



Course: Regulation and control

Lecture 14: Applications & Case Studies

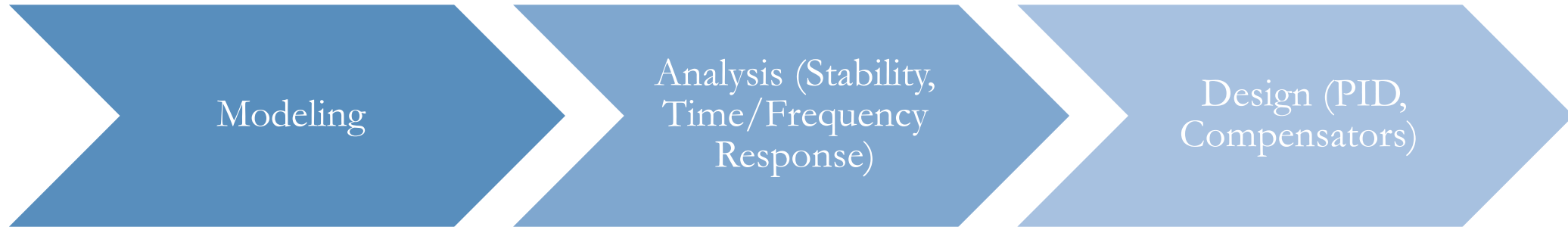
Lecturer: Chalachew Werku

Learning Objectives

- Connect control theory (lecture 1-13) to real-world engineering systems.
- Analyze the control challenges in Industrial Automation, Robotics, and Process Control.
- Understand the design trade-offs (Performance vs. Robustness, Cost vs. Complexity) in different applications.
- Recognize the role of simulation (like Simulink/Simscape) in developing and validating control systems.

The Big Picture: From Model to Implementation

Theoretical Foundation



Practical Implementation



Common Challenges in Real-World Control

- **Nonlinearities:** Friction, Saturation, Backlash
- **Disturbances:** Load changes, wind gusts, sensor noise
- **Time Delays:** Computation, communication, transport lag
- **Unmodeled Dynamics:** Imperfect models, component wear and tear

The Design Compromise Triangle

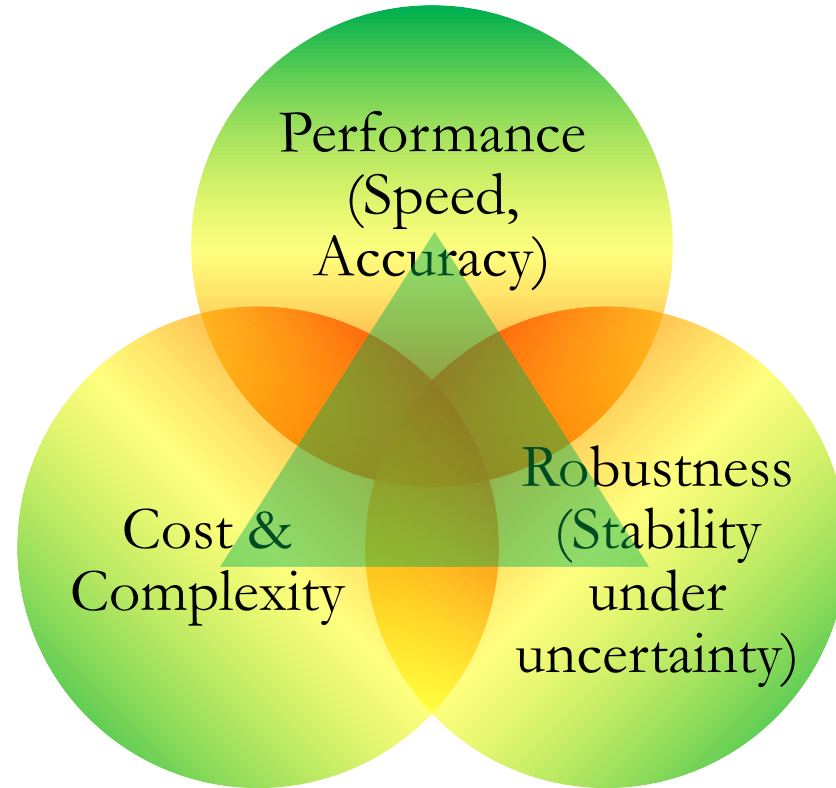


Fig 1: Competing priorities in designing systems. [1]



Part 1 Recap & Roadmap

- Reviewed core concepts.
- Introduced real-world challenges.
- Next: Diving into our first application domain: Industrial Automation.



Precision, Speed, and Reliability.

Industrial Automation

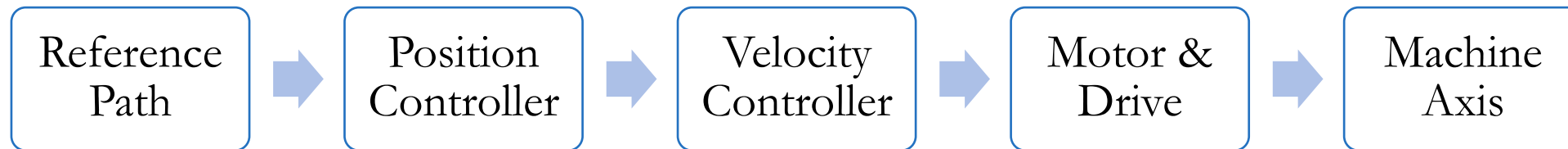
Background picture source: <https://ln.run/bUmLk>

Example 1: CNC Machining & Milling

- **System:** Multi-axis machine tool.
- **Control Goal:** Precise tool positioning and velocity control to follow a complex path.
- **Challenge:** Rejecting disturbances from cutting forces; preventing vibrations (chatter).
- **Control Strategy:** Cascaded PID loops (Position \rightarrow Velocity \rightarrow Current). Often with feedforward control for known paths.

CNC Machine Control Architecture

Block Diagram:



Feedback from: Encoders (position), Tachometers (velocity).

Example 2: Robotic Arm for Pick-and-Place

- **System:** Multi-joint robotic arm.
- **Control Goal:** Move end-effector accurately and quickly between points.
- **Challenge:** Highly nonlinear dynamics, coupled joints, varying payloads.
- **Control Strategy:** Often uses decentralized PID for each joint. Advanced robots use computed-torque control (a model-based linearization technique).

Example 3: Conveyor System Synchronization

- **System:** Multiple conveyor belts in a production line.
- **Control Goal:** Maintain synchronized speed to prevent product jams or gaps.
- **Challenge:** Different motor characteristics, belt slippage, varying load.
- **Control Strategy:** Master-Slave configuration. Master sets speed reference; slaves use PID to follow, often with communication over a network (e.g., EtherCAT).

The Role of PLCs (Programmable Logic Controllers)

- A **PLC (Programmable Logic Controller)** is an industrial digital computer designed to automate electromechanical processes.
- **Core Characteristics:**
 - **Ruggedized:** Built to withstand harsh industrial environments.
 - **Programmable:** Uses a specialized language to define control logic.
 - **Deterministic:** Operates in a predictable scan cycle.
 - **I/O System:** Connects to sensors (inputs) and actuators (outputs) to monitor and control a process.



Industrial Automation: Trade-Offs

- **Performance:** High speed and precision required.
- **Robustness:** Must handle wear, minor load changes.
- **Cost/Complexity:** High component cost, but complexity is managed by using standardized, well-tuned PID solutions.

The background image shows a factory floor with several orange industrial robotic arms. One arm is in the foreground, and others are visible in the background, working on a production line. The scene is overlaid with a blue gradient.

Dynamic Performance, Stability in Unstructured Environments

Robotics Control

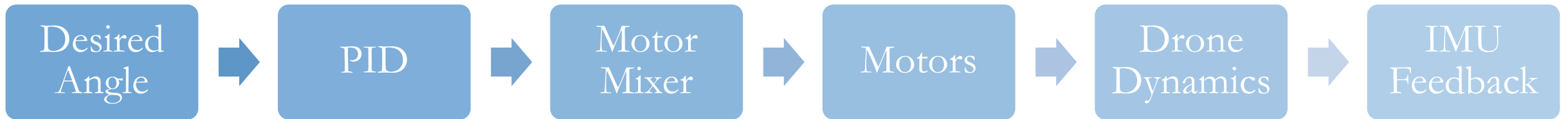
Background picture source: <https://ln.run/eQvOM>

Case Study: Autonomous Drone (UAV) Stability

- **System:** Quadcopter.
- **Control Goal:** Attitude stabilization (pitch, roll, yaw) and altitude hold.
- **Challenge:** Naturally unstable system, highly nonlinear, affected by wind gusts.
- **Control Strategy:** Fast, inner PID loops for attitude control. Outer loops for position/navigation.

Quadcopter Control Architecture

- A multi-loop diagram:
- **Inner Loop:**



- **Outer Loop:**



Drone PID Tuning in Action

- Show three response plots for the inner roll loop:
 1. Too much P: High-frequency oscillations.
 2. Too much D: Jerky, noisy response.
 3. Well-tuned: Fast, stable response to a step input.
- Key Message: Tuning is critical for stability.

Case Study: Self-Balancing Robot (Inverted Pendulum)

- **System:** Cart with a pendulum.
- **Control Goal:** Keep the pendulum upright.
- **Challenge:** Classic nonlinear, unstable system.
- **Control Strategy:** State-Space Control (Full-State Feedback - LQR) is common. PID can work for stabilization but not for swing-up.

Case Study: Mobile Robot Path Following

- **System:** Differential drive robot (like a Roomba).
- **Control Goal:** Follow a predefined path on the floor.
- **Challenge:** Non-holonomic constraints (can't move sideways), wheel slippage.
- **Control Strategy:** A "Pure Pursuit" or "Stanley" controller for steering.
Separate PID for velocity control.

Robotics Control: Trade-Offs

- **Performance:** Agility and fast response are key.
- **Robustness:** Extremely important for drones and autonomous systems.
Must handle significant uncertainties.
- **Cost/Complexity:** High complexity often demands advanced control (LQR, MPC) and powerful processors.

The background image shows an industrial control room. On the left, a monitor displays a complex process control interface with various data points and graphs. Below the monitor is a control panel with a yellow lightning bolt warning sign, a red emergency stop button, and other controls. The right side of the image shows a large industrial facility with several tall, vertical stainless steel tanks or columns. One of the tanks has the 'HRS' logo and the text 'hectic exchangers' on it. The entire scene is overlaid with a semi-transparent blue filter.

Stability, Safety, and Optimization over long periods.

Process Control

Background picture source: <https://ln.run/SSUNr>

Characteristics of Process Control

- Slow-moving processes (time constants of minutes or hours).
- High emphasis on safety and reliability.
- Loops often interact (MIMO systems).
- **Goal:** Regulate variables at setpoints for maximum efficiency and quality.

Case Study 1: Temperature Control in a Chemical Reactor

- **System:** Jacketed chemical reactor.
- **Control Goal:** Maintain a constant, precise temperature for a chemical reaction.
- **Challenge:** Large thermal inertia, time delays, cooling/heating dynamics are different.
- **Control Strategy:** PID control of coolant/heating flow. Often requires anti-windup protection.

Case Study 2: Liquid Level Control in a Tank

- **System:** Surge tank.
- **Control Goal:** Maintain level between limits despite inflow variations.
- **Challenge:** Integrator process (level will keep rising/falling if inflow \neq outflow).
- **Control Strategy:** Often uses P-only control! Why? To avoid instability and windup from integral action. Provides a "proportional band" which is often acceptable.

Case Study 3: Distillation Column

- **System:** A complex, multi-stage separation process.
- **Control Goal:** Maintain composition of top and bottom products.
- **Challenge:** Highly interactive MIMO system, long time delays.
- **Control Strategy:** Advanced Multi-loop PID or Model Predictive Control (MPC). A classic example where simple PID struggles.

Process Control: Trade-Offs

- **Performance:** Steady-state accuracy is often more critical than speed.
- **Robustness:** The highest priority. Failure can be dangerous and expensive.
- **Cost/Complexity:** Varies from simple P-only loops to highly complex, optimizing MPC systems.



Common Threads in Implementation.

Bringing It All Together

The Software Toolchain

- **Design & Simulation:** MATLAB/Simulink, Simscape
- **Implementation:** C/C++ on microcontrollers, Structured Text on PLCs, Python for prototyping.
- **Tuning & Diagnostics:** Oscilloscopes, Data loggers, SCADA (Supervisory Control and Data Acquisition) systems.

Cross-Cutting Themes Summary

- Sensing is Fundamental: Garbage In = Garbage Out.
- Actuator Limits Matter: Saturation is a reality.
- There's No Perfect Controller: It's always a compromise (revisit the triangle).
- Know Your System: The best controller is built on a deep understanding of the physical plant.



Summary

- We bridged the gap between control theory and real-world applications.
- Explored the unique challenges and solutions in Automation, Robotics, and Process Control.
- Highlighted the universal design trade-offs.
- Key Takeaway: You now possess the foundational knowledge to understand, analyze, and begin designing control systems for a vast array of modern technologies.

References

- [1] Chalachew Werku, 2025, Competing priorities in designing systems., Self-created
- [2] Simscape model of a drone,
https://www.mathworks.com/help/examples/simscape_shared/win64/QuadcopterDroneExample_03.png, accessed on November 4, 2025
 - Ogata, K. "Modern Control Engineering. (2010, Pearson)"
 - Norman S. Nise – Control Systems Engineering (2015, Wiley)