CLOSED-LOOP DYNAMIC REAL-TIME OPTIMIZATION WITH STABILIZING MODEL PREDICTIVE CONTROL

CLOSED-LOOP DYNAMIC REAL-TIME OPTIMIZATION WITH STABILIZING MODEL PREDICTIVE CONTROL

by

Praveen Sundaresan Ramesh, B.Tech

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree
Master of Applied Science

McMaster University

MASTER OF APPLIED SCIENCE (2020) McMaster University

(Chemical Engineering) Hamilton, Ontario, Canada

TITLE:	Closed-loop Dynamic Real-Time Optimization with Stabilizing Model Predictive Control
AUTHOR:	Praveen Sundaresan Ramesh, B.Tech (McMaster University, Hamilton, ON)
SUPERVISOR:	Dr. Christopher L.E. Swartz and Dr. Prashant Mhaskar
NUMBER OF PAGES:	ix, 70

ABSTRACT

Real-time optimization (RTO) is a supervisory strategy at the upper level of the industrial process automation architecture that computes optimal set-point trajectories that are in turn passed on to the lower-level advanced controller for tracking. In recent times, process industries have been operating plants in an increasingly dynamic environment thus motivating the replacement of the traditional steady-state RTO with a dynamic RTO (DRTO). Conventionally, the DRTO problem involved computing optimal input trajectories without incorporating the effects of the lower-level controller that motivated the development of a closed-loop DRTO (CL-DRTO).

A previously published CL-DRTO strategy optimizes set-point trajectories based on an economic cost function, while including the lower-level MPC calculations with an assumption that the process is open-loop stable. But the economic optimum, in certain process industries, could lead to operating the plant at or around an unstable steady-state. Therefore, the goal of this research is to develop a CL-DRTO formulation that enables handling unstable operating points via an underlying MPC with stability constraints. In this work, we focus on a traditional two-layer DRTO approach due to its close ties to the industrial automation architecture.

To this end, a stabilizing MPC that handles trajectory tracking for unstable systems is embedded within the upper-level DRTO, resulting in a multi-level dynamic optimization problem. Subsequently, this optimization problem is reformulated by applying a simultaneous solution approach. The economic benefits realized by the proposed strategy are illustrated through applications to both linearized and nonlinear dynamic models for single-input single-output (SISO) and multi-input multi-output (MIMO) CSTR case studies.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my supervisors, Dr. Christopher L.E. Swartz and Dr. Prashant Mhaskar, for their consistent guidance and mentoring throughout my graduate studies at McMaster University. Their dedicated support, both academic and financial, and the thought provoking discussions helped me push my boundaries for the past two years. I would like to acknowledge McMaster Advanced Control Consortium (MACC) for providing me with this opportunity and resources.

I would also like to acknowledge my loving parents and brother for their tremendous support and encouragement. I would like to thank my friends for their constant support and companionship while also being good adventurers to travel with. A special mention to my fellow MACC colleagues for the hours long silly debates that made the past two years very memorable.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we present the motivation for our research along with an outline of the thesis, particularly, focused on the industrial automation architecture owing to its significance to our research.

1.1 Motivation

In process industries, automation has been used as a key strategy to make economic impacting decisions based on market conditions. Process automation engineers are therefore consistently developing novel automation strategies to address the challenges of rising costs, tightening environmental policies and a need for efficient operation. For processes that operate on a large scale like oil refineries, power generation plants and several others, a large number of these decisions have to be made online. One of the traditional approaches for online decision making used in process chemical industries is a two-layer hierarchical architecture. A comprehensive review of the real-time optimization (RTO) strategy and the two-layer architecture in process industries is given in Darby *et al.* [2011].

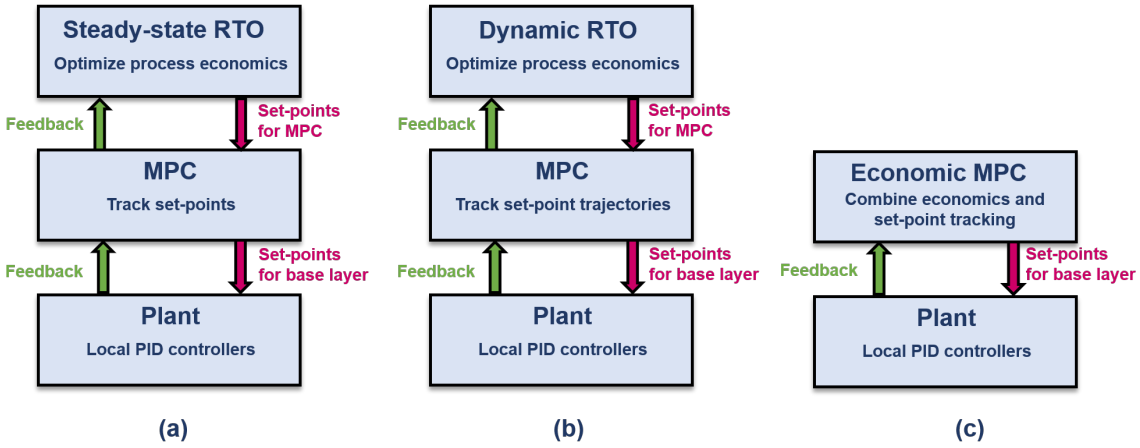

Figure 1.1: Common process automation architectures

Fig 1.1a shows the two-layer process automation architecture associated with the traditional RTO implementation. At the upper level of this architecture is the RTO strategy that optimizes an economic objective function with output set-points as the decision variables, which could in turn be realized by the classical proportional integral derivative (PID) controllers that could be installed alongside the sensors and transmitters of the plant. Owing to its simple structure, the PID control strategy is implemented in the majority of process industries as the primary control system. Despite its widespread use in the industries, PID controllers have several limitations including the inability to directly handle process constraints and process interactions.

For these reasons, an advanced controller, typically model predictive control (MPC), that can readily handle input constraints, state constraints and process interactions (García *et al.* [1989]) is increasingly being utilized in a cascade fashion at the lower level. Typically, the MPC optimization uses a linear dynamic model to predict the output plant response, implemented in a rolling horizon fashion. Centralized MPC, de-centralized MPC and distributed MPC are three ways of implementing the MPC strategy in process industries depending on their operational requirements. A brief review of successful industrial applications of various MPC strategies is given in Qin and Badgwell [2003].

Real-time optimization (RTO) is a supervisory strategy at the upper level of the industrial process automation architecture that computes economically optimal set-point trajectories that are in turn passed on to the lower-level model predictive controller (MPC) for tracking. The set-points provided by the upper level (RTO) to the MPC are used to compute optimal input moves, which are then passed on to the local PID controllers in the plant. In the two-layer architecture, the RTO strategy typically comprises a high fidelity nonlinear model, while the lower-level advanced controller is based on a linear dynamic model. It is important to note that the time scale of the RTO computation is usually a few hours while that of the underlying MPC is a few minutes.

Traditionally, a steady state model of the plant is used at the RTO level. The execution of these steady state optimization calculations is therefore inhibited by the requirement for the system to reach steady state conditions. For processes that have frequent transitions or exhibit slower dynamics, using a steady state RTO could result in a sub-optimal performance. This limitation motivated the development of a dynamic RTO (DRTO) wherein a dynamic model of the plant is used at the RTO level. The first implementations of the DRTO strategy (Kadam *et al.* [2002], Tosukhowong *et al.* [2004]) were based on an open-loop framework wherein, the optimization computations are performed independent of the underlying control system. This might result in sub-optimal performance depending on the tuning of the lower level controller. Recently in Jamaludin and Swartz [2017b], a closed-loop DRTO was proposed wherein the effects of the underlying controller are accounted for in the DRTO formulation.

In Jamaludin and Swartz [2017b,a], an input-constrained quadratic dynamic matrix control (QDMC) based on a linear state-space model is used as the lower level advanced controller. The limitations of a DMC algorithm is given in Lundström *et al.* [1995]. Only stable plants can be described by the internal state-space model of the

DMC. Therefore, the underlying assumption in this control design is that the plant is being operated around an open-loop stable operating point, which comprises the majority of processes, but not all. Open-loop unstable processes occur in process industries typically in the presence of heat integration, recycle or exothermic reactions (Muske and Rawlings [1993b]). While optimizing the economics of the process, an unstable steady state or points around the unstable steady state could be the desirable operating points. Therefore, it is important to control the process at or around an unstable steady state. To ensure the applicability of the CL-DRTO system to unstable processes, it is essential for the underlying controller to be specifically designed to handle unstable operating points, and for the DRTO scheme to be able to encapsulate such a control design.

1.2 Research Problem Statement

The primary goal of this research is to design a CL-DRTO strategy that could readily be applied to open-loop unstable processes while taking into account the effects of the underlying control system. The objectives of this research are to:

1. Choose a stabilizing MPC and modify it to be encapsulated into the DRTO optimization
2. Formulate a closed-loop DRTO by incorporating the stabilizing MPC problems as subproblems, to form a multi-level dynamic optimization problem
3. Reformulate the multi-level dynamic optimization problem into a single-level mathematical program with complementarity constraints (MPCC) using a simultaneous solution approach
4. Apply the stabilizing CL-DRTO strategy to a process described by a nonlinear differential-algebraic equation (DAE) model

1.3 Thesis Outline

1. In **Chapter 2** we present a brief literature review to elaborate on the current state of the art DRTO strategies and existing stabilizing MPC formulations.

2. In **Chapter 3** we present a closed-loop DRTO with a stabilizing MPC at the lower level. This multi-level dynamic optimization is reformulated into a single-level MPCC by replacing the stabilizing MPC subproblems with their equivalent Karush-Kuhn-Tucker (KKT) optimality conditions.

3. In **Chapter 4** we validate the proposed approach by applying the CL-DRTO strategy to single-input single-output (SISO) and multi-input multi-output (MIMO) CSTR case studies. The case studies include linear plant models, nonlinear plant models, target tracking and economic objective functions at the DRTO level.

4. In **Chapter 5** we summarize the key contributions of our study and lay out the potential future directions for this research.

Chapter 2

Preliminaries

In this chapter we provide a brief literature review of state of the art dynamic real-time optimization (DRTO) strategies and stabilizing model predictive control (MPC) strategies to cover the essential background for our work.

2.1 Literature Review: DRTO

Real-time optimization (RTO) is a supervisory strategy at the upper level of the industrial process automation architecture that computes economically optimal set-point trajectories that are in turn passed on to the lower-level model predictive controller (MPC) for tracking. Traditionally, the high fidelity model used at the RTO level is a nonlinear steady-state model. The execution of these steady state optimization calculations is therefore inhibited by the requirement for the system to reach steady state conditions. The resulting optimal solutions therefore do not recognize the transients needed to get to these steady-state optimal values, and as such, may lead to suboptimal dynamic operation. In certain cases, the economic steady-state optimum may not even be achievable dynamically due to the violation of certain (safety) out-

put constraints along the way. As such, this may lead to suboptimal performance for processes that have frequent transitions or exhibit slower dynamics. To circumvent this limitation two different methodologies were proposed; the two-layer dynamic RTO (DRTO) as illustrated in Fig 1.1b and the one-layer economic model predictive control (EPMC) as illustrated in Fig 1.1c.

Economic model predictive control (EMPC) was proposed as an alternative to the traditional two-layer approach in which a single layer combines both the economic and the tracking optimizations. The objective function of an EMPC can either be entirely based on economics, or a hybrid between economic and regulatory performance weighted with appropriate tuning parameters. A key difference between an EMPC and RTO, is that the plant economics are optimized at the controller sampling frequency in the EMPC approach. The EMPC approach is intended to address the issue of model inconsistencies that occur in a traditional two-layer approach. Amrit *et al.* [2011] provided a rigorous analysis of using terminal regions and penalties in EMPC that guarentees closed-loop stability. Heidarinejad *et al.* [2012] proposed an economic MPC based on Lyapunov techniques with two operating modes. Economic optimization is performed in the first operation mode, where the MPC maintains the system in a predefined region. In the second operating mode, the EMPC drives the system to the optimal steady state and the Lyapunov constraints guarantee closed-loop stability. Ellis and Christofides [2014] implemented a Lyapunov based economic model predictive control (LEMPC), that can accomodate a time-varying economic objective function. They perform a theorotical analysis of LEMPC, demonstrating the property of closed-loop stability. A comprehensive review of EMPC approaches, formulations, closed-loop stability and performance is presented in Ellis *et al.* [2014].

A terminal penalty, terminal constraint and the control tracking term in the hybrid objective function might still require optimal targets to track, that are computed by the economic optimization layer. Therefore, the implementation of an EMPC

strategy, particularly for a sufficiently small sampling time, is difficult since the entire optimization should be performed within the sampling time. Recent advances in large-scale dynamic optimization algorithms and increasing computational power are promising developments towards adressing this challenge. However, the two-layer approach is commonly used in the process industries due to its multi-layer reliability. Therefore, in our current work, we focus on the two-layer approach to be consistent with the automation architecture commonly applied in the process industries to enable implementation of the proposed strategy without having to completely re-design the traditional architecture.

In the two-layer approach, one of the earliest attempts to bridge the gap between a traditional steady-state RTO and MPC was to use a two-stage LP-MPC (QP-MPC) cascade control strategy Qin and Badgwell [2003]. In this strategy a new optimization layer is inserted in between the RTO and the MPC layer. The objective of LP-MPC (QP-MPC) is to provide set-points to the lower-level MPC based on the optimal operating points obtained from the infrequent RTO optimization. The model used in this layer is consistent with the dynamic linear model used at the lower-level MPC and is executed at the same frequency as the MPC. Stability analysis and performance of the two-stage LP-MPC (QP-MPC) for both constrained and unconstrained cases are discussed in Ying and Joseph [1999].

Within the DRTO paradigm, the first implementations were based on an open-loop framework. Kadam *et al.* [2002] employed a vertical decomposition approach to form a two-layer structure, in which the upper-level DRTO has an economic objective function that is optimized with the input trajectory as the decision variable. The resulting output response based on the computed optimal input trajectories is used to construct the set-point trajectory for the lower level MPC which then computes the optimal control input. Re-optimization is triggered based on a disturbance sensitivity analysis. Würth *et al.* [2011] proposed a similar approach, using a neighboring-extremal MPC

strategy for the lower level controller. A time scale separation of the disturbances is performed wherein only the slower trends of the disturbances were considered at the upper level.

In Tosukhowong *et al.* [2004], a slow-scale reduced order dynamic model is utilized at the RTO level and the MPC target tracking is implemented at a higher frequency; the upper layer optimizations are performed independent of the underlying control system. Due to the presence of disturbances and modeling errors, the DRTO set-points may not be feasible for the underlying controller. Consequently, they suggest passing the set-points to a coordination level that finds a feasible set-point, achievable by the lower-level controller, that is closest to the solution provided by the DRTO in a least-squares sense. The frequency with which the DRTO is executed is determined based on the eigenvalues of the system around the steady-state.

More recently, a closed-loop DRTO framework was proposed in Jamaludin and Swartz [2017b]. In this study, the DRTO formulation accounts for the effect of the underlying model predictive controller on the predicted response of the plant. In this closed-loop DRTO (CL-DRTO) framework, MPC optimization problems are embedded into the upper-level economic optimization to generate the predicted plant response under the action of constrained MPC, creating a virtual closed-loop system. In comparison to the open-loop approach, the reference trajectories (for both the outputs and the inputs) are the decision variables of the primary economic objective function. These reference trajectories are passed down to the lower-level MPC, that calculates the optimal control inputs to be applied to the plant. Fig 2.1 shows a schematic of the closed-loop DRTO approach. Accurate approximation strategies have been devised in Jamaludin and Swartz [2017a] that significantly reduce the CL-DRTO computation time. In Li and Swartz [2019], the CL-DRTO strategy was extended to dynamic coordination of distributed MPCs. This closed-loop formulation predicts the interaction between the distributed MPCs and helps coordinate the local MPCs with no direct

contact between them.

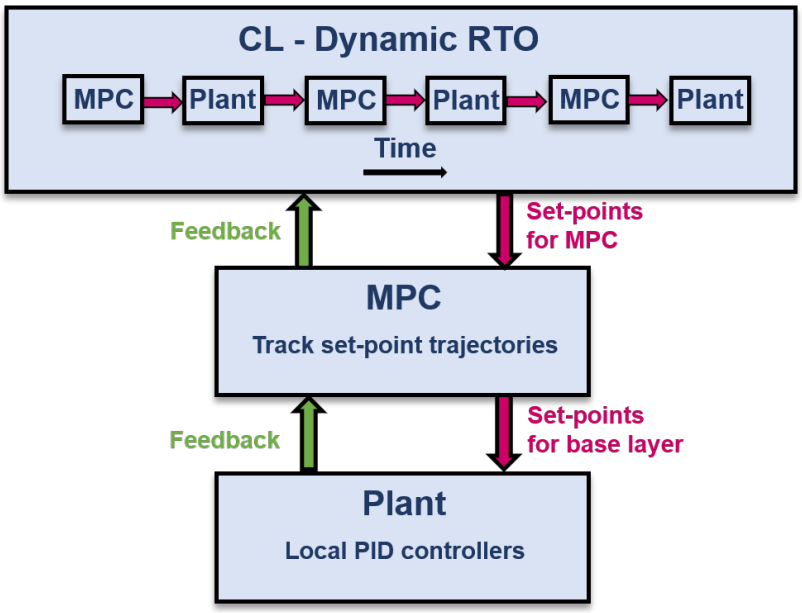

Figure 2.1: Schematic of a closed-loop DRTO approach

2.2 Literature Review: Stabilizing MPC

Open-loop unstable processes occur in process industries typically in the presence of heat integration, recycle or exothermic reactions (Muske and Rawlings [1993a]). For instance, a first order exothermic reaction with recycle in a continuous stirred tank reactor (CSTR) could exhibit three steady-states as shown in Uppal *et al.* [1974], with one of them being an unstable steady state. Although a linearized model (valid over a narrow range around the steady state) can be used in the predictive control framework, a formulation that can handle unstable operating points is necessary. Early work in the area of model predictive control focused primarily on the needs of process industries. Research on model based control was therefore motivated by industrial requirements. Due to its simple nature, step response or impulse response was used to model the plants, and control algorithms were developed using these

models. These input or step response models cannot represent unstable plants and therefore the proponents of MPC for industrial applications restricted the algorithm to stable plants or relied on a long horizon that is equivalent to the settling time of the plant to address the issue of stability. The growing popularity of MPCs propelled academic research where a theoretical study of stability analysis emereged. Several methodologies have been proposed to address the issues of stability, feasibility and optimality of standalone MPC formulations, of which a comprehensive review is given in Mayne *et al.* [2000]. Stability-guaranteeing MPC formulations mostly employ some form of endpoint stability constraint and/or penalty, relying on the control horizon to be sufficiently large to ensure feasibility and stability. Mhaskar *et al.* [2005] proposed a Lyapunov based MPC that guarentees stability along with input and state constraint satisfaction for nonlinear systems. They explicitly characterize a set of initial conditions from which guaranteed feasibility and stability can be achieved. In the next section, Lyapunov techniques are introduced very briefly.

2.2.1 Lyapunov based controller

Stability in the sense of Lyapunov (I.S.L) is very useful, particularly, for nonlinear systems. A function $V(x)$ is a Lyapunov function if, $V(x)$ is positive definite and $\dot{V}(x)$ is negative semidefinite. A system is stable around an equilibrium point (can be chosen as origin without loss of generality) if such a Lyapunov function exists and it is asymptotically stable if $\dot{V}(x) < 0$, that is, if $\dot{V}(x)$ is negative definite. Let a continuous time nonlinear state-space model of the system be represented as follows:

$$\dot{x}(t) = f(x(t)) + G(x(t))u(t) \tag{2.1}$$

$$x \in \mathbf{X} \ , \ u \in \mathbf{U}$$

where $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$ denotes state and input vectors, $\mathbf{X} \subseteq \mathbb{R}^{n_x}$, $\mathbf{U} \subseteq \mathbb{R}^{n_u}$ denotes the constraints on the state variables and the manipulated inputs. Consider the nonlinear system shown in Eqn 2.1 for which a control Lyapunov function, $V(x)$, exists. For this system to be stable I.S.L, it must satisfy the following conditions:

- $V(x)$ must be positive definite
- $\dot{V}(x) \leq 0$, which implies $\frac{\mathrm{d}V(x)}{\mathrm{d}x} * [f(x(t)) + G(x(t))u(t)] \leq 0$

To design a stabilizing controller, the above conditions can be added as contraints to a predictive control optimization. For this Lyapunov based controller, an explicit set of initial conditions can be constructed from which input and state constraint satisfaction can be obtained (Mhaskar *et al.* [2005]).

Controller satisfaction of state and input constraints can be obtained if the closed-loop state evolves within the region $\phi_{x,u}$, where $\phi_{x,u} = \{x \in \mathbf{X} : \dot{V}(x) < 0\}$. Therefore, stability I.S.L is guaranteed if the state trajectory $x(t)$ remains in the set $\phi_{x,u}$. This can be achieved by constructing the largest invariant set contained in $\phi_{x,u}$. Using the level sets of the Lyapunov function $V(x)$, $\Omega_{x,u}$ can be constructed such that $\Omega_{x,u} = \{x \in \mathbb{R}^{n_x} : V(x) \leq C_{x,u}^{max}\}$ where $C_{x,u}^{max}$ is the largest positive number for which $\Omega_{x,u} \subseteq \phi_{x,u}$. A visual representation of this initial set is shown in Fig 2.2.

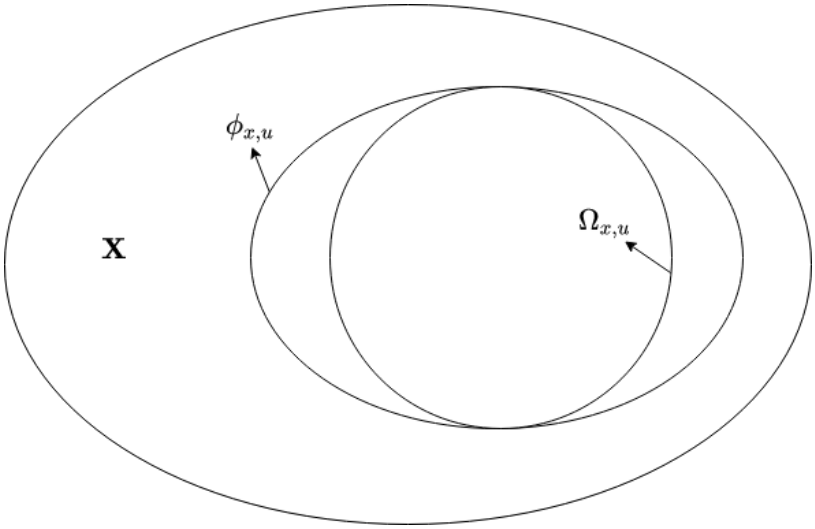

Figure 2.2: A schematic representing constrained stability regions together with state constraints

2.3 Linear and nonlinear models

In our work, both linear and nonlinear discrete time dynamic models of the plant are considered at the DRTO level. In general, nonlinear dynamic model of the plant is represented as follows,

$$\begin{aligned} \hat{x}_{j+1}^{DRTO} &= f^{DRTO}(\hat{x}_j^{DRTO}, \hat{u}_j^{DRTO}) \\ \hat{y}_j^{DRTO} &= h^{DRTO}(\hat{x}_j^{DRTO}) \end{aligned} \tag{2.2}$$

where, $\hat{x}_j^{DRTO} \in \mathbb{R}^{n_x}$, $\hat{y}_j^{DRTO} \in \mathbb{R}^{n_y}$ and $\hat{u}_j^{DRTO} \in \mathbb{R}^{n_u}$ represent the predicted states, outputs and inputs of the system respectively. f^{DRTO} represents the dynamic model of the plant and h^{DRTO} represents the relationship between states and the outputs of the system. This model can be obtained by discretizing the set of nonlinear ordinary differential equations (ODE) system using a scheme such as a backward Euler approximation, orthogonal collocation of finite elements or multiple shooting. In our work, implicit Euler approximation is used with a finite element interval of Δt_{MPC},

the sampling time of the MPC.

A linear model shown in Eqn 2.3 is used at the MPC level. Note that a linear model could also be used at the DRTO level for steady-state calculations as well as to generate the predicted plant response. Linear continuous state-space model could be obtained by discretizing a nonlinear ODE system around the desired steady state of the system and can be converted to a discrete time model as shown below.

$$\begin{aligned} \hat{x}_{j+1}^{DRTO} &= A\hat{x}_{j}^{DRTO} + B\hat{u}_{j}^{DRTO} \\ \hat{y}_{j}^{DRTO} &= C\hat{x}_{j}^{DRTO} \end{aligned} \tag{2.3}$$

2.4 Open-loop DRTO

In OL-DRTO, the objective is to optimize an economic cost function with the input trajectory as the decision variable, that is determined based on the predicted open-loop dynamics of the plant. The resulting plant response based on the computed optimal input trajectories, at the end of each DRTO interval is used to construct the set-point trajectory for the lower level MPC. These optimization computations are performed independent of the underlying control system. In this section the formulation of an OL-DRTO as shown in Jamaludin and Swartz [2017b] is presented.

$$\begin{aligned} \underset{\hat{u}^{DRTO}}{\text{maximize}} \quad & \phi_{econ}^{DRTO}(\hat{x}^{DRTO}, \hat{u}^{DRTO}, \hat{y}^{DRTO}) \\ \text{s.t.} \quad & \hat{x}_{j+1}^{DRTO} = f^{DRTO}(\hat{x}_{j}^{DRTO}, \hat{u}_{j}^{DRTO}) \quad , \quad j = 0, \ldots, N-1 \\ & \hat{y}_{j}^{DRTO} = h^{DRTO}(\hat{x}_{j}^{DRTO}) \quad , \quad j = 1, \ldots, N \\ & y_{min} \leq \hat{y}_{j}^{DRTO} \leq y_{max} \\ & u_{min} \leq \hat{u}_{j}^{DRTO} \leq u_{max} \end{aligned} \tag{2.4}$$

where, $\hat{x}_{j}^{DRTO}$, $\hat{y}_{j}^{DRTO}$, $\hat{u}_{j}^{DRTO}$, f^{DRTO} and h^{DRTO} are defined in the previous section,

ϕ_{econ}^{DRTO} represents an economic objective function, u_{min} and u_{max} are input constraints that describe the physical limitations of the system, y_{min} and y_{max} are the output constraints imposed on the problem. The set-points provided to the lower-level MPC is constructed based on the resulting open-loop plant response computed based on the optimal input trajectories.

Depending on the tuning of the lower level controller and modeling errors, the optimal trajectories determined by the DRTO system may not be achievable, as the economic optimization does not capture the behaviour of the underlying controller. Furthermore, in Jamaludin and Swartz [2017b], it is shown that the CL-DRTO strategy outperforms the OL-DRTO particulary when the lower-level controller is detuned. For these reasons, we focus our work on a stabilizing CL-DRTO approach and in the next section, we present the existing CL-DRTO formulation.

2.5 Closed-loop DRTO

Within a two-layer DRTO configuration, it is important to account for the behavior of the underlying MPC due to its impact on the plant's dynamic response. Jamaludin and Swartz [2017b] proposed embedding MPC subproblems within the DRTO formulation to generate the predicted closed-loop response. These MPC subproblems within the DRTO have to be consistent with the lower-level MPCs formulation and tuning parameters. The primary objective of the CL-DRTO strategy is to compute economically optimal reference trajectories that are provided as set-points to the underlying MPC system. This CL-DRTO strategy can be seen as a cascade MPC framework, with an EMPC scheme in the upper-level providing set-points to the lower level regulatory MPC, both of which can operate at different frequencies.

The models shown in Eqn 2.2 and Eqn 2.3 form the core part of the DRTO opti-

mization and are utilised to make the future predictions of the plant response. As discussed, closed-loop behavior is obtained by having the MPC problems embedded within the economic optimization. Incorporating the MPC subproblems within the DRTO optimization results in a multi-level dynamic optimization problem. This formulation was first implemented in Baker and Swartz [2008] where an interior point algorithm was used to solve multi-level optimization problems that stem from incorporating the constrained MPC response into a primary linear or quadratic optimization problem.

This CL-DRTO framework has a primary DRTO optimization problem that finds economically optimal reference trajectories over a finite DRTO prediction horizon N. Within this primary optimization, a series of MPC subproblems (shown in Eqn 2.9) are executed and the first control move from each MPC subproblem is used by the primary DRTO problem to generate the predicted dynamic response (and in turn compute the objective function and ensure that constraints are satisfied). The formulation of a CL-DRTO is shown in Eqn 2.5 below:

$$
\begin{aligned}
\underset{y^{Ref},u^{Ref}}{\text{maximize}} \quad & \phi_{econ}^{DRTO}(\hat{x}_j^{DRTO}, \hat{u}_{j-1}^{DRTO}, \hat{y}_j^{DRTO}) \\
\text{s.t.} \quad & \hat{x}_{j+1}^{DRTO} = f^{DRTO}(\hat{x}_j^{DRTO}, \hat{u}_j^{DRTO}) \quad , \quad j = 0, \ldots, N-1 \\
& \hat{y}_j^{DRTO} = h^{DRTO}(\hat{x}_j^{DRTO}) \quad , \quad j = 1, \ldots, N \\
& y_{min} \leq \hat{y}_j^{DRTO} \leq y_{max} \\
& h^{Ref}(y^{Ref}, u^{Ref}, y^{SP}, u^{SP}) = 0 \\
& \hat{u}_j^{DRTO} = \hat{u}_{j,0} \quad , \quad j = 0, \ldots, N-1 \\
& \hat{u}_{j,0} \;\; \epsilon \;\; \text{arg min} \;\; \phi_{mpc} \;\; \longrightarrow \;\; \text{Inner MPC subproblems} \\
& \qquad\qquad \text{s.t.} \;\; \overline{u}_{min} \leq \overline{u} \leq \overline{u}_{max}
\end{aligned}
\tag{2.5}
$$

where, $\overline{u}$ represents the optimal control action determined by the inner MPC sub-

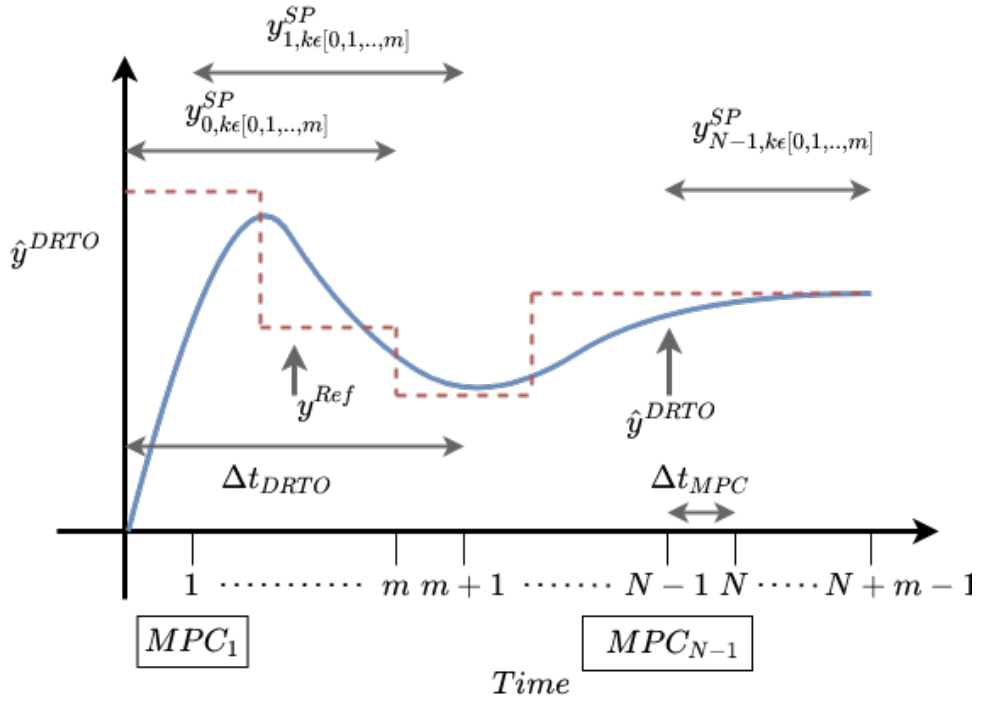

Figure 2.3: Relationship between reference trajectory and the set-point provided to the MPC

problems, $\hat{u}_{j,0} \in \mathbb{R}^{n_u}$ represents the first input move determined by the inner MPC subproblems, h^{Ref} is an equality constraint that relates the reference trajectory to the set-points provided to the lower-level MPC, N represents the DRTO prediction horizon, and ϕ^{DRTO}_{econ} represents the economic objective function that is being optimized. y^{Ref} and u^{Ref} represent the output and input reference trajectories respectively, determined by the upper-level DRTO and y^{SP}, u^{SP} are subsets of the reference trajectory that are passed on to the MPC for tracking. $\overline{u}_{min}$ and $\overline{u}_{max}$ are vectors of input constraints that describe the physical limitations of the system. y_{min} and y_{max} are the output constraints imposed on the problem.

Fig 2.3 presents a visualization of a single DRTO execution. For a DRTO prediction horizon of N, there are N MPC subproblems solved at every interval of sample time Δt_{MPC}. An output reference trajectory y^{Ref} is computed over this prediction horizon and each of the N MPC subproblems receives a subset of this reference trajectory

as it's target to track. Since each of the MPC subproblems has its own prediction horizon and is performed in a rolling horizon manner, the first control move from each of these N subproblems is implemented as the optimal DRTO input move $\hat{u}^{DRTO}$ and this serves as the connection between the DRTO and the embedded MPC problems.

Common solution approaches for dynamic optimization problems are a sequential approach and a simultaneous approach. A sequential solution method would involve solving iteratively the primary DRTO and the closed-loop simulation incorporating the MPC optimization subproblems until convergence. When the control step saturates, the input constraints in the MPC subproblems may result in derivative discontinuities in the primary DRTO problem. In this study, therefore, we follow the simultaneous solution approach presented in Jamaludin and Swartz [2017b]. The CL-DRTO problem is reformulated by replacing the inner MPC subproblems with their equivalent Karush-Kuhn-Tucker (KKT) optimality conditions. This transforms the original multi-level dynamic optimization problem into a single-level mathematical program with complementarity constraints (MPCC). An illustration of a general quadratic program (QP) transformed into its equivalent KKT optimality conditions is shown in Eqn 2.6 and Eqn 2.7.

Remark 1. We remark that closed-loop predictions are commonly used to account for the effects of uncertainty in the predicted response in robust MPC formulations. A brief survey of robust MPC along with the key idea of closed-loop predictions is presented in Bemporad and Morari [1999]. If there is no disturbance or modeling errors, the optimal control step calculated by the first MPC subproblem within the CL-DRTO and the lower level MPC are the same.

This transformation can be illustrated using the general QP form:

$$\begin{aligned} \min_{z} \quad & \frac{1}{2}\,\mathbf{z}^T\,H\,\mathbf{z} + \mathbf{g}^T\,\mathbf{z} \\ s.t. \quad & A\,\mathbf{z} = \mathbf{b}, \\ & \mathbf{z} \geq \mathbf{0} \end{aligned} \tag{2.6}$$

whose KKT optimality conditions are:

$$\begin{aligned} & H\,\mathbf{z} + \mathbf{g} - A^T\,\boldsymbol{\nu} - \boldsymbol{\eta} = \mathbf{0} \\ & A\,\mathbf{z} - \mathbf{b} = \mathbf{0}, \\ & z_i\,\eta_i = 0, \quad i \in I \\ & (\mathbf{z}, \boldsymbol{\eta}) \geq \mathbf{0} \end{aligned} \tag{2.7}$$

More details of the KKT optimality conditions are discussed in Section 3.2. In the next section, we present the stabilizing MPC used in our research.

2.6 Stabilizing MPC

As discussed in Section 2.5, the multi-level dynamic optimization needs to be reformulated and this requires the calculation of KKT optimality conditions of the inner MPC subproblems. Therefore, this approach is valid for several MPC formulations that use a linear model and a quadratic objective function, that result in a convex quadratic problem. Convex QP problems have a unique global optimum for which the KKT optimality conditions are both necessary and sufficient. In this work, as one possible candidate for a stabilizing MPC formulation, the model predictive control formulation proposed in Muske and Rawlings [1993b] is utilized. In this formulation, an infinite horizon open-loop objective function as shown in Eqn 2.8 is minimized to

obtain the optimal input $\tilde{u}^m$, where $\tilde{u}^m = \begin{bmatrix} \tilde{u}_0^T & \tilde{u}_1^T & \dots & \tilde{u}_{m-1}^T \end{bmatrix}^T$. $\tilde{u}_k$ represents the input step determined by the MPC at time k in deviation from its set-point u^{SP}.

$$\underset{\tilde{u}^m}{\text{minimize}} \quad \sum_{k=0}^{\infty} (\tilde{y}_k)^T Q(\tilde{y}_k) + (\tilde{u}_k)^T R(\tilde{u}_k) + \Delta\tilde{u}_k^T S \Delta\tilde{u}_k \tag{2.8}$$

$\tilde{y}_k$ is the output in deviation from y^{SP} and $\Delta\tilde{u}_k$ is defined in Eqn 2.9 as the difference between $\tilde{u}_k$ and $\tilde{u}_{k-1}$. Q, R, and S are symmetric positive semi-definite penalty matrices on the outputs, inputs and change of the inputs respectively. m represents the prediction horizon, which together with Q, R and S form the tuning parameters for the MPC optimization. The infinite horizon problem shown in Eqn 2.8 can be expressed as a finite horizon open-loop objective function by introducing a terminal cost function with a penalty matrix $\overline{Q}$. Thus, the MPC finite horizon optimization can be written as follows,

$$\begin{aligned}
\underset{\tilde{u}^m}{\text{minimize}} \quad & (\tilde{x}_m)^T \overline{Q} (\tilde{x}_m) + \Delta\tilde{u}_m^T S \Delta\tilde{u}_m + \\
& \sum_{k=0}^{m-1} (\tilde{x}_k)^T C^T Q C (\tilde{x}_k) + (\tilde{u}_k)^T R (\tilde{u}_k) + \Delta\tilde{u}_k^T S \Delta\tilde{u}_k \\
\text{s.t.} \quad & \tilde{x}_{k+1} = A\tilde{x}_k + B\tilde{u}_k \ , \ k = 0, \dots, m-1 \\
& \Delta\tilde{u}_k = \tilde{u}_k - \tilde{u}_{k-1} \ , \ k = 0, \dots, m \\
& \tilde{u}_{min} \leq \tilde{u}_k \leq \tilde{u}_{max} \ , \ k = 0, \dots, m-1 \\
& \tilde{u}_k = 0 \ , \ k \geq m \\
& \tilde{x}_m^u = 0
\end{aligned} \tag{2.9}$$

In the state-space model of the plant shown in Eqn 2.9, $\tilde{x}_k$ represents the states of the system in deviation from its set-point x^{SP} and therefore, $\tilde{x}_0$ represents the current state of the plant in deviation form. The constraint $\tilde{x}_m^u = 0$ requires the unstable modes (if any) of the states to go to zero by the end of the horizon (the so-called

terminal stability constraint). Vector $\tilde{u}^m$ that optimizes the infinite horizon objective function is computed and the first input value is implemented on the plant. This strategy is repeated at each time step in a rolling horizon fashion. $\tilde{u}_{min}$ and $\tilde{u}_{max}$ are input constraints that describe the physical limitations of the system in deviation form.

The pair (x^{SP},u^{SP}) can be calculated as $x^{SP} = f^{DRTO}(x^{SP}, u^{SP})$, where f^{DRTO} is the nonlinear model of the plant. If a nonlinear model of the plant is not available, the linearized state-space equation can be used to find x^{SP} and u^{SP} as $x^{SP} = Ax^{SP} + Bu^{SP}$. If an output target y^t is given, then the steady state set-point pair (x^{SP},u^{SP}) can be computed as shown below (Muske and Rawlings [1993b]). The optimization in Eqn 2.10 ensures that the target of y^t is achieved if the corresponding input set-point u^{SP} is within the input constraints, otherwise, a target $y^{SP} = Cx^{SP}$ that is closest to y^t is achieved. Here Q_s represents the error penalty matrix.

$$\begin{aligned} \underset{x^{SP},u^{SP}}{\text{minimize}} \quad & (y^t - Cx^{SP})^T Q_s (y^t - Cx^{SP}) \\ \text{s.t.} \quad & x^{SP} = Ax^{SP} + Bu^{SP} \\ & u_{min} \leq u^{SP} \leq u_{max} \end{aligned} \tag{2.10}$$

For open-loop stable process, $\overline{Q}$ can be used to capture the infinite horizon cost of the stable modes and computed as the solution of the discrete Lyapunov equation shown below,

$$\overline{Q} = C^T Q C + A^T \overline{Q} A \tag{2.11}$$

For open-loop unstable processes $\overline{Q}$ needs to be modified to only capture the infinite horizon cost of the stable modes. To achieve this, the state space is decomposed into stable and unstable modes, and the penalty and terminal stability constraints are

calculated by first computing $\tilde{V}_u$, $\tilde{V}_s$ and J_s as follows:

$$A = VJV^{-1} = \begin{bmatrix} V_u & V_S \end{bmatrix} \begin{bmatrix} J_u & 0 \\ 0 & J_s \end{bmatrix} \begin{bmatrix} \tilde{V}_u \\ \tilde{V}_s \end{bmatrix} \tag{2.12}$$

$$\overline{Q} = \tilde{V}_s^T \Sigma \tilde{V}_s \quad \text{where,} \quad \Sigma = V_s^T C^T Q C V_s + J_s^T \Sigma J_s \tag{2.13}$$

$$\tilde{V}_u(\tilde{x}_m) = 0 \implies \tilde{V}_u \left[A^{m-1}B, A^{m-2}B, \ldots, B\right] \tilde{u}^m = -\tilde{V}_u A^m(\tilde{x}_0) \tag{2.14}$$

In Eqn 2.12, J is the Jordan matrix of A, and all the stable and unstable eigenvalues are contained in J_s and J_u respectively. $\tilde{V}_u$ contains eigenvectors corresponding to the unstable modes of the system. The stable and unstable modes are found by partitioning the matrix A using Jordan decomposition as shown in Eqn 2.12. The quadratic program shown in Eqn 2.9 can be converted to a standard quadratic program with simple algebraic manipulations as shown in Eqn 2.15.

$$\begin{aligned} \underset{\tilde{u}^m}{\text{minimize}} \quad & \frac{1}{2} \left(\tilde{u}^m\right)^T H \tilde{u}^m + \left(G\left(\tilde{x}_0\right) - F\left(\tilde{u}_{-1}\right)\right)^T \tilde{u}^m \\ \text{s.t.} \quad & \overline{\tilde{u}}_{min} \leq \tilde{u}^m \leq \overline{\tilde{u}}_{max} \\ & \tilde{V}_u \left[A^{m-1}B, A^{m-2}B, \ldots, B\right] \tilde{u}^m = -\tilde{V}_u A^m(\tilde{x}_0) \end{aligned} \tag{2.15}$$

$\overline{\tilde{u}}_{min}$, $\overline{\tilde{u}}_{max} \in \mathbb{R}^{mn_u}$ represent vectors of minimum ($\tilde{u}_{min}$) and maximum input ($\tilde{u}_{max}$) constraints respectively and $\tilde{u}_{-1}$ is the previous input in deviation form. Matrices H,

G, F are given in Eqn 2.16 and Eqn 2.17 (Muske and Rawlings [1993b]).

$$H = \begin{bmatrix} B^T\overline{Q}B + R + 2S & B^TA^T\overline{Q}B - S & \dots & B^TA^{T^{N-1}}\overline{Q}B \\ B^T\overline{Q}AB - S & B^T\overline{Q}B + R + 2S & \dots & B^TA^{T^{N-2}}\overline{Q}B \\ \vdots & \vdots & \ddots & \vdots \\ B^T\overline{Q}A^{N-1}B & B^T\overline{Q}A^{N-2}B & \dots & B^T\overline{Q}B + R + 2S \end{bmatrix} \tag{2.16}$$

$$G = \begin{bmatrix} B^T\overline{Q}A \\ B^T\overline{Q}A^2 \\ \vdots \\ B^T\overline{Q}A^N \end{bmatrix} \quad , \quad F = \begin{bmatrix} S \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{2.17}$$

Remark 2. Several stabilizing MPC formulations have been published in the literature and the proposed formulation herein for the CL-DRTO works for any of these MPC strategies as long as the problem remains convex. For MPC formulations that are nonconvex or that use nonlinear models, a different solution strategy, perhaps a sequential solution approach, should be considered since the KKT solutions may no longer be unique.

Remark 3. For nonsquare systems, where the number of inputs are greater than the number of the controlled outputs, the control tracking term $(\tilde{u}_k)^T R(\tilde{u}_k)$ is useful. A nonzero R is chosen and some of the inputs require ideal resting values (IRVs), calculated by the upper level economic optimization, to obtain a unique solution (Qin and Badgwell [2003]).

Chapter 3

Stabilizing CL-DRTO

In this chapter we present the detailed rigorous formulation for the stabilizing CL-DRTO formulation. Firstly, we formulate the stabilizing CL-DRTO problem for linear models, as a multi-level optimization problem. We then provide a suitable solution strategy and reformulate the problem. Subsequently, we introduce nonlinearity in the model at the DRTO level and provide the corresponding formulation. Information on the computation for the proposed approach is also presented in this chapter.

The formulations in this chapter have been submitted to and presented in:

Sundaresan Ramesh, P., Swartz, C. L. E., and Mhaskar, P. (2020) "Closed-loop Dynamic Real-Time Optimization with Stabilizing Model Predictive Control", *Submitted to AICHE Journal*, in review.

Sundaresan Ramesh, P., Swartz, C. L. E., and Mhaskar, P. (2019), "Closed-Loop Dynamic Real Time Optimization with a Stable MPC Formulation", *AICHE Annual Meeting* (2019), Orlando, FL, USA.

Sundaresan Ramesh, P., Swartz, C. L. E., and Mhaskar, P. (2019), "Closed-loop Dynamic Real Time Optimization with Stable Model Predictive Control Formulation",

CCEC Annual Meeting (2019), Halifax, NS, Canada.

3.1 Problem formulation

In this chapter, firstly, we consider the models available for the DRTO formulations are linear dynamic models as shown in Eqn 2.3. As discussed earlier, open-loop unstable processes occur in certain process operations and its unstable steady-state could be the economically optimal operating point. Existing stabilizing MPCs are able to provide an optimal input trajectory that ensures stability of the plant response. In order to guarantee stability of the CL-DRTO strategy it is essential for the CL-DRTO to encapsulate such a design. An arbitrary bounded trajectory given as set-points to a stabilizing MPC could result in infeasibility issues due to its inability to satisfy the stability constraints and input constraints. Thus, the proposed strategy for a stable CL-DRTO involves embedding these stabilizing MPC subproblems within the primary DRTO optimization.

The formulation of this strategy is shown in Eqn 3.1. It can be seen in Eqn 3.1, that the proposed strategy results in a multi-level optimization problem. For a given initial state of the plant, the primary optimization predicts a closed-loop trajectory of the plant based on the nonlinear model, output constraints and the inner MPC subproblems.

Remark 4. The success of MPC in industry can be partly attributed to the ability to enforce constraints on the states or outputs. They are different from input constraints in that they do not typically arise from physical limitations like valve saturation, but rather are imposed to limit the operating range of the process. Some of the feasibility and stability issues arising from incorporating hard output constraints are shown in Scokaert and Rawlings [1999]. More recently, a Lyapunov-based MPC was proposed Mhaskar *et al.* [2005] that guarantees constraint satisfaction from a well characterized

set of initial conditions. In our work, the stabilizing MPC, both at the lower level and at the inner MPC subproblems are based on linear models (and implementation illustration on nonlinear models), resulting in plant-model mismatch (and disabling the feasibility and stability guarantees of Mhaskar *et al.* [2005]). To preserve feasibility therefore, no hard output constraints are enforced at the MPC level. Instead, the hard constraints are present in the outer-loop primary optimization to ensure that the closed-loop plant response is within the desired operating range.

The multi-level optimization shown in Eqn 3.1 is performed every Δt_{DRTO}, the sampling time of DRTO. As discussed in Section 2.5, the inner MPC subproblems are solved at every time step j, and the very first input move determined by each of the N MPC subproblems is implemented in the DRTO model. Therefore, the internal variables of the MPC subproblems change every time step and hence the additional subscript j. $\tilde{x}_{j,0}$ represents the initial value of the state and is calculated as $A\hat{x}_{j-1}^{DRTO} + B\hat{u}_{j-1}^{DRTO}$). $\tilde{x}_{j,k}$ is in deviation from $x_{j,0}^{SP}$. $u_{j,0}^{SP}$ and $x_{j,0}^{SP}$ represent the first input and state set-point provided to the MPC subproblems at every time step j respectively.

Fig 3.1 shown below, illustrates the computational flow within a single CL-DRTO

execution.

$$
\begin{aligned}
\underset{y^{Ref},u^{Ref}}{\text{maximize}} \quad & \phi_{econ}^{DRTO}(\hat{x}^{DRTO},\hat{u}^{DRTO},\hat{y}^{DRTO}) \\
\text{s.t.} \quad & \hat{x}_{j+1}^{DRTO} = A\hat{x}_j^{DRTO} + B\hat{u}_j^{DRTO} \quad , \quad j=0,\ldots,N-1 \\
& \hat{y}_j^{DRTO} = C\hat{x}_j^{DRTO} \quad , \quad j=1,\ldots,N \\
& h^{Ref}(y^{Ref},u^{Ref},y^{SP},u^{SP}) = 0 \\
& y_{min} \le \hat{y}_j^{DRTO} \le y_{max} \\
& \hat{u}_j^{DRTO} = \tilde{u}_{j,0} + u_{j,0}^{SP} \quad , \quad j=0,\ldots,N-1 \\
& \tilde{u}_{j,0} \;\; \epsilon \;\; \underset{\tilde{u}_{j,k}}{\arg\min} \;\; \Phi_j^{MPC} \quad , \quad j=0,\ldots,N-1 \\
& \qquad \text{s.t.} \;\; \tilde{x}_{j,k+1} = A\tilde{x}_{j,k} + B\tilde{u}_{j,k} \quad , \quad k=0,\ldots,m-1 \\
& \qquad\quad \Delta\tilde{u}_{j,k} = \tilde{u}_{j,k} - \tilde{u}_{j,k-1} \quad , \quad k=0,\ldots,m \\
& \qquad\quad \tilde{u}_{min} \le \tilde{u}_{j,k} \le \tilde{u}_{max} \quad , \quad k=0,\ldots,m-1 \\
& \qquad\quad \tilde{u}_{j,k} = 0 \quad , \quad k \ge m \\
& \qquad\quad \tilde{V}_u \left[A^{m-1}B, A^{m-2}B, \ldots, B\right] \tilde{u}_j^m = -\tilde{V}_u A^m(\tilde{x}_{j,0})
\end{aligned}
\tag{3.1}
$$

where, $\Phi_j^{MPC} = (\tilde{x}_{j,m})^T\overline{Q}(\tilde{x}_{j,m}) + \Delta\tilde{u}_{j,m}^T S\Delta\tilde{u}_{j,m} + \sum_{k=0}^{m-1}(\tilde{x}_{j,k})^T C^T QC(\tilde{x}_{j,k}) + (\tilde{u}_{j,k})^T R(\tilde{u}_{j,k}) + \Delta\tilde{u}_{j,k}^T S\Delta\tilde{u}_{j,k}$

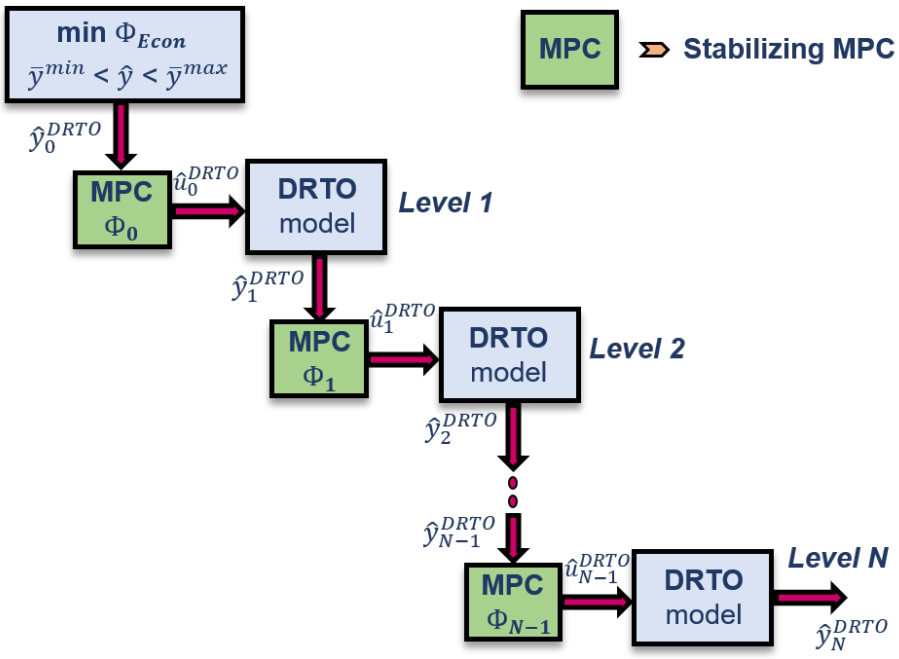

Figure 3.1: Illustration of the closed-loop dynamic optimization for a single DRTO step

3.2 Closed-loop problem reformulation

Common solution approaches for dynamic optimization problems are a sequential approach and a simultaneous approach. A sequential solution method would involve solving iteratively the primary DRTO and the closed-loop simulation incorporating the MPC optimization subproblems until convergence. When the control step saturates, the input constraints in the MPC subproblems may result in derivative discontinuities in the primary DRTO problem. In this study, therefore, we follow the simultaneous solution approach presented in Jamaludin and Swartz [2017b]. The CL-DRTO problem is reformulated by replacing the inner MPC subproblems with their equivalent Karush-Kuhn-Tucker (KKT) optimality conditions. This transforms the original multi-level dynamic optimization problem into a single-level mathematical program with complementarity constraints (MPCC).

The stabilizing MPC used here is a convex quadratic program and hence these condi-

tions are both necessary and sufficient for optimality. In this section, these optimality conditions are derived. Although, Eqn 2.9 and Eqn 2.15 represent the same problem, the KKT conditions are obtained from Eqn 2.15 since they result in a much smaller number of variables and equations.

The Lagrangian of the inner MPC optimization subproblems at every time step j could be written as:

$$\begin{aligned} L_j \;=\; & \frac{1}{2}\left(\tilde{u}_j^m\right)^T H \tilde{u}_j^m \;+\; \left(G\left(\tilde{x}_{j,0}\right) - F\left(\tilde{u}_{j,-1}\right)\right)^T \tilde{u}_j^m \;-\; \left(\overline{\eta}_j^1\right)^T \left(\tilde{u}_j^m - \overline{\tilde{u}}_{min}\right) \;- \\ & \left(\overline{\eta}_j^2\right)^T \left(\overline{\tilde{u}}_{max} - \tilde{u}_j^m\right) \;-\; \overline{\lambda}_j^T (\tilde{V}_u \left[A^{m-1}B, A^{m-2}B, \ldots, B\right] \tilde{u}_j^m + \tilde{V}_u A^m(\tilde{x}_{j,0})) \end{aligned}$$

$\tilde{u}_j^m$ is a vector of control inputs over the entire control horizon m at every DRTO step. $\overline{\eta}_j^i$, i = 1,2 represent composite vectors of $\eta_{j,k}^i$ which are Lagrange multipliers for the inequality input constraints. $\overline{\lambda}_j$ is a vector of Lagrange multipliers for the equality end point stability constraints.

1. Lagrange gradient: Since $\tilde{u}_j^m$ is the only decision variable, at every time step j, in the compact QP MPC problem, the gradient of $\tilde{u}_j^m$ w.r.t the Lagrangian is zero.
   - $\nabla_{\tilde{u}_j^m} L_j = 0$
2. Primal feasibility: Input inequality constraints transformed into equalities using slack variables $\mu_{j,k}^i$, $i = 1, 2$
   - $\tilde{u}_{j,k} - \tilde{u}_{min} - \mu_{j,k}^1 = 0 \quad , \quad$ k = 0,1,. . . ,m-1
   - $\tilde{u}_{max} - \tilde{u}_{j,k} - \mu_{j,k}^2 = 0 \quad , \quad$ k = 0,1,. . . ,m-1
   - $\mu_{j,k}^1$, $\mu_{j,k}^2 \geq 0 \quad , \quad$ k = 0,1,. . . ,m-1

3. Dual feasibility:

   - $\eta^1_{j,k}$, $\eta^2_{j,k} \geq 0$, k = 0,1,. . . ,m-1

4. Complementarity constraints:

   - $\overline{\eta}^i_j \odot \overline{\mu}^i_j = 0$, i = 1,2

$\overline{\mu}^i_j$, i = 1,2 represent composite vectors of $\mu^i_{j,k}$. $\odot$ represents the element-wise Hadamard product of two vectors. The inner MPC optimization subproblems in the primary DRTO optimization can be replaced with the set of equations shown in Eqn 3.2.

$$
\begin{aligned}
& H\tilde{u}^m_j + \left(G\left(\tilde{x}_{j,0}\right) - F\left(\tilde{u}_{j,-1}\right)\right) - \overline{\eta}^1_j + \overline{\eta}^2_j - (\tilde{V}_u \left[A^{m-1}B, A^{m-2}B, \ldots, B\right])^T \overline{\lambda}_j = 0 \\
& \tilde{u}^m_j - \overline{\tilde{u}}_{min} - \overline{\mu}^1_j = 0 \\
& \overline{\tilde{u}}_{max} - \tilde{u}^m_j - \overline{\mu}^2_j = 0 \\
& \tilde{V}_u \left[A^{m-1}B, A^{m-2}B, \ldots, B\right] \tilde{u}^m = -\tilde{V}_u A^m(\tilde{x}_{j,0}) \\
& \overline{\eta}^1_j \odot \overline{\mu}^1_j = 0 \\
& \overline{\eta}^2_j \odot \overline{\mu}^2_j = 0 \\
& \overline{\mu}^1_j \, , \, \overline{\mu}^2_j \, , \, \overline{\eta}^1_j \, , \, \overline{\eta}^2_j \, \geq \, 0
\end{aligned}
\tag{3.2}
$$

In order to validate these KKT equations, the MPC problems presented inSection 4.1 are solved using MATLAB's Quadprog and compared with the solution of the set of nonlinear equations shown in Eqn 3.2 using MATLAB's fsolve. These results are presented in Chapter 4.

3.3 Stabilizing CL-DRTO formulation

In the proposed CL-DRTO strategy, both the DRTO and the MPC subproblems are executed in a rolling horizon fashion. For a single CL-DRTO execution, N stabilizing MPC subproblems are computed and each of the stabilizing MPC subproblem has a requirement to reach the set-point in m steps, enforced by the stability constraints. An infinite horizon regulator with end point equality constraint, as shown in Chapter 2 is used. Target tracking with a stabilizing MPC requires an origin shift to the desired set-point. Finally, the linear models used at the DRTO level use the values of these variables in deviation from nominal steady states and nonlinear models at the DRTO level use the values of these variables in its actual form.

Objective Function

At the upper level of the proposed CL-DRTO formulation, various objective functions could be used. Here, we consider a target tracking problem to investigate comparisons to a standalone MPC (thus considering the case where the economic optimum is a steady state value, and computed a priori), and an economic objective function based optimization.

There are several ways to represent an economic cost function, but typically a profit based function can be written as shown below. Here, ϕ_j represents profit per unit time and is averaged over a period of the DRTO prediction horizon.

$$\Phi_{Econ} = \frac{1}{N}\sum_{j=1}^{N} \phi_j^{DRTO}(\hat{x}_j^{DRTO}, \hat{u}_{j-1}^{DRTO}, \hat{y}_j^{DRTO}) \tag{3.3}$$

The proposed CL-DRTO algorithm can also be used to track predetermined targets. This can be achieved using a least squares form of the difference in the plant response

and the specified target. A target tracking objective function used here takes the following form:

$$\Phi_{Target} = \sum_{i=1}^{N} (\| \hat{y}_j^{DRTO} - y_j^{Target} \|)^2 \tag{3.4}$$

where $\|x\|$ represents 2-norm or inner-product.

Set-point Hold Constraint

Due to possible non-uniqueness in solution with use of an economic objective function, the DRTO could potentially exhibit high variation in the output reference trajectory. One judicious way to mitigate this issue is to have set-point hold (SPH) constraints [Jamaludin *et al.*, 2017]. This constraint ensures that the reference trajectory is held constant over a specific number of discretized intervals. In the next Chapter the effect of SPH on the optimization is shown.

Set-point Generation

As discussed in the previous section, at every DRTO prediction step, a subset of the reference trajectories, y^{Ref} and u^{Ref}, is provided to the MPC optimization subproblem as set-points for tracking. For a single DRTO execution the length of y^{Ref} is $N+m-1$ and for each of the N MPC subproblems a subset y_j^{SP} is provided for tracking. The reference trajectory is held constant based on the chosen SPH and hence could change multiple times within a single DRTO execution. The stabilizing MPC has a rolling horizon window, and every MPC subproblem sees a constant set-point as illustrated in the figure below. Hence, the stability constraints direct the optimal input trajectory to ensure that the states reach that constant set-point in m control steps.

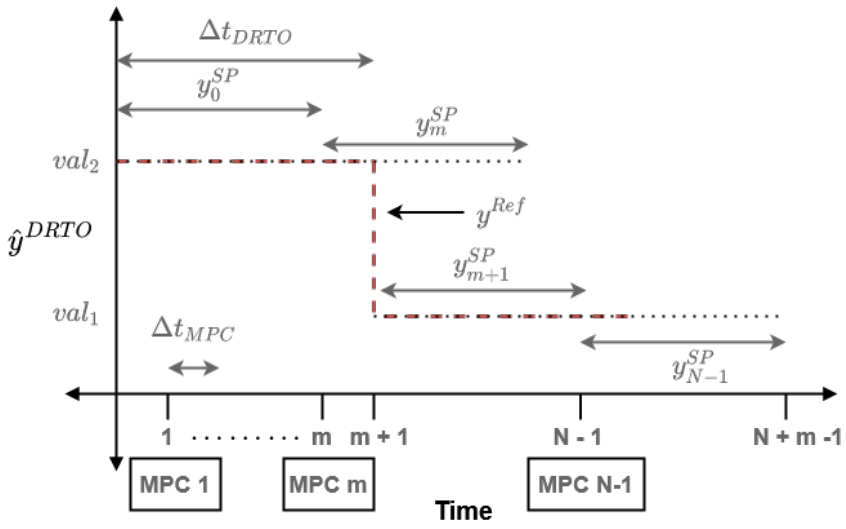

Figure 3.2: Relationship between reference trajectory and the set-point provided to the MPC

In Fig 3.2, the MPC subproblems MPC-(0) to MPC-(m) receive a constant set-point of value val_2, while the MPC subproblems MPC-(m+1) to MPC-(N) receive a constant set-point of value val_1. Therefore, in this approach the stabilizing MPC embedded as subproblems into the DRTO and at the lower level are not aware of any changes in the set-point until the change actually occurs.

Complementarity Constraints

In Eqn 3.2 the equality constraints $\overline{\eta}_j^1 \odot \overline{\mu}_j^1 = 0$ and $\overline{\eta}_j^2 \odot \overline{\mu}_j^2 = 0$ are the complemenentarity constraints. Posing these constraints directly results in MPCC problems that are difficult to solve using standard nonlinear programming (NLP) solvers. Therefore, solving MPCC problems requires alternative strategies; several methods are proposed in the literature which include regularization, a mixed-integer approach and an exact penalty approach (Baker and Swartz [2008], Baumrucker *et al.* [2008], Raghunathan and Biegler [2003]). In this work, an exact penalty approach is used to handle the complementarity constraints [Jamaludin and Swartz, 2017b]. In this approach, the

complementarity constraints are removed and are included as a penalty term in the economic objective function, weighted by a penalty parameter, ρ. The convergence properties of this formulation are presented in Ralph and Wright [2004]. The final formulation of a stabilizing CL-DRTO as a penalty transformed MPCC problem is shown in Eq 3.5.

$$
\begin{aligned}
\underset{y^{Ref},u^{Ref}}{\text{minimize}} \quad & \phi_{Econ}^{DRTO}(\hat{x}^{DRTO},\hat{u}^{DRTO},\hat{y}^{DRTO}) + \rho \sum_{i=1}^{2}\sum_{j=0}^{N-1} \overline{\eta}_j^i \odot \overline{\mu}_j^i \\
\text{subject to} \quad & \hat{x}_{j+1}^{DRTO} = A\hat{x}_j^{DRTO} + B\hat{u}_j^{DRTO} \quad , \quad j = 0,\ldots,N-1 \\
& \hat{y}_j^{DRTO} = C\hat{x}_j^{DRTO} \quad , \quad j = 1,\ldots,N \\
& h^{Ref}(y^{Ref},u^{Ref},y^{SP},u^{SP}) = 0 \\
& y_{min} \le \hat{y}_j^{DRTO} \le y_{max} \\
& \hat{u}_j^{DRTO} = \tilde{u}_{j,0} + u_{j,0}^{SP} \quad , \quad j = 0,\ldots,N-1 \\
& Hu_j^m + \left(G\left(\tilde{x}_{j,0}\right) - F\left(\tilde{u}_{j,-1}\right)\right) - \overline{\eta}_j^1 + \overline{\eta}_j^2 - A_{eq}^T\overline{\lambda}_j = 0 \\
& \tilde{u}_j^m - \overline{\tilde{u}}_{min} - \overline{\mu}_j^1 = 0 \\
& \overline{\tilde{u}}_{max} - \tilde{u}_j^m - \overline{\mu}_j^2 = 0 \\
& \overline{\mu}_j^1 \,,\, \overline{\mu}_j^2 \,,\, \overline{\eta}_j^1 \,,\, \overline{\eta}_j^2 \;\ge\; 0 \\
& A_{eq}\tilde{u}_j^m = -\tilde{V}_u A^m(\tilde{x}_{j,0})
\end{aligned}
\tag{3.5}
$$

where, $A_{eq} = \tilde{V}_u\left[A^{m-1}B, A^{m-2}B, \ldots, B\right]$ and $\tilde{u}_{j,0}$ represents the first control step implemented by the inner MPC subproblems at every time step j.

Computation

MATLAB is used as the primary software environment and the DRTO optimization is implemented in AMPL. All the required parameters are defined in MATLAB and are exported to AMPL to perform the CL-DRTO optimization. The optimal set-

point trajectories computed by solving the CL-DRTO problem are in turn exported to MATLAB, where the standalone MPC implementation is used to compute the control action. The sampling time of the MPC is Δt_{MPC}, while the DRTO is executed once every Δt_{DRTO}. It should be noted that the Δt_{MPC} is often lower than Δt_{DRTO}. The plant simulation based on the dynamic model is performed in MATLAB for a duration of Δt_{DRTO} using the optimal input trajectories calculated every Δt_{MPC} by the lower-level MPC. The primary optimization coded in AMPL is solved using IPOPT (version 3.12.8), and the DRTO calculations and plant simulation are carried out on a PC with 16 GB RAM and Intel(R) Core(TM) i7 - 3770 CPU @ 3.40 GHz processor.

Remark 5. The KKT optimality conditions here are derived for the stabilizing MPC in its standard Quadratic Program (QP) form as shown in Eqn 2.15. Although this results in a problem of smaller dimension, it involves calculation of matrix powers, A^i, and hence for problems with a large number of states or large prediction horizons, this problem could result in numerical issues. Since the case studies presented in this paper involve a maximum of two states and a MPC prediction horizon of five, we adhere to the compact QP problem. The KKT optimality conditions and the stabilizing CL-DRTO formulation for the MPC in its direct form as shown in Eqn 2.9 is presented in the next section.

3.4 Alternate KKT formulation

KKT optimality conditions of the direct form of the stabilizing MPC is shown here. Following a similar approach to Section 3.2, the Lagrangian of the stabilizing MPC shown in Eqn 2.9 for every time step j is shown below.

$$
\begin{aligned}
L_j \;=\; & (\tilde{x}_{j,m})^T \overline{Q}(\tilde{x}_{j,m}) \;+\; \Delta\tilde{u}_{j,m}^T S \Delta\tilde{u}_{j,m} \;+\; \sum_{k=0}^{m-1} (\tilde{x}_{j,k})^T Q(\tilde{x}_{j,k}) \;+\; (\tilde{u}_{j,k})^T R(\tilde{u}_{j,k}) + \Delta\tilde{u}_{j,k}^T S \Delta\tilde{u}_{j,k} \\
& - \sum_{k=0}^{m-1} (\lambda_{j,k+1}^1)(\tilde{x}_{j,k+1} - A\tilde{x}_{j,k} - B\tilde{u}_{j,k}) \;-\; \sum_{k=0}^{m} (\lambda_{j,k}^2)(\tilde{u}_{j,k} - \tilde{u}_{j,k-1} - \Delta u_{j,k}) \;- \\
& + \sum_{k=0}^{m-1} (\eta_{j,k}^1)(\tilde{u}_{j,k} - \tilde{u}_{min}) \;-\; \sum_{k=0}^{m-1} (\eta_{j,k}^2)(\tilde{u}_{max} - \tilde{u}_{j,k}) \;-\; (\lambda_{j,k+1}^3)(\tilde{V_u}\tilde{x}_{j,m})
\end{aligned}
$$

$\eta_{j,k}^i$ for $i = 1,\ 2$ are the Lagrange multipliers for the inequality input constraints and $\lambda_{j,k}^i$ for $i = 1,\ 2,\ 3$ are the Lagrange multipliers for the state space equation, $\Delta\tilde{u}_{j,k}$ equation and the equality endpoint stability constraints respectively. The KKT optimality equations are:

1. Lagrange gradients - $\nabla L_j = 0$
   - $2C^T Q C(\tilde{x}_{j,k}) + A^T \lambda_{j,k+1}^1 - \lambda_{j,k}^1 = 0 \quad , \quad \text{k = 1,2,...,m-1}$
   - $2\overline{Q}(\tilde{x}_{j,k}) - \lambda_{j,k}^1 - \tilde{V_u}^T \lambda_{j,k}^3 = 0 \quad , \quad \text{k = m}$
   - $2S\Delta\tilde{u}_{j,k} + \lambda_{j,k}^2 = 0 \quad , \quad \text{k = 0,1,...,m}$
   - $2R(\tilde{u}_{j,k}) + B^T \lambda_{j,k}^1 - \lambda_{j,k}^2 + \lambda_{j,k+1}^2 - \eta_{j,k}^1 + \eta_{j,k}^2 = 0 \quad , \quad \text{k = 0,1,...,m-1}$
2. Primal feasibility: Input inequality constraints transformed into equalities using slack variable $\mu_{j,k}^i$, $i = 1, 2$
   - $\tilde{u}_{j,k} - \tilde{u}_{min} - \mu_{j,k}^1 = 0 \quad , \quad \text{k = 0,1,...,m-1}$
   - $\tilde{u}_{max} - \tilde{u}_{j,k} - \mu_{j,k}^2 = 0 \quad , \quad \text{k = 0,1,...,m-1}$
   - $\mu_{j,k}^1 \,,\ \mu_{j,k}^2 \geq 0 \quad , \quad \text{k = 0,1,...,m-1}$
3. Dual feasibility:
   - $\eta_{j,k}^1 \,,\ \eta_{j,k}^2 \geq 0 \quad , \quad \text{k = 0,1,...,m-1}$

4. Complementarity constraints:

   - $\overline{\eta}_j^i \odot \overline{\mu}_j^i = 0 \quad , \quad \text{i} = 1,2$

$\overline{\mu}_j^i$ and $\overline{\nu}_j^i$ represent composite vectors of $\mu_{j,k}^i$ and $\nu_{j,k}^i$ respectively for $i = 1, 2$.

Optimization formulation

Embedding these KKT optimality conditions as constraints in the DRTO optimization and using the exact penalty approach results in the following optimization.

$$
\begin{aligned}
\underset{y^{Ref},u^{Ref}}{\text{minimize}} \quad & \phi_{Econ}^{DRTO}(\hat{x}^{DRTO},\hat{u}^{DRTO},\hat{y}^{DRTO}) + \rho \sum_{i=1}^{2}\sum_{j=0}^{N-1} \overline{\eta}_j^i \odot \overline{\mu}_j^i \\
\text{subject to} \quad & \hat{x}_{j+1}^{DRTO} = A\hat{x}_j^{DRTO} + B\hat{u}_j^{DRTO} \quad , \quad j = 0,\ldots,N-1 \\
& \hat{y}_j^{DRTO} = C\hat{x}_j^{DRTO} \quad , \quad j = 1,\ldots,N \\
& h^{Ref}(y^{Ref},u^{Ref},y^{SP},u^{SP}) = 0 \\
& y_{min} \le \hat{y}_j^{DRTO} \le y_{max} \\
& \hat{u}_j^{DRTO} = \tilde{u}_{j,0} + u_{j,0}^{SP} \quad , \quad j = 0,\ldots,N-1 \\
& 2C^TQC(\tilde{x}_{j,k}) + A^T\lambda_{j,k+1}^1 - \lambda_{j,k}^1 = 0 \ , \ k = 1,2,\ldots,m-1 \\
& 2\overline{Q}(\tilde{x}_{j,k}) - \lambda_{j,k}^1 - \tilde{V}_u{}^T\lambda_{j,k}^3 = 0 \ , \ k = m \\
& 2S\Delta\tilde{u}_{j,k} + \lambda_{j,k}^2 = 0 \ , \ k = 0,1,\ldots,m \\
& 2R(\tilde{u}_{j,k}) + B^T\lambda_{j,k}^1 - \lambda_{j,k}^2 + \lambda_{j,k+1}^2 - \eta_{j,k}^1 + \eta_{j,k}^2 = 0 \ , \ k = 0,1,\ldots,m-1 \\
& \tilde{u}_{j,k} - \tilde{u}_{min} - \mu_{j,k}^1 = 0 \ , \ k = 0,1,\ldots,m-1 \\
& \tilde{u}_{max} - \tilde{u}_{j,k} - \mu_{j,k}^2 = 0 \ , \ k = 0,1,\ldots,m-1 \\
& \overline{\eta}_j^i\overline{\mu}_j^i = 0 \ , \ i = 1,2 \\
& \overline{\mu}_j^1 \ , \ \overline{\mu}_j^2 \ , \ \overline{\eta}_j^1 \ , \ \overline{\eta}_j^2 \ \ge \ 0 \\
& \tilde{V}_u\tilde{x}_{j,m} = 0
\end{aligned}
\tag{3.6}
$$

3.5 Inclusion of nonlinearity

So far we have considered a linear dynamic model at the DRTO level and for the lower level MPC. In this section, we introduce nonlinear models at the DRTO level to capture the plant behaviour with better accuracy. Consider a continuous time nonlinear system of ordinary differential equations (ODEs) that model the behaviour of the process plant as shown below:

$$\begin{aligned} \dot{x}^{DRTO}(t) &= f_C^{DRTO}(x^{DRTO}(t), u^{DRTO}(t)) \\ y^{DRTO}(t) &= h_C^{DRTO}(x^{DRTO}(t)) \\ x^{DRTO}(0) &= x_0 \end{aligned} \tag{3.7}$$

where f_C^{DRTO} represents the continuous time nonlinear model of the plant, h_C^{DRTO} is a nonlinear function that relates states to the outputs, $x^{DRTO}(t)$ and $u^{DRTO}(t)$ represent the continuous time state and input vector respectively and $x^{DRTO}(0)$ is the initial state vector. This model can be descritized using a scheme such as a backward Euler approximation, orthogonal collocation of finite elements or multiple shooting. In our work, implicit Euler approximation is used with a finite element interval of Δt_{MPC}, the sampling time of the MPC as shown in Eqn 3.8.

$$\begin{aligned} \frac{x_j^{DRTO} - x_{j-1}^{DRTO}}{\Delta t_{MPC}} &= f_C^{DRTO}(x_j^{DRTO}, u_{j-1}^{DRTO}) \\ y_j^{DRTO} &= h_C^{DRTO}(x_j^{DRTO}) \end{aligned} \tag{3.8}$$

Optimization formulation

In Eqn 3.5, we replace the linear dynamic model with a nonlinear dynamic model as shown in Eqn 3.9.

$$
\begin{aligned}
\underset{y^{Ref},u^{Ref}}{\text{minimize}} \quad & \phi_{Econ}^{DRTO}(\hat{x}^{DRTO},\hat{u}^{DRTO},\hat{y}^{DRTO}) + \rho \sum_{i=1}^{2}\sum_{j=0}^{N-1} \overline{\eta}_j^i \odot \overline{\mu}_j^i \\
\text{subject to} \quad & \frac{\hat{x}_j^{DRTO} - \hat{x}_{j-1}^{DRTO}}{\Delta t_{MPC}} = f_C^{DRTO}(\hat{x}_j^{DRTO},\hat{u}_{j-1}^{DRTO}) \quad , \quad j = 1,\ldots,N \\
& \hat{y}_j^{DRTO} = h_C^{DRTO}(\hat{x}_j^{DRTO}) \quad , \quad j = 1,\ldots,N \\
& h^{Ref}(y^{Ref},u^{Ref},y^{SP},u^{SP}) = 0 \\
& y_{min} \leq \hat{y}_j^{DRTO} \leq y_{max} \\
& \hat{u}_j^{DRTO} = \tilde{u}_{j,0} + u_{j,0}^{SP} \quad , \quad j = 0,\ldots,N-1 \\
& Hu_j^m + \left(G\left(\tilde{x}_{j,0}\right) - F\left(\tilde{u}_{j,-1}\right)\right) - \overline{\eta}_j^1 + \overline{\eta}_j^2 - A_{eq}^T \overline{\lambda}_j = 0 \\
& \tilde{u}_j^m - \overline{\tilde{u}}_{min} - \overline{\mu}_j^1 = 0 \\
& \overline{\tilde{u}}_{max} - \tilde{u}_j^m - \overline{\mu}_j^2 = 0 \\
& \overline{\mu}_j^1 \, , \, \overline{\mu}_j^2 \, , \, \overline{\eta}_j^1 \, , \, \overline{\eta}_j^2 \; \geq \; 0 \\
& A_{eq}\tilde{u}_j^m = -\tilde{V}_u A^m(\tilde{x}_{j,0})
\end{aligned}
\tag{3.9}
$$

Chapter 4

Case Studies

In this chapter we present two case studies to illustrate the efficacy of the proposed approach; a single-input single-output (SISO) CSTR system and a multi-input multi-output (MIMO) CSTR system. In both these case studies, the model at the DRTO level is consistent with the model used as the plant. For instance, if a nonlinear model is used at the DRTO level, the plant is also based on a set of nonlinear differential equations.

The case studies presented in this chapter have been included in the manuscript submitted to:

Sundaresan Ramesh, P., Swartz, C. L. E., and Mhaskar, P. (2020) "Closed-loop Dynamic Real-Time Optimization with Stabilizing Model Predictive Control", *Submitted to AICHE Journal*, in review.

4.1 Case Study 1: Single-Input Single-Output (SISO) CSTR system

In order to demonstrate the proposed framework, a nonisothermal, nonadiabatic, CSTR is considered. A first-order exothermic reaction A $\longrightarrow$ B is carried out in the CSTR [Uppal *et al.*, 1974]. The nonlinear dynamic system can be represented as shown in Eqn 4.1, in which χ_1 represents the dimensionless conversion, χ_2 represents the dimensionless reactor temperature and β, Da, χ_{2c} and B represent the heat transfer coefficient, Damköhler number, heat transfer medium temperature and heat of reaction respectively. The parameter values $Da = 0.05$, $B = 8.0$, $\chi_{2c} = 0$ are taken from Patwardhan *et al.* [1989].

$$\begin{aligned} \frac{\mathrm{d}\chi_1}{\mathrm{d}t} &= -\chi_1 + Da(1-\chi_1)e^{\chi_2} \\ \frac{\mathrm{d}\chi_2}{\mathrm{d}t} &= -\chi_2 + Da(1-\chi_1)Be^{\chi_2} - \beta(\chi_2 - \chi_{2c}) \end{aligned} \tag{4.1}$$

This nonlinear model is linearized around the unstable steady state, $x_{ss} = (0.5011, 3)$ and $u_{ss} = 0.3362$ and the obtained linear state space model is shown in Eqn 4.2;

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \end{aligned} \tag{4.2}$$

where x represents the states χ_1 and χ_2, and $u = \beta$ is the manipulated input variable. Matrices A, B and C for a sampling time of 0.1 used in this study are shown in Eqn 4.3. The dimensionless reactor temperature of this system is the controlled output.

$$A = \begin{bmatrix} 0.7991 & 0.0519 \\ -0.8327 & 1.2838 \end{bmatrix},\ B = \begin{bmatrix} -0.0077 \\ -0.3417 \end{bmatrix},\ C = \begin{bmatrix} 0 & 1 \end{bmatrix} \tag{4.3}$$

The objective of this case study is to apply the proposed CL-DRTO strategy to maximize the conversion. With this goal, a model predictive controller with stability constraints is demonstrated and is embedded within the DRTO optimization as constraints. This approach is implemented first as a target tracking problem before moving on to the economic objective of maximizing the conversion of the reactant.

Model Predictive Control

A model predictive controller as shown in Section 2.6, based on the tuning parameters: $Q = 1, S = 1000, R = 0$ is used on the linearized system. It should be noted that the states and the inputs are in deviation form, as deviations from steady-state x_{ss} and u_{ss} respectively.

The importance of stability constraints is shown in Fig 4.1 and Fig 4.2. The MPC here is required to track a constant set-point target of three units above the steady state. Fig 4.1 shows the response when no stability constraints are used, and it can be seen that the output of the system evolves exponentially, while in Fig 4.2 an endpoint equality constraint is used and the controller is able to track the target. Because, the initial state is an unstable steady state, an end-point quality constraint is necessary for stable operation.

Within the CL-DRTO, as discussed in Section 3.2, the MPC subproblems are replaced by their equivalent KKT conditions. Fig 4.3 compares the solution found by solving the MPC problem as a quadratic program to the optimal solution found by solving a set of nonlinear KKT equations. Since, this MPC formulation is a convex formulation, uniqueness of solution is guaranteed. It can be seen from the figure that the KKT conditions are able to replicate the results of the quadratic program, where a step change is introduced.

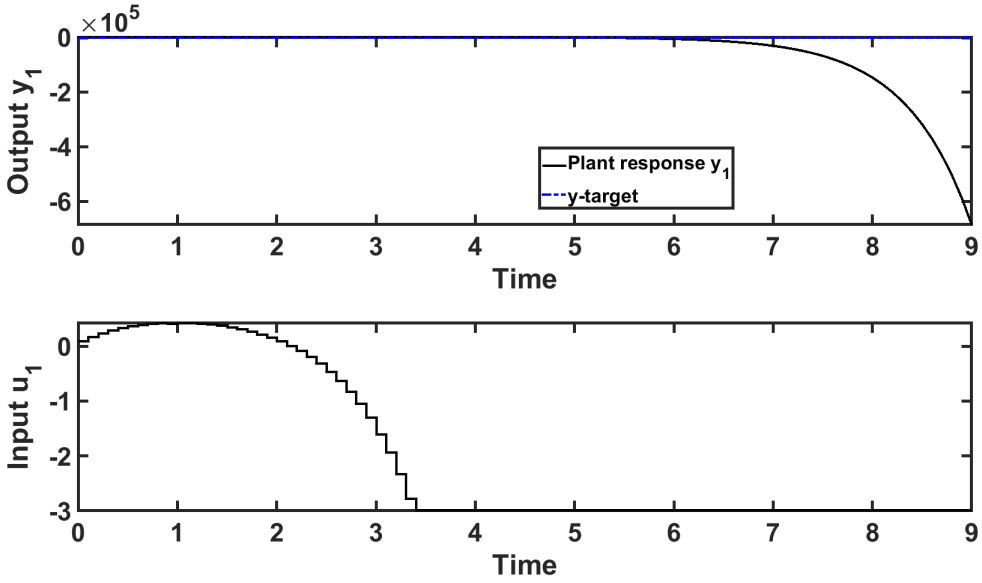

Figure 4.1: Target tracking without the end-point equality constraint

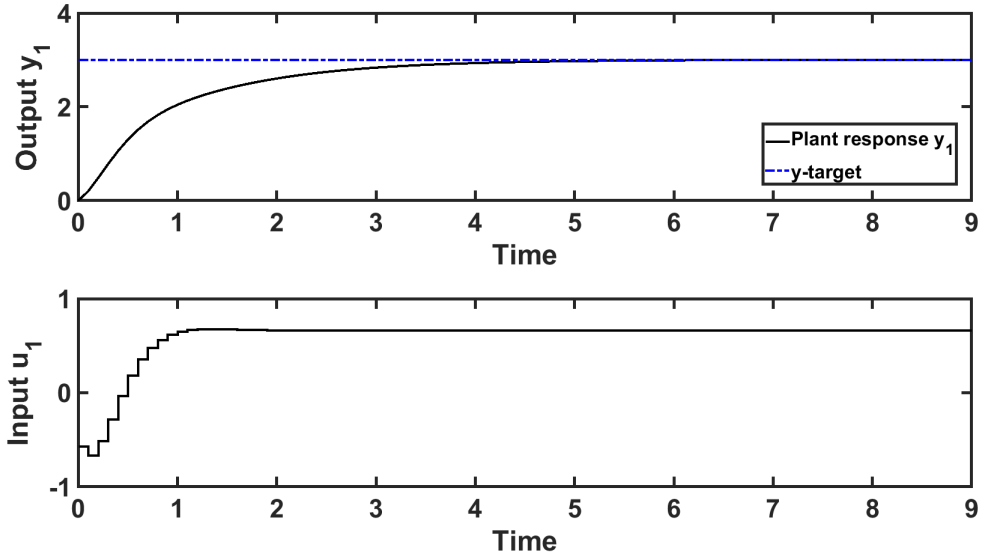

Figure 4.2: Target tracking with the end-point equality constraint

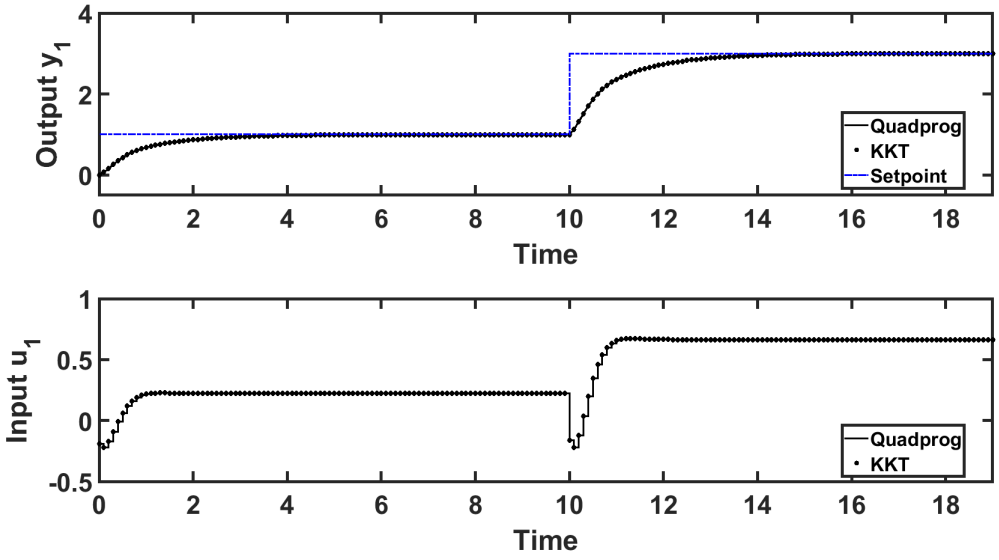

Figure 4.3: Comparison of Quadprog and KKT solution

CL-DRTO: Target Tracking Objective

As discussed in Chapter 3, sometimes the most economic operating condition could be known a priori, resulting in a target-tracking problem. Although a target-tracking objective function can readily be handled by a simple MPC formulation, the proposed strategy is used and is compared to a regular MPC trajectory. The same SISO state space system shown in Eqn 4.2 is used. The design parameters used for the CL-DRTO simulation are shown in Table 4.1. It should be noted that the DRTO executions are usually performed at a lower frequency in comparison to the MPC executions, but in this case study, the DRTO and the MPC operate at the same frequency to illustrate the operation and potential of the proposed strategy. In the next MIMO case study, the effect of set-point hold (SPH) on the system response is demonstrated.

Parameter	**Description**	**Value**
T_{Sim}	Total simulation time	8
Δt_{DRTO}	DRTO Sample time	0.1
Δt_{MPC}	MPC Sample time	0.1
N	DRTO Prediction horizon	10
M	MPC Prediction horizon	5
Q	Output penalty	1
R	Input penalty	0
S	Change in input penalty	1000
u_{min}	Minimum value of u	-3
u_{max}	Maximum value of u	15

Table 4.1: Design parameters for the DRTO and MPC layers in target tracking case study

In Fig 4.4 the proposed CL-DRTO for a target-tracking objective function is shown. In this example, the economically optimal solution is assumed to be three units above the steady-state value. Therefore, a constant target of three is given to the proposed CL-DRTO. For this constant target of three, the set-points provided to the lower-level MPC by the CL-DRTO strategy varies from around -5 to 12. This additional

flexibility from the provided set-points results in an improved performance over a regulatory MPC. The performance of an infinite horizon MPC and the proposed CL-DRTO for this target-tracking objective function is shown in Fig 4.5. It can be seen that the proposed DRTO approach is much quicker to get to the target when compared to the MPC. Note of course, that a differently tuned standard MPC may also be able to achieve this behavior or better performance. This is illustrated by the dash-dotted lines showing the implementation of an MPC with a very low penalty value on change in inputs, 0.001 instead of 1000 and a longer horizon of 20 instead of 5.

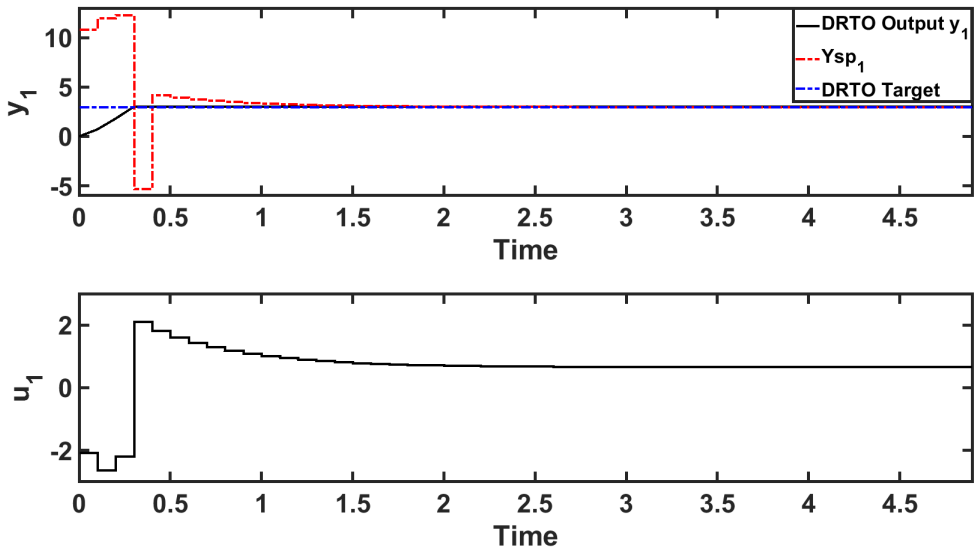

Figure 4.4: CL-DRTO strategy for target tracking

Remark 6. The presence of the embedded MPC subproblems is useful to predict the future interactions between the DRTO process model and the controller, which can be exploited for the plant to be able to reach the target much more quickly than a regular MPC, without changing the MPC tuning parameters. For a given target, the MPC is provided with a constant set-point corresponding to the value of the target with the perfomance dependent on the choice of the tuning parameters. In the CL-DRTO, on the other hand, this tuning is replaced by the costs in the objective

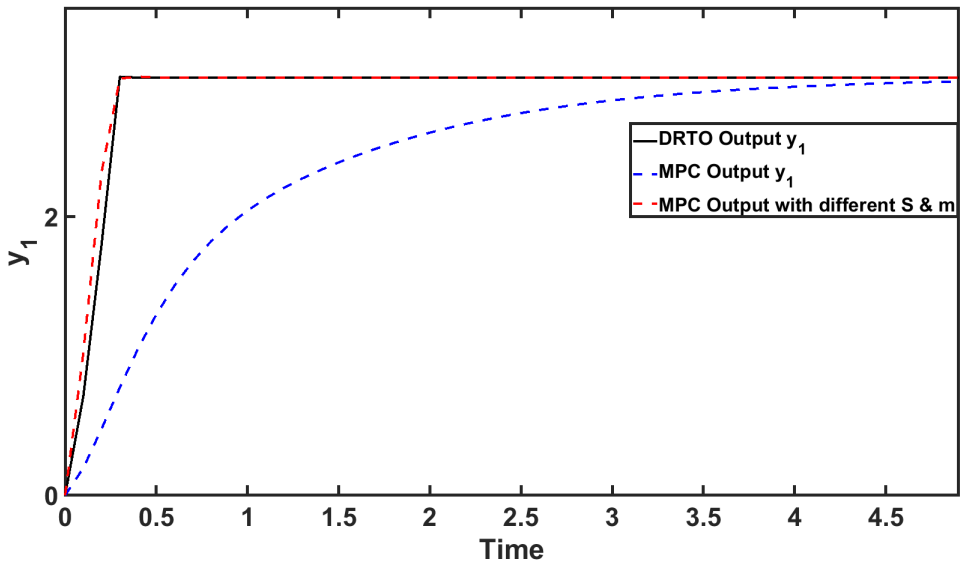

Figure 4.5: Comparison of the CL-DRTO and MPC performance for target tracking

function (for instance, penalizing only the target tracking error), which in turn makes the CL-DRTO utilize the set-points provided to the MPC to achieve quicker target tracking while respecting the output constraints. The stable CL-DRTO approach proposed in our work ensures that this additional flexibility of the variable set-points can be used around unstable steady states as shown in Fig 4.4. The performance of the stand-alone MPC can be attributed to the high penalty on the change in inputs, Δu, and relaxing this penalty matrix S, resulted in a slightly better solution than the CL-DRTO. The proposed approach is able to achieve better results even with a high penalty on Δu. In some sense one of the benefits of the CL-DRTO is in its ability to 'override' the underlying MPC tuning parameter values to move the process in a fashion that is economically optimal, without necessarily having to change the MPC implementation at the lower level.

CL-DRTO: Economic Objective

In this section, the ability of the CL-DRTO to achieve an economic optimum is demonstrated. The economic objective considered here is to maximize the conversion of the reactant. Note that this is not the controlled variable in the lower level MPC. The design parameters used for the CL-DRTO simulation are shown in Table 4.2. For this problem, a set-point hold of 10 is used, thus the reference trajectory is held constant over the entire DRTO sample period.

Parameter	**Description**	**Value**
T_{Sim}	Total simulation time	9
Δt_{DRTO}	DRTO Sample time	1
Δt_{MPC}	MPC Sample time	0.1
N	DRTO Prediction horizon	20
M	MPC Prediction horizon	5
Q	Output penalty	1
R	Input penalty	0
S	Change in input penalty	1000
u_{min}	Minimum value of u	-3
u_{max}	Maximum value of u	15
χ_1^{min}	Minimum value of conv.	0
χ_1^{max}	Maximum value of conv.	0.99

Table 4.2: Design parameters for the DRTO and MPC layers in economic case study

A maximum value of 0.99 for conversion is used to avoid any numerical errors resulting in a conversion greater than one. Since, the objective here is to maximize the conversion (hidden state) and there is an upper limit, the optimal solution is the maximum value. Like in the target tracking case, it can be seen in Fig 4.6 the capability of the proposed approach to use the additional flexibility from the set-points provided to the MPC to achieve better results without violating the output constraints.

Thus far, a linear discretized model of the dynamic system has been used in both the MPC and the DRTO. One of the key advantages of using a CL-DRTO is to be

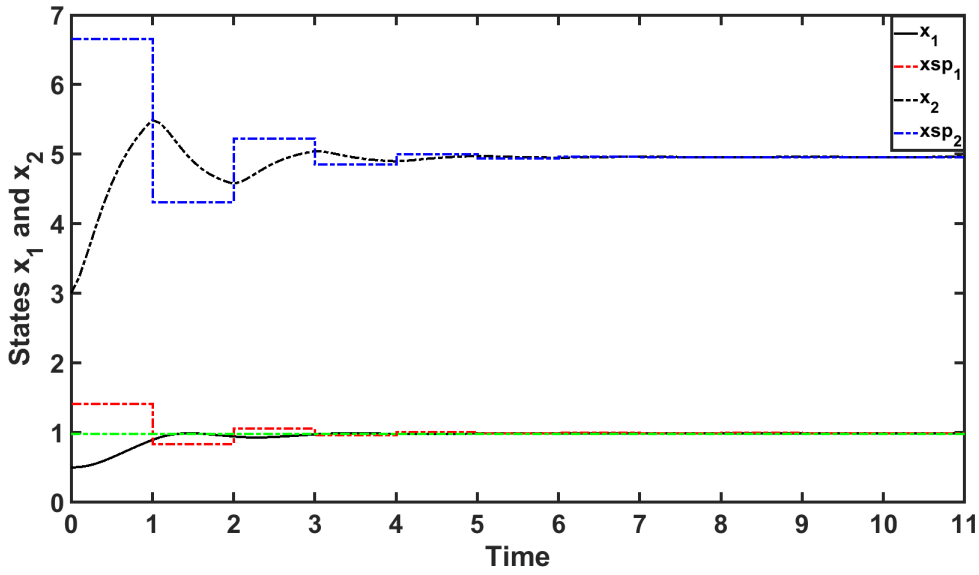

Figure 4.6: CL-DRTO strategy for maximizing conversion

able to handle a difference in the model between DRTO and MPC. In this study, the nonlinear model of the dynamic system is incorporated into the primary optimization by reducing the differential equations into nonlinear difference equations using the Backward Euler method. Fig 4.7 compares the performance of the CL-DRTO strategy with a linear and a nonlinear model at the DRTO level. The solid line represents the performance of the CL-DRTO strategy with a nonlinear model while the dashed line shows the performance with a linearized DRTO model. In the former case, a set of nonlinear differential equations is used as the plant and in the latter case, a linearized model is used as the plant.

Inclusion of nonlinear model at the DRTO level, resulted in a quicker performance (and hence better economic performance) even though the set-points do not move as much as in the linear case. Fig 4.8 attempts to explain this behaviour by comparing the stand-alone MPC trajectories for a linear and a nonlinear plant.

When a nonlinear model is used as the plant, an overshoot can be observed unlike

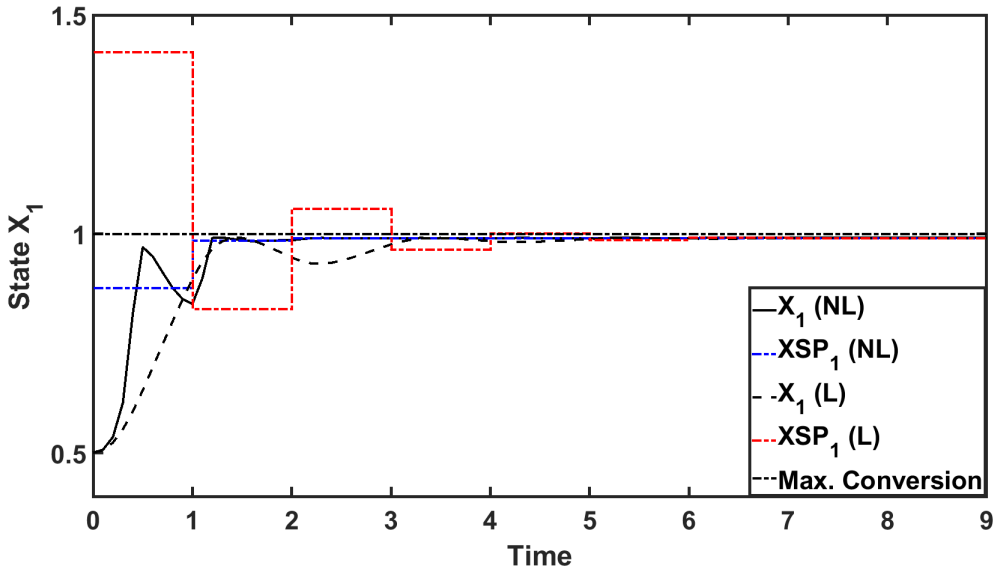

Figure 4.7: Comparison of state evolution for nonlinear (NL) DRTO model and linearized (L) DRTO model

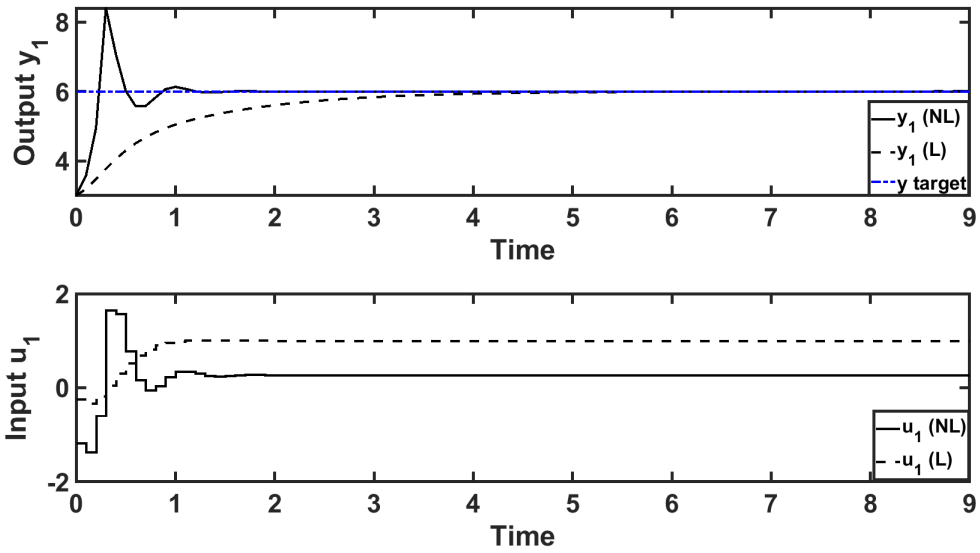

Figure 4.8: Comparison of linear (L) vs nonlinear (NL) plant with a linear model based MPC

with the linear plant. The proposed strategy therfore provides a set-point that does not lead to output constraint violation. This can be attributed to the presence of the control optimization subproblems, thereby, predicting the future interactions between the DRTO process model and the controller.

4.2 Case Study 2: Multi-Input Multi-Output (MIMO) CSTR system

Previously in Section 4.1, the proposed approach was applied to a SISO CSTR case study. In this section, the proposed strategy is demonstrated for MIMO systems. For this case study, all the states are assumed to be measured and hence the term states and outputs are used interchangeably. This case study is taken from Heidarinejad *et al.* [2012].

Consider a second-order, irreversible, exothermic reaction A $\longrightarrow$ B carried out in a well-mixed CSTR. The feed consists of pure A, the reactant, at a flow rate F, inlet temperature T_0 and inlet concentration C_{A0}. Heat is provided or removed from the system with the help of a jacket. This dynamic system can be represented as shown in Eqn 4.4.

$$\begin{aligned} \frac{\mathrm{d}C_A}{\mathrm{d}t} &= \frac{F}{V}(C_{A0} - C_A) - k_0 e^{\frac{-E}{RT}} C_A^2 \\ \frac{\mathrm{d}T}{\mathrm{d}t} &= \frac{F}{V}(T_0 - T) - \frac{\Delta H}{\sigma C_p} k_0 e^{\frac{-E}{RT}} C_A^2 + \frac{Q}{\sigma C_p V} \end{aligned} \tag{4.4}$$

Here, C_A represents the concentration of the reactant, T represents the reactor temperature, V denotes the volume of the CSTR, Q denotes the heat input or removal rate, C_p and σ denote the heat capacity and the density of the fluid in the reactor respectively and ΔH, k_0 and E represent the enthalpy, exponential constant and the

activation energy of the reaction respectively. This system is discretized to a linear state space model with a sampling time of 0.1 units. In this model, the concentration and temperature of the reactant are the states $x_1 = C_A$ $x_2 = T$ while the inlet concentration and heat input/removal are the manipulated variables $u_1 = C_{A0}$ and $u_2 = Q$. The process parameter values used in this case study are shown in Table 4.3.

$T_0 = 300$	K	$F = 5$	$\frac{m^3}{hr}$
$V = 1$	m^3	$E = 5$ x 10^4	$\frac{kJ}{kmol}$
$k_0 = 8.46$ x 10^6	$\frac{1}{hr}$	$\Delta H = -1.15$ x 10^4	$\frac{kJ}{kmol}$
$C_p = 0.231$	$\frac{kJ}{kgK}$	$R = 8.314$	$\frac{kJ}{kmolK}$
$\sigma = 1000$	$\frac{kg}{m^3}$	$C_{As} = 1.9537$	$\frac{kmol}{m^3}$
$T_s = 401.87$	K	$C_{A0s} = 4$	$\frac{kmol}{m^3}$
$Q_s = 0$	$\frac{kJ}{hr}$		

Table 4.3: Parameter values for the dynamic MIMO CSTR system

The inputs are constrained as follows: 0.5 $kmol/m^3 \leq u_1 \leq 14$ $kmol/m^3$ and -5 x 10^5 $kJ/hr \leq u_2 \leq 5$ x 10^5 kJ/hr. The system has several steady states, but the unstable steady state under consideration is $x_{ss} = (1.9537, 401.87)$ and corresponding $u_{ss} = (4,0)$. The steady states of the system are denoted by (C_{As}, T_s) and the steady state input values by (C_{A0s}, Q_s). The state constraints imposed are as follows: 0.4537 $kmol/m^3 \leq x_1 \leq 9.9537$ $kmol/m^3$ and 396.87 $K \leq x_2 \leq 416.87$ K. Hence, the plant is operated in a region in and around the unstable steady state.

The objective of this case study is to apply the proposed CL-DRTO strategy to maximize the production rate. With this goal, a model predictive controller is implemented for a target tracking case and is embedded within the DRTO optimization as constraints. This approach is implemented first as a target tracking problem before moving on to the economic objective.

Model Predictive Control

A model predictive control based on the tuning parameters: $Q = 1, S = 1000, R = 0$ for a target of [3.9537; 405.87] is shown below. As discussed in Chapter 3, an infinite horizon regulator with no state constraints is implemented as the underlying controller. These state constraints are rather present in the primary DRTO optimization as hard constraints. Therefore, for certain targets the unconstrained MPC might lead to constraint violation. For instance, a target of [3.9537; 405.87] constraint violation occurs as shown in Fig 4.9

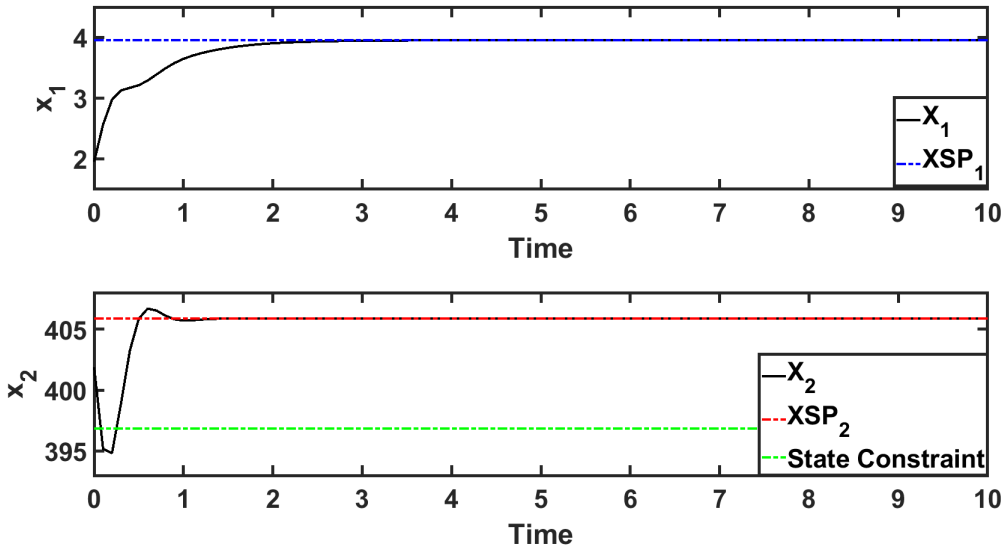

Figure 4.9: Regulatory MPC target tracking

CL-DRTO: Target Tracking Objective

The proposed CL-DRTO approach is implemented on this target tracking problem using the design parameters shown in Table 4.4. A DRTO sampling time of 2 and an MPC sampling time of 0.1 is used. Hence, there are 20 MPC computations within a single DRTO execution.

Parameter	Description	Value
T_{Sim}	Total simulation time	20
Δt_{DRTO}	DRTO Sample time	2
Δt_{MPC}	MPC Sample time	0.1
N	DRTO Prediction horizon	40
M	MPC Prediction horizon	5
Q	Output penalty	diag(1,1)
R	Input penalty	diag(0,0)
S	Change in input penalty	diag(1000,1000)

Table 4.4: Design parameters for the DRTO and MPC layers in target tracking MIMO case study

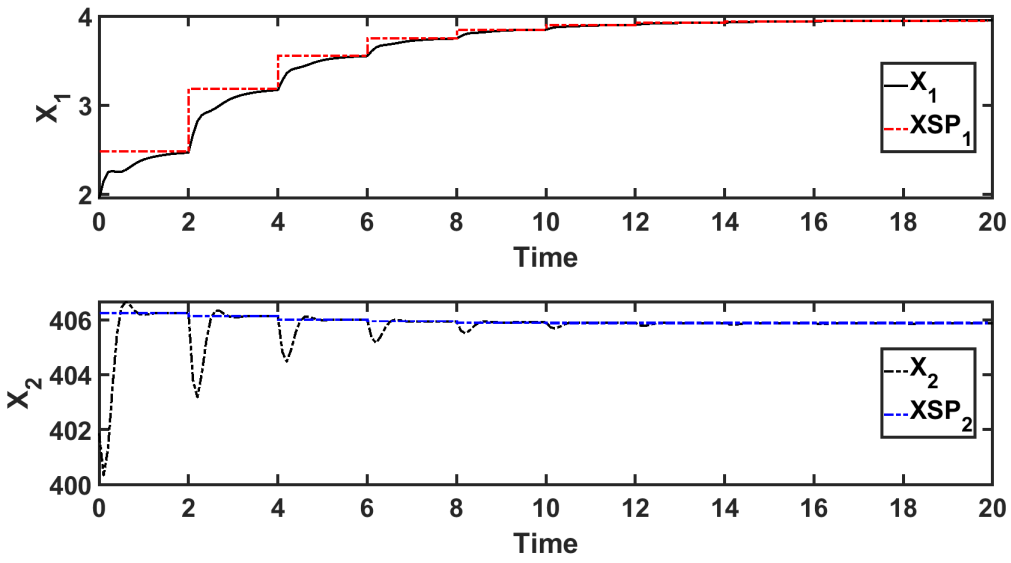

Figure 4.10: CL-DRTO strategy for target tracking with no input scaling

Fig 4.10 shows the performance of the proposed CL-DRTO strategy for target tracking. Although, the performance is better than MPC in terms of cost (target tracking cost), when this strategy was applied to an economic objective function, it resulted in optimization issues. We observed the inputs of the system to be poorly scaled due to a huge difference in their range. Therefore, to avoid numerical issues, u_2 is scaled to kJ/s and Eqn 4.4 is modified accordingly. From here on the MIMO case study

results are presented for the scaled system as shown below.

$$\begin{aligned} \frac{\mathrm{d}C_A}{\mathrm{d}t} &= \frac{F}{V}(C_{A0} - C_A) - k_0 e^{\frac{-E}{RT}} C_A^2 \\ \frac{\mathrm{d}T}{\mathrm{d}t} &= \frac{F}{V}(T_0 - T) - \frac{\Delta H}{\sigma C_p} k_0 e^{\frac{-E}{RT}} C_A^2 + 3600 * \frac{Q}{\sigma C_p V} \end{aligned} \tag{4.5}$$

Fig 4.11 shows the CL-DRTO strategy for the same target of [2;4] above the steady-state value. The hard state constraints present in the primary DRTO optimization result in no constraint violation and the presence of stability constraints ensures stable plant operation around the unstable steady state. The set-points provided by the DRTO are allowed to vary every two MPC sampling units and this corresponds to a set-point hold (SPH) of 2.

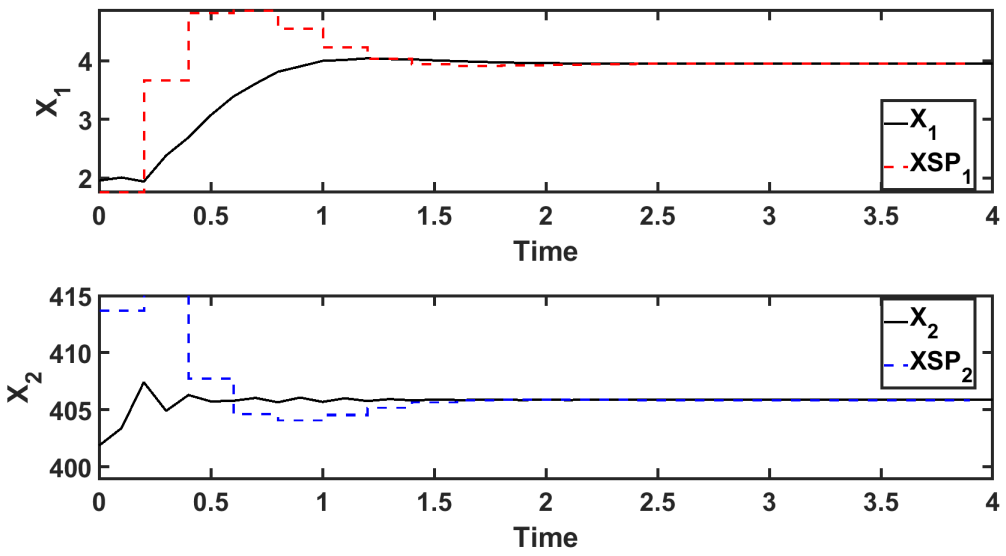

Figure 4.11: CL-DRTO strategy for target tracking with input scaling

The objective is to minimize the error between the target and the actual plant response in a least squares fashion and hence better performance is obtained from a strategy that yields lower value of the objective function. Since the unconstrained MPC as shown in Fig 4.9 violates the state constraints, the comparison is made be-

tween the proposed CL-DRTO and an output constrained MPC. Fig 4.12 compares a constrained MPC and the proposed CL-DRTO strategy. The output constrained MPC resulted in an objective function value of 181.9, while the proposed CL-DRTO strategy resulted in a value of 43.0. Hence, the CL-DRTO performs better than the output constrained MPC for the same set of tuning parameters. Once again, it should be noted that a standalone MPC with a different set of penalty matrices and prediction horizon could result in improved performance.

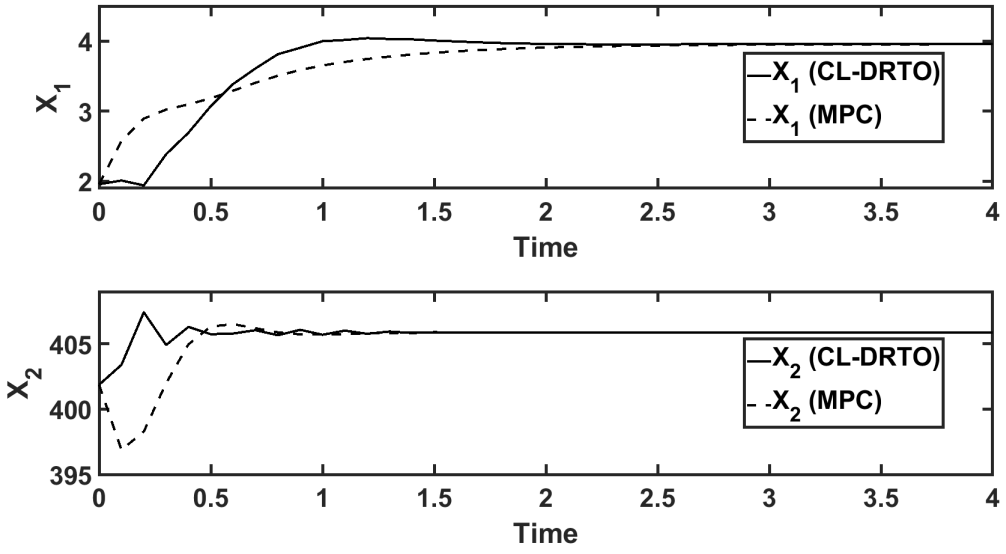

Figure 4.12: CL-DRTO vs constrained MPC for target tracking

Remark 7. It should be noted that for a fixed given value of the MPC tuning parameters, the performance of the proposed CL-DRTO strategy is equal to or better than the output constrained MPC. Also note that, set-point hold (SPH) controls the additional flexibility of the varying set-points provided by the DRTO. The lower the SPH, the higher the variation in DRTO set-points within a single computation and the better the performance; an almost linear trend can be observed between SPH and the target tracking cost in Fig 4.18.

Fig 4.13 - Fig 4.17 shows the performance for varying SPH values, varying from 2 to

20 and Fig 4.18 shows the objective function values for each of these SPH values.

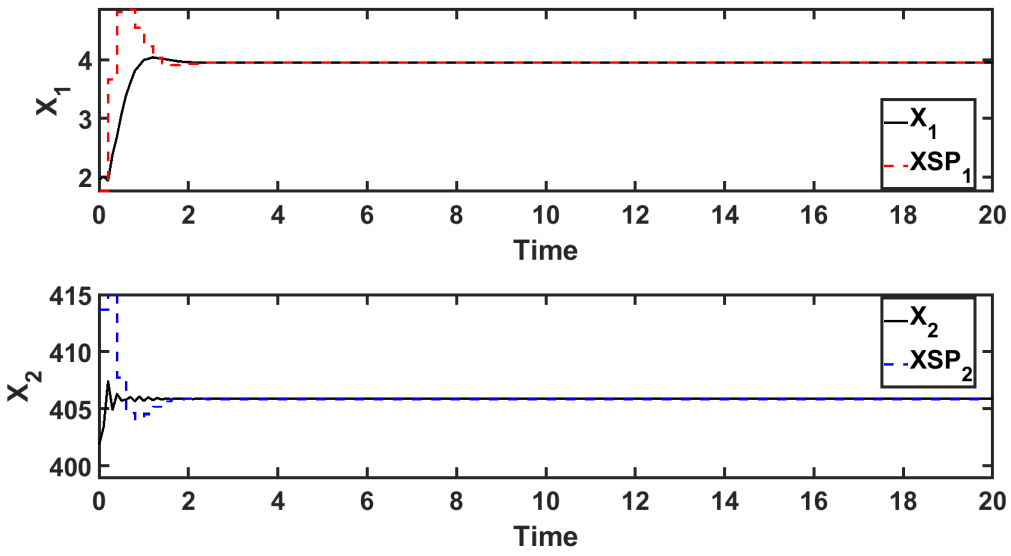

Figure 4.13: CL-DRTO strategy for target tracking with a SPH of 2

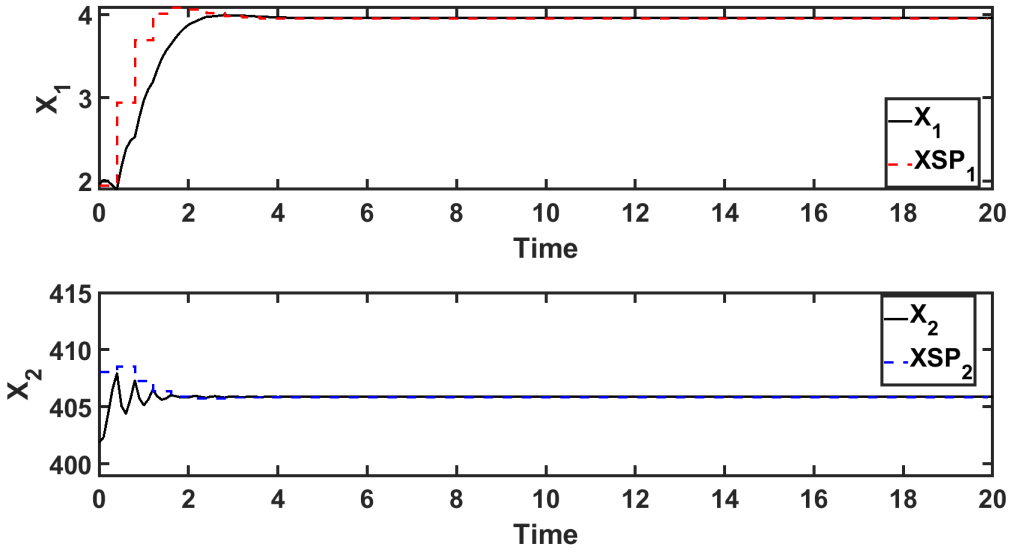

Figure 4.14: CL-DRTO strategy for target tracking with a SPH of 4

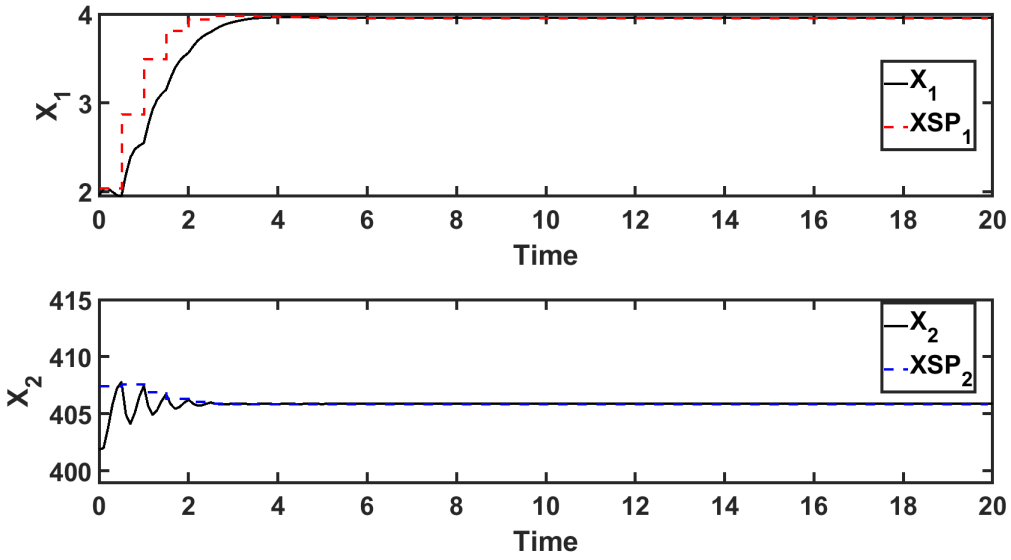

Figure 4.15: CL-DRTO strategy for target tracking with a SPH of 5

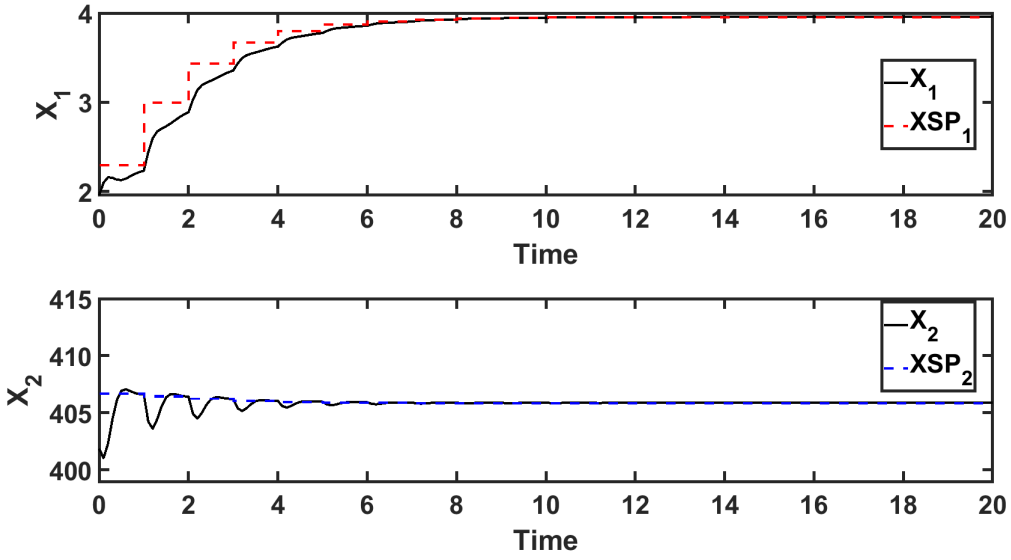

Figure 4.16: CL-DRTO strategy for target tracking with a SPH of 10

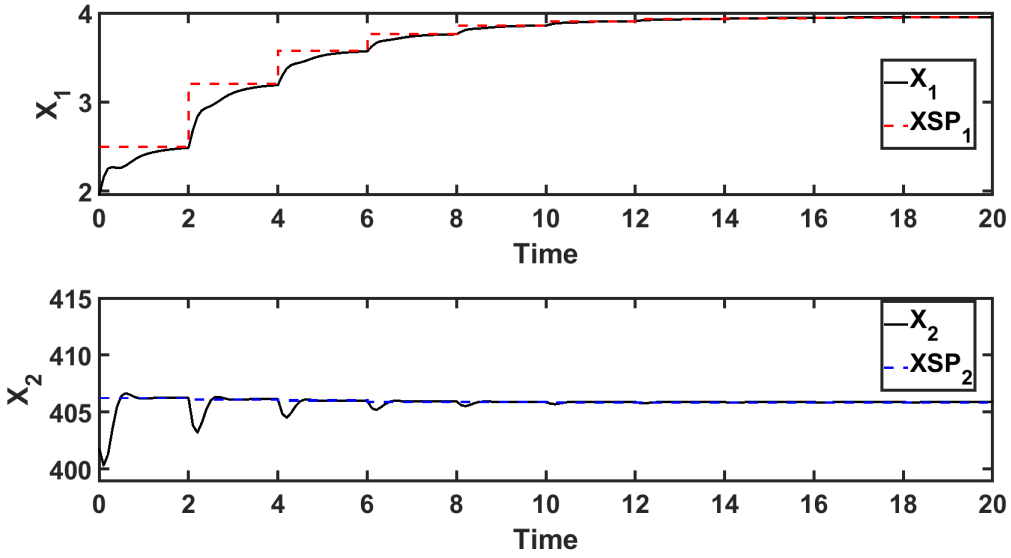

Figure 4.17: CL-DRTO strategy for target tracking with a SPH of 20

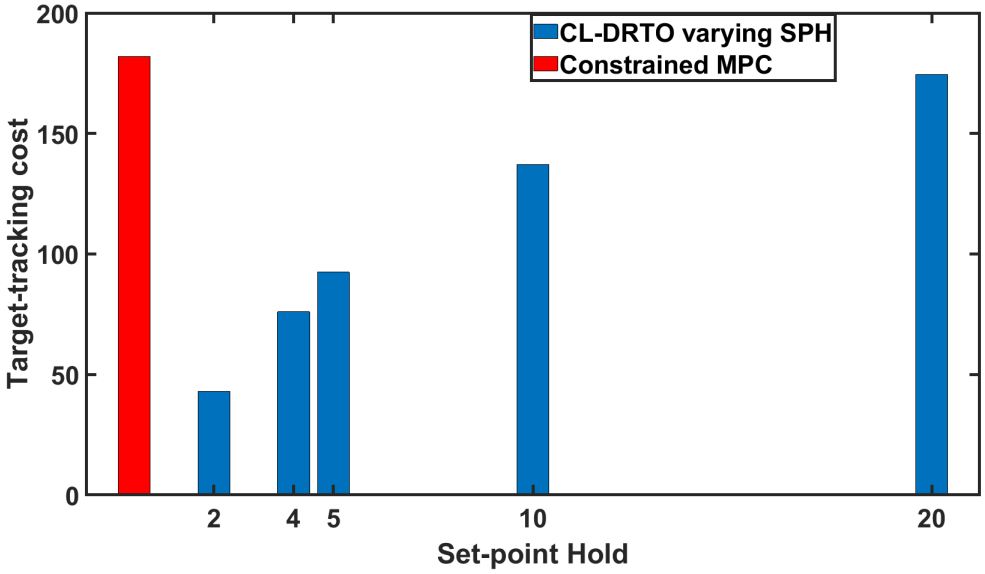

Figure 4.18: Target tracking objective function value of CL-DRTO for various SPH values and constrained MPC

CL-DRTO Economic Cost

The economic measure considered in this case study is to maximize the production rate over the prediction horizon of the DRTO, represented mathematically as shown in Eqn 4.6:

$$\underset{y^{ref}}{\text{maximize}} \quad \sum_{k=0}^{N} k_0 e^{\frac{-E}{RT_k}} C_{Ak}^2 \tag{4.6}$$

The proposed CL-DRTO approach is applied to the MIMO system to maximize the production rate for the design parameters shown in Table 4.5. This optimization is solved offline in GAMS 25.1.2 to find the economically optimal solution to validate the results obtained from the proposed strategy.

Parameter	**Description**	**Value**
T_{Sim}	Total simulation time	10
Δt_{DRTO}	DRTO Sample time	1
Δt_{MPC}	MPC Sample time	0.1
N	DRTO Prediction horizon	40
M	MPC Prediction horizon	5
Q	Output penalty	diag(1,1)
R	Input penalty	diag(0,0)
S	Change in input penalty	diag(1000,1000)

Table 4.5: Design parameters for the DRTO and MPC layers in economic MIMO case study

Fig 4.19 shows the performance of the proposed strategy for the economic objective shown in Eqn 4.6. The final steady-state the system settles at, corresponds to the offline optimal solution computed in GAMS. This case study demonstrates the ability of the proposed approach to reach economically optimal solution whilst staying in the desired operating region and ensuring stable operation. We now introduce nonlinearity in the DRTO model and demonstrate the performance for the economic

objective.

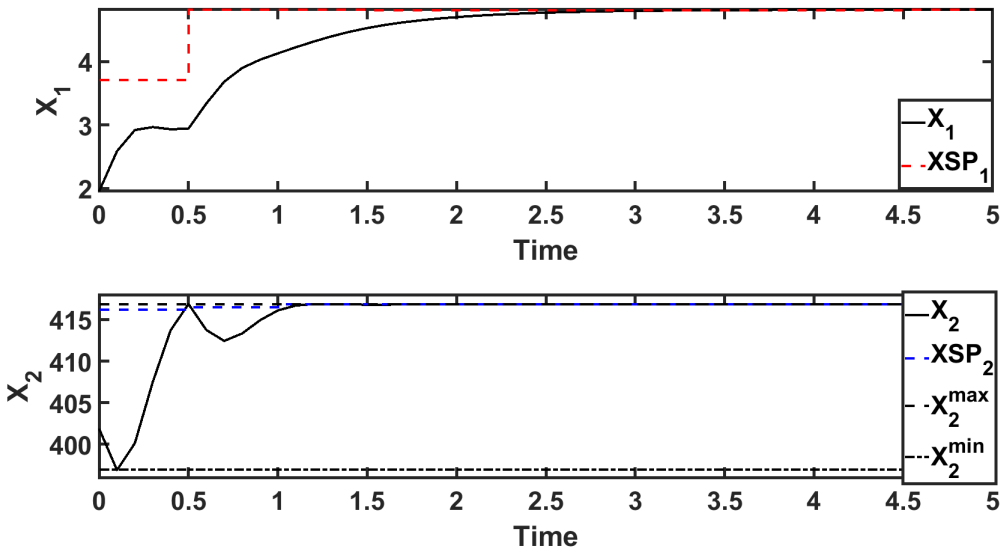

Figure 4.19: CL-DRTO strategy for a MIMO economic optimization

Nonlinear DRTO model

We now introduce the nonlinear model at the DRTO level. Nonlinear equations utilised at the upper level are based on implicit Euler approximation. The same economic optimization is performed and compared with a state constrained MPC and the parameters used are shown in Table 4.6.

Fig 4.20 compares the performance of the proposed CL-DRTO strategy with a nonlinear model at the upper level with a regulatory constrained MPC. The total economic profit for the CL-DRTO strategy is 917.4 and for the constrained MPC is 913.4. For this set of tuning parameters, the proposed strategy outperforms the constrained MPC. To summarise, the proposed strategy is able to perform the task of finding the economic optimum and track it while taking advantage of set-point hold and the additional flexibility from the reference trajectories.

Parameter	Description	Value
T_{Sim}	Total simulation time	1
Δt_{DRTO}	DRTO Sample time	0.1
Δt_{MPC}	MPC Sample time	0.01
N	DRTO Prediction horizon	20
M	MPC Prediction horizon	5
Q	Output penalty	diag(1,1)
R	Input penalty	diag(0,0)
S	Change in input penalty	diag(100,100)

Table 4.6: Design parameters for the DRTO and MPC layers in economic MIMO case study

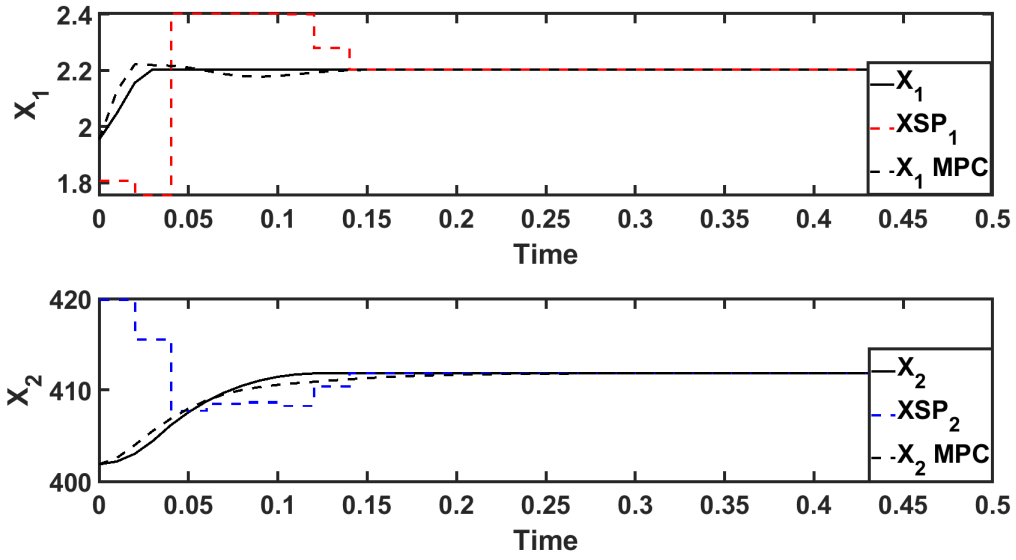

Figure 4.20: Comparison of the proposed nonlinear DRTO strategy and constrained MPC for MIMO economic optimization

Remark 8. We remark that by design, the regulatory MPC only enables tracking a set-point, and thus needs to be provided the final economic optimal steady state to track. In contrast, the proposed CL-DRTO strategy is able to find the economic optimum, and achieve good closed-loop economics.

Chapter 5

Conclusions and Recommendations

In this chapter we summarize the contributions of our study and suggest some recommendations for potential future directions with this research.

5.1 Conclusions

Process industries are operating plants in an increasingly dynamic environment, and the economic optimum in some of these processes could correspond to unstable steady states. The primary focus of this work was to design a DRTO strategy that could readily be applied to open-loop unstable processes while taking into account the effects of the underlying control system. We used a two-layer approach to be consistent with the industrial process automation architecture.

1. Closed-loop DRTO formulation for stabilizing MPCs

We formulated a stabilizing CL-DRTO strategy by embedding stabilizing MPC subproblems as constraints in the DRTO optimization. The inner MPC subproblems were replaced with their equivalent KKT optimality conditions to form a single level

MPCC. The stabilizing CL-DRTO strategy was illustrated through applications to SISO and MIMO CSTR case studies, where the benefits of the proposed approach in its ability to override the tuning parameters of the underlying plant control system and move the process in an economic fashion while ensuring stable operation is illustrated.

2. Inclusion of nonlinearity at the DRTO level

The formulation introduced previously was based on linear dynamic models both at the MPC level and at the DRTO level. We then introduced nonlinear models at the DRTO level by converting them into nonlinear discretized equations based on a backward Euler scheme. The performance of the formulation with a linear and a nonlinear model at the DRTO level was compared and illustrated in the case studies presented.

5.2 Recommendations for future work

Some potential future directions to continue this research are summarized here:

1. Application of approximation techniques to the stabilizing CL-DRTO formulation

Inclusion of the stabilizing MPC subproblems as constraints in the DRTO optimization increases the size of the problem and hence the computational time. This is a potential drawback, particularly for problems with a small DRTO sampling time. Approximation techniques such as hybrid formulation and input clipping have been discussed in Jamaludin and Swartz [2017a]. Application of these techniques to our formulation could help decrease the computational time.

2. A stabilizing CL-DRTO formulation with a Lyapunov based controller

Stabilizing MPC used in our formulation employed an endpoint equality constraint. MPCs formulated with a terminal constraint and or penalty rely on control horizons to be sufficiently long to ensure feasibility. This drawback can be addressed by using Lyapunov techniques to construct a Lyapunov based controller as discussed in Section 2.2.1. By incorporating this controller into the DRTO formulation we could characterize an explicit set of initial conditions from which feasibility and stability can be guarenteed.

3. Extending the formulation to a nonlinear MPC or MPC formulated as a nonconvex optimization

Inclusion of MPC subproblems into the DRTO formulation resulted in a multi-level dynamic optimization and was reformulated by using a simultaneous solution approach. This involved replacing the subproblems with their equivalent KKT optimality conditions. Therefore, it was limited to MPC formulations that are posed as a convex problem and as such do not include nonlinear MPCs. Inclusion of nonlinearity at the controller level could improve the performance; hence, a strategy to include nonconvex optimizations as constraints in the CL-DRTO problem needs to be developed.

4. Extending the formulation from state to output feedback problems

The formulation introduced was limited to state feedback problems, where all the states where assumed to be measured. Industrial applications involve sensor based measurements and not all the states are measured. Therefore, formulating the problem as an output-feedback based CL-DRTO could widen the range of applications. This could involve introducing observers, Kalman filters or some state-estimation technique.

References

Amrit, R., Rawlings, J. B., and Angeli, D. (2011). Economic optimization using model predictive control with a termional cost. *Ann. Rev. Contr.*, **35**, 178–186.

Baker, R. and Swartz, C. L. E. (2008). Interior point solution of multilevel quadratic programming problems in constrained model predictive control applications. *Industrial & Engineering Chemistry Research*, **47**, 81 – 91.

Baumrucker, B., Renfro, J., and Biegler, L. (2008). Mpec problem formulations and solution strategies with chemical engineering applications. *Computers & Chemical Engineering*, **32**(12), 2903 – 2913.

Bemporad, A. and Morari, M. (1999). Robust model predictive control: A survey. In Garulli, A. and Tesi, A. (Eds.), *Robustness in identification and control*, pp. 207–226, London. Springer London.

Darby, M. L., Nikolaou, M., Jones, J., and Nicholson, D. (2011). Rto: An overview and assessment of current practice. *Journal of Process Control*, **21**(6), 874 – 884.

Ellis, M. and Christofides, P. D. (2014). Economic model predictive control with time-varying objective function for nonlinear process systems. *AIChE Journal*, **60**(2), 507–519.

Ellis, M., Durand, H., and Christofides, P. D. (2014). A tutorial review of economic model predictive control methods. *Journal of Process Control*, **24**(8), 1156 – 1178. Economic nonlinear model predictive control.

García, C. E., Prett, D. M., and Morari, M. (1989). Model predictive control: Theory and practice—a survey. *Automatica*, **25**(3), 335 – 348.

Heidarinejad, M., Liu, J., and Christofides, P. D. (2012). Economic model predictive control of nonlinear process systems using lyapunov techniques. *AIChE Journal*, **58**(3), 855–870.

Jamaludin, M. Z., Li, H., and Swartz, C. L. E. (2017). The utilization of closed-loop prediction for dynamic real-time optimization. *The Canadian Journal of Chemical Engineering*, **95**(10), 1968–1978.

Jamaludin, M. Z. and Swartz, C. L. E. (2017a). Approximation of closed-loop prediction for dynamic real-time optimization calculations. *Computers & Chemical Engineering*, **103**, 23 – 38.

Jamaludin, M. Z. and Swartz, C. L. E. (2017b). Dynamic real-time optimization with closed-loop prediction. *AIChE Journal*, **63**(9), 3896–3911.

Kadam, J., Schlegel, M., Marquardt, W., Tousain, R., van Hessem, D., van den Berg, J., and Bosgra, O. (2002). A two-level strategy of integrated dynamic optimization and control of industrial processes—a case study. Vol. 10 of *Computer Aided Chemical Engineering*, pp. 511 – 516. Elsevier.

Li, H. and Swartz, C. L. (2019). Dynamic real-time optimization of distributed mpc systems using rigorous closed-loop prediction. *Computers & Chemical Engineering*, **122**, 356 – 371. 2017 Edition of the European Symposium on Computer Aided Process Engineering (ESCAPE-27).

Lundström, P., Lee, J., Morari, M., and Skogestad, S. (1995). Limitations of dynamic matrix control. *Computers & Chemical Engineering*, **19**(4), 409 – 421.

Mayne, D., Rawlings, J., Rao, C., and Scokaert, P. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, **36**(6), 789 – 814.

Mhaskar, P., El-Farra, N. H., and Christofides, P. D. (2005). Stabilization of nonlinear systems with state and control constraints using lyapunov-based predictive control. *Systems and Control Letters*, **55**(8), 650–659.

Muske, K. R. and Rawlings, J. B. (1993a). Linear model predictive control of unstable processes. *Journal of Process Control*, **3**(2), 85 – 96.

Muske, K. R. and Rawlings, J. B. (1993b). Model predictive control with linear models. *AIChE Journal*, **39**(2), 262–287.

Patwardhan, A., Rawlings, J., and Edgar, T. (1989). Model predictive control of nonlinear processes in the presence of constraints. *IFAC Proceedings Volumes*, **22**(3), 345 – 349. Nonlinear Control Systems Design, Capri, Italy, 14-16 June 1989.

Qin, S. J. and Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, **11**(7), 733 – 764.

Raghunathan, A. U. and Biegler, L. T. (2003). Mathematical programs with equilibrium constraints (mpecs) in process engineering. *Computers & Chemical Engineering*, **27**(10), 1381 – 1392.

Ralph, D. and Wright, S. (2004). Some properties of regularization and penalization schemes for mpecs. *Optim. Methods Softw.*, **19**, 527–556.

Scokaert, P. O. M. and Rawlings, J. B. (1999). Feasibility issues in linear model predictive control. *AIChE Journal*, **45**(8), 1649–1659.

Tosukhowong, T., Lee, J. M., Lee, J. H., and Lu, J. (2004). An introduction to a dynamic plant-wide optimization strategy for an integrated plant. *Computers & Chemical Engineering*, **29**(1), 199 – 208. PSE 2003.

Uppal, A., Ray, W., and Poore, A. (1974). On the dynamic behavior of continuous stirred tank reactors. *Chemical Engineering Science*, **29**(4), 967 – 985.

Würth, L., Hannemann, R., and Marquardt, W. (2011). A two-layer architecture for economically optimal process control and operation. *Journal of Process Control*, **21**(3), 311 – 321. Thomas McAvoy Festschrift.

Ying, C.-M. and Joseph, B. (1999). Performance and stability analysis of lp-mpc and qp-mpc cascade control systems. *AIChE Journal*, **45**(7), 1521–1534.