Design and Implementation of Model Predictive Control (MPC) in MATLAB Simulink

Utkarsh Kumar

Department of Electrical Engineering

Indian Institute of Technology (Indian School of Mines) Dhanbad

1 Introduction

Model Predictive Control (MPC) represents a significant advancement in control systems, offering a robust methodology for managing dynamic systems with constraints. In this project, we explore the implementation of MPC to control a first-order thermal system modeled as a heater. The goal is to compare the performance of MPC with the traditional PID controller, illustrating the advantages and limitations of each approach. The system is modeled using basic building blocks like integrators and summers in Simulink, and the simulation evaluates the system's response to a step input under different control strategies.

2 Dynamic System Modeling and Control Approaches

2.1 First-Order Differential Systems

A first-order differential system is characterized by a single energy storage element, and its behavior can be mathematically represented by a first-order linear differential equation of the form:

$$\tau \frac{dT(t)}{dt} + T(t) = K \cdot u(t) + T_{ambient}$$

Here, $T(t)$ is the system output (temperature in this case), $u(t)$ is the input (heat power), $T_{ambient}$ is the ambient temperature, K is the system gain, and τ is the time constant that dictates the system's response speed. The time constant τ reflects how quickly the system approaches its steady state after a change in input, with smaller values indicating faster responses.

By taking the Laplace transform of the differential equation (assuming zero initial conditions), the system can be expressed in the s-domain as:

$$T(s) = \frac{K}{\tau s + 1} \cdot U(s) + \frac{1}{\tau s + 1} \cdot T_{ambient}(s)$$

This transfer function, $\frac{K}{\tau s+1}$, is a key representation of a first-order system, showing its proportionality and responsiveness to input changes. The term $\frac{1}{\tau s+1}$ highlights the low-pass filter nature of the system, where high-frequency inputs are attenuated.

In this project, the heater's temperature dynamics are modeled as a first-order system using fundamental blocks like integrators and summers. This block-based implementation replicates the transfer function, capturing the interplay of input heat power, thermal inertia (τ), and external influences like ambient temperature ($T_{ambient}$).

2.2 Model Predictive Control (MPC)

Model Predictive Control (MPC) is an advanced control strategy that leverages a dynamic model of the system to predict its future behavior and optimize control actions. At each time step, MPC solves a constrained optimization problem over a finite prediction horizon to determine the optimal control inputs. The prediction horizon represents the future time window over which the system's behavior is forecasted, while the control horizon determines the duration over which the optimized control inputs are applied. Beyond the control horizon, inputs are typically held constant to simplify computations.

MPC's defining feature is its ability to explicitly consider system constraints—such as input and state limits—during the optimization process, ensuring feasible and safe operation. This makes MPC highly suitable for complex systems with multiple interacting variables, such as industrial processes, autonomous vehicles, and energy systems. Additionally, MPC can adapt to changing conditions and disturbances by recalculating control actions at every time step, maintaining robust performance.

Despite its advantages, MPC has limitations. Its reliance on solving optimization problems in real-time demands significant computational resources, which may limit its applicability in systems requiring ultra-low-latency responses. Additionally, MPC requires an accurate mathematical model of the system, and inaccuracies can degrade performance.

Compared to traditional controllers like PID, MPC excels in multivariable and constrained environments, where PID controllers struggle to account for interactions and complex dynamics. However, for simpler systems or when computational constraints exist, PID or state-space controllers may be more practical alternatives.

Future advancements in computational efficiency and machine learning integration may further expand the applicability of MPC, allowing its use in a broader range of real-time systems. Its inherent adaptability and optimization-driven approach make MPC a cornerstone for modern control applications across industries.

2.3 PID Controllers

The Proportional-Integral-Derivative (PID) controller is one of the most fundamental and widely used feedback control strategies in engineering. It adjusts the control input based on three components: proportional (P), integral (I), and derivative (D) actions, each addressing different aspects of the control objective. Mathematically, the PID controller can be expressed as:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

where $u(t)$ is the control input, $e(t)$ is the error (difference between desired and actual outputs), and K_p, K_i, K_d are the proportional, integral, and derivative gains, respectively.

- Proportional control provides a control action proportional to the current error, reducing large deviations.
- Integral control accumulates past errors to eliminate steady-state errors.
- Derivative control predicts future errors based on the rate of error change, improving stability and response time.

PID controllers are simple to implement and computationally efficient, making them suitable for single-input, single-output (SISO) systems like motor speed, temperature, or flow control. However, they are less effective for systems with time delays, nonlinear dynamics, or multivariable interactions. Unlike Model Predictive Control (MPC), PID controllers do not consider constraints or predict future system behavior, which can lead to suboptimal performance in complex scenarios. Despite these limitations, their versatility and ease of tuning ensure their continued relevance in numerous industrial and academic applications.

2.4 PID vs. MPC

While PID controllers excel in simplicity and real-time applications, MPC provides superior control in systems requiring constraint handling and multivariable interactions. MPC's predictive nature allows it to optimize performance over a horizon, reducing overshoot and settling time. Conversely, PID relies on feedback alone, which can lead to suboptimal control in complex systems. The choice between PID and MPC depends on the system's complexity, constraints, and available computational resources.

3 Simulation Methodology

3.1 Modeling the Thermal System

The simulation starts with modeling the thermal system in Simulink to represent first-order dynamics using integrators, summers, and gain blocks. The first-order system's differential equation is

$$\tau \frac{dT(t)}{dt} + T(t) = K \cdot U(t) + T_{ambient}.$$

This is implemented by connecting integrators for $\frac{1}{s}$, summer blocks to handle summations and subtractions, and gains for coefficients like $\frac{1}{\tau}$ and K. This block-based approach avoids using a predefined transfer function block, ensuring flexibility and transparency in the model construction. A step input is applied to simulate a sudden change in heat input, helping visualize the system's transient and steady-state behavior.

3.2 Designing the MPC Controller

The Model Predictive Control (MPC) design utilizes MATLAB's MPC Toolbox. First, the plant model, derived from the thermal system's dynamics, is defined. Next, constraints on control input and output are specified, such as upper and lower temperature limits, to reflect realistic operating conditions. Prediction and control horizons are chosen based on system dynamics and computational feasibility. The prediction horizon captures future system behavior, while the control horizon determines the number of control moves. These parameters balance performance and computation, ensuring real-time applicability.

3.3 Tuning the PID Controller

A PID controller is manually tuned to achieve performance comparable to the MPC. The proportional, integral, and derivative gains are adjusted iteratively to minimize steady-state error, overshoot, and settling time. This ensures a baseline control performance for comparison.

3.4 Simulation and Comparison

Both controllers are integrated into the Simulink model. For MPC, the controller block interfaces with the system, dynamically adjusting control inputs. For PID, gains are set in the PID block. The responses are analyzed under identical conditions, including initial temperature, step input amplitude, and ambient temperature. The simulation helps compare the advantages of MPC over PID in terms of overshoot, settling time, and constraint handling.

3.5 Parameter Selection

Parameters like step time (100 s), stop time (500 s), ambient temperature (15 °C), and step input values (initial 100, final 50) are chosen to reflect practical scenarios. These ensure the simulations are meaningful and relatable to real-world systems.

4 Results and Conclusions

The primary aim of this project was to explore advanced control strategies for first-order systems, focusing on improving system response, constraint handling, and understanding the trade-offs between computational complexity and control performance. By modeling a thermal system as a first-order differential system, this project sought to provide a practical framework for comparing classical and modern control techniques, specifically PID and Model Predictive Control (MPC).

The simulations clearly demonstrate the superiority of MPC over PID in terms of achieving smoother transitions, reduced overshoot, and faster settling time. MPC maintained the system's output within predefined constraints, showcasing its ability to handle complex scenarios with precision. In contrast, the PID controller, while simpler to implement, exhibited larger overshoots and struggled to meet the same performance standards under identical conditions.

Through this project, I achieved a deeper understanding of system modeling, the nuances of tuning control strategies, and the practical challenges of implementing advanced controllers. The experience reinforced the importance of selecting appropriate prediction and control horizons in MPC and highlighted the computational demands associated with real-time optimization. It also provided valuable insights into the limitations of PID controllers, especially in systems with constraints or multivariable interactions.

This project not only improved my technical skills in modeling and simulation but also broadened my perspective on the trade-offs between simplicity and sophistication in control strategies. It underscored the importance of aligning control objectives with system requirements and computational feasibility, lessons that are transferable across engineering domains. These findings strengthen my resolve to pursue further research in advanced control systems, aiming to bridge the gap between theoretical advancements and practical applications.

5 Future Scope

This project is a foundational step in exploring advanced control strategies. While applied to a single-input, single-output (SISO) first-order thermal system, MPC's scope extends to multivariable and non-linear systems. Future work could involve implementing adaptive MPC to address system uncertainties, integrating reinforcement learning to enhance decision-making in non-linear systems, or deploying MPC on hardware platforms for real-world validation. Additionally, comparing MPC with emerging control strategies like neural network-based controllers could provide insights into optimizing performance in dynamic environments. By expanding its application, MPC can contribute significantly to advancing control methodologies in diverse engineering challenges.