

## Chapter 12

# Model Predictive Control



Control design often seeks the best trajectory along which to move a system from its current state to a target state. Most control methods approximate this goal by using the current inputs and system state to calculate the next control signal to drive system dynamics. That standard approach considers only the first step of the full trajectory toward the target state. The idea is that estimating a good first step in the right direction is sufficient, without consideration of the full trajectory from the current location to the final target.

Model predictive control considers the full sequence of input steps required to move the system optimally from its current state to a future target. The control system then applies the first inputs to start the system along that optimal trajectory (Rossiter 2004; Camacho and Bordons 2007; Ellis et al. 2014; Rawlings and Mayne 2015).

After applying the initial inputs, the system does not use the additional sequence of calculated inputs to continue along the planned trajectory. Instead, the system takes updated measures of the external target and the internal state. The new information is used to recalculate an updated optimal trajectory. Using the updated trajectory, the newly calculated first inputs are then applied to the system. The process repeats with each new round of updated external and internal signals.

This approach considers a receding future horizon. At each point in time, the system calculates the optimal trajectory to a particular time point in the future—the horizon. Then, after a small amount of time passes relative to the future horizon, the system recalculates by taking current inputs and advancing the future horizon by the time elapsed.

Intuitively, this approach seems similar to many decisions made by humans. We estimate how we will get to a goal, start off in the best direction, then update our plan as new information arrives.

Our estimate of how we will get to a goal depends on an internal model of our dynamics and on the modulating control signals that we will use to alter our dynamics. The self-correcting process of recalculating the planned trajectory means that we do not need an accurate model of our internal dynamics to perform well. An approximate or misspecified model of dynamics often works well, even for nonlinear processes.

Model predictive control may be enhanced by adaptive feedback that modifies the parameters or the form for the model of internal dynamics. The general approaches of model predictive control and adaptive control provide benefits of robustness with respect to the model of internal dynamics.

### 12.1 Tracking a Chaotic Reference

Figure 12.1 shows the performance of a simple model predictive control system when tracking a chaotic reference signal. The figure caption describes the calculation of the chaotic input signal (blue curve). In this example, the model predictive control system begins with an internal process given by

$$\ddot{x} = u \tag{12.1}$$

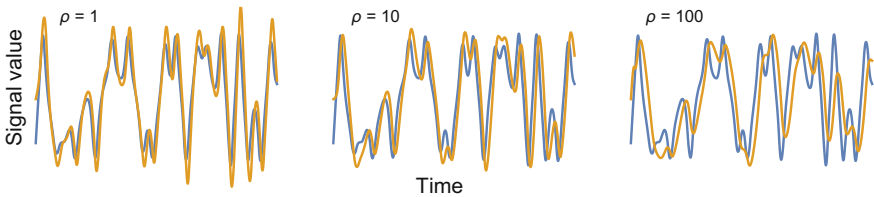
for control input,  $u$ . The input can be thought of as a force that alters the acceleration of the system state.

In this example, the cost function is summed over a series of discrete time points from the present until a future horizon at time  $T$ , yielding

$$\mathcal{J} = \sum_{t=0}^T [x(t) - r(t)]^2 + \rho u(t)^2.$$

The first term is the squared tracking error. The second term is the squared control input signal, weighted by  $\rho$ , which is the relative cost of the input signal to the tracking error. The sum includes a series of input controls and tracking deviations over a finite period from the present to the future horizon,  $T$ . As time passes, the time window  $(0, T)$  moves so that  $t = 0$  is always the present time.

The system has sensors to measure the current locations and time derivatives (velocities) for the system state and the reference signal, given as  $x_0, \dot{x}_0, r_0$ , and



**Fig. 12.1** Tracking an environmental reference signal by model predictive control. The blue curves show the environmental signal, a chaotic form of Duffing’s equation,  $\ddot{r} + 0.15\dot{r} - r + r^3 = 0.3 \cos(t)$ , with initial conditions  $\dot{r} = 1.001$  and  $r = -1$ . The gold curves show the tracking performance for different values for the cost,  $\rho$ , of control input signals relative to the tracking error. A higher control input cost favors weaker control inputs and greater tracking error

$\dot{r}_0$ . From those measurements, and from the projected set of inputs,  $u(t)$ , over the interval  $t = 0, 1, \dots, T$ , the system can project the values of  $x(t)$  and  $r(t)$  and thus estimate the tracking errors and the total cost,  $\mathcal{J}$ . In particular,

$$x(t) = x_0 + \dot{x}_0 t + \sum_{\alpha=0}^{t-1} (t - \alpha) u(\alpha)$$

$$r(t) = r_0 + \dot{r}_0 t.$$

Here,  $u(\alpha)$  is interpreted as an impulse that acts at the continuous time offset,  $\alpha$ , relative to the current time. Because the system has dynamics  $\ddot{x} = u$ , an impulse at  $\alpha$  causes an instantaneous increase in velocity by  $u(\alpha)$ , which then acts to change the future predicted value at time  $t$  by  $(t - \alpha)u(\alpha)$ .

These predicted values for  $x$  and  $r$  allow calculation of the sequence of control inputs  $u(t)$  over the interval  $t = 0, 1, \dots, T$  that minimize the cost,  $\mathcal{J}$ .

The plots in Fig. 12.1 show the system trajectory that reflects the minimization of  $\mathcal{J}$ . At each time step, the system calculates the sequence  $u$  to minimize  $\mathcal{J}$ , then applies  $u(0)$  as the control input. The remaining  $u$  values for  $t = 1, 2, \dots, T$  are ignored. In the next time step, the same procedure gives the new control input for that time period, and the future inputs for the optimal trajectory are again ignored. The process continues for each time period as the future horizon recedes.

## 12.2 Quick Calculation Heuristics

The solution for the input  $u(0)$  typically depends on the full sequence of inputs over  $t = 0, 1, \dots, T$ . In some cases, a relatively simple explicit solution for  $u(0)$  can be obtained that requires only the current measured inputs for  $x_0, \dot{x}_0, r_0$ , and  $\dot{r}_0$ .

If a system applies only the first input,  $u(0)$ , before recalculating in the next time step, then that system only needs the explicit solution for  $u(0)$  to update the control inputs in each time step.

For example, with  $T = 2$ , the exact solution for the above case is

$$u(0) = \frac{\Delta + \rho[3\Delta + 2(\dot{r}_0 - \dot{x}_0)]}{1 + 6\rho + \rho^2},$$

with  $\Delta = (r_0 + \dot{r}_0) - (x_0 + \dot{x}_0)$ . With larger  $T$ , the solution has more terms as powers of  $\rho$ , but nonetheless remains a relatively simple ratio of polynomials in  $\rho$  that could be approximated by a quick heuristic calculation.

### 12.3 Mixed Feedforward and Feedback

In some cases, the sensor and calculation costs of updating in each time step may not provide sufficient benefit. Instead, the system could apply the first few control inputs of the sequence,  $u(0), u(1), \dots, u(\tau)$ , and then update the sequence at time  $\tau < T$ .

A system that uses feedback inputs at one time point to calculate and then apply a future sequence of control inputs is running in partial feedforward mode. The feedback inputs arrive, and then the system runs forward from those inputs without the feedback correction obtained by comparing the changing system output to the potentially changing target reference signal.

After a while, the system may take new input readings and update the projected sequence of future control signals. Each measurement and recalculation acts as a feedback correction process. Thus, systems may combine the simplicity and relatively low cost of feedforward control with the correction and robustness benefits of feedback.

### 12.4 Nonlinearity or Unknown Parameters

This section's example used a simple model of internal dynamics,  $\ddot{x} = u$ , given in Eq. 12.1. That expression, equating acceleration and force, provided a simple way in which to analyze trajectories. That internal model may often perform well even if the true model is nonlinear because the first move along the calculated trajectory often depends on how the force of the applied input alters the acceleration of the system.

Alternatively, one could use a more general expression for the internal model dynamics, with a set of unknown parameters. One could then add an adaptive control layer to the system to provide updated parameter estimates. In some cases, this combination of model predictive control and adaptive control may perform well.

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

