

IMC Controller and Lambda Tuning Method

Contributors

Name1: كريم أسامة السيد عبدالرحمن

Sec: 2

Name2: مصطفى المهدي أحمد عطوة

Sec: 2

1. Introduction to Internal Model Control (IMC)

Internal Model Control (IMC) is a powerful control design method based on the principle that if the controller contains a model of the **self regulatory process**, perfect control can theoretically be achieved. The controller uses this internal model to predict the plant's behavior and correct for disturbances.

The key idea in IMC is:

- Use a model to **simulate** the plant.
- Compare the real plant output to the model output.
- Correct the input based on the **difference** between actual output and predicted output.

2. Objective of IMC

The IMC controller is designed to achieve a desired closed-loop performance.

Suppose we want the closed-loop transfer function:

$$T(s) = 1 / (\lambda s + 1)$$

where:

- λ (lambda) is a **tuning parameter** controlling the **speed of response**.

3. Mathematical Derivation of the IMC Controller

Assume the plant is modeled as a **First-Order Plus Dead Time (FOPDT)** model:

$$G(s) = (K / (\tau s + 1)) e^{-\theta s}$$

Where:

- K = Process gain
- τ = Time constant
- θ = Dead time (delay)

We **cannot invert the dead time** (it is non-causal), so we only invert the delay-free part:

Approximate $G(s)$ as:

$$G^*(s) = K / (\tau s + 1)$$

Thus, the IMC controller is built on Ideal Process:

$$Q(s) = (G^*(s)^{-1})(1 / (\lambda s + 1))$$

then:

$$Q(s) = ((\tau s + 1) / K)(1 / (\lambda s + 1))$$

Thus:

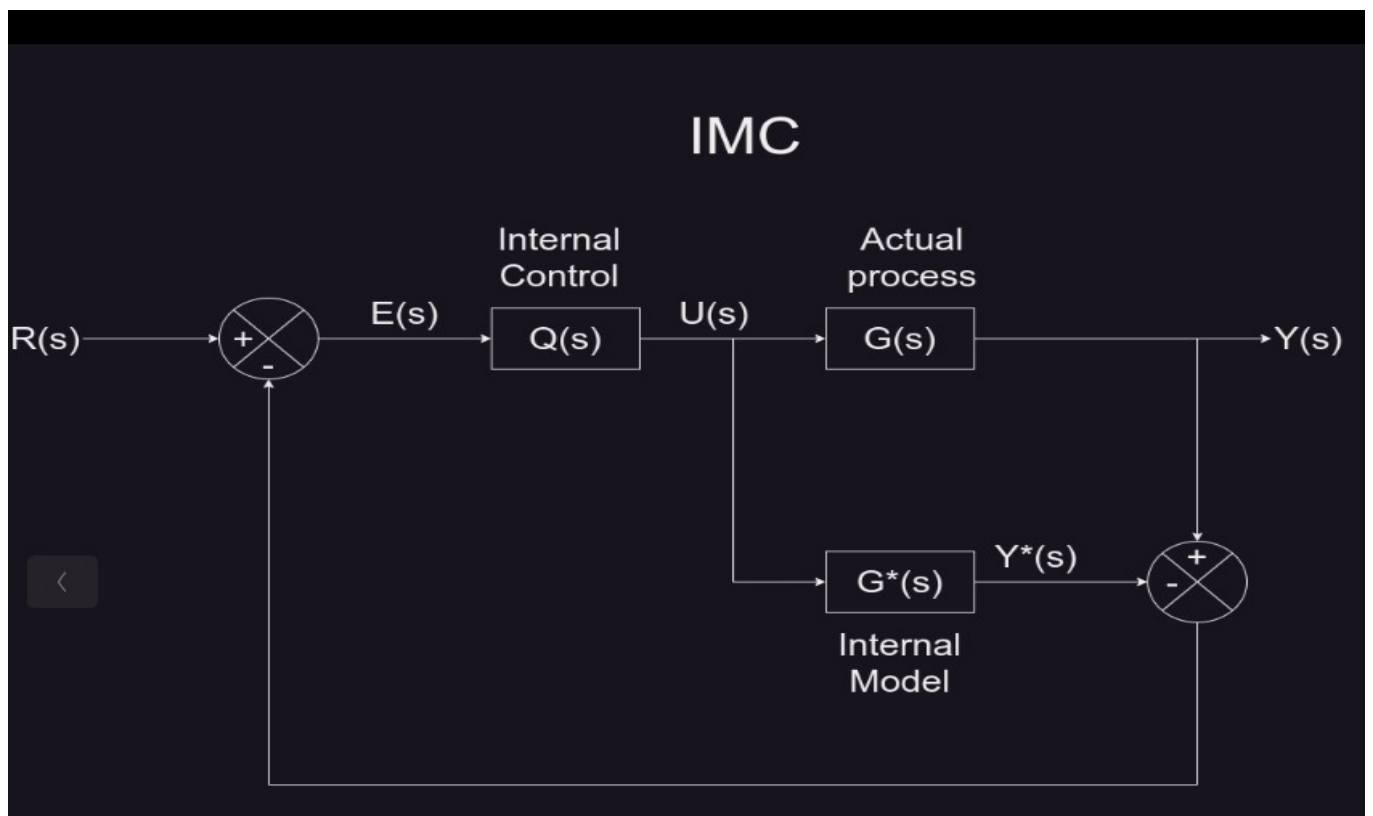
$$Q(s) = (\tau s + 1) / (K(\lambda s + 1))$$

4. IMC Structure

IMC Built on comparing theoretical output from $Q(s)G^*(s)$ with real output $Q(s)G(s)$ feedback them:

The IMC structure consists of:

- A controller $Q(s)$
- The actual process $G(s)$
- A theoretical model of the process $G^*(s)$

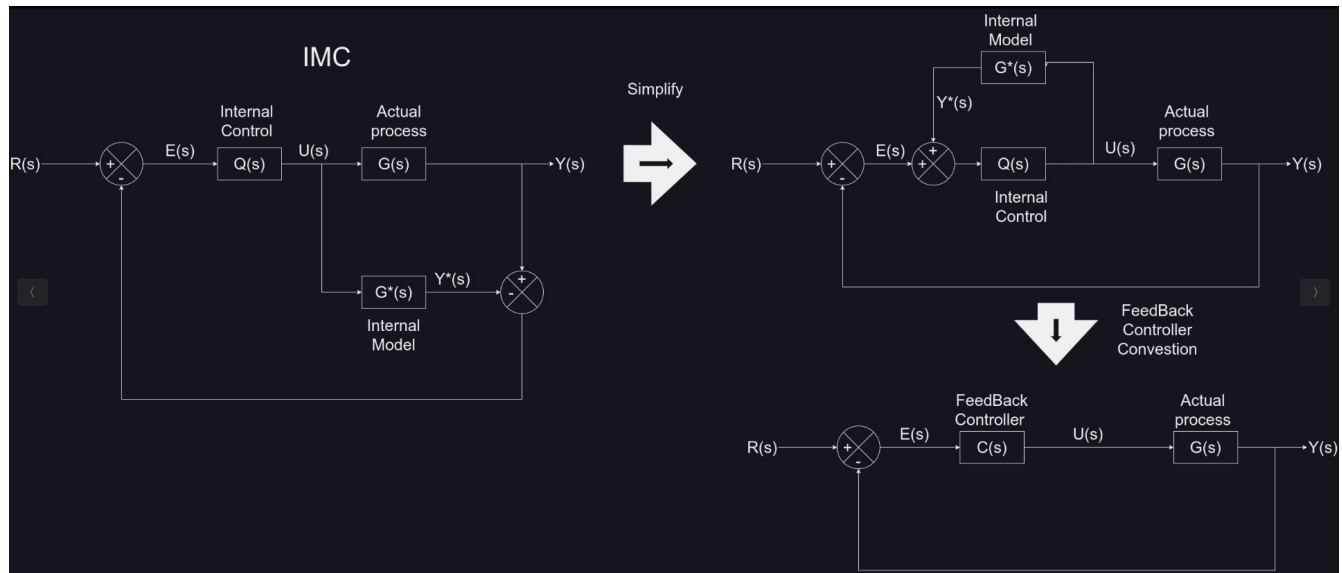


5. Conversion to Feedback Controller

We usually work with **feedback controllers** in practice. To convert IMC controller $Q(s)$ to a feedback controller $C(s)$, use:

$$C(s) = Q(s) / (1 - Q(s)G^*(s))$$

Delivered as:



6. Conversion to PID Controller

First Substituting $G(s)$ and $Q(s)$, and using a Padé approximation for the dead time:

$$e^{(-\theta s)} \approx (1 - \theta s/2) / (1 + \theta s/2)$$

The derivation (after some algebra) shows that $C(s)$ will have a **PID form**:

$$C(s) = K_c(1 + 1/(T_i * s) + T_d * s)$$

where K_c , T_i , and T_d are PID parameters.

The resulting PID parameters are:

- **Proportional Gain:**

$$K_c = (\tau + 0.5\theta) / K(\lambda + 0.5\theta)$$

- **Integral Time:**

$$T_i = \tau + 0.5\theta$$

- **Derivative Time:**

$$T_d = \tau\theta / (2\tau + \theta)$$

Notes:

- The dead time θ is included because it delays the system's response.
- λ is selected by the designer to tradeoff **performance** vs. **stability**.

7. Effect of λ (Tuning Parameter)

- **Small λ :** Fast response, but less stable (more sensitive to model errors).
- **Large λ :** Slower response, but much more stable

8. Application

1. Chemical & Process Industries

- **Distillation Columns:** Controls temperature, pressure, and composition in oil refineries and petrochemical plants.
- **Reactors:** Maintains optimal reaction conditions (temperature, pH, flow rates).
- **Heat Exchangers:** Regulates outlet temperatures in heat transfer processes.
- **CSTR (Continuous Stirred-Tank Reactors):** Ensures stable chemical reactions with minimal oscillations.

2. Oil & Gas Industry

- **Pipeline Flow Control:** Manages fluid transport with dead-time compensation.
- **Pressure & Level Control:** Stabilizes storage tanks and separators.
- **Refining Processes:** Optimizes fractional distillation units.

3. Food & Beverage Industry

- **Pasteurization:** Maintains precise temperature control in dairy and beverage processing.
- **Fermentation Tanks:** Regulates conditions for consistent product quality (e.g., beer, yogurt).

4. Pharmaceutical Manufacturing

- **Bioreactors:** Controls dissolved oxygen, pH, and temperature in drug production.
- **Lyophilization (Freeze Drying):** Manages shelf temperature and vacuum pressure.

Conclusion

The **IMC Lambda tuning method** provides a systematic way to tune PID controllers starting from the model of the plant.

By adjusting λ , we can control the tradeoff between speed and stability easily.

The method leads to simple, practical formulas that account for dead time and time constants, making it widely used in industrial PID tuning.