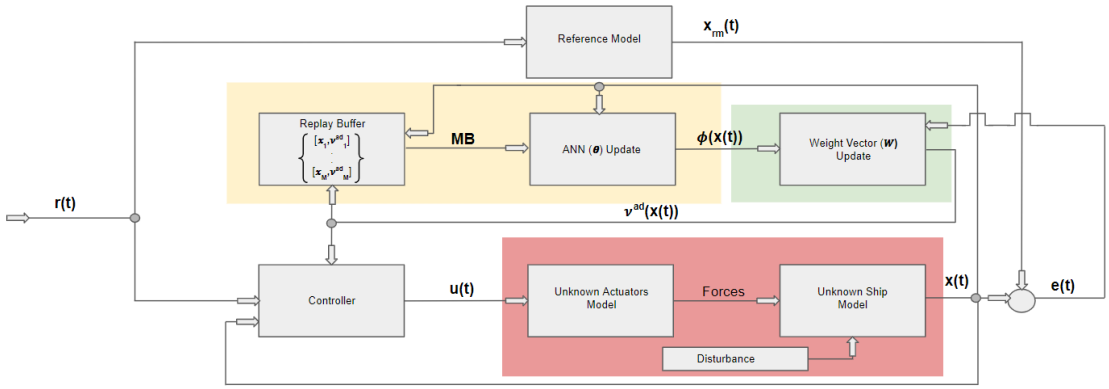
CHALMERS



Adaptive ship stabilization

Control of ship stabilizing actuators using modified deep model reference adaptive control

Master's thesis in Systems, Control & Mechatronics

BERTILSSON, TOBIAS
PERSSON, JOHANNES

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2022

MASTER'S THESIS IN SYSTEMS, CONTROL & MECHATRONICS

Adaptive ship stabilization

Control of ship stabilizing actuators using modified deep model reference adaptive control

Bertilsson, Tobias
Persson, Johannes

Department of Mechanics and Maritime Sciences
Division of Applied Artificial Intelligence
CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2022

Adaptive ship stabilization
Control of ship stabilizing actuators using modified deep model reference adaptive control

Bertilsson, Tobias
Persson, Johannes

Supervisor: Simon Bastås, CPAC Systems AB
Examiner: Peter Forsberg, Mechanics and Maritime Sciences

Master's thesis 2022:28
Department of Mechanics and Maritime Sciences
Division of Applied Artificial Intelligence
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: +46 (0)31-772 1000

Cover:
Block scheme of the proposed controller

Chalmers Reproservice
Göteborg, Sweden 2022

Adaptive ship stabilization
Control of ship stabilizing actuators using modified deep model reference adaptive control
Master's thesis in Systems, Control & Mechatronics

Bertilsson, Tobias
Persson, Johannes

Department of Mechanics and Maritime Sciences
Division of Applied Artificial Intelligence
Chalmers University of Technology

Abstract

For common control methods, there are often an extensive modeling and tuning procedure needed to obtain a good control performance. This thesis investigates a way of omitting that procedure by utilizing a machine learning based control method. An adaptive method is presented of how to control the stabilizing actuators of a ship with limited or no information about the ship's dynamics, and where the core of the adaptiveness lies in the combination of a feature vector, extracted from an artificial neural network (ANN), and a weight vector. The main objective was to reduce the most critical ship movement, rolling motions, using active fin stabilizers. The proposed adaptive controller showed promising results that increased the performance of the ordinary linear controller, while also being able to adapt when changes in the plant were introduced. The second objective was to see if the proposed controller was capable of controlling multiple kinds of actuators. The results indicate that this is the case, however, some additional modifications need to be made to achieve the same performance as in the first case when using only fins.

Keywords: Adaptive control, Ship stabilization, Model reference adaptive control (MRAC), Deep Model Reference Adaptive Control (DMRAC), Machine learning, Online training

Acknowledgements

We would like to thank our examiner Peter Forsberg and supervisor Simon Bastås for providing counsel during the time of this work. We would also like to thank CPAC systems and their employees for welcoming us to their office and providing us with necessary resources.

Tobias Bertilsson & Johannes Persson, Gothenburg June 2022

Thesis advisor: Simon Bastås, CPAC Systems AB
Thesis examiner: Peter Forsberg, Department of Mechanics and Maritime Sciences

Contents

List of Figures

List of Tables

1

Introduction

A ship subjected to waves, winds, and drifts of the ocean move according to the force applied. This is usually not desirable since it results in a less than optimal power consumption, seasickness among the crew or in a worst case scenario, real damage to the cargo or ship. At sea there are six basic movements to consider, based on the degrees of freedom, (DOF): Heaving: vertical movement, Swaying: transverse movement, Surging: longitudinal movement, Rolling: longitudinal rotation, Pitching: transverse rotation, and Yawing: vertical rotation. These basic movements may also combine in a periodic way, which is common when the waves hit in non right angles causing, for example, synchronized rolling and yawing which typically occurs most of the time. In order to mitigate the movements boats have stabilizing systems. Historically, boats have been equipped with static systems like static fins, bilge keels or similar. In modern times dynamic solutions with controlled stabilizers like interceptors, fins, water tanks etc. are used. However, controlling the stabilizers in an optimal way, depending on purpose, is far from trivial due to that ships are highly non linear models, and are exposed to disturbances, both internal and external. Linear control laws are applicable but inefficient when dealing with non linear plants. The efficiency can be increased by applying a nonlinear control law instead [1]. Thus, a non linear control method is wanted to maximize the control performance in a non linear application.

Designing a control law for a stabilization system requires a known and accurate model for common controller designs. It takes time and effort to establish such a model. Conventional controller methods also leads to an iterative tuning procedure, which is a drawback [2]. Therefore it is of interest to design a controller which can adapt to arbitrary ship models by itself taking into account the many combinations of hull, propulsion and actuators.

This work covers ship dynamics modeling, control design and evaluation of the proposed controller.

Figure 1.1: *"Container Ship" by NOAA's National Ocean Service is licensed under CC BY 2.0. [3]*

1.1 Background and related work

This section investigate and evaluate previous papers that this work could benefit from. Below is a literature study of a ship's dynamic model and movements presented. The literature study includes information about existing stabilization systems and methods for controlling these. Furthermore, various types of adaptive control laws, preferable involving machine learning, are discussed and evaluated to determine if any previous findings are usable for this project.

1.1.1 Study on stabilization systems

To reduce the rolling motion of ships, they may be equipped with some kind of stabilizing system. These are either internal or external, i.e installed in or outside of the hull. Furthermore, existing stabilizing systems can be divided into two subgroups, passive and active systems.

Passive systems

Passive systems have no external inputs and do not require any power source, they work by themselves taking advantage of the laws of physics.

Bilge keel is a bar mounted at the hull below the waterline along with the flow of water on both the port and starboard side of a ship. This system increases the moment of inertia about the roll axis, damping the rolling motions of the ship. It is the most popular passive stabilization system [4].

Figure 1.2: *"Bilge keel on a steel vessel" by Dj245 at en.wikipedia is licensed under CC BY-SA 3.0 [5]*

Passive anti roll tank is a system that consists of a U-shaped tank filled with liquid. The slimmer lower limb is delaying the flow of liquid between the two sides causing a counter moment that counteracts rolling motions. The counter moment arises when the central gravity of the liquid mass is not in line vertically with the central gravity of the ship. The full performance of this system can though only be retrieved at one specific frequency of roll motion [4].

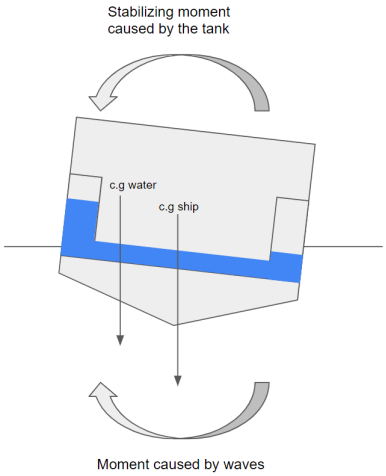


Figure 1.3: *Illustration of a passive anti roll tank system*

Active systems

Active systems require a power source, an external input, of some type that makes it possible to control the regulation of these systems.

Active flaps are adjustable plates placed in the rear underneath the ship attached to the hull. This system consist of a pair of plates which are possible to control separately. This property allows for both roll and pitch stabilization [6], [7]. In [6] was a roll reduction in the range of 40-55 % achieved.

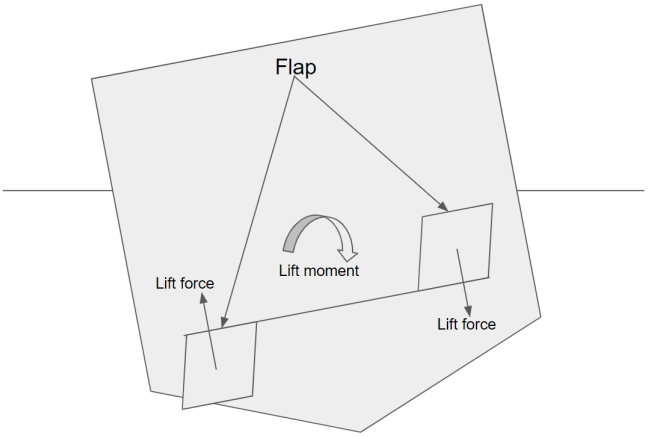


Figure 1.4: *Illustration of active flaps.*

Active anti roll tank systems are almost identical to the earlier mentioned passive anti roll tank system. What makes this system different is the ability to control the flow of liquid in the lower limb. This perk is making the stabilization efficient at a wider frequency range of roll motions. The flow is controlled by a pump which is moving the liquid mass, back and forth, to counter the roll more successfully. A roll reduction of 85 % was achieved for 6 DOF in [8] with the usage of active anti roll tanks. The mentioned ship was subjected to a sinusoidal disturbance where a PID controller was used to control the water flow between the tanks.

Active fin stabilizers is a system that consists of fins that are mounted on the hull of the ship. The most common configuration consists of a pair placed in the middle of the ship, but there exist systems where two pairs of fins are used as well. The fins are mounted on each side of the ship (port side and starboard side), parallel to each other, regardless of the configuration. They are placed far down on the hull such that they always are situated in the water. The fins are "active" which means that their pose is controllable to a certain degree. It is possible to control the angle of the fins and in some cases, it is possible to retract them, if they are not needed. A change in fin angle is causing a change in the forces caused by the fins. However, a flow rate must exist to obtain forces. A flow rate arises when fin and water are moving relative to each other.

The fin system can be used in two ways depending upon if the ship has a forward speed or not. When a speed is present each fin is causing two forces, a lift force and a drag force. The lift force is of interest since it give arise to a moment about the roll axis and can therefore be used to counter roll motions. The generated counter moment depends on flow rate, angle of attack, fin placement, boat dimensions, fin dimensions and water density [9]. At zero speed it is not possible to use the fin system in the way described above, instead it is possible to damp roll motions by moving water mass in order to create damping moments [10].

In [11], a roll reduction of 90.47 % was achieved with a PID controller. Another study [12] achieved a roll reduction of 88.04 % respective 97.19% for two different ships using PID controlled active fins.

Performance comparisons between systems has also been investigated, where the work in [13] compared the performance of an active anti roll tank system to an active fin system on the same ship model. The outcome of their comparison was that a higher roll reduction was achieved for the active fin system. Based on the higher roll reduction potential, this work is therefore using active fins as the stabilization system.

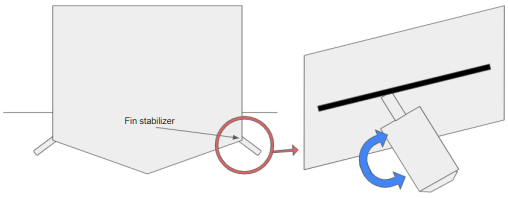


Figure 1.5: *Illustration of an active fin system*

An investigation of roll motion stabilization using an active fin system is done in [11]. In this work a comparison between a linear quadratic regulator (LQR) and a PID regulator is presented. It is mentioned in the paper that a PID has low robustness to model uncertainties and disturbances, which is not feasible for this work. A LQR is more robust but requires a known linear ship model to since it is a model-based controller, which is not of interest. What can be gained from [11] is

that the physical restrictions of the fin system have to be considered, namely the maximum and minimum allowed angle as well as a maximum angular velocity.

It is possible to use a pair of fins [11] as well as two pairs of fins [2]. More fins increase the amount of counter force that can be retrieved from the stabilization system but also increases the complexity of the model. With two pairs of fins, it is possible to stabilize roll and pitch simultaneous which is shown in [2]. An alternative method for stabilizing vessels moving at low speed with active fins is presented in [10]. This method uses forces that arise when a fin is pushing water mass in the moving direction of the fin. In [10] this is refereed to as "Additional Mass Force".

1.1.2 Study on adaptive control laws

Creating an adaptive control law for an unknown system requires an ongoing adjustment procedure that initially converges towards the current state of the system, and then onwards over time, compensates for internal and external changes of it. There exist methods for solving these kinds of problems where the system model is unknown. Two of them are model reference adaptive control (MRAC) and model identification adaptive control (MIAC) [14][15]. In MRAC a reference model is designed that refers to the ideal behaviour of the unknown system. The error between the two is used to change the control of the unknown system. The MIAC method distinguishes from MRAC by iteratively identifying the unknown model. The work in this paper does not require an identification of the unknown system, it only aims to achieve a desired system responses for given input signals. Therefore, the MRAC method is more suitable for this work.

MRAC can handle both linear and non linear unknown plants which are exposed to disturbances of various kinds. The modelling of the disturbances term ν^{ad} can typically be divided into three types: Structured uncertainty: when the structure of the uncertainty feature vector is known and static, Unstructured uncertainty: when the structure of the uncertainty feature vector is unknown and instead is estimated, Unmodeled dynamics: when the plant model consists of dynamics that are not measurable, observable or excluded in the reference model [14].

The uncertainty types to consider for this work is the unstructured uncertainty and unmodeled dynamics since external disturbances are random and also because limited knowledge of the plant is available. Therefore it is required to estimate the feature vector. Previous works have estimated it with good results by the use of Radial Basis Function Neural Networks (RBF) for positioning of a single link flexible manipulator [16] and control of unmanned areal vehicle [17] or with a Deep-Model Reference Generative Network (D-MRGeN) [18] for control of a quadcopter.

1.2 Purpose

This master thesis focuses on finding a novel way of controlling stabilizers on ships by the usage of Artificial Neural Networks (ANN) where the dynamics of the ship and actuators are assumed to be unknown. The ANN introduces non linearities in the control which hopefully outperform conventional controllers.

The main purpose of the introduction of an ANN based control is to get rid of the dependency on an accurate ship model. The idea is to let an ANN identify the ship dynamics in an initialization phase by interacting with the ship and use recorded data to perform backward propagation through the ANN. This introduces generality in the control and makes the control applicable to a wide range of ship configurations with different dynamics and parameters.

Another purpose of introducing an ANN control is to continuously update the ANN based on changes in the setting. The properties of ANNs allow for continuous adjustments, in this case, stabilization corrections based on change of conditions over time in both environment and ship. New conditions can result from wear, changes in the distribution of payload as well as driver behaviour.

1.2.1 Goals

- Develop an adaptive ship stabilizing controller that utilizes machine learning in combination with online learning to improve the present performance of a roughly estimated non-adaptive control configuration.
- Investigate if the proposed controller can handle dual control by including an additional stabilizing actuator.

1.2.2 Limitations

Some simplifications and limitations are made in this work. The external disturbance affecting the ship's dynamics for every basic movement at each time step is considered as a summation of all current disturbances. The ship is always heading straight, no turning motion is considered. It is also assumed that all state measurements of the ship are available and that there is no noise in the measurement model. The simulation environment utilized in this work assume the ship as a rigid body. The model does not involve all 6 DOF of a ship. It only covers 4 DOF where heave and pitch motions are excluded. Heave and pitch motions are usually small and neglectable since the large weight combined with the large transverse moment of inertia of ships allows them to cut through the majority of waves. In the rare case, if large heave and pitch motions arises when ships are exposed to rough seas, it is difficult to do anything about it, partly because it desires huge forces to affect heave and pitch motions because of the heaviness of ships. Neither do heave and pitch motions have a relation of significance to the rest of the basic movements. Thus, we have chosen not to include them in the simulation environment.

When mentioning the stability of a ship it is not directly referring to one specific basic movement, but when it comes to ship safety is longitudinal rotation (roll) the most critical movement [19]. Vertical rotation (yaw) is basically the heading of ships and is not critical for ship stability. Therefore it is reasonable to limit this work to mainly focus on reducing the ship rolling.

2

Theory

This chapter is covering basic theory related to the extent of this work. The chapter gives an introduction to ship modeling techniques, control laws and artificial neural networks (ANN).

2.1 Modeling of ship dynamics

For simulation purposes, a mathematical motion model of a ship is required. To model a ship there is a need for two reference frames, a ship-fixed frame and an Earth-fixed frame [20]. The two frames' relation to each other can be seen in Fig. 2.1, where X_0-Y_0-Z_0 is the Earth fixed frame and X-Y-Z is the body fixed frame.

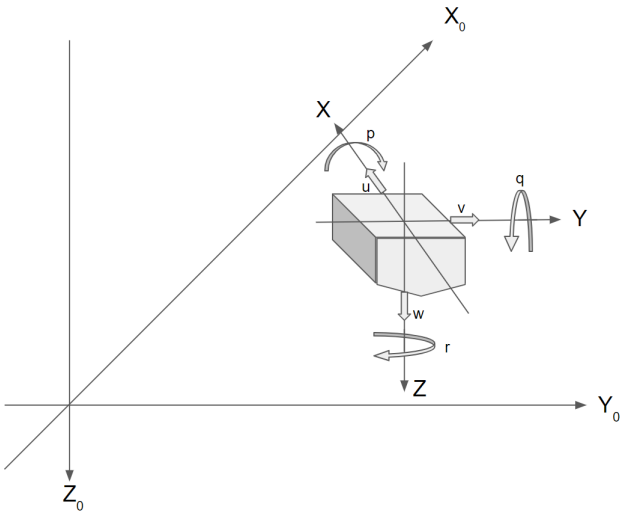


Figure 2.1: *Illustration of earth fixed frame X_0-Y_0-Z_0 and ship fixed frame X-Y-Z*

2.1.1 Motion equations

A ship has 6 degrees of freedom (DOF), three translational and three rotational components [20]. Motions in the X, Y and Z directions are often called surge (u), sway (v) and heave (w) and rotations about the same axes are referred to as roll (p), pitch (q) and yaw (r). From these components, it is possible to construct equations for the non linear ship dynamics based on Newton's second law [21].

$$\begin{aligned} m(\dot{u} + qw - rv) &= X_{tot} \\ m(\dot{v} + ru - pw) &= Y_{tot} \\ m(\dot{w} + pv - qu) &= Z_{tot} \\ I_x\dot{p} + (I_z - I_y)qr &= K_{tot} \\ I_y\dot{q} + (I_x - I_z)rp &= M_{tot} \\ I_z\dot{r} + (I_y - I_x)pq &= N_{tot} \end{aligned} \tag{2.1}$$

m is the mass of the ship. I_x, I_y and I_z are moments of inertia around the corresponding X, Y and Z axis. X_{tot}, Y_{tot} and Z_{tot} are the total forces present in each axis direction of the ship. K_{tot}, M_{tot} and N_{tot} are the total torque present around each axis of the ship.

2.1.2 Transformation between body and Earth frame

The orientation of the ship with respect to the Earth fixed frame can be described with the Euler angels: ϕ, θ and ψ [22]. ϕ, θ and ψ are angles about the Earth frame axes X_0, Y_0 and Z_0. Based on these it is possible to construct a velocity conversion from ship frame to Earth frame, see eq. 2.2. Due to limited space, $sin(\psi) = s\psi, cos(\phi) = c\phi$, etc..

$$\begin{bmatrix} \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.2}$$

2.2 System control

Designing a good control law is crucial for the overall performance of the system. There exist several different types and the most common when dealing with the actuators of a ship is presented in this section.

2.2.1 PID control

A common control strategy that is currently used by most active stabilizing systems is the usage of a PID controller [23]. A PID uses a feedback loop that returns the error, e(t), between the desired state, r(t), and the measured current state, y(t). The error is used throughout the next iterations to adjust the control signal with the target of decreasing the error. Three parameters, K_p, K_i and K_d have to be set manually when designing a PID, they correspond to a proportional, an integral and a derivative gain. The parameters are often tuned in the initialization process of the system and are adjusted to the response of a constant system configuration. The control law, $u(t)$, of a PID controller is given as:

$$\begin{aligned} u(t) &= K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{d}{dt} e(t) \\ e(t) &= r(t) - y(t) \end{aligned} \tag{2.3}$$

The problem with using a PID when controlling the stabilizing actuators of a ship is that in order to do that, a lot of knowledge of the ship dynamics is required. This is of course possible but takes a lot of time. Furthermore, if the ship changes over time the performance of the PID-controller changes as well and could potentially deteriorate until the point where it becomes unstable.

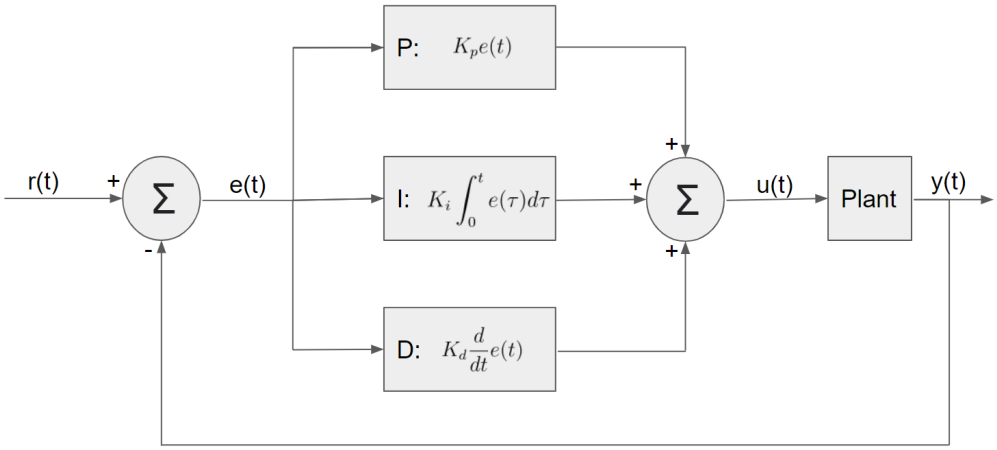
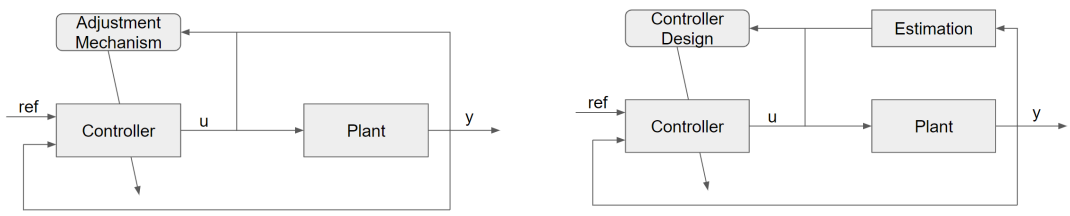


Figure 2.2: *Block scheme of a PID-controller*

2.2.2 Adaptive control

When designing a controller for an arbitrary system, the system dynamics are often assumed to be perfectly known. However, in real life applications, this is often not the case where parameter uncertainty, changes in the plant, non-linearities in the system or unmodeled behaviour leads to uncertainties in the model which in turn can lead to unwanted behaviour of the controlled system [14]. The goal of an adaptive control approach is to estimate these uncertainties and eliminate any unwanted behaviour such that the real system behaves as the modeled system even though the modeled system is imperfectly modeled. Complex systems could benefit from using an adaptive control by reducing the modeling time significantly and allowing the system to change over time without having to retune the controller.

There are generally two classes of adaptive control, direct adaptive control and indirect adaptive control. In direct adaptive control, the controller parameters are changed directly in advance of determining the process characteristics and disturbances. In indirect adaptive control, the plant model and disturbances are estimated instead. The control law is then updated based on this estimation. There exists some architectures that combine the two types which are referred to as combined, composite or hybrid direct-indirect adaptive control [23]. See Fig. 2.3 for an illustration of the differences between direct and indirect adaptive control.



(a) *Direct adaptive control* (b) *Indirect adaptive control*

Figure 2.3: *Example block schemes of direct and indirect adaptive control*

2.2.3 Model reference adaptive control

One of the main approaches to direct adaptive control is the MRAC approach [14]. Because of the unknown non-linear behaviour of the plant that is to be controlled, MRAC aims to design a controller that enforces the plant to follow the desired and known behaviour of a reference model. This is done by designing a controller that consists of two parts, one linear and one adaptive. The linear part is designed based on the dynamics of the reference model. If the reference model and the unknown plant were equal the adaptive part would be zero because the two system's behaviour would be the same. However, this is often not the case which leads to that an adaptive part needs to be introduced.

The adaptive part of the controller aims to eliminate any uncertainties that develop due to difference in dynamics between the reference model and the plant model. Traditionally, this is done by defining the model uncertainty as $\Delta(x) = W^{*\top}\phi(x)$ where ϕ is a known or unknown feature vector and W^* is the ideal feature weight vector. The adaptive part of the controller, ν^{ad}, can then be defined as $\nu^{ad} = W^{\top}\phi(x)$ where W is an estimation of the ideal feature weight vector W^*. A feature vector normally contains combinations of the system states. If $\phi(x)$ is known the uncertainty is called structured uncertainty. For the structured uncertainty case, a feature vector could for example look like the one in eq. 2.4.

$$\phi(x) = \begin{bmatrix} x_1 & x_1x_2 & x_3^2 & |x_4| \end{bmatrix} \tag{2.4}$$

The case when $\phi(x)$ is unknown is called un-structured uncertainty and is much trickier than the structured uncertainty case because the feature vector has to be estimated as well. The easiest way to do this is to approximate $\phi(x)$ as the system states x itself, i.e. $\phi(x) \approx \begin{bmatrix} x_1 & x_2 & ... & x_n \end{bmatrix}$. When the feature vector is established, the feature weight vector W is updated online using the MRAC-update rule. The idea is to move the feature weight vector in the direction of the error e which is the difference between the reference and plant states. The MRAC update rule is formulated as:

$$\dot{W} = -\Gamma\phi(x)e(t)^T PB_{rm} \tag{2.5}$$

Where Γ is a gain vector, B_{rm} comes from the input dynamics of the reference model and P is the solution to the Lyapunov equation $A_{rm}^T P + PA_{rm} + Q = 0$, Q, $P > 0$ where Q is a state-error penalty weighting matrix. The Lyapunov equation originates from stability theory regarding control applications.

2.3 Artificial neural networks

Artificial neural networks can be described as a series of functions combined to approximate an unknown function based on input and output data, see Fig. 2.4 for an example. An ANN consists of an arbitrary number of layers where each layer consists of an arbitrary number of nodes. The more layers and number of nodes in each layer the more complex the network becomes. Each node represents a linear combination between the inputs x and a bias b. The node's output y can mathematically be described as $y = Hx+b$, where H is a weight vector representing the combination of the inputs. The output node is fed forward as input to the next layer's nodes input either directly or through an activation function.

Function approximation is used to estimate an unknown function based on either stored or observed input/output data. An ANN has a universality property that makes it able to approximate any kind of unknown function regardless of the number of inputs and outputs (The Universal Approximation Theorem) [24].

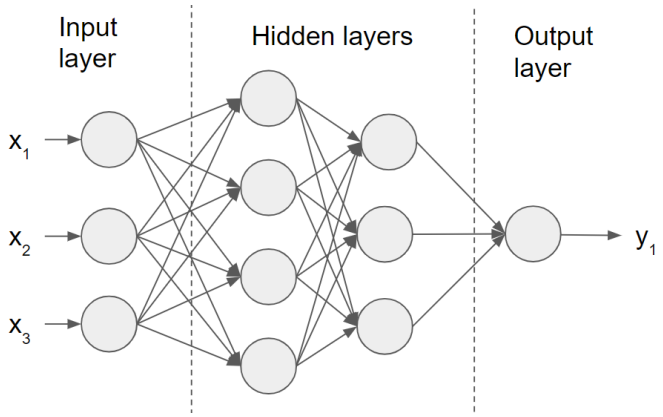


Figure 2.4: *Example of a feed forward network consisting of three inputs, two hidden layers and one output.*

2.3.1 Activation functions

The activation function takes a node's output as input and remaps it according to the specific choice of the activation function. The reason to use an activation function between the layers is to introduce non linearities in the network. Without an activation function, there is no point in having more than one hidden layer since adding additional hidden layers would still yield a linear network and could thus be merged into a single hidden layer. Examples of common choices of activation function are the sigmoid, tanh and rectified linear unit (ReLU) function [25]. The sigmoid activation function takes any real valued input from $(-\infty, \infty)$ and remaps it to the range of $[0, 1]$. Mathematically the sigmoid function can be represented as:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.6}$$

The tanh activation function is very similar to the sigmoid activation function but instead of having an output in the range of $[0, 1]$, the range is in $[-1, 1]$. Mathematically it can be represented as:

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.7}$$

The ReLU activation function sets all negative inputs to zero and forwards all the positive ones linearly which result in an output in the range $[0, \infty)$. Mathematically it can be described as:

$$f(x) = max(0, x) \tag{2.8}$$

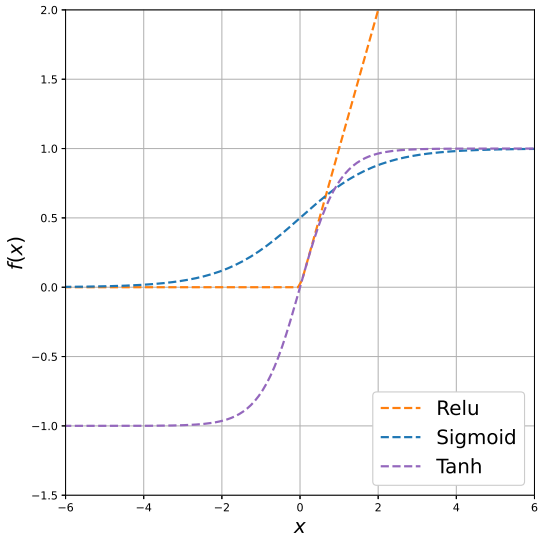


Figure 2.5: *Plot comparing the output of the sigmoid, tanh and ReLU activation function.*

2.3.2 Loss functions

In machine learning, the loss function evaluates how well the ANN is performing in a way that can be described mathematically. If the prediction made by the ANN is off from the truth the loss function's output value is high and if the predictions are close to the truth the loss function's output value is low. There exists numerous different loss functions but depending on the ANN's objective some functions are more suited than the other. For regression tasks the most common choice of loss function is the mean squared error (MSE), or the mean absolute error, (MAE) [25]. The MSE is calculated by taking the mean of the squared difference between the correct value, Y_i, and the ANN's predicted value, $\hat{Y}_i$ while the MAE is calculated by taking the mean of the absolute value of the difference instead. The main difference between the two loss functions is that predictions that are far from the observations are penalized heavier by the MSE than the MAE because of the squared error. The two loss functions are mathematically described in eq. 2.9.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2, \quad MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i| \tag{2.9}$$

2.3.3 Optimizer

The next step is trying to minimize the loss for future iterations. This is done with an optimizer and it is the optimizer's job to modify the net weights and biases of the network such that the loss decreases. A lower loss leads to a better performing ANN and in machine learning terms this is what makes the ANN learnable. Numerous optimizers have come and gone over the years each having its pros and cons. There exist no consensus on which optimizer is the best but one of the more common choices is the Adaptive Moment Estimation (ADAM) optimizer which is a combination of the predecessor's root mean square propagation and adaptive gradient algorithm [26]. The main idea with ADAM is to update the network parameters, θ, by taking a step in the opposite direction of the calculated exponential moving average of the gradient g and squared gradient g^2, where the

gradient g is the gradient of the loss with respect to θ. Mathematically the ADAM parameter update is defined as:

$$\begin{aligned} \theta_k &= \theta_{k-1} - \mu \frac{\hat{m}_k}{\sqrt{\hat{v}_k} + \epsilon} \\ \hat{m}_k &= \frac{m_k}{1 - \beta_1^k} \\ \hat{v}_k &= \frac{v_k}{1 - \beta_2^k} \\ m_k &= \beta_1 m_{t-1} + (1 - \beta_1) g_k \\ v_k &= \beta_2 v_{k-1} + (1 - \beta_2) g_k^2 \end{aligned} \tag{2.10}$$

The hyper parameters β_1, β_2 and ϵ have great default values which in most cases do not need to be deviated from. More specifically the default values are 0.9, 0.999 and 10^{-8} respectively [26]. The step-size parameter μ on the other hand needs to be tuned depending on the use case.

2.3.4 Normalization

Inputs x to an ANN often consist of different features with various units. The range of each input can vary a lot, feature a could be in the range 0-100 while feature b is in the range 0-10000. This causes an imbalance where values of b influence the output y more due its higher values while they are equally important in the evaluation of y. To avoid possible range differences in x it is possible to normalize each element in x such that they are all in the same range (usual range 0-1). Network performance is increased when normalizing input data x [27].

2.3.5 Regularization

Regularization is used in machine learning to affect the update of a learning algorithm to avoid overfitting, making prediction less exact and more general. Regularization can be included in the loss function in order to penalize large net weights.

3

Method

This section describes the methods used to simulate and investigate the performance of the proposed adaptive ship stabilizing controller. In the first part the simulation environment is described, i.e. how the ship, fin and disturbance dynamics are implemented. In the second part, the proposed control method is introduced. The last part describes how the controller performance was determined and measured.

3.1 Simulation environment

The proposed method is built upon the idea that a ship model does not exist or that only limited knowledge about the ship's dynamics is available. But, it assumes access to a physical ship with an active stabilization system installed, which is not available for this work. The physical model is instead replaced with a virtual model and evaluated in a simulation environment. This means that a mathematical ship model has to be formulated, preferably as close to a real ship as possible with high complexity and nonlinearities. Several detailed and complex ship models are available in the Marine Systems Simulator (MSS) Matlab and Simulink library [22]. The model of the naval vessel in MSS was choosen as ship model. It takes the current states and external forces as input and returns the state derivative. The chosen ship has a length of 51.5 meters between its perpendiculars (length), a beam overall (width) of 8.6 meters and weighs 362 000 kg.

Figure 3.1: *A naval vessel [22]*

The naval vessel model originates from a model that was formulated by A. G. Jensen and M. S. Chislett during the 80s and has since then been modified a few times. It was adapted for Matlab in 1996 by Mogens Blanke and Antiono Tiano. The last modification was done in purpose of matching the data of the vessel design of ADI-Limited Australia [22].

3.1.1 Ship dynamics

The vessel model consist of the 4 DOF: Surge, Sway, Roll and Yaw. Which means that the model neglects heave and pitch motions by excluding the equations for Y_{tot} and M_{tot} as well as q and $\dot{\omega}$ from the 6 DOF formulation in eq. 2.1. The vessel model takes two inputs, an external force vector $\tau(t)$ and a state vector $z(t)$, and returns the state derivative vector $\dot{z}(t)$. In the model many parameters are set, such as vessel dimensions, mass, moments of inertia, displacement, center of gravity, center of buoyancy, metacentric height and a lot of hydrodynamic coefficients. The dynamics of the naval vessel are highly non linear. The naval vessel state vector $z(t)$ contains the states

$$z(t) = \begin{bmatrix} v_x(t) \\ v_y(t) \\ \dot{\Phi}(t) \\ \dot{\Psi}(t) \\ \Phi(t) \\ \Psi(t) \end{bmatrix} = \begin{bmatrix} \textit{surge velocity} & \textit{(m/s)} \\ \textit{sway velocity} & \textit{(m/s)} \\ \textit{roll velocity} & \textit{(rad/s)} \\ \textit{yaw velocity} & \textit{(rad/s)} \\ \textit{roll angle} & \textit{(rad)} \\ \textit{yaw angle} & \textit{(rad)} \end{bmatrix}, \quad \tau(t) = \begin{bmatrix} X_e(t) \\ Y_e(t) \\ K_e(t) \\ N_e(t) \end{bmatrix} \tag{3.1}$$

and τ consist of the external forces and moments affecting the vessel, see eq 3.1 for definition. As mentioned earlier, the ship moves at a constant speed. In order to achieve this in the MSS Naval Vessel script a constant thrust is introduced. More specifically it is added to the term X_e in the external force vector τ. Once the state vector $z(t)$ and the external force vector $\tau(t)$ are established, the dynamics of the naval ship can be simulated according to eq. 3.2 where f is the MSS Naval Vessel script.

$$\dot{z}(t) = f(z(t), \tau(t)) \tag{3.2}$$

For simulation purposes the time is discrete and because of this the state vector z(t) is updated using the Euler method described as:

$$z_{k+1} = z_k + \Delta t \dot{z}_k \tag{3.3}$$

where $\Delta t = 0.05s$ is the sample rate.

3.1.2 Fin and rudder dynamics

The MSS Naval vessel script does not have either a fin or rudder system implemented. Thus, their dynamics and dimensions had to be formulated and integrated into the existing motion dynamics in eq. 3.2.

The fin stabilizers and the rudder are controllable actuators, meaning that they can, given input signals and a power source, move to desired positions. They have the physical characteristics of a hydrofoil, which is a solid object shaped in a manner such that two forces arise when water is streaming uniformly around it. The two forces are a lift force F_{lift} and a drag force F_{drag}. The lift force is perpendicular to the water flow while the drag force is parallel to the water flow, see Fig. 3.2 for clarification.

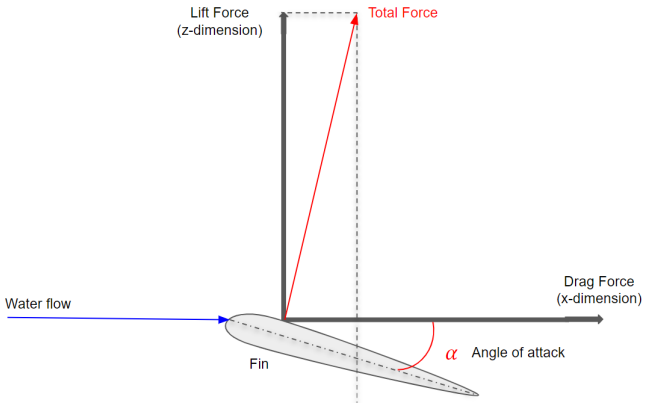


Figure 3.2: *Illustration of the forces acted on a hydrofoil*

The ship has three hydrofoils installed, with similar shape. Their installation positions differ, which results in different hydrodynamic properties for each. Two of the three are placed parallel on each side of the bilge of the ship, beneath the waterline. These are the so-called fin stabilizers. The last one is placed under the stern, behind the propeller. It is referred to as the rudder. A rudder is a vertically rotated fin that is mounted behind the ship's propeller. A rudder has similar characteristics to a fin which can be determined from the rudder equations in [28].

One important aspect to consider, regarding the fin stabilizers, is that they only work as intended when the ship is moving forwards through water. The rudder on the other hand can contribute even if the ship is not moving due to its placement behind the propeller. The propeller can create a water flow at the rudder of a resting ship, making it possible to control the rear of the ship sideways even at low speeds. But at a constant high speed, the propeller's impact on the water flow is small and therefore its effect on the rudder is neglected [28]. It is further assumed that the water flow at each hydrofoil is always homogeneous and still such that the relative velocity between fin and water is equal to the velocity of the ship.

The control of the fin stabilizers are coupled, meaning that they receive the same control signal. The fins rotate an equal amount but in opposite directions around their axes. They would otherwise cancel out the created stabilization moment. The rudder is not connected to the fin system and receives its own control signal. Thus, the controller output either one control signal or two control signals depending on if there is only fin control or fin and rudder control. From now on these two types of control signals will be referred to as single control and dual control.

The generated forces extracted from a hydrofoil depend on several parameters, but the most important, and controllable, is the so-called angle of attack α. It is the angle difference between the angle of the water flow and the hydrofoil's angle where the latter can be controlled. An illustration of this can be seen in Fig. 3.2. The water flow is assumed to always be parallel to the ship's forward motion, which implies that a static angle of water flow is assumed. The drag force is present parallel to the hull and in the opposite direction of the moving direction regardless of the selection of α. A fin with $\alpha = 0$ generate a drag force, while the lift force is equal to zero.

The equations for calculating F_{lift} and F_{drag} is adopted from [22]. Two important coefficients when determining F_{lift} and F_{drag} are the hydrodynamic lift CL and drag CD coefficients. They have to be determined before being able to calculate F_{lift} and F_{drag}. Other parameters present in the coefficients equations are Oswald efficiency number e, wing aspect ratio AR, wing area S, wingspan b, parasitic drag CD_0, and blending parameter γ. The equations below describe CL and CD in relation to α. The hydrodynamic coefficients for the rudder are gained by replacing α with β in eq. 3.4.

$$\begin{aligned} AR &= \frac{b^2}{S} \\ CL_\alpha &= \frac{\pi AR}{1 + \sqrt{1 + (\frac{AR}{2})^2}} \\ CL_{linear} &= \alpha CL_\alpha \\ CD &= CD_0 + \frac{CL_{linear}^2}{\pi e AR} \\ CL &= (1-\gamma)CL_{linear} + 2sign(\alpha)sin(\alpha)^2 cos(\alpha) \end{aligned} \tag{3.4}$$

CL and CD depend on the dimension of the hydrofoil. The dimensions of the two stabilizing fins are identical which means that their lift and drag coefficients are equal. The rudder has not the same dimension as the fins which results in different lift and drag characteristics i.e. different coefficients.

When CL and CD are available it is possible to formulate the equations for F_{drag} and F_{lift} as:

$$\begin{aligned} F_{drag} &= \frac{1}{2}\rho U_r^2 SCD \\ F_{lift} &= \frac{1}{2}\rho U_r^2 SCL \end{aligned} \tag{3.5}$$

Where ρ is the water density and U_r is the relative velocity between hydrofoil and water. The equations in eq. 3.5 holds for all hydrofoils.

F_{drag} and F_{lift} for every hydrofoil affects the motion dynamics components of the ship's total force components X_{tot}, Y_{tot} och Z_{tot}. This work assume for simplicity that the fins are parallel to the Y axis and that the rudder is parallel to the Z axis of the ship. F_{drag} and F_{lift}, for every hydrofoil, have to be transformed into the body force components of the ship. The transformations to the body force components can be seen in eq. 3.6.

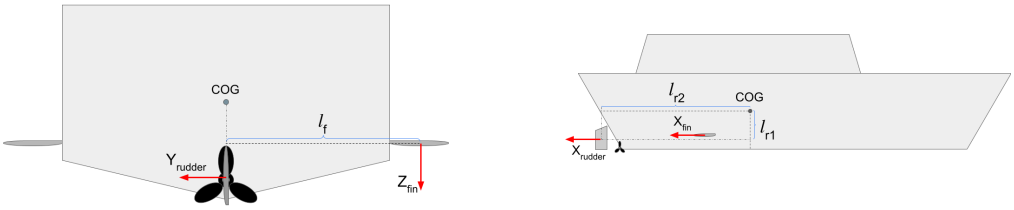
$$\begin{aligned} X_{fin} &= \cos\alpha(-F_{drag}) - \sin\alpha(-F_{lift}) \\ Z_{fin} &= \sin\alpha(-F_{drag}) - \cos\alpha(-F_{lift}) \\ X_{rudder} &= \cos\beta(-F_{drag}) - \sin\beta(-F_{lift}) \\ Y_{rudder} &= \sin\beta(-F_{drag}) - \cos\beta(-F_{lift}) \end{aligned} \tag{3.6}$$

The direction of the force components X_{fin}, Z_{fin}, X_{rudder} and Y_{rudder} is not through the center of gravity (COG), which means that these forces cause torque. The lever arm for the fin forces is l_f. Since the fins always have the same angle of attack and

are placed symmetrical, their generated moments are double about the X axis and canceled about the Z axis. Lever arms for the rudder forces are l_{r1} and l_{r2}.

$$K_{fin} = 2Z_{fin}l_f \tag{3.7}$$
$$N_{fin} = X_{fin}l_f - X_{fin}l_f \tag{3.8}$$
$$K_{rudder} = Y_{rudder}l_{r1} \tag{3.9}$$
$$N_{rudder} = Y_{rudder}l_{r2} \tag{3.10}$$

M_{fin} caused by Z_{fin} and M_{rudder} caused by X_{rudder} are both neglected since the Naval vessel model do not include moments about the Y axis.



(a) (b)

Figure 3.3: *Illustration of the placement of foils and their resulting forces.*

The dimensions of the hydrofoils were selected in relation to the size of the naval vessel. More specifically, the exact dimensions of the fin and rudder are adopted from a NACA0015 foil and can be seen in Tab. 3.1 [29].

Table 3.1: Hydrofoil dimensions

	Fins	Rudder
b (m)	1	1.5
S (m^2)	2.25	3
e	0.7	0.7
CD_0	0.15	0.15
γ	0.5	0.5

3.1.3 Modeling of wave and wind disturbances

In order to test the performance and stability of the control algorithm, some kind of disturbance that resembles wave and wind disturbances has to be introduced to the plant. Without any kind of disturbances the plant's states would converge towards zero since the ship is a stable system and the results would therefore be uninteresting. In a physical environment these kind of disturbances would occur naturally but in the simulation environment they do not. Thus they need to be modeled and implemented in the plant dynamics.

To model and calculate the external forces on the ship hull that are generated by sea waves and winds are computationally very expensive. Instead, the typical control engineering approach to emulate wave and wind behaviour is to superimpose the plant's velocities with a white Gaussian noise that is propagated through a

second-order transfer function [30]. More specifically this was done according to eq. 3.11 on both the roll and yaw axis where x_d is the disturbance state, ω_0 is the peak frequency of the disturbance, λ_d is a damping term and K_d is an amplitude gain.

$$\begin{aligned} \dot{\mathbf{x}}_d &= \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & -2\lambda_d\omega_0 \end{bmatrix} \mathbf{x}_d + \begin{bmatrix} 0 \\ K_d \end{bmatrix} d, \text{ where } d \sim \mathcal{N}(0,1) \\ \eta_d &= \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}_d \end{aligned} \tag{3.11}$$

After η_d is calculated it is added to row 3 and 4 in eq. 3.3.

When the sea condition is in heavy swell and the average wavelength is 100-200 m, the wave time period T_w is in the range of 8 to 11 seconds [31]. If the shortest wave period $T_w = 8$ seconds is assumed this results in a peak frequency of $\omega_0 = \frac{2\pi}{T_w} = \frac{\pi}{4}$. Furthermore, to make the external disturbance challenging but still reasonable for the stabilization system to handle, the selection of K_d and λ_d was tested iteratively. This resulted in the specific parameter settings where, unless stated otherwise, $\omega_0 = \frac{\pi}{4}$, $\lambda_d = 0.1$ and $K_d = 2.4$ for both the roll-rate and yaw-rate disturbance. See Fig. 3.4 for a plot of the typical resulting disturbance and the corresponding response of the naval vessel when not controlled.

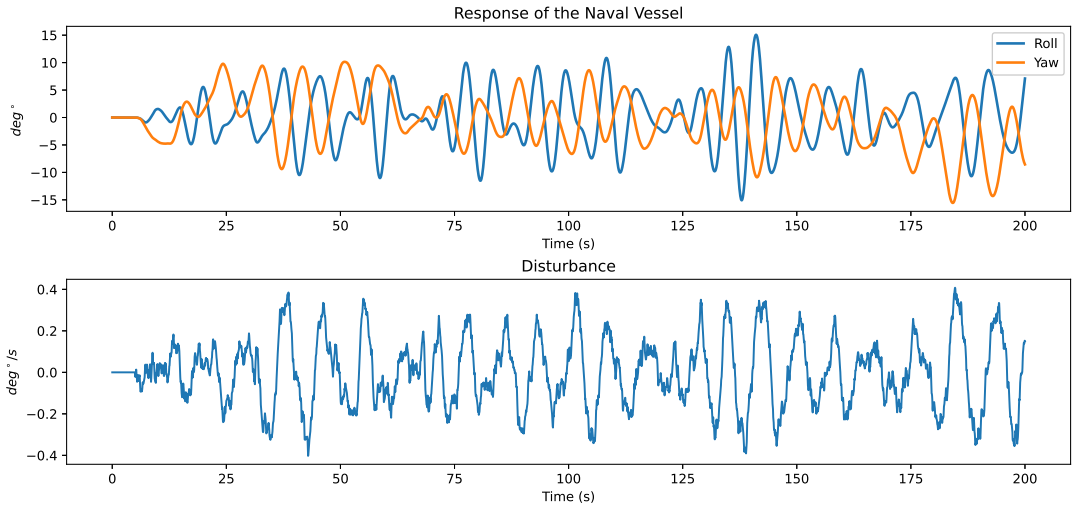


Figure 3.4: *Plot of the typical generated disturbance and the corresponding response of the ship.*

3.2 The proposed control algorithm

This section describe the proposed modified DMRAC control algorithm which is based on the original introduced by [18]. The aim of the control algorithm is to enforce the unknown plant model to track a given reference model. The two models are defined as:

$$\dot{x}(t) = f(x(t), u(t)) \tag{3.12}$$
$$\dot{x}_{rm}(t) = A_{rm}x_{rm}(t) + B_{rm}r(t) \tag{3.13}$$

Where $x(t)$ is the state vector of the unknown ship, $u(t)$ is the control input vector to the unknown ship, $x_{rm}(t)$ is the state vector of the reference model, A_{rm} is the system matrix of the reference model, B_{rm} is the control matrix of the reference model and $r(t)$ is the reference signal which in a stabilizing case, is always set to zero. Furthermore, to also model uncertainties eq. 3.12 can be re-written in form of nominal dynamics as:

$$\dot{x} = Ax(t) + B(u(t) + \Delta(x)) \tag{3.14}$$

Where $\Delta(x) \triangleq f(x(t), u(t)) - Ax(t)$ is the uncertainty of the model. The control law u(t) was then designed on the form:

$$u(t) = u_{pd}(x(t), r(t)) - \nu^{ad}(x(t)) \tag{3.15}$$

Where u_{pd} is a linear PD controller and ν^{ad} is a non linear adaptive controller. The task for the adaptive part of $u(t)$ is to eliminate any unwanted behavior of the plant that the linear control part or external disturbances might introduce, while also capturing the unmodeled nonlinearities the plant possesses. The adaptive part is decided by an ANN as:

$$\nu^{ad} = W^{\top}\phi(x(t)) \tag{3.16}$$

Where W is the feature weight vector and the feature vector $\phi(x(t))$ is an intermediate output of the ANNs last hidden layer. The feature weight vector originates from the MRAC rule update, the principles of it, are described in section 3.2.5.

An overview of the scheme of the control algorithm is shown in Fig. 3.5. In the following sections the details of respective block are presented.

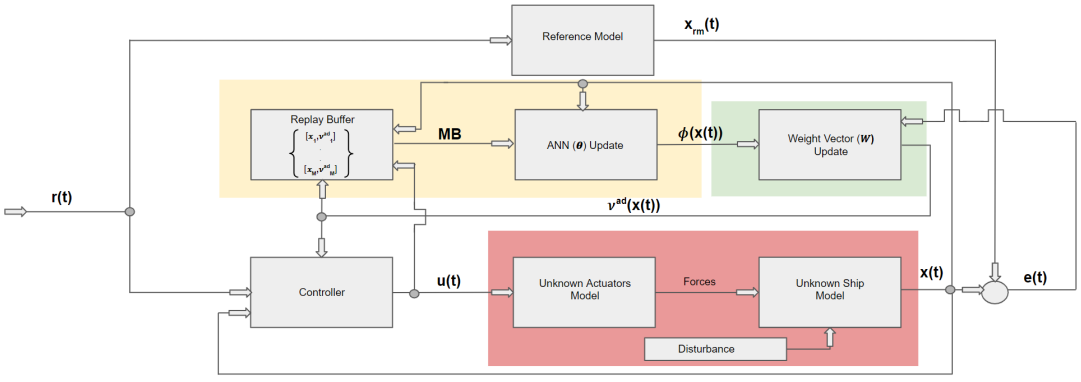


Figure 3.5: *Scheme*

3.2.1 Reference Model

The reference model specifies the desired response of the unknown system to a command input and is used to enforce the unknown model to track the reference model by minimizing the error $e(t) = x(t) - x_{rm}(t)$. For the system to be general and able to handle arbitrary ships the reference model is kept simple, leaving out nonlinearities as well as information about ship dimensions, weight, stabilization system and such. Since the task of the control system is to stabilize the ship around the roll axis or roll and yaw-axis i.e the reference is always zero, a naive choice of reference model could thus be to set all the states to zero since that would represent the perfect behaviour. However, since the reference model is also used to calculate the linear control part of the control law, that does not yield any useful information about the desired dynamics between the states. Instead, the reference model is formulated as a second-order LTI system and is described by the state-space system in eq. 3.17. The specific reference model used when only stabilizing the roll-axis and the reference model used when stabilizing both roll and yaw-axis are presented in state-space form in eq. 3.18 and eq. 3.19.

$$\begin{aligned} \ddot{x}_{rm}(t) + 2\zeta_{rm}\omega_{rm}\dot{x}_{rm}(t) + \omega_{rm}^2 x_{rm}(t) &= \omega_{rm}^2 r(t) \\ \dot{\mathbb{X}}_{rm}(t) = A_{rm}\mathbb{X}_{rm}(t) + B_{rm}r(t), \quad \mathbb{X}_{rm} &= \begin{bmatrix} x_{rm} & \dot{x}_{rm} \end{bmatrix}^\top \end{aligned} \tag{3.17}$$

$$A_{rm} = \begin{bmatrix} 0 & 1 \\ -2\zeta_{rm}\omega_{rm} & -\omega_{rm}^2 \end{bmatrix}, \quad B_{rm} = \begin{bmatrix} 0 \\ \omega_{rm}^2 \end{bmatrix}, \quad x_{rm}(t) = \begin{bmatrix} \Phi_{rm}(t) \\ \dot{\Phi}_{rm}(t) \end{bmatrix} \tag{3.18}$$

$$\begin{aligned} A_{rm} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -2\zeta_{rm}\omega_{rm} & -\omega_{rm}^2 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -2\zeta_{rm}\omega_{rm} & -\omega_{rm}^2 \end{bmatrix}, \quad B_{rm} = \begin{bmatrix} 0 & 0 \\ \omega_{rm}^2 & 0 \\ 0 & 0 \\ 0 & \omega_{rm}^2 \end{bmatrix}, \\ x_{rm}(t) = \begin{bmatrix} \Phi_{rm}(t) \\ \dot{\Phi}_{rm}(t) \\ \Psi_{rm}(t) \\ \dot{\Psi}_{rm}(t) \end{bmatrix} \end{aligned} \tag{3.19}$$

3.2.2 Unknown plant model

The main idea is, as mentioned earlier, to control the unknown ship model. Unknown, in this case, refer to the naval ship including fin and rudder system and disturbances. Therefore, an extension was made to the simulation environment that applied the fins and rudder dynamics to the naval ship. It was done by implementing the equations presented in section 3.1.2 in the simulation environment. This led to that the vessel state update function in eq. 3.2 was extended to also take the control signal $u(t)$ as input, see eq. 3.20. Where $u(t)$ is the desired fin angle α, or fin angle α and rudder angle β, see eq. 3.21 for clarification.

$$\dot{z} = f(z(t), \tau(t), u(t)) \tag{3.20}$$

$$u(t) = \begin{bmatrix} \alpha(t) \end{bmatrix} \quad or \quad u(t) = \begin{bmatrix} \alpha(t) \\ \beta(t) \end{bmatrix} \tag{3.21}$$

In reality, there are physical limitations to the fin and rudder systems. It is not possible to rotate the hydrofoils 360 degrees, nor have a too large relative change of pose between two sample steps. Therefore constraints regarding this were implemented where an absolute allowed angle span and a maximum allowed angle rate were specified.

$$\begin{aligned} |\alpha| &\leq \alpha_{max} \quad |\beta| \leq \beta_{max} \\ |\dot{\alpha}| &\leq \dot{\alpha}_{max} \quad |\dot{\beta}| \leq \dot{\beta}_{max} \end{aligned} \tag{3.22}$$

More specifically the hydrofoils were limited to a maximum angle of $\pm 30°$ and a maximum angle rate of $\pm 30°/s$ [32].

A part of the states in z is passed over to x because z is containing unnecessary states for the control law. More specifically, the values of v_x and v_y are not of interest because these are not included in the reference model. x is formatted depending on the control signal u's dimension, i.e. if there is only roll control, or roll and yaw control.

$$x(t) = \begin{bmatrix} \Phi(t) \\ \dot{\Phi}(t) \end{bmatrix} \quad or \quad x(t) = \begin{bmatrix} \Phi(t) \\ \dot{\Phi}(t) \\ \Psi(t) \\ \dot{\Psi}(t) \end{bmatrix} \tag{3.23}$$

3.2.3 Controller

The control signal $u(t)$ of the unknown ship consists of two main components, a linear part $u_{pd}(x(t), r(t))$ and a non-linear part $\nu^{ad}(x(t))$. As mentioned earlier, the task of the adaptive control signal $\nu^{ad}(x(t))$ is to eliminate the model uncertainty $\Delta(x(t))$. The plant model is unknown which implies that the true uncertainty $\Delta(x(t))$ is as well [18]. Because of this, it is instead estimated. The estimation is done by defining the adaptive part $\nu^{ad}(x(t))$ from eq. 3.16 as the estimated model uncertainty $\hat{\Delta}(x(t))$, i.e. $\hat{\Delta}(x(t)) \triangleq \nu^{ad}(x(t))$. The estimated uncertainty $\hat{\Delta}(x(t))$ is ideally equal to the true uncertainty $\Delta(x(t))$.

The linear part of the control signal, $u_{pd}(x(t), r(t))$ can be divided into two subparts, a feedforward component $k_{ff}r(t)$ and a feedback component $k_{fb}x(t)$. In this work, the feedforward component is neglected since the reference $r(t) = 0$ during the whole simulation. This resulted in the final control law that can be seen in eq. 3.24.

$$u(t) = u_{pd}(x(t), r(t)) - \nu^{ad}(x(t)) = k_{fb}x(t) - \nu^{ad}(x(t)) \tag{3.24}$$

where k_{fb} is a tuneable gain matrix.

3.2.4 ANN update

The ANN is used for the establishment of the feature vector $\phi(x)$ and prediction of the estimated model uncertainty $\hat{\nu}^{ad}(x)$. The ANN consists of an input layer with the same dimension as the state vector $x(t)$, three hidden layers and an output layer with the same dimension as the control signal $u(t)$. Thus, five layers in total. The

input data to the network is the state vector $x(t)$, and the output is the predicted uncertainty $\hat{\nu}^{ad}$. The three hidden layers have 128, 256 and 64 neurons respectively. All layers are fully connected. The ANN parameters θ is updated partly based on the network loss L_{net} gained from feeding collected data from a mini-batch MB with MB_{size} number of previous x - $\nu_{ad}(x)$ pairs through the net where the output $\hat{\nu}^{ad}(x)$ is the net's prediction of the estimated uncertainty $\nu^{ad}(x)$. The total L_{net} loss is the MSE over the whole MB.

$$L_{net} = \frac{1}{MB_{size}} \sum_{i=1}^{MB_{size}} (\nu_i^{ad} - \hat{\nu}_i^{ad})^2 \tag{3.25}$$

L_{net} is combined with a regularization term to form the total loss L. Regularization is applied to avoid the possibility of overfitting the data. A common regularization method is L2 regularization [33]. L2 regularization is used to penalize large net weights and forces them into the neighbourhood of 0. The L2 regularization formula is

$$L_{L2} = \sigma \sum_{i=1}^{N} \omega_i^2 \tag{3.26}$$

where N is the total number of net weights in the ANN, ω_i is the value of weight number i and σ (> 0) is a tuning parameter which is deciding the influence of the regularization term. The regularization is completed by adding L_{L2} and L_{net} together to form L

$$L = L_{net} + L_{L2} \tag{3.27}$$

When the total loss L has been calculated it is then used in the ADAM optimizer to update the net weights θ using eq. 2.10.

To increase network performance, the input $x(t)$ is normalized. Furthermore, as activation function for the three hidden layers, tanh is used. In eq. 3.28-3.31 the mathematical description of the whole network architecture can be seen where θ_n is the vector containing the net weights of the n'th ANN layer and b_n is the bias term of layer n.

$$h_1(x) = tanh(\theta_1 norm(x) + b_1) \tag{3.28}$$
$$h_2(x) = tanh(\theta_2 h_1(x) + b_2) \tag{3.29}$$
$$\phi(x) = tanh(\theta_3 h_2(x) + b_3) \tag{3.30}$$
$$\hat{\nu}^{ad}(x) = \theta_4 \phi(x) + b_4 \tag{3.31}$$

The feature vector $\phi(x)$ is the output of the third hidden layer of the ANN, see eq. 3.30, when feeding forward the state vector x. $\phi(x)$ is calculated every iteration. To decrease the computation time of the control algorithm and prevent overfitting, the net parameters θ were not updated every iteration. Instead, $N_{batches}$ number of MBs, containing MB_{size} samples each, was sampled and trained on every N_θ sample.

3.2.5 Feature weight vector update

The feature weight vector W, which is not to be confused with the network weights θ, was updated at each iteration with the use of the MRAC rule described in eq.

2.5. At the beginning of each simulation, it was initialized as a zero vector.

If the control input, u_k, differs too much from the previous input, u_{k-1}, the velocity and amplitude constraint on the fin's angle of attack are compromised, i.e the controller's output is physically not possible to achieve. If this occurs the control signal is clipped due to not being a valid control signal. A side effect of this is that there is now a risk that the feature weight vector starts to diverge due to the nature of the DMRAC system. If the feature weight vector diverges the performance of the controller deteriorates dramatically and the ship becomes unstable. Ideally, a constraint on the feature weight vector update should be introduced. This is not possible since the equation system describing the feature weight update constraints is underdetermined. Instead, an extra check was added that checked if any constraints were broken by the new input. If that was the case, the feature weight vector update got cancelled and the sample was not added to the replay buffer for that time index.

However, despite the previous counter-measure, the feature weight vector was still able to diverge if the initialization of the controller was unlucky. This was because during the initialization the error between the plant and reference model is very small. This led to that the feature weight vector could potentially become very large while the adaptive control part remained small. When the error increased the adaptive control part got too large and triggered the no update rule mentioned earlier. Since the feature weight vector was too large already led to that as long as the error did not converge the feature weight vector never got updated leading to very poor performance of the controller.

To remedy this an MRAC-gain schedule was implemented. Where the MRAC-gain, Γ, was set very low for the first 10 seconds of the simulation to later be set to the normal gain. The reasoning behind this was to ensure that the feature weight vector and feature vector were properly initialized before enabling the algorithm to make drastic changes that potentially could lead to a feature weight vector that never gets updated. This proved to remove the randomness of success during the initialization and lead to a much more robust adaptive part of the controller.

3.2.6 Replay buffer

The purpose of a replay buffer is to store data, on which the ANN could train. Since the learning is online, data points were collected and stored at each time index. Since the execution of the algorithm could go on for an arbitrarily long time it is not possible to store all of them, hence a maximum of RB_{size} samples are stored in the buffer.

Even if it was possible to store all of the previous samples it would not be desirable to do so. This was because if the ship changes over time the old samples would no longer reflect the new behaviour and could potentially if sampled, lead to a worse performing network. Hence, a first in, first out rule was implemented when the buffer is filled.

The data samples that were stored at each times index were the tuple of the

current state of the plant $x(t)$ and the estimated uncertainty $\Delta(\hat{x}(t))$ which was earlier said to be equal to ν^{ad}. When the replay-buffer was implemented it enabled the algorithm to sample mini-batches (MBs) of data which the slower feature learning could train on. The MBs were sampled uniformly from the replay buffer (RB).

$$RB = \begin{bmatrix} [x_1, \nu_1^{ad}] \\ [x_2, \nu_2^{ad}] \\ . \\ . \\ [x_M, \nu_M^{ad}] \end{bmatrix} \tag{3.32}$$

3.2.7 Algorithm

Simulations of the controlled plant were done according to the algorithm shown in Alg. 1.

Algorithm 1 main loop

Input: $\Gamma, \mu, x_0, \tau_0, \alpha_{max}, \dot{\alpha}_{max}, \beta_{max}, \dot{\beta}_{max}$, RB_{size}, MB_{size}, $N_{batches}$, N_θ
Initialize environment with state $x \leftarrow x_0$ and external force vector $\tau \leftarrow \tau_0$
Initialize reference model with state $x_{rm} \leftarrow x_0$
Initialize controller network with net weights θ and basis layer ϕ
Initialize MRAC feature weight vector W
Initialize empty replay buffer RB
Create disturbance vector η according to eq. 3.11
while True **do**
  compute $\phi(x)$ according to eq. 3.30
  update W according to eq. 2.5
  compute new control u according to eq. 3.24
  apply control u, τ and disturbance ν to environment
  observe new state of the plant
  if u broke the constraints in eq. 3.22 **then**
    revert W to previous value
  end if
  if update ANN **then**
    for Batches **do**
      sample batch from RB
      update θ using MSE over batch and ADAM
    end for
  end if
  if RB full **then**
    remove oldest sample from RB
  end if
  store $[x, \nu^{ad}]$ in RB
end while

3.3 Measurement of performance

In this section, a framework for determining the DMRAC performance is described. To obtain a good performance the selection of parameters is important, thus a parameter evaluation is necessary. Furthermore, test cases where the plant model is exposed to internal and external disturbances to a various extent are introduced. The test cases are divided into two parts, single control and dual control.

3.3.1 Roll reduction ratio

A common index when evaluating the effectiveness of ship stabilization around the roll axis is the Roll Reduction Ratio (RRR) [32]. RRR is a damping ratio (%) telling how effective a controller is at reducing rolling motions by comparing a controlled plant against an uncontrolled plant. The mathematical formulation of RRR can be seen in eq. 3.33 where RSDU is the standard deviation of roll amplitude when the control is deactivated and RSDC is the standard deviation of roll amplitude when the control is turned on. To have a fair comparison, the controlled and uncontrolled plants were exposed to the same internal and external disturbances during the simulations.

$$RRR = 100\frac{RSDU - RSDC}{RSDU} \tag{3.33}$$

Furthermore, the initialisation phase of the simulation was excluded when calculating the RRR for the respective case, more specifically the first 50 seconds were excluded. This is because the control is initially poor before it converges to the neighbourhood of the optimal control.

3.3.2 Criteria for effectiveness of the crew

To understand the working environment for the crew on board ships as well as passenger comfort, a framework was determined for this cause. A list of such is presented in [34] where criterion's of what type of work that is to be expected in different sea states are presented. The criterion is based on what was formulated in [35]. In Table 3.2, a list of certain roll root mean square (RMS) values for different types of work can be seen. The RMS value is calculated according to eq. 3.34 where N is the total number of samples. The first 50 seconds were excluded when calculating roll RMS due to the same reason as in section 3.3.1.

$$RMS = \sqrt{\frac{1}{N - \frac{50}{\Delta t}}(\Phi^2_{(50/\Delta t)+1} + \Phi^2_{(50/\Delta t)+2} + ... + \Phi^2_{(N/\Delta t)-1} + \Phi^2_{N/\Delta t})} \tag{3.34}$$

Table 3.2: Criteria for effectiveness of the crew [34]

RMS	Description of what work to be expected
6 deg	Light manual work
4 deg	Heavy manual work
3 deg	Intellectual work
2.5 deg	Transit passengers
2 deg	Cruise liner

3.3.3 Parameter setting evaluation

The selection of the parameters RB_{size}, MB_{size}, $N_{batches}$, μ, Γ and N_θ are affecting the overall DMRAC performance. Therefore it was of interest to find a setting that yields a good outcome for the single control application. A problem was that there exist unlimited combinations of these parameters. To have a parameter evaluation that was within a reasonable scope, were only a limited amount of parameter combinations considered. In Table 3.3, all parameter values that were investigated can be seen. When combined, it resulted in a total of 96 unique DMRAC settings which were evaluated on the Base case, see section 3.3.4.

Table 3.3: Parameters to be evaluated for single control

Parameter	Values
RB_{size}	[500,1000]
MB_{size}	[25,50]
$N_{batches}$	[3,6]
μ	[0.01, 0,001]
Γ	[1,10]
N_θ	[100,200,400]
Q	I_2

In order to make a fair evaluation, every DMRAC setting was exposed to the same wind and wave disturbance. The disturbance was generated as described in section 3.1.3. The performance of each setting was determined based on RRR, see section 3.3.1. The 10 best performing settings were then further evaluated in an additional step to increase the validity that a setting really had good performance, reducing the fortuity. More specifically, each of the 10 chosen settings were exposed to 10 different wave and wind disturbances. Where the one with the highest average RRR score was selected and used for the proposed test cases in the next section.

The introduction of an additional reference signal and expansion of control for the dual control introduced a different problem configuration. The selection of RB_{size}, MB_{size}, $N_{batches}$, μ and N_θ remains the same since these form ϕ. The uncertainty features are the same for both single and dual control, but the influence of each feature element is different. The difference in influence is caused by W which is two-dimensional for the dual control. W was updated according to eq. 2.5, where Γ and Q are tuneable parameters. Γ was extended in the dual case into a vector as $\Gamma = [\Gamma_\alpha, \Gamma_\beta]^T$, while Q became a four-dimensional square matrix. The penalization of yaw angle deviations was selected to be lower because the main objective was to reduce roll angle deviations. All combinations of the parameters in Table 3.4 were evaluated. The final selection of parameters was based on the obtained RRR score in combination with the achieved Ψ RMS values.

Table 3.4: Parameters to be evaluated for dual control

Parameter	Values
Γ_α	[1,0.1,0.01,0.001]
Γ_β	[0.1,0.01,0.001]
Q	diag(1,1,0.1,0.1), diag(1,1,0.01,0.01)

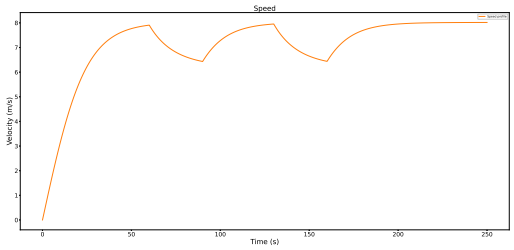
3.3.4 Test cases

The performance of the modified DMRAC control was tested for 6 test cases. They consist of different scenarios where the ship was exposed to internal and/or external disturbances. Information about the characteristics of the disturbance for each case is listed in Table 3.5. In every test case there was wave and wind disturbance, defined in section 3.1.3, affecting the ship. There were also internal disturbances present in every case except the base case. In Test cases 2 and 4 the ship engine suffered a loss in power for two periods of time, more specifically in the range of $[60, 90]$ seconds and $[130, 160]$ seconds, see Fig. 3.6a. Test case 3 and 4 stage a scenario where the fin stabilizers are not movable from the present position at time step 80 and 125 over a period of 5 seconds, see Fig. 3.6b. In Test case 5 is the ship's engine power reduced throughout the whole simulation resulting in a lower speed, see Fig. 3.6c. Test case 6 is the same as the Base case, but the update of the feature weight vector W was not restricted.

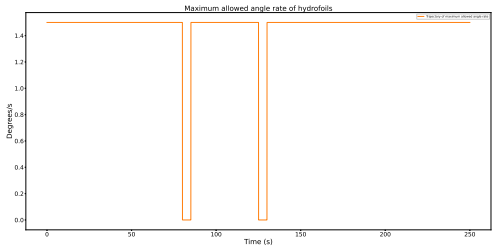
Note that the wave and wind disturbance was generated randomly, thus it was not the exact same disturbance profile for every test case.

Table 3.5: Summary of test cases dimensions

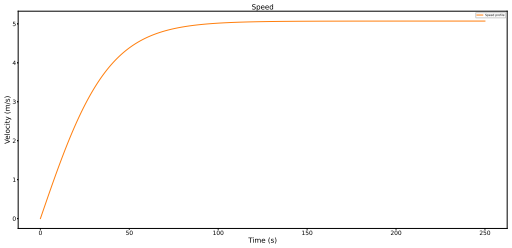
Test case	**External Disturbance**	**Internal Disturbance**
Base	Wave & Wind	None
2	Wave & Wind	Loss in engine power
3	Wave & Wind	Fin failure
4	Wave & Wind	Loss in engine power & Fin failure
5	Wave & Wind	Low engine power
6	Wave & Wind	No W update cancellation



(a) *Loss in engine power profile*



(b) *Fin failure profile*



(c) *Low engine power profile*

Figure 3.6: *Internal Disturbances*

4

Results

This section presents the obtained results of the Test cases and the parameter search. The first subsection of the results covers the single control results and the second presents the dual control results. Performance scores and plots of the system response during simulations of the ship model are also presented for each Test case in Table 3.5.

In order to benchmark the DMRAC performance the simulations also calculated the motion responses of an uncontrolled ship as well as a ship controlled by only the linear control term $u_{pd}(x(t), r(t))$. All three control configurations were exposed to the same disturbance.

4.1 Single control

The results regarding single control are presented in this section, i.e. when DMRAC is only controlling the fin stabilizers.

4.1.1 Parameter evaluation

The 10 of the total 96 settings that achieved the highest performance on the Base case can be seen in Table 4.1. These 10 settings were then further iterated 10 more times and the results of this can be seen in Table 4.2.

Table 4.1: The 10 settings which resulted in the highest RRR score when the ship was exposed to the Base case using active fin stabilizers.

Setting	RB_{size}	MB_{size}	N_θ	$N_{batches}$	μ	Γ	RRR %	RMS
5	500.0	25.0	400.0	3.0	0.01	10.0	99.90	0.1688
39	500.0	50.0	100.0	6.0	0.01	10.0	99.89	0.1804
57	1000.0	25.0	100.0	3.0	0.001	10.0	99.88	0.1878
65	1000.0	25.0	400.0	6.0	0.01	10.0	99.88	0.1913
63	1000.0	25.0	100.0	6.0	0.01	10.0	99.87	0.1953
87	1000.0	50.0	100.0	6.0	0.01	10.0	99.87	0.1978
28	500.0	50.0	200.0	3.0	0.01	10.0	99.87	0.1993
75	1000.0	50.0	100.0	3.0	0.01	10.0	99.86	0.2023
82	1000.0	50.0	200.0	3.0	0.001	10.0	99.86	0.2059
16	500.0	25.0	200.0	6.0	0.01	10.0	99.85	0.2138

Table 4.2: The average RRR score obtained using the settings in Table 4.1 when exposing the ship to 10 unique Base cases using active fin stabilizers.

Iteration	**Setting**									
	5	39	57	65	63	87	28	75	82	16
1	99.89	99.44	99.89	98.29	99.14	99.01	96.98	98.88	99.83	99.80
2	99.92	99.87	86.91	98.92	98.87	99.61	99.82	99.18	99.70	99.14
3	99.89	99.92	85.76	99.86	99.44	99.05	99.87	99.55	86.47	99.89
4	99.79	99.80	98.35	99.78	99.78	99.84	99.79	99.33	98.98	98.54
5	99.27	99.69	99.50	99.71	99.72	99.61	99.86	99.68	99.83	99.82
6	99.87	97.77	83.77	97.68	99.78	99.87	97.52	99.84	99.60	99.86
7	99.74	99.82	91.00	99.80	99.69	99.75	96.82	99.71	86.29	99.91
8	99.48	99.92	99.75	99.89	99.86	99.53	99.90	99.90	97.17	99.05
9	99.51	99.92	86.39	99.90	99.91	99.81	99.93	99.76	99.16	99.68
10	99.78	99.66	99.89	99.90	99.80	99.84	99.85	99.92	99.25	99.80
Average	99.71	99.58	93.12	99.37	99.60	99.59	99.04	99.57	96.63	99.55

Setting 5 achieved the highest average score with a RRR of 99.71 %. Notable is that Setting 57 and 82 had a large variance in RRR score compared to the others which shows the impact regarding choice of learning rate μ.

4.1.2 Performance on Test cases

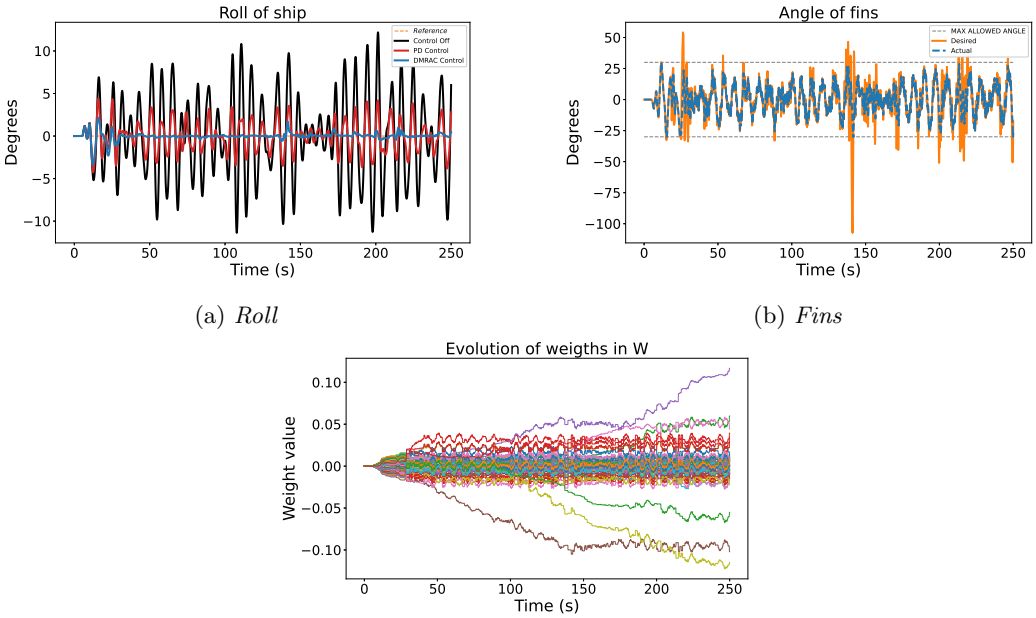
The RRR and RMS results in Table 4.3 was obtained using parameter Setting 5 for each Test case in table 3.5. In Fig. 4.1 - 4.6 the system response for every case can be seen. The plots display the roll response of the ship, how the fin angles of the stabilisation system are changing throughout the simulation and the evolution of the feature weight vector W.

In the plots where the roll trajectories are displayed, the response of the ship without control is in black, the response of the ship controlled with a PD regulator is in red and the response of the DMRAC controlled ship is in blue.

In the plots illustrating the input signal to the foils, the dashed blue line represents the applied control $u(t)$ while the orange line represents the controller's requested control. If they are not equal implies that the requested control signal exceeds the physical constraints of the hydrofoils.

Table 4.3: Effectiveness criteria for each case & roll reduction ratio single control

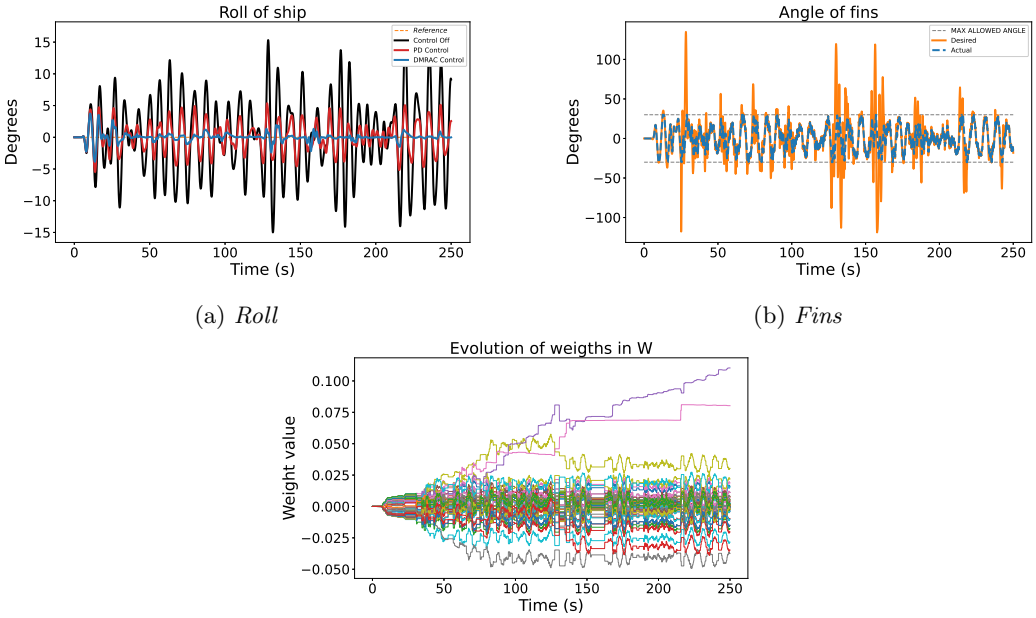
Test case	RRR		RMS		
	DMRAC	PD	DMRAC	PD	UC
Base	99.82 %	88.17 %	0.22	1.89	5.22
2	99.42 %	86.45 %	0.49	2.36	6.41
3	89.53 %	88.00 %	1.72	1.85	5.33
4	94.54 %	85.41 %	1.36	2.23	5.83
5	72.38 %	72.20 %	2.54	2.55	4.84
6	11.58 %	88.91 %	4.62	1.64	4.91



(a) *Roll* (b) *Fins*

(c) W

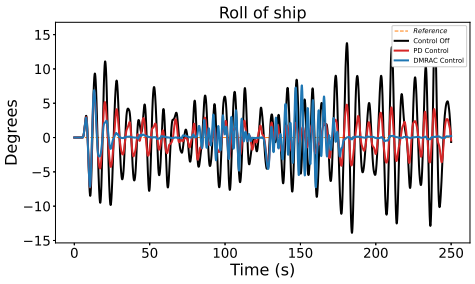
Figure 4.1: *Result Base Case single control*



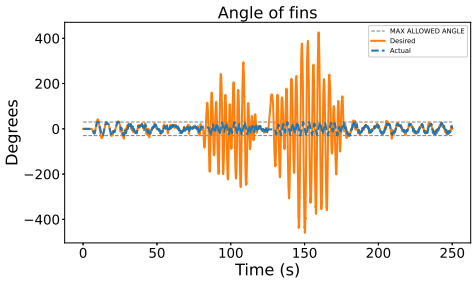
(a) *Roll* (b) *Fins*

(c) W

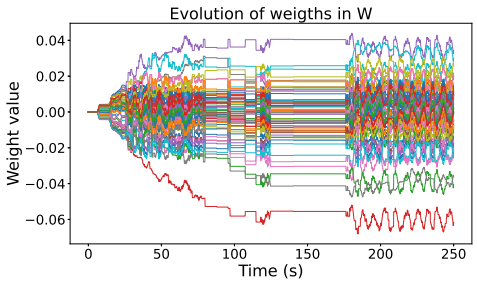
Figure 4.2: *Result Test Case 2 single control*



(a) *Roll*

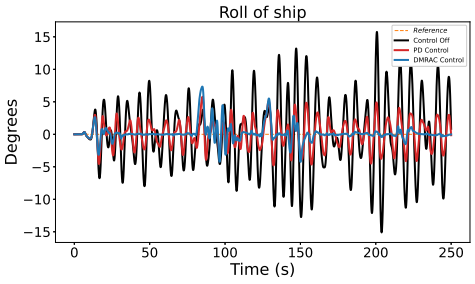


(b) *Fins*

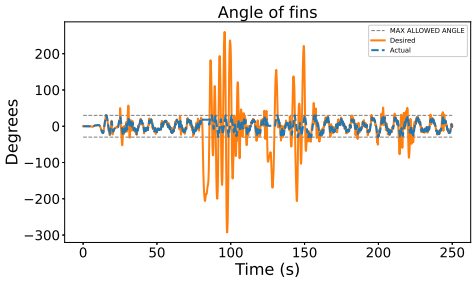


(c) *W*

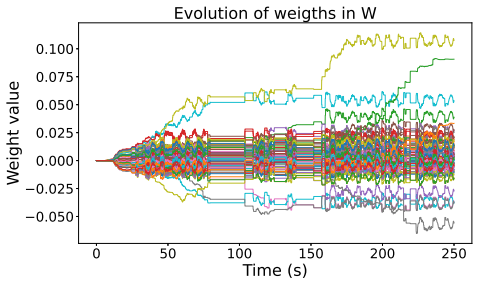
Figure 4.3: *Result Test Case 3 single control*



(a) *Roll*

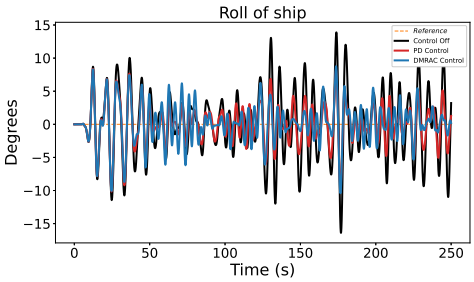


(b) *Fins*

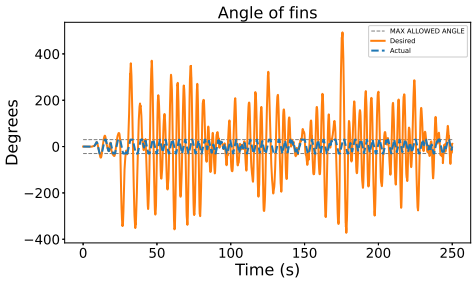


(c) *W*

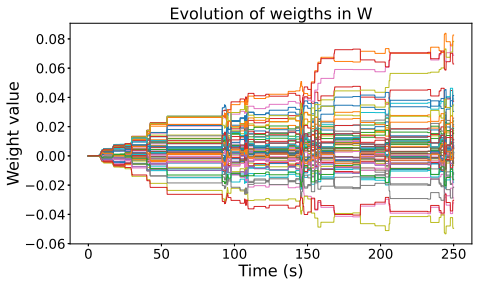
Figure 4.4: *Result Test Case 4 single control*



(a) *Roll*

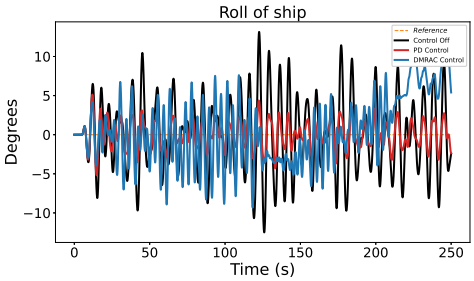


(b) *Fins*

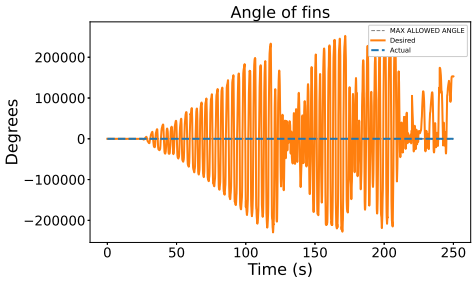


(c) *W*

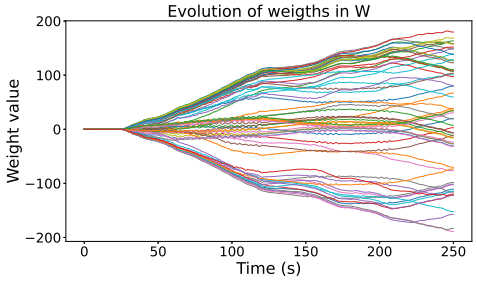
Figure 4.5: *Result Test Case 5 single control*



(a) *Roll*



(b) *Fins*



(c) *W*

Figure 4.6: *Result Test Case 6 single control*

4.2 Dual control

The results shown in this section were received using parameter Setting 5 from Table 4.1, but due to the extended control there was a need for another parameter evaluation to determine a good combination between Γ_α, Γ_β and the penalty matrix Q.

4.2.1 Parameter evaluation

The 10 settings that achieved the highest performance on the Base case can be seen in Table 4.4. These 10 settings were then further iterated 10 more times and the results of this can be seen in Table 4.5 and in Table 4.6.

Setting I, IX and X outperformed the other settings. Because of the high RMS score of Setting I in Iteration 9 it was excluded from the selection. The two remaining settings achieved similar scores with no evident performance difference. The final parameter selection became Setting X due to the slightly lower Ψ RMS score.

Table 4.4: The 10 best settings which resulted in the highest RRR score for all combinations in Table 3.4 exposed to the Base case using active fin stabilizers and rudder

Setting	Γ_α	Γ_β	Q	RRR (%)	Ψ RMS	RMS Φ
I	0.1	0.01	diag(1,1,0.01,0.01)	96.06	6.4915	1.2794
II	1.0	0.01	diag(1,1,0.01,0.01)	95.21	5.7514	1.4110
III	0.01	0.0001	diag(1,1,0.01,0.01)	95.18	5.667	1.4153
IV	0.001	0.0001	diag(1,1,0.01,0.01)	94.65	5.3603	1.4918
V	1.0	0.001	diag(1,1,0.01,0.01)	94.55	6.973	1.5047
VI	0.01	0.001	diag(1,1,0.01,0.01)	93.6	5.3586	1.6304
VII	1.0	0.0001	diag(1,1,0.01,0.01)	92.94	9.1229	1.7128
VIII	0.1	0.001	diag(1,1,0.01,0.01)	92.75	6.6237	1.7362
IX	0.001	0.01	diag(1,1,0.01,0.01)	92.54	6.517	1.7608
X	0.01	0.01	diag(1,1,0.01,0.01)	92.05	7.3852	1.8176

Table 4.5: The average RRR score obtained using the settings in Table 4.4 when exposing the ship to 10 unique Bases cases using active fin stabilizers and rudder.

Iteration	**Setting**									
	I	II	III	IV	V	VI	VII	VIII	IX	X
1	89.92	73.03	89.64	83.69	21.62	91.04	10.93	93.99	90.25	94.42
2	95.94	17.38	91.32	89.70	34.15	90.80	42.63	95.43	92.05	91.43
3	95.04	-97.52	86.66	87.38	90.28	90.24	50.74	92.11	89.82	94.13
4	92.05	87.74	91.37	87.48	82.12	95.26	51.61	92.88	93.86	94.99
5	91.49	85.67	89.85	88.52	88.09	88.69	20.50	95.07	88.26	92.80
6	96.39	74.18	92.36	83.68	80.38	91.80	81.78	90.80	87.29	84.40
7	92.05	91.43	84.74	86.65	90.15	84.98	52.46	91.78	87.74	84.37
8	96.50	96.50	86.64	89.52	76.55	-1.23	93.25	61.11	95.55	91.29
9	89.81	67.07	89.41	89.87	95.00	90.01	89.37	93.29	92.39	93.46
10	91.20	37.81	86.33	88.96	65.28	65.28	83.76	9.35	95.31	88.53
Average	93.04	53.33	88.83	87.55	72.36	78.69	57.70	81.58	91.25	90.98

Table 4.6: The average Ψ RMS score using the settings in Table 4.4 when exposing the ship to 10 unique Bases cases using active fin stabilizers and rudder.

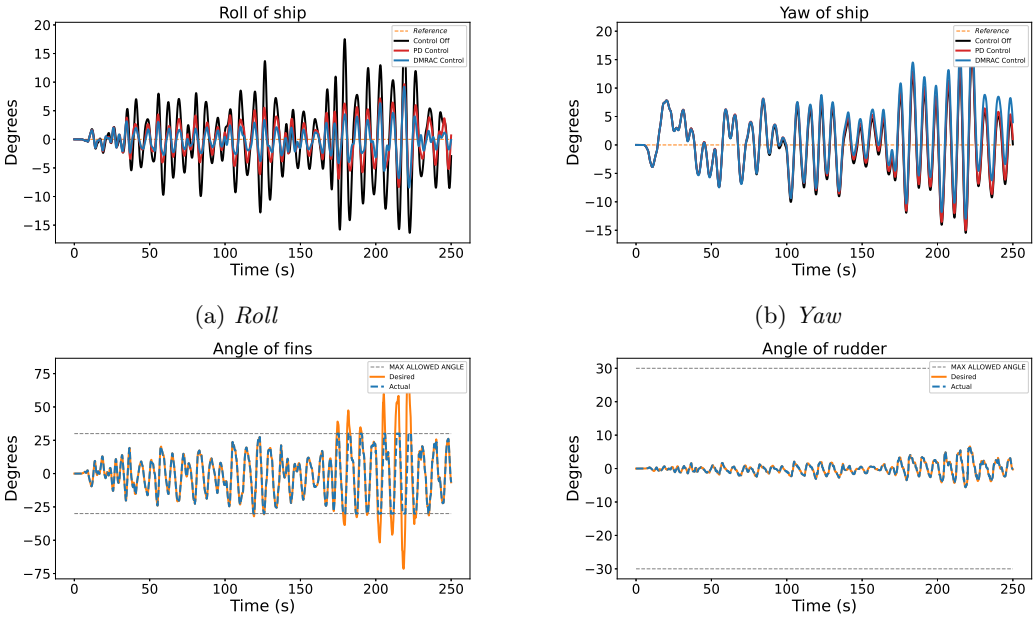
Iteration	**Setting**									
	I	II	III	IV	V	VI	VII	VIII	IX	X
1	5.51	6.29	4.83	6.82	10.08	5.33	13.25	5.18	4.84	5.59
2	6.39	7.42	4.75	4.59	8.77	7.14	9.84	9.59	6.65	5.11
3	6.64	8.20	7.24	6.90	6.45	9.53	9.61	9.63	8.13	6.91
4	7.21	6.24	8.66	9.42	9.53	8.59	10.25	6.28	6.69	8.22
5	5.83	4.93	5.77	6.37	6.20	6.37	6.29	5.74	5.31	5.65
6	4.59	5.64	7.40	9.30	6.84	4.90	4.93	7.22	5.09	6.90
7	5.26	9.21	6.40	5.08	5.92	6.59	6.72	5.04	6.77	4.87
8	5.80	5.27	4.38	5.33	11.08	4.52	4.71	5.53	5.09	4.44
9	12.98	8.45	6.51	6.91	7.34	8.00	5.89	8.56	8.39	6.45
10	5.63	9.99	7.05	7.76	5.98	8.55	6.20	5.65	6.95	6.66
Average	6.58	7.16	6.30	6.85	7.82	6.95	7.77	6.84	6.39	6.08

4.2.2 Performance on Test cases

Setting 5 from Table 3.3 in combination with Setting X from Table 3.4 was used in all of the Test cases. In Fig. 4.7 - 4.12 the system response for each case is shown. The figures also contain the trajectory of the ship's yaw angle and the trajectory of the rudder angle.

Table 4.7: Performance comparison of the different controllers on each Test case using dual control.

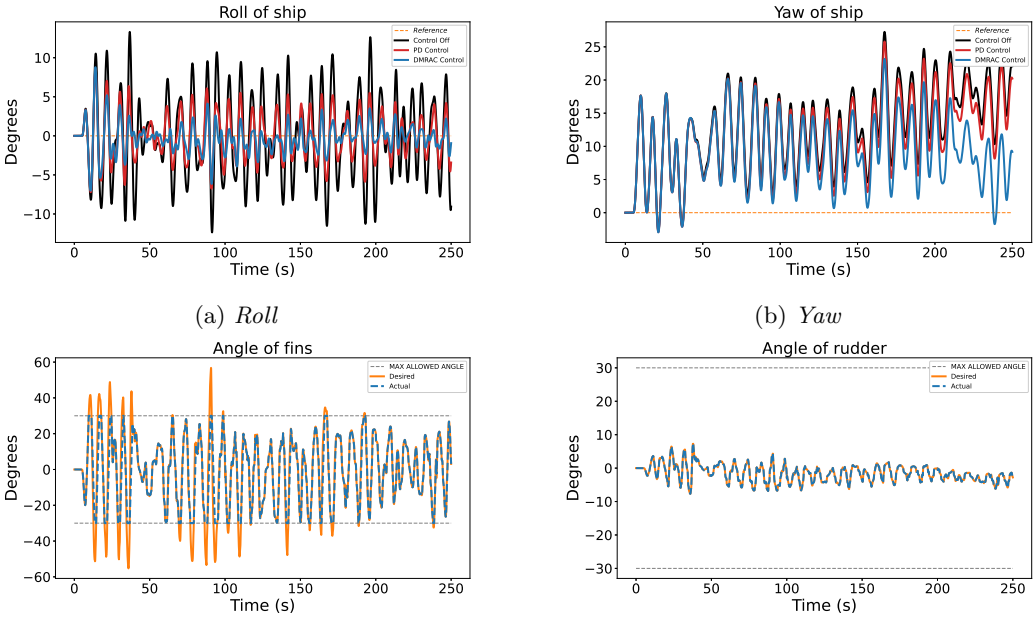
Test case	RRR		RMS (Φ/Ψ)		
	DMRAC	PD	DMRAC	PD	UC
Base	88.02 %	75.62 %	2.25/5.95	3.21/5.68	6.50/5.56
2	91.81 %	72.09 %	1.51/10.70	2.79/13.64	5.28/15.03
3	79.75 %	68.62 %	2.53/7.18	3.15/5.93	5.63/5.75
4	72.22 %	63.13 %	2.92/6.92	3.36/6.11	5.54/6.01
5	43.80 %	49.01 %	3.69/14.26	3.52/6.64	4.93/6.88
6	92.88 %	77.03 %	1.47/5.74	2.65/6.74	5.54/6.43



(a) *Roll* (b) *Yaw*

(c) *Fins* (d) *Rudder*

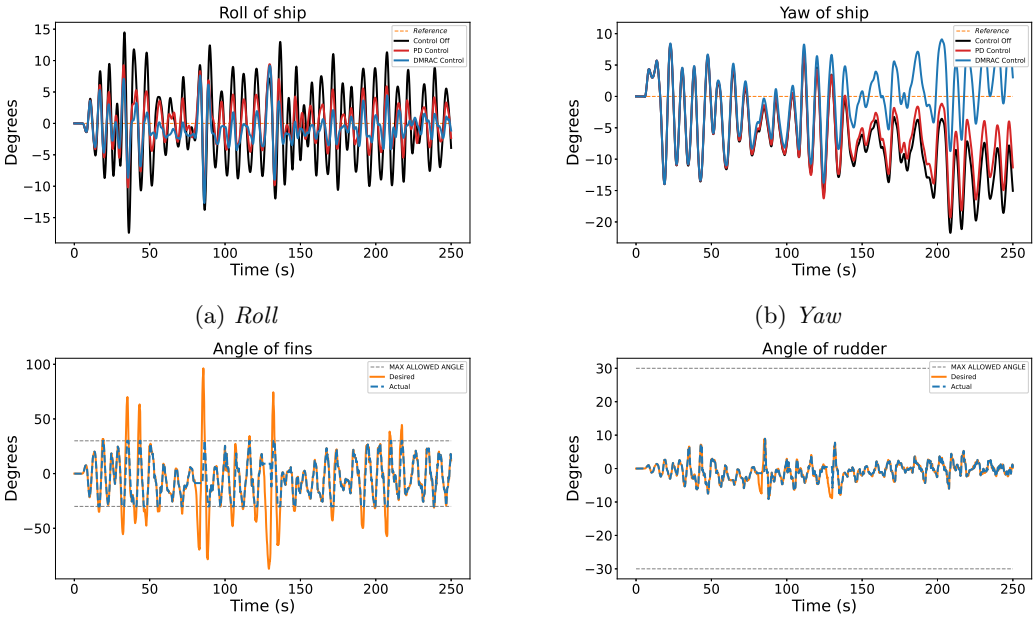
Figure 4.7: *Result Base Case using dual control*



(a) *Roll* (b) *Yaw*

(c) *Fins* (d) *Rudder*

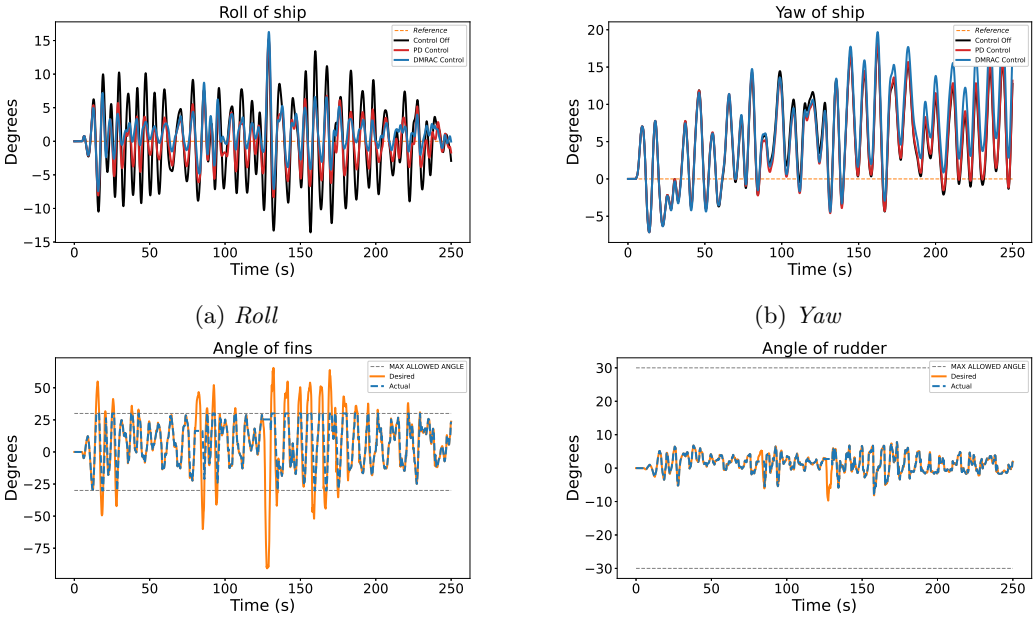
Figure 4.8: *Result Test Case 2 using dual control*



(a) *Roll* (b) *Yaw*

(c) *Fins* (d) *Rudder*

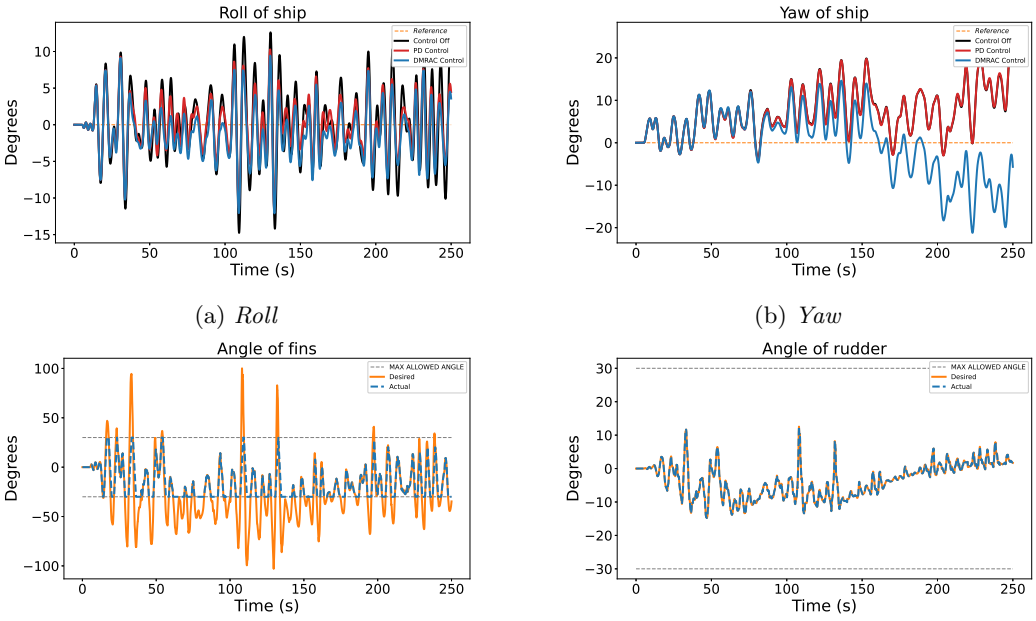
Figure 4.9: *Result Test Case 3 using dual control*



(a) *Roll* (b) *Yaw*

(c) *Fins* (d) *Rudder*

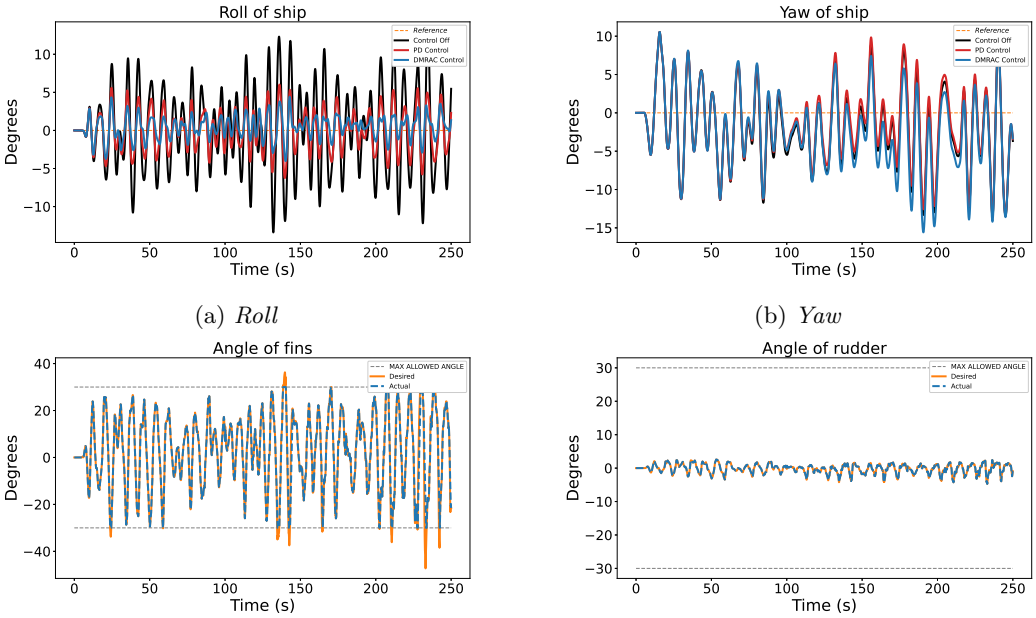
Figure 4.10: *Result Test Case 4 using dual control*



(a) *Roll* (b) *Yaw*

(c) *Fins* (d) *Rudder*

Figure 4.11: *Result Test Case 5 using dual control*



(a) *Roll* (b) *Yaw*

(c) *Fins* (d) *Rudder*

Figure 4.12: *Result Test Case 6 using dual control*

5
Discussion

This section discusses the obtained results from the previous section.

5.1 Roll stabilization

Appointing the modified DMRAC to regulate the active fin stabilizer system to counter arising roll motions, while exposing the ship to a random disturbance, showed initially promising results. The DMRAC performed significantly better than the PD controller, 99.82 % compared to 88.17 % in RRR and 0.22 compared to 1.89 in RMS for the Base case. The control signal was solid throughout the whole simulation except for a few roll angle peaks, but they were still lower than the corresponding peaks for the PD controlled ship.

The addition of internal disturbances introduced a more complex scenario where DMRAC handled the loss in engine power well, it was able to recover and converge towards the optimal control afterward. DMRAC had better performance than the PD control in Test case 2 regarding both RRR and RMS. DMRAC's aggressive response to the speed reduction is seen by the large deviations between the Actual and Desired fin angle in Fig. 4.2b. The activation of the no update rule can be seen in Fig. 4.2c where all weight trajectories remain at the same value over a slim period of time.

The introduction of disturbance in the fin stabilizer system showed a weakness with DMRAC since the PD control was able to handle the fin failure better in the near time after the re-activation of the fins. The desired control signal in Fig. 4.3b was fluctuating over a long period compared to Fig. 4.2b. In Test case 3 the linear PD controller was able to converge faster back to the neighborhood of its normal control. The combined internal disturbance affecting the ship in Test case 4 resulted in a better result than in Test case 3. This is surprising, more disturbance should usually result in poorer performance. This result highlights a problem regarding the validity of a potential result comparison between the Test cases since the same disturbance was not applied in each test case. The overall DMRAC performance is better than the PD - control for Test case 4, but during the periods of internal disturbance it is hard to tell. DMRAC counters roll angle peaks for a few occurrences better, while it is worse in a few others.

As seen in Fig. 4.2, when reducing the ship's speed the performance is reduced substantially for both DMRAC and PD - control, but the former is reducing the roll deviations better throughout the whole simulation. The reduction from 99.82 % to 72.38 % for the similar disturbance demonstrates the performance limitations of the chosen stabilization system where the force generated is in a direct relation with the speed of the ship. A certain ship speed has to be present, or more actuators have to be introduced, to ensure that it is possible to generate enough moment. DMRAC

seems to being able to handle any realistic wave disturbance, but it requires the right conditions in terms of enough potential stabilization moment to have a good performance. This also highlights a drawback with the DMRAC which shows that it can not retain general knowledge about the ships speed i.e. it has to relearn every time the speed changes. Thus, it would be interesting to investigate what happens if the speed of the vessel also would be included in the state vector x. Such that it would affect the formation of the feature vector $\phi(x)$ by being included in the training of the ANN.

The last Test case, seen in Fig. 4.6, shows what happens when the feature weight vector W is allowed to update freely without restrictions. The no weight update rule is crucial to avoid divergence due to DMRAC's incapability to understand the constraints of the hydrofoils. This is a major problem if not taken into consideration. In the future some kind of modified loss function that heavily penalizes the breakage of the hydrofoil constrains needs to be implemented. This could potentially allow the network to learn the specific constraints of the hydrofoils.

For disturbances of normal characteristics the DMRAC deliveres a better stabilization performance than a conventional non adaptive controller, even though the ship dynamics and stabilisation system is unknown. It adapts to changes over time quiet well, but it is not outperforming the PD control when there is disruptions in the stabilisation system.

5.2 Roll & yaw stabilization

Letting DMRAC control the rudder in harmony with the active fin stabilizer system gave a roll reduction ratio of 88.02 % which was better than the 75.62 % of PD control, but the performance was a lot worse compared to single control which had 99.82 % in RRR. When studying the yaw response of the different control configurations there are notable performance distinctions. For Test case 2 and 3 is DMRAC able to track the reference signal better than PD, as can be seen in Fig. 4.8b and Fig. 4.9b. However, in Test Case 4 and 6 is the performance opposite, the yaw trajectory for PD is tracking the reference better, see Fig. 4.10b and Fig. 4.12b. Thus, any distinct conclusion could not be made about whether DMRAC or PD was superior.

The reduction in RRR compared to the single control was overall lower for every Test case except Test case 6. For that case there was a huge increase in RRR score, a rise from 13 % to 92 %, which not followed the trend. This occurrence was probably only a fluke where the dual control got lucky in the initial phase of the simulation, avoiding an exploding control signal as mentioned in section 3.2.5. Since the controller is partly a black box it is not possible to exactly pinpoint what causes the control to diverge.

In general, trying to stabilize around zero for both the roll and yaw axis seems to be problematic since a decrease in yaw angle often leads to an increase in roll angle due to the nature of the plant dynamics. This is also one of the main reasons the RRR performance of the proposed controller decreased when the control of

the rudder was introduced. It is not possible to adjust yaw without affecting the roll. It is also the reason why the performance increased when the penalty on the yaw states was decreased since that leads to a lower activity on the rudder control. To remedy this, some type of optimal reference generation probably needs to be introduced. For example, if the boat wants to turn 5 degrees, which roll angle should the controller then stabilize around to make the turn motion optimal?

6

Conclusion

The purpose of this thesis was to enhance an already existing controller with a non linear element whose target was to cover modeling errors and adapt to disturbances affecting the ship. Despite not knowing anything about the naval vessel or the stabilizing actuators, the proposed controller showed promising results when only stabilizing around the roll axis. The proposed controller showed both adaptability, by being able to increase the roll stabilization when unmodeled disturbances were introduced, as well as identifying the dynamic characteristics of the ship. However, when appointing the proposed controller to track two references using both fins and rudder, the performance was reduced. This reduction was probably not a shortcoming of the algorithm itself, but rather a consequence of how the reference generation was made in combination with the lack of potential of the actuators. The results also indicate that the controller can't learn the physical constraints of the actuators which if not dealt with can lead to poor performance.

The developed ANN controller is powerful but fragile. The results indicate that this controller could be taken off and applied to a similar ship in terms of dimension, displacement and stabilizing system and without modification be able to stabilize that ship. It would be interesting in future work to examine the performance of the proposed controller on another ship model, and also implement the proposed controller on an actual ship.

References

[1] E. F. Camacho and C. Bordons, "Nonlinear model predictive control," *Model Predictive control*. London: Springer London, 2007, pp. 249–288, ISBN: 978-0-85729-398-5. DOI: 10.1007/978-0-85729-398-5_9. [Online]. Available: https://doi.org/10.1007/978-0-85729-398-5_9.

[2] J.-H. Kim and Y.-H. Kim, Motion control of a cruise ship by using active stabilizing fins, *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment* **225**, no. 4 2011, 311–324, 2011. DOI: 10.1177/1475090211421268. [Online]. Available: https://doi.org/10.1177/1475090211421268.

[3] "Container ship," NOAA's National Ocean Service, 2011. [Online]. Available: https://www.flickr.com/photos/40322276@N04/5369581593.

[4] E. Tupper, *introduction to Naval Architecture*. Butterwoth Heinemann, 1996, ISBN: 0-7506-2529-5.

[5] "Bilge keel on a steel vessel," Dj245 at en.wikipedia, 2007. [Online]. Available: https://commons.wikimedia.org/wiki/File:Bilgekeel.jpg.

[6] K. Young-Bok, A study on an anti-rolling system design of a ship with the flaps, *KSME International Journal* **18** 2004, 1312–1318, 2004. DOI: 10.1007/BF02984245.

[7] P. Yu, M. Yan, J. Zhang, Z. Liu, and G. Li, "A robust finite time-based anti-pitching control method for a high-speed multihull," *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 3293–3298. DOI: 10.23919/CCC50068.2020.9189277.

[8] T. W. Treakle, "A time-domain numerical study of passive and active anti-roll tanks to reduce ship motions," 1998, p. 50.

[9] K. Kula, An overview of roll stabilizers and systems for their control, *International Journal on Marine Navigation and Saftey of Sea Transportation* **9**, no. 3 2015, 409–410, 2015. DOI: 10.12716/1001.09.03.14.

[10] Z.-g. Qi, H.-z. Jin, W.-y. Liu, and Y. Xu, "Research on active fin stabilizer at low speed and its application to ship roll stabilization," *OCEANS 2014 - TAIPEI*, 2014, pp. 1–6. DOI: 10.1109/OCEANS-TAIPEI.2014.6964511.

[11] N. Patil, A. Dubey, and V. Subramanian, Fin based active control for ship roll motion stabilization, *MATEC Web of Conferences* **272** Jan. 2019, 01026, Jan. 2019. DOI: 10.1051/matecconf/201927201026.

[12] T. Sunil, M. Jayachandra, and R. Prasad, "Control of ship's roll using active fin stabilizers and neural network controller," *2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, 2016, pp. 516–519. DOI: 10.1109/ICACCCT.2016.7831693.

[13] W. Zhao, C. Guo, H. Jin, and D. Yu, "Optimal design on ship synthetic stabilizer system based on fin stabilizer and anti-rolling tank," *Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE)*, 2011, pp. 1437–1440. DOI: 10.1109/TMEE.2011.6199477.

[14] N. T. Nguyen, "Model-reference adaptive control," *Model-Reference Adaptive Control: A Primer*. Cham: Springer International Publishing, 2018, pp. 83–123, ISBN: 978-3-319-56393-0. DOI: 10.1007/978-3-319-56393-0_5. [Online]. Available: https://doi.org/10.1007/978-3-319-56393-0_5.

[15] H. Müller and N. Villegas, Runtime evolution of highly dynamic software Dec. 2014, 229–264, Dec. 2014. DOI: 10.1007/978-3-642-45398-4_8.

[16] S. Kamalasadan and A. Ghandakly, "A neural network based intelligent model reference adaptive controller," *2004 IEEE International Conference onComputational Intelligence for Measurement Systems and Applications, 2004. CIMSA.*, 2004, pp. 174–179. DOI: 10.1109/CIMSA.2004.1397257.

[17] D. Noble and S. Bhandari, "Neural network based nonlinear model reference adaptive controller for an unmanned aerial vehicle," *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 94–103. DOI: 10.1109/ICUAS.2017.7991337.

[18] G. Joshi and G. Chowdhary, "Deep model reference adaptive control," *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 4601–4608. DOI: 10.1109/CDC40024.2019.9029173.

[19] WÄRTSILÄ, "Stability," *Encyclopedia of Marine and Energy Technology*, 2021. [Online]. Available: https://www.wartsila.com/encyclopedia/term/stability.

[20] SNAME, "Nomenclature for treating the motion of a submerged body through a fluid," *Technical and Research Bullentin*, 1950. [Online]. Available: https://www.itk.ntnu.no/fag/TTK4190/papers/Sname%5C%201950.PDF.

[21] W. Gierusz, "Modelling the dynamics of ships with different propulsion systems for control purpose," *POLISH MARITIME RESEARCH*, vol. 23, 2016, pp. 31–36. [Online]. Available: https://sciendo.com/pdf/10.1515/pomr-2016-0005.

[22] T. I. Fossen and T. Perez, *Marine Systems Simulator (MSS)*, 2021. [Online]. Available: https://github.com/cybergalactic/MSS.

[23] K. J. Åström and B. Wittenmark, *Adaptive control*. Dover Publications, 1994, ISBN: 0486462781.

[24] M. Sahay, "Neural networks and the universal approximation theorem," 2020. [Online]. Available: https://towardsdatascience.com/neural-networks-and-the-universal-approximation-theorem-8a389a33d30a.

[25] T. Sergios, *Machine Learning: A Bayesian and Optimization Perspective*. Academic Pr, 2020, ISBN: 0128188030.

[26] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2014. DOI: 10.48550/ARXIV.1412.6980. [Online]. Available: https://arxiv.org/abs/1412.6980.

[27] J. Sola and J. Sevilla, Importance of input data normalization for the application of neural networks to complex industrial problems, *IEEE Transactions on Nuclear Science* **44**, no. 3 1997, 1464–1468, 1997. DOI: 10.1109/23.589532.

[28] L. Liang and Y. Wen, "Feed-forward disturbance compensation model predictive control for rudder roll stabilization," *2017 36th Chinese Control Conference (CCC)*, 2017, pp. 4790–4795. DOI: 10.23919/ChiCC.2017.8028110.

[29] H. Ghassemi, F. Dadmarzi, P. Ghadimi, and B. Ommani, Neural network-pid controller for roll fin stabilizer, *Polish Maritime Research* **17** Jan. 2010, Jan. 2010. DOI: 10.2478/v10012-010-0014-3.

[30] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley, 2021, ISBN: 1119575052.

[31] "Waves," Australian Government Bureau of Meteorology. [Online]. Available: http://www.bom.gov.au/marine/knowledge-centre/reference/waves.shtml.

[32] F. Alarçïn, Internal model control using neural network for ship roll stabilization, *Journal of Marine Science and Technology* **15** 2007, article 9, 2007, ISSN: 2. [Online]. Available: https://jmstt.ntou.edu.tw/journal/vol15/iss2/9.

[33] M. Paul, "Regularization," 2018. [Online]. Available: https://cmci.colorado.edu/classes/INFO-4604/files/slides-6_regularization.pdf.

[34] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons Ltd, 2011, ISBN: 978-1-119-99149-6.

[35] O. Faltinsen, *Sea Loads on Ships and Offshore Structures*, ser. Cambridge Ocean Technology Series. Cambridge University Press, 1990, ISBN: 9780521372855. [Online]. Available: https://books.google.se/books?id=MTVgQgAACAAJ.

[38] K. Hanjalic *et al.*, A robust near-wall elliptic-relaxation eddy-viscosity turbulence model for cfd, *Int. J. Heat Fluid Flow* **25** 2004, 1047–1051, 2004.

[39] *Matlab manual, Ordinary differential equations*, version 7.8, Mathworks, 2008. [Online]. Available: http://www.mathworks.com/access/helpdesk/help/techdoc/ref/ode45.html.

[40] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press, 2004, ISBN: 0-521-55655-4.

[41] W. Luo, B. Hu, and T. Li, Neural network based fin control for ship roll stabilization with guaranteed robustness, *Neurocomputing* **230** 2017, 210–218, 2017, ISSN: 0925-2312. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231216314990.

[42] M. Perez Tristan; Blanke, "Mathematical ship modeling for control applications," 2002. [Online]. Available: https://backend.orbit.dtu.dk/ws/portalfiles/portal/137166308/Mathematical_Ship_Modeling_for_Control_Applications_Perez_Blanke.pdf.

[43] M. Ertogan, S. Ertugrul, and M. Taylan, Application of particle swarm optimized pdd^2 control for ship roll motion with active fins, *IEEE/ASME Transactions on Mechatronics* **21**, no. 2 2016, 1004–1014, 2016. DOI: 10.1109/TMECH.2015.2479401.

[44] A. H. Marzouk Osama A.; Nayfeh, "Control of ship roll using passive and active anti-roll tanks," *Ocean Engineering*, vol. 36, 2009, pp. 661–671. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801809000675.

[45] Z. Öreg, H.-S. Shin, and A. Tsourdos, "Model identification adaptive control - implementation case studies for a high manoeuvrability aircraft," *2019 27th Mediterranean Conference on Control and Automation (MED)*, 2019, pp. 559–564. DOI: 10.1109/MED.2019.8798513.

[46] "A computational tool for the design of ride control systems for fast planing vessels," *International Shipbulding Progress*. [Online]. Available: https://content.iospress.com/articles/international-shipbuilding-progress/isp74.

[47] R. Palnitkar and J. Cannady, "A review of adaptive neural networks," *IEEE SoutheastCon, 2004. Proceedings.*, 2004, pp. 38–47. DOI: 10.1109/SECON.2004.1287896.

[48] K. B. Ariyur and M. Krstić, *Real-time optimization by extremum seeking control*, 2003. [Online]. Available: https://onlinelibrary.wiley.com/doi/book/10.1002/0471669784.

[49] I. DOURATSOS and J. GOMM, Neural network based model reference adaptive control for processes with time delay, *International Journal of Information & Systems Sciences* **3** Jan. 2007, Jan. 2007.

[50] ScienceDirect, "Stability, inertia, and robust stability," *Lyapunov Stability Theory*. [Online]. Available: https://www.itk.ntnu.no/fag/TTK4190/papers/Sname%5C%201950.PDF.

[51] I. M. Mareels, B. D. Anderson, R. R. Bitmead, M. Bodson, and S. S. Sastry, Revisiting the mit rule for adaptive control, *IFAC Proceedings Volumes* **20**, no. 2 1987, 161–166, 1987, 2nd IFAC Workshop on Adaptive Systems in Control and Signal Processing 1986, Lund, Sweden, 30 June-2 July 1986, ISSN: 1474-6670. DOI: 10.1016/S1474-6670(17)55954-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667017559546.

[52] SKF, "Compact and proven design for roll reduction," *Retractable fin stabilizers type Z*, 2021. [Online]. Available: https://www.skf.com/uk/industries/marine/stabilizing-systems-and-steering-gear/retractable-fin-stabilizer-type-z.

[53] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization 2014, 2014. DOI: 10.48550/ARXIV.1412.6980. [Online]. Available: https://arxiv.org/abs/1412.6980.

[54] K. S. Narendra and R. Shorten, Hurwitz stability of metzler matrices, *IEEE Transactions on Automatic Control* **55**, no. 6 2010, 1484–1487, 2010. DOI: 10.1109/TAC.2010.2045694.

[55] I. M. Mareels, B. D. Anderson, R. R. Bitmead, M. Bodson, and S. S. Sastry, Revisiting the mit rule for adaptive control, *IFAC Proceedings Volumes* **20**, no. 2 1987, 161–166, 1987, 2nd IFAC Workshop on Adaptive Systems in Control and Signal Processing 1986, Lund, Sweden, 30 June-2 July 1986, ISSN: 1474-6670. DOI: https://doi.org/10.1016/S1474-6670(17)55954-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667017559546.

[56] Austal, "Armidale class patrol boat (royal australian navy)." [Online]. Available: https://www.austal.com/ships/armidale-class-patrol-boat-royal-australian-navy.

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS