DEGREE PROJECT IN ELECTRICAL ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2018

Robust Model Predictive Control for Marine Vessels

ALLAN ANDRE DO NASCIMENTO

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Robust Model Predictive Control for Marine Vessels

ALLAN ANDRE DO NASCIMENTO

Master Thesis in Control Theory
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden 2018

KTH School of Electrical Engineering and
Computer Science - Automatic Control Department
SE-100 44 Stockholm
SWEDEN

Abstract

This master thesis studies the implementation of a Robust MPC controller in marine vessels on different tasks. A tube based MPC is designed based on system linearization around the target point guaranteeing local input to state stability of the respective linearized version of the original nonlinear system. The method is then applied to three different tasks: Dynamic positioning on which recursive feasibility of the nominal MPC is also guaranteed, Speed-Heading control and trajectory tracking with the Line of sight algorithm. Numerical simulation is then provided to show technique's effectiveness.

Keywords: Robust control, Tube MPC, marine vessel control

Sammanfattning

Detta examensarbete studerar design och implementering av en robust modellprediktiv regulator (MPC) för marina fartyg. En tub-baserad MPC är designad baserad på linjärisering av systemdynamiken runt en målpunkt, vilket garanterar local insignal-till-tillstånds stabilitet av det linjäriserade systemet. Metoden är sedan applicerad på tre olika uppgifter: dynamisk positionering, för vilken vi även kan garantera rekursiv lösbarhet för den nominella regulatorn; riktningsstyrning; och banfötljning med en siktlinje-algoritm. Numeriska simuleringsstudier bekräftar metodens effektivitet.

Keywords: Robust kontroll, Tube MPC, fartygskontroll

Acknowledgements

First and foremost I would like to thank God for giving me the strength to endure and the chance to be finishing this program.

I also would like to deeply thank ABB Corporate Research for the opportunity of working once again with a very interesting topic and to be in contact with fantastic people, both in an intellectual so as in a personal level. This work would definitely not be possible without the help of my supervisors Winston Garcia Gabin, Hamid Feyzmahdavian and Mikael Johansson, whose great support and guidance had helped me a lot to accomplish this milestone in my life.

This journey would never be possible without the support and love from all my family. In particular, I would like to thank my cousin Bruno, for all his support and encouragement throughout my life.

Most importantly, none of this would be possible without my fantastic mom Maria de Fatima and my fantastic aunt and "second mom" Regina, to whom I will aways be grateful for all the love and support during hard times. The same applies to my fiancee Thais Barbosa who has been fundamental through her love, patience and kindness towards me. I certainly must also thank my sister Fernanda, who has always encouraged and been there for me, so as my father Francisco.

Allan Andre do Nascimento,
Västerås, November 14, 2018

Contents

List of Figures

Chapter 1

Introduction

Ships have played a very important role throughout history. Since old handcrafted ships, to modern vessels, the naval industry have changed its scope a lot since its birth and together with it, the type of technology used in it. Nowadays, following other industries trend, the marine industry so as the academic community have been experiencing a growing interest in the development of automated ships [3], [4]. Such interest does not come off as surprising once the potential to reduce marine accidents, which nowadays can be more than 90% of the times linked to human error, is enormous [5]. Not only accidents, but costs due to human mistakes which yearly are believed to cost the marine industry more than \$541 million dollars [5] could be heavily mitigated by the development of such technology. Among the different milestones still to be accomplished in this venture lies the development of suitable controllers capable of addressing all the desired behaviors by a vessel.

1.1 Background

One of the first attempts to apply control principles in vessels dates back to 1922 when the first PID controller's formal mathematics for ship implementation [6] was developed. Since then, for the most different tasks, lots of different techniques have been studied and applied towards the goal of fully automated ships. For dynamic positioning for example, which can be defined as the vessel's capacity of keeping its position while rotating to a pre-defined attitude [3], LQG has been successfully implemented by [7]. Also using a linear ship model, H_∞ has been proposed in [8]. The inherent vessel nonlinearities, and thus nonlinear mathematical models describing it, made researchers also implement nonlinear control models such as backstepping in [9]. Traditional and adaptive sliding mode controllers were used in [10] and [11] respectively with the goal of trajectory tracking, while adaptive feedback linearization for automatic ship steering was proposed by [12]. Artificial intelligence techniques also have been abundantly applied in this area. Artificial neural networks and fuzzy logic for instance, have been applied to rudder control in [13], showing an even smoother result when compared to PID controllers applied in the same endeavor. Neural networks are also used in [14], to estimate unknown vessel dynamics, while an adaptive neural network is adopted for vessel control, guaranteeing steady state control performances. Different control techniques with observers are also found on the literature as in [15] on which a PD controller is used in tandem with a passive observer in order to develop a controller for dynamic positioning [16] or in [17] on which an extended state observer was used together with a nonlinear heading controller to be implemented in vessels sailing in restricted waters.

1.2 Motivation

Although the techniques above present several benefits, its main problem is the lack of explicit input constraint accountability by the controller. A method allowing to address such issue during the problem formulation and thus addressing constraints directly is Model Predictive Control (MPC). MPC is an established and efficient technique for constrained multivariable control [18], having shown high adaptability to different versions of the "core technique" such as Hybrid MPC and Robust MPC for instance [19], in order to incorporate different desired behaviors. MPC has been implemented on numerous studies, ranging from MPC for dynamic positioning in [20] on which thrust allocation is also included in the MPC formulation to produce final near optimal constrained controller output, to MPC for waypoint tracking including Line of sight algorithm [2] on which the MPC formulation also included the cross tracking error of the vessel with respect to the line connecting waypoints to optimize this variable while producing constrained rudder action. MPC has also successfully been mixed with AI methods as in [21] on which a single-layer-recurrent neural network is used along with the MPC formulation to solve the QP problem generated with the goal of trajectory tracking by an under-actuated vessel.

1.3 Master Thesis Objectives

Handling constraints albeit important is not the only major concern when designing vessel controllers. It is known that system uncertainties and disturbance from wind, waves and ocean curret s can deteriorate a lot control performance [4]. Having in mind the necessity to fulfill constraints while handling disturbances, it is then reasonable to consider the usage of robust MPC controllers. Several different robust MPC control techniques are available and can be a suitable approach to avoid infeasibility problem when relying only on the inherent robustness of the traditional MPC formulation [22]. Min-Max MPC for instance, which is a method considering the worst case scenario disturbance, has been proposed in [23] and [24]. Nonetheless, some drawbacks are well known outcomes of this technique implementation such as intractable computational time and decreased performance due to method inherent conservativeness [22]. Tube based MPC is another robust MPC approach which has, by now, been successfully implemented on different situations and systems such as in [25], [18] and [26]. Given the good disturbance rejection capacity and reasonable computational time, when compared to the previous cited method, the main objective of this master thesis is the implementation of Tube based MPC for disturbance rejection in marine vessels.

1.4 Master Thesis Contributions

The contribution of this master thesis is to handle external disturbances in marine vessels, while guaranteeing local input to state stability (near the target point) of the equivalent linear system derived by linearization of the original vessel nonlinear dynamics around the target point. Recursive feasibility, depending on the goal task, is also guaranteed on this robust MPC method. Technique implementation followed by numerical simulation on three different tasks, namely: Dynamic positioning, speed-heading control and waypoint tracking by means of the line of sight algorithm [2] is done in order to evaluate method's suitability and performance.

1.5 Master Thesis Organization

This thesis is organized in five chapters as follows:

- Chapter 1 - Introduction - In this chapter we briefly go over a large number of different approaches implemented for the purpose of marine vessel control. A brief evaluation of desired characteristics missing in the previous techniques takes place, which encourages the usage of Model Predictive Control (MPC). Several MPC techniques are listed and it is also argued how this technique may help objectives to be reached. Finally the solution built is approached in a more specific manner when contributions of this work are listed.

- Chapter 2 - Literature Review - In this chapter we will start by reviewing traditional vessel models and then cover some disturbance and uncertainty modeling found in the literature. Following this first more general presentation, relevant robust model predictive control techniques will be studied, being followed by a discussion over advantages and disadvantages of each technique in order to motivate the best control technique for our purposes.

- Chapter 3 - Tube MPC theory and application - In this chapter we will start by covering the mathematical model of ship dynamics used in this work. The general Tube MPC theoretical explanation will follow. This will serve as the backbone for technique implementation to three different tasks, namely: Dynamic Positioning, Speed Heading Control and Line of sight, for which an extra section will be presented in order to introduce the reader to the line of sight path planning algorithm.

- Chapter 4 - Simulation Results - This chapter will show the simulation results for the technique implemented to the three tasks mentioned. Each task will be subject to different types of disturbance, including wind and ocean currents of different speeds and magnitudes. In the end a technique discussion will take place to highlight important points and observations gathered during the development of this work

- Chapter 5 - Conclusion and Future Work - In this final chapter a summary of the whole work will be presented with some extra relevant comments. This chapter will then be finished with a discussion of possible future research directions this work has pointed out.

Chapter 2

Literature Review

In this chapter we are going to cover briefly the ship model used, go through different disturbance model approaches and then move to a more thorough review of the control techniques available for our problem solution.

2.1 System Model

The general ship model found in the literature is usually described by a highly nonlinear system of equations, depicting the 6 degrees of freedom motion of the ship in the world, which describes not only the ship rigid body dynamics but also takes into account the effect of waves, ocean currents, wind and thrust provided by the propellers [1]. Usually, the very core of ship dynamical equations are derived by rigid body dynamics, whereas the external forces and moments acting on a marine craft can be derived in different manners [1], under different assumptions, as described below:

- Maneuvering Theory - Assumes that the ship moves at a constant speed in calm waters and that the hydrodynamic coefficients are also considered to be independent of wave frequencies. In this approach, added mass and damping terms (to be explained later) are approximated by constant hydrodynamic derivatives. Such technique may yield linear or nonlinear vessel models depending on the complexity required [1].
- Seakeeping theory - In this case, considering also that the ship is moving at constant speed, wave forces and hydrodynamic coefficients are calculated as a function of the wave excitation frequency, while taking into account ship mass distribution and hull geometry [1]. Again, in this case linear and nonlinear models may be derived [1].

Although both methods aim the same goal of modeling a ship, one of the main differences among these methods is that they express the desired equations on different referentials [27]. In the former, equations are written in the body fixed frame, whereas in the latter, a frame is attached to a virtual vessel emulating the average motion of the real vessel [27]. Although these are standard methods, for the purpose of simulation and control in time domain, a more suitable model can be chosen by adopting state space equations written with a "composition of frames", where part of the equations are expressed in the body frame and a part expressed in the inertial frame, as is done in robotics and aeronautics [27]. Such method can be named "Fossen's Robot-Like Vectorial Model for Marine Craft" [1] and will be the model chosen in this work following closely the approach by [28].

2.2 Disturbance and Uncertainty Modeling

One of the challenges when dealing with disturbance and uncertainty modeling is, to choose a model precise enough to address our needs but simple enough so that the final model is not unnecessarily overcomplicated resulting in waste of computational resources. Skjetne, Smogeli and Fossen [28] for instance, have considered the disturbance on a vector of external forces and moment (the ship model used is a 3-DOF one). Disturbances are summed to the equation, being then time varying additive disturbances (they are all lumped into one term $w(t)$). In [29], Fossen and Strand also model external disturbances as additive disturbances in a ship model similar to the previous paper (low frequency/surface motion ship model). It is considered that a slow varying given force will be acting on a given position of the ship and without any sensors ship is claimed to control itself to required position.

In [30] disturbances are treated as time varying disturbances. The authors also consider uncertainties in the model such as uncertainties in the Coriolis and Damping terms, uncertainty in the gravitational forces, so as intrinsic model uncertainties such as neglected dynamics and measurement errors by the sensors. All these terms are lumped into an additive disturbance term which is then divided according to the nature of the uncertainty. The sum of Coriolis and Damping uncertainties for instance are bounded by a constant times the sum of the vector's norm 2 and its squared norm 2. On the other hand it is considered that environmental disturbances, unmodeled dynamics and gravitational uncertainty can all be bounded together by a given constant.

In line with previous approaches, Qu, Xiao, Fu and Yuan consider in [31] uncertainty in ship's inertia, Coriolis and Damping matrices together with external disturbances. These uncertainties are lumped into an additive term and bounds are considered for ship's uncertainties. Norm of Inertia uncertainty is bounded by the norm of inertia's nominal value, norm of Coriolis uncertainty is bounded by the norm of Coriolis nominal value, and the same approach is used for the damping matrix. No explicit norm is considered for external disturbances.

After reviewing disturbance and uncertainty models for vessels, it seems reasonable to follow the approach that both uncertainty and disturbances will be considered as a lumped, additive and time varying bounded vector entering the system, similar to what has been considered in [31].

Having finished the system literature review, in the next section, we will proceed to review the control literature and evaluate which would be the most appealing method to tackle the problem.

2.3 Control Techniques

In order to solve the problem approached in this master thesis, several different control techniques can be used. In this section, some of the most relevant techniques given the nature of the problem, will be enumerated and reviewed. Examples of its application will be given and finally a decision on which is the most suitable one will be reached.

2.3.1 Min-Max MPC

Min-Max Model Predictive Control (MMMPC) is a method that computes the optimal control signal by minimizing the worst case cost. The worst case scenario cost is obtained by maximizing the current considered cost function with respect to disturbances and uncertainties that may appear on the system modeled [24]. The Min-Max robust controller approach was originally conceived by Witsenhausen [32] and later was applied on the scope of model predictive controllers by Campos and Morari [32]. Scokaert and Mayne [32] have then adapted this approach for systems suffering from additive disturbances.

It has been claimed that the original technique as proposed by Campos and Morari, in the scope of MMMPC suffered from a huge computational burden [23] , which has prompted researchers to find ways around it. One usual proposed approach is the calculation of an upper bound by means of a chosen performance index (in general including LMI calculations) [23]. A related method which is claimed to be even more efficient has been proposed, for instance, by Ramirez, Alamo and Camacho [23] who have made use of matrix relations in order to find an estimate of the usual upper-bound found by means of LMI's. Even though such approach has shown interesting results, it was heavily reliant on the average of matrix elements for a good performance in terms of accuracy and computational speed when compared to the original LMI approach [23]. Another approach proposed by the same authors and Gruber [24] presented a similar approach, by estimating an upper bound of the worst case scenario using matrix decomposition. Now, authors were able to prove input to state practical stability of such technique and in order to prove its effectiveness, this MMMPC technique was tested in a nonlinear continuous stirred tank reactor, which was modeled as a linear system (by means of system identification) with additive bounded disturbance. Their approach has shown very good results when compared to the standard MPC being simulated under the same conditions, but a downside of such approach was that, even though it has managed to reduce a lot the computational burden when compared to the traditional Min-Max MPC, it still presented computational times 10 to around 20 times higher than the traditional MPC using very short prediction horizons. In case a system needs larger prediction horizons in order to attain the terminal set, such approach can still lead to high computational times [24].

Another relevant application of Min-Max MPC has been suggested by Liu and collaborators [33] who have proposed a Robust Self triggered Min-Max MPC for discrete time nonlinear systems. Researchers claim that the proposed algorithm is recursively feasible and the closed loop system is input to state practically stable under disturbance and system uncertainty. By the end of the paper they study the approach proposed in a small nonlinear system of two states. Although the suggested method shows satisfactory performance, managing even to reduce the computational time with respect to the traditional min-max MPC, as calculations are only performed during triggering times, it is unclear whether this approach would have a reasonable computational burden for a large system with a large prediction horizon (prediction horizon used in the studied example is rather small). Its is also unclear how such system can be compared to the traditional MPC in terms of computational time, as such information has not been disclosed.

Although Min-Max MPC would be a reasonable method to be followed, and despite researchers effort to develop new techniques enabling reduction of computational burden, Min-Max MPC still has limited applications, due to the inherent huge computational cost of solving this type of optimization problem [24].

2.3.2 Tube based MPC

Tube based MPC is a type of Robust MPC technique, which first appeared on early 2000's proposed by Mayne and Kerrigan [34]. In a very broad manner, it makes use of a nominal MPC, on which the model to be controlled is considered to be free of disturbances and on top of the former, a feedback controller operates which based on current measurements corrects the system with respect to its nominal predicted value. An advantage of this method is that we have an "indirect" upper bound on the worst case location the ship will possibly be, given that we have bounded disturbances [34]. This is guaranteed, as the ancillary, or "external" feedback guarantees that the disturbed system will be located inside a robust positively invariant set which in this case will be characterized by the cross-section of the designed tube, centered on the predicted nominal trajectory [34]. There are several different versions of this method, from which two were considered to be covering a good spectra of relevant characteristics to the proposed problem and will be discussed in more detail below. It is also important to mention that, such methods are relevant for linear and nonlinear systems, fitting our problem regardless of the system description chosen.

2.3.2.1 Tube MPC - Lipschitz bounds

In this subclass of tube MPC approach, the method maintains the overall idea mentioned above but in this case, in order to ensure the overall stability when adopting the ancillary feedback and to guarantee that the nonlinear system will be kept inside the robust invariant set upon disturbance, authors usually bound the nonlinear part of the problem by a Lipschitz bound. Borelli and collaborators for instance [25] have implemented a Tube MPC in order to handle bounded additive disturbance in a nonlinear discrete-time car model. Linear feedback was calculated using the mentioned difference between nominal and actual trajectories so as the robust invariant set was used to bound the maximum deviation of nominal and actual state under the given feedback law. The Robust invariant set (minimal positively invariant set) is proposed for the nominal system. Authors then claim that invariance of nominal system is equivalent to linear error system where nonlinearity is "upper-bounded" by a Lipschitz constant. This upper bound is then added and lumped to a new disturbance, which must belong to an enlarged disturbance set. The linear feedback gain is calculated as the gain for an infinite LQR problem. Finally, not only stability is guaranteed but recursive feasibility is ensured in case the nominal MPC is initially feasible. In such case, the MPC problem will always have a solution and the vehicle is guaranteed to keep the actual trajectory within the robust invariant set, under the influence of any admissible disturbances within the predefined bounds. It is important to highlight here that authors make the assumption that disturbances are relatively small, and the reason is that the usage of Lipschitz bounds lead to a very conservative approach.

In another example of such approach, Allgöwer and collaborators [18] developed a general method using a very similar approach when compared to the previous paper. An "inner-loop" nonlinear MPC controller addressing the nominal system is proposed, along with an ancillary feedback (in this case a linear feedback gain designed with decay rate), which was designed to handle external bounded additive disturbances. In this work, robust stability (input to state stability) and recursive feasibility is guaranteed if the optimization problem is initially feasible. In such paper, authors acknowledge the conservativeness of the traditional Lipschitz bounds and propose a less conservative approach by usage of one sided Lipschitz bounds. The benefit of such approach is that, by using an inner product they are actually able to identify and use on their favor the "direction of the nonlinearty", instead of only considering a bounded norm. In order to illustrate such approach, a constrained (state and input) 2-dof nonlinear system is controlled successfully under the bounded norm disturbance.

2.3.2.2 Tube MPC - Contraction theory

The concept of contraction theory applied to nonlinear systems analysis has been introduced by Slotine and Lohmiller [35], who adopted concepts from fluid mechanics and differential geometry in order to analyze convergence of nonlinear systems. Such idea, although implemented by very few researchers in the context of designing the ancillary feedback controller for tube MPC, has presented quite good results for simple systems and therefore will be presented here.

In [26] researchers exploited the contracting dynamics of nonlinear systems in order to develop a robust constrained MPC technique to systems being exposed to external, bounded additive disturbance. Sufficient conditions for recursive feasibility of the optimization problem under this new approach are derived (given that the optimization problem is initially feasible) so as sufficient conditions to bound the real state trajectories inside the tube like regions are derived (based on bounds of the sampling time, desired radius of the tube and maximum disturbance). As a result, the closed loop system state approaches asymptotically a control invariant set under disturbances. Authors claim that the advantage of adopting this method when compared to Lipschitz continuity, for instance is that it can tolerate larger disturbances (for a given robust invariant set) and could enlarge the feasible region accordingly [26].

Another paper approaching Robust MPC problems with such technique is [36]. Here, optimized contraction based tubes, inside which the state is guaranteed to remain within is developed. The incrementally stabilizing ancillary feedback controller is then used in order to steer actual states to a nominal trajectory calculated for a system considered to be free of disturbances. The advantages mentioned by these researchers when compared to funnel libraries [37] is that this method does not need to make use of a fixed library of maneuvers. This approach is also considered to be less conservative for highly non-linear systems when compared to methods treating nonlinearities as bounded disturbances, and also does not require decomposition into linear and nonlinear components (relying on Lipschitz bounds). Application of the proposed technique in a 2 states system with "numerically well behaved" matrices has attained very good results.

2.3.3 Online MPC and observers

In this subsection, different methods will be presented, although all of them will have the same underlying principle, which is usage of online MPC along with some sort of observer, be it for estimating disturbances or be it to estimate originally unobservable states.

Usage of such approach is presented by Pant, Abbas and Mangharam [38] who have approached the robust control of nonlinear systems problem by using feedback linearization with MPC. In their solution they consider input and state constraints, and in order to manage the original constraints, robust constraints were introduced in the system resulting from feedback linearization. Robust constraints were calculated online by means of reachability analysis and method was claimed to be recursively feasible and stable. The method was applied to a single joint flexible manipulator, presenting satisfactory results. The downside of such approach is that, constraints are not taken into account explicitly, demanding thus online reachability analysis to compute robust constraints. Furthermore, according to these researchers, numerical limitations have shown up when computing such sets online via the Multi-Parametric Toolbox (MPT) [39], not to mention the conservativeness of computed sets.

Another application of a similar method has been proposed by Yang and collaborators [30] on which they design an observer to estimate unknown disturbances and provide such measurement to surface marine vessel, making use of a controller using vectorial backstepping in order to deal with this robust non-linear control problem. The system's main goal is to track a reference, and the closed loop system is claimed to be globally uniformly ultimately bounded. Results shown depict a very good trajectory tracking capacity upon usage of the designed controller, although one of the greatest problems of such approach is that authors do not consider input constraints, which for our particular application is not suitable.

Control of marine vessels has also been the topic of research of Yin and Xiao in [4], in which a very similar approach has been adopted when compared to the previous paper. In this work, once again the main objective was vessel trajectory tracking under time varying disturbances, but this time authors have explicitly considered system uncertainties. Researchers proposed a controller composed by two parts. One portion of the controller would calculate the input signal by means of backstepping, in order to guarantee that the vessel (free of uncertainties and disturbances) would accomplish the trajectory tracking task, whereas the second term of the input signal would be calculated with an adaptive version of the previously designed controller in order to handle disturbances and uncertainties. Although very good tracking capability has been shown by the controller, once again no actuator constraints have been taken into account, which undermines this approach and makes it not viable to our purposes.

2.4 Literature review analysis

After listing techniques capable of addressing our specific problem of robust control, a comparative analysis will be done below, so that a decision on the most suitable technique can be reached.

The first technique presented, Min-Max MPC, although has shown interesting final results by researchers, suffers from the issue of high computational burden as reported by the great majority of them. Even if the worst case disturbance is adopted, in case we are to adopt it online, a very demanding computational load is believed to take place due to the high order of the system and due to the nature of its nonlinearities. Thus, this technique does not seem to be the most suitable one to tackle our problem.

The second technique, Tube MPC, on the two versions presented, has shown interesting applications and results. The version using Lipschitz bounds, suffers from low tolerance to large disturbances as sufficient conditions derived are rather strict, although it does not mean the system is not able to handle large disturbances, it just means that the system is not guaranteed to be stable anymore. The second approach presented, contraction theory, has shown very interesting results in the few papers adopting this technique. Overall approach of tube MPC techniques is very similar, although the second one takes into account the nonlinearity of the system and uses it in the contraction dynamics, in order to build the ancillary feedback. Even though this second approach seems to be very interesting, it has been only displayed in very small order and "numerically well behaved" systems.

The third technique presented, online MPC with observers, has several different types of application on the literature, including nonlinear vessel control. This technique has shown extremely good results, but such results come at a price. Usually the biggest problem with such approach is that input constraints are not considered, allowing the input signal to be as high and fast as possible, what usually explains the marvelous results attained. Even when input constraints are considered, the controller is indirectly tailored to suit the constraints and not the more ideal approach on which constraints are explicitly taken into account and then a control solution is produced.

In terms of overall behavior, Tube MPC when compared to Min-Max MPC shows a much smaller computational burden. On top of that, the computational burden may be a reason of minor concern in case a model of the system is available as the nominal MPC in itself can be calculated offline, working in a very similar fashion to a trajectory planner. When it comes to the online MPC with observers, neglecting control inputs seems rather unrealistic and not applicable to our real world problem, despite its fantastic results. Therefore, Tube MPC seem to be the most clear technique to be used in order to tackle this problem.

Chapter 3

Tube MPC theory and application

In this chapter the main theory developed in this thesis will be outlined. We will start by describing the ship dynamics as in [28], in which the control technique will be applied. Then we will move to the tube MPC theoretical explanation. The version used as guideline for this work has been based on [18] and will be described here in a generalistic fashion [18], being first tailored to dynamic positioning and to speed heading control. At last, the Line of Sight path following algorithm [1], [2] will be introduced and then the above tube MPC technique will finally be adapted to accommodate such waypoint tracker.

3.1 Mathematical model of ship dynamics

As stated before, a relevant model for simulation and control in the time domain, depicting all ship's major features has been addressed by Fossen in [28] and will be the core ship model used here. For the purpose of this work the general 6-DOF model including surge, sway, heave, roll, pitch and yaw as depicted below, can be simplified to a 3-DOF model, only considering motions in surge, sway and yaw as described in [28].

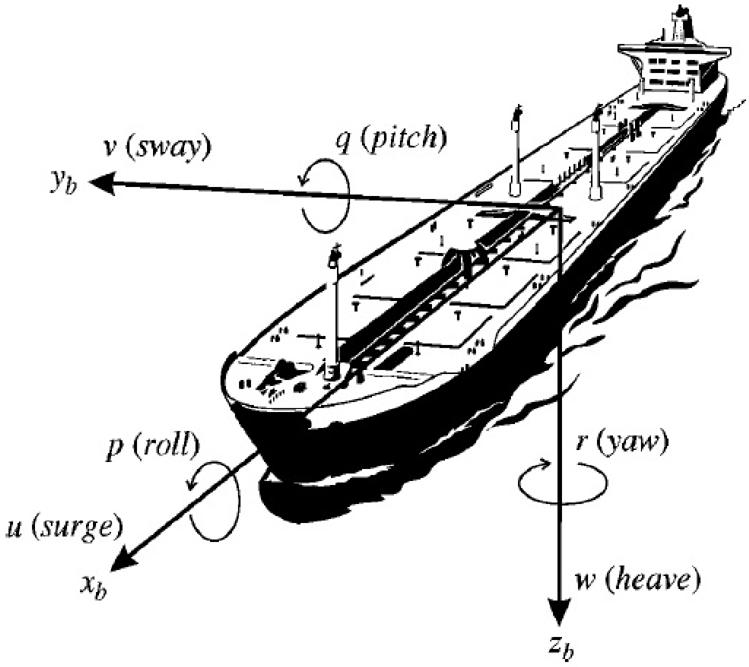


Figure 3.1: General ship motion [1]

The equations describing the ship behavior can be depicted in a compact manner below [28]:

$$\dot{\eta} = R(\psi)\nu \tag{3.1}$$

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu = \tau \tag{3.2}$$

where $\eta = [x, y, \psi]^T$ is the vector containing ship's center of mass position in the north (x) and east (y) components so as heading ψ, all of them written in the inertial frame, $\nu = [u, v, r]^T$ is the surge and sway velocity and yaw rate respectively. The ν vector is expressed in the body fixed frame (moving with the ship) and finally, $\tau = [\tau_x, \tau_y, \tau_z]^T$ or $\tau = [F_x, F_y, \tau_z]^T$ is the vector of actuator forces. As the first three components of the state vector are expressed in the inertial frame, while the last three are expressed in the body frame, the matrix R, or rotation matrix, enters then to connect these apparently separate components, by rotating the last three speed components in the body frame and relating them to speeds in the earth frame in (3.1). The M matrix is called Inertia matrix, the C matrix is called Coriollis matrix and the D matrix is called Damping matrix. They can be written as follows:

- Rotation Matrix:

$$R(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

- Inertia Matrix:

$$M = M_{RB} + M_A \tag{3.4}$$

  where:

$$M_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix} \quad M_A = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix} \tag{3.5}$$

- Coriolis Matrix:

$$C(\nu) = C_{RB}(\nu) + C_A(\nu) \tag{3.6}$$

  where:

$$C_{RB}(\nu) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix} \quad C_A(\nu) = \begin{bmatrix} 0 & 0 & c_{13}(\nu) \\ 0 & 0 & c_{23}(\nu) \\ -c_{13}(\nu) & -c_{23}(\nu) & 0 \end{bmatrix} \tag{3.7}$$

  and:

$$c_{13}(\nu) = Y_{\dot{v}} v + \frac{1}{2}(N_{\dot{v}} + Y_{\dot{r}})r, \quad c_{23}(\nu) = -X_{\dot{u}} u \tag{3.8}$$

- Damping Matrix:

$$D(\nu) = D_L(\nu) + D_{NL}(\nu) \tag{3.9}$$

  where:

$$D_L(\nu) = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix} \quad D_{NL}(\nu) = \begin{bmatrix} -d_{11}(\nu) & 0 & 0 \\ 0 & -d_{22}(\nu) & -d_{23}(\nu) \\ 0 & -d_{32}(\nu) & -d_{33}(\nu) \end{bmatrix} \tag{3.10}$$

  and:

$$d_{11}(\nu) = X_{|u|u}|u| + X_{uuu}u^2, \quad d_{22}(\nu) = Y_{|v|v}|v| + Y_{|r|v}|r| \tag{3.11}$$

$$d_{23}(\nu) = Y_{|v|r}|v| + Y_{|r|r}|r|, \quad d_{32}(\nu) = N_{|v|v}|v| + N_{|r|v}|r| \tag{3.12}$$

$$d_{33}(\nu) = N_{|v|r}|v| + N_{|r|r}|r| \tag{3.13}$$

In the above relations, $m[kg]$ is the mass of the ship, x_g is the center of gravity distance from the origin of the body frame, I_z is the moment of inertia of the vessel around the z_b axis. The numerous other coefficients showing up on the relations are hydrodynamical coefficients, which together with the previously listed parameters, are considered to be given. This model still does not consider disturbances and uncertainties, which will be addressed below.

Using an approach similar to [31], yields the final ship model with uncertainties and disturbances:

$$\dot{\eta} = R(\psi)\nu \tag{3.14}$$

$$M_0\dot{\nu} + C_0(\nu)\nu + D_0(\nu)\nu = \tau + \Delta f \tag{3.15}$$

Here the first relation is identical to the equivalent relation presented in (3.1). Nonetheless, the second relation includes the subscript zero, to denote that now we are considering the nominal model, that is, the Inertia, Coriollis and Damping matrices are adopted using measurements to the best of our knowledge, while uncertainties are not considered in these matrices. The new term Δf will accommodate all uncertainties from the previously mentioned matrices so as external disturbances and can be expressed as follows:

$$\Delta f = \tau_d - (\Delta M\dot{\nu} + \Delta C(\nu)\nu + \Delta D(\nu)\nu) \tag{3.16}$$

Now, we use Δ in front of a given component to denote uncertainty in each respective component [1], whereas τ_d denotes external disturbance being introduced in the model [31]. The assumption denoted in this thesis [31], is that $\|\Delta f\| \leq G$, where G is a positive constant. This means that disturbances and uncertainties can all be upper bounded by a given constant to be decided.

3.2 Tube MPC - General theory

In this subsection the general theory of tube MPC will be explained in order to lay ground to its more specific application. Unless told otherwise the technique and equations adopted here were inspired in [18].

Consider the continuous time system :

$$\dot{x}(t) = A_c x(t) + g(x(t)) + B_c u(t) + B_w w(t) \tag{3.17}$$

where $x(t) \in R^n$ denotes the state vector, $u(t) \in R^m$ denotes the input vector, $g(x(t))$ is a function which collects all the nonlinearities appearing in the system,

[1] Note that Δ alone will be used on the LOS algorithm. Uncertainty will never come with Δ alone, will always be followed by another letter representing the component on which uncertainty is considered.

A_c and B_c describe the linear part of the system, B_w is the disturbance matrix and finally $w(t)$ is the vector of disturbances. According to [18], given that we define an upper bound on the norm of the disturbance, namely $\|w(t)\| \leq w_{max}$, the above system can be guaranteed to be inside a robust invariant set defined by w_{max} along with parameters to be set by the user. Nonetheless, the above situation occurs only in case some sufficient conditions, listed by the paper [18] are respected. Before introducing them, let us first introduce some relevant concepts.

The nominal system associated to the original system is denoted by:

$$\dot{\bar{x}}(t) = A_c\bar{x}(t) + g(\bar{x}(t)) + B_c\bar{u}(t) \tag{3.18}$$

The nominal system will always be denoted by having a bar on top of its state and input vectors. This system is considered to be free of disturbances and uncertainties, and this is why B_w and w do not appear on (3.18).

The error system between the original system and the nominal system is defined by:

$$\dot{z}(t) = (A_c + B_cK)z(t) + [g(x(t)) - g(\bar{x}(t))] + B_w w(t) \tag{3.19}$$

The above equation describes the difference between the original system considering disturbance and the nominal system neglecting it. Note that, the input value has disappeared. The reason is that we are considering $u = \bar{u} + K(x - \bar{x})$, in other words, the total input to be given to the system will be the nominal input calculated by the MPC problem plus a term penalizing deviations of the actual trajectory of the system with respect to the nominal predicted trajectory of the system. It was also used the transformation of variable $z = x - \bar{x}$, which can be interpreted to be the error with respect to the nominal state value.

In the approach presented by the paper [18], it is also assumed $g(x)$ is a Lipschitz function, in other words:

$$\|g(x_1) - g(x_2)\| \leq L\|x_1 - x_2\|, \forall x_1, x_2 \in \mathcal{X} \tag{3.20}$$

where, L is the smallest possible constant satisfying the inequality.

Before stating the sufficient conditions ensuring that the system will be staying inside the designed robust invariant set when subject to bounded disturbances, we will also formulate the MPC problem. In the paper, the proposed MPC problem was introduced in the continuous time domain. Here we will assume that the system is sampled often enough such that we can use a discrete time formulation for the solution of the nominal MPC. The problem is depicted below [40]:

$$\begin{aligned}
&\min \sum_{k=0}^{N-1} \bar{x}_k^T Q \bar{x}_k + \bar{u}_k^T R \bar{u}_k + \bar{x}_N^T Q_f \bar{x}_N \\
&subject \quad to: \\
&\bar{x}(k+1) = f(\bar{x}_k, \bar{u}_k) \\
&x(k) \in \mathcal{X}_0 \\
&u(k) \in \mathcal{U}_0 \\
&\bar{x}_N \in \mathcal{X}_f
\end{aligned} \tag{3.21}$$

In the above problem, Q is the state penalty matrix, R is the input penalty matrix, Q_f is the terminal penalty matrix, $f(x_k, u_k)$ is the system to be controlled, $\mathcal{X}_0$ is the state set constraint, $\mathcal{U}_0$ is the input set constraint and finally $\mathcal{X}_f$ is the terminal set constraint. Note that the sets $\mathcal{X}_0$ and $\mathcal{U}_0$ have the subscript 0. The reason is that, as we are using a controller "on top of" the nominal MPC, then, constraints for the nominal controller must take this fact into account. Thus, in order to guarantee that the original constraints will be respected, a set difference was introduced by the authors in [18] and can be defined as follows:

- Let two sets A_{set} and $B_{set} \subset R^n$, then the Pontryagin set difference is defined as [41]:

$$A_{set} \ominus B_{set} = \{x \in R^n | x + y \in A_{set}, \forall y \in B_{set}\} \tag{3.22}$$

For our specific problem, it is also important to mention the concept of how to design the pair terminal penalty and terminal set and also the concept of ellipsoids generated by positive definite matrices, which, unless stated otherwise, will be adopted in the design of the terminal set constraint. Thus, let us define these two important concepts below:

- Given a positive definite matrix $P \in R^{nxn}, P > 0$, an ellipsoid may be generally defined in the space as follows [42]:

$$\mathcal{E} = \{x \in R^n | x^T P x \leq 1\} \tag{3.23}$$

- The set $\mathcal{X}_f = \{x \in E(x) \leq \alpha\}$ with $\alpha > 0$, and the function $E(x)$ (in our case chosen to be $E(x) = x^T Q_f x$) will be a terminal set and terminal penalty function, respectively, if there exists an admissible control law $\pi(x)$ ($\pi(x) = Lx$ in our case) such that:

    1. $\mathcal{X}_f \subseteq \mathcal{X}_0$
    2. $\pi(x) \in \mathcal{U}_0$, for all $x \in \mathcal{X}_f$

3. $E(x)$ satisfies the inequalities:

$$\alpha_3(\|x\|) \leq E(x) \leq \alpha_4(\|x\|) \quad \text{and} \quad E(k+1) - E(k) + l(x, \pi(x)) \leq 0$$

where both α_3 and α_4 are $\mathcal{K}_\infty$ functions and $l(x, \pi(x))$ is the running cost of the MPC problem.

The concepts above will be of paramount importance in this work and will be used in order to ensure that the nominal MPC respects the original constraints and to guarantee its stability and recursive feasibility.

It is also important to note that we will be only doing set differences on the "input domain", as the only real constraint our problems has will come from thrusters. Definition of sets and set differences operations were performed in Matlab, by usage of the Multi Parametric Toolbox (MPT) [39], while the nominal MPC problem was formulated in Matlab using Yalmip [43]. The more specific implementation steps will be described in the next section.

After acknowledging these concepts we may now introduce the mentioned sufficient conditions using the Lemma 2 of [18]:

Suppose that there exists positive definite matrix $X \in R^{nxn}$, non-square matrix $Y \in R^{mxn}$, and scalars $\lambda_0 > \lambda > 0$ and $\mu > 0$ such that:

$$\begin{bmatrix} (A_c X + B_c Y)^T + A_c X + B_c Y + \lambda_0 X & B_w \\ B_w^T & -\mu I \end{bmatrix} \leq 0 \tag{3.24}$$

and

$$L \leq \frac{(\lambda_0 - \lambda)\alpha_{\min} P}{2\|P\|} \tag{3.25}$$

Then the set

$$\Omega = \{z \in R^n | z^T P z \leq \frac{\mu w_{\max}^2}{\lambda}\} \tag{3.26}$$

is a robust invariant set for the error system under the designed ancillary feedback.

In the above LMI λ_0 is the decay rate and μ is a "tuning variable". The LMI provides the positive definite matrix P ($P = X^{-1}$) shaping the ellipsoidal robust invariant set, centered around the nominal trajectory calculated by the "nominal version" of (3.21) and inside which the actual trajectory of the system is guaranteed to be, given that we give an upper bound for w ($\|w\| \leq w_{\max}$). Another byproduct of this LMI is the ancillary feedback gain K ($K = YX^{-1}$) which will be used to steer the actual disturbed system back to the nominal trajectory. The pair P, K is not separable due to the sufficient relations derived in [18].

The above sufficient conditions together with the nominal MPC problem form the core of this technique. The conditions developed in the paper [18] are quite conservative, but, as they are only sufficient conditions, it means that even if no guarantees can be given in case such conditions are not fulfilled, the system may still be stable.

Although the above theory is developed for nonlinear systems, in this work, it will be implemented for linear systems (or successively linearized systems depending on the task) despite the high nonlinearity of the ship. The reason is that, throughout the development of this dissertation, the nominal MPC problem has been attempted to be solved with the original nonlinear system. The original nonlinear system, although in theory was believed to be the best one to be adopted when solving the nominal MPC problem (demanding the solution of a nonlinear MPC problem), turned out to yield several problems. First, nonlinear model predictive control problems are not only hard to be solved but also, depending on the system, solutions may get stuck on local minima, sometimes generating a "not even close" solution to the global optimum. Furthermore, it usually takes a huge time for the problem to be solved when using traditional solvers such as fmincon, ipopt [44] and so on. On top of all that, the most critical problem is the stability guarantee required by this method. In order to guarantee the stability of the system, the pair terminal set/terminal penalty must be designed such that it respects conditions 1, 2 and 3 in the third bullet point of this section. Such conditions, when respected usually result in the generation of very small sets, which in turn, demand a very large prediction horizon resulting in an incredibly long computational time. Due to all these practical issues, this approach was abandoned.

Once we are using a linear model, conditions (3.24), (3.25) and (3.26) can be simplified. As there is no more nonlinearity considered in the system, condition (3.25) does not need to be evaluated anymore, and (3.26) becomes:

$$\Omega = \{z \in R^n | z^T P z \leq \frac{\mu w_{\max}^2}{\lambda_0}\} \tag{3.27}$$

Therefore, only condition (3.24) will be checked in order to give guarantees that the equivalent linear system will be inside the robust invariant set produced by (3.27). Having explained the overall idea of the tube MPC technique, we may now proceed to its more specific application in the first task, which is dynamic positioning.

3.3 Tube MPC - Dynamic Positioning (DP)

In this section we are going to present the specific application of tube MPC to dynamic positioning. In the first subsection we are going to present the concept of dynamic positioning along with the necessary mathematical derivations in order to set the problem up for a subsequential implementation of tube MPC. After this part, we then explain the implementation of tube MPC for this task.

In many offshore activities, such as oil exploration and even construction work on the sea, it may be required for the ship to maintain its position and heading fixed despite all the environmental disrupting forces acting on the system like wind, waves and ocean currents [45], [46]. The activity of automatically holding its heading and position despite external forces by using thrusters is known in the marine industry as dynamic positioning [45], [46].

Implementing a robust control technique for vessel control is then required for such task and the necessary mathematical derivations for tube MPC implementation will be done below.

3.3.1 Error dynamics in DP

First, as we will be designing a terminal set and the heading position can be any value in the range $[-\pi, \pi]$, we either would have to shift the terminal set or calculate the error dynamics for the system. As direct shift of the terminal set calculated on the origin may produce "non-invariant" parts on this new set [40], or, in case only the invariant part in this new shifted set is considered, it results in an invariant set smaller than the original one [40], it was decided to use the error dynamics to perform implementation of the tube MPC technique. Making use of equations (3.1) and (3.2), we derive the error dynamics as below:

$$e_{dyn} = \begin{bmatrix} \eta - \eta_{target} \\ \nu \end{bmatrix} = \begin{bmatrix} e \\ \nu \end{bmatrix} \tag{3.28}$$

Where η_{target} is the desired target position and heading. The variable e_{dyn} will be reserved for the complete error vector while $e = \eta - \eta_{target}$. It is important to note that ν is the velocity vector of the ship written on the body frame and as the target velocity vector on the body frame is zero for the dynamic positioning task, the last three components for the error vector are equal to its actual value. Before we have the final implemented equation, it is still helpful to perform a variable change so that we can eliminate the trigonometric functions appearing in the original equations. The transformation can be seen below:

$$e_b = R^T(\psi)(\eta - \eta_{target}) \tag{3.29}$$

where e_b is the position and heading error written with respect to the body frame (whereas e collects the same variables as e_b but they are written with respect to the inertial frame). Then, taking the derivative of (3.29) with respect to time, we have as follows:

$$\dot{e_b} = \dot{R}^T(\psi)(\eta - \eta_{target}) + R^T(\psi)(\dot{\eta} - \dot{\eta}_{target}) \tag{3.30}$$

Noting that $\dot{\eta}_{target} = 0$ and that $\dot{R}^T(\psi) = S^T R^T$, where:

$$S(r) = \begin{bmatrix} 0 & -r & 0 \\ r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.31}$$

equations (3.1) and (3.2) can be rewritten as below:

$$\dot{e}_{dynb} = \begin{bmatrix} \dot{e_b} \\ \dot{\nu} \end{bmatrix} = \begin{bmatrix} S(r)^T e_b + \nu \\ M_0^{-1}(-C_0(\nu)\nu - D_0(\nu)\nu + \tau + w) \end{bmatrix} \tag{3.32}$$

Note that, the position and heading components of the state vector are written now with respect to the body frame and this is the reason why we adopt e_{dynb} instead of e_{dyn} to denote the error state vector. With this formulation now, we are ready to implement the tube MPC technique. As stated before, here we will be addressing the linear case, which will be explained in more detail below.

3.3.2 Tube MPC implementation in DP

Having made the necessary derivations, we may now move to the tube MPC implementation as explained in the paper [18]. In order to obtain a linear model of this highly nonlinear system, we linearize the system around the desired target state, which in this case as we are using the error dynamics, is always the zero vector. After linearization of (3.32) around $[0,0,0,0,0,0]^T$ we obtain a system having a general form as follows:

$$\dot{x} = Ax + Bu + B_w w \tag{3.33}$$

where:

$$A = \begin{bmatrix} 0 & I \\ 0 & -M_0^{-1}D_l \end{bmatrix} \qquad B = B_w = \begin{bmatrix} 0 \\ M_0^{-1} \end{bmatrix} \tag{3.34}$$

being I the $3x3$ identity matrix, D_l the linear part of the damping matrix D_0 and 0 denoting the zero matrix of equivalent size. For this specific problem $x = e_{dynb}$ (we will be using x for the remaining explanations). It is important to note that x here will be used as a state whereas X will be used as a variable in the LMI on the next step of the tube MPC implementation.

On possession of the linear system, we may now calculate the LMI below:

$$\begin{bmatrix} (AX+BY)^T + AX + BY + \lambda_0 X & B_w \\ B_w^T & -\mu I \end{bmatrix} \leq 0 \tag{3.35}$$

After calculation of the robust invariant set defined by the matrix P above along with the ancillary feedback gain K, we may calculate the "input" robust invariant set, which together with the original input constraints, will define the set difference presented on the previous section. The set difference in our specific case is then performed as follows:

- The original input constraint (for the actual vessel) is considered to be a box constraint. Such box is defined in MPT [39] as a set (this set is fixed due to physical limitations of the propellers). Let's call this set S_1, which can be defined in MPT as below:

  $$\begin{bmatrix} I \\ -I \end{bmatrix} u \leq \begin{bmatrix} U_{in} \\ U_{in} \end{bmatrix} \tag{3.36}$$

  where U_{in} is a vector of $3x1$ containing $[F_x^{\max}, F_y^{\max}, \tau_z^{\max}]^T$, the maximum forces in the surge and sway directions and the maximum torque in the yaw direction provided by the thrusters, while u is the thruster input decision variable.

- The robust invariant ellipsoidal set $z^T P z \leq \frac{\mu w_{\max}^2}{\lambda_0}$ is approximated by a bounding box and then defined as a set in MPT. Although such approximation is an over approximation, it has been decided to do in this manner, so that we could use the MPT toolbox to perform set differences (once the toolbox only performs set differences between polytopes). In a previous version of the algorithm used here, the bounding box was calculated by a built in function in MPT. It was noted that bounding boxes created by this function were not always accurate, not to mention its over conservativeness. It was then decided to build a bounding box based on matrix decomposition of P which can be described as follows: Let $P > 0$ be a positive definite matrix. Then we know that such matrix can be decomposed as shown below:

  $$P = R \Lambda R^T \tag{3.37}$$

  R above is the matrix containing the normalized eigenvectors of P, while Λ is the matrix containing the eigenvalues of P. The bounding box can then be built as a set in MPT defined by the following inequality:

  $$\begin{bmatrix} R^T \\ -R^T \end{bmatrix} z \leq \begin{bmatrix} \tilde{\Lambda} \\ \tilde{\Lambda} \end{bmatrix} \tag{3.38}$$

  where $\tilde{\Lambda}$ is a diagonal matrix containing the square root of the inverse of the eigenvalues appearing in Λ ($\tilde{\Lambda}$ measures the ellipsoid's main distances when it is aligned to its principal axis). For better comprehension we will provide an example. Suppose we have a robust invariant ellipsoidal set defined by:

$$A_o = \begin{bmatrix} 5 & 4 \\ 4 & 5 \end{bmatrix} \quad s.t. \quad x^T A_o x \leq 1 \quad , \quad x = [x_1, x_2]^T \tag{3.39}$$

This defines and ellipsoid with its major axis making $-45°$ with the x_1 axis. The major length of the ellipsoid on its principal axis is of 1 while the minor axis is of $\frac{1}{3}$, making an angle of 45° with the x_1 axis. Thus, using the above method to over approximate the ellipsoid yields the following bounding box:

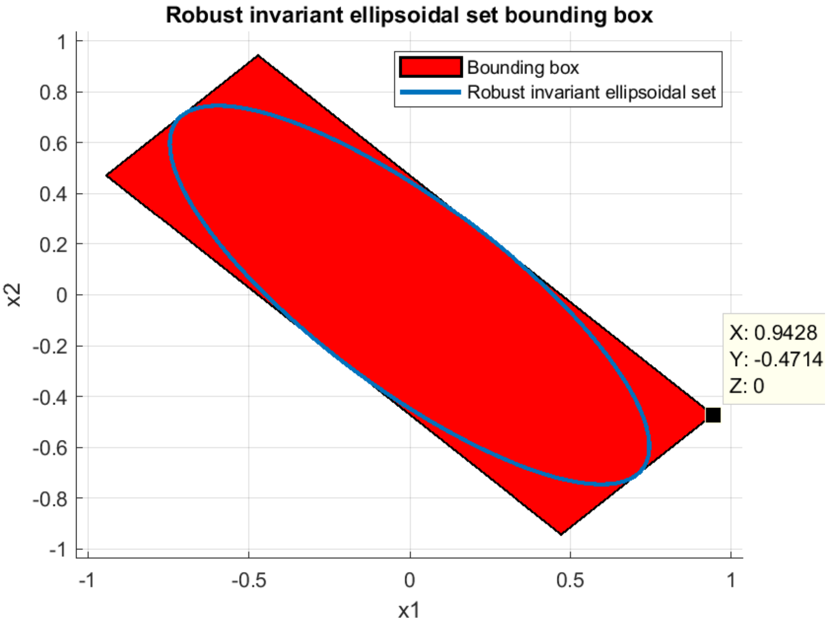


Figure 3.2: Bounding box enclosing the robust invariant ellipsoidal set defined by $x^T A_o x \leq 1$

Using the information from the main distances of the ellipsoid in the principal axis and also its direction enable us to compute the corner of the bounding box to confirm its correctness. The corner marked on the figure should have its coordinate values equal to:

$$x_1 = 1\cos(45°) + \frac{1}{3}\sin(45°) = 0.9428 \qquad x_2 = -1\sin(45°) + \frac{1}{3}\cos(45°) = -0.4714 \tag{3.40}$$

which confirms the approach chosen.

- The above bounding box is then "multiplied" by the ancillary feedback gain K (to translate) from the "state domain" to the "input domain". Now we have another set in the input domain. Let's call this set S_2
- Now the set difference is performed as $S_3 = S_1 \ominus S_2$, where the $\ominus$ symbol, as explained before, represents the Pontryagin difference. Building a bounding box based on matrix decomposition, coupled with scaling of the system for better numerical treatment has helped a lot in reduction of empty set solutions. It is important to mention that the resulting set S_3 produced by MPT is usually a box like set with its main axis aligned with the cartesian main axis, even though the bounding box calculated by the matrix decomposition does not have necessarily its main axis aligned with the cartesian axis. The reason is that when performing the set difference, MPT gets the extremal points of the set generated by the matrix decomposition and build a box like set in order to perform the set difference.

The result of the above approach for one simulation can be seen below:

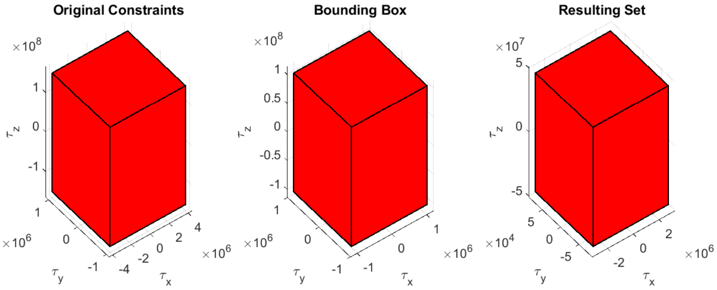


Figure 3.3: Leftmost set (S_1) comes from thruster constraints, middle set (S_2) is the bounding box of the "input" robust invariant set and the rightmost set (S_3) comes from $S_3 = S_1 \ominus S_2$. Here $w_{max} = 10^5$ and $\lambda_0 = 0.01$

The above created set S_3 is then the reduced set constraint to be imposed on the nominal system which will be calculated next. Again, as we don't have state constraints there is no need to perform another set difference. In possession of the reduced set constraint and the ellipsoid defining the robust invariant set, we may now proceed to the calculation of the nominal MPC, which is performed offline as the above set difference.

The nominal linear MPC is calculated with the matrices A and B calculated from the linearization around the target point after subsequential discretization through zero order hold. As we are calculating the nominal controller the part related to the disturbance will not be considered. The final penalty has been chosen as $x^T Q_f x$, where Q_f is the solution of the discrete algebraic Riccati equation, along with an ellipsoidal terminal set $V = \{x | x^T Q_f x \leq \alpha_1\}$, where α_1 has been chosen such that $\forall x \in V, L_\infty x \in S_3$ [18], where L_∞ is the gain obtained when solving the discrete algebraic Riccati equation. Note that with such controller we have "assumed" the ship to be linear in its dynamics, which is not true, but has shown to be a good approximation for dynamic positioning (and may also be for other low speed applications). It is also important to note that, as this step is performed offline, no model mismatch will show up here (will be handled by the ancillary feedback), thus, recursive feasibility and stability can be guaranteed for this model.

As a result of the nominal MPC calculations, a sequence of predicted states and control inputs is obtained. We store the nominal state and inputs for the simulation in Simulink. The final input, to be introduced in the Simulink model, will then be:

$$u = \bar{u} + K(x - \bar{x}) \tag{3.41}$$

where variables with a bar on top of it denote the nominal variables calculated as a result of the nominal MPC and x is the actual state of the system. The variable u then denotes the total input given to the system. This total input will be held constant in between sample times and recalculated again at each new sample time.

The above procedure can be a bit convolute at a first glance, but can be summarized by the following steps:

1. Calculation of the error dynamics using the three first components of e_{dyn} as $e = \eta - \eta_{target}$.

2. Change of variables is performed so that we can eliminate trigonometric functions - Now we have all the states written with respect to the body frame and such state vector is denoted by e_{dynb}.

3. Linearize the error dynamics presented on step 2 around the target error.

4. Solve the LMI presented based on decay rate.

5. Calculate the input set difference.

6. Solve the Nominal Linear MPC problem.

7. Store nominal predicted states and inputs and export them for the simulation in Simulink.

Having explained in detail the technique used, we may now proceed to the next tube MPC task implementation.

3.4 Tube MPC - Speed Heading (SH)

In a very similar setting to the previous section, now we will explain the tube MPC implementation for speed-heading control. Due to the same terminal set considerations as before, here we consider the error dynamics. We first explain very briefly, how the error dynamics was developed for this case. Then we move on to explaining the implementation of this technique in the vessel.

Speed heading control consists, as the name say, in steering heading, surge speed, sway speed and yaw rate by means of automatic thruster usage. Such task can be very important in the open sea, when the vessel must maintain the same speed and heading to reach a new desired destination. Disturbance may be very noxious in such task, when usually it is not uncommon to see the heading component deviate a lot from its desired value. Thus, once again, a robust controller is required to improve system's performance under disturbance.

3.4.1 Error dynamics in SH

As explained before, due to the same terminal set reasons, in this part the error dynamics will be developed. It is important to note though, that instead of having six states as before, in this problem we will be dealing only with the four last states of e_{dyn}. In particular, the three last states of this vector will suffer a small change. Before the target value for the three last components of e_{dyn} were zero, as we were only performing dynamic positioning. On the other hand, now, we will consider that they can have a constant value different from zero as target value. In other words, the error vector for this task is as below:

$$e_{dyn_{sh}} = \begin{bmatrix} \psi - \psi_{tgt} \\ u - u_{tgt} \\ v - v_{tgt} \\ r - r_{tgt} \end{bmatrix} \quad and \quad \eta_{tgt_{sh}} = \begin{bmatrix} \psi_{tgt} \\ u_{tgt} \\ v_{tgt} \\ r_{tgt} \end{bmatrix} \tag{3.42}$$

We now write (3.17) as a function of $e_{dyn_{sh}}$ getting an expression as below:

$$\dot{e}_{dyn_{sh}} = A_c(e_{dyn_{sh}} + \eta_{tgt_{sh}}) + g(e_{dyn_{sh}} + \eta_{tgt_{sh}}) + B_c u + B_w w \tag{3.43}$$

In the above expression $e_{dyn_{sh}}$ is also a function of time, and we have eliminated the explicit time dependence in the notation in order to make it lighter. The error dynamic calculation step for speed heading control is now complete and we may move to the tube MPC implementation.

3.4.2 Tube MPC implementation in SH

Although the implementation of tube MPC for speed heading has some slight differences when compared to the previous task, the overall idea is fairly similar. We start by linearizing (3.43) around the target error which is $[0, 0, 0, 0]^T$. The general expression obtained from such linearizaton is:

$$\dot{x} = A(\eta_{tgt_{sh}})x + Bu + B_w w \tag{3.44}$$

Note that we have defined the A matrix now to be a function of $\eta_{tgt_{sh}}$. This is a result of having an original equilibrium point not located on the origin. It is also important to bear in mind that whenever $\eta_{tgt_{sh}}$ changes, a new linearization must be performed and a new system will come out of this linearization. This is the situation of the speed heading control with sequential target points which will be explained later.

Having linearized the nonlinear system we may now calculate the LMI as in (3.35) but now using $A(\eta_{tgt_{sh}})$. We get once again the P matrix defining the robust invariant set inside which the system will stay under disturbance, so as the ancillary feedback gain K. Again, we have to repeat this step every time we have a new target point.

Now, a major change comes in. The calculation of the input set difference is different from what we have done previously. With the target point velocities now being different from zero, we know that whenever our original system, written with respect to the error state reaches its steady state value, the equilibrium point for (3.43) will be $(0, u_{ref})$, u_{ref} being the steady state inputs necessary to keep the system at the target state. In order to shift the whole system equilibrium to $(0, 0)$ it was decided to make a variable change when solving the nominal MPC, namely: $\bar{u} = \Delta\bar{u} + \bar{u}_{ref}$ and use $\Delta\bar{u}$ as decision variable for the MPC problem instead of $\bar{u}$. Thus we will only be calculating values around this steady state input serving as a feed-forward component, which will be, given the structure of our system, equal to:

$$\bar{u}_{ref} = -\begin{bmatrix} 0 & M_1 & 0 & 0 \\ 0 & 0 & M_2 & 0 \\ 0 & 0 & 0 & M_3 \end{bmatrix} (A_c(\eta_{tgt_{sh}}) + g(\eta_{tgt_{sh}})) \tag{3.45}$$

where M_i, $i = 1, 2, 3$ are the diagonal elements of the vessel's mass matrix. Knowing that we will be having a "steady state component" on our final input, force us to "subtract" this used quantity of input from the thruster available input capacity, requiring us to make adjustments on the previous set difference approach. The new overall procedure, using the same notation as the previous section, is explained below:

1. We perform the set difference in order to generate S_3, or the nominal input available for the nominal MPC as before.

2. Knowing that the feed-forward input component $\bar{u}_{ref}$ will be used in the system, we must do the set difference between S_3 and the set generated by

the elements of $\bar{u}_{ref}$, so that we have the actual available input signal for $\Delta\bar{u}$. A new set $S_4 = S_3 \ominus U_{ref}$, where U_{ref} is the box generated by the absolute values of the components of $\bar{u}_{ref}$ is then calculated and will be the input capacity available for the thrusters when determining the optimal values of $\Delta\bar{u}$ on the nominal MPC calculations.

Having performed the set difference, we now proceed to the nominal MPC calculation, which has also some changes when compared to the way it has been calculated for the previous task.

The nominal linear MPC is now calculated with the matrices $A(\eta_{tgt_{sh}})$ and B obtained by the linearization procedure, after subsequential discretization through zero order hold. Here, disturbance is also not considered as we are dealing with the nominal controller. The final penalty has been again chosen as $x^T Q_f x$, with Q_f being the solution of the Riccati equation. On the other hand, the terminal set has suffered a slight change. Before we were using the ellipsoidal terminal set $V = \{x|x^T Q_f x \leq \alpha_1\}$, which as already explained, requires the usage of second order conic solvers such as Sedumi [47] and SDPT-3 [48], [49], in order to generate the solution of the nominal MPC problem. These solvers, apart from being quite slow, were not giving very good solutions for this problem. It was decided then to adopt a terminal set based on a "diamond" shaped solid, generated by linear inequalities, contained inside the original ellipsoidal terminal set, so that the solver quadprog could be used. The generation of this set uses the same matrix decomposition used in (3.37) but now to decompose Q_f. The idea now is that, instead of building a bounding box with the generated decomposition, we create a "diamond" shaped solid contained by the ellipsoidal set defined by $V = \{x|x^T Q_f x \leq \alpha_1\}$. Such solid is generated by getting the "extreme points" of the ellipsoid calculated by:

$$vert = \begin{bmatrix} \tilde{\Lambda}R^T \\ -\tilde{\Lambda}R^T \end{bmatrix} \tag{3.46}$$

and using a built in command on MPT to generate the set defined by such vertices. For example, suppose that we have a matrix defining an ellipsoidal terminal set as below:

$$A_m = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.01 \end{bmatrix} \quad s.t. \quad x^T A_m x \leq 1 \quad , \quad x = [x_1, x_2]^T \tag{3.47}$$

we know that our maximum distance in x_1 is $x_{1max} = \sqrt{10}$, whereas our maximum distance in x_2 is $x_{2max} = 10$. Connecting the vertices with coordinates $v_1 = [\sqrt{10}, 0]^T$, $v_2 = [-\sqrt{10}, 0]^T$, $v_3 = [0, 10]^T$ and $v_4 = [0, -10]^T$ we define the inner box generated by the matrix A_m as can be seen below:

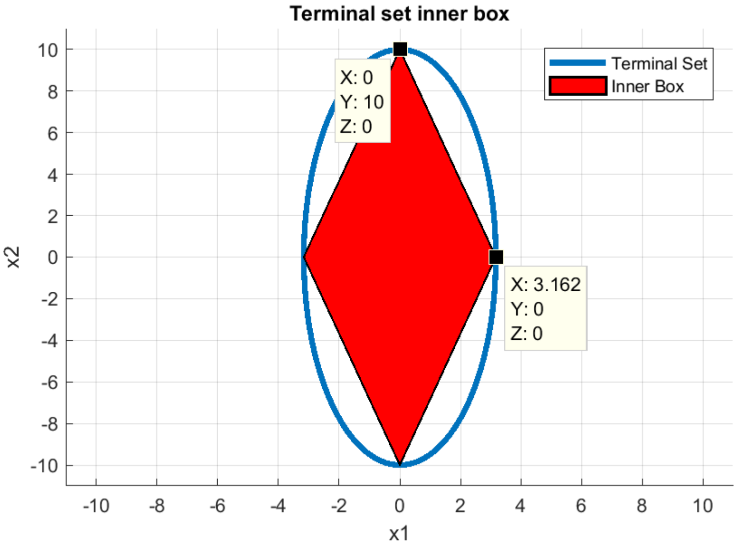


Figure 3.4: Inner box of the terminal set generated by $x^T A_m x \leq 1$.

For our simulations the above example is adapted to the correct dimensionlity, matrix and terminal set $V = \{x | x^T Q_f x \leq \alpha_1\}$.

Having defined all the necessary changes for solving the nominal MPC problem for this task, we solve it (again using Yalmip [43]) making sure that the input constraints are subject to S_4 and the terminal set used when solving the MPC problem is the inner box of the set generated by $V = \{x | x^T Q_f x \leq \alpha_1\}$. In case we are dealing with the sequential target point task we may have to repeat a few steps (to be explained below), or we solve the overall steps only once, in case we have a single goal. One last point to be highlighted is that, as we have used $\Delta \bar{u}$ as decision variable for our nominal MPC problem, we must sum again the $\bar{u}_{ref}$ value found by using relation (3.45), in order to find the total value of $\bar{u}$ to get exported. After solving the nominal MPC problem, we now store the optimal input/state sequence to export to Simulink.

In the next chapter, results from simulation will include speed heading control with sequential target points. In this task, a sequence of goals is given, along with a sequence of state penalty matrices Q. The above tasks are performed and the nominal MPC is calculated. The overall system, terminal penalty, terminal sets, ancillary feedback gain and the matrix defining the robust invariant set will remain the same until the norm of the system states converge to a value below a pre-established upper bound. Whenever such thing happen, we linearize again the original non-linear system around the new $\eta_{tgt_{sh}}$ and re-calculate the terminal penalty, terminal sets, ancillary feedback gain and the matrix defining the robust invariant set. In such task, it is of paramount importance to check, everytime the system changes, whether set differences are not empty and that the terminal set, when multiplied by the infinite horizon gain L_{∞}, is contained inside S_4.

The above procedure can be resumed in the steps below:

1. Calculation of the error dynamics using relation (3.42).
2. Linearize the error dynamics presented by relation (3.43) around the target point $[0, 0, 0, 0]^T$.
3. Solve the LMI presented using $A(\eta_{tgt_{sh}})$ found on step 2.
4. Calculate the value for $\bar{u}_{ref}$ using the relation (3.45).
5. Calculate the input set difference. Make sure to do the two step set difference and obtain the final set S_4 as explained in this section.
6. Calculate the inner box for the terminal set as explained.
7. Solve the Nominal Linear MPC problem. Make sure to subject the nominal input to be contained inside S_4 and also to use the inner box calculated on the previous step as the terminal set of the nominal MPC problem.
8. In case you have more than one target point, as in the sequential target point approach, the nominal MPC should be running subject to the norm condition. If the system state norm gets below a given threshold, steps 2 to 7 must be repeated with the new target, otherwise just keep solving the nominal MPC problem. In case you have just one task, or your current task is the last one, by the end of the nominal MPC calculation you may move to the last step.
9. Store nominal predicted states and inputs and export them for the simulation in Simulink.

Having described this second task in detail, we may now move to the explanation of the Line of sight algorithm.

3.5 Line of Sight (LOS)

In several marine applications, it is also of high interest to steer a ship along a predetermined path [50]. In such applications, given that waypoints (x_k, y_k), where x_k is the north direction component of the k_{th} waypoint and y_k is the east direction component of the k_{th} waypoint are provided, and for each line segment connecting two waypoints a velocity vector is also given, the vessel must be able to maintain the desired velocity vector while passing by the waypoints [50]. According to Fossen in [50], such path following assignment can be divided into two control problems. The first one is named geometric assignment, and the second one is named speed assignment (which usually implies the prescription of the surge speed) [50]. The Line of sight algorithm then enters to handle (along with system dynamics modification) the geometric assignment part, whereas the speed assignment is handled in a similar fashion as was done by the speed heading controller.

More specifically, the idea of the LOS algorithm is to mimic the actions taken by a helmsman when guiding a ship [2], in other words, the heading angle is adjusted based on the cross tracking error denoted by e_{LOS} in the figure below, or the perpendicular line connecting the center of mass of the vessel and the straight line connecting the "previous waypoint" to the current one, and the lookahead distance [2] denoted by Δ in the figure below, which in our case will be defined to be a constant parameter as in [2] (although such parameter may vary as described in [1]).

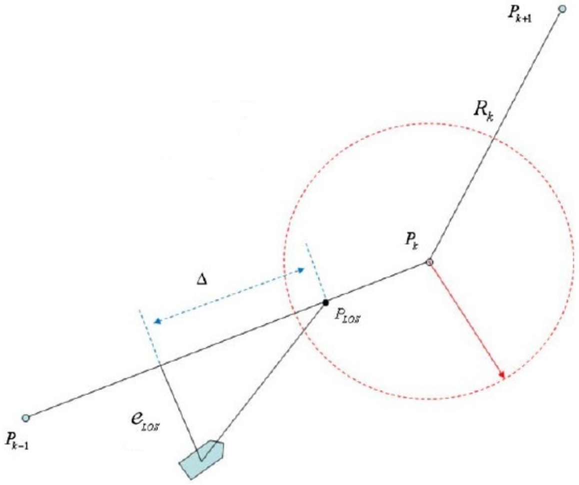


Figure 3.5: Line-of-sight scheme for straight line connection between waypoints [2].

The idea is then to steer e_{LOS} to zero, while the vessel heading relative to the angle made by the straight line connecting the past and the current desired waypoints in space must converge to the angle Ψ_{LOS} [2]. In particular the idea here would be then to drive the angle $\Psi_r = \Psi - \Psi_s$ as depicted below:

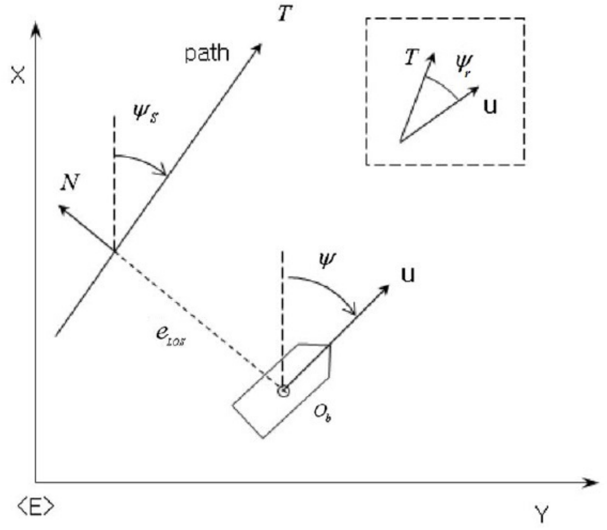


Figure 3.6: Different frames and angles used on the LOS algorithm [2].

to Ψ_{LOS} expressed in the equation below:

$$\Psi_{LOS} = -\frac{e_{LOS}}{\sqrt{e_{LOS}^2 + \Delta^2}} \tag{3.48}$$

Another aspect proposed on this algorithm is a condition on when to change the "current tracked waypoint". Usually, the condition imposed states that, whenever the vessel reaches a circle (as shown in Figure 3.5), with pre-defined radius centered on the current waypoint, then the next waypoint will be taken as the current waypoint. When such change occurs, the variable Ψ_s will have to be recalculated, so as e_{LOS}, which now will be computed having as reference the line connecting the two new used waypoints.

Having explained the overall idea of the LOS algorithm, we now move to the implementation of the LOS algorithm in the tube MPC.

3.6 Tube MPC - Line of Sight (T-LOS)

The line of sight algorithm used here will be the one presented on the previous section or the "linear like LOS" (inspired on a linear controller with varying gain), as in [2]. This LOS algorithm has some simplifications when compared to the more traditional one presented in [1]. First, the arctan function describing the Ψ_{LOS} is approximated by the relation (3.48) presented on the previous section. Another approximation is that in [2] the sideslip angle, or the angle between the resulting velocity (calculated by $U_{res} = \sqrt{u^2 + v^2}$) and the surge speed velocity u can be neglected. Bearing this in mind we may now start the necessary system modifications for our specific case.

3.6.1 Error dynamics in T-LOS

Here once again due to the same terminal set aspect we must derive the error dynamics, which is rather similar to the error dynamics developed for SH. The difference is that now we will have 5 states instead of 4 and we will not be dealing directly with the vessel heading anymore but with the "relative heading" Ψ_r. From now on, the Ψ variables reserved for the LOS algorithm theoretical explanation will be substituted by ψ, with its respective subscripts. The error vector can be depicted as below:

$$e_{dyn_{LOS}} = \begin{bmatrix} e_{_{LOS}} \\ \psi_r - \psi_{_{LOS}} \\ u - u_{tgt} \\ v - v_{tgt} \\ r - r_{tgt} \end{bmatrix} \quad and \quad \eta_{tgt_{LOS}} = \begin{bmatrix} 0 \\ \psi_{_{LOS}} \\ u_{tgt} \\ v_{tgt} \\ r_{tgt} \end{bmatrix} \tag{3.49}$$

The continuous time error dynamics can now be written as:

$$\dot{e}_{dyn_{LOS}} = A_c(e_{dyn_{LOS}} + \eta_{tgt_{LOS}}) + g(e_{dyn_{LOS}} + \eta_{tgt_{LOS}}) + B_c u + B_w w \tag{3.50}$$

In the above expression $e_{dyn_{LOS}}$ is also a function of time, and we have eliminated the explicit time dependence in the notation in order to make it lighter. The tricky part of the equation above is that $\psi_{_{LOS}}$ is a function of $e_{_{LOS}}$ and this demands some changes on the way the tube MPC is implemented.

3.6.2 Tube MPC implementation in T-LOS

The implementation of tube MPC in LOS is in general similar to the implementation of tube MPC in SH but now, the fact that our linearized function as shown below:

$$\dot{x} = A(\eta_{tgt_{LOS}})x + Bu + B_w w \tag{3.51}$$

depends on the target $\eta_{tgt_{LOS}}$ on which the component $\psi_{_{LOS}}$ is a function of the current state, prompt us to make some changes on the way the implementation is derived. The big problem now is that our system will be changing on each sampling time and in theory we must perform steps 2 to 7 from the SH task every sampling time. Condition 8 also exists here and now is subject to the entrance of the center of mass of the vessel in the circle defined around the waypoint. Only then step 9 would be performed. Steps 3,5 and 6 when solved at each sample time generate a quite large computational burden. On top of that, it increases the chance of numerical problems due to the set difference step. Therefore two versions were created for this task. In version 1, the calculations are performed as they should be and can be summarized in the following steps below (performed in the same way as in SH, only making the necessary dimensionality changes):

1. Calculation of the error dynamics using relation (3.49).

    - Now, at each sample time perform the following actions:

2. Linearize the error dynamics presented by relation (3.50) around the target point $[0, 0, 0, 0, 0]^T$.

3. Solve the LMI presented using $A(\eta_{tgt_{LOS}})$ found on step 2.

4. Calculate the value for $\bar{u}_{ref}$ using now the relation:

$$\bar{u}_{ref} = -\begin{bmatrix} 0 & 0 & M_1 & 0 & 0 \\ 0 & 0 & 0 & M_2 & 0 \\ 0 & 0 & 0 & 0 & M_3 \end{bmatrix} (A_c(\eta_{tgt_{Los}}) + g(\eta_{tgt_{Los}})) \tag{3.52}$$

5. Calculate the input set difference. Make sure to do the two step set difference and obtain the final set S_4 as explained in the previous section, but now using the values of u_{ref} from this section.

6. Calculate the inner box for the terminal set as explained.

7. Check whether the two set difference operation above return non-empty sets. If so, we may move to the next step, if not we will give the linearized system, ancillary feedback gain, terminal penalty, terminal set and reduced input constraints from the most recent sample time when both sets existed. Note that due to the assumption of having non-empty sets on the first waypoint and whenever we change waypoints, we will not have problems with empty sets along the way.

8. Solve the Nominal Linear MPC problem. Make sure to subject the nominal input to be contained inside S_4 and also to use the inner box calculated on step 6 as the terminal set of the nominal MPC problem (given that the condition above is respected). The dynamical system used here, so as the constraints will be valid only during this sampling time.

9. Store the current optimal input and trajectory.

10. Estimate the current position of the center of mass of the vessel with forward Euler approximation. This will be necessary in order to check whether the vessel is inside the "threshold circle" on step 12.

11. Update the value of $\psi_{_{LOS}}$ and subsequently $\eta_{tgt_{LOS}}$. On the next sampling time the system will use this new information in (3.50) prior to the linearization.

12. As we will be dealing with multiple target points, use the position estimation found on step 10 to compare the location of the vessel with the current target point as follows:

$$\sqrt{(pos_x - p_k(x))^2 + (pos_y - p_k(y))^2} \leq R_{thresh} \tag{3.53}$$

where pos_x and pos_y are the current components in the north and east direction of the vessel's center of mass, whereas $p_k(x)$ and $p_k(y)$ are the same components but this time for the current waypoint to be tracked and finally R_{thresh} is the circle radius as depicted in figure (3.5). In case the condition (3.53) holds, we may move to the next waypoint. In such case calculation of $e_{_{LOS}}$, ψ_s and $\psi_{_{LOS}}$ will change and must be updated for the next iteration. If such condition does not hold we keep repeating steps 2 to 12 up to when we can change the waypoint and make the necessary modifications.

- Then after the end of the algorithm:

13. Gather stored nominal predicted states and inputs and export them for the simulation in Simulink.

In the above approach, not only a large computational burden appears but it may happen that, in case we remove condition 7 and leave the same tuning for the LMIs during the whole simulation, empty sets may appear during the computation. To avoid such problem, the condition on step 7 is of paramount importance. If current calculations return an empty set difference when calculating S_4, or the empty set difference returned comes from "lack of available input in S_4" for keeping the state x_N inside the terminal set (with L_∞ as control action), then we force the system to use during the current sampling time the same linearized system, same ancillary feedback gain (generating then the same robust invariant set as we have the same ancillary feedback gain, the same LMI tuning and the same linearized system), the same reduced input constraint, terminal constraint and terminal penalty as appearing in the closest sampling time returning two nonempty sets. Next loop will then evaluate the system around a new linearization point, meaning that if in the next round the sets exist, we go back to the "normal code situation" otherwise we keep again the same parameters as stated before.

In version 2 of the Line of Sight implementation everything remains the same, but the set differences will only be computed when the target changes. This implies in maintaining the ancillary feedback gain, robust invariant set, reduced set constraint, so as the terminal set for the nominal MPC and only changing it whenever we change the waypoint. The system, on the other hand, will be linearized and updated at each sampling time. Although this is not in theory, the most correct way to perform the calculations as we only guarantee local stability for the MPC and system obeying input restrictions properly when waypoint changes, it has shown to be an interesting approach, reducing the computational burden while maintaining the good system performance.

It is very important to mention though, that the two versions assume that at least when waypoints are getting changed, the two set differences are nonempty. Having explained how the tube MPC for LOS has been implemented we now move to the simulation results of the techniques explained in this chapter.

Chapter 4

Simulation Results

In this chapter we are going to evaluate the results of the above technique implementation. Numerical simulation done in Matlab/Simulink was performed in a real sized vessel named ABB Azipod Cruiser, with a length of 294 m, a beam of 37.9 m and a draft of 8 m. For the simulations it is assumed that we will have access to all the states on the system and also that a given bound on the maximum disturbance prior to the simulations is available. Disturbance from wind and ocean current will then be introduced on the system, and performance will be measured.

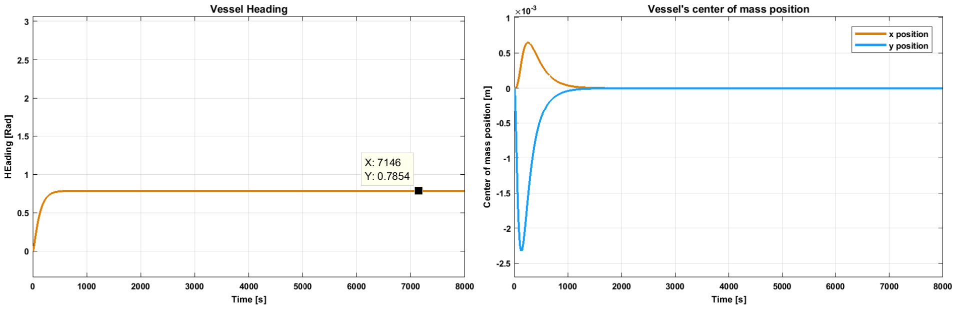
An important remark, before diving into the simulations, is that for each task, given the mismatches between model and system, the ancillary feedback must be tuned properly, otherwise the closed loop system behavior may become unstable. Very aggressive gains, which would generally come alongside small robust invariant sets generated by LMI calculation, would often drive the system instantly unstable, as we would be operating very far from the assumed linear region. On the other hand, very weak gains for the ancillary feedback, would let the system deviate a lot from predicted inputs, making the ship drift away too much, producing very bad results. Thus a good balance must be found for each task. Depending on the tuning, such balance may produce sometimes the side effect, in theory, of low disturbance tolerance (due to empty set differences), which will not necessarily mean that the actual system will not be able to handle higher disturbances. Having this in mind we may move to the simulation results. In the first three subsections we are going to evaluate results for dynamic positioning, for speed heading control and for tube MPC for line of sight and then we will finalize this chapter with a discussion about results obtained by this technique .

4.1 Simulation Results - Dynamic Positioning

In this section the steps presented on the dynamic positioning part of the previous chapter were implemented in a Simulink model in order to simulate the vessel behavior under external disturbances. Our evaluation approach here will be to analyze how close to the desired task the vessel can stay when bounded disturbances act into it. It is important to remember that the stability guarantee is only for the equivalent linear system in this case, and thus implementation of this controller on the real ship can yield, in theory, a closed loop unstable system despite its good performances in simulation.

For dynamic positioning, limitations on the upper bound disturbance on relation (3.27) has appeared upon realization of the set difference operations with disturbance norms higher than 10^5. For this task, and for the tuning chosen for the LMI, which would lead to good performances of the ancillary feedback on the actual vessel, set differences operations with disturbances norm above the mentioned limit would return an empty set. Therefore, when generating the model in Matlab workspace a maximum upper bound of $\|w\| \leq 10^5$ will be adopted, although we will see on simulations that the system ends up handling well higher disturbances than this imposed upper bound. It is also important to bear in mind that this theoretical bound on the norm is subject to the system tunning, thus, higher disturbance of the equivalent linear system could be, in principle, allowed if we choose a different tuning (although performance with such tuning may not be good).

In order to evaluate system performance, we will simulate the model having two different target headings, namely, 45° and 160°. First, results without disturbances will be shown, then results with wind disturbances norm around 10^5 will be disclosed, and finally simulation with wind disturbances norm around 10^7 along with ocean currents will be displayed. Simulation results are depicted below for a simulation time of 8000 s, target heading of 45° or 0.7854 rad , sampling time of 4 s, prediction horizon N=200 and no disturbance :



(a) Heading of 45° - no disturbance.

(b) Center of mass - no disturbance.

Figure 4.1: Vessel 45° heading task without disturbance

The closed loop system without disturbance attains its final goal of steering the ship to the 45° heading angle while maintaining its center of mass in the origin.

Below it can be shown the input demanded by such task:

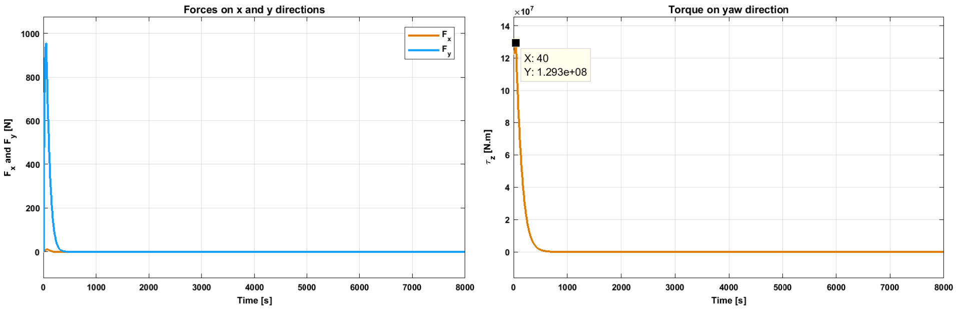


Figure 4.2: Forces and Torque used during undisturbed simulation for heading of 45°

Total forces (nominal input signal + input signal from ancillary feedback) demanded by the thrusters are below the constraints. An important remark is that, although the input signal looks continuous, by zooming in the figure it is possible to see that the value is held constant in between sampling times, generating then a piecewise constant signal. Now we move on to the same analysis for wind disturbances having norm around 10^5. It is important to acknowledge that disturbances will be entering the system in all possible 6-Dof although our controller has only used a 3-Dof linear model of the vessel. The disturbance used can be seen below:

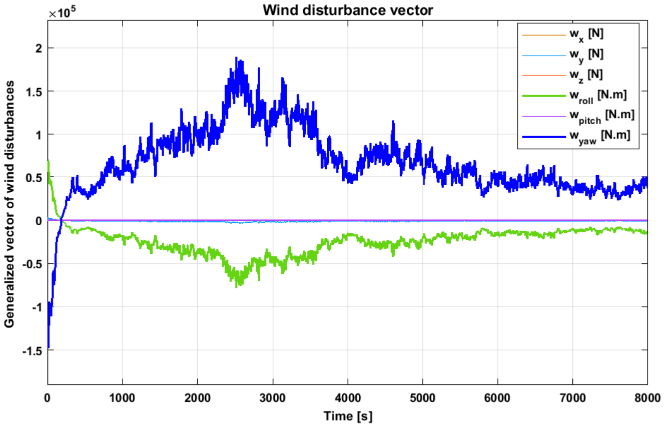
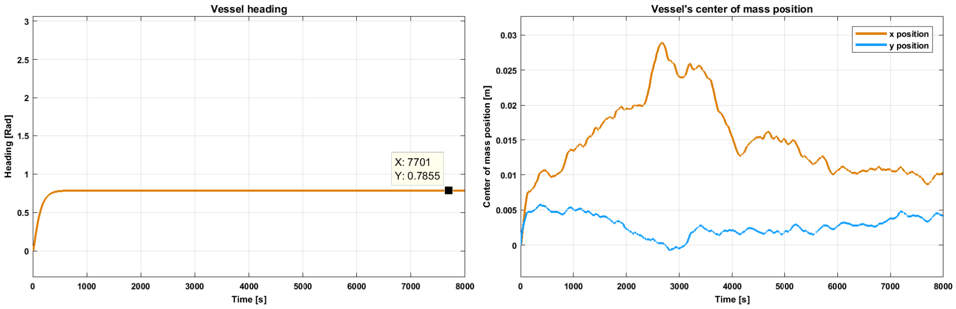


Figure 4.3: Wind disturbance entering the system.

Results of the vessel behavior when such disturbance enters the system can be seen below:

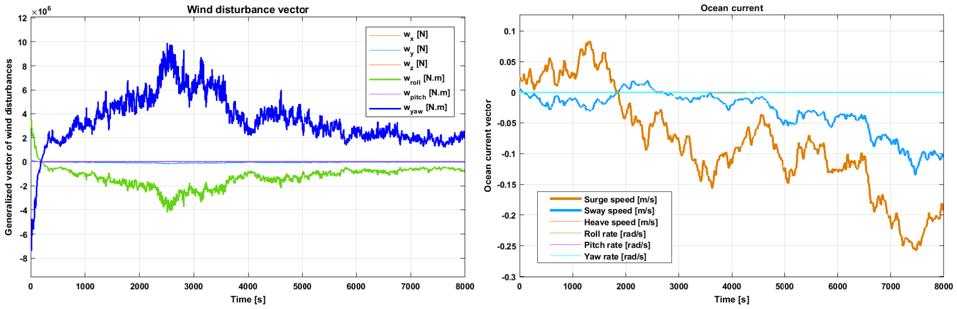


(a) Vessel heading angle

(b) Vessel's center of mass motion

Figure 4.4: Vessel 45° heading task with "low" disturbance

Note that in the figure 4.4, although it seems that the heading is constant it varies with time as in the center of mass motion figure (although in a smaller scale). The reason for it to appear constant is, first, because of the scaling of the axis, and second, the ancillary feedback controller for this task has been designed such that deviations with respect to the nominal values of heading and yaw rate will be, comparatively, much more penalized than deviations from the nominal values of x position, y position, surge speed and sway speed, explaining deviations for this simulation much smaller in the heading component when compared to other vessel motion components under disturbance. The input signal for this task will not be disclosed in order to avoid repetition, but has also respected the required constraints. All in all, the task can be said to be successfully attained given the small movement of the center of mass coupled with the heading being held very close to the desired position. Now we move on to the high disturbance case along with ocean currents enabled. Below we see the disturbances entering the system.

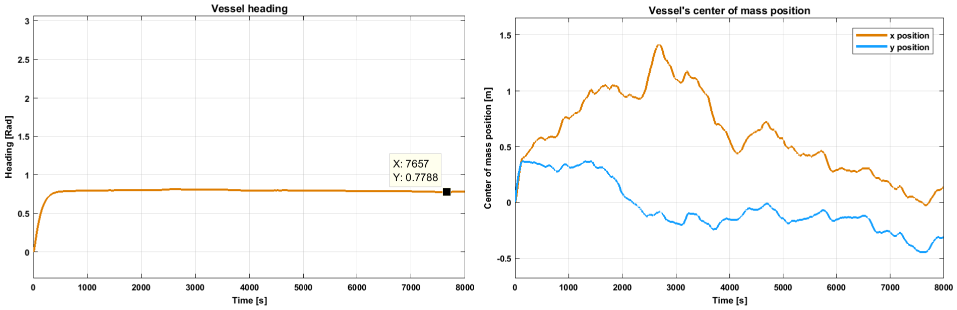


(a) High wind disturbance

(b) Ocean currents

Figure 4.5: Vessel 45° heading task subject to "high" disturbances

Results of such disturbance entering the system can be seen below:



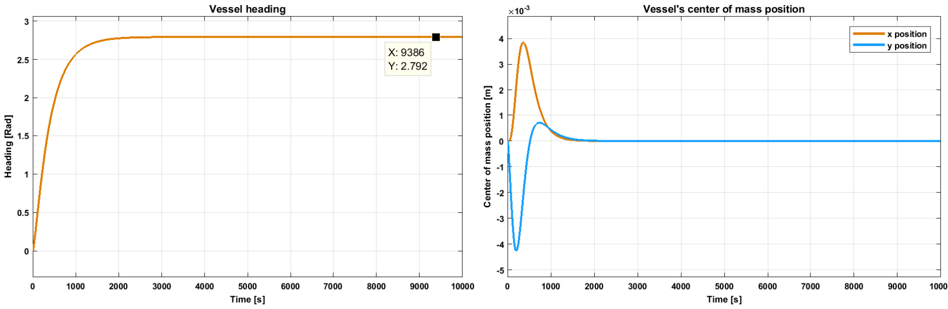
(a) Vessel heading

(b) Center of mass motion

Figure 4.6: Vessel 45° heading system response with high disturbance

Although disturbances are bigger than the defined upper bound chosen during the design phase, model mismatch can also be quite big (mainly on transient phases) and despite ocean current introduction, the system behaves relatively well holding its position as required. More oscillation is seen on the heading and center of mass components, which can somehow be expected given the high disturbance the system is subject to. The system behaved quite well to the 45° heading task despite big disturbances.

Now we move on to the simulation results of the above tests with the heading set to 160° or 2.1925 rad, same sampling time and prediction horizon, but this time with a simulation time of 10000 s. This time the tuning for the nominal MPC controller has been less aggressive with respect to the heading angle while remaining penalties were maintained. We start with the undisturbed system:



(a) Vessel heading angle

(b) Vessel's center of mass motion

Figure 4.7: Vessel 160° heading task undisturbed

Input results will be omitted in order to avoid repetition but have been as well below the required constraint. Despite the model mismatch and the greater length traveled by the ship, leaving "more space" for the nonlinearities to be "activated" in a harmful manner, the system has shown a good behavior. We present below the low disturbance case (wind disturbance norm around 10^5) and ocean currents disabled:

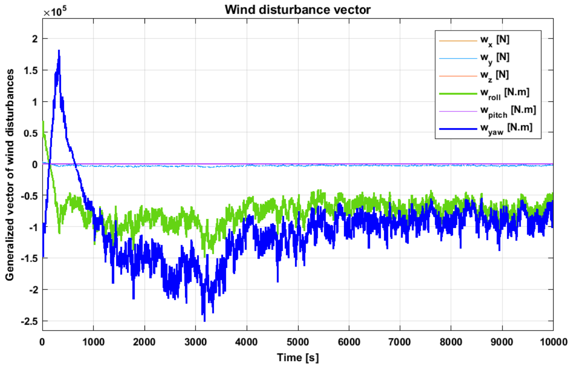
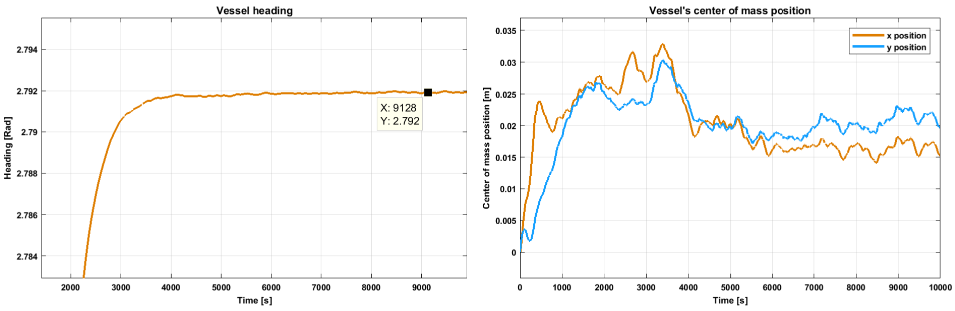


Figure 4.8: Wind disturbance entering the system with a heading task of 160°.

The controlled system response to the above disturbance is as follows:

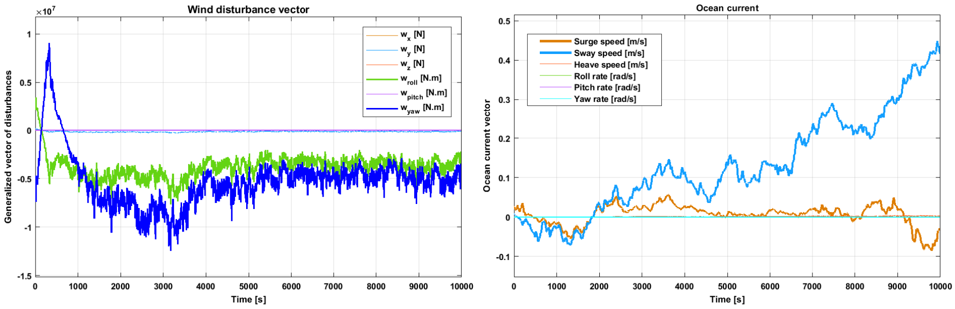


(a) Vessel heading angle

(b) Vessel's center of mass motion

Figure 4.9: Vessel 160° heading task with wind disturbance only

The heading graph has been zoomed in so that the small deviations from the target heading, due to external disturbances, could be seen. Heading keeps quite close to the desired value, whereas the position of the center of mass, shown on the right also remains small. This task can be considered harder than the first one, as briefly mentioned before, the longer turning required for task completion makes the model more susceptible to the nonlinearities of the system leaving room for worse performance due to the model mismatch. Nonetheless the system has behaved quite well while managing to keep inputs under constraint. Higher disturbances along with ocean current will now enter the system as below:

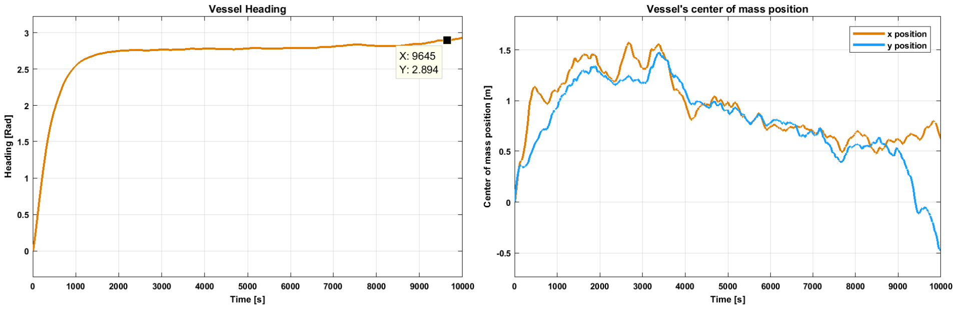


(a) High wind Disturbance

(b) Ocean currents

Figure 4.10: Vessel 160° heading task with high wind disturbance and ocean currents enabled

Including ocean currents further increases the challenge for the controller as the model used for control development has assumed no ocean current. This, coupled with the wind heavily disturbing the system makes this task very challenging for the system. Results can be seen below.

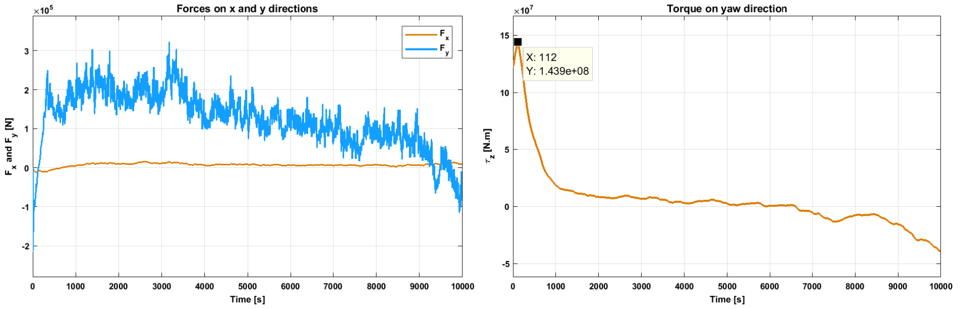


(a) Heading task with wind and current disturbance

(b) Motion of the center of mass under wind and current disturbance

Figure 4.11: Vessel 160° heading task with high wind disturbance and ocean currents enabled

Both the heading and the center of mass have shown some reasonable behavior with the heading task, oscillating but maintaining its value close to the target, and the center of mass position variating more but still maintained under control. By the end of the simulation time it can be seen that the results start deteriorating a bit. The reason for this is that the sway component of the ocean current started to increase a lot (on top of a reasonably high wind disturbance). Input results for this simulation are as follows:



(a) Surge and Sway actuator forces

(b) Yaw torque

Figure 4.12: Total input required by the task

Inputs are still below the constraints and as said before, although inputs look continuous they are hold for the duration of the sample time ($T_s = 4s$ in this case), before they get changed. The 160° heading target can also be said to be performing relatively well despite all the mismatch and disturbances the system has been imposed. We now may move on to the speed heading control results.

4.2 Simulation Results - Speed Heading

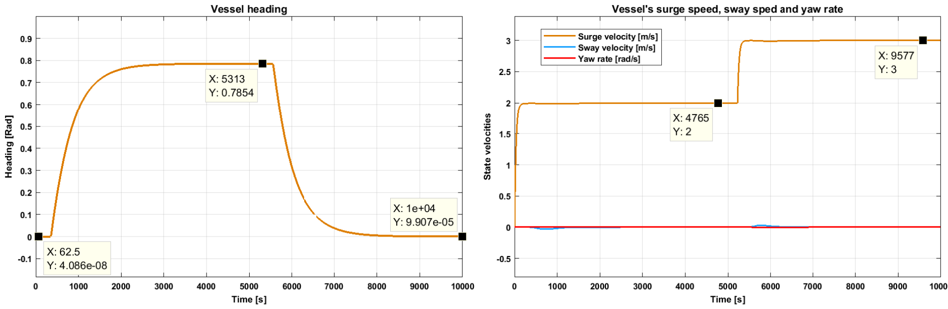
The technique studied on chapter 3 has also been implemented for speed heading control. As for the dynamic positioning simulation, some considerations are still valid here and are worth remembering. First, the stability is guaranteed only for the equivalent linear system and no statements are made about the stability of the overall system, despite interesting results on simulations. Here, there has also been some limitations regarding the amount of disturbance that could enter the system. Set difference, during the sequential task computation for example, would become empty for the chosen ancillary feedback tuning whenever the overall disturbance norm was bigger than 10^4. Note that for the sequential task speed heading simulations, set difference must be nonempty for all tasks (as for each task we have a different linearized system), which makes the offline design and solution of the nominal MPC + ancillary feedback alone much harder. System evaluation will be done on the same basis, by examining how close to the actual desired task we get under external disturbance.

The simulation shown here will be a sequential task speed heading control. In other words, the system will have to complete the following tasks from left to right as shown below:

$$\eta_{tgt_1} = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 0 \end{bmatrix} \quad \eta_{tgt_2} = \begin{bmatrix} \frac{\pi}{4} \\ 2 \\ 0 \\ 0 \end{bmatrix} \quad \eta_{tgt_3} = \begin{bmatrix} \frac{\pi}{4} \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad \eta_{tgt_4} = \begin{bmatrix} 0 \\ 3 \\ 0 \\ 0 \end{bmatrix} \tag{4.1}$$

Thus, departing from the origin, task 1 consists of making the vessel move with 0° of heading, surge speed of 2 m/s, sway speed of 0 m/s and a yaw rate of 0 rad/s. Then in task 2 the vessel must change its heading to 45° while maintaining its surge speed, sway speed and yaw rate. Task 3 requires the vessel to increase its surge speed to 3 m/s while maintaining its previous components as they are. Finally task 4 will demand the system to move with 0° of heading, surge speed of 3 m/s and keep the sway speed and yaw rate to zero. As mentioned before each task will be considered as "done", when upon nominal MPC solution, the error state, or in other words, $\bar{e}_{dyn_{sh}}$ gets below a given value, which for simulation has been chosen to be below 5x10^{-5}. Whenever this happens, unless this is not the last task, case on which the system will remain solving the nominal MPC with the current "information" up to the completion of simulation time, the system will move on to the next task.

For simulations a sampling time of $T_s = 2s$ has been chosen. Total simulation time has been of 10000 s and prediction horizon is N=250. For each task, a different tuning of Q and R matrices (ideally, different tuning also for the LMI as well) has also to be provided to the system. For this simulation, penalties of Q would on all cases be quite big for heading and very big on the sway component when performing tasks 2 and 4. The reason is that, changing heading while having a "reasonably" large surge speed activates the nonlinearities of the system quite badly. Thus, during the offline simulations, whenever sway speed increased too much, the tuning chosen would produce a very bad behavior on the actual system. Thus, whenever a change of heading is planned while moving the system, it is advised to penalize heavily the sway component. The simulation results for the undisturbed system can be seen below:

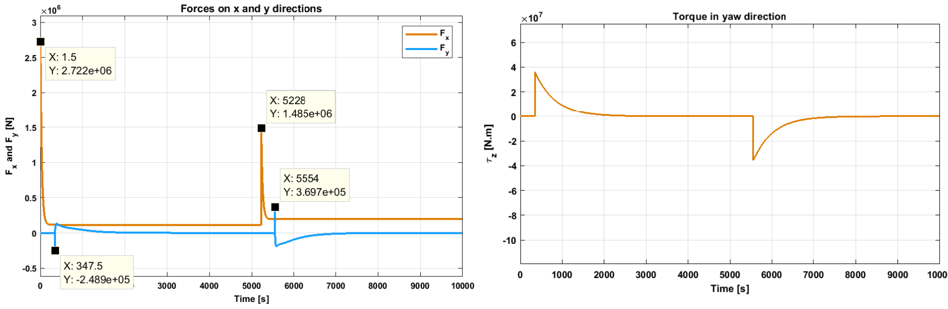


(a) Vessel Heading

(b) Surge speed, sway speed and yaw rate

Figure 4.13: Sequential task speed heading control - undisturbed

The system behaves quite well despite the model mismatch. It can be seen on the right figure that whenever task two and four are starting, sway velocity increases a bit, but it is managed to be controlled and steered back to zero. The total input used for this task can be seen below:



(a) Forces on surge and sway directions

(b) Torque on yaw direction

Figure 4.14: Input signals for sequential task speed heading control - undisturbed

The above signals are total input signals (nominal input + ancillary feedback). Nonetheless, in this specific simulation it was noted that the ancillary feedback was "weekly" activated, meaning that the choice of a linear model to predict states was quite accurate for this low speed, undisturbed application despite all the vessel nonlinearities. All peaks are below the constraints and have been in its great extent "predicted by the nominal MPC". All in all the system has behaved quite well and tasks were properly performed. We now move to the low disturbance case (disturbance norm around 10^5). Below the disturbance which the system has been subject to is displayed:

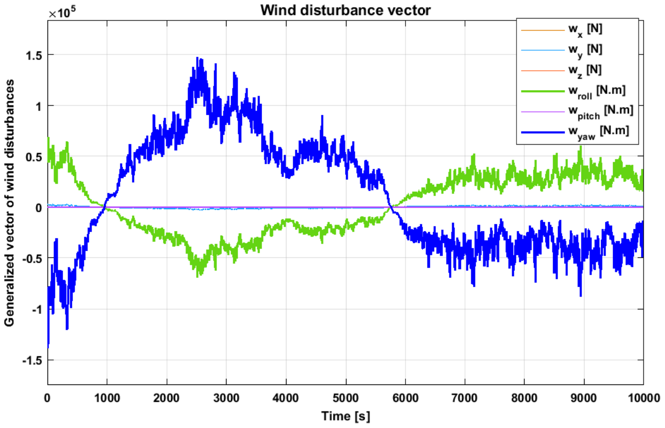
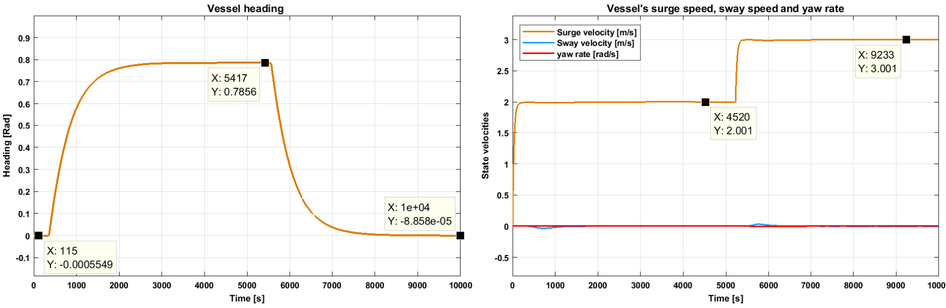


Figure 4.15: Wind disturbance entering the system.

System behavior can be seen below:

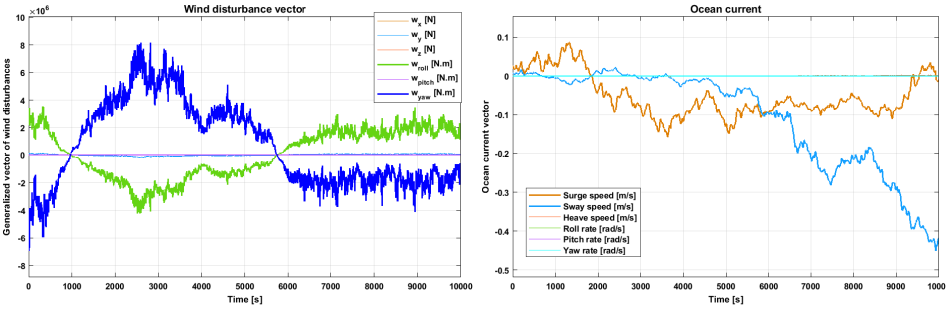


(a) Vessel heading

(b) Surge speed, sway speed and yaw rate

Figure 4.16: Vessel behavior for sequential task speed heading control - low disturbance

Input signals will not be shown to avoid repetition but were also below constraints. The system has once again behaved fairly well and both heading, surge speed, sway speed and yaw rate have not deviated too much under the small disturbance case. We now move to the last case incuring high disturbance and ocean currents enabled. Disturbances entering the system can be seen below:

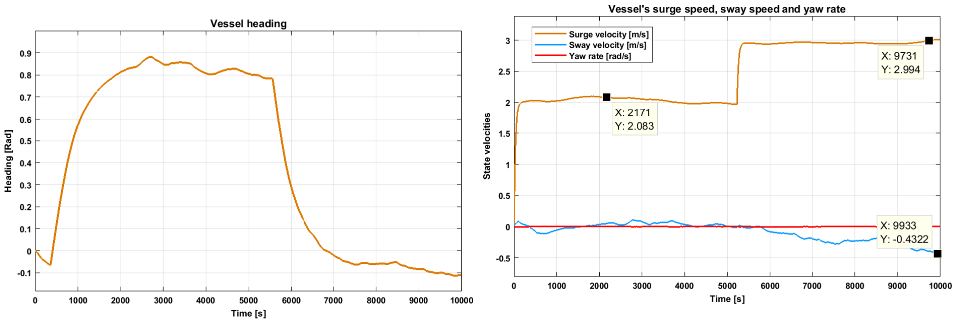


(a) High wind Disturbance

(b) Ocean currents

Figure 4.17: High wind disturbance and ocean currents entering the vessel

The vessel behavior under this type of disturbance can be seen below:

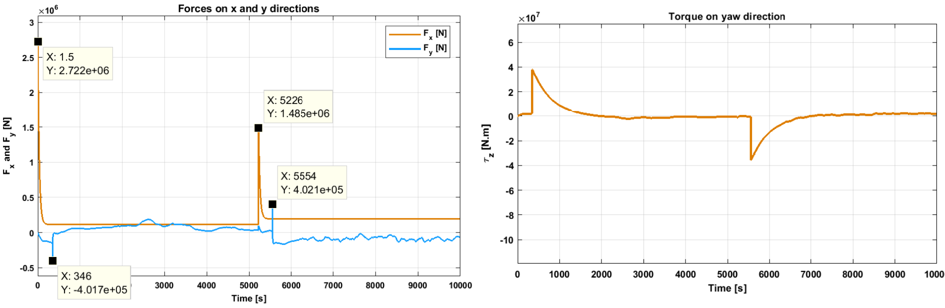


(a) Heading under high disturbance

(b) Vessel heading and speed under high disturbance

Figure 4.18: Vessel sequential speed heading task with high wind disturbance and ocean currents enabled

In this figure it can be seen that heading is attained relativelly well, so as surge speed and yaw rate but sway speed deteriorates quite a lot in the end. Again, the tuning chosen for the ancillary feedback was to penalize much more heading and yaw rate deviations than any other component of the state. The reason for this choice was that, during simulation time it was seen that once heading deviates too much under disturbance, the whole system task may get jeopardized. It is also important to mention that, for this task, it was quite hard to find a tuning in which sway velocity deviations would get more penalized by the ancillary feedback controller. Due to the lower input capacity in the sway direction, whenever more aggressive gains were demanded from the ancillary feedback when solving the LMI, the set difference would usually return an empty set. Thus it is usually the case that deviations in the sway component would not get "as penalized as it should". Once again here we see that introduction of ocean current (mainly due to its component on sway direction) on top of high wind disturbance is quite noxious component to the closed loop system performance. The input signals used in this simulation are as follows:



(a) Forces on surge and sway directions

(b) Torque on yaw direction

Figure 4.19: Input for sequential speed heading task with high wind disturbance and ocean currents enabled

On the left figure we see that the force on sway direction is oscillating much more than the force on surge direction which seems almost constant. Disturbances on surge direction were much less intense and gains in surge direction from the ancillary feedback designed by the LMI are relatively small.

4.3 Simulation Results - Line of Sight

In chapter 3 we saw that two different approaches were created to model this task, named version 1 and version 2. Given that not much performance improvement was seen when implementing version 1 over version 2, and having in mind the possible downsides of version 1, we will be showing here results for version 2 only. Thus we will be considering set differences whenever the waypoint changes and will keep the reduced nominal input constraint, terminal set and ancillary feedback constant in between waypoints. Here we also have the disturbance norm constraint, which for this case will be $\|w\| \leq 10^4$.

For this task it was required for the vessel to, starting from p_0, go from p_1 to p_4 as listed below:

$$p_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad p_1 = \begin{bmatrix} 1000 \\ 100 \end{bmatrix} \quad p_2 = \begin{bmatrix} 2000 \\ 250 \end{bmatrix} \quad p_3 = \begin{bmatrix} 5000 \\ -50 \end{bmatrix} \quad p_4 = \begin{bmatrix} 7000 \\ 150 \end{bmatrix} \tag{4.2}$$

Note that points are written on the NED coordinate system, meaning that the first coordinate (x) is the vertical axis, while the second coordinate (y) is the horizontal axis. Associated to each path, a surge velocity, sway velocity and yaw rate has been attributed. Thus vel_1 for instance will be the velocity required on the path between p_0 and p_1, vel_2 the velocity between p_1 and p_2 and so on. The desired velocities can be seen below:

$$vel_1 = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} \quad vel_2 = \begin{bmatrix} 4 \\ 0 \\ 0 \end{bmatrix} \quad vel_3 = \begin{bmatrix} 4 \\ 0 \\ 0 \end{bmatrix} \quad vel_4 = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} \tag{4.3}$$

The first vector component is the surge speed, the second one is sway speed, both in $[m/s]$ and the last one is yaw rate in $[rad/s]$. It is also necessary to define some more parameters on this technique such as presented on the previous chapter. For simulation purposes Δ has been set to be 3 times the length of the ship as in [2] and R_{thresh}, although in [2] has been chosen to be equal to the ship's length, here has been set to be equal to 100 m so that waypoints could be closely tracked. The sampling time chosen was $T_s = 2s$, the prediction horizon has been set to be N=250 and the simulation time was $T = 2900s$. Results for the undisturbed system can be shown below:

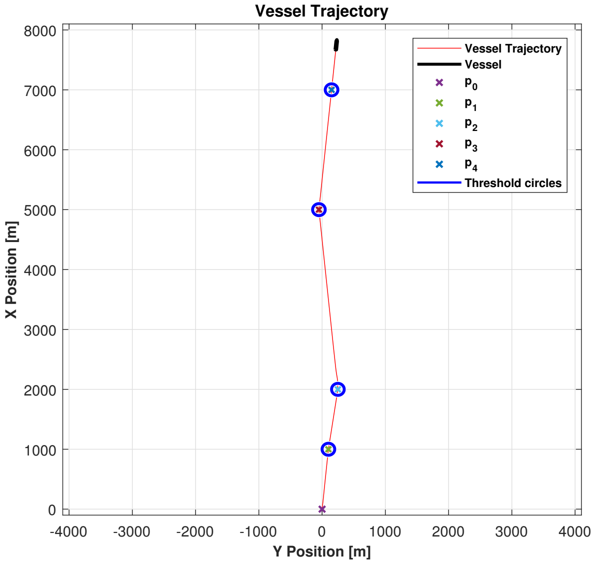


Figure 4.20: Vessel trajectory on the sea - Undisturbed case

Note that although the vessel tracks quite well the points, it is only guaranteed that the vessel will enter the threshold circles on the linear nominal MPC calculation, where the waypoint switching condition is implemented. Nonetheless, the nonlinear system is always entering the threshold circle which means that the ancillary feedback is managing to maintain the simulation points close to the predicted ones. Velocities between the waypoints so as input required are also shown below:

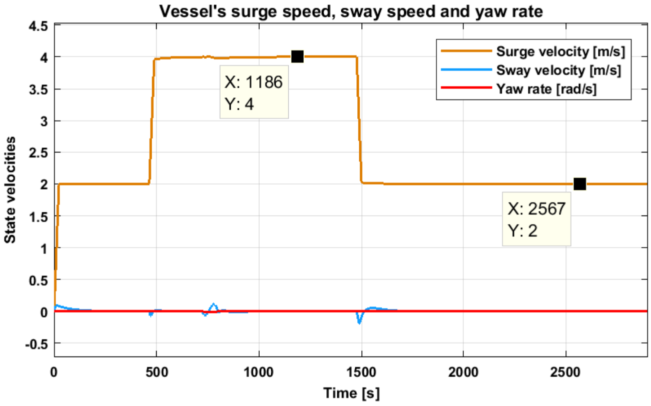
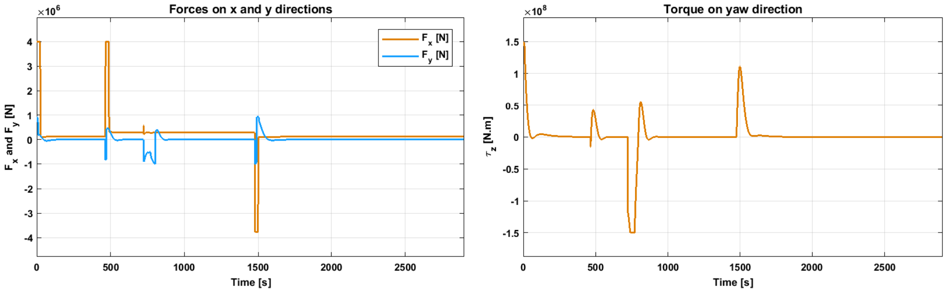


Figure 4.21: Target velocities - Undisturbed case



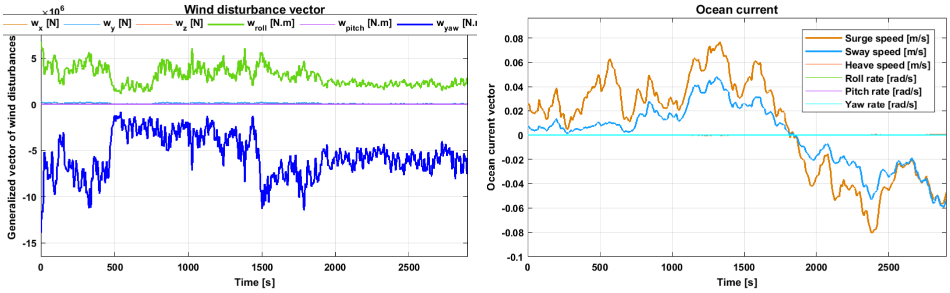
(a) Forces on x and y direction

(b) Torque on yaw direction

Figure 4.22: Input required for Line of Sight -undisturbed case

As it can be seen above, velocities are also properly tracked. Inputs are heavily demanded by this task, mainly when going from p_2 to p_3 and from p_3 to p_4, tasks on which a change in direction is required. For the LOS task a more aggressive tuning is used in the ancillary feedback, so as in the nominal MPC (mainly for heading and surge speed).

Having presented the undisturbed behavior, now we present the controller performance under disturbance. Unlike the previous simulations, here it will only be shown the system and controller behavior for the highest level of disturbance, which will include wind disturbances around 10^7 and ocean currents. Note also that although ocean currents have been chosen to be a bit smaller than on previous simulations, wind disturbance has been chosen to be a bit higher. Values can be seen below:



(a) High wind disturbance

(b) Ocean currents

Figure 4.23: External disturbance on LOS simulation

Vessel trajectory under these disturbances are as follows:

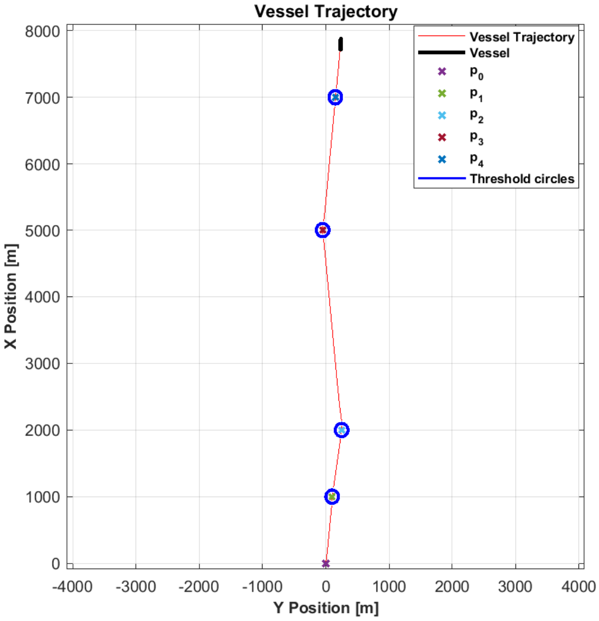


Figure 4.24: Vessel trajectory on the sea - High disturbance case

Although it may be a bit hard to see due to the scaling, the vessel now is not as accurate on tracking the waypoints but still manage to have a good tracking capability. Velocity requirements under the same disturbance can be seen below:

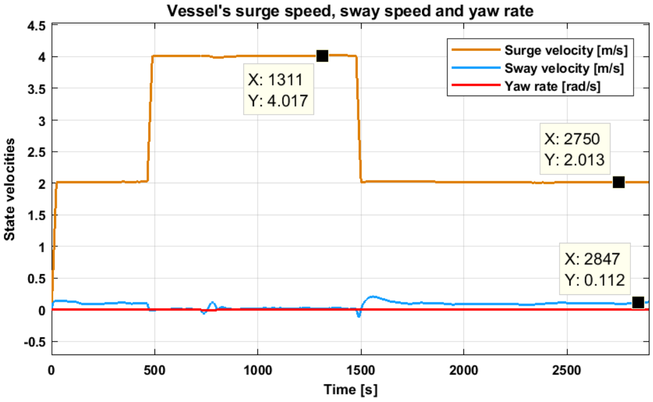
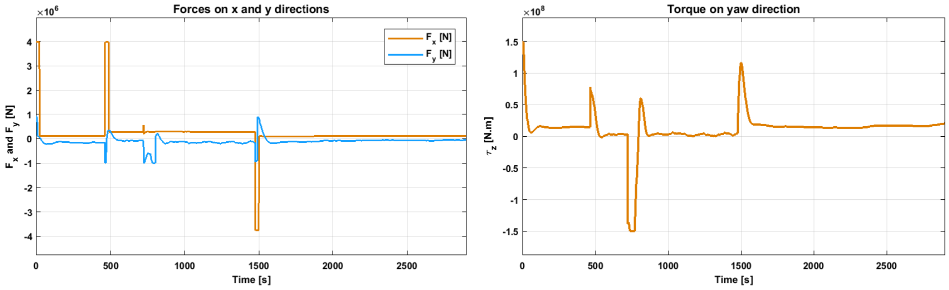


Figure 4.25: Target velocities - High disturbance case

Finally the inputs required can be seen below:



(a) Forces on x and y direction

(b) Torque on yaw direction

Figure 4.26: Input required for Line of Sight - high disturbance case

As can be seen above, velocity requirements are maintained quite close to the goal while input constraints are respected. Having presented the last result for the line of sight simulations an evaluation of the suitability of this technique will now be presented.

4.4 Technique discussion

In the previous sections the chosen tube MPC technique performance was presented by means of numerical simulations. Throughout these simulations some important remarks could be gathered about the technique:

1. High number of variables to tune can make the system hard to tune - LMI tuning + MPC tuning + terminal set design when put together encompass numerous design parameters which have sometimes direct and indirect effects on the system, increasing the overall difficulty of finding suitable values for the variables to be used during the implementation phase.

2. Iterative tuning process - Due to the model mismatch the tuning process is also iterative. As said before very high gains on the ancillary feedback controller makes the system go far from the linear model used and also may produce empty set differences. On the other hand weak ancillary feedback gains makes the system behave very poorly. In the end, the system must be designed so that set differences exist in the first place. Then it must be implemented on the vessel and the behavior must be studied and based on results maybe the controller must be re-designed. The iterative process may take some time and usually must be "taylor made" for each task when tasks differ a lot in style (like dynamic position and speed heading).

3. The technique implemented has different sources of conservativeness - Different sources of conservativeness were put "on top of each other" leading to a conservative final system. First, theoretical conditions developed for this method in [18] are sufficient conditions, meaning that it can be guaranteed that the linear system will be maintained inside the robust invariant set while conditions are respected, but it does not mean that the system will be unstable if conditions are not respected. This can be seen during simulations. Although we do not respect the maximum disturbance bound, found during the offline step, on the simulations, the system still have a good performance.

   On top of that, when using this type of tube MPC (also known as rigid tube MPC), a hidden assumption when performing the set difference offline is that the error between the nominal system and the actual system is always on the edge of this robust invariant set. This may or may not happen during the simulation. When the simulation starts for example, we have zero difference between nominal system and actual state.

   At last, whenever we are calculating set differences in MPT, even if we build a tight box around the ellipsoid by matrix decomposition, the package will construct a box using the extreme points of the bounding box, requiring an input effort which will never be used (if we are inside the robust invariant set) and as a by product generating empty set differences for larger disturbances (tuning parameters maintained), when the actual set difference is nonempty.

4. If nominal state predictions are bad system performance is ruined - For the low speed applications the linear system has been a good choice. For higher speeds, on the other hand, it may not be the best choice as mismatch may

grow even bigger making state predictions rather poor. As the technique is heavily reliant on these predictions (ancillary feedback acts using nominal predictions as its reference values), overall performance will also not be good.

5. Reduced online computational burden - The online computational burden sums up to a linear controller while the offline is a linear MPC requiring a reasonably long prediction horizon due to the terminal set computed plus set differences and LMI calculations.

6. If system is properly tuned it can work quite well - Results have shown some satisfactory behavior despite disturbances, model mismatch and tasks making such model mismatch deliberately worse as in the LOS case when direction change was required (due to waypoints) along with velocity changes.

All in all the technique can give good results, but one must bear in mind the long tuning process and hard set difference requirement that the technique impose. After this throrough discussion we may now move to the thesis conclusions.

Chapter 5

Conclusion and Future Work

In this master thesis we have approached the problem of robust vessel control. Such problem is of great interest in several different applications, from safe oil extraction to fully autonomous navigation. In order to address this problem, we started, in chapter two by briefly covering different vessel dynamics approaches. Then, also in chapter two, we covered different suitable control techniques to tackle this problem, and by the end of it we advocated by one, namely, tube MPC. In chapter three we have explained in more detail the vessel model used and the tube MPC technique utilized. In chapter 3, the technique application was explained in even more detail for the tasks chosen for this master thesis, namely: Dynamic positioning, sequential speed heading and waypoint tracking by the LOS algorithm. In chapter 4 we have presented the simulation of this method implemented on a real sized vessel subject to large disturbances and have evaluated its performance.

The tube MPC method chosen has been adapted from [18], and has shown benefits and points to be improved. During the implementation of this method several problems were faced. First, given the nature of the system, several numerical problems were found along the way, prompting a system scaling so that such issues could be overcome. The method used has also shown to be quite conservative in several different manners. First, stability conditions developed are sufficient and comparison of the offline estimation when performing set differences and the actual simulation, have indicated rather conservative conditions. Another source of conservativeness was MPT [39], the package used to perform the set differences. In first trials, MPT would have problems on the generation of ellipsoid's bounding boxes. To overcome this problem, bounding boxes generated by matrix decomposition were proposed. Nonetheless, MPT overestimates the set generated by matrix decomposition by apparently generating internally, a bounding box "connecting extreme points" of this set to perform the set difference operation.

Another source of conservativeness was the model mismatch adopted to describe the system. During some time in this project, it was attempted to use the nonlinear model of the vessel and solve the problem by a nonlinear MPC. The nonlinear approach has shown several problems such as, very long computational time and if the system was not scaled, solvers would get stuck in local minima generating a very bad input and state sequence. Although these problems could be overcame, respecting the sufficient conditions that ensure nonlinear MPC stability, while keeping the computational time to a reasonable value was very hard. Conditions found would usually demand to set one component of the state vector to zero by the end of the prediction horizon, making the terminal set rather small and enforcing the prediction horizon to be rather long. Even if all these problems were overcame, as said before, conditions developed in [18] based on Lipschitz conditions produced very low values of the Lipschitz constant, making it even harder for conditions to be respected by the system. The overall down side of not using a nominal nonlinear model is that we cannot guarantee stability for the nonlinear system, but only for the equivalent linear system. We are also "pushing" all the nonlinearity as an uncertainty of the system for the ancillary feedback to handle it.

This technique has also shown some good benefits. The choice of a linearized model around the target point, although simplistic, has shown to be a good approximation (performance wise). Another good point of choosing a linear model is that, the local minima problem disappears. The online computational cost is reduced, while offline cost includes the calculation of a linear MPC plus LMI and set differences. Simulation results with the linear system had shown that the overall system has a quite good capacity of handling disturbance and dealing with model mismatch. Furthermore, the conservativeness of the linear system chosen for control design has proven during simulation time, not to be a big issue, since simulations have depicted the system's reasonable steering capacity.

There are, nonetheless, several points to be improved in this technique. One of them, is to use the full nonlinear model to reduce the conservativeness of the approach and also reach the bigger goal of guaranteeing nonlinear system stability. Another source of conservativeness, the rigid tube calculated offline, may be substituted by some technique using Homothetic tubes [40], so that the cross-section of the tube can be updated based on online information (downside is that nominal MPC would also have to be calculated online as constraints depend on new set obtained online), or tube MPC based on contraction theory [36], [26] could be attempted as well.

All in all, this work leaves the contribution of disturbance handling by means of a robust MPC technique, presenting relatively low online computational costs and reasonable results at the expense of no overall system stability guarantee, but guarantees of stability and constraint satisfaction for the equivalent linear system.

Bibliography

[1] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.

[2] S.-R. Oh and J. Sun, "Path following of underactuated marine surface vessels using line-of-sight based model predictive control," *Ocean Engineering*, vol. 37, no. 2-3, pp. 289–295, 2010.

[3] C. Shen, Y. Shi, and B. Buckham, "Lyapunov-based model predictive control for dynamic positioning of autonomous underwater vehicles," *environment*, vol. 1, p. 2, 2017.

[4] S. Yin and B. Xiao, "Tracking control of surface ships with disturbance and uncertainties rejection capability," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 3, pp. 1154–1162, 2017.

[5] R. Apostol-Mates and A. Barbu, "Human error-the main factor in marine accidents," *Sci. Bull.'Mircea Cel. Batran'Nav. Acad*, vol. 19, no. 2, pp. 451–454, 2016.

[6] W. T. free enciclopedia. (2018) PID controller. [Online]. Available: https://en.wikipedia.org/wiki/PID_controller

[7] A. J. Sørensen, S. I. Sagatun, and T. I. Fossen, "Design of a dynamic positioning system using model-based control," 1996.

[8] M. Katebi, M. Grimble, and Y. Zhang, "H∞ robust control design for dynamic ship positioning," *IEE Proceedings-Control Theory and Applications*, vol. 144, no. 2, pp. 110–120, 1997.

[9] M. Krstic, I. Kanellakopoulos, P. V. Kokotovic *et al.*, *Nonlinear and adaptive control design*. Wiley New York, 1995, vol. 222.

[10] H. Ashrafiuon, K. R. Muske, L. C. McNinch, and R. A. Soltan, "Sliding-mode tracking control of surface vessels," *IEEE transactions on industrial electronics*, vol. 55, no. 11, pp. 4004–4012, 2008.

[11] Y.-l. Liao, L. Wan, and J.-y. Zhuang, “Backstepping dynamical sliding mode control method for the path following of the underactuated surface vessel,” *Procedia Engineering*, vol. 15, pp. 256–263, 2011.

[12] T. I. Fossen and M. J. Paulsen, “Adaptive feedback linearization applied to steering of ships,” in *Control Applications, 1992., First IEEE Conference on.* IEEE, 1992, pp. 1088–1093.

[13] R. Stelzer and K. Jafarmadar, “History and recent developments in robotic sailing,” in *Robotic sailing.* Springer, 2011, pp. 3–23.

[14] S.-L. Dai, M. Wang, and C. Wang, “Neural learning control of marine surface vessels with guaranteed transient tracking performance,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 3, pp. 1717–1727, 2016.

[15] A. Loria, T. I. Fossen, and E. Panteley, “A separation principle for dynamic positioning of ships: Theoretical and experimental results,” *IEEE Transactions on Control Systems Technology*, vol. 8, no. 2, pp. 332–343, 2000.

[16] J. Du, X. Hu, M. Krstić, and Y. Sun, “Robust dynamic positioning of ships with disturbances under input saturation,” *Automatica*, vol. 73, pp. 207–214, 2016.

[17] J. Wang and Z. Zou, “A nonlinear heading controller based on eso for a ship sailing in restricted waters,” in *Control and Decision Conference (CCDC), 2016 Chinese.* IEEE, 2016, pp. 6443–6448.

[18] S. Yu, C. Maier, H. Chen, and F. Allgöwer, “Tube mpc scheme based on robust control invariant set with application to lipschitz nonlinear systems,” *Systems & Control Letters*, vol. 62, no. 2, pp. 194–200, 2013.

[19] J. B. Rawlings and D. Q. Mayne, “Model predictive control: Theory and design,” 2009.

[20] A. Veksler, T. A. Johansen, F. Borrelli, and B. Realfsen, “Dynamic positioning with model predictive control,” *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1340–1353, 2016.

[21] Z. Yan and J. Wang, “Model predictive control for tracking of underactuated vessels based on recurrent neural networks,” *IEEE Journal of Oceanic Engineering*, vol. 37, no. 4, pp. 717–726, 2012.

[22] Z. Sun, Y. Xia, L. Dai, K. Liu, and D. Ma, “Disturbance rejection mpc for tracking of wheeled mobile robot,” *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 6, pp. 2576–2587, 2017.

[23] D. Ramirez, T. Alamo, and E. Camacho, “Computationally efficient min-max mpc,” *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 462–467, 2005.

[24] J. Gruber, D. Ramirez, T. Alamo, and E. Camacho, "Min–max mpc based on an upper bound of the worst case cost with guaranteed stability. application to a pilot plant," *Journal of Process Control*, vol. 21, no. 1, pp. 194–204, 2011.

[25] Y. Gao, A. Gray, A. Carvalho, H. E. Tseng, and F. Borrelli, "Robust nonlinear predictive control for semiautonomous ground vehicles," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 4913–4918.

[26] X. Liu, Y. Shi, and D. Constantinescu, "Robust constrained model predictive control using contraction theory," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014, pp. 3536–3541.

[27] T. Perez and T. I. Fossen, "Kinematic models for manoeuvring and seakeeping of marine vessels," 2007.

[28] R. Skjetne, Ø. Smogeli, and T. I. Fossen, "Modeling, identification, and adaptive maneuvering of cybership ii: A complete design with experiments," in *Proc. IFAC Conf. on Control Applications in Marine Systems*, 2004, pp. 203–208.

[29] T. I. Fossen and J. P. Strand, "Nonlinear passive weather optimal positioning control (wopc) system for ships and rigs: experimental results," *Automatica*, vol. 37, no. 5, pp. 701–715, 2001.

[30] Y. Yang, J. Du, H. Liu, C. Guo, and A. Abraham, "A trajectory tracking robust controller of surface vessels with disturbance uncertainties," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1511–1518, 2014.

[31] Y. Qu, B. Xiao, Z. Fu, and D. Yuan, "Trajectory exponential tracking control of unmanned surface ships with external disturbance and system uncertainties," *ISA transactions*, 2018.

[32] A. Bemporad, F. Borrelli, and M. Morari, "Min-max control of constrained uncertain discrete-time linear systems," *IEEE Transactions on automatic control*, vol. 48, no. 9, pp. 1600–1606, 2003.

[33] C. Liu, H. Li, J. Gao, and D. Xu, "Robust self-triggered min–max model predictive control for discrete-time nonlinear systems," *Automatica*, vol. 89, pp. 333–339, 2018.

[34] D. Q. Mayne and E. C. Kerrigan, "Tube-based robust nonlinear model predictive control," in *Proc. of the 7th IFAC Symposium on Nonlinear Control Systems*, 2007, pp. 110–115.

[35] W. Lohmiller and J.-J. E. Slotine, "On contraction analysis for non-linear systems," *Automatica*, vol. 34, no. 6, pp. 683–696, 1998.

[36] S. Singh, A. Majumdar, J.-J. Slotine, and M. Pavone, "Robust online motion planning via contraction theory and convex optimization," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5883–5890.

[37] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *The International Journal of Robotics Research*, vol. 36, no. 8, pp. 947–982, 2017.

[38] Y. V. Pant, H. Abbas, and R. Mangharam, "Robust model predictive control for non-linear systems with input and state constraints via feedback linearization," in *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, 2016, pp. 5694–5699.

[39] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari, "Multi-parametric toolbox 3.0," in *Control Conference (ECC), 2013 European*. IEEE, 2013, pp. 502–510.

[40] M. Cannon and B. Kouvaritakis, "Model predictive control-classical, robust and stochastic," 2016.

[41] R. Findeisen, F. Allgöwer, and L. T. Biegler, *Assessment and future directions of nonlinear model predictive control*. Springer, 2007, vol. 358.

[42] Y. J. Leung, W. V. Li *et al.*, "Optimal ellipsoids and decomposition of positive definite matrices," *Journal of Mathematical Analysis and Applications*, vol. 331, no. 2, pp. 1452–1466, 2007.

[43] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in matlab," in *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[44] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.

[45] S. S. Ge, C. Y. Sang, and B. V. E. How, "Dynamic positioning system for marine vessels," *The Impact of Control Technology*, 2011.

[46] O. E. Education. (2016) Introduction to dynamic positioning. [Online]. Available: http://www.offshoreengineering.com/education/dynamic-positioning-dp/what-is-dynamic-positioning

[47] J. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11–12, pp. 625–653, 1999, version 1.05 available from `http://fewcal.kub.nl/sturm`.

[48] K.-C. Toh, M. J. Todd, and R. H. Tütüncü, "Sdpt3-a matlab software package for semidefinite programming, version 1.3," *Optimization methods and software*, vol. 11, no. 1-4, pp. 545–581, 1999.

[49] R. H. Tütüncü, K.-C. Toh, and M. J. Todd, "Solving semidefinite-quadratic-linear programs using sdpt3," *Mathematical programming*, vol. 95, no. 2, pp. 189–217, 2003.

[50] T. I. Fossen, M. Breivik, and R. Skjetne, "Line-of-sight path following of underactuated marine craft," *IFAC Proceedings Volumes*, vol. 36, no. 21, pp. 211–216, 2003.

TRITA-EECS-EX-2018:689
www.kth.se