Controller Design for Steady-State Error Specifications

Compensating Error. Anti-Windup Control.

M.V. Iordache, *EEGR3523 Mechatronics*, Spring 2021, LeTourneau University

The Steady-State Error

- Without loss of generality, it is typically studied on systems with unity feedback.
- Assuming a constant steady-state error, its value can be found with the Final-Value Theorem.
- The result depends on the dc gain of the transfer functions and notably, on the loop transfer function.
  - The dc gain of $G(s)$ is $G(0)$.
  - The loop transfer function $L(s)$ is the open-loop transfer function of the system.

The Steady-State Error

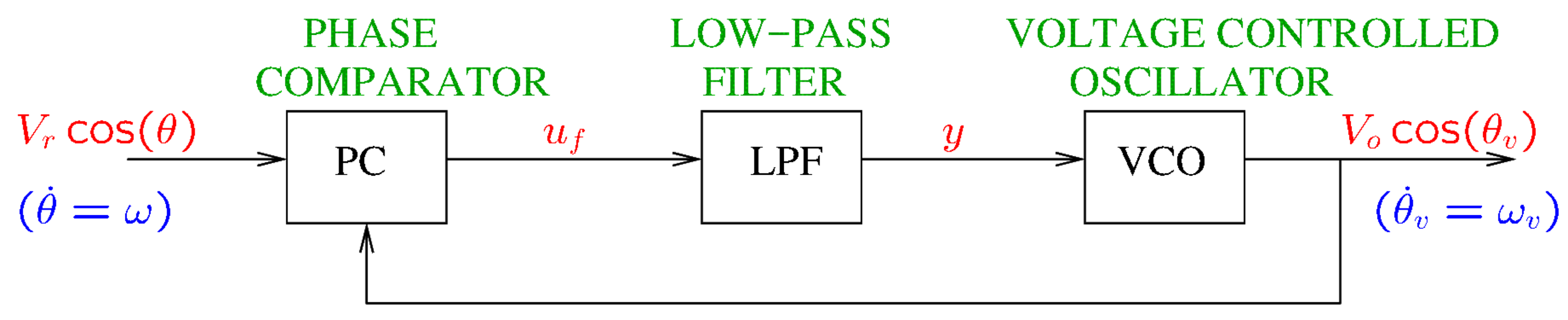
- The steady-state error is zero when the system is stable and either:
  - The input has no poles outside of the LHP.
  - The closed-loop TF cancels the poles of the input that are outside of the LHP (Internal Model Principle).

The Steady-State Error

- In a type-n system $L(s)$ has n poles at 0.
  - The definition is for systems with unity dc gain feedback.
  - The steady-state error depends on the input and the error constant $K = \lim_{s \to 0} s^n L(s)$.

		Steady-state error e_{ss}		
INPUT		$n = 0$	$n = 1$	$n = 2$
Unit Step	$r(t) = aU(t)$	$\frac{a}{1+K}$	0	0
Ramp	$r(t) = at$	∞	$\frac{a}{K}$	0
Parabolic	$r(t) = \frac{at^2}{2}$	∞	∞	$\frac{a}{K}$

Example: Phase-Locked Loop



- The analysis neglects the effect of higher-order harmonics (they are eliminated by the filter).

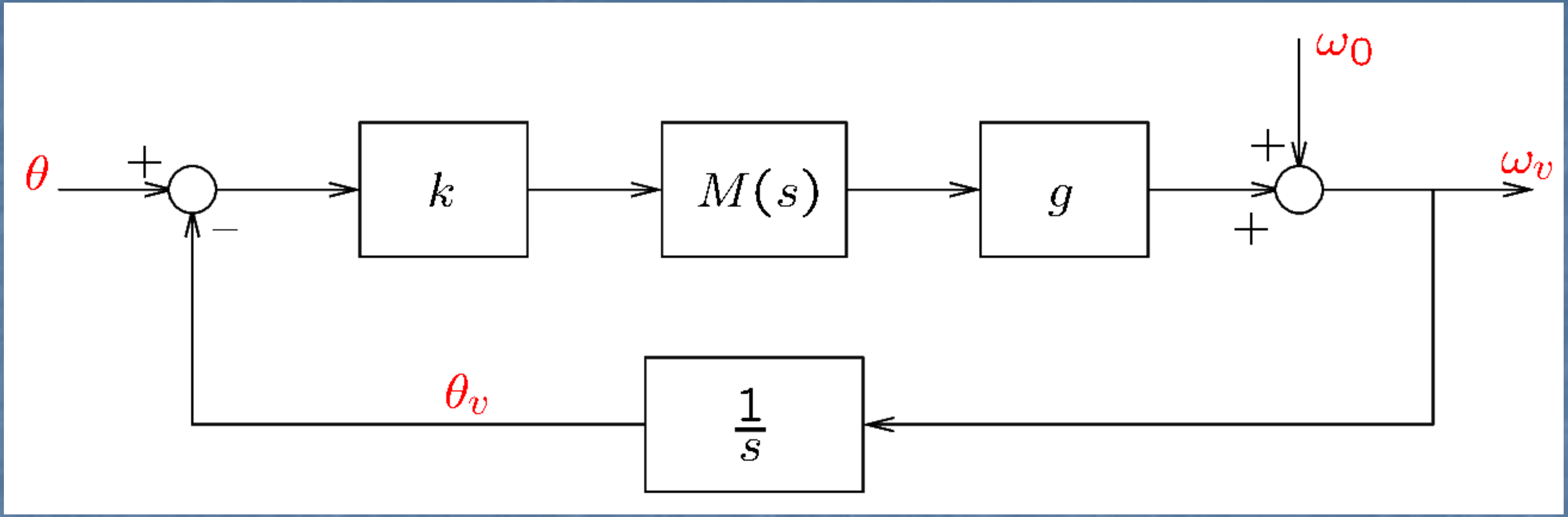
$$u_f = k(\theta - \theta_v)$$

- The low-pass filter could be represented by:

$$Y = M(s)U_f$$

- The VCO outputs $\omega_v = \omega_0 + gy$, where $\dot\theta_v = \omega_v$.

Example: Phase-Locked Loop



- Assume the input $\theta = \omega t + \theta_0$.
- The error is:

$$E(s) = \frac{\Theta(s) - \omega_0/s}{1 + gkM/s}$$

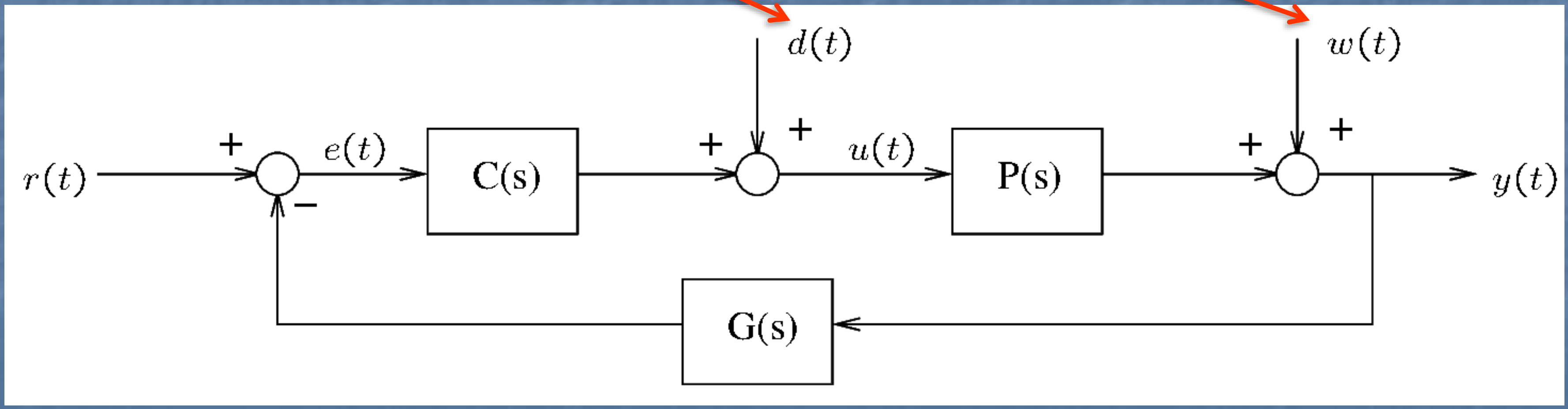
- The steady-state error is $e_{ss} = \frac{\omega - \omega_0}{kgM(0)}$.

The Steady-State Error

- A constant non-zero error could be corrected by pre-multiplying the input with a constant gain. However:
  - This is not a robust solution: system parameters can change somewhat in time.
  - This will not compensate disturbances.
- Sources of disturbances:
  - Unknown inputs from the environment (e.g. wind, temperature)
  - Modeling errors (e.g. linearization error)
  - Unmodeled phenomena (e.g. friction)

The Steady-State Error

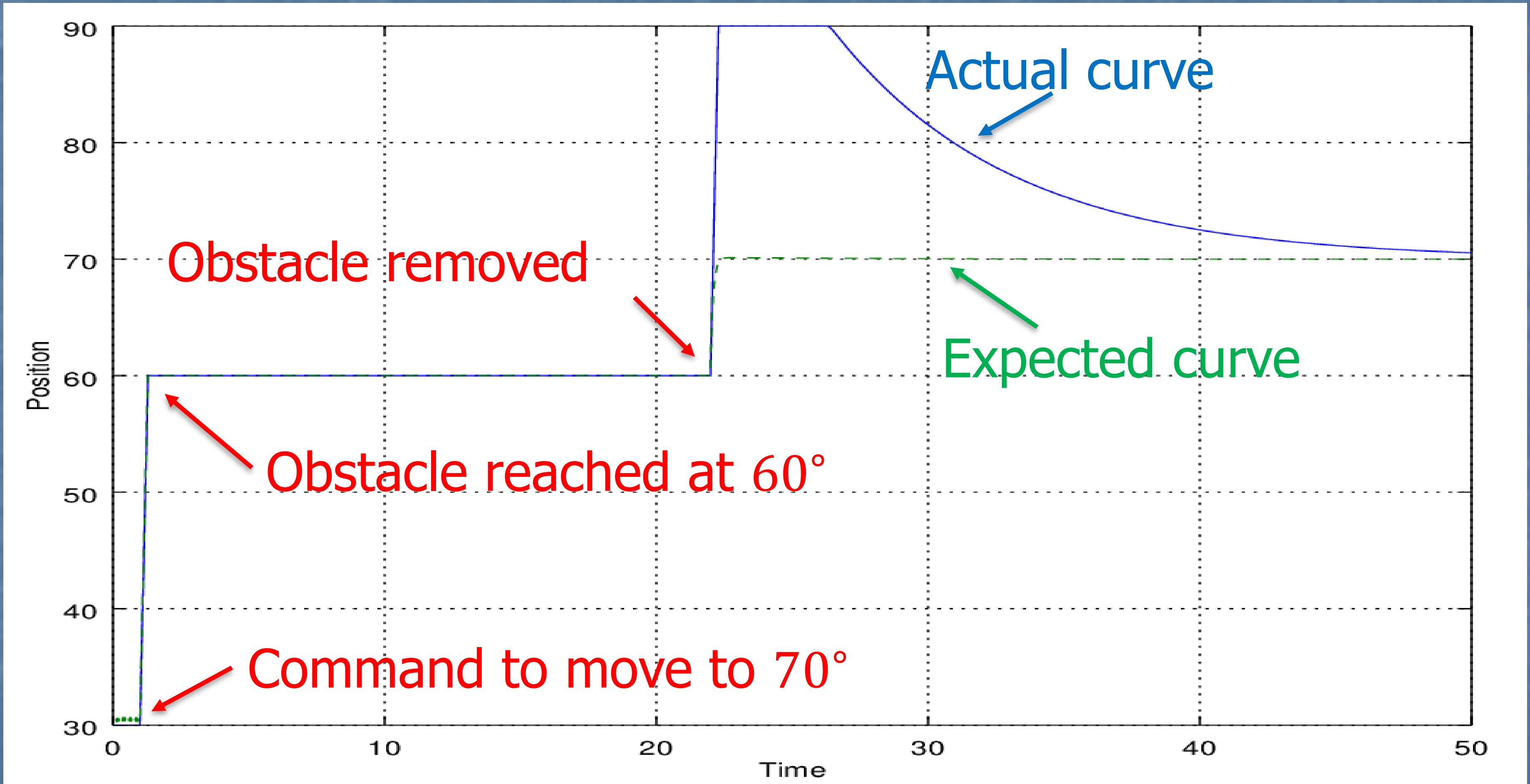
- Error is affected by disturbances.
- Disturbances can be modeled as
  - Output disturbances
  - Input disturbances



Implementation—Example

- Assume a robotic arm is physically limited to angles in the range $-90° \ldots + 90°$.
- Position is controlled with a PID controller.
- The initial angle is $30°$.
- At time $t = 1$, the command is to move to $70°$.
- However, an obstacle keeps the arm at $60°$.
- Eventually the obstacle is removed.
- Will the arm move immediately to $70°$?

Implementation—Example



Anti-Windup Control

- In the example, the problem is caused by *integrator windup* (also known as *reset windup*).
- The solution is to use an *anti-windup* method.
- In general, *anti-windup control* is necessary due to *actuator saturation*.

Anti-Windup Control by Clamping

- Set a low and a high limit for the control signal.
- Use regular PID control as long as the control signal is within the set limits.
- Disable integration when integration would make the control signal go further beyond the set limits.

- In this way integration is enabled only when needed (when the error is reasonably small and can be corrected).