Robust Model Based Control of Constrained Systems

by

Reza Ghaemi

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering - Systems)
in The University of Michigan
2010

Doctoral Committee:

Professor Jing Sun, Co-Chair
Professor Ilya Vladimir Kolmanovsky, Co-Chair
Professor Jessy W. Grizzle
Professor N. Harris McClamroch
Assistant Professor Domitilla Del Vecchio

To my parents, Sadreddin and Zohreh, for their endless love and patience.

Acknowledgements

I would like to express my gratitude to the people who made this thesis possible. First I owe a great debt of gratitude to my advisors, Professor Jing Sun and Professor Ilya Vladimir Kolmanovsky , who have patiently mentored me to think independently and given me liberty so that I could explore new ideas and apply them creatively. Their superb knowledge of control theory and practice contributed tremendously in theoretical and practical contributions in this research. Their tolerance with my naivety and their confidence in my abilities as a researcher exemplifies their leadership and will live with me as an asset forever.

I am extremely thankful to my dissertation committee member Professor Jessy Grizzle for his support throughout my PhD studies, his inspirational remarks and valuable feedbacks on my research and his outstanding teaching which has equipped me with important fundamental knowledge relating to nonlinear control and functional analysis, and introducing me to Professor Sun which marked the beginning of my research at the University of Michigan. I would also like to thank my dissertation committee members, Professor Harris McClamroch and Professor Domitilla Del Vecchio, for their help. I regret not availing myself more fully to their expertise. Furthermore, I enjoyed collaboration with Professor Del Vecchio which started from a term project and, thanks to her knowledge and experience, exposed me to the new research area of system biology. I would also like to give my grateful and sincere thanks to all my friends and colleagues in the RACE Lab at the University of Michigan for stimulating discussions and enjoyable friendship. Particularly, I would like to thank Mahdi Sadeghi, Gayathri Seenumani, Soryeok Oh, Yanhui Xie, Handa Xi, Christopher Vermillion, Vasilios Tsourapas, Amey Karnik, Jian Chen, and Jacob Faust.

Finally, I would like to thank my parents for their continues support and constant encouragement during this endeavor.

Table of Contents

List of Figures

List of Tables

List of Appendices

Abstract

This dissertation is concerned with control of systems subject to input and state constraints. Model Predictive Control (MPC) is one promising control technique that is capable of dealing with constraints. Its flexible formulation also provides mechanisms to tune the closed loop system for desired performance. However, due to computational complexity and its dependency on accurate models of the system, the MPC applications for systems with fast dynamics or with model uncertainties are not wide spread. The focus of this dissertation is to develop methodologies and tools that can enhance the computational efficiency and address robustness issues of constrained dynamic systems. The core contribution of this dissertation is that it provides a computational efficient MPC solver, referred to as InPA-SQP (Integrated Perturbation Analysis and Sequential Quadratic Programming).

The main results include four major components. First, a neighboring extremal control method is proposed for discrete-time optimal control problems subject to a general class of inequality constraints. A closed form solution for the neighboring extremal (NE) control is provided and a sufficient condition for existence of the neighboring extremal solution is specified. Second, the NE method is integrated with sequential quadratic programming that leads to InPA-SQP. Third, a robust control method is introduced for linear discrete-time systems subject to mixed input-state constraints. Unlike conventional MPC, the method does not require repeatedly solving an optimization problem online while guarantees states convergence to a minimal invariant set. Fourth, it is shown that if the dynamics of disturbances are incorporated, the attractor set associated with the proposed constrained robust control methods can be considerably smaller, leading to a much less conservative design.

Applications of the InPA-SQP and proposed constrained robust control constitute the other key element of the study. The InPA-SQP is employed in two experimental applications: one for voltage regulation of a DC/DC converter and another for path following of a model ship. Both applications show effectiveness of the method in terms of computation and constraints handling. These applications not only serve as validation platforms but also motivate new research topics for further investigation.

Chapter 1

Introduction

This dissertation is concerned with control of systems subject to input and state constraints. Most systems are subject to constraints due to physical and operational limitations such as actuators saturation. The control of these constrained dynamical systems has been a subject of research for decades. Conventional control systems are designed such that systems operate conservatively away from boundaries of constraints while prominent properties such as stability are attained.

Another approach which has increasingly become popular, is to use available dynamic model to predict system behavior as a function of control variations and choose the control action that produces the best behavior. This idea, which involves an optimization in each control decision making, led to Model Predictive Control (MPC), also known as Receding Horizon Control technology. MPC has been primarily used in the petro-chemical and process control industries [3]. In these industries, the operating points are obtained by solving linear programs and due to economic considerations they are required to be on the boundary of feasibility. This made MPC quite attractive.

However, due to computational complexity and its dependency on accurate models of the system, the MPC applications for systems with fast dynamics or with model uncertainties are not wide spread. With the advent of faster and cheaper computers, it was expected that this technology can be used beyond process control. The focus of this dissertation is to develop methodologies and tools that can enhance the computational efficiency of MPC or other optimization-based control strategies and address robustness issues of constrained dynamic systems.

1.1 Background

One of the most prominent optimization-based control methods is MPC [1], while other forms of optimization-based control, such as a reference governor [2], have also been developed to deal with constrained dynamic systems. In MPC, a control sequence is determined at every sampling time instant to minimize a specified cost function defined for a discrete-time system model, then the first element of the optimal sequence is used as the control action. The cost function is a tool for MPC to achieve a desired performance. The accuracy of the model also affects the performance as well as constraints satisfaction.

The system to be controlled is usually described, or approximated, by an ordinary differential equation but, since the control is normally piecewise constant, is usually modelled, in the MPC literature, by a difference equation:

$$x(k+1) = f(x(k), u(k)), \tag{1.1}$$

where $x \in \mathbb{R}^n$ is the state vector and $u \in \mathbb{R}^m$ is the input vector. $u(\cdot)$ or $\mathbf{u}$ is employed to denote a control sequence and $x^{\mathbf{u}}(\cdot; x)$ denotes a state trajectory resulting from the initial state x and control sequence $\mathbf{u}$. The control and state must satisfy the constraint $(u(k), x(k)) \in \Omega \subset \mathbb{R}^{m+n}$. For the observed state $x(k)$, the cost is defined by

$$J(x(k), \mathbf{u}) := \sum_{i=k}^{k+N-1} L(x^{\mathbf{u}}(i; x(k)), u(i)) + \Phi(x^{\mathbf{u}}(N+k; x(k))), \tag{1.2}$$

where $\mathbf{u} = \{u(k), u(k+1), \cdots, u(k+N-1)\}$, N is length of the prediction horizon, and $L(x, u)$ and $\Phi(x)$ are non-negative functions of (x, u) and x, respectively. A terminal constraint $x^{\mathbf{u}}(N+k; x(k)) \in \mathbb{X}_f$ is sometimes imposed to guarantee stability of the system. At time instant k, the state $x(k)$ is observed and the optimization problem $\mathscr{P}(x(k))$ defined by

$$\begin{aligned} &\mathscr{P}(x(k)) : J^*(x(k)) = \min_{\mathbf{u}} J(x(k), \mathbf{u}) \\ &\text{subject to the dynamic equation (1.1)} \\ &\text{and constraints } (u(i), x^{\mathbf{u}}(i; x(k))) \in \Omega \end{aligned} \tag{1.3}$$

is typically solved numerically to obtain the, not necessarily unique, optimal control sequence $\mathbf{u}^*$. The first element of $\mathbf{u}^*$ is applied to the plant until new measurements

become available at the next sampling time instant at which point the optimization is repeated.

For simplicity, the control, prediction and constraint horizons in (1.3) are all assumed to be equal to N. There are many variations of the problem (1.3) which can help reduce the computational effort or improve performance. For instance, a control horizon within the prediction horizon can be defined to reduce the optimization dimension by assuming u is constant beyond the control horizon [4]. The horizon over which the constraints are enforced can also be different from the control horizon. Reference [5] proposes the block MPC that uses a subsequence of optimizing control inputs to achieve stabilization and to reduce the frequency of optimization.

The MPC framework can explicitly address constraints as the constraints can be easily incorporated in the on-line optimization problem. In addition, it provides a flexible mechanism for shaping the transient response by adjusting the weights in the cost function and for handling hybrid/switching dynamic systems as well as reconfigurable control applications. For instance, failures or system changes can be handled by MPC with relative ease as long as these changes are reflected in the model used during the on-line optimization.

However, there are major challenges associated with MPC such as stability, computational efficiency, robustness, and imperfect measurement that are discussed in the sequel.

1.1.1 Stability of MPC

The standard MPC formulation (1.3) for system (1.1) may not always lead to a stable closed-loop system. In fact, it is well-known that "optimality in the MPC framework does not imply stability" and the issue of ensuring stability has been long recognized as fundamental. Starting from the work by Keerthi and Gilbert [6], considerable progress has been made in the last 20 years, and many different algorithms and mechanisms have been proposed to assure closed-loop stability. See, for instance, the book [7]. In fact, the theory of stability of model predictive control has reached a relatively mature stage, as elaborated in the 2000 survey paper by Mayne et. al. [1].

There are several mechanisms for guaranteeing stability in MPC. They include extending the prediction horizon, incorporating an appropriately defined terminal cost, imposing constraints on the final state at the end of the prediction horizon and imposing an artificial condition on a positive-definite function of system terminal states. The assumption of the existence of a local stabilizing controller, whose function

is to stabilize the system within a neighborhood of the origin without violating the constraints, has played a prominent role in the stability treatments of MPC. The value function is almost universally employed as a Lyapunov function for stability analysis of MPC. In [1], seeking to unify the existing results, four axioms are formulated as sufficient conditions for MPC closed-loop system stability. Various modifications of MPC aimed at guaranteeing stability are shown to differ only in their choice of terminal cost, terminal constraint set and local stabilizing controller.

It is interesting to note that in practical applications of MPC, the above mechanisms for guaranteeing stability are not always used because they may limit the performance of the system. An a posteriori check of stability via either simulations or construction of Lyapunov functions may be used if the above mechanisms for guaranteing stability are not incorporated a priori.

Another critical issue in practical applications of MPC is feasibility in tracking or disturbance rejection. Final constraint sets or sufficiently long length of prediction horizon are sometimes required, especially if state constraints are involved.

1.1.2 Computational considerations in MPC

Parallel to the stability research, much effort has been dedicated to improving the computational aspects of MPC, with the goal to broaden the range of its applications to systems with fast dynamics and limited computing capability.

Several different venues have been explored. Some of the basic ideas involved are briefly described below.

Model complexity reduction

Several schemes aimed at reducing computational time can be found in the literature, including model reduction techniques [8, 9, 10, 11].

Model complexity reduction can dramatically reduce the required computing resources. The reduced order models used for MPC can be developed through various mechanisms. For example, the required computing resources can be dramatically reduced by leveraging the time scale decomposition of dynamics of different components and focusing just on dominant dynamics. See, for instance, [12, 13]. Some methods have achieved a reduction in computation by using regional linear approximation of nonlinear models [14, 16].

Another idea involves approximation of a nonlinear model with a linear model (either a time-invariant linear model or a time-varying linear model obtained by linearizing the nonlinear system around the solution trajectory to the MPC problem from the previous time instant), of approximation of the cost with a quadratic function, and approximation of the constraints by linear constraints. By tightening the constraints to appropriately account for the differences between the linear and nonlinear model, feasibility can be assured, while the optimization reduces to solving a quadratic programming problem.

However, model reduction leads to model uncertainties which might reduce performance or potentially cause instability.

Explicit MPC

For linear and piecewise affine linear systems with 1-, 2- or ∞- norm based costs and affine constraints, an explicit solution to the MPC problem can be generated off-line by using multi-parameteric quadratic programming and linear programming solvers [17, 18, 19]. The explicit solution has the form of a set of polyhedral regions and in each region the control is an affine function of the state. Once such a solution is computed off-line, the on-line computations reduce to finding the polyhedral region to which the state belongs and computing the affine control function defined for that region. When applied to nonlinear systems, this method requires approximation of nonlinearities by piecewise affine functions. Recent implementation of the piecewise affine system identification toolbox (PWAID) [20, 21] greatly facilitates the application of this method. The state dimension and control horizon limit the applicability of this method due to the rapid growth in the number of the polyhedral regions as these parameters increase. The closed loop system using an explicit MPC law is amenable to an a posteriori stability check using LMI techniques [22]. However, for the approach with explicit solution, polyhedral partitions grow exponentially with respect to the length of horizon which causes problems when long horizon is required for stability.

Reformulation to a lower dimensional optimization problem

By reformulating the MPC as a lower dimensional optimization problem, the computational complexity can be reduced thereby making the on-line solution more computationally feasible. The reference governor is a representative example along these

lines, where a one-dimensional optimization is typically employed, together with the maximum admissible set concept, to provide both stability and constraint enforcement properties for the closed loop system [23, 24, 2]. The parameter governor approaches proposed in [25] have a similar flavor.

Decentralized and hierarchical MPC

With a decentralized or hierarchical implementation of MPC, the treatment of large scale MPC problems may become feasible as an original, large size, optimization problem can be reduced to a set of smaller and more tractable optimization problems. In multi-agent systems, the agents can solve these smaller optimization problems in parallel. The work in this direction includes references [26, 27, 28, 29]. It is interesting that the treatment of hierarchical MPC problems in which a higher level MPC controller calculates set-points for lower level MPC controllers have received less attentions, even though a higher level MPC solution may be used by the lower level MPC controllers as a preview. For recent work in this direction, see [30].

Special purpose computing hardware

Special purpose hardware solutions may play a role in the future implementation of MPC algorithms. The work in this direction includes [31, 32].

1.1.3 Online optimization

In many applications, due to nonlinearity of the system and the need for a long length of horizon for stability and enhanced performance, on-line implementation of MPC is the best, if not the only, choice.

In MPC algorithms based on online optimization, Quadratic Programming (QP) problems often arise as subproblems during the iterative nonlinear solution procedure, so that several QPs may need to be solved in each sampling time. In most MPC algorithms, the arising QPs are treated using well tested and efficient standard methods for optimization. When sampling times become so short that the computation times for QP solutions can no longer be neglected, specialized algorithms that exploit the structure of the QPs become an interesting alternative to standard QP solvers. Interior Point Method [33] is a category of approaches to solve the QPs associated with

MPC [34]. The drawback of the method is that, so far, no efficient warm start, i.e., suitable initial guess, exists for implementation of MPC at each sample time.

On the other hand, approximating the solution of the MPC optimization problem using a pre-computed nominal optimal solution, i.e., an optimal solution corresponding to a fixed initial state, can also reduce the on-line MPC computational requirements. If the current state is sufficiently close to the initial state associated with the nominal solution, the optimal solution corresponding to the current state can be approximated. The nominal solution can be pre-computed off-line for different regions, or it can be computed online between two sample instants, using state predictions [35], [36]. The idea of using the previous solution as the warm start for solving QPs arising in MPC is introduced in [37] where QPs are solved based on an active set method. In MPC strategy, the previous observed state and current state are close for short sampling times. Therefore, the idea of using previous optimal solution to compute current one can be interpreted as calculating the perturbation in an optimal solution, once the initial state is perturbed. The approximation of such a perturbation can follow the well known Neighboring Extremal method, which was developed in the 1960's. Given an optimal control problem and an optimal solution with a nominal initial state, the Neighboring Extremal (NE) method provides a closed form first order approximation to the optimal solution corresponding to an initial state perturbed from the nominal value. MPC is one of many applications where knowing a nominal optimal solution, it is desired to calculate the NE solution.

The neighboring extremal solution for unconstrained continuous-time systems is presented in several papers [38, 39], [40], [41] and [42], while its discrete-time counterpart can be found in [43, 44, 45]. Subsequently, the NE solution for continuous-time systems with inequality constraints and discontinuities has been derived using multipoint boundary value techniques, as presented in [40].

For discrete-time systems subject to constraints, the finite horizon optimal control problem, in general, can be reformulated as a nonlinear programming problem. Consequently, exploiting sensitivity analysis for the nonlinear programming problem, the NE solution can be calculated as shown in [46]. If N denotes the length of the horizon in the optimization problem, the computational complexity of the corresponding nonlinear programming problem and of the method in [46] is of the order N^3. A constrained NE method, i.e., extension of the existing NE method to constrained dynamic optimization problems, is needed to calculate first order approximation of optimal solution of order of N. The constrained NE can be employed in MPC optimization to provide a computationally efficient MPC solver. Moreover, the constrained NE

method can be used in applications where a pre-computed nominal optimal solution needs to be modified online due to the presence of state disturbances.

1.1.4 Robust MPC

It is important for MPC strategy to maintain stability and constraint compliance in the presence of model uncertainties and disturbances. With regard to stability, it is shown experimentally that MPC is robust with respect to some degree of model mismatch. Moreover, it is shown theoretically that MPC is inherently robust with respect to special class of model uncertainties, such as small gain perturbations [15]. Even though it is shown that MPC is inherently robust to sufficiently small disturbances, for general disturbances stability is not guaranteed. To guarantee stability, open-loop input control sequence MPC strategies, proposed in [16], in which the control action is taken as the first element of an optimal control sequence, may result in a small (or even empty) domain of attraction in the presence of disturbances. As a remedy, a min-max optimization over open loop control trajectories or feedback strategies is proposed [47]. However, optimization over arbitrary feedback policies, in the presence of constraints, may be especially difficult. Therefore, this problem is considered as open for general systems. A special, but important, class of systems are linear systems subject to polyhedral constraints on states and inputs and bounded additive state disturbances. The robust control of this class of systems has been studied employing reference governors [24, 48] and model predictive controllers [1, 49] with guaranteed stability and convergence properties.

Robust MPC for this class of systems is based on the idea of assuring robustness of the resulting controlled system by tightening the constraints on states and controls over the prediction horizon. This was proposed initially in [50] as well as in [51, 52, 53, 54]. The key idea is to retain a suitable margin from constraint violation over the prediction horizon so that feasibility is guaranteed for the future iterations, in the presence of allowable disturbances. In the framework of the constraint tightening approach, some MPC strategies, which were focused on affine feedback policies, were employed where the state feedback gain(s) are calculated off-line and optimization was performed over constant terms [52, 55, 56].

1.1.5 MPC with output feedback

MPC strategy generally relies on full state information [1]. However, in practical applications, perfect information of the state is not available. For linear systems with no disturbance, an observer and a linear controller provide closed-loop stability based on separation principle. However, in the presence of disturbances and constraints, combination of an observer and the predictive controller does not necessarily guarantees stability. This problem is dealt with by involving the state estimator error in the design of the predictive controller. Assuming that the estimator error is bounded, control of linear system with unknown states can be transformed to model predictive control of a linear system subject to bounded additive disturbance [57]. Control of the reformulated system is widely addressed in the robust MPC literature as discussed in Section 1.1.4. Hence, the same techniques are used for control of constrained systems with imperfect state information.

1.2 Contributions

This dissertation deals with approaches to improve efficiency and robustness of MPC. For computational efficiency, the constrained NE method is developed and is integrated with SQP to provide integrated perturbation analysis and SQP (InPA-SQP). The method is employed in two experimental applications that demonstrate its effectiveness. Even though the constrained NE method was developed for efficient MPC implementation, it has a broader formulation and application in approximating a perturbed optimal solution [58].

Regarding robustness of MPC, it is shown that without repeatedly solving an optimization problem, stability and successive feasibility in the presence of additive disturbance can be achieved for a class of linear systems. Moreover, it is shown that if the dynamics of disturbances are taken into account, considerably less conservative robust controllers can be designed. In the following subsections, the highlighted contributions are briefly reviewed.

1.2.1 Constrained Neighboring Extremal Method and InPA-SQP

As a step towards a computationally efficient MPC algorithm, a NE method is developed in this dissertation for discrete-time systems subject to general inequality constraints on inputs and states with the computational complexity of order N as opposed to N^3 [59]. Moreover, a second order sufficient optimality condition for the nominal optimal solution is provided that is computationally verifiable of order N (including the case where linearization is performed on-line). This is the same condition for existence of NE solution and for local convexity of the nonlinear dynamic optimization in the vicinity of the nominal optimal solution. The NE approach has application beyond efficient MPC implementation. It can be employed in any circumstance where an optimal solution is available a priori and the optimal solution corresponding to new initial state needs to be approximated, see [40, 60]. The developed NE method (or perturbation analysis) is combined with sequential quadratic programming (SQP) with active set method to provide Integrated Perturbation Analysis and Sequential Quadratic Programming (InPA-SQP) approach, for the MPC implementation [61]. It synergistically combines the solutions derived using perturbation analysis and SQP to solve the optimization problem with an initial state perturbation and input/state constraints. Numerical examples and simulation results are provided in this dissertation that show effectiveness of the InPA-SQP method. InPA-SQP is based on the same idea as introduced in [37], [62] where, based on active set method, anominal optimal solution is used to calculate the current MPC optimal solution. However, thanks to the developed constrained NE method, as mentioned above, it enjoys the following properties which are lacking in other active set based strategies:

- InPA-SQP provides an explicit solution for QPs with equality constraints associated with active set method iteration. InPA-SQP has computational complexity of order N.

- The second order sufficient conditions derived in the dissertation can be verified to order N, making it possible to check optimality of the solutions calculated by any optimization solver.

- The necessary and sufficient conditions for degeneracy of the QPs arising in MPC are determined.

It is important to note that the constrained NE method provides second order sufficient optimality conditions (SOSC) for discrete-time dynamic optimization prob-

lems. All dynamic optimization problem solvers are based on first order necessary conditions (KKT conditions) and therefore once the solution converges it is not guaranteed that the solution is actually optimal.

The computational complexity being of order N, is the property that Interior Point based MPC solvers also posses. However, interior point methods, as opposed to SQP, lack the advantage of warm start using the previous optimal solution as previously mentioned. That is why sequential quadratic programming based on active set method is better suited for solving MPC optimization problems.

1.2.2 InPA-SQP implementation and Experimental Results

Another important contribution in this dissertation is the application of the MPC, and InPA-SQP algorithms. InPA-SQP is employed in two experimental applications. As the first application, MPC is implemented via the proposed InPA-SQP to regulate the output voltage of a DC/DC converter with peak current protection. A DC/DC converter has very fast dynamics and therefore requires an efficient MPC implementation algorithm to achieve sub-millisecond sampling time. The full bridge DC/DC Converter was initially proposed in previous studies [63], [64] for both high power density and high power applications. It is very attractive because of its zero voltage switching, low component stresses, and high power density features [65], [66]. Moreover, its high frequency transformer prevents fault propagation and enables a high output/input voltage ratio. Therefore, with a full bridge DC/DC converter as the power conditioning system, low voltage energy systems can be applied to high DC voltage applications, such as the DC zonal electrical distribution system of an all electric ship [67]. To investigate the voltage regulation of a full bridge DC/DC based power conditioning system, an experimental testbed was developed at the University of Michigan to support model development and to facilitate a model based control design approach [68]. The voltage regulation problem is formulated as an MPC problem using a nonlinear model to predict the future plant behavior. The peak current protection requirement is formulated as a nonlinear constraint. To achieve 300 μ s sampling time and handle the nonlinear constraint, the InPA-SQP method is employed to solve the constrained optimal control problem. The InPA-SQP solver can significantly improve computational efficiency while effectively handling the nonlinear constraints, making real-time implementation of MPC feasible for a power electronics systems with fast dynamics. The experimental results reveal that the NMPC algorithms successfully achieve voltage regulation and peak current protection. The

details about the DC/DC converter test bed, and implementation of InPA-SQP will be explained [69].

As the second application, InPA-SQP is also used for control of a model ship in Marine Hydro-Dynamic Lab (MHL) to follow a pre-specified path. InPA-SQP is used for the path following problem for a model ship via rudder control. 3-degree-of-freedom simplified nonlinear and linear models are adopted in the controller design and a corresponding 6-degree-of-freedom nonlinear container ship model is used in simulations in order to study and compare the performance of the MPC using the linear and nonlinear models. The InPA-SQP algorithm is used to implement both the linear and nonlinear MPC on a model ship to experimentally validate the algorithm and compare the performance. Experimental results show the effectiveness of the proposed MPC solver [70].

1.2.3 Robust MPC

To achieve successive feasibility and stability, for constrained linear dynamic systems, it turns out that it is not necessary to solve repeatedly an on-line optimization problem. In this dissertation, a robust control method is introduced for linear discrete-time systems subject to mixed input-state constraints. The proposed scheme, which is based on the constraint tightening approach [51]-[54], has several special features. First, unlike robust MPC approaches, our proposed method does not involve repeated online optimization to determine the control action. Second, under appropriate and easily verifiable conditions, the proposed controller guarantees feasibility. Third, the minimal invariant set corresponding to the off-line calculated state feedback is an attractor, i.e., all trajectories converge to this set. Fourth, our approach does not require the terminal constraint set to be contained in the desired target set, which is a typical assumption made in the prior literature, except for [53]. In fact, the terminal constraint set, namely the set to which the final predicted state must belong, can be much larger than the target set. Finite-time convergence to the target set is guaranteed as long as the target set contains the minimal invariant set. Moreover, our method requires no explicit knowledge of the minimal invariant set.

As an example to illustrate the applications of the proposed algorithm, the roll control problem for a high speed ship equipped with stabilizing fins is considered in this dissertation. Control of the roll motion of ships has been extensively considered in the literature [71]-[73]. As elaborated in [72], large roll motions induced by ocean waves can severely affect the safety and performance of surface ships. To reduce

the roll motion, different devices have been developed, including the so called "fin stabilizer" used for high speed ships [74, 75]. The fin stabilizer reduces roll motion by controlling the mechanical angle of the fin according to the ship roll angle and roll rate. However, the fin stabilizer is only effective in reducing the roll motion within a certain range of operating conditions determined by the ship's state and input variables. We show that the roll motion of the ship in the presence of wave disturbances can be stabilized using the proposed algorithm while the input-state constraints and input saturation constraints can be effectively enforced and the domain of recoverable initial ship states is large [76, 77].

1.2.4 Minimal attractor sets

In robust control strategies, concerned with control of constrained linear systems subject to additive disturbances, minimal robust positive invariant sets (mRPI) are important for synthesis of controllers for uncertain systems and for computing maximal robust positive invariant sets [78, 79]. The set mRPI is defined as the set of states that can be reached from the origin under a bounded state disturbance. The mPRI sets are attractors in robust MPC with constraints tightening approach [52] and are essential in the synthesis of tube MPC [53]. Since computation of mPRI sets is prohibitive, the characterization and the computation of approximations of mPRI sets have been considered (see [80] and references therein).

In all of the aforementioned approaches, it is assumed that disturbances are confined to a given compact set and, at any time instant, allowed to take arbitrary values within the set. However, this assumption may lead to conservative results in the case where the disturbance dynamics are known or can be estimated. A special and prevailing case is when the disturbances are generated by physical processes and are inherently rate limited. Another case is when disturbances can be modelled as an output of a dynamic system driven by a set-bounded signal. Finally, the disturbances may represent the effects of omitted nonlinearities of dynamic systems and their bounds may be state-dependent. Rate-bounded additive disturbances, as special cases of general disturbance dynamics, are considered in [81] where the rate bound on the disturbance is used to calculate an approximation to maximal control admissible set. Note that this approximation set is less conservative than in the case when no rate bound is assumed. Since the disturbance at each time instance is dependent on previous values, it is expected that incorporating the previous values of disturbance in the controller may provide better control performance including smaller minimal

invariant set and larger basin of attraction.

Considering the dynamics of the system and disturbances, the invariant set analysis is concerned with the augmented system of state and disturbance where the disturbance is confined in a set which is dependent on the augmented state. It turns out that if the set containing disturbances is state dependent, then the minimal invariant set may not be an attractor and therefore can not be used in robust control synthesis and analysis. Hence, the notion of minimal robust invariant attractor (MRIA) is introduced in this dissertation which is used in robust control design.

The MRIA set for linear systems subject to additive disturbances confined in a state-dependent and bounded set are analyzed and characterized. In particular, existence of a MRIA set is proved and the set is characterized when the state dependent set is upper-semi continuous. Moreover, built on such characterization, the existence of a minimal attracting invariant set is established for the case when disturbances evolve within a compact set according to a linear dynamic model. The MRIA set is smaller if the disturbance model is used in comparison to the case where only the boundedness of the additive disturbance is assumed. A numerical example is provided that shows the size of the minimal invariant attracting set is considerably different in the two cases. Furthermore, we have shown that the MRIA set can be employed in the design of robust MPC strategies, such as tube MPC [53], to achieve robust stability, improve control response and to reduce conservativeness.

1.3 Dissertation Outline

This dissertation is organized as follows:

- In Chapter 2 the constrained NE method and an efficient MPC solver, referred to as InPA-SQP, are introduced. Two numerical examples are provided in Chapter 2 to show the benefits and advantages of the InPA-SQP algorithm.

- In Chapter 3, the effectiveness of InPA-SQP is validated by applying it to regulate the output voltage of a full bridge DC/DC based power conditioning system in an experimental testbed developed at the University of Michigan. The dynamics of the DC/DC converter, experimental setup, and experimental results are illustrated. Chapter 3 also shows how the InPA-SQP method is used for path following of a model ship (based on MPC) in Marine Hydro-Dynamic Lab (MHL) at the University of Michigan.

- Chapter 4 provides a robust control strategy for constrained linear systems with less conservative region of attraction and no online optimization required.
- To provide a less conservative robust control strategy for linear systems subject to bounded additive disturbances, Chapter 5 characterizes MRIA sets once the dynamics of the bounded additive disturbance has been incorporated. It is shown that MRIA sets are smaller and results in less conservative robust control strategies once the dynamic of disturbance is incorporated.

Chapter 2

Integrated Perturbation Analysis and Sequential Quadratic Programming

In this chapter, we consider the general problem of optimal control of discrete-time systems. The constrained NE method is introduced for a general class of constraints. A second order sufficient condition (SSC) for optimality of the nominal optimal solution for the problem $\mathscr{P}(x)$, defined in (1.3) is provided. The same condition is also proven to be a sufficient condition for existence of the neighboring extremal solution and it is computationally verifiable of order N. In Section 2.1, the NE method is illustrated and its application in approximating optimal solutions is provided. In Section 2.2, it is shown how the NE method is combined with SQP to exploit the special structure of MPC and provide a computationally efficient MPC solver.

2.1 First order approximation of the optimal solution: A neighboring extremal approach

In this section, the focus is on NE method that is used to make the optimization-based control methods, including MPC, computationally feasible for real-time implementation. The NE method provides a first order approximation of the optimal solution corresponding to a perturbed initial state for systems subject to input-state constraints. Such an approximation will be used to accelerate computation of MPC-optimal solution, as will be described in Sections 2.2 and 2.3.

As mentioned before, the optimization problem $\mathscr{P}(x(k))$ (defined in (1.3)) is solved at each time instant k. Given that the change in state from one sample to the next is often incremental, i.e., $x(k+1) - x(k)$ is small, it is desired to approximate the optimal solution for the problem $\mathscr{P}(x(k+1))$ using the solution of the problem $\mathscr{P}(x(k))$. In other words, the optimal solution is approximated, if the initial state

$x(k)$ for the problem $\mathscr{P}(x(k))$, is perturbed by $x(k+1) - x(k)$. This provides an optimal solution for the problem $\mathscr{P}(x(k+1))$.

In this section, a closed form NE solution is provided under the assumption that the activity status of constraints does not change due to a small state perturbation. The proposed NE solution require order N computations [61]. If the perturbation in initial state is large enough to violate this assumption, an algorithm is provided in Subsection 2.1.5 that handles a large perturbation. Moreover, a second order sufficient condition for optimality of the nominal optimal solution is provided that is the same as the sufficient condition for existence of the NE solution. This condition is computationally verifiable of order N.

The NE approach can be employed in circumstances where an optimal solution is available a priori and the optimal solution corresponding to a perturbed initial state needs to be approximated, thereby avoiding the burden of solving again the optimization problem, see [40, 60]. In particular, these results can be used for the development of fast MPC algorithms [61], that is described in Section 2.2.

2.1.1 NE Formulation

Consider the following optimal control problem

$$\mathcal{P}(x_0) : \min_{u:[0,N]\to\mathbb{R}^m, x:[0,N]\to\mathbb{R}^n} J[u(\cdot), x(\cdot)], \tag{2.1}$$

where

$$J[u(\cdot), x(\cdot)] = \sum_{k=0}^{N-1} L(x(k), u(k)) + \Phi(x(N)) \tag{2.2}$$

subject to:

$$\begin{aligned} x(k+1) &= f(x(k), u(k)), \quad f : \mathbb{R}^{n+m} \to \mathbb{R}^n; && (2.3)\\ x(0) &= x_0,\ x_0 \in \mathbb{R}^n; && (2.4)\\ C(x(k), u(k)) &\leq 0,\ C : \mathbb{R}^{n+m} \to \mathbb{R}^l, && (2.5)\\ \bar{C}(x(k)) &\leq 0\ \ \bar{C} : \mathbb{R}^n \to \mathbb{R}^{\bar{l}}. && (2.6) \end{aligned}$$

We assume that the functions L, Φ, f, C, and $\bar{C}$ are twice continuously differentiable with respect to their arguments. Here, the state-only constraints $\bar{C}$ are separated from the mixed state-input constraints C, for reasons which will become apparent later on in Section 2.1.3.

Let $x^o(k)$, $u^o(k)$, $k \in [0, N]$ be the state and control vector sequences corre-

sponding to the optimal solution in the problem of minimizing (2.2) subject to the constraints (2.3)-(2.5) with the initial condition $x(0)$. The solution $(x^o,\ u^o)$ which is assumed to be given in the context of NE analysis, is referred to as the nominal solution.

Let $C^a(x(k), u(k))$ be a vector consisting of those elements of the vector $C(x(k), u(k))$ which correspond to active inequality constraints. That is, $C^a(x(k), u(k))$ is the empty vector if no inequality constraints are active at the time instant k and $C^a(x(k), u(k)) \in \mathbb{R}^{l'}$, if l'(out of l) constraints are active. Hence, once number of active constraints change, the dimension of the vector $C^a(x(k), u(k))$ changes. Similar definitions apply to $\bar{C}^a$.

Moreover, let $\mu(k)$ and $\bar{\mu}(k)$ be the Lagrange multipliers associated with constraints C^a and $\bar{C}^a$ and $\lambda(k+1)$ be the Lagrange multiplier associated with the equality constraint (2.3), which is traditionally referred to as the vector of co-states. We can then define the Hamiltonian function as follows:

$$H(x(k), u(k), \lambda(k+1), \mu(k)) = L(x(k), u(k)) + \bar{\mu}(k)^T \bar{C}^a(x(k)) + \lambda(k+1)^T f(x(k), u(k)) + \mu(k)^T C^a(x(k), u(k)). \tag{2.7}$$

Before proceeding, the following compact notation is defined for partial derivatives that will be used for the rest of the section, and where for notational simplicity the dependency of the partial derivatives on x, u has been dropped and replaced by k:

$$\begin{aligned}
H_{uu}(k) &:= \frac{\partial^2 H}{\partial u^2}(x(k), u(k), \lambda(k+1), \mu(k)), & f_u(k) &:= \frac{\partial f}{\partial u}(x(k), u(k)),\\
H_{ux}(k) &:= \frac{\partial}{\partial x}\left(\frac{\partial H^T}{\partial u}(x(k), u(k), \lambda(k+1), \mu(k))\right), & f_x(k) &:= \frac{\partial f}{\partial x}(x(k), u(k)),\\
H_{xu}(k) &:= H_{ux}(k)^T, & C^a_u(k) &:= \frac{\partial C^a}{\partial u}(x(k), u(k)),\\
H_{xx}(k) &:= \frac{\partial^2 H}{\partial x^2}(x(k), u(k), \lambda(k+1), \mu(k)), & C^a_x(k) &:= \frac{\partial C^a}{\partial x}(x(k), u(k)),\\
\bar{C}^a_x(k) &:= \frac{\partial \bar{C}^a}{\partial x}(x(k)), & &
\end{aligned}$$

for $k = 0, \cdots, N-1$ and

$$\Phi_x(N) := \frac{\partial \Phi}{\partial x}(x(N)),\ \Phi_{xx}(N) := \frac{\partial \Phi}{\partial x}(\Phi_x(N))$$

Since the nominal solution $x^o(\cdot)$ and $u^o(\cdot)$ is optimal, it satisfies the following necessary optimality conditions, or Karush-Kuhn-Tucker (KKT) conditions, namely

$$\begin{aligned}
&\lambda(k) = H_x(k),\ k = 0, \cdots, N-1,\\
&H_u(k) = 0,\ k = 0, \cdots, N-1,\\
&\lambda(N) = \frac{\partial \Phi}{\partial x}(x(N)) + \bar{\mu}(N)^T \bar{C}_x^(N),\\
&\mu(k) > 0,\ \bar{\mu}(k) > 0\ k = 0, \cdots, N.
\end{aligned} \tag{2.8}$$

Definition 2.1.1. *The Neighboring Extremal (NE) solution refers to the state and control sequences which minimize the second order variation of the Hamiltonian function $H(\cdot)$ subject to linearized constraints, i.e., it is a solution of the following optimization problem:*

$$\min_{\delta u(\cdot), \delta x(\cdot)} \delta^2 \bar{J} \tag{2.9}$$

$$\begin{aligned}
\delta^2 \bar{J} = &\frac{1}{2} \delta x(N)^T (\Phi_{xx}(N) + \frac{\partial}{\partial x}(\bar{C}_x^T(x(N))\bar{\mu}(N)))\delta x(N)\\
&+ \frac{1}{2} \sum_{k=0}^{N-1} \begin{bmatrix} \delta x(k) \\ \delta u(k) \end{bmatrix}^T \begin{bmatrix} H_{xx}(k) & H_{xu}(k) \\ H_{ux}(k) & H_{uu}(k) \end{bmatrix} \begin{bmatrix} \delta x(k) \\ \delta u(k) \end{bmatrix},
\end{aligned} \tag{2.10}$$

subject to the constraints:

$$\delta x(k+1) = f_x(k)\delta x(k) + f_u(k)\delta u(k), \tag{2.11}$$

$$\delta x(0) = \delta x_0, \tag{2.12}$$

$$C_x^a(x(k), u(k))\delta x(k) + C_u^a(x(k), u(k))\delta u(k) = 0, \tag{2.13}$$

$$\bar{C}_x^a(x(k))\delta x(k) = 0. \tag{2.14}$$

Remark 2.1.1. *It can be verified that the NE solution approximates the optimal state and control sequences for the perturbed initial state, provided that the perturbation is sufficiently small [46]. Specifically, the NE solution is a first order correction to the optimal state and control sequences so that the necessary conditions (2.8) for optimality are maintained for the perturbed initial condition.*

In the sequel, the NE solution for the problem (2.1) is developed.

2.1.2 Constraint back-propagation

Before providing the NE solution, an approach is introduced which can be used to transform the set of constraints (2.11)-(2.14) to an equivalent set of constraints, (2.11), (2.12) and

$$\tilde{C}_x(x(k),u(k))\delta x(k)+\tilde{C}_u(x(k),u(k))\delta u(k)=0, \tag{2.15}$$

with $\tilde{C}_u(x(k),u(k))$ being of full row rank. This is necessary to avoid singularity in the NE solution, as it will be clarified in the sequel.

When the number of active inequality constraints at time k is greater than the number of inputs, i.e., C_u^a has more rows than columns, $C_u^a(k)$ has dependent rows and it can be transformed into the following form

$$\begin{bmatrix} \tilde{C}_u(k) \\ 0 \end{bmatrix}$$

for some $\tilde{C}_u(k)$ with independent rows. Therefore, equation (2.13) can be decomposed into

$$\tilde{C}_x(k)\delta x(k) + \tilde{C}_u(k)\delta u(k) = 0, \tag{2.16}$$

$$\hat{C}_x(k)\delta x(k) = 0, \tag{2.17}$$

for appropriately defined $\tilde{C}_x(k)$ and $\hat{C}_x(k)$. Using the linearized version of (2.3), namely

$$\delta x(k+1) = f_x(k)\delta x(k) + f_u(k)\delta u(k) \tag{2.18}$$

for $k > 0$, (2.17) can be rewritten as

$$\hat{C}_x(k)\left(f_x(k-1)\delta x(k-1) + f_u(k-1)\delta u(k-1)\right) = 0. \tag{2.19}$$

Therefore, one can effectively replace the constraints (2.13) by (2.16), and the remaining constraints (2.17) are back-propagated to the time instant $k-1$, thereby imposing constraints on $\delta x(k-1)$ and $\delta u(k-1)$. This technique, which refines the constraints at time k and shifts other state-only constraints to $k-1$, is referred to in this chapter as constraint backpropagation.

Remark 2.1.2. *While the concept of back-propagation is illustrated here for the case where the constraint is a function of x and u, it can be applied to the case when the constraints are not dependent explicitly on u.*

Since the back-propagated constraints (2.19) have to be enforced, together with other active constraints, at time $k-1$, the resulting active constraints may outnumber control inputs at the time $k-1$. If this does indeed happen, then the same technique has to be applied repeatedly until at some time instant $k-j$ the back-propagated constraints can be absorbed by the matrix $C^a(k-j)$ while $\tilde{C}_u(k)$ has full row rank for $k = 0, \cdots, N-1$.

Having the general constraint back-propagation for the neighboring extremal solution explained conceptually, the detailed formulation of the neighboring extremal solution is provided in the next section.

2.1.3 Neighboring Extremal solution for discrete time optimal control problem subject to general constraints

In this section a NE solution is derived for nonlinear systems subject to general input-state constraints. Let matrix sequences $\tilde{C}_u(\cdot)$, $\hat{C}_x(\cdot)$, $\tilde{C}_x(\cdot)$ and $S(\cdot)$ be defined using the following backward, recursive equations. Let

$$\begin{aligned} &\hat{C}_x(N) := \bar{C}_x^a(x(N)), \\ &S(N) := \Phi_{xx}(N) + \frac{\partial}{\partial x}(\bar{C}_x^T(x(N))\bar{\mu}(N)), \end{aligned} \tag{2.20}$$

and, at the time instant k, define

$$\begin{aligned} &C_{aug}(k) := \begin{bmatrix} C_u^a(k) \\ \hat{C}_x(k+1)f_u(k) \end{bmatrix}, \\ &\tilde{r}_k := rank(C_{aug}(k)). \end{aligned} \tag{2.21}$$

At each time instant k, there is a matrix $P(k)$ that transforms the matrix $C_{aug}(k)$ into the block form,

$$P(k)C_{aug}(k) = \begin{bmatrix} \tilde{C}_u(k) \\ 0 \end{bmatrix}, \tag{2.22}$$

with $\tilde{C}_u(k) \in \mathbb{R}^{\tilde{r}_k \times m}$ having linearly independent rows. Note that if $C_{aug}(k)$ is of full row rank, then $P(k) = I$, and the zero matrix on the right hand side of (2.22) is empty.

By denoting

$$\Gamma(k) := \begin{bmatrix} P(k) \begin{bmatrix} C_x^a(k) \\ \hat{C}_x(k+1)f_x(k) \end{bmatrix} \\ \bar{C}_x^a(k) \end{bmatrix}, \tag{2.23}$$

and assuming that γ_k is the number of rows of matrix $\Gamma(k)$, Γ can be partitioned into a block matrix as $\Gamma(k) = \begin{bmatrix} \tilde{C}_x(k) \\ \hat{C}_x(k) \end{bmatrix}$, where $\tilde{C}_x$, $\hat{C}_x$ being the first $\tilde{r}_k$ and last $\gamma_k - \tilde{r}_k$ rows of Γ respectively, namely:

$$\begin{aligned} \tilde{C}_x(k) &:= [I_{\tilde{r}_k \times \tilde{r}_k} \; 0_{\tilde{r}_k \times (\gamma_k - \tilde{r}_k)}]\Gamma(k) \in \mathbb{R}^{\tilde{r}_k \times m}, \\ \hat{C}_x(k) &:= [0_{(\gamma_k - \tilde{r}_k) \times \tilde{r}_k} \; I_{(\gamma_k - \tilde{r}_k) \times (\gamma_k - \tilde{r}_k)}]\Gamma(k) \in \mathbb{R}^{(\gamma_k - \tilde{r}_k) \times m}. \end{aligned} \tag{2.24}$$

The above manipulations (2.21)-(2.24) enable our constraint back-propagation approach in which the state equation $x(k) = f(x(k-1), u(k-1))$ is used to transform the constraints to overcome the issue with $C_u^a(k)$ not being full rank.

Define $Z_{uu}(\cdot)$, $Z_{ux}(\cdot)$ and $Z_{xx}(\cdot)$ as

$$\begin{aligned} Z_{uu}(k) &:= H_{uu}(k) + f_u^T(k) S(k+1) f_u(k), \\ Z_{ux}(k) &:= Z_{xu}(k)^T = H_{ux}(k) + f_u^T(k) S(k+1) f_x(k), \\ Z_{xx}(k) &:= H_{xx}(k) + f_x^T(k) S(k+1) f_x(k). \end{aligned} \tag{2.25}$$

The matrix $S(k)$ for $k < N$ is defined as follows

$$S(k) = Z_{xx}(k) - [Z_{xu}(k) \; \tilde{C}_x^T(k)] K_0(k) \begin{bmatrix} Z_{ux}(k) \\ \tilde{C}_x(k) \end{bmatrix}, \tag{2.26}$$

where

$$K_0(k) = \begin{bmatrix} Z_{uu}(k) & \tilde{C}_u(k)^T \\ \tilde{C}_u(k) & 0 \end{bmatrix}^{-1}. \tag{2.27}$$

Using equation (2.20) as an initial condition for backward iteration, the matrix sequences $Z_{uu}(\cdot)$, $Z_{ux}(\cdot)$, $Z_{xx}(\cdot)$, $\tilde{C}_u(\cdot)$, $\tilde{C}_x(\cdot)$, $\hat{C}_x(\cdot)$, $S(\cdot)$ and $P(\cdot)$ are calculated according to equations (2.22), (2.24), (2.25), and (2.26). The role of constraint back-propagation becomes apparent in (2.27), where the invertibility of the matrix on the right-hand side requires that $\tilde{C}_u(\cdot)$ is full row rank, once Z_{uu} is strictly positive definite.

Lemma 2.1.1. *If $\hat{C}_x(0)$ is empty, and*

$$Z_{uu}(k) \succ 0 \text{ for } k \in [0, N-1],$$

then the problem (2.9) subject to constraints (2.11)-(2.14) is convex.

Proof. The existence of neighboring extremal solution is guaranteed if the quadratic cost (2.9) is positive definite in the linear variety defined by equations (2.11)-(2.14). Note that, the equality constraints (2.13) and (2.14) is equivalent to constraints (2.61). It is therefore sufficient to prove that the optimization problem with the cost (2.9) subject to constraints (2.11), (2.12), and (2.61) is convex.

Assuming that at the time instant k $rank(\tilde{C}_u(k)) = r_k$, and $\{e_{k,1}, \cdots, e_{k,m-r_k}\}$ form a basis for the null space of the matrix $\tilde{C}_u(k)$ such that

$$e_{k,t}^T Z_{uu}(k) e_{k,s} = \begin{cases} 1, \text{ if } t = s; \\ 0, \text{ otherwise.} \end{cases} .$$

Let us define the vector $\Omega_{k,i}$ as follows

$$\Omega_{k,i} := [\delta x_{k,i}(N)^T,\ \delta u_{k,i}(N-1)^T,\ \delta x_{k,i}(N-1)^T,\ \cdots \delta u_{k,i}(0)], \tag{2.28}$$

where

$$\begin{aligned} &\delta u_{k,i}(j) = 0 \ j = 0, \cdots, k-1, \\ &\delta x_{k,i}(j) = 0 \ j = 0, \cdots, k, \\ &\delta u_{k,i}(k) = e_{k,i}, \\ &\delta x_{k,i}(k+1) = f_u(k) e_{k,i}, \\ &\delta x_{k,i}(j+1) = (f_x(j) + f_u(j) K^*(j)) \delta x_{k,i}(j) \ , \\ &j = k+1, \cdots, N-1, \\ &\delta u_{k,i}(j) = K^*(j) \delta x_{k,i}(j),\ j = k+1, \cdots, N-1, \end{aligned} \tag{2.29}$$

for $k = 0, \cdots, N-1$ and $i = 1, \cdots, m - r_k$.

In addition, let us define

$$\Omega^* := [\delta x^*(N)^T,\ \delta u^*(N-1)^T,\ \delta x^*(N-1)^T,\ \cdots \delta u^*(0)] \tag{2.30}$$

where

$$\begin{aligned} &\delta x^*(0) = \delta x(0), \\ &\delta u^*(k) = K^*(k) \delta x^*(k), \\ &\delta x^*(k+1) = (f_x(k) + f_u(k) K^*(k)) \delta x^*(k),\ k = 0, \cdots, N-1. \end{aligned} \tag{2.31}$$

It can then be seen that $\Omega^* + [((\Omega_{k,i})_{i=1}^{m-r_k})_{k=1}^{N-1}]$ forms the linear variety characterized

by equations (2.11-2.13). To show that the problem is convex if $Z_{uu}(k) \succ 0,\ k = 0, \cdots, N-1$, it is sufficient to show that

$$\delta^2 \bar{J} := 1/2 \delta x_{m,i}(N)^T \Phi_{xx}(N) \delta x_{n,j}(N) + 1/2 \sum_{k=0}^{N-1} \begin{bmatrix} \delta x_{m,i}(k) \\ \delta u_{m,i}(k) \end{bmatrix}^T \begin{bmatrix} H_{xx}(k) & H_{xu}(k) \\ H_{ux}(k) & H_{uu}(k) \end{bmatrix} \begin{bmatrix} \delta x_{n,j}(k) \\ \delta u_{n,j}(k) \end{bmatrix}$$
$$= \begin{cases} e_{m,i}^T Z_{uu}(k) e_{m,i} & \text{if } m = n \text{ and } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2.32}$$

Lemma 2.1.2. *if $p > m, n$ then*

$$\delta^2 \bar{J}_p := \delta x_{m,i}(p+1)^T S(p+1) \delta x_{n,j}(p+1) + \begin{bmatrix} \delta x_{m,i}(p) \\ \delta u_{m,i}(p) \end{bmatrix}^T \begin{bmatrix} H_{xx}(p) & H_{xu}(p) \\ H_{ux}(p) & H_{uu}(p) \end{bmatrix} \begin{bmatrix} \delta x_{n,j}(p) \\ \delta u_{n,j}(p) \end{bmatrix}$$
$$= \delta x_{m,i}(p)^T S(p) \delta x_{n,j}(p) \tag{2.33}$$

Proof. From equations (2.29) and (2.11) we have

$$\begin{aligned} &\delta x_{m,i}(p+1)^T S(p+1) \delta x_{n,j}(p+1) \\ &= \delta x_{m,i}(p)^T (f_x(p) + f_u(p) K^*(p))^T S(p+1) (f_x(p) + f_u(p) K^*(p)) \delta x_{n,j}(p). \end{aligned} \tag{2.34}$$

Using the definitions (2.47), (2.25) and equation (2.34) , we have

$$
\begin{aligned}
&\delta^2 \bar{J}_p = \delta x_{m,i}(p)^T (f_x(p)+f_u(p)K^*(p))^T S(p+1)(f_x(p)+f_u(p)K^*(p))\delta x_{n,j}(p)+ \\
&\begin{bmatrix} \delta x_{m,i}(p) \\ \delta u_{m,i}(p) \end{bmatrix}^T \begin{bmatrix} H_{xx}(p) & H_{xu}(p) \\ H_{ux}(p) & H_{uu}(p) \end{bmatrix} \begin{bmatrix} \delta x_{n,j}(p) \\ \delta u_{n,j}(p) \end{bmatrix} \\
&= \delta x_{m,i}(p)^T Z_{xx}(p) \delta x_{n,j}(p) \\
&+ \delta x_{m,i}(p)^T ([f_u(p)\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})^T S(p+1)([f_u(p)\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})\delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T ([f_u(p)\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})^T S(p+1) f_x(p) \delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T f_x(p)^T S(p+1)([f_u(p)\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})\delta x_{n,j} \\
&+ \delta x_{m,i}(p)^T ([I\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})^T H_{uu}(p)([I\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})\delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T ([I\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})^T H_{ux}(p) \delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T H_{ux}(p)^T ([I\ 0]K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix})\delta x_{n,j}(p).
\end{aligned}
$$

The above equation can be further simplified, using definition (2.25), to the following form

$$
\begin{aligned}
&\delta^2 \bar{J}_p = \delta x_{m,i}(p)^T Z_{xx}(p) \delta x_{n,j}(p) + \delta x_{m,i}(p)^T [Z_{ux}(p)^T\ \tilde{C}_x(p)^T] \times \\
&K_0(p) \begin{bmatrix} Z_{uu}(p) & 0 \\ 0 & 0 \end{bmatrix} K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix} \delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T [Z_{ux}(p)^T\ \tilde{C}_x(p)^T] K_0(p) \begin{bmatrix} Z_{ux}(p) \\ 0 \end{bmatrix} \delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T \begin{bmatrix} Z_{ux}(p) \\ 0 \end{bmatrix}^T K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix} \delta x_{n,j}(p).
\end{aligned}
\tag{2.35}
$$

According to equality (2.13), we have

$$\begin{aligned}
&\delta x_{m,i}(p)^T[Z_{ux}(p)^T\ \tilde{C}_x(p)^T]K_0(p)\begin{bmatrix}0\\ \tilde{C}_u(p)\delta u_{n,j}(p)+\tilde{C}_x(p)\delta x_{n,j}(p)\end{bmatrix}\\
&+\delta x_{n,j}(p)^T[Z_{ux}(p)^T\ \tilde{C}_x(p)^T]K_0(p)\begin{bmatrix}0\\ \tilde{C}_u(p)\delta u_{n,j}(p)+\tilde{C}_x(p)\delta x_{m,i}(p)\end{bmatrix}\\
&=: N_p = 0
\end{aligned} \tag{2.36}$$

Using the optimal feedback equation (2.46) and (2.47), N_p can be expanded to the following form

$$\begin{aligned}
&N_p = \delta x_{m,i}(p)^T[Z_{ux}(p)^T\ \tilde{C}_x(p)^T]K_0(p)\begin{bmatrix}0\\ \tilde{C}_u(p)\end{bmatrix}[I\ 0]K_0(k)\times\\
&\begin{bmatrix}Z_{ux}(p)\\ \tilde{C}_x(p)\end{bmatrix}\delta x_{n,j}(p)+\delta x_{m,i}(p)^T\begin{bmatrix}Z_{ux}(p)\\ \tilde{C}_x(p)\end{bmatrix}^T K_0(p)\begin{bmatrix}I\\ 0\end{bmatrix}\times\\
&[0\ \tilde{C}_u(p)^T]K_0(p)\begin{bmatrix}Z_{ux}(p)\\ \tilde{C}_x(p)\end{bmatrix}\delta x_{n,j}(p)\\
&+\delta x_{m,i}(p)^T[Z_{ux}(p)^T\ \tilde{C}_x(p)^T]K_0(p)\begin{bmatrix}0\\ \tilde{C}_x(p)\end{bmatrix}\delta x_{n,j}(p)\\
&+\delta x_{m,i}(p)[0\ \tilde{C}_x(p)^T]K_0(p)\begin{bmatrix}Z_{ux}(p)\\ \tilde{C}_x(p)\end{bmatrix}\delta x_{n,j}(p).
\end{aligned} \tag{2.37}$$

From equations (2.35), (2.36), and (2.37) and some algebraic simplification we have

$$
\begin{aligned}
&\delta^2 \bar{J}_p = \delta^2 \bar{J}_p + N_p \\
&= \delta x_{m,i}(p)^T Z_{xx}(p) \delta x_{n,j}(p) + \delta x_{m,i}(p)^T [Z_{ux}(p)^T \ \tilde{C}_x(p)^T] \times \\
&K_0(p) \begin{bmatrix} Z_{uu}(p) & \tilde{C}_u(p)^T \\ \tilde{C}_u(p) & 0 \end{bmatrix} K_0(p) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix} \delta x_{n,j}(p) \\
&- \delta x_{m,i}(p)^T [Z_{ux}(p)^T \ \tilde{C}_x(p)^T] K_0(k) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix} \delta x_{n,j}(p) \\
&- \delta x_{n,j}(p)^T [Z_{ux}(p)^T \ \tilde{C}_x(p)^T] K_0(k) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix} \delta x_{m,i}(p) \\
&= \delta x_{m,i}(p)^T (Z_{xx}(p) - [Z_{ux}(p)^T \ \tilde{C}_x(p)^T] K_0(k) \begin{bmatrix} Z_{ux}(p) \\ \tilde{C}_x(p) \end{bmatrix}) \\
&\times \delta x_{n,j}(p) = \delta x_{m,i}(p)^T S(p) \delta x_{n,j}(p). \qquad \blacksquare
\end{aligned}
\tag{2.38}
$$

□

From equation (2.33) and definition of $\delta^2 \bar{J}$ in equation (2.32), we can see that if $m = n$ then

$$
\begin{aligned}
\delta^2 \bar{J} &= e_{m,i}^T (f_u(p)^T S(p+1) f_u(p) + H_{uu}(p)) e_{m,j} \\
&= e_{m,i}^T Z_{uu}(p) e_{n,j},
\end{aligned}
\tag{2.39}
$$

and therefore the equation (2.32) is satisfied.

If $m \neq n$ then without lost of generality we can assume that $n > m$. Then

$$
\begin{aligned}
\delta^2 \bar{J} &= \delta x_{m,i}(n)^T (f_x(n) + f_u(n) K^*(n))^T S(n+1) f_u(n) e_{n,j} \\
&+ \delta u_{m,i}(n)^T H_{uu}(n) e_{n,j} + \delta x_{m,i}(n)^T H_{xu}(n) e_{n,j}.
\end{aligned}
\tag{2.40}
$$

Substituting equation (2.47) in (2.40) and using equation (2.25) we have

$$\begin{aligned}
\delta^2\bar{J} &= \delta x_{m,i}(n)^T Z_{xu}(n) e_{n,j} \\
&+ \delta x_{m,i}(n)^T \begin{bmatrix} Z_{ux}(n) \\ \tilde{C}_x(n) \end{bmatrix}^T K_0(n) \begin{bmatrix} -f_u(n)^T S(n+1) f_u(n) \\ 0 \end{bmatrix} e_{n,j} \\
&+ \delta x_{m,i}(n)^T \begin{bmatrix} Z_{ux}(n) \\ \tilde{C}_x(n) \end{bmatrix}^T K_0(n) \begin{bmatrix} -H_{uu}(n) \\ 0 \end{bmatrix} e_{n,j} \\
&= \delta x_{m,i}(n)^T Z_{ux}(n) e_{n,j} \\
&+ \delta x_{m,i}(n)^T \begin{bmatrix} Z_{ux}(n) \\ \tilde{C}_x(n) \end{bmatrix}^T K_0(n) \begin{bmatrix} -Z_{uu}(n) \\ 0 \end{bmatrix} e_{n,j}.
\end{aligned} \tag{2.41}$$

Since $e_{n,j}$ belongs to the null space of the matrix $\tilde{C}_u(n)$, the equation (2.41) can be written as

$$\delta^2\bar{J} = \delta x_{m,i}(n)^T Z_{xu}(n) e_{n,j} + \delta x_{m,i}(n)^T \begin{bmatrix} Z_{ux}(n) \\ \tilde{C}_x(n) \end{bmatrix}^T K_0(n) \begin{bmatrix} -Z_{uu}(n) \\ -\tilde{C}_u(n) \end{bmatrix} e_{n,j}. \tag{2.42}$$

Substituting $K_0(n)$ from equation (2.27) we have

$$\delta^2\bar{J} = \delta x_{m,i}(n)^T Z_{xu}(n) e_{n,j} + \delta x_{m,i}(n)^T \begin{bmatrix} Z_{ux}(n) \\ \tilde{C}_x(n) \end{bmatrix}^T \begin{bmatrix} -I \\ 0 \end{bmatrix} e_{n,j} = 0. \tag{2.43}$$

Equations (2.39) and (2.43) show that the sufficient condition (2.32) is satisfied and the proof is complete.

□

It should be noted that the system is linearized at the optimal solution which is time-varying and hence the linearized system is time-varying. The following theorem provides sufficient condition for the existence of the NE solution.

Theorem 2.1.1. *If $\hat{C}_x(0)$ is empty and $x^o(\cdot)$ and $u^o(\cdot)$ satisfy the necessary condition for optimality (2.8) and*

$$Z_{uu}(k) \succ 0 \text{ for } k \in [0, N-1], \tag{2.44}$$

then $x^o(\cdot)$ and $u^o(\cdot)$ satisfy the strong second order optimality condition and a NE solution subject to the inequality constraints and initial state perturbation $\delta x(0)$ exists and is unique.

Proof. The condition of $\hat{C}_x(0)$ being empty implies linear independence of the gradients of active constraints (including equality constraint (2.3)). Moreover, the condition (2.44) guarantees strong convexity of the problem $\mathcal{P}(x_0)$ at the nominal solution $x^o(\cdot)$ and $u^o(\cdot)$. Therefore, assuming that $x^o(\cdot)$ and $u^o(\cdot)$ satisfy the necessary optimality condition (2.8) with strict complementarity, i.e., $\mu(k) > 0$ and $\bar{\mu}(k) > 0$, the strong second order sufficient condition for the problem $\mathcal{P}(x_0)$ is satisfied and therefore the pair $x^o(\cdot)$ and $u^o(\cdot)$ is a strong local minimum. This shows existence of NE solution [83]. Uniqueness is the direct consequence of strong convexity that is implied by (2.44) and Lemma 2.1.1. □

Remark 2.1.3. *If conditions of Theorem 2.1.1 are satisfied, and Lagrange multipliers associated with inequalities (2.5) and (2.6) are strictly positive, then the strong second order sufficient condition (SSC) for optimality is satisfied at the nominal solution $u^o(\cdot)$ and $x^o(\cdot)$. The SSC has an important consequence that there exists a neighborhood of initial state x_0, $\mathcal{N}(x_0)$, and continuously differentiable functions $u(k)(x_0) : \mathcal{N}(x_0) \to \mathbb{R}^m$, for $k = 0, \cdots, N-1$, and $x(k)(x_0) : \mathcal{N}(x_0) \to \mathbb{R}^n$, for $k = 1, \cdots, N$, such that for all $x_0' \in \mathcal{N}(x_0)$:*

1. *The control and state sequences $u(\cdot)(x_0')$ and $x(\cdot)(x_0')$ are the optimal solution to the problem (2.1) with initial state x_0'.*

2. *The active constraints corresponding to the optimal solutions $u(\cdot)(x_0')$ and $x(\cdot)(x_0')$ are the same as those of the nominal optimal solution $x^o(\cdot) = x(\cdot)(x_0)$ and $u^o(\cdot)(x_0) = u(\cdot)(x_0)$.*

These results can be found in [46, 83]. The NE solution is the first order approximation of $u(\cdot)(x_0')$ and $x(\cdot)(x_0')$ in terms of initial state variation x_0' in the neighborhood $\mathcal{N}(x_0)$.

Remark 2.1.4. *The convexity condition (2.44) in the absence of inequality constraints (2.5)-(2.6) reduces to the convexity condition provided in [45]. The proposed NE method provides a unified framework to calculate the NE solution and check a sufficient condition for the existence of the solution for systems subject to general constraints, including the unconstrained problem as a special case. However, note that the proof of sufficiency of condition (2.44) for the existence of NE solution requires a different approach than that in [43], [45]. Note that for continuous-time systems, convexity of the problem in the vicinity of the optimal solution has been considered as a mechanism to assure existence of NE solution in the prior work [40, 82].*

Remark 2.1.5. *Note that only gradients of active constraints are considered when checking the strong second order sufficient condition in a general nonlinear programming problem.*

Remark 2.1.6. *For the infinite length of horizon, i.e., $N = \infty$, the existence of an optimal control sequence $u(\cdot)$ that renders a cost finite is usually guaranteed by stabilizability assumption on the linearized system (2.3). However, for finite length of horizon, i.e., if $N < \infty$, the cost is always finite and such stabilizability assumption is not required.*

Remark 2.1.7. *The condition $\hat{C}_x(0)$ being empty implies that the constraint back propagation does not produce a constraint on the initial state variation $\delta x(0)$ which is not a variable. The treatment of the case when $\hat{C}_x(0)$ is empty in the context of receding horizon optimal control is presented in Appendix A.*

The following Theorem provides the NE solution for the problem formulated in Section 2.1.1.

Theorem 2.1.2. *Suppose the perturbation $\delta x(0)$ in the initial state $x(0)$ does not change the activeness status of the constraints, i.e., the optimal solution corresponding to initial state $x(0) + \delta x(0)$ has the same active constraints as the optimal solution to $x(0)$, where $\delta x(0)$ represents a perturbation in initial state. If*

$$Z_{uu}(k) \succ 0 \text{ for } k \in [0, N-1], \tag{2.45}$$

then the NE solution for the initial state perturbation $\delta x(0)$, i.e., solution to (2.9), is $\delta x(k)$ and $\delta u(k)$, $k \in [0, N]$ where

$$\delta u(k) = K^*(k)\delta x(k), \tag{2.46}$$

and K_0, Z_{ux}, and $\tilde{C}_x$ are defined in (2.27), (2.25), (2.24), respectively, and

$$K^*(k) = -[I\ 0]K_0(k) \begin{bmatrix} Z_{ux}(k) \\ \tilde{C}_x(k) \end{bmatrix}. \tag{2.47}$$

Proof. Let us assume that $\delta\lambda(\cdot)$, $\delta\mu(\cdot)$ and $\delta\bar{\mu}(\cdot)$ are the Lagrange multipliers associated with constraints (2.11), (2.13) and (2.14), respectively. Hereafter, the superscript a is dropped for notational simplicity, assuming that the constraints appearing in the equations are active.

Assuming (2.44) is satisfied, by applying the Karush-Kuhn-Tucker (KKT) conditions to the problem (2.9) for the time instant $k = N$, and with $\delta u(N) = 0$

$$\begin{aligned}\delta\lambda(N) &= \left(\Phi_{xx}(N) + \frac{\partial}{\partial x}(\bar{C}_x^T(x^o(N))\bar{\mu}(N))\right)\delta x(N) + \bar{C}_x^T(x^o(N))\delta\bar{\mu}(N),\\ \bar{C}_x(x^o(N))\delta x(N) &= 0.\end{aligned} \tag{2.48}$$

Defining $\delta\hat{\mu}(N) := \delta\bar{\mu}(N)$, $T(N) := 0$, $\hat{C}_x(N) := \bar{C}_x(x(N))$ and

$$S(N) := \Phi_{xx}(N) + \frac{\partial}{\partial x}(\bar{C}_x^T(x^o(N))\bar{\mu}(N)),$$

the first equality in (2.48) can be expressed as

$$\delta\lambda(N) = S(N)\delta x(N) + T(N) + \hat{C}_x^T(N)\delta\hat{\mu}(N). \tag{2.49}$$

Now assume that for the time instant $k+1$, $\delta\lambda(k+1)$ can be represented as

$$\delta\lambda(k+1) = S(k+1)\delta x(k+1) + T(k+1) + \hat{C}_x^T(k+1)\delta\hat{\mu}(k+1) \tag{2.50}$$

and

$$\hat{C}_x(k+1)\delta x(k+1) = 0. \tag{2.51}$$

Equation (2.51) can be written as

$$\hat{C}_x(k+1)f_u(k)\delta u(k) + \hat{C}_x(k+1)f_x(k)\delta x(k) = 0. \tag{2.52}$$

Applying the Karush-Kuhn-Tucker (KKT) conditions to the problem (2.9) at time k,

$$\delta\lambda(k) = H_{xx}\delta x(k) + H_{xu}\delta u(k) + f_x^T(k)\delta\lambda(k+1) + C_x^T(k)\delta\mu(k) + \bar{C}_x^T(k)\delta\bar{\mu}(k). \tag{2.53}$$

Combining equation (2.52) with equations (2.13) and (2.14):

$$\begin{bmatrix} C_u(k) \\ \hat{C}_x(k+1)f_u(k) \\ 0 \end{bmatrix}\delta u(k) + \begin{bmatrix} C_x(k) \\ \hat{C}_x(k+1)f_x(k) \\ \bar{C}_x(k) \end{bmatrix}\delta x(k) = 0. \tag{2.54}$$

Substituting $\delta\lambda(k+1)$, using the expression given by (2.50) into (2.53), obtain:

$$\delta\lambda(k) = Z_{xx}(k)\delta x(k) + Z_{xu}(k)\delta u(k) + f_x^T(k)T(k+1) + \begin{bmatrix} C_x(k) \\ \hat{C}_x(k+1)f_x(k) \\ \bar{C}_x(k) \end{bmatrix}^T \begin{bmatrix} \delta\mu(k) \\ \delta\hat{\mu}(k+1) \\ \delta\bar{\mu}(k) \end{bmatrix}. \tag{2.55}$$

Using the definition of $\Gamma(k)$, the following can be written

$$\begin{aligned} &\begin{bmatrix} C_x(k) \\ \hat{C}_x(k+1)f_x(k) \end{bmatrix}^T \begin{bmatrix} \delta\mu(k) \\ \delta\hat{\mu}(k+1) \end{bmatrix} + \bar{C}_x(k)\delta\bar{\mu}(k) \\ &= \left(P(k) \begin{bmatrix} C_x(k) \\ \hat{C}_x(k+1)f_x(k) \end{bmatrix} \right)^T P(k)^{-T} \begin{bmatrix} \delta\mu(k) \\ \delta\hat{\mu}(k+1) \end{bmatrix} + \bar{C}_x(k)\delta\bar{\mu}(k) \\ &= \Gamma(k) \begin{bmatrix} P(k)^{-T} \begin{bmatrix} \delta\mu(k) \\ \delta\hat{\mu}(k+1) \end{bmatrix} \\ \delta\bar{\mu}(k) \end{bmatrix}. \end{aligned} \tag{2.56}$$

By defining

$$\begin{aligned} \delta\tilde{\mu}(k) &:= [I_{\tilde{r}_k \times \tilde{r}_k} \; 0_{\tilde{r}_k \times (\gamma_k - \tilde{r}_k)}] \begin{bmatrix} P(k)^{-T} \begin{bmatrix} \delta\mu(k) \\ \delta\hat{\mu}(k+1) \end{bmatrix} \\ \delta\bar{\mu}(k) \end{bmatrix}, \\ \delta\hat{\mu}(k) &:= [0_{(\gamma_k - \tilde{r}_k) \times \tilde{r}_k} \; I_{(\gamma_k - \tilde{r}_k) \times (\gamma_k - \tilde{r}_k)}] \begin{bmatrix} P(k)^{-T} \begin{bmatrix} \delta\mu(k) \\ \delta\hat{\mu}(k+1) \end{bmatrix} \\ \delta\bar{\mu}(k) \end{bmatrix} \end{aligned} \tag{2.57}$$

where $\delta\tilde{\mu}(k) \in \mathbb{R}^{\tilde{r}_k}$, $\delta\hat{\mu}(k) \in \mathbb{R}^{(\gamma_k - \tilde{r}_k)}$, and referring to (2.23) for the definition of $\tilde{C}_x(\cdot)$ and $\hat{C}_x(\cdot)$

$$\delta\lambda(k) = Z_{xx}(k)\delta x(k) + Z_{xu}(k)\delta u(k) + f_x^T(k)T(k+1) + \tilde{C}_x(k)^T\delta\tilde{\mu}(k) + \hat{C}_x(k)\delta\hat{\mu}(k). \tag{2.58}$$

Moreover, note that equation (2.51), using (2.11), can be written as

$$\hat{C}_x(k+1)f_u(k)\delta u(k) + \hat{C}_x(k+1)f_x(k)\delta x(k) = 0. \tag{2.59}$$

Equations (2.59), (2.13), and (2.14) imply

$$\begin{bmatrix} C_x(k) \\ \hat{C}_x(k+1)f_u(k) \\ 0 \end{bmatrix} \delta u(k) + \begin{bmatrix} C_x(k) \\ \hat{C}_x(k+1)f_x(k) \\ \bar{C}_x(k) \end{bmatrix} \delta x(k) = 0. \tag{2.60}$$

Using (2.21), (2.22) and (2.24), it can be concluded that

$$\begin{bmatrix} \tilde{C}_u(k) \\ 0 \end{bmatrix} \delta u(k) + \begin{bmatrix} \tilde{C}_x(k) \\ \hat{C}_x(k) \end{bmatrix} \delta x(k) = 0$$

or

$$\begin{aligned} &\tilde{C}_u(k)\delta u(k) + \tilde{C}_x(k)\delta x(k) = 0, \\ &\hat{C}_x(k)\delta x(k) = 0 \end{aligned} \tag{2.61}$$

where $\tilde{C}_u(k)$ has independent rows.

In addition, by applying the Karush-Kuhn-Tucker (KKT) conditions to the problem (2.9)-(2.14), $\delta x(k)$, $\delta u(k)$, $\delta\lambda(k)$ and $\delta\mu(k)$ should satisfy the following equation,

$$H_{ux}(k)\delta x(k) + H_{uu}(k)\delta u(k) + f_u(k)^T\delta\lambda(k+1) + C_u(k)^T\delta\mu(k) = 0. \tag{2.62}$$

Using equations (2.62), (2.50), and (2.22)

$$Z_{uu}(k)\delta u(k) + \tilde{C}_u^T(k)\delta\tilde{\mu}(k) = -Z_{ux}(k)\delta x(k) - f_u^T(k)T(k+1). \tag{2.63}$$

Since $\tilde{C}_u(k)$ is full row rank and $Z_{uu}(k)$ is positive definite, the matrix

$$K_0(k) = \begin{bmatrix} Z_{uu}(k) & \tilde{C}_u^T(k) \\ \tilde{C}_u(k) & 0 \end{bmatrix}^{-1} \tag{2.64}$$

is well defined and from equations (2.61) and (2.63)

$$\begin{bmatrix} \delta u(k) \\ \delta\tilde{\mu}(k) \end{bmatrix} = -K_0(k) \begin{bmatrix} Z_{ux}(k) \\ \tilde{C}_x(k) \end{bmatrix} \delta x(k) - K_0(k) \begin{bmatrix} f_u^T(k)T(k+1) \\ 0 \end{bmatrix}. \tag{2.65}$$

Applying equation (2.65) to (2.58),

$$\delta\lambda(k) = S(k)\delta x(k) + T(k) + \hat{C}_x^T(k)\delta\hat{\mu}(k) \tag{2.66}$$

where $\hat{C}_x^T(k)$ and $S(k)$ are calculated from equation (2.24) and (2.26), respectively, and $T(k) \equiv 0$. Therefore, from (2.65), (2.46) is achieved.

$\hat{C}_x(0)$ being empty guarantees that (2.11)-(2.14) are linearly independent. If $\hat{C}_x(0)$ is non-empty and $\hat{C}_x(0)\delta x(0) \neq 0$, then (2.11)-(2.14) have no solution. If $\hat{C}_x(0)\delta x(0) = 0$, then (2.11)-(2.14) are linearly dependent.

□

Remark 2.1.8. *Our approach is based on the assumption that an optimal solution is known a priori. If such an a priori known solution is suboptimal, then the NE solution can be modified using the InPA-SQP method which is illustrated in Section 2.2, to improve it towards optimality. Moreover, if the optimal solution is pre-computed and stored for a grid of initial states, the NE solution can be used to correct the stored optimal solutions and provide a first order approximation for optimal solutions with initial states inside the grid.*

2.1.4 NE Algorithm

To implement the NE method, one needs to be able to calculate the Lagrange multipliers $\lambda(\cdot)$, $\mu(\cdot)$ and $\bar{\mu}(\cdot)$. A method for these calculation is proposed in Appendix B.

The procedure for determining the NE solution can be summarized as follows, once the Lagrange multipliers are calculated:

- Initialize matrices $S(N)$ and $\hat{C}_x(N)$ using equation (2.20).
- Calculate, in a backward run, matrix sequences $P(\cdot)$ (according to equation (2.22)), $\tilde{C}_u(\cdot)$ and $\tilde{C}_x(\cdot)$ (using equations (2.22) and (2.24)), $Z_{uu}(\cdot)$, $Z_{ux}(\cdot)$, $Z_{xx}(\cdot)$ (using equation (2.25)) and $S(\cdot)$ (using equation (2.26)).
- Given initial state variation $\delta x(0)$, in a forward run, calculate $\delta x(\cdot)$ and $\delta u(\cdot)$ using equation (2.11) and (2.46).

The assumption that the constraint activeness status remains unchanged, when the initial state is perturbed, is essential for Theorem 2.1.2 to hold. Large perturbations for which this assumption may be violated can be handled by repeated application of Theorem 2.1.2 as illustrated in next subsection.

2.1.5 Handling large perturbation in $x(0)$

Theorem 2.1.2 is derived under the assumption that activity status of constraints does not change. To deal with the initial state variation that is large enough to change

activity status of constraints, the perturbed Lagrange multipliers associated with the inequality constraints are analyzed and the perturbed value of $C(x(k), u(k))$ to determine the status of constraint activity after perturbation. The following proposition provides the relation between the initial condition perturbation and the Lagrange multipliers that will allow us to predict the constraint activity change.

Proposition 2.1.1. *If the initial condition $x(0)$ is perturbed by $\delta x(0)$, the optimal Lagrange multiplier perturbation $\delta\mu(k)$ at the time instant k when the constraint is active can be approximated as follows:*

$$\begin{aligned}
&\delta\mu(k) = -[0\ I]K_0(k)\begin{bmatrix} Z_{ux}(k) \\ \tilde{C}_x(k) \end{bmatrix}\Upsilon(k)\delta x(0), \\
&\Upsilon(k) = \prod_{i=0}^{k-1} M(i), \\
&M(i) := f_x(i) + f_u(i)K^*(i),\ i = 0, \cdots, k-1
\end{aligned} \tag{2.67}$$

In addition, if the constraint is not active, then the constraint perturbation can be expressed as

$$\delta C(x(k), u(k)) = (C_x(k) + C_u(k)K^*(k))\Upsilon(k)\delta x(0), \tag{2.68}$$

with $K_0(\cdot)$ and $K^*(\cdot)$ being defined as in Theorem 2.1.2.

Proof. By combining equations (2.65), (2.46) and (2.61), the expression (2.67) can be derived. (2.68) follows directly by taking partial derivatives of $C(\cdot, \cdot)$ and noting that $\delta x(k) = \Upsilon(k)\delta x(0)$. □

Note that the perturbed optimal Lagrange multiplier associated with active constraints $\mu^1(k)$ is:

$$\mu^1(k) = \mu(k) + \delta\mu(k), \tag{2.69}$$

where $\mu(\cdot)$ is the nominal Lagrange multiplier and $\delta\mu(\cdot)$ is calculated from (2.67). If $\mu^{(1)}(k) \geq 0$, one can conclude that the constraint will remain active at the time k for the perturbed solution. Otherwise, it may become inactive because the Lagrange multiplier must always be greater than or equal to zero. Similarly, using equation (2.68), the value of the constraint function corresponding to the perturbed optimal solution is:

$$C(x^{(1)}(k), u^{(1)}(k)) = C(x(k), u(k)) + \delta C(x(k), u(k)), \tag{2.70}$$

where $x^{(1)}(k)$ and $u^{(1)}(k)$ are the following linear approximations of the optimal so-

lution

$$x^{(1)}(k) = x(k) + \delta x(k),\ u^{(1)}(k) = u(k) + \delta u(k). \tag{2.71}$$

If $C(x^{(1)}(k), u^{(1)}(k)) < 0$, the constraint remains inactive. Otherwise, it will become active.

For large perturbation $\delta x(0)$ which causes changes in the constraint activeness status, we consider the line connecting the initial state $x(0)$ and perturbed state $x(0)+\delta x(0)$. We may identify several intermediate points where the status of constraints activeness changes, while between two consecutive points the status remains the same. The solution to the original problem is then formulated by combining several solutions.

Using Proposition 2.1.1, the following algorithm is proposed that identifies intermediate initial states and calculates the corresponding NE solution to approximate the optimal solution when initial state perturbations change the status of the constraints activity:

1. Set $i = 0$, $\delta x^{(0)}(0) = \delta x(0)$ and $x^0(0) = x(0)$;

2. If the constraint is active at the time instant k, compute α_{ik} as:

$$\alpha_{ik} = \frac{\mu^{(i)}(k)}{\delta\mu^{(i)}(k)}. \tag{2.72}$$

   If it is inactive at k, compute α_{ik} as

$$\alpha_{ik} = -\frac{C(x^{(i)}(k), u^{(i)}(k))}{\delta C(x^{(i)}(k), u^{(i)}(k))}, \tag{2.73}$$

   where $\delta C(x^{(i)}, u^{(i)})$ is calculated by (2.68), $\delta\mu^{(i)}(k)$ is calculated by (2.67), and all involved matrices should be evaluated at $x^{(i)}(k)$ and $u^{(i)}(k)$. Then find the smallest $\alpha_{ik} \in [0, 1]$ such that the perturbation $\alpha_{ik}\delta x^{(i)}(0)$ changes the status of the constraint at least at one instant, namely:

$$\alpha_i = \min_k \{\alpha_{ik}, k = 0, \cdots, N-1\ \ 0 \le \alpha_{ik} \le 1\}.$$

   If, for all $k \in [0 : N]$, $\alpha_{ik} < 0$ or $\alpha_{ik} > 1$, set $\alpha_i = 1$.

3. Compute an approximation to the perturbed optimal solution $\delta x^{(i)}(\cdot)$, $\delta u^{(i)}(\cdot)$ for the intermediate perturbation $min\{\alpha_i, 1\}\delta x^{(i)}(0)$ and initial condition $x^{(i)}(0)$ using the perturbation analysis developed in Section 2.1.3.

4. If $\alpha_i = 1$, terminate. Otherwise:

- If $\alpha_i = 0$, change the activity status of the corresponding constraint accordingly. That is, if α_i corresponds an active (inactive) constraint, set the constraint inactive (active). Go to step 2.
- If $\alpha_i < 1$ set

$$\begin{aligned} &\delta x^{(i+1)}(0) = (1-\alpha_i)\delta x^{(i)}(0), \\ &x^{(i+1)}(0) = x^{(i)}(0) + \alpha_i \delta x^{(i)}(0), \\ &i = i+1. \end{aligned}$$

  Go to step 2.

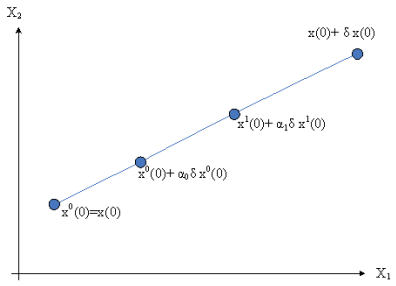

Figure 2.1: Intermediate initial states which handle the large perturbation.

Figure 2.1 shows the line connecting the initial and perturbed states , where the intermediate points are highlighted. Note that the intermediate perturbed initial states lie on the line connecting $x(0)$ to $x(0) + \delta x(0)$. These intermediate states are nominal states at which Theorem 2.1.2 can be repeatedly applied to derive approximations to the optimal solution. Therefore the perturbed optimal control solution corresponding to a large perturbation $\delta x(0)$ can be approximated by augmenting the nominal solution as:

$$u(k) + \sum_i \delta u^i(k) \ .$$

So far, a perturbation analysis is introduced which provides first order approximation of the optimal solution corresponding to the perturbed initial state. The developed perturbation analysis is used in next section to construct a fast MPC algorithm, exploiting the special structure of MPC.

2.2 An Integrated Perturbation Analysis and Sequential Quadratic Programming Approach to MPC

In this section, exploiting the developed perturbation analysis, a computationally efficient method is introduced, which is refered to as the Integrated Perturbation Analysis and Sequential Quadratic Programming (InPA-SQP) approach, for the MPC implementation. It synergistically combines the solutions derived using perturbation analysis and SQP to solve the optimization problem with initial state perturbation and input/state constraints.

The InPA-SQP integrates the perturbation analysis and SQP into a single unified framework to speed up the calculation of the optimal solution for a given nominal solution. The unification is critically dependent on a special formulation of the SQP algorithm which will be presented in Section 2.2.1. This special formulation allows us to cast the solution of the SQP into the same formula as that of the perturbation analysis, thereby facilitating their seamless integration. With the integrated algorithm, one does not need to make a choice of whether to use perturbation analysis or SQP. Instead, the algorithm renders itself to the proper formula automatically when certain conditions are satisfied.

It is worth noting that the InPA-SQP formulation is used to calculate the optimal solution instead of estimating the solution using perturbation analysis or using SQP method alone to compute it. Because of the structure of the unifying formulation, the algorithm is equivalent to SQP method when there is no initial condition perturbation and it falls into perturbation formulation when the optimal nominal solution is given together with a perturbed initial condition. Therefore in either case the InPA-SQP formulation is used to calculate the optimal solution.

2.2.1 Sequential quadratic optimal control based on active set method

In this section, the sequential quadratic programming (SQP) method is formulated for the optimization problem (2.1) as a prelude to introducing the InPA-SQP approach in the next section. This formulation is different from the one of [108] in the sense that it provides a closed form solution based on recursive matrix calculations. When there is no constraint, this formulation is reduced to the one proposed by [43]. Therefore the SQP method to be presented in this section can be considered as an extended version of the one presented in [43] to the case where there is an input-state constraint.

We start with a feasible initial guess of the optimal control $u(k)$, state $x(k)$, co-state $\lambda(k)$ and with Lagrange multipliers $\mu(k)$ and $\bar{\mu}(k)$ associated with the inequality constraints such that they satisfy equations (2.3)- (2.6) and the KKT conditions (2.8). Note that since the initial guess is not an optimal solution, it may not satisfy the optimality condition

$$H_{u(k)}(k) = 0. \tag{2.74}$$

The active inequality constraints at the time instants k are treated as the equality constraints during the active set iteration. The corrections $\delta u(k)$ and $\delta x(k)$ are obtained by solving the following equality constrained quadratic programming problem (QP)

$$\begin{aligned}
&\min_{\delta u(\cdot),\delta x(\cdot)} \sum_{k=0}^{N-1} H^T_{u(k)}(k)\delta u(k) + \delta^2 \bar{J} \\
&\text{subject to: } \delta x(k+1) = f_x(k)\delta x(k) + f_u(k)\delta u(k), \\
&\qquad\qquad \delta x(0) = 0, \\
&\qquad\qquad C^a_x(x(k),u(k))\delta x(k) + C^a_u(x(k),u(k))\delta u(k) = 0, \\
&\qquad\qquad \bar{C}^a_x(x(k))\delta x(k) = 0
\end{aligned} \tag{2.75}$$

where $\delta^2 \bar{J}$ is defined in (2.10).

Proposition 2.2.1. *Let $u(k)$, $x(k)$ and $\lambda(k)$ be the control, state and co-state, respectively, that satisfy the constraints (2.3)- (2.6) and (2.8) and let $\mu(k)$ be the Lagrange multiplier associated with the inequality constraint. In addition, assume that the matrix $Z_{uu}(k)$, defined in (2.25), is positive definite for $k = 1, \ldots, N$. Then the solution of the QP with equality constraint (2.75) is given by*

$$\begin{aligned}
&\begin{bmatrix} \delta u(k) \\ \delta \mu(k) \end{bmatrix} = -K_0(k) \begin{bmatrix} Z_{ux}(k)\delta x(k) + f^T_u(k)T(k+1) + H_u(k) \\ \tilde{C}_x(k)\delta x(k) \end{bmatrix} \\
&\delta x(k+1) = f_x(k)\delta x(k) + f_u(k)\delta u(k) \\
&\delta x(0) = 0
\end{aligned} \tag{2.76}$$

where $K_0(k)$, $Z_{uu}(k)$, $Z_{ux}(k)$ and $Z_{xx}(k)$ are defined in (2.25). Moreover, the matrices

$S(\cdot)$ and $T(\cdot)$ are calculated using the following backward recursive equations

$$\begin{aligned}
&S(N) = \Phi_{xx}(N),\\
&S(k) = Z_{xx}(k) - [Z_{xu}(k)\ \tilde{C}_x^T(k)]K_0(k)\begin{bmatrix} Z_{ux}(k) \\ \tilde{C}_x(k) \end{bmatrix},\\
&T(N) = 0,\\
&T(k) = f_x^T(k)T(k+1) - [Z_{xu}\ \tilde{C}_x^T(k)]K_0(k)\begin{bmatrix} f_u^T(k)T(k+1) + H_u(k) \\ 0 \end{bmatrix}.
\end{aligned} \tag{2.77}$$

Using the result of Proposition 2.2.1, the active set method, which is introduced in [109], is implemented as follows:

First the minimum value of $0 < \alpha \leq 1$ is found such that there exists a time instant k and an element j in the vector $C^a(x(k), u(k))$ or $\bar{C}^a(x(k))$, i.e., $C_j^a(x(k), u(k))$ or $C_j^a(x(k))$ that satisfy

$$C_j^a(x(k) + \alpha\delta x(k), u(k) + \alpha\delta u(k)) = 0 \text{ or } \bar{C}_j^a(x(k) + \alpha\delta x(k)) = 0. \tag{2.78}$$

where $\delta u(k)$ and $\delta x(k)$ are calculated using equation (2.76). If there exist such α that satisfies the condition (2.78) then the corresponding inactive inequality constraint is added to the set of active constraints. $\delta u(k)$ and $\delta x(k)$ are calculated using (2.76) and the equality constraint problem (2.75) is solved at the next iteration with the initial solution $x(\cdot) + \alpha\delta x(k)$ and $u(k) + \alpha\delta u(k)$.

If no such α exists, then the sign of Lagrange multipliers $\mu(k)$, calculated using equation (2.67), is examined. If all the Lagrange multipliers $\mu(k)$ are nonnegative, then the necessary optimality conditions are satisfied. If, in addition, $Z_{uu}(k) \succ 0$ then a local optimal solution has been reached. In the other case, one inequality constraint with negative multiplier is deleted from the set of active constraints.

2.2.2 InPA-SQP Approach

In section 2.1.5, a method is introduced to deal with large initial state perturbation that causes change in the set of active constraints. The idea is based on moving along the line which connects the nominal initial condition $x(0)$ to the point $x(0) + \delta x(0)$ until the status of one of the constraints changes. Then the optimal correction is calculated using Theorem 2.1.2, and this process is repeated for each intermediate point up to the point $x(0)+\delta x(0)$. Note that Theorem 2.1.2 can be used to calculate the op-

timal correction at each intermediate point only if the nominal solution corresponding to each intermediate point is optimal at each iteration. However, since the nominal solution used at each iteration of the algorithm introduced in Section 2.1.5 is the first order approximation of the optimal solution, the optimality condition $H_u(k) = 0$ may not be satisfied for the nominal solution.

In this section, a method that unifies NE method and SQP (with active set method formulation) is introduced to achieve faster convergence in calculating the perturbed optimal solution, given the nominal optimal solution. This algorithm combines the computational advantages of NE method and the optimality of the SQP solution, and provides a convenient formulation to calculate the perturbed solution while satisfying the optimality condition.

Let us assume that the optimal solution corresponding to the initial condition $x(0) = x_0$ is $u(k),\ k = 0, \cdots, N-1$, with the corresponding state trajectory $x(k),\ k = 0, \cdots, N-1$. As the first step to calculate the optimal solution corresponding to the initial condition $x(0) = x_0 + \delta x(0)$, let us assume that the perturbation $\delta x(0)$ is small enough so that the activity status of the constraints along the horizon does not change. Therefore, the results of Theorem 2.1.2 can be employed to determine the first order approximation of the optimal solution because $u(\cdot)$ and $x(\cdot)$ satisfy the optimality conditions.

If the activity status of the constraints changes when the initial condition is perturbed, that is the set of active constraints corresponding to the optimal solution is changed, the optimal solution corresponding to some intermediate initial conditions can be approximated, as illustrated in Section 2.1.5. According to the method proposed in Section 2.1.5, the first intermediate point is the closest initial condition to $x(0)$ on the line connecting the nominal initial condition $x(0)$ to the point $x(0)+\delta x(0)$ where the activity status of the constraints change.

Searching for the intermediate point is performed by constructing the set of different options using equations (2.69) and (2.70) and singling out the smallest one as illustrated in the second step of the algorithm proposed in Section 2.1.5. Once the first intermediate point is determined, the second intermediate point is determined using the algorithm proposed in Section 2.1.5 and the iteration is continued in the same way.

Since Theorem 2.1.2 provides the first order approximation of the optimal solution, the optimal solution corresponding to the first intermediate initial condition, which is calculated using Theorem 2.1.2, may not satisfy the necessary optimality conditions, namely the condition $H_u(k) = 0$. Therefore, the approximate optimal solution can not

be used as the nominal solution in Theorem 2.1.2 to calculate the next approximated optimal solution corresponding to the next intermediate initial condition.

To address the above issue the following ancillary remedy is introduced. We perform SQP iterations described in Section 2.2.1 to achieve the exact optimal solution corresponding to the intermediate initial condition using equation (2.76). Then the optimal solution can be used as the initial state in Theorem 2.1.2 to calculate the next intermediate initial state and its corresponding optimal solution. Therefore, after specifying each intermediate point, the following two steps can be performed:

First, the optimization problem (2.75) is solved iteratively, which results in equation (2.76), to obtain the optimal solution corresponding to the intermediate point. Second, Theorem 2.1.2 is used, to find the next intermediate point based on active set method and its corresponding optimal solution approximation.

Instead of applying the two-step method illustrated above, a unifying approach is introduced which exploits the SQP formulation to modify the large perturbation analysis so that optimality condition is considered. If one compare the two optimization problems (2.9) and (2.75), an interesting observation is that the optimization problem (2.75) is identical to (2.9) if $H_u(k) = 0$ and $\delta x(0) = \delta x_0$. By applying the same substitutions, the optimal solution (2.76) can be also converted to (2.46). This fact, for each intermediate initial state, can be interpreted as follows: equation (2.76) moves the approximated solutions toward the optimal solution when $x(0) = x_0$ while equation (2.46) moves the approximated solutions toward the approximated optimal solution when $x(0) = x_0 + \delta x(0)$.

Based on the above observations, the following formulation is proposed which merges the two optimization steps into one:

$$\begin{aligned}
&\delta u(k) = -[I\ 0]K_0(k)\begin{bmatrix} Z_{ux}(k)\delta x(k) + f_u^T(k)T(k+1) + H_u(k) \\ \tilde{C}_x(k)\delta x(k) \end{bmatrix} \\
&\delta x(k+1) = f_x(k)\delta x(k) + f_u(k)\delta u(k) \\
&\delta x(0) = \delta x_0,
\end{aligned} \tag{2.79}$$

where $H_u(k)$ is the derivative of the Hamiltonian function with respect to control u at the time instance k.

Using this approach at each iteration, if the nominal solution is not optimal such that $H_u(k)$ assumes considerable non-zero values, the optimal correction $\delta u(k)$ not only takes the initial state perturbation $\delta x(0)$ into account but it moves the nominal solution $u(k)$ and $x(k)$ in the decent direction calculated according to the SQP

method. Based on this approach, the steps of the algorithm described in Subsection 2.1.5 are followed, while in step 3 the perturbed optimal solution is calculated using equation (2.79). Through such combination, solving one optimization problem at each iteration is spared.

Remark 2.2.1. *If one consider the equality $x(0) - x_0 = 0$ as an equality constraint with $\lambda(0)$ being the associated Lagrange variable calculated using (2.8), then the augmented cost function can be reformulated by adding the term $\lambda(0)(x(0) - x_0)$. If the SQP method is applied to the reformulated problem, the equality constraint $\delta x(0) = x_0 - x(0)$ appears in the equation (2.76) and forms the equation (2.79). Therefore, the InPA-SQP is the result of applying SQP on the reformulated problem and consequently benefits from the convergence property that SQP provides.*

Remark 2.2.2. *It should be noted that both InPA-SQP and SQP methods solve the following necessary optimality conditions*

$$\begin{bmatrix} -I & H_{xx}(0) & H_{xu}(0) & C_x(0)^T & f_x(0)^T & 0 \\ 0 & H_{ux}(0) & H_{uu}(0) & C_u(0)^T & f_u(0)^T & 0 \\ 0 & I & 0 & 0 & 0 & 0 \\ f_x(0) & f_u(0) & 0 & 0 & 0 & -I \\ 0 & C_x(0) & C_u(0) & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta\lambda(0) \\ \delta x(0) \\ \delta u(0) \\ \delta\mu(0) \\ \delta\lambda(1) \\ \delta x(1) \end{bmatrix} = \begin{bmatrix} 0 \\ -H_u(0) \\ x_0 - x(0) \\ 0 \\ 0 \end{bmatrix} \tag{2.80}$$

$$\begin{bmatrix} -I & H_{xx}(k) & H_{xu}(k) & C_x(k)^T & f_x(k)^T & 0 \\ 0 & H_{ux}(k) & H_{uu}(k) & C_u(k)^T & f_u(k)^T & 0 \\ f_x(k) & f_u(k) & 0 & 0 & 0 & -I \\ 0 & C_x(k) & C_u(k) & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta\lambda(k) \\ \delta x(k) \\ \delta u(k) \\ \delta\mu(k) \\ \delta\lambda(k+1) \\ \delta x(k+1) \end{bmatrix} = \begin{bmatrix} 0 \\ -H_u(k) \\ 0 \\ 0 \end{bmatrix} \tag{2.81}$$

for $k = 1, \cdots, N-1$, *and*

$$\begin{bmatrix} -I & \Phi_{xx}(N) \end{bmatrix} \begin{bmatrix} \delta\lambda(N) \\ \delta x(N) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{2.82}$$

The difference is that in the InPA-SQP, initially, $H_u(k) = 0$, *for* $k = 0, \cdots, N-1$ *and* $x_0 - x(0) \neq 0$. *However, in SQP, initially,* $H_u(k) \neq 0, k = 0, \cdots, N-1$ *and* $x_0 - x(0) = 0$. *In both methods, the iterations continue until the norm of the vector* Λ *is sufficiently small where*

$$\Lambda = [\ 0\ , -H_u(0), (x_0 - x(0))^T,\ 0\ ,\ 0\ ,\ 0\ , H_u(1),\ 0,\ 0,\ \cdots,\ 0,\ 0\ ]. \tag{2.83}$$

We can show that a small change in the initial condition can lead a trajectory far away from the optimal one, due to large $H_u(\cdot)$ *[59]. For the case of systems with fast dynamics, it is noted that the norm of* Λ *for SQP method is initially considerably larger than that for InPA-SQP. So it can be expected that InPA-SQP converge to the optimal solution faster than SQP.*

2.2.3 MPC implementation using InPA-SQP approach

The InPA-SQP approach introduced in the previous section can be employed to reduce the computational time of solving the MPC optimal control problem comparing to the conventional SQP-based approach.

As illustrated before, according to MPC strategy, at time instant k with observed state $x(k)$ the optimization problem $\mathscr{P}(x(k))$ (defined in (1.3)) is solved, rendering the optimal control sequence

$$\mathbf{u}^*(x) = \{u_0^*(x), u_1^*(x), ..., u_{N-1}^*(x)\}, \tag{2.84}$$

and the model predictive control law

$$h(x) := u_0^*(x). \tag{2.85}$$

At the time instant $k+1$, the state $x(k+1)$ is observed and the optimal control problem $\mathscr{P}_N(x(k+1))$ must be solved. It should be noted that by the time instant $k+1$, the solution to the problem $\mathscr{P}_N(x(k))$ is available, which can be exploited to improve the efficiency of optimization. The MPC implementation strategy which is

described in the following uses the optimal solution calculated in the previous step k to approximate the optimal solution at the time $k+1$ so as to reduce the required computational time for solving the optimization problem. Defining

$$dx(k) := x(k+1) - x(k) \tag{2.86}$$

as the initial state perturbation, one can use the InPA-SQP algorithm to approximate the solution of the problem $\mathscr{P}(x(k+1))$ using the solution of $\mathscr{P}(x(k))$. If the approximated optimal solution is close to the actual optimal solution, that is

$$H_u(k) \approxeq 0,$$

then the approximated solution is applied. Otherwise, more SQP iteration is performed to achieve the optimal solution. Since the InPA-SQP method takes into account the initial state disturbance and optimality simultaneously, it is expected that the computational time is reduced considerably comparing to the SQP optimization with negligible effect on the performance of the MPC. In addition, it should be noted that calculating the perturbed optimal solution using the InPA-SQP method can also be used as the compensation part of the forecasting MPC [107].

2.2.4 Numerical results

As test cases, the InPA-SQP is employed to solve the optimal control problem associated with the Model Predictive Control (MPC) in two examples: an inverted pendulum on a cart as a toy example and a ship steering problem as an application. Computational advantages of the proposed approach are clearly demonstrated through examples in numerical computation.

Inverted Pendulum

First, the MPC is implemented using the SQP optimization method as well as the InPA-SQP approach, described so far in this chapter, on an inverted pendulum on a cart to compare the computational time and performance corresponding to each of these methods. The following model, taken out from [110], is used for numerical simulation:

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{-0.5(x_2^2 sinx_1 cosx_1 + sinx_1)}{1 + 0.5sin^2(x_1)} - \frac{cosx_1}{1+0.5sin^2(x_1)}u \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{1.5(x_2^2 sinx_1 - cosx_1 sinx_1)}{1 + 0.5sin^2(x_1)} + \frac{3}{1 + 0.5sin^2(x_1)}u
\end{aligned}
$$

where x_1, x_2 are the angle and angular velocity of the pendulum (see Figure 2.2), x_3, x_4 are the cart position and velocity, respectively. The control u is the force applied to the cart, as shown in Figure 2.2.

The control objective is to keep the pendulum at the upright position and keep the cart at the origin. We assume that this is a computational resource limited application where the processor and communication channels have sample interval of 100 ms and the computation plus communication delays could add up to as much as 60 $msec$. In addition, there is uncertainty up to 40% in the length of pendulum. The length of pendulum is 1.47 m. Here, a saturation limit on the control input is imposed to limit $u(\cdot)$ to be in the range $[-520\ 520]$.

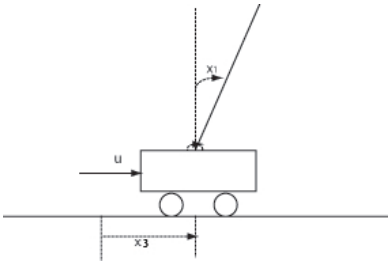

Figure 2.2: Inverted pendulum on cart: an example

In Figure 2.3, the result of MPC using SQP with active set method is shown by the solid line while the result of implementing MPC using the InPA-SQP is shown by the dot line. The threshold for $H_u(k)$ according to which the iterations is terminated in both optimization methods is 10^{-3}. Figure 2.4 compares the cumulative computation time of SQP method and InPA-SQP method at each time step. It can be seen from Figures 2.3 and Figure 2.4 that the two methods yield almost equal performance while the average computational time using the InPA-SQP approach is 40% less than that of SQP with the active set method. Note that the simulations are performed on a computer with Intel(R) CPU @ 1.83GHz and computation time is measured using CPU time.

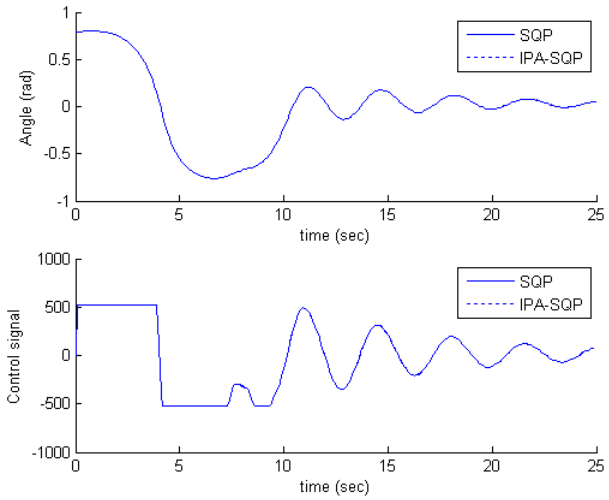

Figure 2.3: Implementing MPC using SQP with active set method and InPA-SQP approach.

Ship Steering Problem

The following ship model, taken out from [111], is used for numerical simulation:

$$\begin{aligned}
\dot{x}_1 &= x_5 cos(x_3) - (r_1 x_4 + r_3 x_4^3) sin(x_3), \\
\dot{x}_2 &= x_5 sin(x_3) + (r_1 x_4 + r_3 x_4^3) cos(x_3), \\
\dot{x}_3 &= x_4, \\
\dot{x}_4 &= -a x_4 - b x_4^3 + c u_r, \\
\dot{x}_5 &= -f x_5 - W x_4^2 + u_t,
\end{aligned} \tag{2.87}$$

where x_1 and x_2 are the ship position (in nautical miles (nm)) in the $X_1 - X_2$ plane, x_3 is the heading angle (in radians (rad)), x_4 is the yaw rate (rad/min), and x_5 is the forward velocity (nm/min). The two control inputs are: the rudder angle u_r (rad), and the propeller's thrust u_t (nm/min^2). Moreover, it is assumed that all states are measurable.

The model parameters are summarized in Table I. With these parameters, the ship has a maximum speed of .25 nm/min = 15 knots for a maximum thrust of 0.215 nm/min^2. For maximal rudder angle of 35^o, the stationary rate of turn is 1^o/sec.

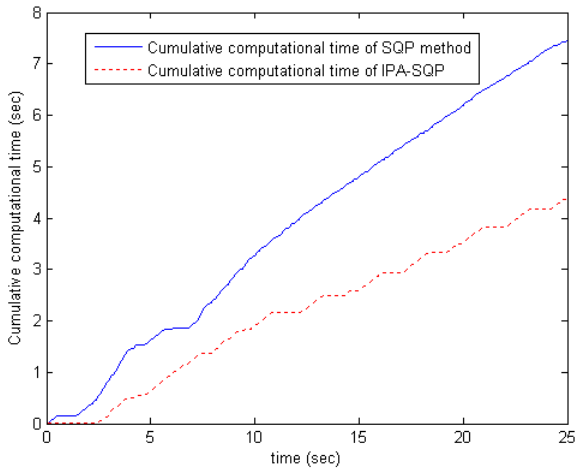

Figure 2.4: Cumulative computation time of SQP with active set method and InPA-SQP approach for inverted pendulum on cart.

Table I: Constant parameters of ship model

Parameter	value	unit
a	1.084	1/min
b	0.62	min/rad^2
c	3.553	$1/\text{min}^2$
r_1	-0.0375	nm/rad
r_3	0	$\text{Nm.min}^2/\text{rad}^3$
f	0.86	1/min
W	0.067	nm/rad^2

The discrete-time model of ship dynamics is derived using Euler approximation with sampling period $T = 0.1$ *sec*. The target position is described by a circle with a radius 0.1 (nm) around the origin. To minimize the energy consumption during the maneuvering, define

$$\begin{aligned} L(x(k), u(k)) &= 0.1u_r(k)^2 + 10u_t(k)^{3/2} \\ \Phi(x) &= 2000(x_1^2 + x_2^2) \end{aligned} \tag{2.88}$$

and with $N = 140$ as the length of horizon. The resulting MPC optimization problem

is

$$\min_{u(\cdot),x(\cdot)} \sum_{k=0}^{139} (0.1u_r(k)^2 + 10u_t(k)^{3/2}) + 2000(x_1(140)^2 + x_2(140)^2) \tag{2.89}$$

subject to constraints:

$$\begin{aligned} 0.02 \leq u_r(k) \leq 0.61\, rad \\ -0.215 \leq u_t(k) \leq 0.215\ nm/min^2. \end{aligned} \tag{2.90}$$

The total number of optimization variables, including states and inputs, for the length of horizon $N = 140$, is 978, which is substantial from computational point of view. The initial optimal solution for $k = 0$ is calculated off-line using SQP algorithm.

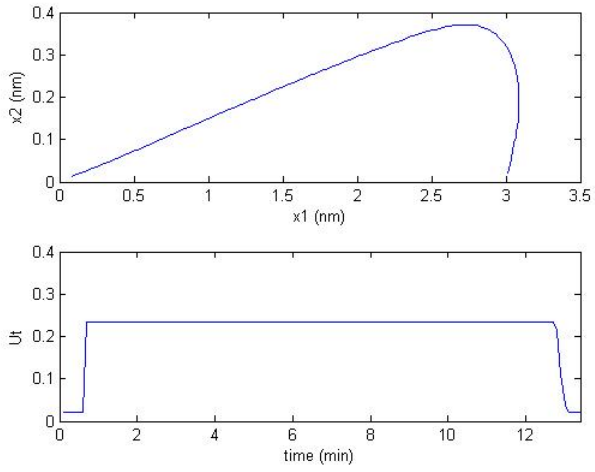

Figure 2.5: Implementing MPC using SQP with active set method and InPA-SQP approach on ship.

For a fair comparison, the formulation proposed in Section 2.2.1 is implemented for SQP, which is equivalent to the InPA-SQP in the absence of initial state perturbation. Simulations are performed on a computer with Intel(R) CPU @ 1.83GHz and computation time is measured using CPU time and controller code is implemented in Matlab.

Figure 2.5 shows the ship trajectory in the $X_1 - X_2$ plane and the propeller's thrust using both SQP and InPA-SQP for initial condition of $x(0) = [3, 0, \pi/3, 0, 0.25]$. The two solutions overlap in Figure 2.5 as they are nearly identical. The computational time of the two methods are compared in Figure 2.6. The InPA-SQP results in almost 280% reduction in the average computational time when compared with the SQP.

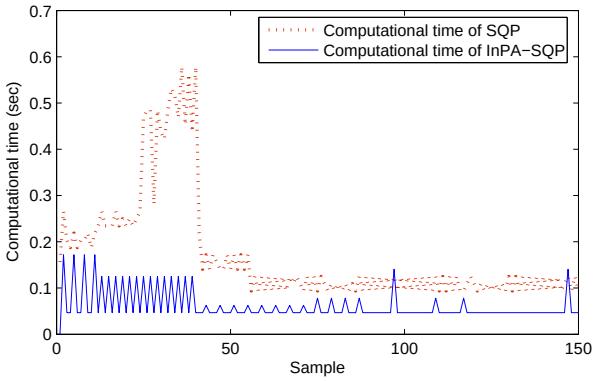

Figure 2.6: Computational time of SQP with active set method and InPA-SQP approach for ship steering problem.

In the next section, another NE-based approach is introduced for MPC implementation.

2.3 Forecasting MPC (FMPC) with local compensation: an application of the perturbation analysis solution

In this section, exploiting the NE method described in Subsections 2.1.1-2.1.3, the so-called forecasting MPC implementation strategy is introduced [36, 107], which aims at eliminating the computational time delay of solving the MPC-optimization problem.

In order to compensate for the computational delay, the solution to the problem $\mathscr{P}(\hat{x}(k))$ is pre-computed within the interval $[k-1, k]$ where $\hat{x}(k) = f(x(k-1), u(k-1))$ is the predicted state at the time instant k using the measurement $x(k-1)$ and $x(k-1)$ is the measured state at the time instant $k-1$. Denoting the optimal control sequence as $\mathbf{u}^*(\hat{x}(k))$, the FMPC control law at the time instant k is

$$u(k) = \mathbf{u}_1^*(\hat{x}(k)), \tag{2.91}$$

where $\mathbf{u}_1^*$ denotes the first element in the sequence $\mathbf{u}^*$. This strategy assures that the control input $u(k)$ is available at time k, therefore providing an MPC action with minimal computational time delay.

In the presence of disturbances or model uncertainties, however, the predicted states $\hat{x}(k)$ and $x(k)$ may not match exactly. Using the perturbation analysis de-

veloped in the previous sections, the optimal solution corresponding to $x(k)$ can be derived by making a correction to the control signal (2.91) (computed in advance) with a perturbation term δu, that is computed once the state measurement is taken at time instant k. Namely,

$$u(x(k)) \approx \mathbf{u}_1^*(\hat{x}(k)) + \delta u_1(\delta x_0) \tag{2.92}$$

where $\delta x_0 = x(k) - \hat{x}(k)$. The second term in (2.92), which is designed to compensate the effects of disturbances and unmodeled dynamics in the predicted solution, is referred to as the local compensation in [107] and is calculated using the NE solution described in Subsection 2.1.3. Since δu_1 can be computed much faster than the solution of the original optimization problem, the effects of uncertainty and disturbance can be immediately compensated.

It should be noted that the FMPC law (2.91) can also be calculated using the perturbation analysis approach. In this case, $\delta x_0 = \hat{x}(k) - x(k-1)$. Several extensions of our basic approach can be proposed. One is to situations when the solution to the optimization problem takes larger than one sampling interval or even random time interval to compute. Opportunities also exist to combine this approach with the block MPC in [5]. With these approaches, the optimization problem can be solved less frequently, making an implementation of MPC strategy feasible for systems with fast dynamics or with slow computing hardware.

2.4 Summary

So far in this chapter, the InPA-SQP solver is introduced and two numerical examples are provided which demonstrate the efficiency of the method. In the next chapter, two experimental applications, a DC/DC converter and ship maneuvering, are presented and details specific to how these applications are treated using InPA-SQP are discussed.

Chapter 3

Applications of MPC using InPA-SQP

In this chapter, it will be shown how the InPA-SQP algorithm is employed successfully in two experimental applications. The first application is output voltage regulation of a DC/DC converter on an experimental testbed at the University of Michigan [69]. The second one is path following of a model ship constructed in the Marine Hydrodynamic Laboratories (MHL) at the University of Michigan. Details of models, experimental setups, and experimental results for DC/DC converter and the model ship are presented in Section 3.1 and 3.2 respectively.

3.1 DC/DC converter

The full bridge DC/DC Converter was initially proposed [63], [64] for both high power density and high power applications. It is very attractive because of its zero voltage switching, low component stresses, and high power density features. Moreover, its high frequency transformer prevents fault propagation and enables a high output/input voltage ratio. Therefore, with a full bridge DC/DC converter as the power conditioning system, low voltage energy system can be applied to high DC voltage applications, such as the DC zonal electrical distribution system of an all electric ship [67]. To investigate the voltage regulation of a full bridge DC/DC based power conditioning system, an experimental testbed was developed at the University of Michigan to support model development and to facilitate a model based control design approach [68]. Figure 3.1 depicts the configuration of the power stage of a full bridge DC/DC converter, while parameters of the full bridge DC/DC converter are shown in Table 3.1.

Several challenges arise for the DC/DC converter control design. First, the power devices of the DC/DC converters have very complicated time varying switching behavior which defines the shape of the inductor current, making the dynamic model

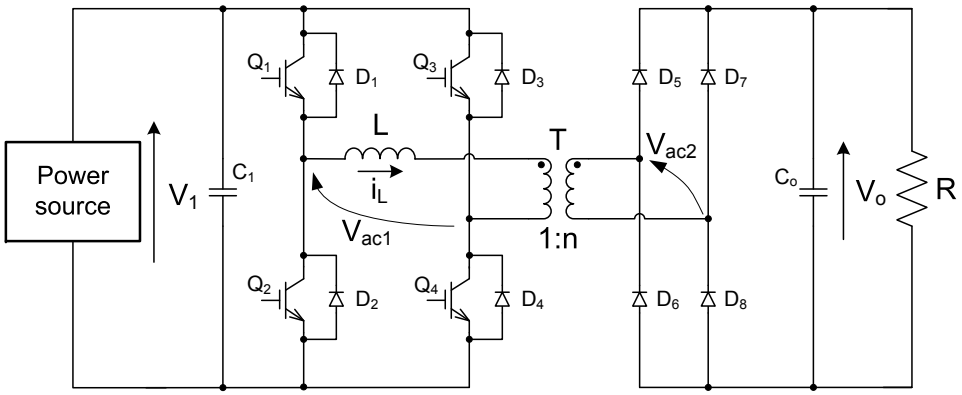

Figure 3.1: A full bridge DC/DC converter.

Table 3.1: Parameters of the testbed prototype

Item	Parameter
Inductor L	$10.5\mu H$
Capacitor C_o	$1410\mu F$
Transformer turn ratio n	2
Switching period T	$100\mu s$
V_1 (nominal)	$60V$
Desired V_o	$80V$
Nominal load R	6.4Ω
Rated power	$1000W$

development of power converters a challenge. Second, DC/DC converters as power conditioning devices typically have a wide range of operating conditions, further complicating the control design. Furthermore, the control input is bounded due to physical limitations of power converters. Finally, safe operation requirements such as peak current limitation may impose additional nonlinear constraints.

Traditionally, there are two classes of algorithms for DC/DC converter control, namely the voltage mode control and current mode control. Voltage mode control achieves voltage regulation through a single-loop voltage control scheme. To limit the current during transient operation within safe operation range, the feedback control gain must be carefully chosen, otherwise an additional protection circuit has to be incorporated. In addition to a voltage feedback loop, current mode control employs an inner inductor current feedback loop to improve performance. Performance enhancements, including superb line regulation and inherent over-current protection, can be achieved for current mode control. However, current mode control has a subharmonic oscillation problem when the duty ratio is greater than 0.5 [84]. Besides, this method requires inductor current sensing, which increases system cost and tends

to have noise sensitivity problems. The development of advanced control algorithms, together with the increased computational power of microprocessors, enables us to deal with the control problem from a new perspective. For example, Model Predictive Control (MPC) has been implemented in power converters [85, 88] and in an electric drive system for direct torque control [86, 87]. For the full bridge DC/DC converter under investigation, the peak current protection problem can be formulated as a constraint for an optimal control problem, which can be effectively dealt with using MPC.

This section is concerned with the closed loop system performance of the MPC schemes for a full bridge DC/DC converter. The control objective is to regulate the output voltage without violating the peak current constraint. The voltage regulation problem is formulated as an MPC problem using a nonlinear model to predict the future plant behavior. The peak current protection requirement is formulated as a nonlinear constraint. To achieve $300\mu s$ sampling time and handle the nonlinear constraint, the InPA-SQP method is employed to solve the constrained optimal control problem. The InPA-SQP solver can significantly improve computational efficiency while effectively handling the nonlinear constraints, making the implicit MPC feasible for a power electronics system with very fast dynamics. The experimental results reveal that the MPC algorithms successfully achieved voltage regulation and peak current protection.

The rest of the section is organized as follows: In Subsection 3.1.1, the inductor peak current constraint of the full bridge DC/DC converter will be presented. Subsection 3.1.2 is devoted to an observer design for states and parameter estimation using a large signal dynamic model. Subsection 3.1.3 focuses on MPC problem formulation. Experimental results will be presented in Subsection 3.1.4, followed by conclusions in Section 3.1.5.

3.1.1 Inductor peak current constraint

The full bridge DC/DC converter is typically modulated by the phase shift modulation signals $V_{Q_1} \sim V_{Q_4}$ shown in Figure 3.2(a), where $\beta \in [0, 1]$ is the normalized phase shift between the two half bridges composed of Q_1/Q_2 and Q_3/Q_4, respectively. By shifting the phase between the two half bridges, different combinations of V_{ac1} and V_{ac2} can be applied to shape the current i_L and consequently to manipulate the power flow. Based on the shape of i_L, there are two operation modes for the full bridge converter, namely the Discontinuous-Conduction-Mode (DCM) and the Continuous-Conduction-Mode (CCM) [68].

For a full bridge DC/DC converter operating with Discontinuous-Conduction-Mode (DCM), the ideal voltage waveforms of V_{ac1} and V_{ac2} are shown in Figure 3.2(b) and Figure 3.2(c). The voltage across the inductor is $V_L = V_{ac1} - V_{ac2}/n$, leading to the ideal inductor current i_L slope at each segment shown in Figure 3.2(d). Given the inductor current slopes shown in Figure 3.2(d), the peak current ($i_L(t_0 + \frac{\beta T}{2})$) for DCM can be calculated as [69]:

$$i_L(t_0 + \frac{\beta T}{2}) = \frac{(nV_1 - V_o)\beta T}{2nL}. \tag{3.1}$$

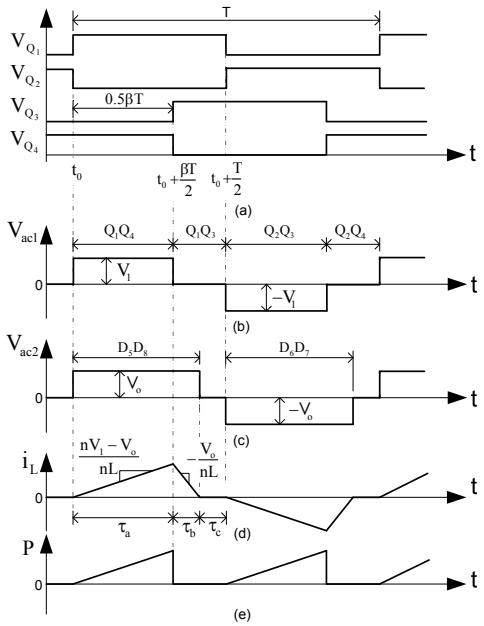

Figure 3.2: Modulation sequence and ideal waveforms of the full bridge DC/DC converter for DCM.

Similarly, the peak current ($i_L(t_0 + \frac{\beta T}{2})$) for Continuous-Conduction-Mode (CCM) can be calculated as:

$$i_L(t_0 + \frac{\beta T}{2}) = \frac{(nV_1 - V_o)(V_o + nV_1\beta)T}{8nLV_1}. \tag{3.2}$$

The operating mode of the DC/DC converter is determined by V_1, V_o and β. For different combinations of V_1 and V_o, the phase shift boundary line $L_{\beta b}$ between the CCM and DCM can be calculated as follows if we set $\tau_c = 0$ for DCM:

$$L_{\beta b} = \{(\beta, V_1, V_o) | \beta = \frac{V_o}{nV_1}\}. \tag{3.3}$$

Moreover, let i_{pk} denote the maximum tolerable peak current of the converter.

Using (3.1) and (3.2), one can determine the limits on the phase shift to avoid over-peak-current. If the converter operates with CCM, the phase shift constraint curve $L_{\beta c}$ can be calculated from (3.2) as follows:

$$L_{\beta c} = \{(\beta, V_1, V_o) | \beta = \frac{8Li_{pk}}{T(nV_1 - V_o)} - \frac{V_o}{nV_1}\}. \tag{3.4}$$

Similarly, if the converter operates with DCM, the phase shift constraint curve $L_{\beta d}$ can be calculated from (3.1) as follows:

$$L_{\beta d} = \{(\beta, V_1, V_o) | \beta = \frac{2nLi_{pk}}{T(nV_1 - V_o)}\}. \tag{3.5}$$

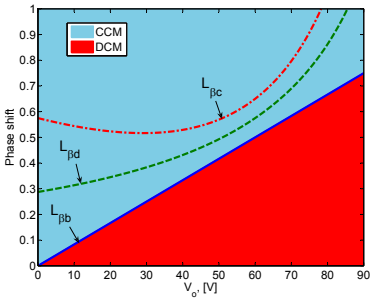

Figure 3.3: DCM/CCM boundary line $L_{\beta b}$ and peak current constraint curves $L_{\beta d}$ and $L_{\beta c}$ for $V_1 = 60V$ and $i_{peak} = 75A$.

Figure 3.3 shows the phase shift boundary line $L_{\beta b}$ and the peak current constraint curves $L_{\beta d}$ and $L_{\beta c}$ for $V_1 = 60V$ and $V_o = 0V \sim 90V$. Note that: (a) the full bridge DC/DC converter operates with the CCM if the phase shift is larger than the corresponding boundary value; (b) the peak current constraint curves $L_{\beta d}$ and $L_{\beta c}$ are calculated using equations (3.4) and (3.5) for $i_{pk} = 75A$. For our system with a nominal output power of $1000W$, the phase shift at the nominal operating point is 0.62 which is smaller than the boundary value 0.67. Therefore, the converter operates with the DCM at steady state for the nominal output power. From Figure 3.3, the DCM peak current constraint curve $L_{\beta d}$ is always above the boundary line $L_{\beta b}$, so the peak current constraint will not be violated if the power converter operates with DCM at steady state. However, for the cases of starting process and overload, the power converter operates at CCM, where the CCM peak current constraint may be violated. Therefore, an active constraint enforcement mechanism needs to be incorporated to protect the converter.

3.1.2 Dynamic model development and observer design

Given the challenges the control of power converters faces, it is desirable to employ a model based control design approach to achieve satisfactory closed loop system performance. Since the full bridge DC/DC converter has a wide operating range, it is necessary to derive a large signal dynamic model for the system to facilitate the model based control design. For this work, an averaged dynamic model as developed for other power converters [89, 90, 92] is needed for control design. Following the same technique in [90], the dynamic model of the full bridge DC/DC converter can be derived as:

$$\frac{d\bar{i}_L}{dt} = \frac{\beta V_1}{L} - \frac{4\bar{i}_L \bar{V}_o}{\beta T(nV_1 - \bar{V}_o)}, \tag{3.6}$$

$$\frac{d\bar{V}_o}{dt} = \frac{\bar{i}_L}{nC_o} - \frac{\bar{V}_o}{RC_o}, \tag{3.7}$$

$$y = \bar{V}_o. \tag{3.8}$$

Note that: $\bar{i}_L$ and $\bar{V}_o$ represent the average current and the average output voltage over a switching period. The output capacitor acts as a filter for the ripple on the output voltage. Hence, output voltage samples, with sampling rate as much as switching frequency, represent average output voltage over switching period.

For the dynamic system represented by (3.6)-(3.8), the implementation of advanced control strategies requires a current sensor to obtain the average current $\bar{i}_L$ for state feedback. On one hand, the current sensor must have high bandwidth to accurately reconstruct the current signal. On the other hand, due to electromagnetic interference, it is often necessary to use a low-pass filter to remove noise. However, a low-pass filter typically introduces additional phase lag for the closed loop system. To overcome those drawbacks, a nonlinear observer is used to estimate the average current $\bar{i}_L$ while keeping the voltage $\bar{V}_o$ as the only measured variable. The nonlinear observer is expressed as follows:

$$\frac{d\hat{\bar{i}}_L}{dt} = \frac{\beta V_1}{L} - \frac{4\hat{\bar{i}}_L \hat{\bar{V}}_o}{\beta T(nV_1 - \hat{\bar{V}}_o)} + H_1(y - \hat{y}), \tag{3.9}$$

$$\frac{d\hat{\bar{V}}_o}{dt} = \frac{\hat{\bar{i}}_L}{nC_o} - \frac{\hat{\bar{V}}_o}{RC_o} + H_2(y - \hat{y}), \tag{3.10}$$

$$\hat{y} = \hat{\bar{V}}_o. \tag{3.11}$$

To derive the gains H_1 and H_2 for the nonlinear observer, the plant is linearized at the

desired operating point. The faster eigenvalue is around -800. We chose $H_1 = 2759$ and $H_2 = 2859$ to place the eigenvalues of the observer at -2000 to achieve fast convergence of the observer.

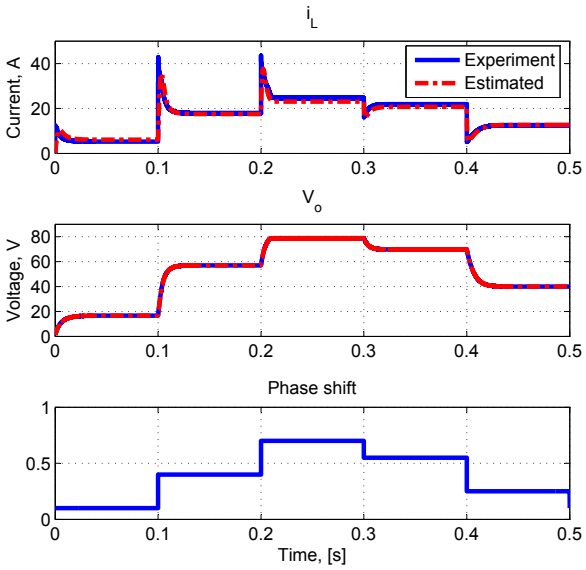

Figure 3.4: Comparison of estimated and measured states.

Figure 3.4 compares the estimated states with the actual measured states. The observed states closely track the real states for different operating points. This figure also confirms the accuracy of the nonlinear model.

3.1.3 MPC formulation

This section presents the formulation of the MPC controller for the voltage regulation problem of the full bridge DC/DC converter. The dynamic system represented by (3.6)-(3.8) can be easily linearized with nominal value $x^o = [25, 80]^T$ and $u^o = 0.62$. Let $x_1 = \bar{i}_L - 25$, $x_2 = \bar{V}_o - 80$ and $u = \beta - 0.62$, the system can be transformed into its discrete-time version for a specific sampling time:

$$x(k+1) = f(x(k), u(k)) := Ax(k) + Bu(k), \tag{3.12}$$

$$y(k) = Fx(k). \tag{3.13}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $F \in \mathbb{R}^{m\times n}$. Note that $n = 2$ and $m = 1$ for the system under investigation.

For a given i_{peak}, the CCM peak current (3.2) must satisfy

$$\frac{(nV_1 - V_o)(V_o + nV_1 a)T}{8nLV_1} \leq i_{peak}. \tag{3.14}$$

(3.14) can be rewritten in terms of the state and control variables as:

$$E_1(x(k), u(k)) \leq 0. \tag{3.15}$$

where,

$$\begin{aligned} E_1(x(k), u(k)) = &\frac{(nV_1 - (x_2^o + x_2(k)))((x_2^o + x_2(k)))}{V_1} \\ &+ \frac{(nV_1 - (x_2^o + x_2(k)))(nV_1(u^o + u(k)))}{V_1} - \frac{8nLi_{peak}}{T}. \end{aligned} \tag{3.16}$$

Then the MPC online optimization problem can be formulated as follows: at the time instant k, the state of the system, $x(k)$, is observed and the following optimal control problem $\mathscr{P}_N(x(k))$ is solved

$$\mathscr{P}_N(x_0) : V_N^*(x_0) = \min_{\mathbf{u}} \{V_N(x_0, \mathbf{u})\} \tag{3.17}$$

$$V_N(x_0, \mathbf{u}) = \sum_{k=0}^{N-1} G(x(k), u(k)) + \Phi(x(N)) \tag{3.18}$$

subject to

$$x(k+1) = f(x(k), u(k)), \quad f : \mathbb{R}^{n+m} \to \mathbb{R}^n; \tag{3.19}$$

$$x(0) = x_0, \ x_0 \in \mathbb{R}^n; \tag{3.20}$$

$$E(x(k), u(k)) \leq 0, \ E : \mathbb{R}^{n+m} \to \mathbb{R}^l. \tag{3.21}$$

where,

$$\mathbf{u} = \{u(0), u(1), ..., u(N-1)\}, \tag{3.22}$$

is the control sequence,

$$x(k) := x^{\mathbf{u}}(k; x), \tag{3.23}$$

is the state trajectory at time instant k resulting from an initial state x_0. Since the separation principle does not hold for nonlinear systems with observer, here we rely on inherent robustness of MPC to deal with the effect of the observer. The incremental

cost is

$$G(x(k), u(k)) = x(k)^T Q x(k) + u(k)^T W u(k), \tag{3.24}$$

and $\Phi(x(N))$ is the penalty for the final states, $Q \in \mathbb{R}^{n \times n}$ and $W \in \mathbb{R}^{m \times m}$ are the corresponding weighting matrices which are used to penalize the deviation of the output and the control input to their corresponding desired value, N is the prediction horizon, and $E(x(k), u(k))$ is the constraint matrix and can be written as follows with $l = 3$:

$$\begin{bmatrix} u(k) - (1 - u^o) \\ -u(k) - u^o \\ E_1(x(k), u(k)) \end{bmatrix}. \tag{3.25}$$

Note that each component in (3.25) is bounded above by zero.

Since the full bridge DC/DC converter has the millisecond level time constant, a rational choice of the sampling time is between $100\mu s$ and $200\mu s$ [91]. The length of the prediction horizon N is a basic tuning parameter for MPC controllers. Generally speaking, the closed loop system performance improves as N increases. However, additional computational effort associated with a long horizon could be troublesome for implicit MPC of power electronics systems. We choose $150\mu s$ as the sampling time for the controller and $N = 10$ as the prediction horizon. The weighting matrices Q and W are the main tuning parameters of the quadratic cost function (3.18) to shape the closed-loop response for desired performance. The closed loop performance criteria is defined: (1) to achieve fast output response with small output overshoot; and (2) to avoid high frequency control input oscillation which might cause high slew rate for the inductor current and high stress for switching components. We evaluate the performance to different combinations of weighting matrix using a virtual hardware. The virtual hardware is developed using MATLAB/Simulink/SimPowerSystems toolbox and has the same parameters as the real hardware. The preliminary evaluation results lead to the choice of $Q = [0\ 0; 0\ 0.01]$ and $W = 1$. Furthermore, the final states $x(N)$ is not penalized, meaning $\Phi(x(N)) = 0$.

Given the fast dynamics of the converter, a fast algorithm needs to be applied to solve the above optimization problem online in real-time. Therefore, InPA-SQP method is employed for experimental implementation of MPC. The next section shows experimental results of such implementation.

3.1.4 Experimental validation

The goal of this section is to present the experimental results to validate the effectiveness of the MPC controller using the InPA-SQP as the optimization solver.

Experimental setup

Figure 4.5(a) demonstrates a DC hybrid power system testbed which includes RT-LAB® system, power converters, power sources and electronic loads. Figure 4.5(b) shows the full bridge DC/DC converter(DC/DC1) under investigation which delivers power from power source1 to load1. The RT-LAB® system is a PC cluster based platform which can perform real-time simulation, hardware in the loop test and rapid control prototyping for large scale system. For this work, the RT-LAB® system serves the following three functions: (1) as the real-time simulator to control the programmable power source1 such that it emulates the behavior of a PEM fuel cell; (2) as the rapid control prototyping unit to generate the $10kHz$ modulation signals for the full bridge DC/DC converter according to feedback information; (3) as the data acquisition device to acquire and store experimental data to enable detailed offline analysis. Note that only one target (Target1) is used in this application although our RT-LAB® system has four targets. Parameters of the full bridge DC/DC converter are shown in Table.3.1.

Experimental results

First, the closed-loop performance is investigated in the presence of a large step change in the load resistance R. Figure 3.5 compares simulation and experimental waveforms for a step-down change of R when the control algorithm is applied to control the nonlinear model represented by (3.6)-(3.8) and the full bridge DC/DC converter shown in Figure 4.5(b). Initial R is 12.8Ω ($500W$ output power). A step-down change of the load resistance R is then applied to demand $1000W$ output power (nominal). The transient responses of the MPC applied to both the large signal dynamic model and the actual DC/DC converter are essentially the same. Moreover, for both of the loads, the output voltage is regulated to the desired value, which confirms the robustness of the control scheme. The peak of the current of inductor does not get close to the upper limit of 75 A. As can be seen, the peak of the current is less than 50 A and therefore, the constraint remains inactive during transient.

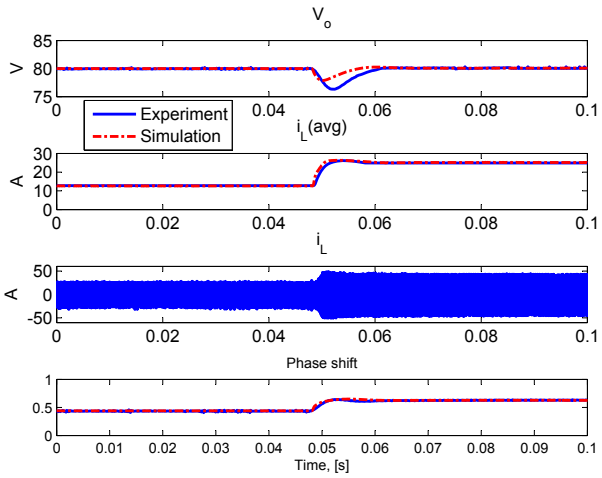

Figure 3.5: Simulation and experimental waveforms for a step-down change of R from 12.8Ω to 6.4Ω ($i_L(avg) = \bar{i}_L$).

Figure 3.6 shows the experimental waveforms for the starting process. The peak current is limited within the maximum tolerable value $75A$ while the output voltage is regulated to $80V$. From the third plot, the phase shift (control input) first hits the nonlinear constraint and then is constrained by the upper limit during the starting process.

Finally, Figure 3.7 shows the experimental waveforms for the over load case. During the steady state, the peak current is limited within the maximum tolerable value $75A$ although the peak current is slightly higher than $75A$ for about $1ms$ during the transient. This is partially due to the fact that a current sensor is not used in the control scheme. The output voltage drops from $80V$ to $32V$ during the transient since i_L is constrained. From the third plot, the phase shift (control input) first hits the upper limit and then is constrained by the peak current constraint.

The results reveal that the MPC controller successfully achieves voltage regulation and peak current protection. The successful implementation of MPC in real-time verifies that the InPA-SQP can significantly improve computational efficiency while gracefully handling the nonlinear constraint. Therefore, it is feasible to apply implicit MPC for fast dynamic systems such as the power electronics system if the InPA-SQP solver is applied.

3.1.5 Conclusion

In this section, the operation of the full bridge DC/DC converter is analyzed. Based on the analysis, a large signal dynamic model for the full bridge DC/DC converter

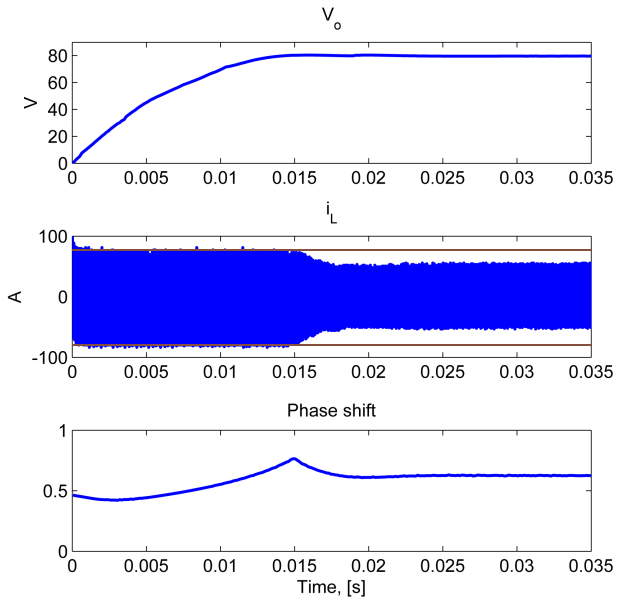

Figure 3.6: Experimental waveforms for starting process with $R = 6.4\Omega$ (nominal).

is developed. The voltage regulation problem of the converter is formulated in the context of MPC, where the peak current protection requirement is represented as a mixed state and control input constraint. To achieve sub-millisecond level sampling time and simultaneously handle the nonlinear constraint, the InPA-SQP method is employed to solve the constrained optimal control problem. The InPA-SQP solver can meet the computational efficiency demand while handling the nonlinear constraint. The effectiveness of the proposed control algorithm, including the peak current protection capability, has been verified with experimental results. In the next section, InPA-SQP is employed experimentally to control a model ship to follow a certain path.

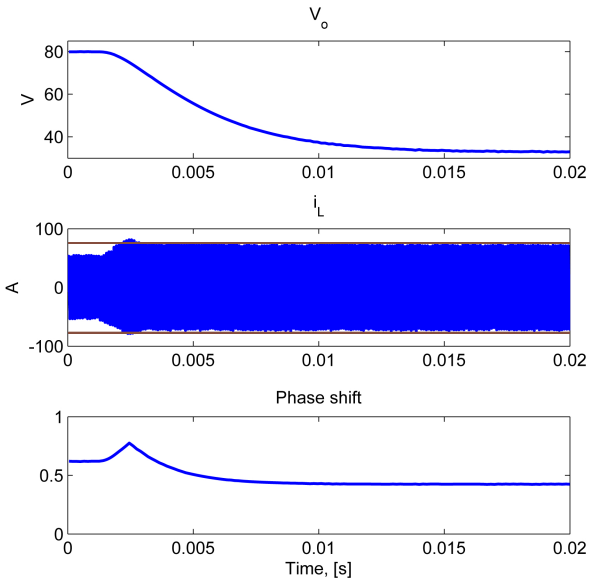

Figure 3.7: Experimental waveforms for over load $R = 1.6\Omega$.

3.2 Path following of the Model Ship

Controlling of marine surface vessels to follow a prescribed path or track a given trajectory has been a representative control problem for marine applications and it has attracted considerable attention from the control community [93]-[94]. One challenge for path following problem is that the surface vessel system is underactuated, as the path-following and heading control are often achieved using rudder angle as the only control input. Another challenge is the physical constraints imposed on the system such as rudder saturation. Model Predictive Control (MPC) is an attractive candidate to achieve zero cross tracking error and heading angle error via minimizing a suitable cost function while taking into account physical constraints. In this section, the effectiveness of MPC that is implemented via InPA-SQP method for path following of a model ship is experimentally demonstrated.

MPC has been employed for tracking control of marine surface vessels under rudder saturation constraint, [95], and for roll motion control with fin stabilizers [96]. In [98], the path-following control problem is considered where MPC is used and the performance is evaluated using simulations when the ship is subject to roll constraints and rudder saturation. This work focuses on the experimental validation of the algorithm, reported in [59], for path following control.

This section presents the MPC design using InPA-SQP for the path following problem for a model ship via rudder control. 3-DoF simplified nonlinear and linear models are adopted in the controller design and a corresponding 6-DoF nonlinear

container model is used in simulations in order to study and compare the performance of the MPC with the linear and nonlinear model. The InPA-SQP algorithm is used to implement both the linear and nonlinear MPC on a model ship to experimentally validate the algorithm and compare the performance. Finally the effects of delay in the feedback loop on control signals are analyzed via simulation and a delay compensation method, which is based on estimating the current state of the system in the MPC controller, is applied to mitigate oscillations in the rudder angle commands.

3.2.1 Ship dynamic model

The control objective is to achieve path following for a model ship constructed in the Marine Hydrodynamic Laboratories (MHL) at the University of Michigan. The model ship, a 1:16 replica of a replenish vessel whose principal parameters are shown in Table I, is actuated with two contra-rotating main propellers and two rudders aft. Propellers and rudders are actuated by two DC servo motors fitted with encoders and tachometers, respectively. Both the propeller speed and rudder angles are controlled by an embedded processor (PC 104) through proper mechatronic interfaces.

The control code is loaded on a PC 104 through which the control signal is generated for servo motors to position the rudder accordingly. However, when the model ship is tested in the towing tank located in the Marine Hydrodynamics Laboratories (MHL) of the University of Michigan, the GPS signals are not available (even they were available, the accuracy of GPS signals is not high enough for the model test). Four infra-red cameras are used, in lieu of the GPS system, to provide the feedback signals to the control system. A picture of the instrumented model is shown in Figure 3.8.

Table 3.2: Parameters describing the Model Ship.

Item	Symbol	Value
Length	L	1.60 m
Breadth	B	0.38 m
Height	H	0.17 m
Mass	m	38 kg
Inertia	I_z	2.7 kgm^2

The following model characterizes the dynamics of the model ship. The procedure

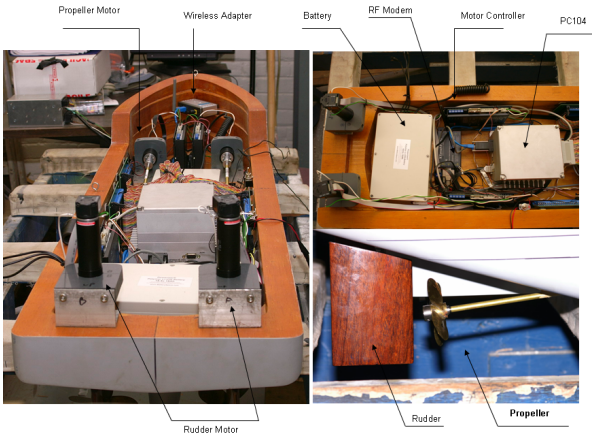

Figure 3.8: A system overview of the fully instrumented model ship.

of model derivation and experimental validation is explained in [99].

$$
\begin{aligned}
\dot{e} &= u\sin(\psi) + v\cos(\psi) \\
\dot{\phi} &= r \\
\dot{u} &= \frac{1}{m - X_{\dot{u}}}(mvr + X_u u + X_{uu}|u|u + X_{uuu}u^3 + 1.8902 \times 10^{-5} \times |U_p| \times U_p) \\
\dot{v} &= \frac{1}{m - Y_{\dot{v}}}(-mur + Y_v v + Y_r r + Y_{vv}|v|v + (-0.0298 + 1.435 \times 10^{-4} \times 350)\delta) \\
\dot{r} &= \frac{1}{I_z - N_{\dot{r}}}(N_v v + N_r r + (0.0227 - 1.0002 \times 10^{-4}350)\delta)
\end{aligned}
\tag{3.26}
$$

with the parameters shown in Table 3.3 where U_p is the propeller speed, e is the distance of the ship from the desired path, called cross tracking error, and u, v, r, and ψ are surge, sway, yaw velocities and heading angle respectively. The system is subject to saturation constraint on the rudder input δ with saturation constraint

$$-30^o \leq \delta \leq 30^o. \tag{3.27}$$

For way-point paths, the path following problem can be formulated as a regulation problem where the goal is steering the cross tracking error and heading angle error (3.26) to the origin. Given the constraints, MPC is employed to address the regulation problem. To proceed with the MPC design, first the following assumption is made:

Assumption 3.2.1. *The vessel surge velocity u is constant.*

Note that the propeller speed is not treated as a control variable in our path follow-

Table 3.3: Parameters of the dynamics of the Model Ship.

Parameter	Value
m	38.0
I_z	2.7
$X_{\dot{u}}$	-6.8558
X_u	3.0211
X_{uu}	-12.9059
X_{uuu}	2.7759
$Y_{\dot{v}}$	-17.5
Y_v	-20.5
Y_r	-0.835
Y_{vv}	-24.2
$N_{\dot{r}}$	-1.2
N_v	1.1965
N_r	-1.2522

ing problem formulation. Assumption 3.2.1 can be satisfied by properly controlling the propeller speed to maintain constant vessel surge speed. With Assumption 3.2.1, the state u can be eliminated and the following simplified model can be used for MPC design [99]:

$$\begin{aligned} \dot{v} &= 0.029975\delta - 0.31755v - 0.47213ur \\ \dot{r} &= -0.13155\delta - 0.2841r + 1.654175uv \\ \dot{e} &= u\cos(\psi) + v\sin(\psi) \\ \dot{\psi} &= r, \end{aligned} \tag{3.28}$$

with the nominal value $u = .4\ m/s$. The state of the system is $x = [v\ r\ e\ \psi]^T$ and δ is the rudder angle which acts as the only input to the system.

3.2.2 Path following using MPC

For the ship path following problem where the objective is to regulate the cross track and heading angle to zero, subject to the dynamic equation (3.28) and inequality constraint (3.27), the MPC optimization problem $\mathscr{P}_N(x(k))$ (defined by equation

1.3) is solved online where

$$\begin{aligned} L(x(k), u(k)) &= x(k)^T Q x(k) + u(k)^T R u(k), \\ \Phi(x(N)) &= \alpha x(N)^T S_f x(N), \\ Q &= diag([0\ 0\ 100\ 800]), \end{aligned} \tag{3.29}$$

and S_f is the solution of the matrix Ricatti equation associated with R and Q, i.e.,

$$S_f = Q + A^T(P - PB(R + B^T PB)^{-1} B^T P)A,$$

where $x(k+1) = Ax(k) + Bu(k)$ is the linearized version of the nonlinear dynamic (3.28). For MPC optimization, one can either use the nonlinear model (3.28), which leads to a non-linear MPC (NLMPC), or a linearized version of (3.28) which results in linear MPC (LMPC). In this work, both LMPC and NLMPC are considered and their performances are analyzed for ship path following under the given implementation hardware.

Consider the initial state of the system (3.26) as $x_0 = [2, 0, 0.5, 0, 0]^T$. To compare the linear and nonlinear model, the optimization problem $\mathcal{P}(x_0)$ is solved with $N = 200$ and sampling time 0.01, for both linear and nonlinear systems. Figure 3.9 shows the resulting open loop optimal control command for linear and nonlinear systems.

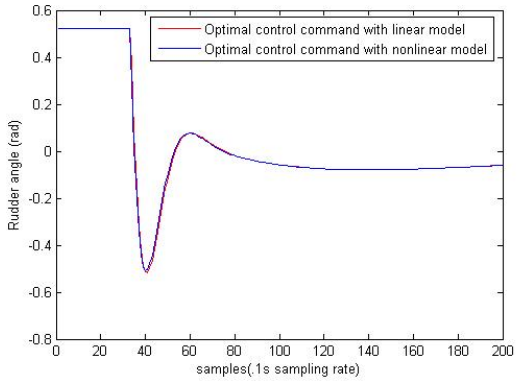

Figure 3.9: Open loop optimal control commands determined by nonlinear and linear MPC.

It is shown that the optimal solution for both systems are almost identical. When used in the MPC feedback control, the response of the closed-loop LMPC and NLMPC are shown in Figures 3.11 and 3.12 for the cross tracking error and rudder command respectively. Note that both LMPC and NLMPC are applied to the nonlinear model (3.26) in our simulations. It can be seen that while the performance of the two controllers are the same, the control commands for the nonlinear and linear MPC

are not similar. Accumulation of the small difference, shown in Figure 3.9, over the horizon is the cause of such difference.

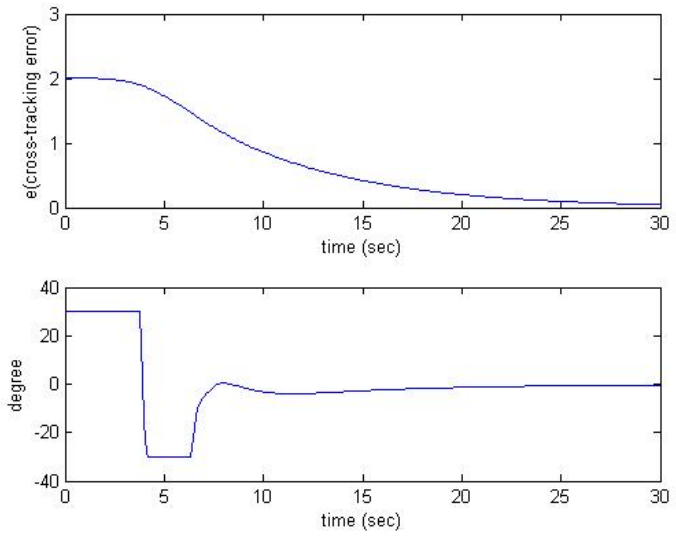

Figure 3.10: Cross tracking error and control command for nonlinear system with $N = 7$

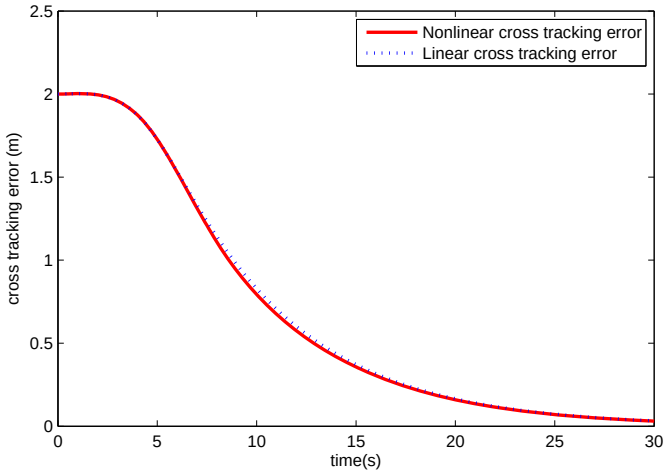

Figure 3.11: The resulting cross tracking error generated by MPC

Besides the performance, computational demand is another important consideration in deciding which class of MPC is more appropriate for our application. With the InPA-SQP algorithm, both LMPC and NLMPC are evaluated on a real-time simulator with a sampling time of 0.01 *sec*. We note that both LMPC and NLMPC are computationally feasible without any overrun.

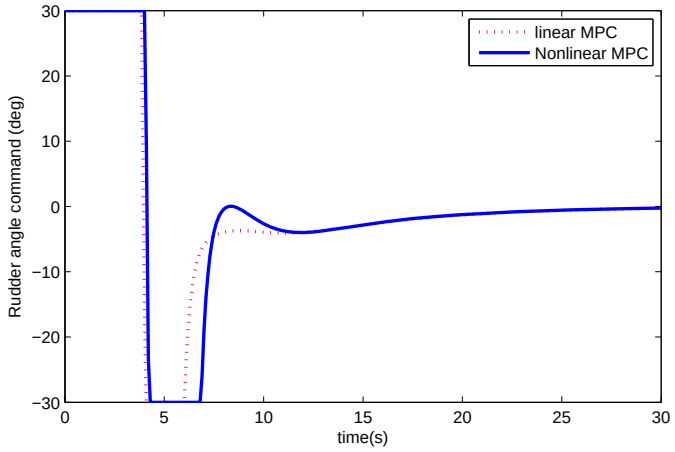

Figure 3.12: Control command generated by MPC

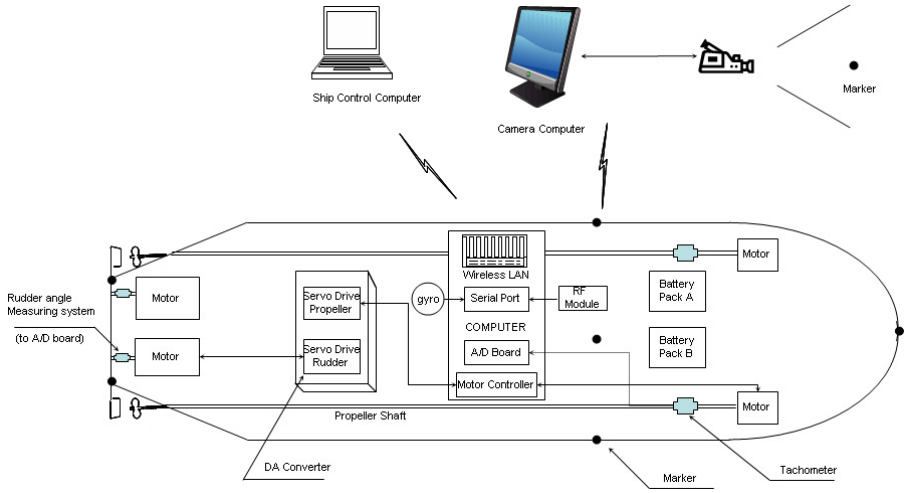

Figure 3.13: Wireless link between devices.

3.2.3 Experimental Platform and Experimental Results

For real-time implementation, feedback control is accomplished using a PC-based PC104 hardware which runs the QNX real-time operating system. PC104 communicates, through a wireless LAN, to a host PC, on which the control algorithm is programmed and tuned, data acquisition function is performed. Ship position signals are collected from the camera system and transmitted to PC104 via RF modems. The key control and communication devices are shown in Figure 3.13, together with the connections among the devices.

To implement MPC with a nonlinear model, the InPA-SQP algorithm is employed for efficient solution of $\mathscr{P}_N(x(k))$ at each time instant k. For NLMPC, the parameters

R and α in (3.29) are tuned to $R = .5$ and $\alpha = 1.5$ and for LMPC, the parameters are tuned to $R = .8$ and $\alpha = 1.5$. The length of prediction horizon $N = 7$. Figures 3.14 and 3.15 show the experimental results of implementing linear and non-linear MPC on the ship with sampling rate 0.01 *sec*. In the figures, Y_{pos} represents the position of the ship. As expected, the results observed in simulations are achieved with the same MPC parameters, except that now oscillations in control command are observed. The observed oscillation can be attributed to delays incurred in the wireless communication, in the computation of the position of the ship from signals provided by cameras, and unmodelled delay of the system. In the next section, the effects of the delay in the feedback loop is considered and a compensating strategy is introduce to mitigate its effects.

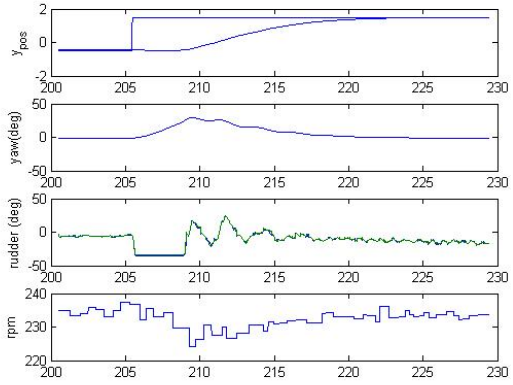

Figure 3.14: Response of LMPC with $N = 7$

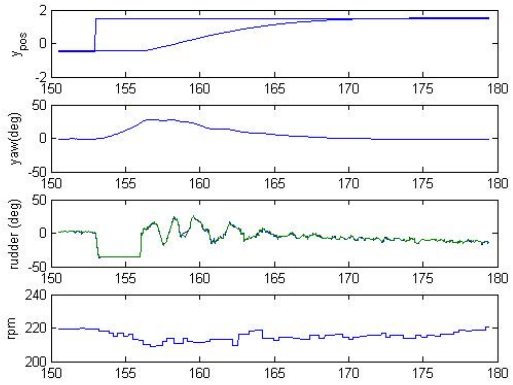

Figure 3.15: Response of NLMPC with $N = 7$

3.2.4 Delay Compensation

Given that the position of the ship is captured via cameras and the images are sent to the monitoring computer to calculate the states before the results are sent back to the ship controller, the whole process involves communication and computation delays as well as delay is system dynamics. It is shown in experiment and simulation that such delays can deteriorate the performance of the system in the sense that they cause oscillations in the rudder control command. Since the time-delay is often the main cause of performance degradation and instability, it has been treated extensively in the control literature (see [100], [101] and references therein).

For MPC, however, only a few algorithms have been published that handle time-delay systems explicitly, such as [102, 103, 104]. In [102], the time delay is handled by augmenting the state space representation such that the augmented model has structural uncertainty without delay. Then the robust MPC is applied on the augmented model which requires solving Linear Matrix Inequalities (LMI). In [103], a simple receding horizon control is suggested for continuous time systems with state delays, where a reduction technique is used so that an optimal control problem for system with delay in state is transformed to an optimal control problem for delay-free ordinary systems. However, the method is confined to unconstrained systems. In [104], an MPC strategy is introduced where an upper bound to a worst value of the cost (for different values of delay) is minimized subject to constraints of the system and an LMI which is imposed to guarantee stability.

For the ship control problem under consideration, the delay in the observation of the state is faced in implementation of the MPC strategy. Therefore, if the delay in the loop is d and the current state of the system is $x(k)$, the observed state is $x(k-d)$, with $d \times$ (sampling time) represents the overall delay in the closed loop system. We can compensate the delay simply by estimating the current state of the system $x(k)$, using the observed state $x(k-d)$, the control command applied to the system from the time $k-d$ to the the current time k, and the model of the system. While the methods proposed in literature can handle observation delays, the prediction method, described in the following, compensates the delay without solving LMI, leading to a simpler solution.

Consider the system

$$\begin{aligned} &x(k+1) = f(x(k), u(k)), \\ &y(k) = x(k-d), \\ &x(k) \in \mathbb{X},\ u(k) \in \mathbb{U} \end{aligned} \tag{3.30}$$

where $\mathbb{X}$ and $\mathbb{U}$ are state and input constraint sets respectively, $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$, and d is the delay in the observation that is due to sensor dynamics or communication networks. The system without delay is assumed to be fully observable or the states can be measured real-time. Defining $\hat{x}(k) := x(k-d)$, (3.30) can be transformed into a model with a time delay d in the input, i.e.,

$$\hat{x}(k+1) = f(\hat{x}(k), u(k-d)). \tag{3.31}$$

Defining the new states as

$$\eta(k) := [u(k-d)^T, \cdots, u(k-1)^T]^T, \tag{3.32}$$

the system (3.31) is written in the following form

$$\begin{bmatrix} \hat{x}(k+1) \\ \eta(k+1) \end{bmatrix} = \begin{bmatrix} f(\hat{x}(k), C\eta(k)) \\ D\eta(k) \end{bmatrix} + \begin{bmatrix} 0_{(n+(d-1)m)\times m} \\ I_{m\times m} \end{bmatrix} u(k), \tag{3.33}$$

where

$$\begin{aligned} C &= [I_{m\times m} \quad 0_{m\times dm}], \\ D &= \begin{bmatrix} 0 & I_{m\times m} & & \cdots & 0 \\ 0 & 0 & I_{m\times m} & \cdots & 0 \\ & & \vdots & & \\ 0 & 0 & \cdots & & I_{m\times m} \\ 0 & 0 & \cdots & & 0 \end{bmatrix}. \end{aligned} \tag{3.34}$$

Suppose that at each time instant k $\eta(k)$, the history of applied control sequence up until k, and $y(k) = x(k-d) = \hat{x}(k)$ are available. Therefore, the states of the system (3.33) are observed at the time instant k and the state $x(k)$ can be estimated as follows:

$$\begin{aligned} x(k-d+1) &= f(x(k-d), u(k-d)), \\ &\vdots \\ x(k) &= f(x(k-1), u(k-1)). \end{aligned} \tag{3.35}$$

We can use the estimated current state of the system, i.e., $x(k)$, as the initial state for the MPC optimization problem.

We propose the following MPC strategy to tackle the delay in the system (3.30).

- At each time instant k, predict the state $\hat{x}(k+d) = x(k)$ using the states $\eta(k)$,

$\hat{x}(k) = x(k-d)$ and (3.35).

- Solve the optimization problem $\mathscr{P}_N(x(k))$ (defined by equation 1.3), and implement the first element of the optimal control sequence.

It can be easily seen that the delay in the observation is compensated by estimation of the current state of the system ($x(k)$) if the model of the system is perfectly known.

To validate the algorithm, a delay of $0.3s$ is assumed in the communication channel and the system (3.26) is simulated with and without delay compensation. From Figure 3.16, it can be seen that oscillations in the rudder command are effectively compensated by estimating the current state of the system ($x(k)$), considering the delay of the system. It can be observed in Figure 3.17 that the cross tracking error with almost the same settling time is achieved while the ruder oscillation is mitigated.

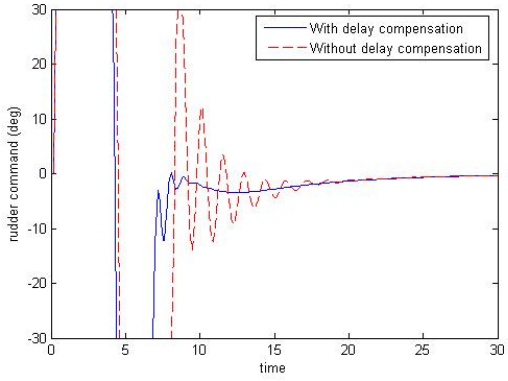

Figure 3.16: Rudder command for systems with and without delay compensator

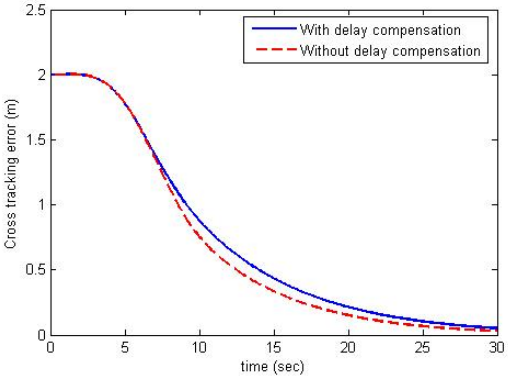

Figure 3.17: Cross tracking error for systems with and without delay compensator

3.2.5 Conclusion

In this section, InPA-SQP is employed for path following of a model ship subject to input constraints. The effectiveness of the proposed MPC has been verified with experimental results. Moreover, the effects of the communication delay and the delay in system dynamics are considered and mitigated through estimating the current (actual) state of the system using stored control input history and observed state of the system.

So far, the computational feasibility of MPC has been the main focus and the InPA-SQP method is developed to address this issue and to facilitate implementation of MPC. However, another major concern regarding MPC is robustness of the resulting closed loop system with respect to model uncertainties or disturbances. As illustrated in Section 1.1.4, for general nonlinear systems, guaranteed robust strategies is an open problem. However, many applications fall in the category of constrained linear systems subject to additive bounded disturbances. In the next chapter, it is shown that it is possible to guarantee robustness and stability and comply with constraints for such class of systems, while the requirement of repeatedly solving an optimization problem is relaxed.

Chapter 4

Robust control of constrained linear systems

In this chapter, a control problem is considered for constrained discrete time linear systems that are subject to bounded additive disturbances. Our goal is to provide a control method that enforces specified state and input constraints in the presence of disturbances and steers state trajectories to a given target set.

4.1 Problem Statement

Consider a class of linear, time-invariant, discrete-time systems described by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w(k), \\ x(k) &\in \mathbb{R}^n,\ u(k) \in \mathbb{R}^m,\ w(k) \in \mathbb{R}^n \end{aligned} \tag{4.1}$$

where $x(k)$, $u(k)$ and $w(k)$ are, respectively, the state, control and disturbance vectors; $x(k+1)$ denotes the successor state of $x(k)$ and $k \in \mathbb{N}$, where $\mathbb{N}$ is the set of non-negative integers.

We assume that the disturbance w belongs to a polytope W, the control and state are subject to hard constraints, i.e.,

$$(u(k), x(k)) \in \Omega \subset \mathbb{U} \times \mathbb{X} \text{ and } w(k) \in W, \tag{4.2}$$

where $\mathbb{U}$ and W are (convex, compact) polytopes, containing the origin in their interior, and $\mathbb{X}$ is a (convex) closed polyhedron. Finally, a target constraint set $\mathbb{X}_t$ is given by

$$\mathbb{X}_t = \{x \in \mathbb{R}^n | Yx \leq q\}, Y \in \mathbb{R}^{r \times n},\ q \in \mathbb{R}^r. \tag{4.3}$$

We assume that $\mathbb{X}_t$ is bounded (so it is a polytope) and $0 \in int(\mathbb{X}_t)$. The control objective is to find u that steers the state into the target set $\mathbb{X}_t$. Moreover, the existence of a feedback gain matrix $K \in \mathbb{R}^{m\times n}$ is assumed such that $A_K = A+BK$ is an exponentially stable matrix and the minimal robust invariant set[1] F_K for the system $x(k+1) = A_K x(k) + w(k)$, defined in [78], satisfies

$$F_K \subseteq \mathbb{X}_t. \tag{4.4}$$

In the sequel, the Pontryagin difference [78] is used which, for two sets $S \subset \mathbb{R}^n$ and $T \subset \mathbb{R}^n$, is defined as $S \sim T = \{x | x + t \in S,\ \forall\ t \in T\}$.

4.2 Robust Control Algorithm

For any initial state $x \in \mathbb{X}$, the following control sequence

$$\mathbf{u}^*(x) := \{u_0^*(x), u_1^*(x), \cdots, u_{N-1}^*(x)\}$$

and associated state sequence

$$\mathbf{x}^*(x) := \{x_0^*(x), x_1^*(x), \cdots, x_N^*(x)\}$$

are feasible if they satisfy the following set of constraints $C(x)$:

$$\begin{aligned} &x_0^*(x) = x, \\ &x_{i+1}^*(x) = Ax_i^*(x) + Bu_i^*(x),\ i = 0, \cdots, N-1, \end{aligned} \tag{4.5}$$

$$\begin{aligned} &\Omega_0 = \Omega, \\ &\Omega_{i+1} = \Omega_i \sim [K^T \quad I]^T A_K^i W,\ i = 0, \cdots, N-1, \\ &(u_i^*(x), x_i^*(x)) \in \Omega_i\ i = 0, \cdots, N-1, \end{aligned} \tag{4.6}$$

$$x_N^*(x) \in \mathbb{X}_f, \tag{4.7}$$

---

[1]The robust invariant set F_K for the system $x(k+1) = A_K x(k) + w(k)$ is minimal if for all closed robust invariant sets X such that $A_K X + W \subset X$, it follows that $F_K \subset X$ [78].

where $\mathbb{X}_f$ is a robust invariant set for the system

$$x(k+1) = A_K x(k) + w(k),\ k \in \mathbb{N} \tag{4.8}$$

with $w \in A_K^N W$, i.e.,

$$A_K \mathbb{X}_f + A_K^N W \subset \mathbb{X}_f \tag{4.9}$$

which satisfies the constraint

$$[K^T \quad I]^T \mathbb{X}_f \subset \Omega_N. \tag{4.10}$$

Let us assume that for an initial state $x(0)$, $\mathbf{u}^*(x(0))$ and $\mathbf{x}^*(x(0))$ are feasible control and state sequences. Computing these initial sequences involves finding a point inside the polyhedron defined by (4.5)-(4.7). Now the following iterative algorithm is proposed, where at each time instant k, the feasible control sequence $\mathbf{u}^*(x(k))$ is constructed using the feasible control and state sequences $\mathbf{u}^*(x(k-1))$ and $\mathbf{x}^*(x(k-1))$, where $x(k)$ is the observed state at the current time instant and $x(k-1)$ denotes the predecessor state:

$$\begin{aligned} &u_i^*(x(k)) = u_{i+1}^*(x(k-1)) + K(x_i^*(x(k)) - x_{i+1}^*(x(k-1)), \\ &\text{for } i = 0, \cdots, N-2, \\ &u_{N-1}^*(x(k)) = K x_{N-1}^*(x(k)); \end{aligned} \tag{4.11}$$

$$\begin{aligned} &x_0^*(x(k)) = x(k), \\ &x_{i+1}^*(x(k)) = A x_i^*(x(k)) + B u_i^*(x(k)),\ i = 0, \cdots, N-1. \end{aligned} \tag{4.12}$$

At each time instant, the first element of the feasible control sequence is applied as the control signal, therefore the robust control law is

$$u(k) = \kappa_N^*(x(k)) := u_0^*(x(k)). \tag{4.13}$$

Theorem 4.2.1. *Suppose the set of constraints $C(x(0))$ is satisfied with the feasible control, $\boldsymbol{u}^*(x(0))$, and state, $\boldsymbol{x}^*(x(0))$, sequences. Then the state and input trajectories of the system (4.1) with the control law defined by (4.13) satisfy the input and state constraints (4.2). Furthermore, the set of constraints $C(x(k))$ is satisfied by the control and state sequences $\boldsymbol{u}^*(x(k))$ and $\boldsymbol{x}^*(x(k))$, defined by (4.11) and (4.12), for all $k > 0$.*

Proof. Here, denote the state at the time instant k as the current state x and state

at the time instant $k+1$ as the successor state $x(k+1)$. Assume $\mathbf{u}^*(x)$ and $\mathbf{x}^*(x)$ are feasible control and state sequences for $C(x)$. Considering the state evolution (4.12) and control update (4.11),

$$\begin{aligned} x^*_{i+1}(x(k+1)) &= Ax^*_i(x(k+1)) + Bu^*_i(x(k+1)) \\ &= Ax^*_i(x(k+1)) + Bu^*_{i+1}(x(k)) + BK(x^*_i(x(k+1)) - x^*_{i+1}(x(k))) \\ &= A_K(x^*_i(x(k+1)) - x^*_{i+1}(x(k))) + x^*_{i+2}(x(k)),\ i = 0, \cdots, N-2 \end{aligned} \tag{4.14}$$

where the last equality is achieved by adding and subtracting $A_K x^*_{i+1}(x(k))$ and using equation (4.12). From (4.12),

$$x^*_0(x(k+1)) - x^*_1(x(k)) = x(k+1) - Ax(k) - Bu^*_0(x(k)) = w_0 \in W, \tag{4.15}$$

and using (4.14) it can be easily shown that

$$x^*_i(x(k+1)) - x^*_{i+1}(x(k)) = A^i_K w_0 \in A^i_K W,\ i = 0, \cdots, N-1. \tag{4.16}$$

Moreover, from (4.11) and (4.16)

$$u^*_i(x(k+1)) - u^*_{i+1}(x(k)) = KA^i_K w_0 \in KA^i_K W,\ i = 0, \cdots, N-2. \tag{4.17}$$

From equations (4.16) and (4.17),

$$\begin{bmatrix} u^*_i(x(k+1)) \\ x^*_i(x(k+1)) \end{bmatrix} = \begin{bmatrix} u^*_{i+1}(x(k)) \\ x^*_{i+1}(x(k)) \end{bmatrix} + \begin{bmatrix} K \\ I \end{bmatrix} A^i_K w_0, \tag{4.18}$$

for $i = 0, \cdots, N-2$. Considering (4.6) and the feasibility of $u^*(x(k))$, $x^*(x(k))$, (4.18) can be written as follows:

$$\begin{aligned} &\begin{bmatrix} u^*_i(x(k+1)) \\ x^*_i(x(k+1)) \end{bmatrix} \in \Omega_{i+1} + \begin{bmatrix} K \\ I \end{bmatrix} A^i_K W \\ &= (\Omega_i \sim \begin{bmatrix} K \\ I \end{bmatrix} A^i_K W) + \begin{bmatrix} K \\ I \end{bmatrix} A^i_K W \subseteq \Omega_i. \end{aligned} \tag{4.19}$$

From (4.11) and (4.16), where $i = N-1$,

$$\begin{aligned}\begin{bmatrix} u^*_{N-1}(x(k+1)) \\ x^*_{N-1}(x(k+1)) \end{bmatrix} &= \begin{bmatrix} K \\ I \end{bmatrix} x^*_{N-1}(x(k+1)) \\ &= \begin{bmatrix} K \\ I \end{bmatrix} (x^*_N(x(k)) + A_K^{N-1} w_0).\end{aligned} \tag{4.20}$$

Since (according to the terminal predicted state constraint (4.10)) $x^*_N(x(k)) \in \mathbb{X}_f$, (4.20) implies that

$$\begin{bmatrix} u^*_{N-1}(x(k+1)) \\ x^*_{N-1}(x(k+1)) \end{bmatrix} \in \Omega_N + \begin{bmatrix} K \\ I \end{bmatrix} A_K^{N-1} W \subseteq \Omega_{N-1}, \tag{4.21}$$

where the last inclusion follows from (4.6).

On the other hand, from equations (4.11) and (4.12),

$$\begin{aligned} x^*_N(x(k+1)) &= A x^*_{N-1}(x(k+1)) + B u^*_{N-1}(x(k+1)) \\ &= (A+BK) x^*_{N-1}(x(k+1)) = A_K x^*_{N-1}(x(k+1)). \end{aligned} \tag{4.22}$$

From (4.16), where $i = N-1$,

$$x^*_{N-1}(x(k+1)) - x^*_N(x(k)) \in A_K^{N-1} W. \tag{4.23}$$

Multiplying (4.23) by A_K and using (4.22),

$$x^*_N(x(k+1)) \in \{A_K x^*_N(x(k))\} + A_K^N W. \tag{4.24}$$

Since $x^*_N(x(k)) \in \mathbb{X}_f$ and the set $\mathbb{X}_f$ is a robust invariant set for the system (4.8) and disturbance set $A_K^N W$,

$$\{A_K x^*_N(x(k))\} + A_K^N W \subset A_K \mathbb{X}_f + A_K^N W \subset \mathbb{X}_f.$$

Thus, $x^*_N(x(k+1)) \in \mathbb{X}_f$. This and (4.21) imply that $\mathbf{x(k)}^*(x(k+1)),\ \mathbf{u}^*(x(k+1))$ satisfy constraints (4.6)-(4.7). □

To investigate convergence properties of the controller (4.13), first recall that the

minimal robust invariant set F_K can be expressed as [78]:

$$F_K = \sum_{i=0}^{\infty} A_K^i W. \tag{4.25}$$

We need the following auxiliary results to establish the domain of attraction for the proposed control algorithm.

Lemma 4.2.1. *Let $\boldsymbol{u}^*(x(k))$ and $\boldsymbol{x}^*(x(k))$ be feasible control and state sequences corresponding to state $x(k)$, and let $\boldsymbol{u}^*(x(k+1))$ and $\boldsymbol{x}^*(x(k+1))$ be control and state sequences generated by (4.11) and (4.12), where $x(k+1)$ is the successor state defined in (4.1). Moreover, assume F_K is the minimal disturbance invariant set as defined earlier. Then*

$$d(x_i^*(x(k+1)), A_K^i F_K) \leq d(x_{i+1}^*(x(k)), A_K^{i+1} F_K),\ i = 0, \cdots, N-1. \tag{4.26}$$

Proof. Equations (4.15) and (4.16) imply that for $i = 0, \cdots, N-1$,

$$\exists w \in A_K^i W \ s.t.\ x_i^*(x(k+1)) = x_{i+1}^*(x(k)) + w. \tag{4.27}$$

Moreover, if $w \in A_K^i W$, from (4.25)

$$A_K^{i+1} F_K + \{w\} \subset A_K^{i+1} F_K + A_K^i W = A_K^i F_K. \tag{4.28}$$

Therefore, from (4.27) and (4.28)

$$\begin{aligned} d(x_i^*(x(k+1)), A_K^i F_K)) &\leq d(x_{i+1}^*(x(k)) + w, A_K^{i+1} F_K + \{w\}) \\ &= d(x_{i+1}^*(x(k)), A_K^{i+1} F_K) \\ \text{for } i = 0 \cdots, N-1. & \end{aligned} \tag{4.29}$$

□

Lemma 4.2.2. *Let $x(k+1) = A_K x(k) + w(k),\ w \in A_K^N W$, P be the Lyapunov matrix corresponding to the stable matrix A_K, i.e., $P \succ 0$ and $\exists Q \succ 0$ s.t. $A_K^T P A_K - P = -Q$, and the norm $\|\cdot\|_p$ be defined as $\|x\|_p := \sqrt{x^T P x},\ x \in \mathbb{R}^n$. If the distance is defined in the normed space $(\mathbb{R}^n, \|\cdot\|_p)$ and $\|D\|_p$ is the induced norm of any square matrix $D \in \mathbb{R}^{n \times n}$, then*

$$\begin{aligned} &\exists\ 0 < \alpha < 1 \ s.t.\ \|A_K\|_p \leq \alpha \\ \textit{and}\quad &d(x(k+1), A_K^N F_K) \leq \alpha d(x(k), A_K^N F_K) \end{aligned} \tag{4.30}$$

Proof. Given $A_K^T P A_K - P = -Q$,

$$\forall x \in S_u = \{x \in \mathbb{R}^n |\ x^T P x = 1\}$$
$$x^T A_K^T P A_K x = 1 - x^T Q x.$$

Since S_u is compact

$$\exists\ \bar{x} \in S_u\ s.t.$$
$$\|A_K\|_p = \sup_{x \in S_u} \sqrt{x^T A_K^T P A_K x} = \sqrt{1 - \bar{x}^T Q \bar{x}} < 1.$$

The last inequality is due to the fact that $Q \succ 0$.

Moreover,

$$d(x(k+1), A_K^N F_K) = d(A_K x(k) + w(k), A_K^{N+1} F_K + A_K^N W).$$

Since $w \in A_K^N W$,

$$\begin{aligned} & d(A_K x(k) + w, A_K^{N+1} F_K + A_K^N W) \\ & < d(A_K x(k) + w, A_K^{N+1} F_K + \{w\}) \\ & = d(A_K x(k), A_K^{N+1} F_K). \end{aligned} \tag{4.31}$$

According to the definition of distance in the normed space

$$\begin{aligned} d(A_K x(k), A_K^{N+1} F_K) &= \inf_{w \in A_K^N F_K} \|A_K x - A_K w\|_p \\ &\leq \|A_K\|_p \inf_{w \in A_K^N F_K} \|x - w\|_p \\ &= \|A_K\|_p d(x, A_K^N F_K). \end{aligned} \tag{4.32}$$

From (4.31) and (4.32),

$$d(x(k+1), A_K^N F_K) \leq \|A_K\|_p d(x(k), A_K^N F_K), \tag{4.33}$$

and the proof is complete. □

Theorem 4.2.2. *If for an initial state $x(0)$, there exist control and state sequences satisfying the set of constraints $C(x(0))$, then the set F_K is robustly attractive (all trajectories converge to F_K despite disturbances) for the controlled uncertain system*

$$x(k+1) = Ax(k) + B\kappa_N^*(x(k)) + w(k), \tag{4.34}$$

where $w(k) \in W$. Furthermore, the region of attraction is

$$R = \{x \in \mathbb{R}^n |\ C(x)\ \textit{is feasible}\}.$$

Proof. Let us define the cost $J(x(k), \mathbf{u}^*(x(k)))$ as follows

$$J(x(k), \mathbf{u}^*(x(k))) := \sum_{i=0}^{N} d(x_i^*(x(k)), A_K^i F_K), \tag{4.35}$$

where $x_i^*(x(k)),\ i = 0, \cdots, N$ is defined in (4.12). If $x(k+1)$ is the successor state defined in (4.1), according to Lemma 4.2.1

$$\sum_{i=0}^{N-1} d(x_i^*(x(k+1)), A_K^i F_K) \leq \sum_{i=1}^{N} d(x_i^*(x(k)), A_K^i F_K). \tag{4.36}$$

From definition (4.35) and inequality (4.36) and the fact that $x(k) = x_0^*(x(k))$,

$$J(x(k+1), \mathbf{u}^*(x(k+1))) - J(x(k), \mathbf{u}^*(x(k))) \leq d(x_N^*(x(k+1)), A_K^N F_K) - d(x(k), F_K). \tag{4.37}$$

Making summation over the inequality (4.37) from time instant 0 to $M < \infty$, the following is obtained

$$\begin{aligned} &\sum_{k=0}^{M} d(x(k), F_K) \leq \sum_{k=1}^{M+1} d(x_N^*(x(k)), A_K^N F_K) \\ &+ J(x(0), \mathbf{u}^*(x(0))) - J(x(M+1), \mathbf{u}^*(x(M+1))) \\ &\leq J(x(0), \mathbf{u}^*(x(0))) + \sum_{k=1}^{M+1} d(x_N^*(x(k)), A_K^N F_K). \end{aligned} \tag{4.38}$$

On the other hand, from inequality (4.30),

$$d(x_N^*(x(k)), A_K^N F_K) \leq \alpha^k d(x_N^*(x(0)), A_K^N F_K). \tag{4.39}$$

Therefore, for all integer M,

$$\sum_{k=1}^{M+1} d(x_N^*(x(k)), A_K^N F_K) \leq \frac{\alpha}{1-\alpha} d(x_N^*(x(0)), A_K^N F_K). \tag{4.40}$$

From (4.38) and (4.40),

$$\sum_{k=0}^{M} d(x(k), F_K) \leq J(x(0), \mathbf{u}^*(x(0))) + \frac{\alpha}{1-\alpha} d(x_N^*(x(0)), A_K^N F_K). \tag{4.41}$$

Since $\|x(0)\|_p < \infty$ and the set $\mathbb{U}$ is compact, the right hand side of the above inequality is bounded. Therefore, the sequence $\{V_M := \sum_{k=0}^{M} d(x(k), F_K)\}$ is bounded and non-decreasing in $\mathbb{R}^n$. Hence, $\{V_M\}$ is convergent and, as the result,

$$d(x(M), F_K) = V_{M+1} - V_M \to 0, \ as \ M \to \infty. \tag{4.42}$$

Therefore, F_K attracts all trajectories with feasible initial state. □

Remark 4.2.1. *The important feature of the proposed method is that the attraction to F_K is achieved without involving any repeated optimization or minimal robust invariant set approximation, while in MPC based methods [52, 53] attraction to F_K is achieved by solving online an optimization problem.*

Remark 4.2.2. *The proposed robust control method may be viewed as based on tightening constraints, at each time instance over the prediction horizon, by $A_K^i W$. In this regard, the proposed scheme is similar to [49, 54]. However, the advantage of the proposed method is that it does not require the terminal constraint set $\mathbb{X}_f$ to be a subset of the desired target set $\mathbb{X}_t$. In fact, the target set $\mathbb{X}_t$ is only required to contain the minimal robust invariant set F_K, i.e., $F_K \subset \mathbb{X}_t$, in order to be attractive.*

In the next section the proposed method is employed for the roll control of a ship equipped with the stabilizer fins.

4.3 Application: Control of Ship Fin Stabilizer

Control of the roll motion of ships has been extensively considered in the literature (see [71]-[72] and references therein). As elaborated in [72], large roll motions induced by ocean waves can severely affect the safety and performance of surface ships. For ships that normally operate above certain speeds, using fins is one of the most effective roll control techniques [74]. Ship fin stabilizers consist of a pair of fins located approximately amidship on the bilge of the hull, as indicated in Figure 4.1. These fins have the freedom to rotate in a certain range, and the control system changes

the mechanical angle of the fins, α_m, according to a control algorithm that uses measurements of the roll angle, ϕ, and roll rate, p. Defining the angle of attack, α_e, as the angle of the flow with respect to the fin, hydrodynamic forces, proportional to the angle of attack, are induced on the fins. Due to the location of the fins on the hull, these forces produce a moment that counteract and reduce the wave-induced roll motion.

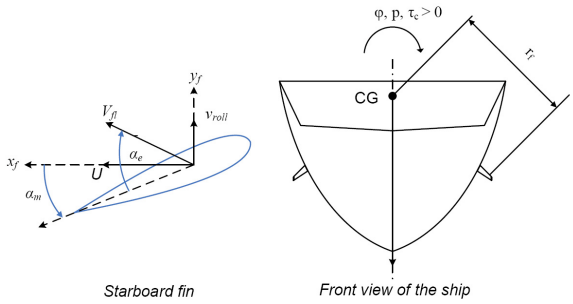

Figure 4.1: Ship roll fin stabilizer.

Depending on the size of the ship and the severity of the sea state, the effectiveness of the fin stabilizer can be degraded due to nonlinear effects associated with unsteady hydrodynamics of the fin. This phenomenon is called the dynamic stall. For a small angle of attack, the roll moment starts to increase linearly as a function of the angle of attack. When the angle of attack exceeds a certain threshold, the roll moment generated by the fin decreases nonlinearly, as the angle of attack increases. This gain reversal in the nonlinear hydrodynamic moment results in the loss of control in the fin stabilizer.

The dynamic stall depends on the operation of the fins and their location on the hull. It usually occurs when a group of high waves appears over a short time interval and makes the angle of attack exceed a certain value, α_{stall} [97]. Under the dynamic stall condition, the control system becomes ineffective and as the result, the roll angle, in the presence of high waves, increases rapidly and significantly. A common approach to deal with these effects in practice is to reduce the gain of the controller. Since the conditions for dynamic stall may not be always present, this conservative approach reduces the overall performance when dynamic stalls are not present. MPC is employed in [71] as an alternative approach to enforce input constraints associated with the mechanical angle of the fin as well as the output constraint associated with the effective angle of attack of the fin. The fin stabilizer control problem is considered as a robust control problem in this section, where the linear dynamics of the system are affected by a bounded additive disturbance, and the robust control algorithm

proposed in Section 4.2 is employed, which does not require repeated on-line solution of an optimization problem.

4.3.1 Equations of motion

For roll stabilization, the ship model from [71, 72] is employed for control design. The following linear equations describe the roll motion in a frame fixed at the center of gravity of the ship:

$$\begin{aligned} &\dot{\phi} = p, \\ &I_{\phi\phi}\dot{p} + Dp + G\phi = \tau_c + \tau_w, \end{aligned} \tag{4.43}$$

where ϕ is the roll angle, p is the roll rate, τ_c is the control moment produced by the fins, and τ_w is the wave excitation moment. Moreover, $I_{\phi\phi}$ is the total inertia in roll about the axis along the ship longitudinal direction, D is the equivalent linear damping (which accounts for potential and viscous effects), and G is the linear roll restoring coefficient [71, 73].

For a ship fin stabilizer, the effective angle of attack can be calculated as follows

$$\alpha_e = -\alpha_{pu} - \alpha_m, \tag{4.44}$$

where α_m is the mechanical angle of the fin (control input) and α_{pu} is the flow angle induced by the combination of forward speed, U, and roll rate, p. It is calculated as follows

$$\alpha_{pu} = arctan(r_f p/U) \approx \frac{r_f}{U} p. \tag{4.45}$$

If the angle of attack is less than a certain value, i.e.,

$$\alpha_e < \alpha_{stall},$$

the roll moment generated by one fin is approximately proportional to the angle of attack as follows:

$$\tau_c \approx K_\alpha \alpha_e. \tag{4.46}$$

The above linear relation does not hold if the angle of attack (α_e) exceeds α_{stall} [97].

4.3.2 Constraints

We consider two set of constraints:

- Input constraint which reflects saturation of the mechanical angle of the fin:

$$|\alpha_m| \leq \alpha_{sat}, \tag{4.47}$$

- Input-state constraint that is aimed at preventing dynamic stall:

$$|\alpha_e| = |\frac{r_f}{U}p + \alpha_m| \leq \alpha_{stall}. \tag{4.48}$$

4.4 Controller design and Simulation results

To proceed with the controller design and performance evaluation of the proposed system, the vessel model introduced in [72] is used where parameters for (4.43)-(4.46) are given. The vessel travels at 15 kts forward speed, i.e., $U = 15$ kst, with a magnitude constraint for the mechanical angle of the fin of 0.436 rad, and a magnitude constraint for the angle of attack of 0.41 rad. Moreover, the coefficients in (4.43), (4.45) are:

$$\begin{aligned} I_{\phi\phi} &= 3.4263 \times 10^6 \ Kgm^2/rad, \\ D &= 0.5 \times 10^6 \ Kgm^2/(rad/sec), \\ G &= 3.57 \times 10^9 \ Nm/rad, \ r_f = 4.22 \ m. \end{aligned} \tag{4.49}$$

A discrete-time model of (4.43), with sampling period T_s=0.1 sec, is

$$x(k+1) = A_d x(k) + B_d u(k) + B_w \tau_w(k) \tag{4.50}$$

where $x = [\phi \ p]^T$, $u = \alpha_m$ and

$$A_d = \begin{bmatrix} 0.99 & 0.095 \\ -0.08 & 0.90 \end{bmatrix}, \ B_d = \begin{bmatrix} -0.007 \\ -0.142 \end{bmatrix}, \ B_w = \begin{bmatrix} 0.004 \\ 0.095 \end{bmatrix}. \tag{4.51}$$

Assuming $|\tau_w| \leq 0.2 I_{\phi\phi}$, according to the general formulation (4.1), the disturbance set W is:

$$W = \{B_w w, |w| \leq .2 I_{\phi\phi}\}.$$

The feedback gain $K = [-6.31 \ -3.66]$ is designed using LQR technique with weight $R = 10$ for control input and the weight $Q = diag[10\ 2]$ for the states. With the designed feedback gain K, the corresponding minimal invariant set is a subset of the target set

$$\mathbb{X}_t = \{(\phi\ p) | \phi \in [-0.02\ 0.02],\ p \in [-0.06\ 0.06]\}$$

for the disturbance set W. Considering the constraints (4.47) and (4.48), the sets $\Omega_i, i = 1, \cdots, N$ are defined as follows:

$$\Omega_i = \left\{ (u,x) \left| \begin{array}{l} |c_u u + c_x x| \leq \alpha_{stall} - \sum_{j=0}^{i-1} h_{A_k^j W}(c_x + c_u K) \\ |u| \leq \alpha_{sat} - \sum_{j=0}^{i-1} h_{A_k^j W}(K) \end{array} \right. \right\} \tag{4.52}$$

where

$$c_u = 1,\ c_x = [0\ \frac{r_f}{U}] \tag{4.53}$$

and for a set $S \subset \mathbb{R}^n$, $h_S(\cdot)$ denotes its support function, see e.g. [78]. The value of $N = 10$ was chosen to provide large domain of attraction.

Moreover, for this example, the set $\mathbb{X}_f$ in the robust control algorithm is chosen as the maximal robust invariant set. The set $\mathbb{X}_f$ is contained in the following set as shown in Figure 4.2,

$$\mathbb{X}_N := \left\{ x \left| \begin{array}{l} |(c_u K + c_x)x| \leq \alpha_{stall} - \sum_{j=0}^{N-1} h_{A_k^j W}(c_x + c_u.K) \\ |Kx| \leq \alpha_{sat} - \sum_{j=0}^{N-1} h_{A_k^j W}(K) \end{array} \right. \right\}. \tag{4.54}$$

Given the sets $\mathbb{X}_f$ and Ω_i, the control law for the fin stabilizer is determined according to (4.11) and (4.12).

The simulation of the closed loop was performed for a given sinusoidal wave torque profile with period of $7sec$ and magnitude of $0.2I_{\phi\phi}$.

Figure 4.2 shows the trajectory of the system with initial condition $[\phi\ p] = [0\ rad\ \ 0.45\ rad/sec]$. It can be seen that in the presence of sinusoidal wave disturbance, the ship roll motion is stabilized around the origin within a minimal invariant set characterized by the matrix A_K and the set W, while saturation constraints as well as the constraint on the angle of attack α_e are satisfied. Figure 4.3 shows the

angle of the fin and the angle of attack.

For comparison, the robust MPC method is also applied to the roll control problem. The set of tightened constraints (4.52) and (4.54) was used along with the cost function

$$\sum_{i=0}^{9} Ru(i)^2 + x(i)^T Q x(i) + x(10)^T S_f x(10),$$

where S_f is the solution of the associated discrete-time Riccati equation for the infinite horizon problem. Figures 4.4 and 4.5 show the roll angle and angular velocity when the proposed robust method and robust MPC are employed. Referring to Figures 4.4 and 4.5, similar performance can be observed. For the initialization of the proposed algorithm, a LP problem is solved at $k = 0$ to provide a feasible trajectory, no optimization problem is solved for $k > 0$, which leads to considerably less computational time compared to the MPC at each time instant $k > 0$ (according to simulation results, three order of magnitude less than the computational time of MPC). As one can see from these figures, the constraints are satisfied. The region of attraction of robust controller is shown in Figure 4.6. The MPC is implemented using the same set of constraints as the robust controller; thereby providing the same region of attraction.

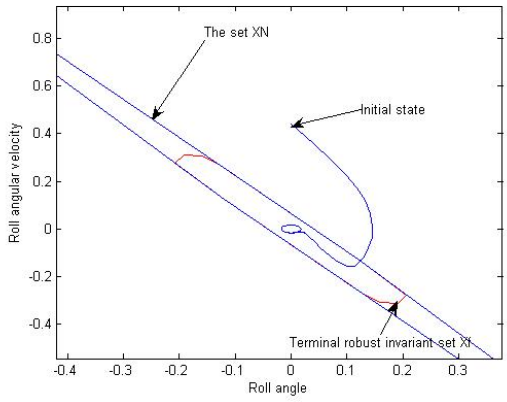

Figure 4.2: Trajectory of the system with initial condition [ϕ p]=[0 rad, 0.45 rad/sec].

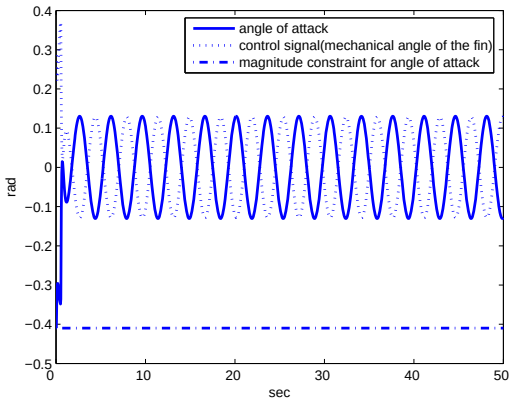

Figure 4.3: Angle of attack and fin angle of the system with initial condition [ϕ p]=[0 rad, 0.45 rad/sec]. Angle of attack is constrained to $\pm$ 0.41 *rad* and fin is constrained to $\pm$ 0.436 *rad*.

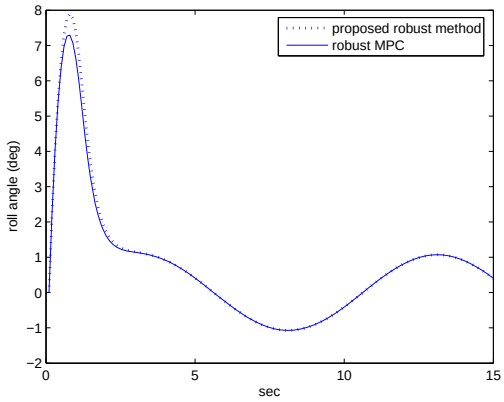

Figure 4.4: Roll angle of the system with initial condition [ϕ p]=[0 rad, 0.45 rad/sec].

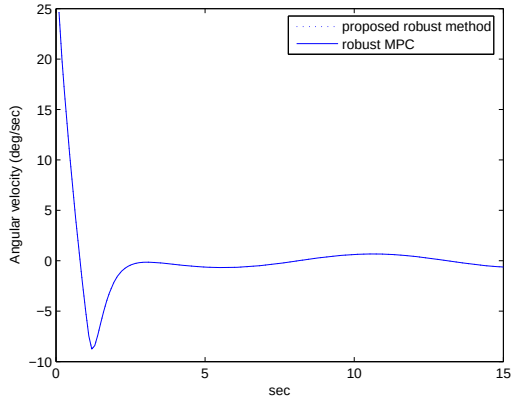

Figure 4.5: Angular velocity of the system with initial condition [ϕ p]=[0 rad, 0.45 rad/sec].

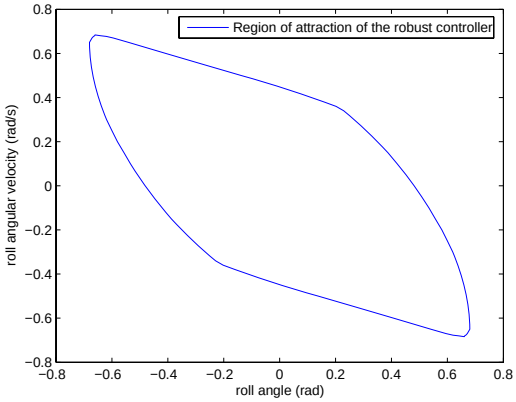

Figure 4.6: Region of attraction of the proposed robust controller.

4.5 Conclusion

This chapter presented a robust controller for a class of constrained linear systems subject to mixed state and input constraints with bounded disturbances. The novel feature of the robust controller is that the control action is a linear combination of known data at each sampling time and therefore it is highly computationally effective. The proposed controller guarantees convergence of state trajectory to a minimal invariant set of the desired system while explicit specification or approximation of such set is not required. The method was employed in the chapter for control of roll motion of a high speed ship, to enforce the dynamic stall and fin saturation constraints. Convergence to a desired target set in the presence of sea waves has been demonstrated. Simulation results were presented to show the effectiveness of the proposed method. Other constraints, e.g., rate limits, can be handled similarly. In this chapter, it was assumed that the disturbance is confined in a compact set W. While in practical applications the disturbances are rate bounded or follow the low-pass-filtering nature of physical systems, this assumption is conservative since it allows the disturbance to vary freely inside W or even jump from one point to another. In the next chapter the dynamic of disturbances are incorporated, leading to a smaller attractor set. Consequently, smaller attractor allows smaller target set X_t (defined in (4.3)).

Chapter 5

Robust Control of Constrained Systems with Filtered Bounded Disturbances

In this chapter, the minimal robust invariant attractor (MRIA) set for linear systems subject to additive disturbances confined in a state-dependent and bounded set is analyzed and characterized. In particular, existence of a MRIA set is proved and the set is characterized when the state dependent set is upper-semi continuous. Built on this result, the existence of a minimal attracting invariant set is established for the case when disturbances evolve within a compact set according to a linear dynamic model. The MRIA set is smaller if the disturbance model is considered compared to the case where only the boundedness of the additive disturbance is assumed. A numerical example is provided that shows the size of the minimal invariant attracting set is considerably different in the two cases. Not only does the analysis provide smaller target set X_t in Chapter 4, but it characterizes the MRIA set that can be employed in the design of robust MPC strategies, such as tube MPC [53], to achieve robust stability, improve control response and reduce conservativeness.

This chapter is organized as follows: in Section 5.1, the existence of MRIA set is established for linear systems subject to additive disturbance where the disturbance belongs to an upper-semi-continuous state dependent set. In Section 5.2, the system where the disturbance evolves inside a compact set according to a known disturbance model is introduced and shown to be a special case of the system described in Section 5.1. Given the dynamics of disturbance, existence of a MRIA set is established in the lifted space of state and disturbance in Section 5.3. Robust MPC design using the MRIA set in the lifted space is described in Section 5.4 followed by a numerical example where it is shown that the MRIA set is considerably less conservative if the the dynamics of disturbances are considered.

Basic definitions

The sets of non-negative integers and reals are denoted by $\mathbb{N}$ and R_+, respectively, i,e, $\mathbb{N} := \{0, 1, 2, \cdots\}$ and $\mathbb{R}_+ := \{x \in R \ : \ x \geq 0\}$. Given two sets $X \subset R^n$ and $Y \subset R^n$, the Minkowski set addition is defined by $X \oplus Y := \{x+y \ : \ x \in X, \ y \in Y\}$. We write $x \oplus X$ instead of $\{x\} \oplus X$. Given a set X and a real matrix M of compatible dimension, let $MX := \{Mx \ : \ x \in X\}$. Given a matrix $M \in R^{n \times n}$, $\rho(M)$ denotes the largest absolute value of its eigenvalues. A family of non-empty compact subsets of $\mathbb{R}^n$ is denoted by $Com(\mathbb{R}^n)$. For $X \subset \mathbb{R}^n$, $Cl(X)$ denotes the closure of X. A function $\delta : Com(\mathbb{R}^n) \times Com(\mathbb{R}^n) \to \mathbb{R}$ is a Hemimetric on $Com(\mathbb{R}^n)$ if $\delta(A, B) \leq \delta(A, C) + \delta(C, B)$ for all $A, B, C \in Com(\mathbb{R}^n)$ and $\delta(A, A) = 0$ for all $A \in Com(\mathbb{R}^n)$. If δ is a Hemimetric, then an open ball centered at $X \in Com(\mathbb{R}^n)$ and radius ϵ is denoted by $B_\epsilon(X) := \{Y \in Com(\mathbb{R}^n) \mid \delta(X, Y) < \epsilon\}$.

In Hemimetric space $(Com(\mathbb{R}^n), \delta)$, a set is open if it is a union of open balls, defined by Hemimetric δ. The Hemimetrics δ_l and δ_u are defined as $\delta_l(X, Y) := inf\{\epsilon > 0 \mid X \subset B_\epsilon(0) \oplus Y\}$ and $\delta_u(X, Y) := \delta_l(Y, X)$, respectively. The Hausdorff distance of two sets $A, \ B \subset \mathbb{R}^n$ is a metric given by $\delta_H(A, B) = max\{\delta_l(A, B), \delta_u(A, B)\}$. A multivalued function $f : \mathbb{R}^n \to Com(\mathbb{R}^n)$ is upper-semicontinuous / lower-semicontinuous if for any open set A in Hemimetric space $(Com(\mathbb{R}^n), \delta_u)$/ $(Com(\mathbb{R}^n), \delta_l)$, $f^{-1}(A) := \{x \in \mathbb{R}^n \mid f(x) \in A\}$ is open in $\mathbb{R}^n$. A multi valued function is continuous if it is upper-semicontinuous and lower-semicontinuous, i.e., it is continuous with respect to space $Com(\mathbb{R}^n)$ and metric δ_H.

The Graph of a multi-valued function $f : \mathbb{X} \subset \mathbb{R}^n \to Com(\mathbb{R}^n)$ is $Gr \ f := \{(x, y) \mid x \in \mathbb{X}, \ y \in f(x)\}$. A set valued function is called a correspondence. For a correspondence f, strong pre-image of a set $A \subset \mathbb{R}^n$ is $f^s(A) := \{x \in \mathbb{R}^n \mid f(x) \subset A\}$. For a set sequence $\{A_i\}, \ i \in \mathbb{N}$, $\limsup_{i \to \infty} A_i = \bigcap_{n=1}^{\infty} \bigcup_{i=n}^{\infty} A_i$.

5.1 Characterization of MRIA Sets: The General Case

Consider a discrete-time linear time-invariant system:

$$z(k+1) = Az(k) + \omega, \ \omega \in \mathcal{W}(z(k)), k \in \mathbb{N} \tag{5.1}$$

where $z(k) \in \mathbb{R}^r$ is the current state, $z(k+1)$ is the successor state and $\omega \in \mathbb{R}^r$ is an unknown disturbance taking values in a state dependent set $\mathcal{W}(z(k))$ where

$\mathcal{W}: \mathbb{R}^r \to Com(\mathbb{R}^r)$. The standing assumptions are

Assumption 1. *The state transition matrix A is strictly stable.*

Assumption 2. *The multi-valued function $\mathcal{W}$ is upper-semicontinuous.*

Assumption 3. *There exists a compact set $\mathbb{W} \subset \mathbb{R}^r$ such that for all $z \in \mathbb{R}^r$, $\mathcal{W}(z) \subseteq \mathbb{W}$.*

These assumptions are reasonable for applications of our subsequent results to Model Predictive Control. The following Lemma provides another interpretation of upper-semicontinuity, which is introduced in the Definition Section, [113].

Lemma 5.1.1. *A correspondence f is upper-semicontinuous if for every open set G in $\mathbb{R}^r$, the pre-image $f^s(G)$ is open in $\mathbb{R}^r$.*

We recall the following standard definition [114].

Definition 5.1.1. *A set Ω is an invariant set for the system (5.1) if for all $z \in \Omega$, $Az \oplus \mathcal{W}(z) \subseteq \Omega$.*

Definition 5.1.2. *A nonempty set Ω is an attractor for system (5.1), if for all initial conditions $z(0) \in \mathbb{R}^r$, any solution $z(k)$ at the dynamics $z(k) = Az(k-1) + \omega,\ \omega \in \mathcal{W}(z(k-1))$ converges to Ω as $k \to \infty$. A set is a minimal attractor if it is contained in any attractor of the system.*

In this chapter, existence of the MRIA set for the system (5.1) under the Assumptions 1-3 is established.

Consider the map $\mathcal{W}$, $\mathcal{R}: Com(\mathbb{R}^r) \to 2^{\mathbb{R}^r}$, induced by the set valued function given by:

$$\mathcal{R}(Z) := \{Az + \omega \ : \ z \in Z,\ \omega \in \mathcal{W}(z)\} \tag{5.2}$$

where $2^{\mathbb{R}^r}$ denotes the set of all subsets of $\mathbb{R}^r$. We denote by $\mathcal{R}^j(\cdot)$ the j-th iterate of the map $\mathcal{R}(\cdot)$, given by (5.2), and $\mathcal{R}^0(Z) = Z$ for $Z \subset \mathbb{R}^r$. If $z \in \mathbb{R}^r$, $\mathcal{R}(z) := \{Az + \omega \ : \ \omega \in \mathcal{W}(z)\}$. Thus a set $\Omega \subset \mathbb{R}^r$ is invariant if and only if $\mathcal{R}(\Omega) \subseteq \Omega$. A set $\Omega \subset \mathbb{R}^r$ is an attractor if and only if for all $Z \in Com(\mathbb{R}^r)$, $\limsup_{j\to\infty} \mathcal{R}^j(Z) \subseteq \Omega$.

Lemma 5.1.2. *For any $Z \in Com(\mathbb{R}^r)$, $\mathcal{R}(Z)$ is compact.*

Proof. Since $\mathcal{W}: Z \to Com(\mathbb{R}^r)$ is upper-semicontinuous, so is $\mathcal{R}: Z \to Com(\mathbb{R}^r)$ given by $\mathcal{R}(z) = Az + \mathcal{W}(z)$. Moreover, $\mathcal{R}$ is compact valued and given that Z is compact, $\mathcal{R}(Z) = \bigcup_{z\in Z} \mathcal{R}(z)$ is compact (See page 90 of [113]). □

Therefore, $\mathcal{R}(\cdot)$ maps $Com(\mathbb{R}^r)$ to itself.

Lemma 5.1.3. *Assume Assumptions 1 and 3 hold. Then the set*

$$F := Cl(\bigoplus_{i=0}^{\infty} A^i \mathbb{W}) \tag{5.3}$$

is an attracting invariant set, i.e., $\mathcal{R}(F) \subseteq F$ and for any $K \in Com(\mathbb{R}^r)$,

$$\limsup_{j\to\infty} \mathcal{R}^j(K) \subseteq F$$

.

Proof. The Lemma follows from Assumption 3 and the fact that F is an attracting invariant set for the system $z(k+1) = Az(k) + \omega(k),\ \omega(k) \in \mathbb{W}$. [112]. □

Lemma 5.1.4. *Suppose $\{Z_i\},\ i \in \mathbb{N}$ is a sequence of non-empty compact sets in $\mathbb{R}^r$ where $Z_{i+1} \subseteq Z_i$ for $i \in \mathbb{N}$. Then $\bar{Z} := \bigcap_{i=1}^{\infty} Z_i \neq \emptyset$. Moreover, for all open sets $G \subset \mathbb{R}^r$ such that $\bar{Z} \subset G$, there exists $M \in \mathbb{N}$ such that $Z_M \subset G$.*

Remark 5.1.1. *This is a standard result, see [116].*

Theorem 5.1.1. *Suppose Assumptions 1-3 hold and F is defined by (5.3). Then*

$$\Omega^* := \bigcap_{j=0}^{\infty} \mathcal{R}^j(F) \tag{5.4}$$

is non-empty and compact, i.e., $\Omega^* \in Com(\mathbb{R}^r)$, and

$$\mathcal{R}(\Omega^*) = \Omega^*. \tag{5.5}$$

Proof. The proof is similar to the one given for Theorem 8.2.1 in [113]. According to Lemma 5.1.3, $\mathcal{R}(F) \subseteq F$. Therefore, $\mathcal{R}^j(F) \subseteq \mathcal{R}^{j-1}(F)$ for $j \in \mathbb{N}$. Since $\mathcal{W}(z) \neq \emptyset$ for all $z \in \mathbb{R}^r$, $\mathcal{R}^j(F) \neq \emptyset$. Hence, according to Lemma 5.1.4, $\Omega^* = \bigcap_{j=0}^{\infty} \mathcal{R}^j(F) \neq \emptyset$. If $z \in \Omega^*$, then for all $n > 0$, $z \in \mathcal{R}^n(F)$. Hence, $\mathcal{R}(z) \subset \mathcal{R}^{n+1}(F)$, $n > 0$. Therefore, $\mathcal{R}(z) \subset \Omega^*$ and $\mathcal{R}(\Omega^*) \subseteq \Omega^*$. Now the reverse inclusion need to be proved. Assume $z_0 \in \Omega^* \backslash \mathcal{R}(\Omega^*)$. Then $\mathcal{R}(\Omega^*)$ and z_0 can be separated by disjoint open sets U_1 and U_2. Since $\mathcal{R}(\cdot)$ is upper-semicontinuous, $\mathcal{R}^s(U_1)$ is open and since $\mathcal{R}(\Omega^*) \subset U_1$, $\Omega^* \subset \mathcal{R}^s(U_1)$. Therefore, according to Lemma 5.1.4, there exists $M \in \mathbb{N}$ such that $\mathcal{R}^M(F) \subset \mathcal{R}^s(U_1)$. Hence, $\Omega^* \subset \mathcal{R}^{M+1}(F) \subset U_1$ and $z_0 \in \Omega^* \subset U_1$ which contradicts $z_0 \in U_2$. Therefore, $\Omega^* \backslash \mathcal{R}(\Omega^*) = \emptyset$ and $\mathcal{R}(\Omega^*) = \Omega^*$. □

Theorem 5.1.2. *Suppose Assumptions 1-3 hold. Then $\Omega^* \in Com(\mathbb{R}^r)$, defined in (5.4), is a minimal robust attractor with the basin of attraction being the whole space, $Com(\mathbb{R}^r)$.*

Proof. Given $K \in Com(\mathbb{R}^r)$, according to Lemma 5.1.3,

$$\forall\ \epsilon > 0, \exists\ M_1 \in \mathbb{N},\ s.t\ \forall n \geq M_1,\ \mathcal{R}^n(K) \subset F \oplus B_\epsilon(0) \tag{5.6}$$

where $B_\epsilon(z_0) := \{z \in \mathbb{R}^r \mid \|z - z_0\| < \epsilon\}$. For any set $S \in Com(\mathbb{R}^r)$, $S \oplus B_\epsilon(0) = \bigcup_{z \in S} B_\epsilon(z)$. Therefore, $S \oplus B_\epsilon(0)$ is an open set. From Assumption 2, $\mathcal{R}(\cdot)$ is upper-semicontinuous. Hence, according to Lemma 5.1.1, $\mathcal{R}^s(\mathcal{R}(S) \oplus B_\epsilon(0))$ is open. Since the set F, defined in (5.3), is compact and contained in the open set $\mathcal{R}^s(\mathcal{R}(F) \oplus B_\epsilon(0))$, for $\epsilon > 0$, there exists $\delta(\epsilon)$ such that $F \oplus B_{\delta(\epsilon)}(0) \subset \mathcal{R}^s(\mathcal{R}(F) \oplus B_\epsilon(0))$. Hence, for all $\epsilon > 0$, there exists $\delta(\epsilon) > 0$ such that

$$\mathcal{R}(F \oplus B_{\delta(\epsilon)}(0)) \subset \mathcal{R}(F) \oplus B_\epsilon(0). \tag{5.7}$$

According to Lemma 5.1.4 and (5.4), given $\epsilon > 0$, there exists $M_2(\epsilon) > 0$ such that

$$\mathcal{R}^{M_2}(F) \subset \Omega^* \oplus B_{\epsilon/2}(0). \tag{5.8}$$

Considering (5.7), it is implied that $\mathcal{R}(\mathcal{R}^{M_2-1}(F) \oplus B_{\delta(\epsilon/2)}(0)) \subset \mathcal{R}^{M_2}(F) \oplus B_{\epsilon/2}(0)$. Applying (5.7) recursively, it is deduced

$$\mathcal{R}^{M_2}(F \oplus B_{\delta^{M_2}(\epsilon/2)}(0)) \subset \mathcal{R}^{M_2}(F) \oplus B_{\epsilon/2}(0) \tag{5.9}$$

where $\delta^{i+1}(\epsilon/2) := \delta^i(\epsilon/2)$ and $\delta^0(\epsilon/2) = \epsilon/2$. (5.8) and (5.9) imply

$$\mathcal{R}^{M_2}(F \oplus B_{\delta^{M_2}(\epsilon/2)}(0)) \subset \Omega^* \oplus B_\epsilon(0). \tag{5.10}$$

According to (5.6), if $\hat{\delta} := \delta^{M_2}(\epsilon/2))$, there exists $M_1(\hat{\delta}) \in \mathbb{N}$ such that for all $n > M_1(\hat{\delta})$

$$\mathcal{R}^n(K) \subset F \oplus B_{\hat{\delta}}(0). \tag{5.11}$$

(5.11), for all $n > M_1 + M_2$, implies that $\mathcal{R}^n(K) = \mathcal{R}^{M_2}(\mathcal{R}^{n-M_2}(K)) \subset \mathcal{R}^{M_2}(F \oplus B_{\hat{\delta}}(0))$ and from (5.10), $\mathcal{R}^{M_2}(F \oplus B_{\hat{\delta}}(0)) \subset \Omega^* \oplus B_\epsilon(0)$. Therefore, for all $n > (M_1 + M_2)(\epsilon)$, $\mathcal{R}^n(K) \subset \Omega^* \oplus B_\epsilon(0)$. Hence, $\limsup_{n\to\infty} \mathcal{R}^n(K) \subseteq \Omega^*$. Therefore Ω^* is an attractor. Assume Ω is another attractor. Then $\limsup_{n\to\infty} \mathcal{R}^n(\Omega^*) \subseteq \Omega$. Since $R(\Omega^*) = \Omega^*$, $\Omega^* \subset \Omega$. □

In the sequel, as a special case, existence and uniqueness of minimal attracting sets for linear systems subject to constraints and additive bounded disturbance that evolves according to a given dynamic model are considered.

5.2 Linear System and Disturbance Models

In this section, the problem, where the disturbance evolves inside a compact set according to a known disturbance model is formulated. We consider a discrete-time linear system

$$x(k+1) = Ax(k) + Bu(k) + Cw(k),\ k \in \mathbb{N} \tag{5.12}$$

where $x(k) \in \mathbb{R}^n$ is the current state, $u(k) \in \mathbb{R}^m$ is the current control, $w(k) \in \mathbb{R}^p$ is the disturbance, $x(k+1) \in \mathbb{R}^n$ is the successor state and matrices A, B and C are of compatible dimensions.

The disturbance equation can be equivalently expressed as follows

$$w(k) = Fw(k-1) + Gv(k-1), \tag{5.13}$$

where $w(k-1) \in \mathbb{R}^p$ is the predecessor disturbance , $v(k-1) \in \mathbb{R}^l$ is the predecessor disturbance model input and $w(k) \in \mathbb{R}^p$ is the current disturbance.

The disturbance model state and input are subject to the following constraints:

$$w(k) \in W,\ w(k-1) \in W \text{ and } v(k-1) \in V, \tag{5.14}$$

where $W \in Com(\mathbb{R}^p)$ and $V \in Com(\mathbb{R}^l)$.

Remark 5.2.1. *Assuming that C is full column rank, at each time instant k, the predecessor disturbance $w(k-1)$ is accessible via the following relation:*

$$w(k-1) = (C^TC)^{-1}C^T(x(k) - Ax(k-1) - Bu(k-1)).$$

However, the current value of disturbance $w(k)$ and the disturbance model input $v(k)$ is unknown.

Remark 5.2.2. *By measuring the predecessor disturbance, the disturbance model (5.13)-(5.14) can be constructed using system identification techniques applied to disturbance measurement.*

In the following, it is demonstrated that (5.12)-(5.14) can be represented as linear system with additive, state-dependent bounded disturbance.

The constraint (5.14) induces the following constraint

$$\begin{aligned} w(k) &\in \{Fw(k-1) + Gv(k-1) \mid v(k-1) \in V,\ Fw(k-1) + Gv(k-1) \in W\} \\ &= (Fw(k-1) + GV) \cap W. \end{aligned} \tag{5.15}$$

Defining a correspondence $g : W \to Com(W)$ as

$$g(w) := (Fw + GV) \cap W. \tag{5.16}$$

Note that (5.13) and(5.14) imply

$$w(k) \in g(w(k-1)),\ w(k-1) \in W. \tag{5.17}$$

Clearly for all $w \in W,\ g(w) \neq \emptyset$. It is assumed that at each time instant, the current state $x(k)$ and the predecessor disturbance $w(k-1)$ are accessible to the controller but $v(k-1)$ or the current value of disturbance, $w(k)$, are not. Given the known state and disturbances $x(k)$ and $w(k-1)$ at each time instant k, it is assumed that the controller applies a linear feedback given by:

$$u(k) = Lx(k) + Mw(k-1), \tag{5.18}$$

where L is chosen such that the matrix $A_L := A+BL$ is strictly stable. Incorporating the control law (5.18), the system and disturbance dynamics (5.12) and (5.13) subject to disturbance constraints (5.14) take the following closed-loop form:

$$\begin{pmatrix} x(k+1) \\ w(k) \end{pmatrix} = \begin{pmatrix} A+BL & BM \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x(k) \\ w(k-1) \end{pmatrix} + \begin{pmatrix} C \\ I \end{pmatrix} w(k), \tag{5.19}$$

$$w(k) \in g(w(k-1)). \tag{5.20}$$

Introducing the augmented state $z(k) = (x(k), w(k-1))$, (5.19) takes the following form:

$$z(k+1) = \tilde{A}z(k) + \omega,\ \omega \in \mathcal{W}(z(k)), \tag{5.21}$$

where $\mathcal{W} : \mathbb{R}^n \times W \to \mathbb{R}^n \times W$ is given by

$$\mathcal{W}(z) := \tilde{B}g([0\ I]z), \tag{5.22}$$

and

$$\tilde{A} := \begin{pmatrix} A+BL & BM \\ 0 & 0 \end{pmatrix}\ ,\ \tilde{B} := \begin{pmatrix} C \\ I \end{pmatrix}, \tag{5.23}$$

Remark 5.2.3. *Note that by augmenting (5.12) to (5.13) and viewing $v(k-1)$ as a disturbance input, a linear discrete time system with a set bounded input is obtained. While such a system can be treated using existing techniques in the literature, less conservative results can be developed by including additional information that $w(k) \in W,\ w(k-1) \in W$.*

The augmented system (5.21) has the form of system (5.1). In the sequel the results of Section 5.1 are used to characterize the MRIA set for the augmented system (5.19). Our assumptions for system (5.19) that are related to assumptions 1-3 are now summarized as follows:

Assumption 4. *The sets V and W are compact.*

Assumption 5. *For all $w \in W$, $g(w) = (Fw \oplus GV) \cap W \neq \emptyset$.*

Assumption 6. *The matrix $A_L := A + BL$ is strictly stable.*

Note that Assumption 6 can be easily satisfied if the pair (A, B) is stabilizable, e.g. by any pole placement techniques.

5.3 Characterization of MRIA Sets: The Special Case

In this section, existence of a MRIA set is established once the disturbance dynamically evolves inside a compact set according to a dynamic. To analyze existence and uniqueness of a minimal invariant set for augmented system (5.21), existence and uniqueness of the fixed point for the mapping

$$\mathcal{R}(\Omega) := \{\tilde{A}z + w \mid z \in \Omega,\ w \in \mathcal{W}(z)\} \tag{5.24}$$

is considered.

Proposition 5.3.1. *Suppose Assumptions 4 and 5 hold. Then the correspondence g defined in (5.16) is upper-semicontinuous.*

Proof. The graph of g is

$$Gr\ g = \{(x, y) \mid x \in W,\ y \in (Fx + GV) \cap W\}.$$

Since V and W are compact, the sets $G_1 := \{(x, y) \mid x \in W,\ y \in Fx + GV\}$ and $G_2 := \{(x, y) \mid x \in W,\ y \in W\}$ are compact. Hence, $Gr\ g = G_1 \cap G_2$ is compact. Therefore, since the range of g is the compact set W and its graph is closed, g is upper-semicontinuous [115]. □

Remark 5.3.1. *We note that even if the sets V and W are compact and convex with zero in their interior, the multifunction g may not be continuous.*

The following example illustrates the above observation (see also Figure fig3MRIA:

Example 5.3.1. *Let $\mathbb{W}$ be the following convex cone:*

$$\mathbb{W} = \left\{(1-\alpha)\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T + \alpha \begin{bmatrix} r\cos(\theta) & r\sin(\theta) & 0 \end{bmatrix}^T | r \in [0\ 1], \alpha \geq 0,\ \theta \in [-\pi\ \pi]\right\} \cap \{w | \|w\|_\infty \leq 10\} \tag{5.25}$$

and $\mathbb{V} = [-1\ 1]$. Moreover, let us assume $F = I$ and $G = [0\ 0\ 1]^T$. Consider the sequence $\{w_n := [\cos(\pi/2n)\ \sin(\pi/2n)\ 0]^T\}$. According to definition (5.16),

$$g(w_n) = (w_n + G\mathbb{V}) \cap \mathbb{W} = \{[\cos(\pi/2n)\ \sin(\pi/2n)\ \alpha]^T | \alpha \in [-1\ 0],\ n \in \mathbb{N}\} \tag{5.26}$$

if $w^0 := [1\ 0\ 1]^T$ and $\hat{w} := [1\ 0\ 0]^T$, then, if Euclidian norm is used, the following holds true: $\lim_{n\to\infty} w_n \to \hat{w}$, $w^0 \in g(\hat{w})$, and

$$\lim_{n\to\infty} d(w^0, g(w_n)) = \sqrt{((1 - \cos(\pi/2n))^2 + \sin(\pi/2n)^2 + 1)} \to 1.$$

Therefore $g(\cdot)$ is not continuous.

Proposition 5.3.2. *Suppose Assumptions 4 and 5 hold. Then the correspondence $\mathcal{W}$, defined in (5.22), is upper-semicontinuous.*

Proof. Functions $h_1 : \mathbb{R}^{n+p} \to \mathbb{R}^p$ given by $h_1(z) := \{[0\ I]z\}$ and $h_2 : \mathbb{R}^p \to \mathbb{R}^{n+p}$ given by $h_2(z) := \{\tilde{B}z\}$ are continuous and g is upper-semicontinuous (according to Proposition 5.3.1). Therefore, $\mathcal{W}(z) = h_2 \cdot\ g \cdot\ h_1$ is upper-semicontinuous. □

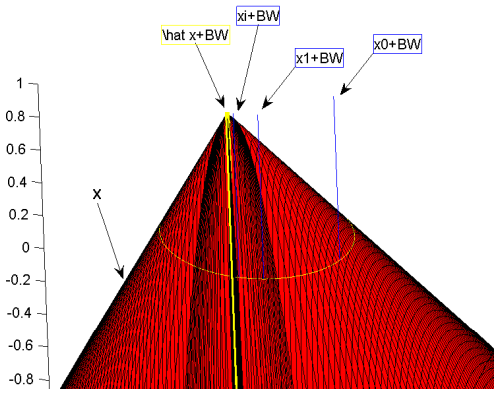

Figure 5.1: Pictorial description of $\mathbb{W}$ and function $g(\cdot)$

Theorem 5.3.1. *Suppose Assumptions 4-6 hold and*

$$F := Cl(\bigoplus_{i=0}^{\infty} \tilde{A}^i \tilde{B} W). \tag{5.27}$$

Then

$$\Gamma^* := \bigcap_{j=0}^{\infty} \mathcal{R}^j(F) \tag{5.28}$$

is non-empty and compact, i.e., $\Gamma^* \in Com(\mathbb{R}^n)$*, and* $\mathcal{R}(\Gamma^*) = \Gamma^*$*, with* $\mathcal{R}$ *defined in (5.24). Moreover,* Γ^* *is robust invariant and is the minimal attractor for the augmented system (5.21) with the basin of attraction being the whole space,* $Com(\mathbb{R}^n)$*.*

Proof. According to Assumption 6, $\tilde{A}$ is stable. Hence, Assumption 1 is satisfied for augmented system (5.21). According to Assumptions 4 and 5 and Proposition 5.3.2, $\mathcal{W}$ is upper-semicontinuous. So Assumption 2 is satisfied for augmented system (5.21). Since $g(w) \subseteq W$ for all $w \in W$, $\mathcal{W}(z) \subseteq \tilde{B}W$ for all $z \in \mathbb{R}^n \times W$. Therefore, Assumption 3 is satisfied for augmented system (5.21). With Assumptions 1-3 satisfied, according to Theorem 5.1.1, Γ^* is non-empty and is a fixed point of $\mathcal{R}$. Moreover, according to Theorem 5.1.2, Γ^* is the minimal attractor for the augmented system (5.21) with the basin of attraction being the whole space. The robust invariance is the immediate result of $\mathcal{R}(\Gamma^*) = \Gamma^*$. □

We consider the uniqueness of the fixed point Γ^* of the mapping $\mathcal{R}$ in the sequel.

Lemma 5.3.1. *Suppose Assumptions 4-6 hold and* $\Delta \in Com(W)$ *be the fixed point of* g*, i.e.* $g(\Delta) = \Delta$*. If* $\bar{\Gamma}_n := \{\sum_{i=0}^{n} \tilde{A}^{n-i} \tilde{B} w_i | w_0 \in \Delta, w_{i+1} \in g(w_i),\ i = 0, \cdots, n-1\}$*,*

there exists $\bar{\Gamma} \in Com(\mathbb{R}^n \times W)$ *s.t*

$$\bar{\Gamma}_n \to \bar{\Gamma}, \ as \ n \to \infty, \tag{5.29}$$

in the space $Com(\mathbb{R}^n \times W)$ *with Hausdorff metric.*

Proof. Let us define $g^{-1} : \Delta \mapsto Com(\Delta)$ as follows

$$g^{-1}(w) := \{x \in \Delta | g(x) = w\}. \tag{5.30}$$

Since $g(\Delta) = \Delta$, $\forall\ w \in \Delta,\ g^{-1}(w) \neq \emptyset$. It can be easily shown that

$$\begin{aligned} \mathbb{E}^n &:= \{[w_0, \cdots, w_n] | w_0 \in \Delta, w_{i+1} \in g(w_i),\ i = 0, \cdots, n-1\} \\ &= \{[w_0, \cdots, w_n] | w_n \in \Delta,\ w_i \in g^{-1}(w_{i+1}), i = 0, \cdots, n-1\}. \end{aligned} \tag{5.31}$$

Therefore $\bar{\Gamma}_n = \{\sum_{i=0}^n \tilde{A}^i \tilde{B} w_i | w_0 \in \Delta, w_{i+1} \in g^{-1}(w_i),\ i = 0, \cdots, n-1\}$. Since A_L is Schur, so is $\tilde{A}$. Therefore, the set $F_\infty := \bigoplus_{i=0}^\infty \tilde{A}^i \tilde{B} \Delta$ is bounded. Therefore,

$$\forall\ \epsilon > 0,\ \exists\ N(\epsilon) > 0,\ s.t.\ \forall n > N(\epsilon),\quad \tilde{A}^n F_\infty \subset B_\epsilon(0) = \{w | \|w\| < \epsilon\} \tag{5.32}$$

Given $\epsilon > 0$, and $r > N(\epsilon/2)$, if $x \in \bar{\Gamma}_r$, then

$$x = \sum_{i=0}^r \tilde{A}^i \tilde{B} w_i = \sum_{i=0}^N \tilde{A}^i \tilde{B} w_i + \sum_{i=N+1}^r \tilde{A}^i \tilde{B} w_i, \text{ for some } [w_0, \cdots, w_r] \in \mathbb{E}^n \tag{5.33}$$

But $\sum_{i=0}^N \tilde{A}^i \tilde{B} w_i \in \bar{\Gamma}_N$ and $\sum_{i=N+1}^r \tilde{A}^i \tilde{B} w_i \in B_{\epsilon/2}$. Therefore for all $x \in \bar{\Gamma}_r$, $d(x, \bar{\Gamma}_N) < \epsilon/2$ and hence,

$$\bar{\Gamma}_r \subset \bar{\Gamma}_N \oplus B_{\epsilon/2}(0) \tag{5.34}$$

If $x \in \bar{\Gamma}_N$, $\exists\ [w_0, \cdots, w_N] \in \mathbb{E}^N,\ s.t.\ x = \sum_{i=0}^N \tilde{A}^i \tilde{B} w_i$. Choosing $\hat{w}_1, \cdots, \hat{w}_{r-N}$ s.t. $[\hat{w}_0, w_1, \cdots, \hat{w}_{r-N}] \in \mathbb{E}^{r-N}$, and $\hat{w}_0 \in g(w_N)$, we have $x + \sum_{i=1}^{r-N} \tilde{A}^{N+i} \tilde{B} \hat{w}_i \in \bar{\Gamma}_r$. Since $\sum_{i=1}^{r-N} \tilde{A}^{N+i} \tilde{B} \hat{w}_i \in B_{\epsilon/2}(0)$, we have $\forall\ x \in \bar{\Gamma}_N\ d(x, \bar{\Gamma}_r) < \epsilon/2$, and hence,

$$\bar{\Gamma}_N \subset \bar{\Gamma}_r \oplus B_{\epsilon/2}(0). \tag{5.35}$$

From (5.34) and (5.35), we have

$$\forall\ r > N,\ \delta_H(\bar{\Gamma}_N, \bar{\Gamma}_r) < \epsilon/2. \tag{5.36}$$

From (5.36), we have

$$\forall\ r_1, r_2 > r,\ \delta_H(\bar{\Gamma}_{r_1}, \bar{\Gamma}_{r_2}) < \delta_H(\bar{\Gamma}_N, \bar{\Gamma}_{r_1}) + \delta_H(\bar{\Gamma}_N, \bar{\Gamma}_{r_2}) < \epsilon/2 + \epsilon/2 = \epsilon. \tag{5.37}$$

Therefore, $\{\bar{\Gamma}_n : n \in \mathbb{N}\}$ is Cauchy and by completeness of Hausdorff metric in $Com(\mathbb{R}^n \times W)$, $\bar{\Gamma}_n$ converges to some $\bar{\Gamma} \in Com(\mathbb{R}^n \times W)$ □

Lemma 5.3.2. *Suppose Assumptions 4-5 hold. Then the mapping $g(\cdot)$ has a minimal attracting fixed point Δ, that is,*

$$g^n(W) \to \Delta \text{ as } n \to \infty. \tag{5.38}$$

Proof. Proof follows from the fact that $g^{i+1}(W) \subset g^i(W)$ and Lemma 5.1.4. □

Theorem 5.3.2. *Suppose Assumptions 4-6 hold. Then,*

$$\forall\ \Lambda \in \mathbb{R}^n,\ \mathcal{R}^n(\Lambda \times W) \to \bar{\Gamma} \text{ as } n \to \infty, \tag{5.39}$$

where $\bar{\Gamma}$, defined in (5.29). Moreover, $\bar{\Gamma} = \Gamma^*$, defined in (5.28).

Theorem 5.3.3. *If in (5.16), $F = G = I$ and W and V are compact, and have zero in the interior, and W is path connected, then W is the unique attracting fixed point of the mapping g. Moreover,*

$$\forall\ \Lambda \in Com(W),\ g^n(\Lambda) \to W,\ \text{as } n \to \infty.$$

Remark 5.3.2. *We note that if W and V are compact, convex, and zero in their interior and the matrix F is Schur, the mapping g may not have a unique fixed point. Consider the sets:*

$$W = \{x \in \mathbb{R}^2 \mid |[4\ 1]x| \le 4,\ |[4\ -1]x| \le 4\},\ V = [-1\ 1] \in Com(\mathbb{R}) \tag{5.40}$$

with $F = \begin{bmatrix} 0 & -0.5 \\ 0.5 & 0 \end{bmatrix}$ and $G = [2\ 4]^T$. It can be checked that the sets W and $\{[0\ 4]^T\}$ are fixed points of the mapping g defined in (5.16). Therefore, the mapping g may not have a unique fixed point even if W and V are compact and convex and have zero in their interior and F is Schur.

Remark 5.3.3. *We note that the results of this section holds as long as A_L is Schur, independent of the choice of the Matrix M. However, the matrix M contributes in the size of the minimal attractor $\bar{\Gamma}$.*

5.4 MRIA for Robust MPC

In this section, the control of the system (5.12) subject to additive disturbance with the dynamics (5.13) is considered. We use the minimal attractor, Γ^*, which is introduced in Section 5.2 to construct an MPC controller which provides the robust asymptotic stability of Γ^*.

The system (5.12) and the disturbance dynamics (5.13) can be written in the following augmented form:

$$z(k+1) = \begin{pmatrix} A & CF \\ 0 & F \end{pmatrix} z(k) + \begin{pmatrix} B \\ 0 \end{pmatrix} u(k) + \begin{pmatrix} CG \\ G \end{pmatrix} v(k-1) \tag{5.41}$$

where $z(k) = [x(k)^T \ (w(k-1))^T]^T$. We note that at any time instant k the controller knows the current state $x(k)$ and the predecessor disturbance $w(k-1)$ but does not know the disturbance model input $v(k-1)$ nor the current disturbance $w(k)$. Therefore, at each time instant the state $z(k)$ is known and the knowledge of the predecessor disturbance is used in the feedback law generated by Tube MPC. We define the corresponding nominal system

$$y(k+1) = \bar{A}y(k) + \bar{B}u(k), \ k \in \mathbb{N} \tag{5.42}$$

where $y(k)$ is the current state, $u(k)$ is the current control action, and $y(k+1)$ is the successor state of the nominal system and

$$\bar{A} = \begin{pmatrix} A & CF \\ 0 & F \end{pmatrix}, \ \bar{B} = \begin{pmatrix} B \\ 0 \end{pmatrix} \tag{5.43}$$

We assume the system (5.41) is subject to constraints:

$$x(k) \in \mathbb{X}, u(k) \in \mathbb{U}, w(k) \in W, \ w(k-1) \in W \text{ and } v(k-1) \in V, \tag{5.44}$$

where the sets $\mathbb{X}$, W, and V are compact and convex with zero in their interior. Using Theorem 3.1, if $u = [L \ M]z$, the MRIA set for the system (5.41), Γ^*, is calculated by applying the mapping $\mathcal{R}(\cdot)$ on the set F defined in (5.27).

Considering the system (5.41) and the corresponding nominal system (5.42), the

tube MPC approach introduced in [53] is applied. Let us define

$$V_N(x, \mathbf{u}) := \sum_{i=0}^{N} l(x(i), u(i)) + V_f(x(N)),$$

where $\mathbf{u} = [u(0), u(1), , \cdots, u(N-1)]$ and l and V_f are positive definite, chosen to properly guarantee the stability of the linear system, once the MPC strategy is applied to the linear system in the absence of additive disturbance. Moreover, let us define

$$K := [L\ M],\ \bar{\mathbb{Z}} := \mathbb{X} \times W \sim \Gamma^*,\ \bar{\mathbb{U}} := \mathbb{U} \sim K\Gamma^* \tag{5.45}$$

Observing the current state z, the optimization problem $P_N(z)$ is solved:

$$\begin{aligned} &\min_{z_0, \mathbf{u}} V_N(z_0, \mathbf{u}) \\ &\text{subject to: } z(i+1) = \bar{A}z(i) + \bar{B}u(i),\ z(i) \in \bar{\mathbb{Z}},\ z(N) \in \mathbb{Z}_f,\ u(i) \in \bar{\mathbb{U}},\ z \in z_0 \oplus \Gamma^* \end{aligned} \tag{5.46}$$

where $\mathbb{Z}_f$ is an invariant set for the nominal system (5.42) contained in $\bar{\mathbb{Z}}$. The solution to $P_N(z)$ yields the optimal control sequence $\mathbf{u}^*(z) := [u^*(0; z), \cdots, u^*(N-1; z)]$ and state sequence $\mathbf{z}^*(z) := [z^*(0; z), \cdots, z^*(N-1; z)]$. The implicit control law $\kappa_N^*(\cdot)$ is

$$\kappa_N^*(z) := u^*(0; z) + K(z - z^*(0; z)). \tag{5.47}$$

Using an analysis similar to the one introduced in [53], it can be shown that by applying the control law (5.47), the state in the system (5.41) converges to the MRIA set Γ^*.

Remark 5.4.1. *Note that as guaranteed by Theorem 5.3.1, the minimal attractor set Γ^* is robust positive invariant.*

5.5 Computation of an approximation of MRIA

As illustrated in Lemma 5.3.1, to calculate the MRIA set $\bar{\Gamma}$, infinite number of iterations is required. However, the set $\bar{\Gamma}$ can be approximated as closely as desired by $\bar{\Gamma}_n$ when n is sufficiently large. In this section, we consider the rate-bounded disturbance where in (5.13), $F = G = I$ and propose a method to calculate $\bar{\Gamma}_n$ recursively as n increases. We assume that the sets W and V are polytopes.

Since disturbance dynamics can be written as $w(k) = w(k-1)+v(k-1),\ w(k-1) \in$

$W,\ v \in \mathcal{V}(w(k-1)) := \{v \in V \mid w(k-1)+v \in W\}$, (5.21) can be written as

$$z(k+1) = \hat{A}z(k) + \tilde{B}v(k),\ z(k) = [x(k)^T\ w(k-1)^T]^T \tag{5.48}$$

where $v(k) \in \mathcal{V}([0\ I]z(k))$ and

$$\hat{A} := \begin{pmatrix} A+BL & BM+C \\ 0 & I \end{pmatrix} \tag{5.49}$$

Since $\bar{\Gamma}_n = \mathcal{R}^n(\{0\})$, by induction, it can be easily shown that $\bar{\Gamma}_n$ is a polytope for all $n > 0$. To calculate an arbitrary close approximation of $\bar{\Gamma}$ by $\bar{\Gamma}_k$ for sufficiently large k, we proceed by providing a method to calculate $\bar{\Gamma}_{n+1}$ once $\bar{\Gamma}_n$ is a given polytope. Given $\bar{\Gamma}_n$, we have

$$\bar{\Gamma}_{n+1} = \mathcal{R}(\bar{\Gamma}_n) = \{\hat{A}z + \tilde{B}v \mid z \in \bar{\Gamma}_n,\ v \in \mathcal{V}([0\ I]z)\}.$$

If we define $y := \hat{A}z + \tilde{B}v$. Then

$$\bar{\Gamma}_{n+1} = \left\{ y \in \mathbb{R}^{n+p} \mid \exists\ v \in V\ s.t.\ \begin{array}{c} \hat{A}^{-1}(y-\tilde{B}v) \in \bar{\Gamma}_n \\ v + [0\ I]\hat{A}^{-1}(y-\tilde{B}v) \in W \end{array} \right\}.$$

If the set Θ_n is defined as

$$\Theta_n := \left\{ [y^T\ v^T]^T \in \mathbb{R}^{n+p+l} \;\middle|\; \begin{array}{c} v \in V \\ \hat{A}^{-1}(y-\tilde{B}v) \in \bar{\Gamma}_n \\ v + [0\ I]\hat{A}^{-1}(y-\tilde{B}v) \in W \end{array} \right\}. \tag{5.50}$$

then

$$\bar{\Gamma}_{n+1} = \{y \in \mathbb{R}^{n+p} \mid \exists\ v \in \mathbb{R}^l\ s.t. [y^T\ v^T]^T \in \Theta_n\} \tag{5.51}$$

Let the projection of a set $A \subset S \times X$ onto the space S be defined as

$$Proj_S(A) : \{x \in S \mid \exists\ y \in X\ s.t.\ [x^T\ y^T] \in S \times X\}.$$

Then (5.51) can be written as

$$\bar{\Gamma}_{n+1} = Proj_{\mathbb{R}^{n+p}}(\Theta_n). \tag{5.52}$$

Since $\bar{\Gamma}_n$ is a polytope, Θ_n defined in (5.50) is also a polytope, hence the projection $\bar{\Gamma}_{n+1}$ is also a polytope.

The Keerthi-Gilbert projection algorithm [117] may, for instance, be used to calculate the projection of a polytope into a subspace.

5.6 Numerical Example

In this section, we demonstrate the effect of rate bound of the disturbance on the MRIA set. We revisit roll dynamics of a ship equipped with fin stabilizers described in Chapter 4. The discrete-time model of roll dynamics, (4.50), with feedback gain K introduced in Section 4.4 leads to

$$x(k+1) = (A_d + KB_d)x(k) + B_w w(k) \tag{5.53}$$

where $x(k) = [\phi(k)\ p(k)]^T$, $w(k) : \frac{\tau_w(k)}{I_{\phi\phi}}$ is the normalized wave moment with $|w(k)| \leq 1$ which acts as unmeasured disturbance, and matrices A_d, K, B_d, and B_w are introduced in (4.51).

As illustrated before, the MRIA set for the system (5.53) plays a pivotal roll in the design of constrained robust controllers. Practically, the wave moment does not vary arbitrarily and is rate-bounded. We consider two cases where the normalized rate bound on the wave moment, i.e. $r_w := w(k+1) - w(k)$, is ∞ and 0.1, respectively. Figure 5.2 shows the MRIA sets corresponding to the two rates in the lifted space, viewed from above. Due to point of view, it also shows the projections of MRIA sets onto $\mathcal{R}^2$ which are MRIA sets in the state space of the system (5.53). Clearly, in the lifted space the MRIA is considerably smaller once the rate-boundedness of the disturbance w is considered. Moreover, the MRIA set for system (5.53), i.e. projection of MRIA set from lifted space onto $\mathcal{R}^2$, is sizably larger if the rate-boundedness is neglected. The reduction of size in MRIA set when rate bounds are considered leads to less conservative robust controller design.

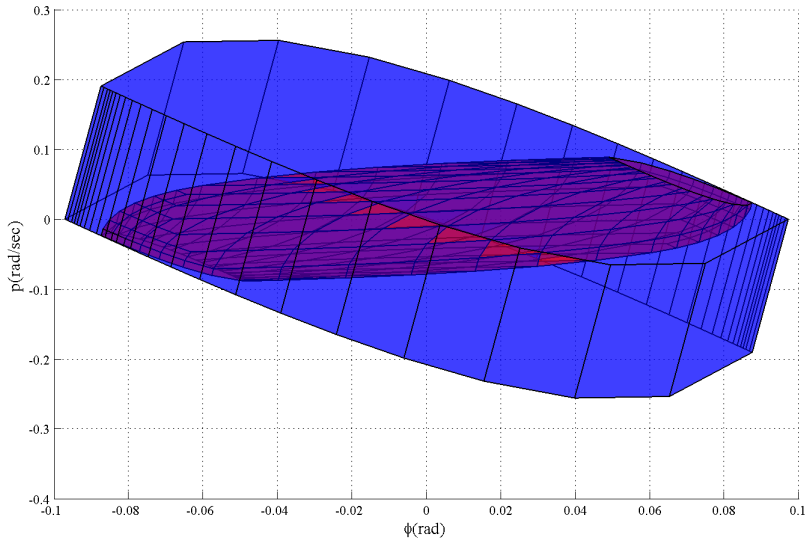

Figure 5.2: MRIA sets for $r_w = \infty$ (blue) and $r_w = 0.1$ (red), viewed from above (third axis is not shown).

5.7 Conclusion

In this chapter, the attractiveness and minimality of invariant sets for linear systems subject to bounded state-dependent additive disturbances is examined. Assuming upper-semi-continuity and boundedness of the correspondence associated with additive disturbance, existence and uniqueness are proved for a minimal attractor set, which is robust invariant. In case where the dynamics of disturbance are known, the results are use to characterize a minimal attractor set. The minimal attractor set can be employed in the design of robust MPC strategies to achieve robust stability, improve control response and reduce conservativeness.

Chapter 6

Conclusion and future research directions

6.1 Conclusion

The purpose of this thesis was to develop a fast MPC solver and a efficient robust MPC approach to reduce computation time and effort as well as conservatism. In MPC, a control sequence is determined at every sampling time instant to minimize a specified cost function defined for a discrete-time system model, then the first element of the optimal sequence is used as the control action. The cost function of MPC is chosen so that a desired performance is achieved.

Even though, many off-line computed strategies were proposed so far for implementation of MPC, in many applications, on-line implementation of MPC is the best, if not the only, choice due to model changing on-line, nonlinearity of the system or constraints and the need for a long length of horizon. Long length of control horizon may be necessary in many applications to acquire feasibility, large domain of attraction, stability, and enhancement of performance. However, the number of partitions of state space in off-line strategies grows exponentially as the length of control horizon increases. Hence, long control horizons are not directly achievable with off-line approach.

In this thesis a computationally efficient MPC solver is introduced, using the optimal solution at the time $(k-1)$ to approximate and further refine the solution at the time k. The idea can be realized by calculating the perturbation in optimal solution, viewing the state deviation at the time k from state at the time $(k-1)$ as a perturbation in initial state in an optimal control problem. The perturbation from optimal solution can be computed using Neighboring Extremal (NE) method originally developed in the 70's for problems without state constraints.

In this dissertation, the NE method is generalized to discrete-time systems subject to general inequality constraints on inputs and states. Its computational complexity

is of order N, where N is length of horizon, as opposed to N^3 for SQP-like approaches [59]. Moreover, a second order sufficient optimality condition for the nominal optimal solution is provided that is verifiable with computational complexity of order N.

Another contribution of this dissertation was to combine the developed NE method (or perturbation analysis) with sequential quadratic programming (SQP) based on active set method to provide Integrated Perturbation Analysis and Sequential Quadratic Programming (InPA-SQP) approach. The InPA-SQP provides faster convergence to an optimal solution as compared to SQP.

The InPA-SQP was demonstrated in two experimental applications. As the first application, MPC is implemented based on the proposed InPA-SQP to regulate the output voltage of a DC/DC converter with peak current protection. To investigate the voltage regulation of a full bridge DC/DC based power conditioning system, an experimental testbed was developed at the University of Michigan. The DC/DC converter has very fast dynamics and therefore requires an efficient MPC implementation algorithm to achieve sub-millisecond sampling time. By employing InPA-SQP method to solve the constrained optimal control problem, 300 μs sampling times is achieved and nonlinear constraints are handled. The experimental results reveal that the MPC algorithms successfully achieved voltage regulation and peak current protection.

As the second application, InPA-SQP is also used for control of a model ship in Marine Hydro-Dynamic Lab (MHL) to follow a pre-specified path. InPA-SQP was used for the path following problem for a model ship via rudder control. 3-DoF simplified nonlinear and linear models are adopted in the controller design and a corresponding 6-DoF nonlinear container model was used in simulations in order to study and compare the performance of the MPC with the linear and nonlinear model. The InPA-SQP algorithm was used to implement both the linear and nonlinear MPC on a model ship to experimentally validate the algorithm and compare the performance. Experimental results show the effectiveness of the proposed MPC solver.

Besides computational feasibility, it is important for MPC strategy to maintain stability and constraint compliance in the presence of model uncertainties and disturbances. In this dissertation, a robust control method was introduced for linear discrete-time systems subject to mixed input-state constraints. The proposed scheme, which is also based on the constraint tightening approach, has several special features. First, unlike the robust MPC approaches, our proposed method does not involve repeated online optimization to determine the control action. Second, under appropriate and easily verifiable conditions, the proposed controller guarantees recursive

feasibility. Third, the minimal invariant set corresponding to the off-line calculated state feedback is an attractor, i.e., all trajectories will converge to this set. As an example to illustrate the applications of the proposed algorithm, the roll control problem for a high speed ship equipped with stabilizing fins was considered. We show that the roll motion of the ship in the presence of wave disturbance can be stabilized using the proposed algorithm while the input-state constraints and input saturation constraints can be effectively enforced while, no optimization problem is solved except for a linear programming problem to address initial feasible solution seeking.

Moreover, it was shown in this thesis that if the dynamics of measured disturbance are taken into account the attractor set associated with the constrained robust control methods can be made considerably smaller thereby leading to a less conservative controller design.

In summary, the contributions of this dissertation are highlighted as follows:

- A NE method for constrained discrete-time systems is developed and a sufficient condition for existence of NE solution was provided.
- The NE method was integrated with SQP to provide an efficient MPC solver, called InPa-SQP.
- The InPA-SQP method effectiveness was demonstrated in two experimental applications. One application is voltage regulation of a DC/DC converter. The other one is path following of a model container ship.
- A robust model based constrained control strategy was proposed that does not require repeated solving an optimization problem on-line while achieving convergence to a minimal invariant set.
- Existence of minimal invariant attractor sets for systems subject to bounded additive state-dependent disturbance was shown for linear systems.
- The above result is used to characterize minimal robust invariant attractor sets (MRIA) for linear systems subject to bounded disturbances where the dynamics and past values of disturbance is known. It is shown that the MRIA sets can be much smaller if the dynamics of disturbance are taken into account.

In the next section possible future directions are discussed.

6.2 Future research directions

6.2.1 NE solution for distributed systems

The closed form NE solution was derived using the sparsity of the cost matrices appearing in the arising QP problem. In case when distributed systems are cooperating to achieve a common objective, like flocking problem of UAV's or a power network, the matrices $f_x(k)$ and $f_u(k)$ in (2.11) are large but sparse. If any system has limited and well defined connections with neighboring systems, then it is expected that a closed form NE solution for a network of systems can be derived. With this approach MPC implementation on networked systems becomes possible.

6.2.2 Maximal invariant sets for systems with constrained rate-bounded disturbances

In this thesis, minimal invariant attracting set was characterized and a computation method was provided for the set to be approximated. Another important set is the maximal invariant set which characterizes the region of attraction of a stabilizing linear controller and is used in robust constrained control methods such as the reference governor. Considering a system as

$$x(k+1) \in f(x(k)),\ x \in \mathbb{X} \subset \mathbb{R}^n$$

where f is a correspondence, traditionally the maximal invariant set is derived by

$$\bigcap_{i=0}^{\infty} \mathcal{R}^i(\mathbb{X})$$

where $\mathcal{R}(\Omega) := f^s(\Omega)$ and $\mathcal{R}^{i+1}(\Omega) = \mathcal{R}(\mathcal{R}^i(\Omega))$. Confining to the case where $f(x)$ is polytop for all x and $\mathbb{X}$ is polytop and f is defined by (5.48), $\mathcal{R}^i(\mathbb{X})$ is not convex but union of polytops. This fact shows the need for future research for have a new approach, probably different than traditional one, to calculate the maximal invariant set which is highly important in robust control of linear systems.

6.2.3 MPC with time varying disturbance

In many applications such as ship control in a wave field, the wave response is predictable for short future horizon, but it is time varying. In this case, robust control theory even with negligence towards practicality is not well developed and needs extensive research. Ship control in a wave field subject to constraints is a specific application of MPC with time varying disturbances. A time varying robust control strategy that satisfies roll angle constraint and follows a pre-specified path is subject of future research.

6.2.4 Distributed MPC in Power Networks

MPC can be employed in a network of systems to achieve a global objective subject to constraints. Specially in power networks, decentralized robust MPC can lead to an increase in power system transition capacity, achieve energy conservation, and enhance the stability of power networks, which is crucial for advanced power networks (conceptualized as smart grids). The broader impact of the research is on decentralized cooperative systems which has broader application beyond power systems, such as formation flight control and cooperative UAV's.

The developed computationally efficient MPC method can be employed for efficient control of constrained systems with fast dynamics such as power electronic converters, which are essential parts of power networks. However, in a power network, local controllers should be designed to operate such that imperfect information or disturbances imposed by network do not affect performance of the controller at the local level, i.e, the controller must be robust. Even though there exists a body of previous work on robust MPC, the existing robust MPC methods are mostly confined to a special class of systems or they are computationally intensive. In this regard, the future research plan is to modify the existing fast MPC to a robust version to achieve the desired performance in the presence of disturbances inflicted through the power network. High performance local controllers in the power network contribute in the stability of the network. For example, efficient control of flexible AC transmission devices (FACTS), leads to increase inpower system transmission capability and enhances the stability of weakly coupled systems in the event of critical faults.

On the other hand, local controllers in a power network must act cooperatively using the information that is provided by other neighboring controllers to achieve global objectives and overall stability of the power network. In this regard, the

plan for future research is to design a mechanism that provides an objective for each controller at the local level (that is reflected in the cost function of the MPC controller) so that a system-wide objective, such as large scale optimal energy consumption and power network stability, is achieved.

Appendices

Appendix A

Degeneracy

In this Appendix, the case where $\hat{C}_x(0)$ is non-empty is considered. In this case two cases may happen. If $\hat{C}_x(0)\delta x(0) \neq 0$ then the problem is infeasible. If $\hat{C}_x(0)\delta x(0) = 0$, it means that linear constraints (2.11)-(2.14)are dependent. In this case, the problem is degenerate and instead of solving the original QP problem, we now shift our attention to finding a feasible descent direction.

In order to enforce the original nonlinear constraints, we modify the linear equality constraints

$$\begin{aligned} &C_x^a(k)\delta x(k) + C_u^a(k)\delta u(k) = -C^a(x(k), u(k)), \\ &\bar{C}_x^a(k)\delta x(k) = -\bar{C}^a(x(k)), \end{aligned}$$

into the following linear inequality constraints:

$$\begin{aligned} &C_x^a(k)\delta x(k) + C_u^a(k)\delta u(k) \leq -C^a(x(k), u(k)), \\ &\bar{C}_x^a(k)\delta x(k) \leq -\bar{C}^a(x(k)). \end{aligned}$$

We now introduce the following linear programming (LP) problem:

$$\min_{\delta u(\cdot), \delta x(\cdot)} \sum_{k=0}^{N-1} (L_x(k)\delta x(k) + L_u(k)\delta u(k)) + \Phi_x(N)\delta x(N)$$

subject to:

$$\begin{aligned} &\delta x(k+1) = f_x(k)\delta x(k) + f_u(k)\delta u(k), \\ &\delta x(0) = \delta x_0, \\ &C_x(k)\delta x(k) + C_u(k)\delta u(k) \leq -C(x(k), u(k)), \\ &\bar{C}_x(k)\delta x(k) \leq -\bar{C}(x(k)). \end{aligned} \tag{A.1}$$

to find the next feasible direction. In solving the LP problem (A.1) we achieve: (i)

the value of the active constraints will not be increased, but can be decreased which will lead to a reduced number of active constraints for the next iteration; (ii) the original cost function will be decreased whenever possible.

Note that solving the LP problem (A.1) can be performed in a time efficient manner so that it will not be a barrier for fast optimization. In addition, the LP problem is solved only when the QP problem is degenerate or infeasible, identified by the condition $\hat{C}_x(0)$ being non-empty.

Appendix B

Calculating Lagrange multipliers for NE method

Assuming the nominal state and control sequences are available, we need to calculate the Lagrange multipliers $\mu(\cdot)$ and $\lambda(\cdot)$ in order to be able to determine the Hessian matrices in (2.25), so that the neighboring extremal algorithm described above can be applied. In general, numerical algorithms for computing an optimal solution u^0 and x^0, also yield a satisfactory approximation for Lagrange multipliers Λ, μ, and $\bar{\mu}$. An alternative procedure, to calculate Λ, μ, and $\bar{\mu}$ on-line, which avoids to store these values, is given below.

Let us assume that the optimal control and state vector sequences are, respectively, $u^o(k),\ k = 1, \cdots, N-1$, and $x^o(k),\ k = 0, \cdots, N$. From (2.8), we have

$$\begin{aligned} \lambda(k) &= H_x(k)^T, \\ H_u(k) &= 0, \end{aligned}$$

which can be written, using (2.3) and (2.7), as

$$\begin{aligned} \lambda(k) = L_x(x^o(k), u^o(k))^T + f_x(k)^T\lambda(k+1) + C_x(x^o(k), u^o(k))^T\mu(k) + \bar{C}_x(k)^T\bar{\mu}(k), \\ L_u(x^o(k), u^o(k))^T + f_u(k)^T\lambda(k+1) + C_u(k)^T\mu(k) = 0 \end{aligned} \tag{B.1}$$

for $k = 1, \cdots, N-1$. As in Appendix II, the superscript a has been dropped for notation simplicity, assuming that the constraints appearing in the equations are active. In addition, we have

$$\lambda(N) = \Phi_x(N)^T + \bar{C}_x(N)^T\bar{\mu}(N). \tag{B.2}$$

Let us define $\hat{\mu}(N) := \bar{\mu}(N)$, $D(N) := \Phi_x(N)^T$, and $\hat{C}_x(N) := \bar{C}_x(N)$. From (B.2),

we have

$$\lambda(N) = D(N) + \hat{C}_x(N)^T \hat{\mu}(N). \tag{B.3}$$

Now assume

$$\lambda(k+1) = D(k+1) + \hat{C}_x(k+1)^T \hat{\mu}(k+1).$$

Using $\tilde{C}_u(k)$, $\hat{C}_x(k)$, and $\tilde{C}_x(k)$ as defined in (2.22) and (2.24) and applying the transformation

$$\begin{bmatrix} \tilde{\mu}(k) \\ \hat{\mu}(k) \end{bmatrix} := \begin{bmatrix} P(k)^{-T} \begin{bmatrix} \mu(k) \\ \hat{\mu}(k+1) \end{bmatrix} \\ \bar{\mu}(k) \end{bmatrix} \tag{B.4}$$

to (B.1), we obtain

$$\begin{aligned} &\lambda(k) = L_x(x^o(k), u^o(k))^T + f_x(k)^T D(k+1) + \tilde{C}_x(k)^T \tilde{\mu}(k) + \hat{C}_x(k)^T \hat{\mu}(k), \\ &L_u(x^o(k), u^o(k))^T + f_u(k)^T D(k+1) + \tilde{C}_u(k)^T \tilde{\mu}(k) = 0. \end{aligned} \tag{B.5}$$

Since $\tilde{C}_u(k)$ is of full row rank, we have

$$\tilde{\mu}(k) = -(\tilde{C}_u(k)\tilde{C}_u(k)^T)^{-1}\{L_u(k)^T + f_u(k)^T D(k+1)\}. \tag{B.6}$$

If we define

$$D(k) := L_x(k)^T + f_x(k)^T D(k+1) + \tilde{C}_x(k)^T \tilde{\mu}(k), \tag{B.7}$$

then

$$\lambda(k) = D(k) + \hat{C}_x(k)^T \hat{\mu}(k). \tag{B.8}$$

Now the algorithm for calculating Lagrange multipliers can be summarized as follows:

- In a backward run, calculate $D(k)$, with $D(N) = \Phi_x(N)^T$, using (B.7), and $\tilde{\mu}(k)$, using (B.6).
- If $\hat{C}_x(0)$ is empty, set $\hat{\mu}(0) =$ empty matrix. Now with $\tilde{\mu}(\cdot)$ and $P(\cdot)$ being avoidable, one can calculate $\mu(k)$, $\hat{\mu}(k+1)$, and $\bar{\mu}(k)$ in a forward run, using transformation (B.4) which is rearranged as follows[B1]:

$$\begin{bmatrix} \mu(k) \\ \hat{\mu}(k+1) \\ \bar{\mu}(k) \end{bmatrix} = \begin{bmatrix} P(k)^T & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \tilde{\mu}(k) \\ \hat{\mu}(k) \end{bmatrix},$$

[B1] Note that number of rows of $P(k)$ is not necessarily equal to that of $\tilde{\mu}(k)$.

and $\lambda(k)$ using (B.8).

Bibliography

[1] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert, "Constrained model predictive control: stability and optimality," *Automatica*, Vol. 36, pp789-814, 2000.

[2] E. Gilbert, I. V. Kolmanovsky, and K. T. Tan, "Discrete time reference governors and the nonlinear control of systems with state and control constraints," *International Journal of Robust and Nonlinear Control,* vol. 5, pp. 487-504, 1995.

[3] T. Binder, L. Blank, H. G. Bock, R. Bulirsch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J. P. Schlöder, O. von Stryk,"Introduction to model based optimization of chemical processes on moving horizons," In M. Grötschel, S. O. Krumke, J. Rambau(editors):, *Online Optimization of Large Scale Systems,* Springer-Verlag, Berlin, 2001.

[4] N. L. Ricker, "Use of quadratic programming for constrained internal model control," *Ind. Eng. Chem. Process Design and Development,*, Vol. 24, pp. 925-936, 1985.

[5] J. Sun, I.V. Kolmanovsky, R. Ghaemi, S.H. Chen, "A stable block model predictive control with variable implementation horizon," *Automatica*, Vol. 43, No. 11, pp. 1945-1953, 2007.

[6] S.S. Keerthi and E. Gilbert, "Optimal, infinite-horizon feedback laws for a general class of constrained discrete-time systems, *Journal of Optimization Theory and Applications*, vol. b57, pp. 265-293, 1988.

[7] Goodwin, G., M. Seron and J. DeDona, *Constrained Control and Estimation: An Optimisation Approach,* Springer, 2004.

[8] P. Dufour, Y. Toure, D. Blanc, P. Laurent, "On nonlinear distributed parameter model predictive control strategy: on-line calculation time reduction and application to an experimental drying process," *Computers and Chemical Engineering*, 27, pp. 1533-1542, 2003.

[9] M. Hovd, J. H. Lee and M. Morari, "Truncated step response models for model predictive control," *Journal of Process Control*, Vol. 3 (2), pp. 67-73, 1993.

[10] W. Marquardt, "Nonlinear model reduction for optimization based control of transient chemical processes," *Proceedings of the CPC VI*, Tucson, Arizona, pp. 30-60, 2001.

[11] J. Kang, I. Kolmanovsky and J. Grizzle, "Approximate dynamic programming solutions for lean burn engine aftertreatment," *Proceeding of the IEEE Conference on Decision and Control*, Phoenix, AZ, December 7-10, 1999.

[12] M. Hovd, J. H. Lee and M. Morari, "Truncated Step Response Models for Model predictive Control," *Journal of Process Control*, 3 (2), pp. 67-73, 1993.

[13] Marquardt, W, "Nonlinear model reduction for optimization based control of transient chemical processes," *Proceedings of the CPC VI*, Tucson, Arizona, pp. 30-60, 2001.

[14] C. E. Garcia, "Quadratic dynamic matrix control of nonlinear processes. An application to a batch reactor process," *Proceeding of AIChE Annual Meeting*, San Francisco, CA, 1984.

[15] G. De Nicolao, L. Magni, and R. Scattolini, "On the robustness of receding horizon control with terminal constraints," *IEEE Transactions on Automatic Control*, Vol. 41, pp. 451-453, 1996.

[16] A. Zheng, "A computationally efficient nonlinear MPC algorithm," *Proceedings of the American Control Conference Albuquerque*, New Mexico June 1997.

[17] Bemporad, A., M. Morari, V. Dua, E.N. Pistikopoulos, "The Explicit linear quadratic regulator for constrained systems," *Automatica*, 38, pp. 3-20, 2002.

[18] T. A. Johnson, "Approximate explicit receding horizon control of constrained nonlinear systems," *Automatica,* 40, pp. 293-300, 2004.

[19] T.A. Johansen, I. Petersen, O. Slupphaug, "Explicit sub-optimal linear quadratic regulation with state and input constraints," *Automatica*, Vol. 38, Issue 7, pp. 1099-1111, July 2002.

[20] Bemporad, A., A. Garulli, S. Paoletti, and A. Vicino, "A bounded-error approach to piecewise affine system identification," *IEEE Transactions on Automatic Control,* Vol. 50, No. 10,pp. 1567-1580, October 2005.

[21] PWAID - Piecewise affine identification toolbox for Matlab, http://www.control.isy.liu.se/ roll/PWAID/.

[22] M. Lazar, W.P.M.H. Heemels, S. Weiland and A. Bemporad, "Stabilization conditions for Model Predictive Control of constrained PWA systems," *Proceedings of the 43rd IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, pp. 4595-4600, 2004.

[23] E. G. Gilbert and I. V. Kolmanovsky, "Nonlinear tracking control in the presence of state and control constraints: a generalized reference governor," *Automatica,* vol. 38, no. 12, pp. 2063-2073, 2002.

[24] A. Bemporad, "Reference governor for constrained nonlinear systems," *IEEE Transactions on Automatic Control,* vol. 43, no. 3, pp. 415-419, 1998.

[25] I. V. Kolmanovsky and J. Sun, "Parameter governors for discrete-time nonlinear systems with pointwise-in-time state and control constraints," *Automatica*, Vol. 42, No. 5, pp. 841-848, 2006.

[26] W.B. Dunbar and R.M. Murray, "Distributed receding horizon control with application to multi-vehicle formation stabilization," *Automatica*, Vol. 42, pp. 549-558, 2006.

[27] T. Keviczky, F. Borrelli and G.J. Balas, "Decentralized receding horizon control for large scale dynamically decoupled systems," *Automatica,* Vol. 42, No. 12, pp. 2105-2115, 2006.

[28] D.M. Raimondo, L. Magni and R. Scattolini, "Decentralized MPC of nonlinear systems: An input-to-state stability approach," *International Journal of Robust and Nonlinear Control,* Vol. 17, pp. 1651-1667, 2007.

[29] Venkat, A.N., I.A. Haskins, J.B. Rawlings and S.J. Wright, "Distributed output feedback MPC for power system control," *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 4038-4045, 2006.

[30] Scattolini, R. and P. Colaneri, "Hierararchical Model Predictive Control," *Proceedings of IEEE Conference o Decision and Control*, New Orleans, LA, USA, Dec. 12-14, 2007, pp. 4803-4808.

[31] L.G. Bleris, P.D. Vouzis, M.G. Arnold and M.V. Kothare, "A co-processor FPGA platform for the implementation of real-time model predictive control," *Proceedings of 2006 American Control Conference*, Minneapolis, MN, USA, pp. 1912-1917, 2006.

[32] T.R. Johansen, W. Jackson, R. Schrieber and P. Tondel, "Hardware synthesis of explicit Model Predictive Controllers," *IEEE Transactions on Control Systems Technology*, vol. 15, No. 1, pp. 191-197, 2007.

[33] Wright SJ. Primaldual Interior-point Methods. SIAM: Philadelphia, PA, 1997. 8.

[34] Rao CV, Wright SJ, Rawlings JB. Application of interior-point methods to model predictive control. Journal of Optimization Theory and Applications 1998; 99:723-757.

[35] M. B. Milam, R. Franz, J. E. Hauser, R. Murray,"Receding horizon control of a vectored thrust flight experiment," IEE Proc.-Control Theory Appl., Vol. 152, No. 3, May 2005.

[36] B. Gholami, B. W. Gordon, C. A. Rabbath, "Uncertain nonlinear receding horizon control systems subject to non-zero computation time," *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Sevelle, Spain, December 12-15, 2005.

[37] H.J. Ferreau, H.G. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit MPC. International Journal of Robust and Nonlinear Control, 2007.

[38] J. V. Breakwell, J. L. Speyer, and A. E. Bryson, "Optimization and control of nonlinear systems using the second variation," *SIAM Journal on Control*, Vol. 1, pp. 193-223, 1963.

[39] J. V. Breakwell and Y. C. Ho,"On the conjugate point condition for the control problem," *International Journal of Engineering and Science*, Vol. 2, pp. 565-579, 1965.

[40] B. Kugelmann, H. J. Pesch, "A new general guidance method in constrained optimal control, Part 1: Numerical method," *Journal of Optimization Theory and Applications*, Vol. 67, pp. 421-435, 1990.

[41] K. Malanowski, H. Maurer, "Sensitivity analysis for parametric control problems with control-state constraints," *Computational Optimization and Applications*, Vol 5, pp. 253-283, 1996.

[42] H. Maurer, H. J. Pesch, "Solution differentiability for parametric nonlinear control problems with control-state constraints," *Journal of Optimization Theory and Applications*, Vol. 86, pp. 285-309, 1995.

[43] S. R. McReynolds and A. E. Bryson,"A successive sweep method for optimal programming problems," Auto control conference, 1965.

[44] F. L. Lewis, *Optimal Control,* Wiley, New York, 1986.

[45] A. E. Bryson and Y. Ho, *Applied Optimal Control,* Hemisphere Publishing Corp, Washington D. C., 1975.

[46] C. Büskens, H. Maurer, "Sensitivity analysis and real-time control of parametric optimal control problems using nonlinear programming method, In M. Grötschel, S. O. Krumke, J. Rambau(editors):, *Online Optimization of Large Scale Systems,* Springer-Verlag, Berlin, 2001.

[47] P. O. M. Scokaert and D. Q. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Transactions on Automatic Control*, vol. 43, pp. 1136-1142, 1998.

[48] E. G. Gilbert and I. Kolmanovsky, "Fast reference governors for systems withstate and control constraints and disturbance inputs," Int. J. Robust and Nonlinear Control, vol. 9, no. 15, pp. 1117-1141, Dec. 1999.

[49] D. L. Marruedo, T. Alamo, E. F. Camacho, "Stability analysis of systems with bounded additive uncertainties based on invariant sets: Stability and feasibility of MPC," *Proceedings of American Control Conference*, pp. 364-369, Alaska 2002.

[50] H. Michaska and D. Q. Mayne, "Robust receding horizon control of constrained nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 38, No. 11, pp. 1623-1633, November, 1993.

[51] D. Q. Mayne and W. Langson, "Robustifying model predictive control of constrained linear systems," *Electronic Letters*, vol. 37, pp. 1422-1423, November, 2001.

[52] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: Predictive control with restrictive constraints," *Automatica*, vol. 37, pp. 1019-1028, 2001.

[53] D. Q. Mayne, M. M. Seron, and S. V. Rakovic, "Robust Model Predictive Control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41(2), pp. 1136-1142, 2005.

[54] A. Richards and J. How, "Robust stable model predictive control with constraint tightening," *Proceedings of the IEEE American Control Conference*, 2006.

[55] Y. I. Lee, B. Kouvaritakis, "Robust receding control for systems with uncertain dynamics and input saturation," *Autimatica*, vol. 36, pp. 1497-1504, 2000.

[56] W. Langson, I. Chryssochoos, S. V. Rakovic and D. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, pp.125-133, 2004.

[57] D. Q. Mayne, S. V. Rakovic, R. Findeisen, and F. Algöwer, "Robust output feedback model predictive control of constrained linear systems" *Automatica*, vol. 42, pp. 1217-1222, 2006.

[58] X.C. Ding, A. Schild, M. Egerstedt, and J. Lunze, "Real-Time Optimal feedback Control of Switched Autonomous Systems," *IFAC Conference on Analysis and Design of Hybrid Systems*, Zaragoza, Spain, Sept. 2009.

[59] R. Ghaemi, J. Sun, and I. Kolmanovsky, "An Integrated Perturbation Analysis and Sequential Quadratic Programming Approach for Model Predictive Control," *Automatica*, Vol. 45, pp. 2412-2418, 2009.

[60] A. Y. Lee, "Neighboring extremals of dynamic optimization problems with path equality constraints", *Journal of Optimization Theory and Applications*, Vol. 57, pp. 519-536, 1988.

[61] R. Ghaemi, J. Sun, and I. Kolmanovsky, "Neighboring Extremal Solution for Nonlinear Discrete-Time Optimal Control Problems with State Inequality Constraints," *IEEE Transaction on Automatic Control*, Vol. 54, No. 11, 2009.

[62] N. Haverbeke, M. Diehl, H. J. Ferreau, "Efficient numericalmethods for nonlinear mpc and moving horizon estimation". in Chapter 32 of Nonlinear Model Predictive Control, (Magni L., Raimondo M.D., and Allgower F., eds.), Lecture Notes in Control and Information Sciences, 384:391, 2009.

[63] R. W. De Doneker, D. M. Divan, and M. H. Kheraluwala, "A three-phase soft-switched high-power-density dc/dc converter for high-power applications," IEEE Trans. Industry Applications, vol. 27, pp.797-806, Jan./Feb. 1991.

[64] M. H. Kheraluwala, R. W. Gascoigne, D. M. Divan, and E. D. Baumann, "Performance characterization of a high-power dual active bridge dc-to-dc converter," IEEE Trans. Industry Applications, vol. 28, pp.1294-1301, Nov./Dec. 1992.

[65] D. Fu, F. C. Lee, Y. Qiu, F. Wang, "Novel high-power-density three-level LCC resonant converter with constant-power-factorcontrol for charging applications," IEEE Trans. Power Electron., pp. 2411-2420, 2008.

[66] D. Fu, Y. Liu, F. C. Lee, M. Xu, Novel driving scheme for synchronous rectifiers in LLC resonant converters, IEEE Trans. Power Electron., pp. 1321-1329, 2009.

[67] Y. Xie, G. Seenumani, J. Sun, Y. Liu, and Z. Li, "A PC-cluster based real-time simulator for all electric ship integrated power systems analysis and optimization," in Proc. IEEE Electric Ship Technology Symposium (ESTS), 2007, pp.396-401.

[68] Y. Xie, J. Sun, C. Mi, and J. S. Freudenberg, "Analysis and modeling of a DC hybrid power system testbed for power management strategy development," in Proc. IEEE Vehicle Power and Propulsion Conf. (VPPC'09), 2009.

[69] Y. Xie, R. Ghaemi, J. Sun, , and J. S. Freudenberg, "Implicit Model Predictive Control of A Full Bridge DC/DC Converter", *IEEE Transactions on Power Electronics,* vol. 24, pp. 2704 - 2713, Dec. 2009.

[70] R. Ghaemi, S. Oh, J. Sun, "Path following of a model ship using model predictive control with experimental verification," American Control Conference, 2010

[71] T. Perez and G. C. Goodwin, "Constrained predictive control of ship fin stabilizers to prevent dynamic stall," *Control Engineering Practice*, vol. 16, pp. 482-494, 2008.

[72] T. Perez, "Ship motion control: Course keeping and roll reduction using rudder and fins," *Advances in industrial control*. London:Springer, 2005.

[73] I. T. Fossen, "Marine control systems:Guidance, navigation and control of ships, rigs and underwater vehicles," *Trondheim: Marine Cybernetics*, 2002.

[74] F. H. Sellars and J. P. Martin, "Selection and evaluation of ship roll stabilization systems," *Marine Technology, SNAME* Vol. 29(2), pp. 84-101, 1992.

[75] A. R. J. M. Lloyd, Seakeeping: *ship behaviour in rough weather*, A.R.J.M, 1988.

[76] R. Ghaemi, J. Sun, I. Kolmanovsky, "Robust Control of Ship Fin Stabilizers Subject to Disturbances and Constraints," American Control Conference, St. Louis, Missouri, 2009.

[77] R. Ghaemi, J. Sun, I. Kolmanovsky, "Less Conservative Robust Control of Constrained Linear Systems with Bounded Disturbances," Proceedings of the IEEE Conference on Decision and Control, Cancun, Mexico, 2008.

[78] I. Kolmanovsky and E. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical Problems in Engineering: Theory, Methods and Applications*, vol. 4, pp. 317-367, 1998.

[79] S. V. Raković and Mirko Fiacchini. "Invariant Approximations of the Maximal Invariant Set or "Encircling the Square"". In Proceedings of the 17th IFAC World Congress IFAC 2008, Seoul, South Korea, July 2008.

[80] Z. Artstein and S. Rakovic, Feedback and invariance under uncertainty via set-iterates, Automatica, vol. 44, pp. 520525, 2007.

[81] I. Komanovsky, E. G. Gilbert, and N. H. McClamroch, "Towards less conservative design of multimode controllers for systems with state and control constraints," *Proceeding of the Conference on Decision and Control*, pp. 1647-1652, Tampa, 1998.

[82] Y. Liu, S. Ito, K. L. Teo, "A semi-infinite programming approach to continuously constrained LQ optimal control problems". *38th Annual Conference Proceedings of the SICE, 1149 - 1154, 1999.*

[83] A. V. Fiacco, "Introduction to sensitivity and stability analysis in nonlinear programming," *Mathematics in Science and Engineering*, Academic press, New York, 1983.

[84] R. Redl and I. Novak, "Instabilities in current-mode controlled switching voltage regulators," in *Proc. IEEE Power Electronics Specialists Conf. (PESC),* 1981, pp. 17-28.

[85] S. Kouro, P. Cortes, R. Vargas, U. Ammann, and J. Rodriguez, "Model predictive control-a simple and powerful method to control power converters," in *IEEE Trans. Industrial Electronics,* vol. 56, pp. 1826-1838, Jun. 2009.

[86] T. Geyer, G. Papafotiou, and M. Morari, "Model predictive direct torque control-part I: concept, algorithm, and analysis," in *IEEE Trans. Industrial Electronics,* vol. 56, pp. 1894-1905, Jun. 2009.

[87] T. Geyer, G. Papafotiou, and M. Morari, "Model predictive direct torque control-part II: implementation and experimental evaluation," in *IEEE Trans. Industrial Electronics,* vol. 56, pp.1906-1915, Jun. 2009.

[88] E. I. Silva, B. P. McGrath, D. E. Quevedo, and G. C. Goodwin, "Predictive control of a flying capacitor converter," in *Proc. American Control Conference*, 2007, pp.3763-3768.

[89] J. Sun and H. Grotstollen, "Averaged modelling of switching power converters: reformulation and theoretical basis," in *IEEE Appl. Power Electron. Conf. Expo (APEC),* 1992, pp. 1165-1172.

[90] J. Sun, D. M. Mitchell, M. F. Greuel, P. T. Krein, and R. M. Bass, "Averaged modeling of PWM converters operating in discontinuous conduction mode," *IEEE Transactions on Power Electronics,* vol. 16, pp. 482-492, Jul. 2001.

[91] W. S. Levine, *The control handbook,* CRC Press and IEEE Press, 1996.

[92] A. Davoudi, J. Jatskevich, and T. De Rybel, "Numerical state-space average-value modeling of PWM DC-DC converters operating in DCM and CCM," *IEEE Transactions on Power Electronics,* vol. 21, pp. 1003-1012, Jul. 2006.

[93] K. Do and J. Pan, "Underactuated ships follow smooth paths with integral actions and without velocity measurements for feedback: theory and experiments," *IEEE transactions on Control Systems Technology*, vol. 14, no. 2, pp. 308-322, 2006.

[94] R. Skjetne, T. I. Fossen, and P. Kokotovic, "Robust output maneuvering for a class of nonlinear systems," *Automatica*, vol. 40, pp. 373-383, 2004.

[95] A. Wahl and E. Gilles, "Track-keeping on waterways using model predictive control," *Proceedings of Control Applications in Marine Systems*, 1998.

[96] T. Perez and G. C. Goodwin, "Constrained predictive control of ship fin stabilizers to prevent dynamic stall," *Control Engineering Practice*, vol. 16, pp. 482-494, 2008.

[97] F. S. Hoerner, V. H. Borst, *"Fluid-dynamic lift"* (2nd ed.). Hoerner Fluid Dynamics, 1985.

[98] Zhen Li, Jing Sun and Soryeok Oh, "Path Following for Marine Surface Vessels with Rudder and Roll Constraints: an MPC Approach", *Proceedings of American Control Conference*, 2009.

[99] S. Oh, J. Sun, Z. Li, E. A. Celkis, and D. Parsons, "System Identification of a Model Ship Using a Mechatronic System",

[100] S. S. L. Chang and T. K. C. Peng, "Adaptive guaranteed cost control of systems with uncertain parameters," *IEEE Trans. Autom, Control*, vol. 39, no. 4, pp. 474-483, 1972.

[101] M. S. Mahmoud and L. Xie, "Guaranteed cost control of uncertain discrete systems with delay," *Int. J. Control,* vol. 73, pp. 105-114, 2000.

[102] M. V. Kothare, V. Balakishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica,* vol. 32, pp. 1361-1379, 1996.

[103] W. H. Kwon, J. W. Kang, Y. S. Lee, and T. S. Moon,"A simple receding horizon control for state delayed systems and its stability criterion," *J. Process Control,* vol. 13, pp. 539-551, 2003.

[104] S. C. Jeong and P. Park, "Constrained MPC Algorithm for Uncertain Time-Varying Systems with State-Delay," *IEEE Trans. Autom, Control*, vol. 50, no. 2, pp. 257-263, 2005.

[105] J. M. Maciejowski, *Predictive Control with Constraints*, Pearson Education, 2002.

[106] F. Allgower, A. Zheng (Eds.), Nonlinear Model Predictive Control,, Springer, 2000.

[107] R. Ghaemi, J. Sun, I. Kolmanovsky, "Computationally efficient model predictive control with explicit disturbance mitigation and constraint enforcement," *44th IEEE Conference on Decision and Control*, San Diego, pp. 4842-4847, December 2006.

[108] T. Glad and H. Johnson,"A method for state and control constrained linear quadratic control problems," *Proceedings of the 9th IFAC World Congress*, Budapest, Hungary, pp. 1583-1587, 1984.

[109] R. Flecher,"Practical methods of Optimization", Vol. 2: Constrained Optimization, Wiley, Chichester, England, 1981.

[110] J. W. Grizzle, Lecture notes on Nonlinear Control, University of Michigan, winter 2005.

[111] M. H. Casado, A. Fernandez, and J. Iglesis,"Optimization of the course in the ship's movement by input-output linearization", *Proceedings of IFAC Conference on Control Application and Marine Systems*, pp. 419-424, 2001.

[112] S. V. Rakovic, "Minkowski Algebra and Banach Contraction Principle in Set Invariance for Linear Discrete Time Systems," *Proceeding of the 46th IEEE Conference on Decision and Control*, pp. 2169-2174, 2007.

[113] E. Klein, A. C. Thompson, "Theory of Correspondences," *Wiley-Interscience*, New York, 1984.

[114] F. Blanchini, S. Miani, *"Set-Theoretic Methods in Control,"* Birkhauser, 2008.

[115] C. D. Aliprantis and K. C. Border, "Infinite Dimentional Analysis" *Springer*, 2005.

[116] J. L. Kelley, "General Topology" *Springer*, 1975.

[117] S. Keerthi and E. Gilbert, "Computation of minimumtime feedback control laws for discrete-time systems with state-control constraints," IEEE Transactions on Automatic Control, Vol. 32, No. 5, pp. 432-435, 1987.