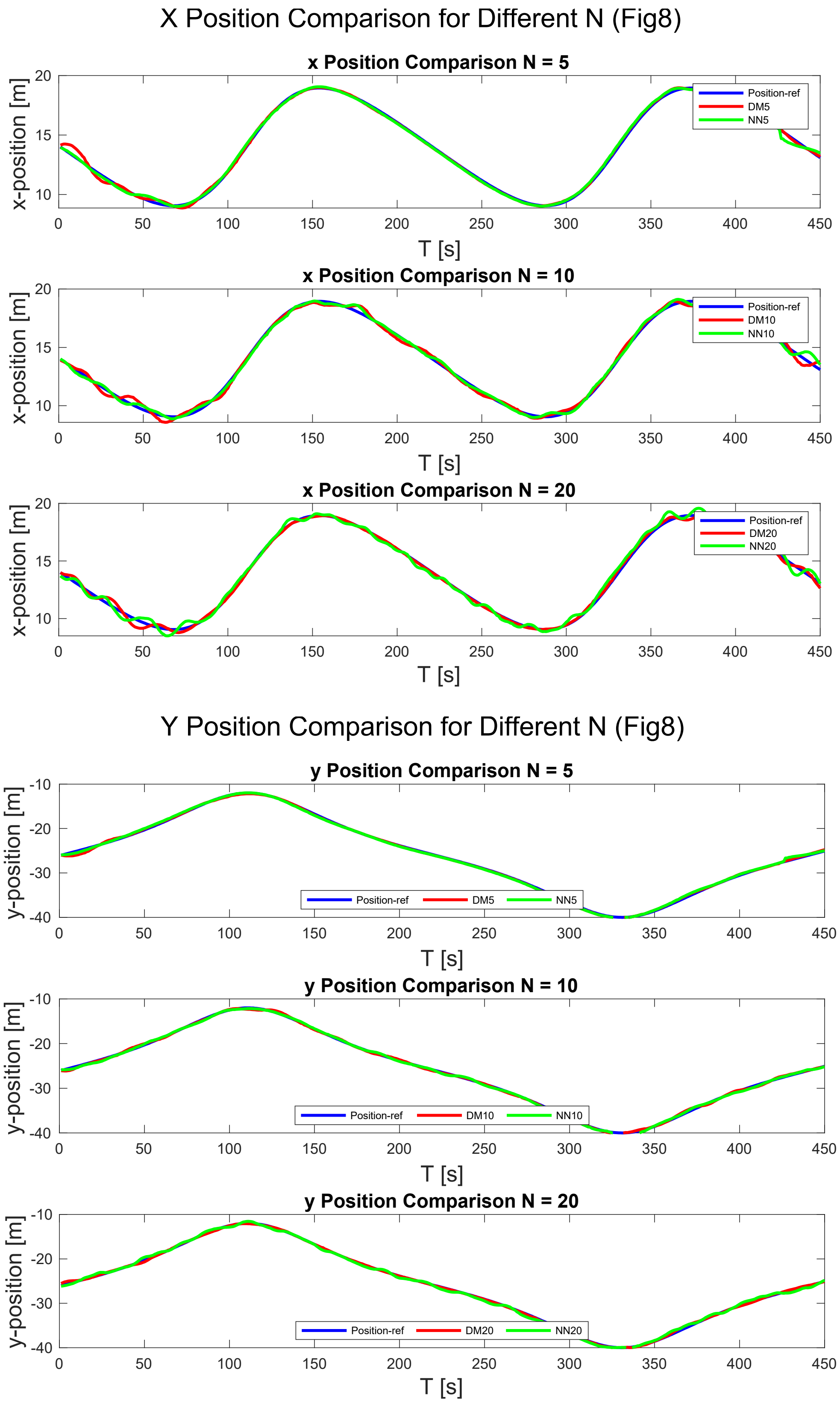


Figure 4-49: Comparison of X and Y positions for the Figure 8 trajectory

Figure 4-48 shows the trajectory tracking results for the figure 8 shaped trajectory. In the figure 4-49 the X and Y position tracking for the different prediction horizons with three different models are shown. From the error diagram it is noticed that like the previous cases the mechanical or neural network model with a lower prediction horizon works well. On the other hand, the higher prediction horizon is more prone to higher errors for these models. For position and heading angle the neural network and the mechanistic model are well suited in this case when a prediction horizon of 5 is being used. If we observe the Figure 4-52 the mechanistic model is showing a lesser error for both the position and heading angle calculation. But it may not be suitable because of the less consistent tracking. The sum of error is smaller but the tracking performance from the figure 4-48 shows that it has a lot of points which are out of the track which is not useful.

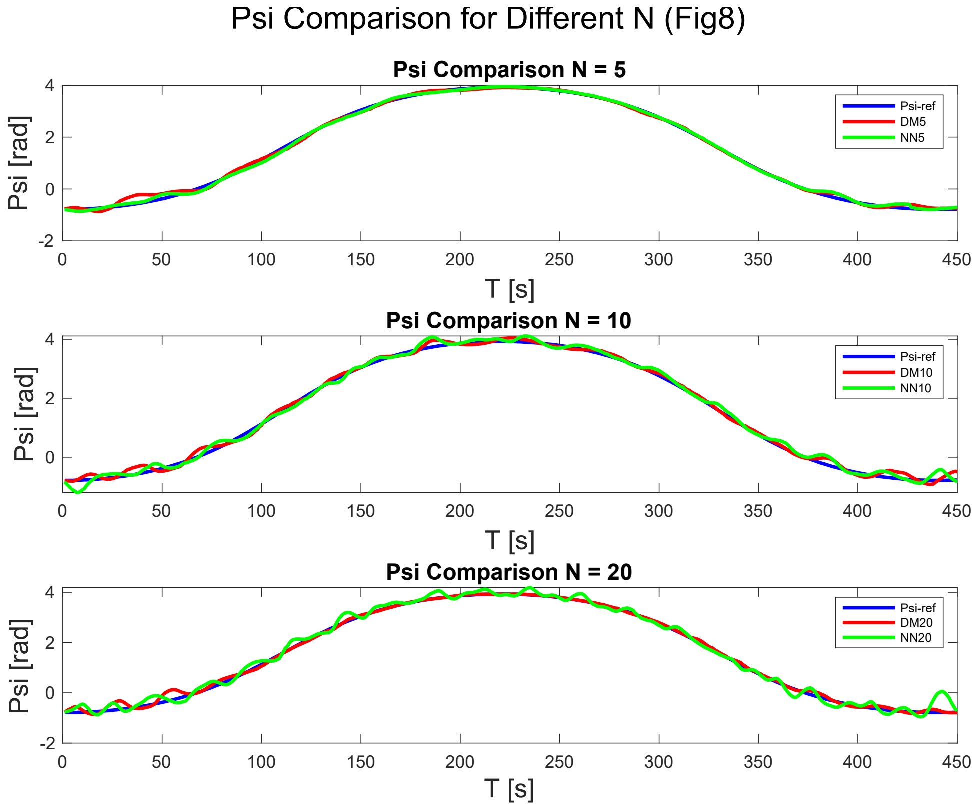


Figure 4-50: Comparison of heading angles for the Figure 8 trajectory

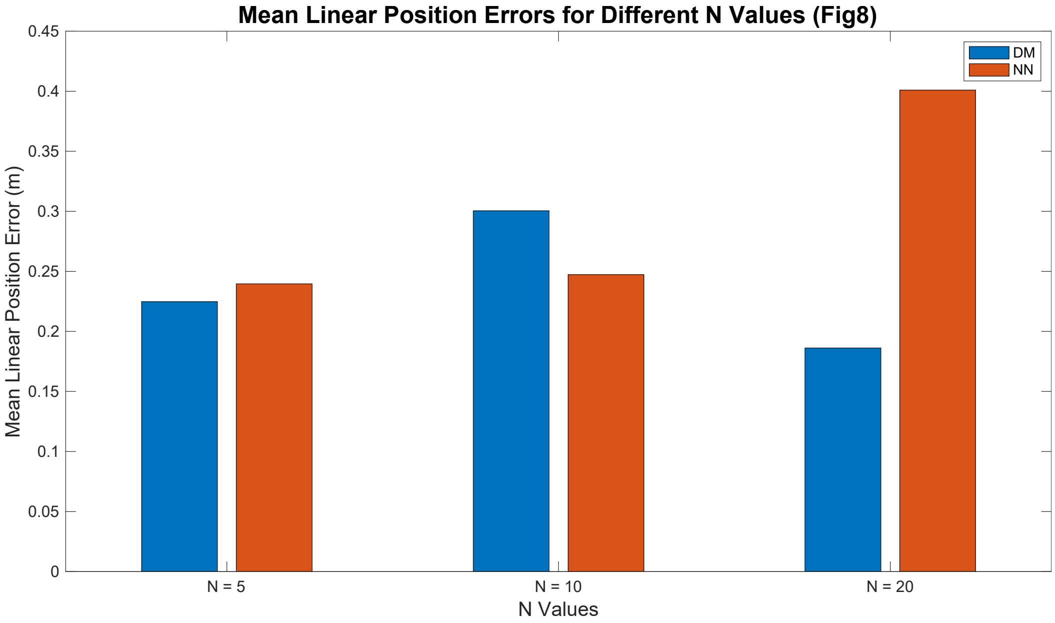


Figure 4-51: Position error at different prediction horizons for Figure 8

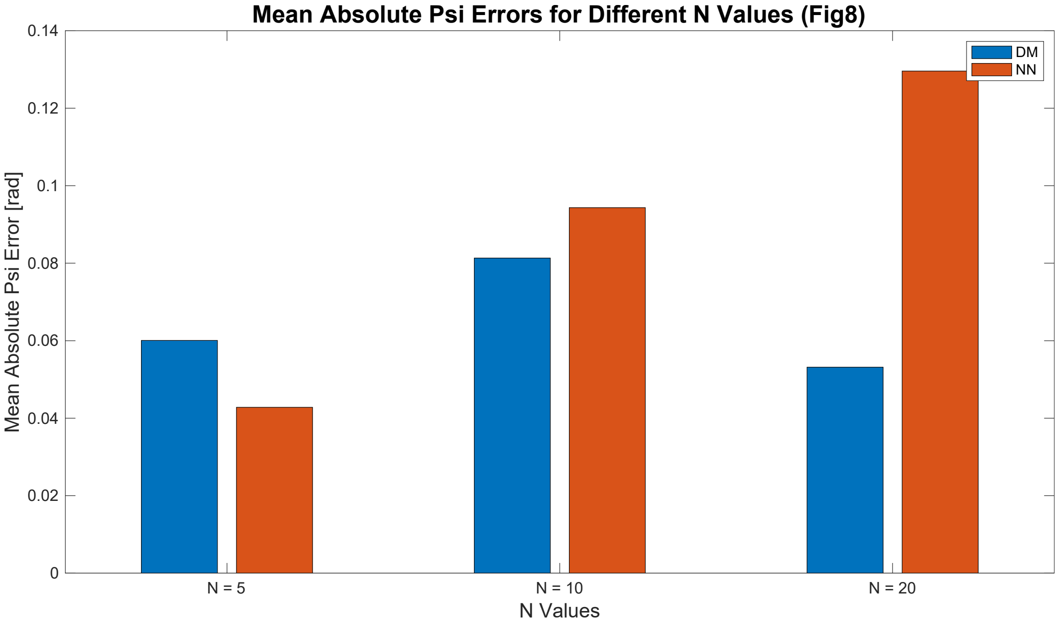


Figure 4-52: Heading angle error at different prediction horizons for Figure 8

From the figures 4-53 to figure 4-58 the velocities are also following the same trend. The lower prediction horizon is working well with all the models.

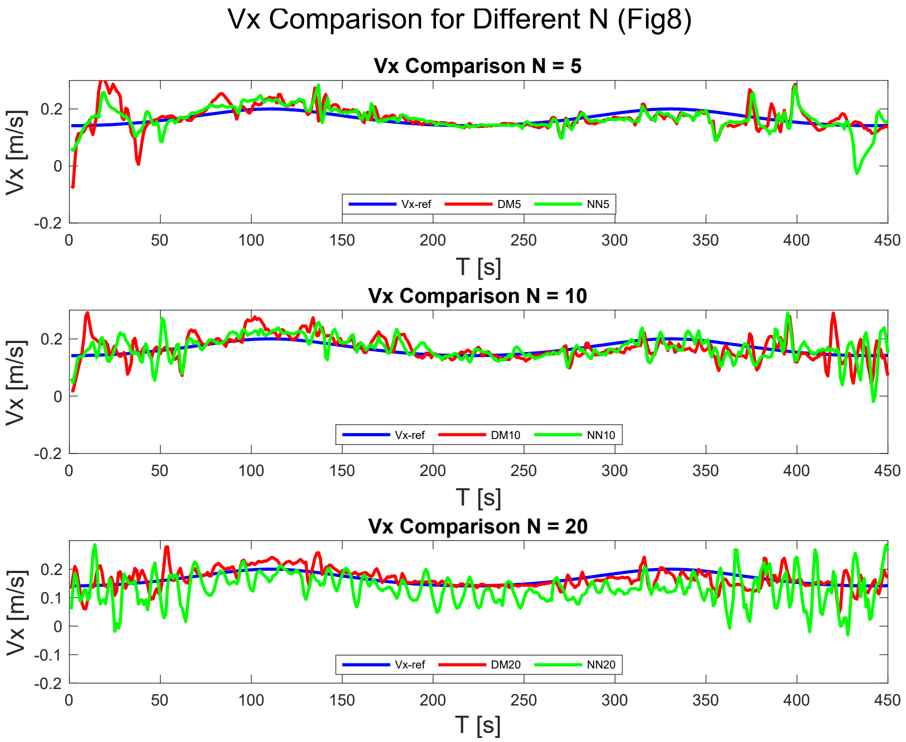


Figure 4-53: Comparison of X-velocity component $v_x$for the Figure 8 trajectory

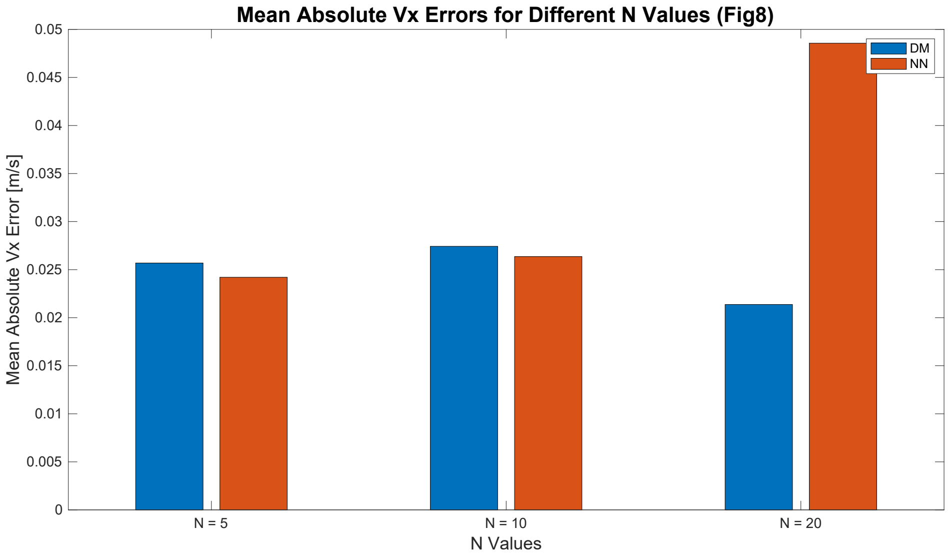


Figure 4-54: The error in v_x at different prediction horizons for Figure 8 trajectory

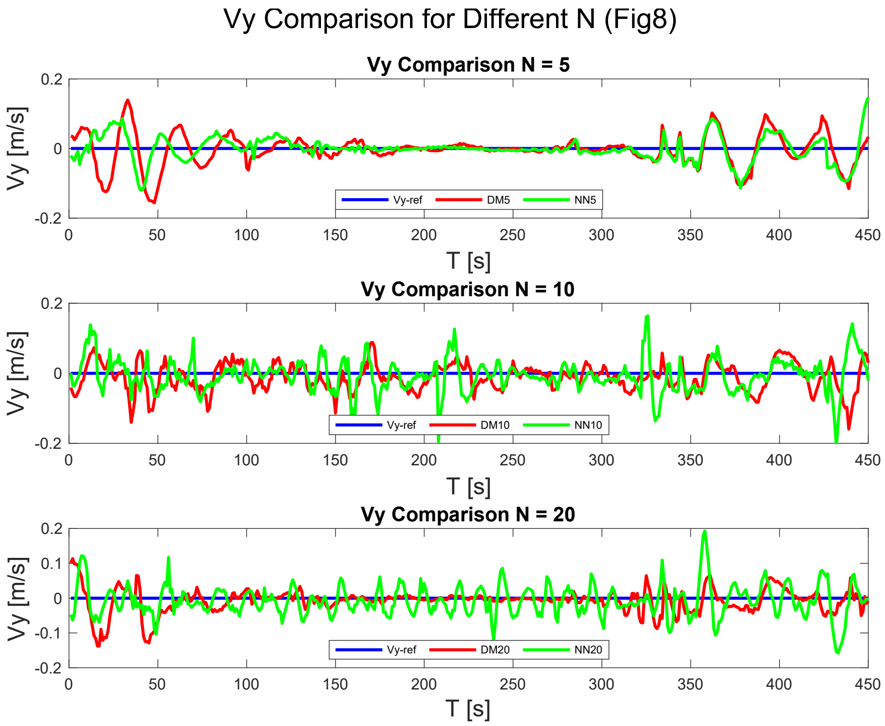


Figure 4-55: Comparison of Y-velocity components v_y for the Figure 8 trajectory

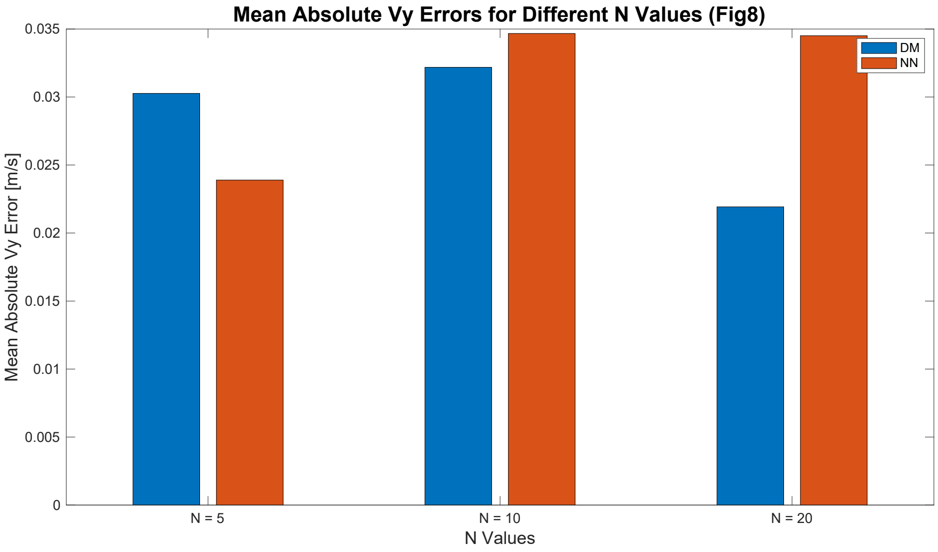


Figure 4-56: The error in v_y at different prediction horizons for Figure 8

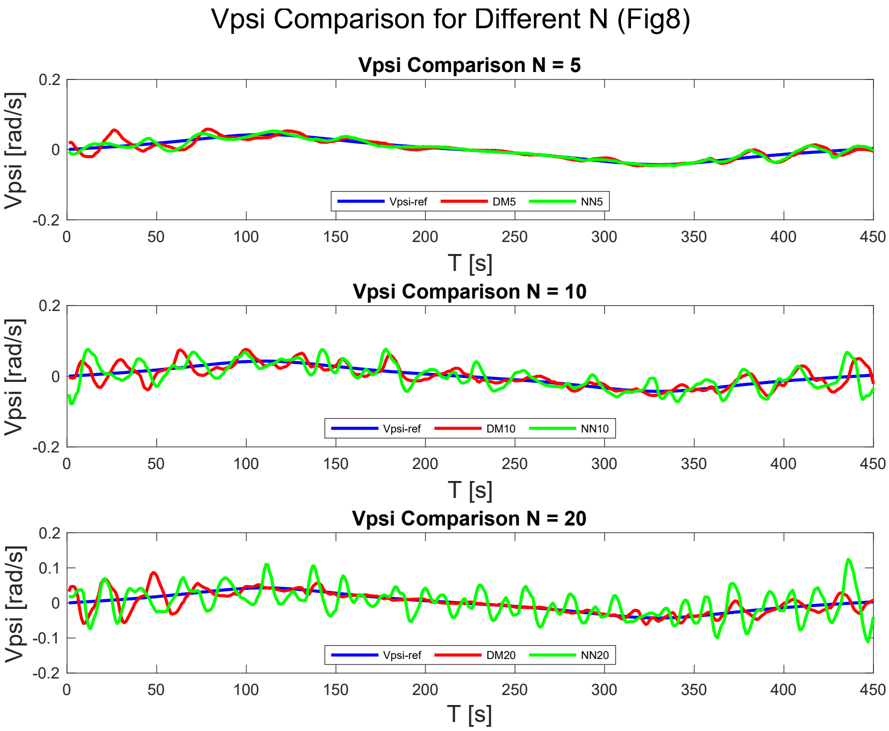


Figure 4-57: Comparison of yaw rate v_ψ for the Figure 8 trajectory

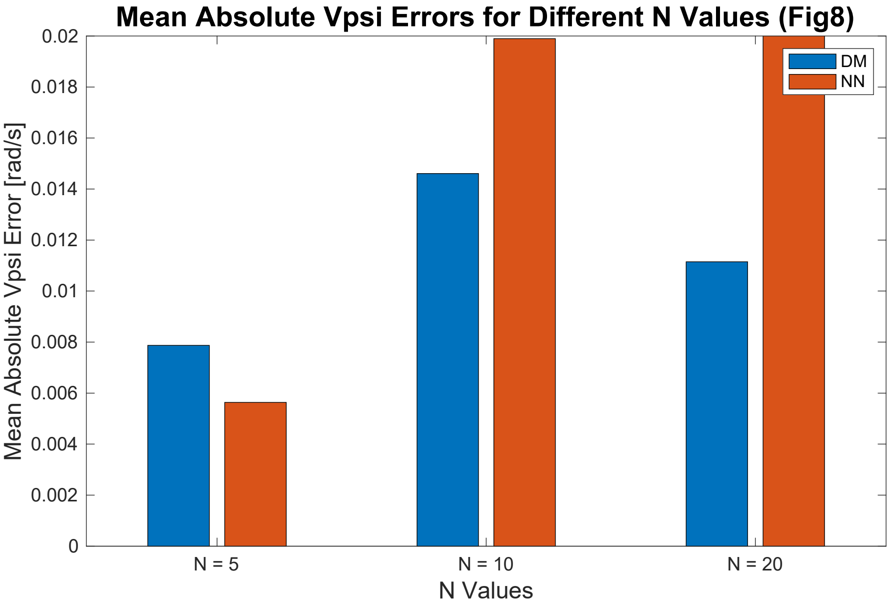
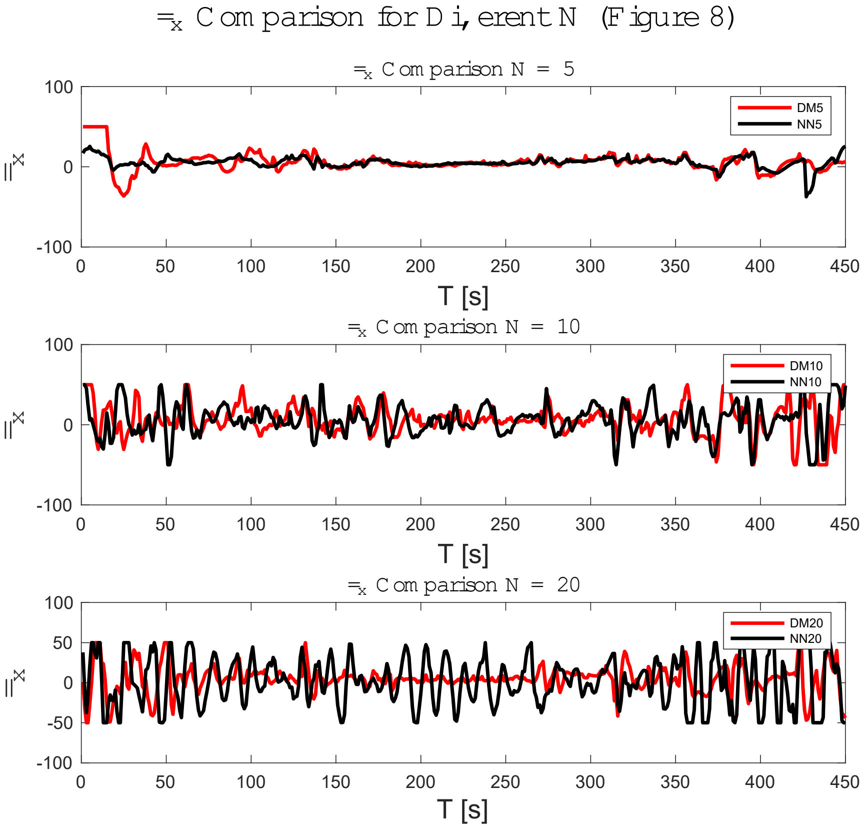


Figure 4-58: The error in v_ψ at different prediction horizons for Figure 8 trajectory

From the figure 4-59 we get a similar type of trend for the force distribution. For figure 8 trajectory the lower prediction horizon produced the smoothest control action. The table 4-10 shows the summary of the errors and the table 4-11 shows the force distribution for all the ship models used in the NMPC.



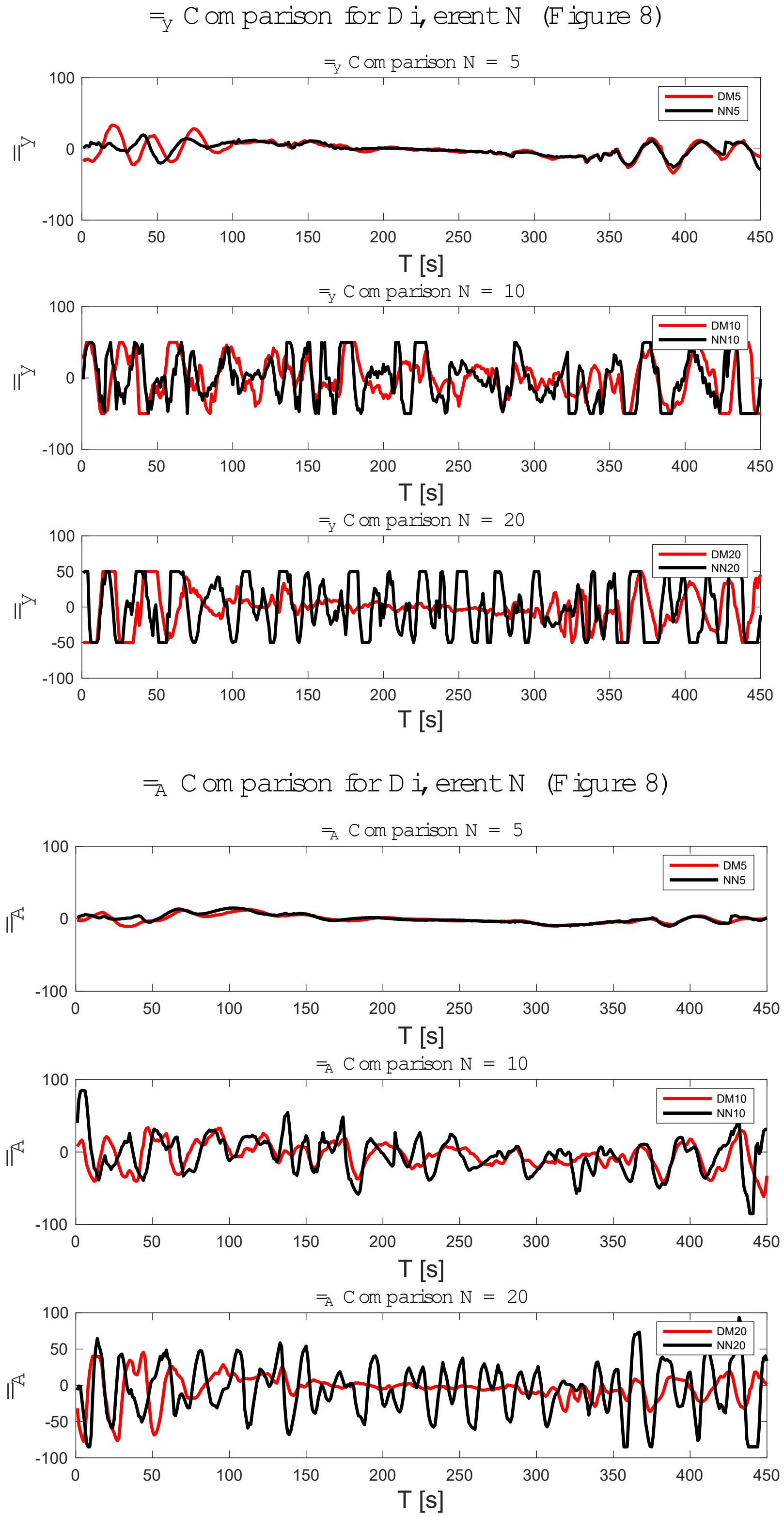


Figure 4-59: Forces applied for the Figure 8 Trajectory

Table 4-10: Error comparisons for Figure 8 trajectory for all the models

Parameters	**Mechanistic model**			**Neural Network model**		
Prediction Horizon, N	5	10	20	5	10	20
Sum of root mean squared position error, e_{xy} (m)	0.2248	0.3005	0.1862	0.2396	0.2473	0.4009
Heading Angle error, e_{ψ} (rad)	0.0601	0.0813	0.0532	0.0428	0.0943	0.1296
Surge velocity error, e_{v_x} (m/s)	0.0257	0.0274	0.0214	0.0242	0.0264	0.0486
Sway Velocity error, e_{v_y} (m/s)	0.0303	0.0322	0.0219	0.0239	0.0347	0.0345
Yaw rate error, e_{v_ψ}(rad/s)	0.0079	0.046	0.0111	0.0056	0.0199	0.0290
Average Execution Time per iteration,T_s(s)	0.0918	0.3899	0.8339	0.1973	0.9518	2.0024

Table 4-11: comparison of the forces for different prediction horizon for the Figure 8 trajectory

Models used with NMPC		**Surge force (τ_x) range (N)**			**Surge force (τ_y) range (N)**			**Yaw moment (τ_ψ) range (Nm)**		
		N =5	**N =10**	**N=20**	**N =5**	**N =10**	**N=20**	**N =5**	**N =10**	**N=20**
Mechanistic Model	min	-36.55	-50.00	-49.99	-34.36	-50.00	-50.00	-10.73	-61.61	-77.45
	max	50	50	49.99	33.49	50.00	50.00	13.43	33.47	45.36
Neural Network Model	min	-37.42	-50.00	-49.99	-29.10	-50.00	-50.00	-10.27	-85.00	-84.99
	max	25.62	50.00	50.00	19.58	50.00	50.00	14.86	84.95	93.49

Overall observation of the experiment

The experimental results clearly show that all the models are performing well for lower prediction horizon for this ship model. Looking at tables 4-9 to 4-11, we get a clear idea of how the models have worked. The neural network work exceptionally well for a low prediction horizon, given that the real-life experiments are far more critical than the simulation results. The factors that affect the experimental results are mainly- (i) Accuracy of the model; (ii) the time required for the application of the control action; (iii) the sensor noise and disturbances while experimenting with the testing base; and (iv) the computational limitations were observed on the higher prediction horizons. These differences can undermine the controller's performance. But overall performance for the trajectory tracking was satisfactory.

Improving the model using the Real Data set

In the preceding section, a mechanistic model was employed to generate a dataset for developing and observing the NMPC (Nonlinear Model Predictive Control) with a neural network structure. Despite the availability of previous experimental data for the Magne Viking, utilizing this dataset was unfeasible due to its insufficient data points and poor training outcomes. The time constraints of the experiment prevented thorough training of the dataset with various options. Consequently, the same methods outlined in section 4.3.1 were applied to the real dataset prior to the experiment. However, these training efforts failed to yield satisfactory results necessary for creating a well-functioning model compatible with NMPC. The figure 4-60 shows the validation results using the real dataset. The results were not satisfactory when NN model was trained with real data set.

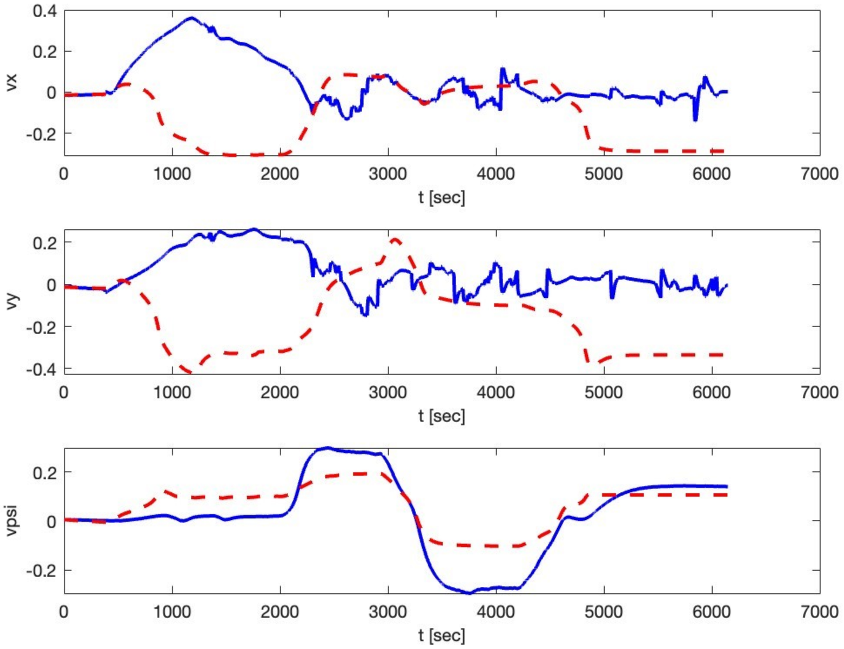


Figure: 4-60 Initial validation results by training the neural network with natural data

To enhance the neural network model, a deep learning technique was implemented. While the shallow neural network initially produced a model closely resembling the mechanistic model, it was eventually refined to a satisfactory level for utilization in experiments and simulations. Moving forward, the trained model utilizing real dataset can serve as a high-quality model for NMPC, offering promising outcomes for future research and application.

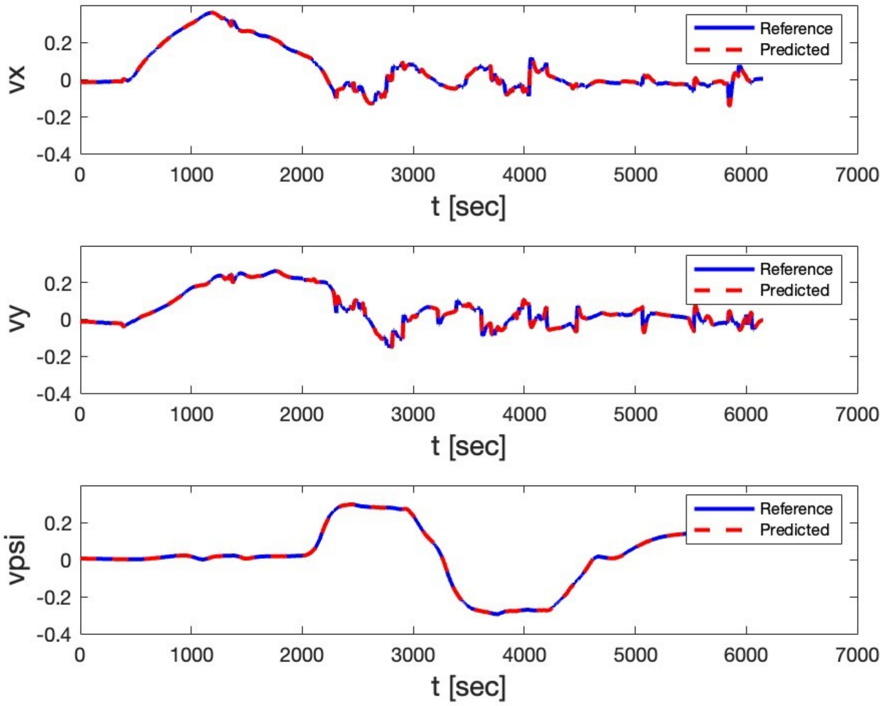


Figure: 4-61 Improved model by training with deep neural network

The neural network structure used here for the training with this real data set consists of 7 hidden layers. All the layers except the output layers used hyperbolic tangent activation functions with different number of nodes in them. The neural network structure is shown in the figure 4-62.

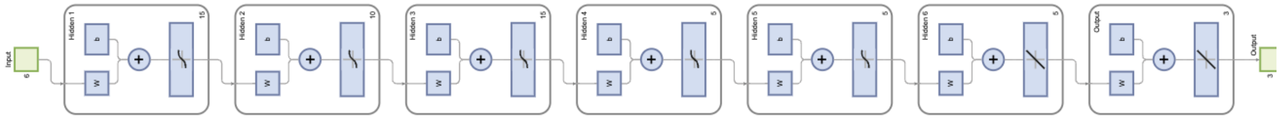


Figure 4-62: Deep Neural Network structure for training with the real data set

30000 data points from the real data set were selected to train the model. And 6000 data points were used for model validation which is presented in figure 4-61. A summary of the structure used for deep learning method is described here.

Table 4-12 : Summary of the Deep Neural Network Training Algorithm

Data
Predictors – 30000 observations with 6 features
Responses – 30000 observations with 3 features
Training set - 70%
Testing and validation – 30%
Algorithm
Data division – Random
Training Algorithm – Bayessian Regularization
Performance – Mean Squared Error

4.4 Conclusion

In this study, the simulation results as well as the experimental results were presented in details. From the simulation results we can conclude that neural network structure worked very well with the NMPC. Even in some cases the neural network model worked better than the mechanistic model. For different trajectory different models worked well having the same cost function weights in NMPC in the presence of same wave disturbance models. The feedback of the experimental ship model is dependent on the Qualisys motion capture system. It involved some dropouts which involved feedback errors. Even though the experimental setup does not provide ideal conditions like simulation the results obtained from the experiment was showing effective tracking with a shorter prediction horizon. The longer prediction horizon showed poor performance with all ship models.

When trained and tested with real datasets, the enhanced neural network model is promising for advancing NMPC applications. Unfortunately, we could not thoroughly test the neural network model trained with real data due to time constraints and insufficient convincing results. However, it's crucial to acknowledge that when subjected to testing with the Magne Viking ship in future studies, this model could yield compelling and convincing results.

The potential of this neural network model lies in its ability to learn from real-world data, offering a more accurate representation of the system dynamics compared to purely mechanistic models or those trained on synthetic data. By leveraging the rich information captured in real datasets, the neural network can better capture the intricacies and complexities of the system under consideration.

Although we couldn't explore its full potential in this instance, the groundwork laid by training the neural network with real data sets the stage for future investigations. Once time allows for thorough testing and validation, this model will demonstrate its efficacy in NMPC applications with the Magne Viking ship. As we continue to refine and validate this neural network model, it promises to significantly enhance our ability to control and optimize complex systems like the Magne Viking ship.

Chapter 5

5.0 Conclusions and Recommendations

5.1 Conclusions

In this study, an NMPC-based controller was designed and tested for trajectory tracking of autonomous vessels. This study's objective was designing a robust controller that can overcome harsh environmental situations and follow the path within the required time. For designing and testing a NMPC trajectory tracking controller the Magne Ship model was used. Starting from the conventional mechanistic model, the controller used a neural network model. After getting satisfactory simulation results, the controller was tested at NRC, Canada's testing facility, for further validation of the controller's performance. In all cases, to check the versatility of the controller, two different types of trajectories were used for the ship to track. The circular trajectory was used as it was one of the most conventional forms of trajectory tracking path. On the other hand, the controller followed a complex figure eight-shaped trajectory to check its adaptability and dynamic abilities for different paths. From the simulation and the experiments, we can come to these conclusions.

- The simulation and experimental setup may follow different trend for the different trajectories. For example, the circular trajectory tracking in simulations and experiments concluded that the hybrid model showed better results. In the case of Figure eight shaped trajectory, the neural network, and the mechanistic model both showed promising results. It was also seen that a moderate trajectory tracking prediction horizon worked better for all the cases. The simulation results were smooth even in the presence of the disturbances model, which indicates that the Unscented Kalman Filter did an excellent job of filtering the first order waves frequencies and estimating the states.

- A higher prediction horizon may not work when the purpose is merely trajectory tracking. A smooth trajectory tracking performance requires fast control actions. But when the prediction horizon is a large, the computational time increases. As a result, a delayed control action may provide a delayed response to the ship. This type of control action is not desired in real-life scenarios. A big prediction horizon accumulates errors with time, which compromises some accuracy. A higher prediction horizon is more suitable for obstacle avoidance tasks.
- The neural network model was trained to be generalized and follow any given trajectory to the controller. It showed the model's versatility and adaptability even with the simulated wave disturbances. The controller design provided a foundation for testing the autonomous vessel in real oceans.
- Integrating this neural network structure within the NMPC framework offers several notable advantages. Firstly, by encapsulating complex system dynamics, the neural network enhances the controller's ability to predict future states and optimize control actions accurately. This predictive capability is essential for navigating dynamic and uncertain environments, such as those encountered by autonomous vessels in real oceans. Furthermore, the neural network's adaptability enables it to learn and adjust to varying operating conditions, ensuring optimal performance across various scenarios. This adaptive nature is precious for NMPC, where system dynamics may evolve or exhibit nonlinear behavior that conventional control methods struggle to accommodate.

5.2 Recommendations

- While doing experiments at the NRC, the wave generators were not functional. Any new experiments for trajectory tracking with the same controller can be tested in the presence of a wave generator.
- Different machine-learning techniques can be used to build models for NMPC. The LSTM model can be a good choice if it can provide quick control actions. In this case, reinforcement learning can also be utilized for trajectory tracking. The neural network model trained with real dataset can be used in the future for getting tracking results with lower or higher prediction horizons.
- At the time of the experiment, the feedback system used for the trajectory tracking is vital. We experienced intermittent feedback errors that caused some issues. However, we also had extended periods of reliable feedback, which allowed us to effectively test the controller.
- To optimize trajectory tracking performance, the impact of an adaptive dynamic control horizon demands further exploration.
- An auto-learning dynamic neural network model that can learn and update the predictions from the collected data can be another great addition to the machine learning-based trajectory tracking system.

References

[1] Rolls-Royce plc. (2017). [Rolls-Royce]. Retrieved from https://www.rolls-royce.com/products-and-services/power-systems/marine.aspx

[2] A. Haseltalab and R. R. Negenborn, “Adaptive control for autonomous ships with uncertain model and unknown propeller dynamics,” *Control Eng Pract*, vol. 91, p. 104116, Oct. 2019, doi: 10.1016/J.CONENGPRAC.2019.104116.

[3] S. L. Dai, M. Wang, and C. Wang, “Neural Learning Control of Marine Surface Vessels with Guaranteed Transient Tracking Performance,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1717–1727, Mar. 2016, doi: 10.1109/TIE.2015.2504553.

[4] M. Abdelaal, M. Fränzle, and A. Hahn, “NMPC-based Trajectory Tracking and Collison Avoidance of Underactuated Vessels with Elliptical Ship Domain,” *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 22–27, Jan. 2016, doi: 10.1016/J.IFACOL.2016.10.316.

[5] H. Zheng, R. R. Negenborn, and G. Lodewijks, “Trajectory tracking of autonomous vessels using model predictive control,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, IFAC Secretariat, 2014, pp. 8812–8818. doi: 10.3182/20140824-6-za-1003.00767.

[6] J.-X. Zhang and G.-H. Yang, “Fault-Tolerant Fixed-Time Trajectory Tracking Control of Autonomous Surface Vessels With Specified Accuracy,” *IEEE Transactions on Industrial Electronics*, vol. 67, no. 6, pp. 4889–4899, Jun. 2020, doi: 10.1109/TIE.2019.2931242.

[7] Y. Shi, C. Shen, H. Fang, and H. Li, “Advanced Control in Marine Mechatronic Systems: A Survey,” *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 3, pp. 1121–1131, Jun. 2017, doi: 10.1109/TMECH.2017.2660528.

[8] S. A. Tutturen and R. Skjetne, “Hybrid Control to Improve Transient Response of Integral Action in Dynamic Positioning of Marine Vessels∗∗This work was supported by the Research Council of Norwa through the Centres of Excellence funding scheme, project number 223254 - AMOS.,” *IFAC-PapersOnLine*, vol. 48, no. 16, pp. 166–171, 2015, doi: 10.1016/j.ifacol.2015.10.275.

[9] B. Bidikli, E. Tatlicioglu, and E. Zergeroglu, “Robust dynamic positioning of surface vessels via multiple unidirectional tugboats,” *Ocean Engineering*, vol. 113, pp. 237–245, Feb. 2016, doi: 10.1016/j.oceaneng.2015.12.057.

[10] J. Ghommam and F. Mnif, “Coordinated Path-Following Control for a Group of Underactuated Surface Vessels,” *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 3951–3963, Oct. 2009, doi: 10.1109/TIE.2009.2028362.

[11] Y. Yang, J. Du, H. Liu, C. Guo, and A. Abraham, “A Trajectory Tracking Robust Controller of Surface Vessels With Disturbance Uncertainties,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1511–1518, Jul. 2014, doi: 10.1109/TCST.2013.2281936.

[12] K. D. Do and J. Pan, “Global robust adaptive path following of underactuated ships,” *Automatica*, vol. 42, no. 10, pp. 1713–1722, Oct. 2006, doi: 10.1016/j.automatica.2006.04.026.

[13] Y. Yang, J. Du, H. Liu, C. Guo, and A. Abraham, “A Trajectory Tracking Robust Controller of Surface Vessels With Disturbance Uncertainties,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1511–1518, Jul. 2014, doi: 10.1109/TCST.2013.2281936.

[14] C. Hu, R. Wang, F. Yan, and N. Chen, “Robust Composite Nonlinear Feedback Path-Following Control for Underactuated Surface Vessels With Desired-Heading Amendment,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 10, pp. 6386–6394, Oct. 2016, doi: 10.1109/TIE.2016.2573240.

[15] Z.-P. Jiang, “Global tracking control of underactuated ships by Lyapunov’s direct method,” *Automatica*, vol. 38, no. 2, pp. 301–309, Feb. 2002, doi: 10.1016/S0005-1098(01)00199-6.

[16] X. Jin, “Fault tolerant finite-time leader–follower formation control for autonomous surface vessels with LOS range and angle constraints,” *Automatica*, vol. 68, pp. 228–236, Jun. 2016, doi: 10.1016/j.automatica.2016.01.064.

[17] X. Jin, “Adaptive finite-time fault-tolerant tracking control for a class of MIMO nonlinear systems with output constraints,” *International Journal of Robust and Nonlinear Control*, vol. 27, no. 5, pp. 722–741, Mar. 2017, doi: 10.1002/rnc.3596.

[18] A. Rupp and M. Stolz, “Survey on Control Schemes for Automated Driving on Highways,” in *Automated Driving*, Cham: Springer International Publishing, 2017, pp. 43–69. doi: 10.1007/978-3-319-31895-0_4.

[19] A. Kanarat, R. H. Sturges, M. Ahmadian, D. J. Leo, C. F. Reinholtz, and C. Woolsey, “Motion Planning and Robust Control for Nonholonomic Mobile Robots under Uncertainties,” 2004.

[20] H. Talebi Abatari and A. Dehghani Tafti, “Using a fuzzy PID controller for the path following of a car-like mobile robot,” in *2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, IEEE, Feb. 2013, pp. 189–193. doi: 10.1109/ICRoM.2013.6510103.

[21] Th. Fraichard and Ph. Garnier, “Fuzzy control to drive car-like vehicles,” *Rob Auton Syst*, vol. 34, no. 1, pp. 1–22, Jan. 2001, doi: 10.1016/S0921-8890(00)00096-8.

[22] F. Riaz and M. A. Niazi, “Enhanced emotion enabled cognitive agent-based rear-end collision avoidance controller for autonomous vehicles,” *Simulation*, vol. 94, no. 11, pp. 957–977, Nov. 2018, doi: 10.1177/0037549717742203.

[23] U. Nunes and L. C. Bento, “Data fusion and path-following controllers comparison for autonomous vehicles,” *Nonlinear Dyn*, vol. 49, no. 4, pp. 445–462, Sep. 2007, doi: 10.1007/s11071-006-9108-y.

[24] Y. Wu, L. Wang, J. Zhang, and F. Li, “Path Following Control of Autonomous Ground Vehicle Based on Nonsingular Terminal Sliding Mode and Active Disturbance Rejection Control,” *IEEE Trans Veh Technol*, vol. 68, no. 7, pp. 6379–6390, Jul. 2019, doi: 10.1109/TVT.2019.2916982.

[25] H. Guo, D. Cao, H. Chen, Z. Sun, and Y. Hu, “Model predictive path following control for autonomous cars considering a measurable disturbance: Implementation, testing, and verification,” *Mech Syst Signal Process*, vol. 118, pp. 41–60, Mar. 2019, doi: 10.1016/j.ymssp.2018.08.028.

[26] A. Norouzi, R. Kazemi, and S. Azadi, “Vehicle lateral control in the presence of uncertainty for lane change maneuver using adaptive sliding mode control with fuzzy boundary layer,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 232, no. 1, pp. 12–28, Jan. 2018, doi: 10.1177/0959651817733222.

[27] T. C. Martin, M. E. Orchard, and P. V. Sánchez, “Design and Simulation of Control Strategies for Trajectory Tracking in an Autonomous Ground Vehicle,” *IFAC Proceedings Volumes*, vol. 46, no. 24, pp. 118–123, Sep. 2013, doi: 10.3182/20130911-3-BR-3021.00096.

[28] H. Cao, X. Song, S. Zhao, S. Bao, and Z. Huang, “An optimal model-based trajectory following architecture synthesising the lateral adaptive preview strategy and longitudinal velocity planning for highly automated vehicle,” *Vehicle System Dynamics*, vol. 55, no. 8, pp. 1143–1188, Aug. 2017, doi: 10.1080/00423114.2017.1305114.

[29] S. E. Li, H. Chen, R. Li, Z. Liu, Z. Wang, and Z. Xin, “Predictive lateral control to stabilise highly automated vehicles at tire-road friction limits,” *Vehicle System Dynamics*, vol. 58, no. 5, pp. 768–786, May 2020, doi: 10.1080/00423114.2020.1717553.

[30] C. E. Beal and J. C. Gerdes, “Model Predictive Control for Vehicle Stabilization at the Limits of Handling,” *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1258–1269, Jul. 2013, doi: 10.1109/TCST.2012.2200826.

[31] E. Kayacan, H. Ramon, and W. Saeys, “Robust Trajectory Tracking Error Model-Based Predictive Control for Unmanned Ground Vehicles,” *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2, pp. 806–814, Apr. 2016, doi: 10.1109/TMECH.2015.2492984.

[32] A. Franco and V. Santos, “Short-term Path Planning with Multiple Moving Obstacle Avoidance based on Adaptive MPC,” in *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, IEEE, Apr. 2019, pp. 1–7. doi: 10.1109/ICARSC.2019.8733653.

[33] W. Sha and K. L. Edwards, “The use of artificial neural networks in materials science based research,” *Mater Des*, vol. 28, no. 6, pp. 1747–1752, Jan. 2007, doi: 10.1016/j.matdes.2007.02.009.

[34] D. Zafeiris, S. Rutella, and G. R. Ball, “An Artificial Neural Network Integrated Pipeline for Biomarker Discovery Using Alzheimer’s Disease as a Case Study.,” *Comput Struct Biotechnol J*, vol. 16, pp. 77–87, 2018, doi: 10.1016/j.csbj.2018.02.001.

[35] E. Chatzimichail *et al.*, “γ -H2AX: A Novel Prognostic Marker in a Prognosis Prediction Model of Patients with Early Operable Non-Small Cell Lung Cancer,” *Int J Genomics*, vol. 2014, pp. 1–6, 2014, doi: 10.1155/2014/160236.

[36] J. Kabzan, L. Hewing, A. Liniger, and M. N. Zeilinger, “Learning-Based Model Predictive Control for Autonomous Racing,” *IEEE Robot Autom Lett*, vol. 4, no. 4, pp. 3363–3370, Oct. 2019, doi: 10.1109/LRA.2019.2926677.

[37] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, “A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles,” *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, Mar. 2016, doi: 10.1109/TIV.2016.2578706.

[38] M. Graf Plessen, “Automating Vehicles by Deep Reinforcement Learning Using Task Separation with Hill Climbing,” 2020, pp. 188–210. doi: 10.1007/978-3-030-12385-7_16.

[39] S. Wang, Z. Sun, Q. Yuan, Z. Sun, Z. Wu, and T.-H. Hsieh, “Autonomous piloting and berthing based on Long Short Time Memory neural networks and nonlinear model predictive control algorithm,” *Ocean Engineering*, vol. 264, p. 112269, Nov. 2022, doi: 10.1016/j.oceaneng.2022.112269.

[40] N. Mizuno, M. Kuroda, T. Okazaki, and K. Ohtsu, “Minimum time ship maneuvering method using neural network and nonlinear model predictive compensator,” *Control Eng Pract*, vol. 15, no. 6, pp. 757–765, Jun. 2007, doi: 10.1016/j.conengprac.2007.01.002.

[41] H. Qu, L. Song, Y. Zhang, Z. Gao, and D. Shi, “The Effect of Prebiotic Products on Decreasing Adiposity Parameters in Overweight and Obese Individuals: A Systematic Review and Meta-Analysis,” *Curr Med Chem*, vol. 28, no. 2, pp. 419–431, Dec. 2020, doi: 10.2174/0929867327666191230110128.

[42] J.-Y. Park and N. Kim, “Design of an adaptive backstepping controller for auto-berthing a cruise ship under wind loads,” *International Journal of Naval Architecture and Ocean Engineering*, vol. 6, no. 2, pp. 347–360, Jun. 2014, doi: 10.2478/IJNAOE-2013-0184.

[43] Y. A. Ahmed and K. Hasegawa, “Automatic ship berthing using artificial neural network trained by consistent teaching data using nonlinear programming method,” *Eng Appl Artif Intell*, vol. 26, no. 10, pp. 2287–2304, Nov. 2013, doi: 10.1016/j.engappai.2013.08.009.

[44] S. J. Julier and J. K. Uhlmann, “Unscented Filtering and Nonlinear Estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004, doi: 10.1109/JPROC.2003.823141.

[45] E. A. Wan and R. Van Der Merwe, “The unscented Kalman filter for nonlinear estimation,” in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and*

Control Symposium (Cat. No.00EX373), IEEE, pp. 153–158. doi: 10.1109/ASSPCC.2000.882463.

[46] Julier. S and J. Uhlmann, “A New Extension of the Kalman Filter to Nonlinear Systems”.

[47] E. Bradford and L. Imsland, “Stochastic Nonlinear Model Predictive Control with State Estimation by Incorporation of the Unscented Kalman Filter,” Sep. 2017, doi: 10.1016/j.ifacol.2018.09.336.

[48] H. Efheij, A. Albagul, and N. Ammar Albraiki, “Comparison of Model Predictive Control and PID Controller in Real Time Process Control System,” in *2019 19th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, IEEE, Mar. 2019, pp. 64–69. doi: 10.1109/STA.2019.8717271.

[49] D. Yu, F. Deng, H. Wang, X. Hou, H. Yang, and T. Shan, “Real-Time Weight Optimization of a Nonlinear Model Predictive Controller Using a Genetic Algorithm for Ship Trajectory Tracking,” *J Mar Sci Eng*, vol. 10, no. 8, p. 1110, Aug. 2022, doi: 10.3390/jmse10081110.

[50] S. J. Qin and T. A. Badgwell, “A survey of industrial model predictive control technology,” *Control Eng Pract*, vol. 11, no. 7, pp. 733–764, Jul. 2003, doi: 10.1016/S0967-0661(02)00186-7.

[51] Z. Li, J. Sun, and S. Oh, “Path following for marine surface vessels with rudder and roll constraints: An MPC approach,” in *2009 American Control Conference*, IEEE, 2009, pp. 3611–3616. doi: 10.1109/ACC.2009.5160302.

[52] Z. Li and J. Sun, “Disturbance Compensating Model Predictive Control With Application to Ship Heading Control,” *IEEE Transactions on Control Systems Technology*, 2011, doi: 10.1109/TCST.2011.2106212.

[53] A. Pavlov, H. Nordahl, and M. Breivik, “MPC-based optimal path following for underactuated vessels,” *IFAC Proceedings Volumes*, vol. 42, no. 18, pp. 340–345, 2009, doi: 10.3182/20090916-3-BR-3001.0065.

[54] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011. doi: 10.1002/9781119994138.

[55] H. Zheng, R. R. Negenborn, and G. Lodewijks, “Trajectory tracking of autonomous vessels using model predictive control,” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 8812–8818, 2014, doi: 10.3182/20140824-6-ZA-1003.00767.

[56] M. Morari and U. Maeder, “Nonlinear offset-free model predictive control,” *Automatica*, vol. 48, no. 9, pp. 2059–2067, Sep. 2012, doi: 10.1016/j.automatica.2012.06.038.

[57] A. Nandan and S. Imtiaz, “Nonlinear model predictive control of managed pressure drilling,” *ISA Trans*, vol. 69, pp. 307–314, Jul. 2017, doi: 10.1016/j.isatra.2017.03.013.

[58] T. I. Fossen and T. Perez, “Kalman filtering for positioning and heading control of ships and offshore rigs,” *IEEE Control Syst*, vol. 29, no. 6, pp. 32–46, Dec. 2009, doi: 10.1109/MCS.2009.934408.

[59] R. Skjetne, Ø. N. Smogeli, and T. I. Fossen, “A Nonlinear Ship Manoeuvering Model: Identification and adaptive control with experiments for a model ship,” *Modeling, Identification and Control: A Norwegian Research Bulletin*, vol. 25, no. 1, pp. 3–27, 2004, doi: 10.4173/mic.2004.1.1.

[60] Thor I. Fossen, *Marine control systems: guidance, navigation and control of ships, rigs and underwater vehicles*. Trondheim, Norway: Springer, 2002.

[61] T. I. Fossen and J. P. Strand, “Passive nonlinear observer design for ships using lyapunov methods: full-scale experiments with a supply vessel,” *Automatica*, vol. 35, no. 1, pp. 3–16, Jan. 1999, doi: 10.1016/S0005-1098(98)00121-6.

[62] H. Yu, J. Duan, S. Taheri, H. Cheng, and Z. Qi, “A model predictive control approach combined unscented Kalman filter vehicle state estimation in intelligent vehicle trajectory tracking,” *Advances in Mechanical Engineering*, vol. 7, no. 5, p. 168781401557836, May 2015, doi: 10.1177/1687814015578361.

[63] G. G. Rigatos, “Sensor fusion-based dynamic positioning of ships using Extended Kalman and Particle Filtering,” *Robotica*, vol. 31, no. 3, pp. 389–403, May 2013, doi: 10.1017/S0263574712000409.

[64] D. Zhan, H. Zheng, and W. Xu, “Tracking Control of Autonomous Underwater Vehicles with Acoustic Localization and Extended Kalman Filter,” *Applied Sciences*, vol. 11, no. 17, p. 8038, Aug. 2021, doi: 10.3390/app11178038.

[65] Lokukaluge Prasad Perera and Carlos Guedes Soares, “Ocean Vessel Trajectory Estimation and Prediction Based on Extended Kalman Filter,” 2010.

[66] P. Batista, C. Silvestre, and P. Oliveira, “Optimal position and velocity navigation filters for autonomous vehicles,” *Automatica*, vol. 46, no. 4, pp. 767–774, Apr. 2010, doi: 10.1016/j.automatica.2010.02.004.

[67] F. Deng, H.-L. Yang, and L.-J. Wang, “Adaptive Unscented Kalman Filter Based Estimation and Filtering for Dynamic Positioning with Model Uncertainties,” *Int J Control Autom Syst*, vol. 17, no. 3, pp. 667–678, Mar. 2019, doi: 10.1007/s12555-018-9503-4.

[68] Y. Peng, J. Han, and Q. Song, “Tracking Control of Underactuated Surface Ships: Using Unscented Kalman Filter to Estimate the Uncertain Parameters,” in *2007 International Conference on Mechatronics and Automation*, IEEE, Aug. 2007, pp. 1884–1889. doi: 10.1109/ICMA.2007.4303838.

[69] Y. Wang, S. Chai, and H. D. Nguyen, “Unscented Kalman Filter trained neural network control design for ship autopilot with experimental and numerical approaches,” *Applied Ocean Research*, vol. 85, pp. 162–172, Apr. 2019, doi: 10.1016/j.apor.2019.01.030.

[70] B. Allotta et al., “An unscented Kalman filter based navigation algorithm for autonomous underwater vehicles,” *Mechatronics*, vol. 39, pp. 185–195, Nov. 2016, doi: 10.1016/j.mechatronics.2016.05.007

[71] B. J. Guerreiro, C. Silvestre, R. Cunha, and A. Pascoal, “Trajectory tracking nonlinear model predictive control for autonomous surface craft,” in *2009 European Control Conference (ECC)*, IEEE, Aug. 2009, pp. 1311–1316. doi: 10.23919/ECC.2009.7074587.

[72] M. Abdelaal and A. Hahn, “NMPC-based trajectory tracking and collision avoidance of unmanned surface vessels with rule-based colregs confinement,” in *2016 IEEE Conference on Systems, Process and Control (ICSPC)*, IEEE, Dec. 2016, pp. 23–28. doi: 10.1109/SPC.2016.7920697.

[73] L. Tang, L. Wang, Y. Wang, and Y. Zhang, “An enhanced trajectory tracking control of the dynamic positioning ship based on nonlinear model predictive control and disturbance observer,” *Ocean Engineering*, vol. 265, p. 112482, Dec. 2022, doi: 10.1016/j.oceaneng.2022.112482.

[74] Alagili, O., Fernando, E., Ahmed, S., Imtiaz, S., Murrant, K., Gash, B., Islam, M., & Zaman, H. (2024). “Experimental investigations of an energy-efficient dynamic positioning controller for different sea conditions”. *Ocean Engineering*, *299*(117297), 117297. https://doi.org/10.1016/j.oceaneng.2024.117297