Introduction to Model Predictive Control

Model Predictive Control (MPC) is the only advanced control technique that's seen widespread impact on industrial process control

$\Rightarrow$ It's the *only* control technology that can deal with *constraints*

- Operation near constraints can bring about profitable and efficient operation
- The concepts behind predictive control are:
  - Easy to understand
  - Extensible to multivariable systems
  - More powerful than Proportional-Integral-Derivative (PID) control
- The origins of predictive control go back 50 years – but all formulations have the following common elements:
  - An explicit internal model
  - Idea of a *receding horizon*
  - Computation of control signal by optimizing predicted plant behavior
- Interestingly, these ideas fundamentally reflect innate human behavior!
- Predictive Control describes an *approach* to control design and not a specific algorithm – there are many types of predictive control laws

- Main components of model predictive control are shared $\Longrightarrow$
  - Actions depend on predictions
  - Predictions are based on a model
  - Current input is based on achieving a *best* output
  - Limited time window (receding horizon)
  - Precise control requires an accurate model
  - Handles constraints in a systematic way

Principles of Predictive Control

Prediction

- Why is prediction important?
- How far ahead should we predict?
- What happens if we don't predict accurately?
- How do we predict?
- You are driving a car down a winding road in the darkness. Your headlights illuminate the roadway ahead for a fixed distance. How does the headlamp distance affect your driving performance?

Receding Horizon

- What is a receding horizon?
- Why is it essential?
  - What are the consequences of uncertainty?
- How is it embedded into model predictive control?

- What advantages does it bring and what consequences does it have on prediction accuracy?

Model

- What is the model used for?
- How does the model use impact the modeling approach?
- What type of model is required? (e.g., FIR, state-space, etc.)
- How accurate must the model be?
  - Transient behavior
  - Steady-state behavior
- What is the relationship between the model and the computational environment?

Performance Index

- Why do we need a performance index?
- How should a performance index be designed?
- What kind of trade-offs should we consider?
- How does horizon length affect the performance index?

Constraints

- How are constraints addressed in most control strategies?
  - Classical control?
  - Modern control?

- Do you use *a posteriori* or *a priori* design?
- How do humans embed constraints into their behavior?
- How is MPC different from most conventional approaches?