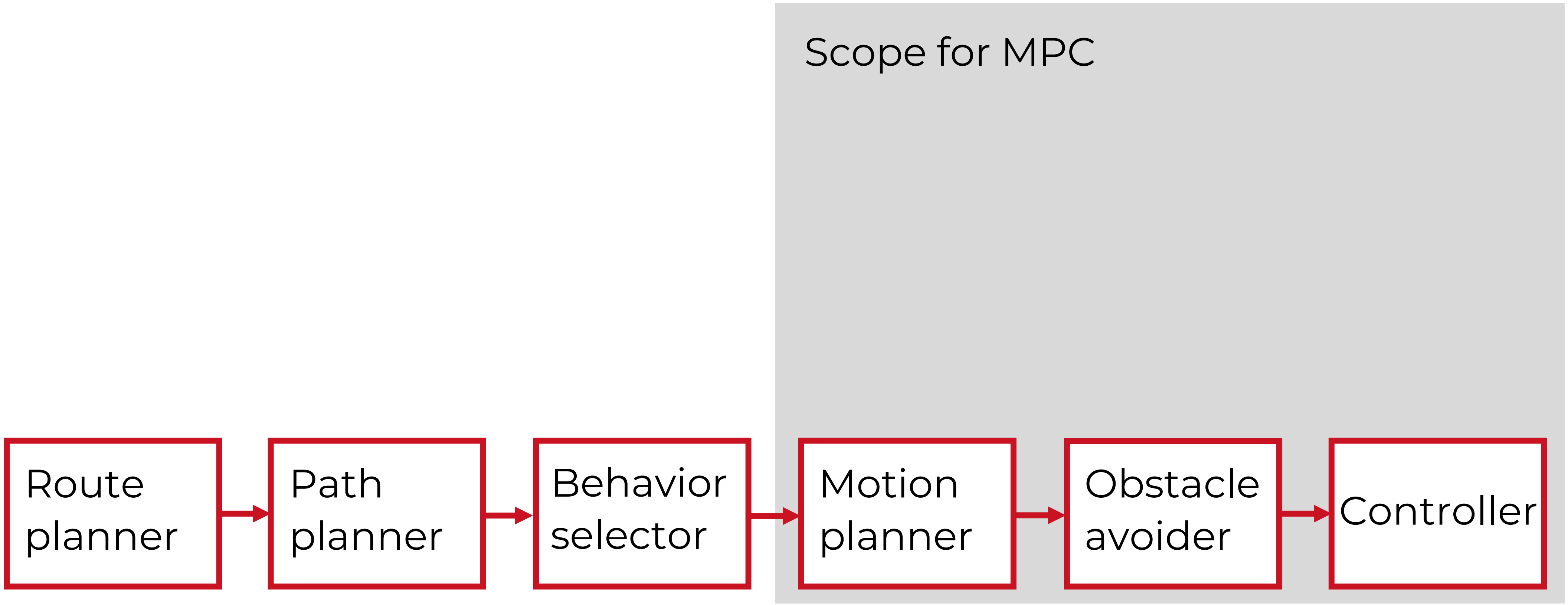
Model Predictive Control - FAQs

embotech


Scope for MPC
Route planner
Path planner
Behavior selector
Motion planner
Obstacle avoider
Controller

embotech

WHAT IS MPC?

An advanced feedback control technique

… But let‘s start from the beginning

WHAT IS CONTROL?

To act on a system to make it behave in a desired manner

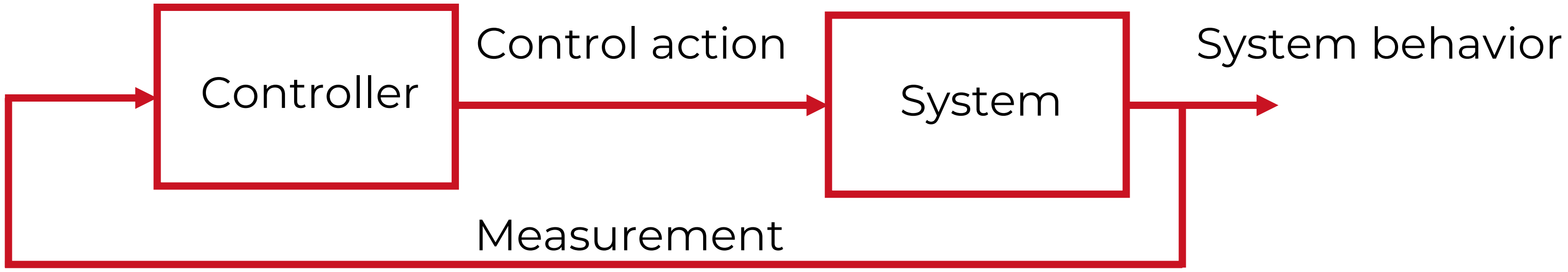
If we have **no uncertainty** all we need is this

In reality, there is almost always **uncertainty**

WHAT IS FEEDBACK CONTROL?

The only way to deal with uncertainty

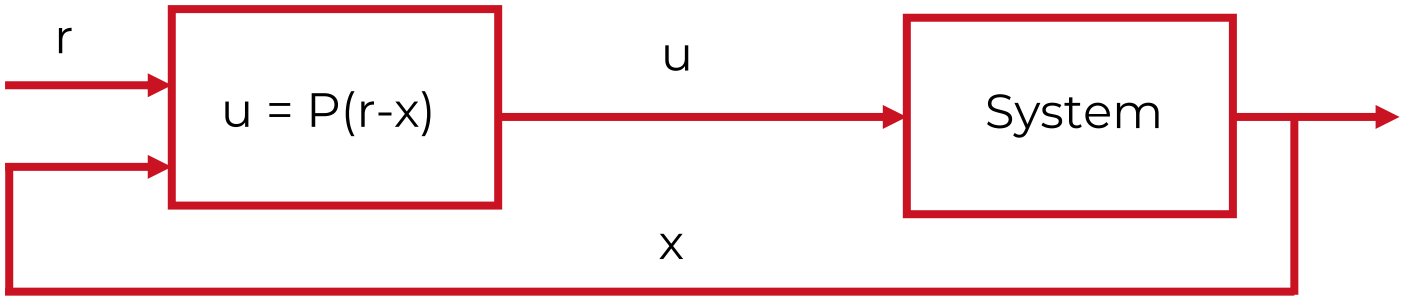
In feedback control we **measure, calculate, act**



The **Controller** is where we excecute our control algorithm

WHAT IS FEEDBACK CONTROL?

The simplest control algorithm is a **Proportion** of an error between where we want the system to be and where it is



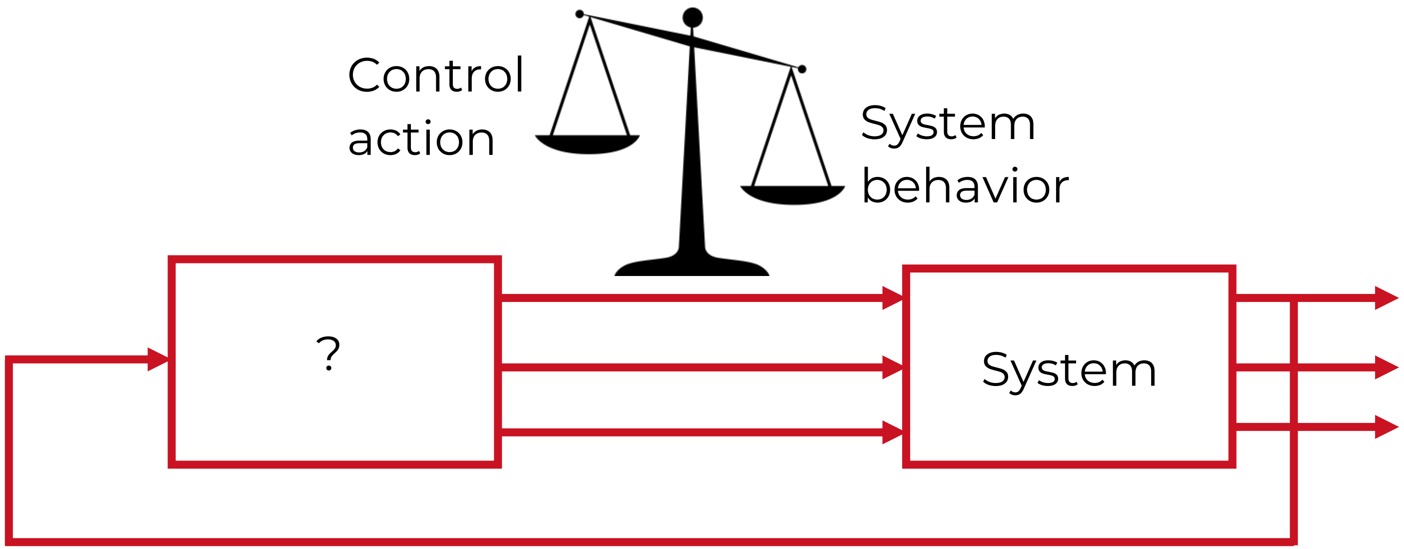
WHAT IS FEEDBACK CONTROL?

The control algrithm that computes the sum of the Proportion, the Integral and the Derivative of the error is the **PID controller**

WHAT IF WE HAVE MULTIPLE INPUTS/OUTPUTS?

WHAT IF WE WANT TO CONTROL OPTIMALLY?

E.g. to find the best trade-off between the effort/energy required to actuate and the ability to reject disturbances



WHAT IS OPTIMAL CONTROL?

A control technique that achieves **the best possible result**

WHAT IS THE BEST POSSIBLE RESULT?

The minimum of a mathematical function often called the **objective function**. For instance, this:

$$\sum_{i=0}^{\infty}\left(\underline{{x_i}^T Q x_i} + \underline{u_i^T R u_i} + \underline{2x_i^T M u_i}\right)$$

From now until the end of time

How much the system deviates from the desired behavior

The effort required

Not necessary, included for completeness

WHAT IS LQR?

A control technique that achieves the best possible result for a **quadratic objective** function and for a system with **linear** dynamics (Linear Quadratic Regulator)

WHAT IS LQG?

An LQR controller where some the system behavior (the states of the controlled system) are estimated via a Kalman Filter

WHAT ABOUT CONTRAINTS?

Often, we need to make sure the system behavior and/or the control action stay within certain limits

E.g. there is a maximum force an actuator can deliver or a maximum overshoot that can be tolerated

Such limits are called **constraints**

WHAT IS MPC?

An implementation of LQR (or LQG) that is also able to satisfy constraints (in its simplest form)

LINEAR MPC

Constrained LQR looks like this

minimize $x_N^\top P x_N + \sum_{i=0}^{N-1} x_i^\top Q x_i + u_i^\top R u_i$ ← Objective function

subject to $x_0 = x$ ← Measurement or estimate of the system states

$x_{i+1} = Ax_i + Bu_i$ ← Equality constraints (system dynamics)

$\underline{x} \leq x_i \leq \bar{x}$ ← Inequality constraints on the system’s states

$\underline{u} \leq u_i \leq \bar{u}$ ← Inequality constraints on the control actions

LINEAR MPC

- A and B are the system state-space model matrices, which we have
- Q, R and P are „weights“ that can be chosen (Q and R are normally diagonal and there are methods to choose P wisely)
- N is a trade off. Larger N can lead to better closed-loop results but slower computation

HOW DO WE IMPLEMENT MPC?

- We measure or estimate the system states
- We solve an optimization problem
- We take from the solution only the result of the first control action
- We apply the control action

This is also called Receding Horizon Control

WHEN SHOULD WE USE MPC?

For feedback control problems where we have constraints and more than one input/output

CAN MPC RUN IN REAL-TIME?

Very often these days, thanks to progress in algorithms development and hardware optimization. MHz rates are not unusual

MPC VS DYNAMIC PROGRAMMING

- They are both methods to compute optimal feedback control laws
- MPC uses online optimization to solve an open-loop optimal control problem. A feedback control law is defined implicitly by repeating the optimization calculation after a feedback update of the state at each sample time
- DP derives an explicit feedback law offline by deriving and solving so-called Bellman's optimality equation

MPC VS REINFORCMENT LEARNING?

- RL is model-free and adaptive (the system model is learnt online)
- RL has lower online complexity than MPC
- RL is not as mature as MPC in terms of stability, constraint handling and robustness
- RL not suitable for safety critical applications

TAKE AWAY POINTS

- Use MPC when you have constraints and multiple inputs/outputs
- MPC particularly useful for trajectory generation
- Use readily available interfaces and solvers, some are free and powerful

embotech.com