



Centurion
UNIVERSITY
Shaping Lives...
Empowering Communities...

System Modeling & Control

Presented by

Prof. Amit Kumar Sahoo
CUTM, BBSR

Steady-State Errors in Unity-Feedback Control Systems

Steady-state error: Steady-state error is the difference between the input and the output for a prescribed test input as time approaches to infinity.

Consider the unity-feedback control system with the following open-loop transfer function $G(s)$:

$$G(s) = \frac{K(T_a s + 1)(T_b s + 1) \cdots (T_m s + 1)}{s^N (T_1 s + 1)(T_2 s + 1) \cdots (T_p s + 1)}$$

A system is called **type 0**, **type 1**, ..., if $N = 0, N = 1, N = 2, \dots$, respectively.

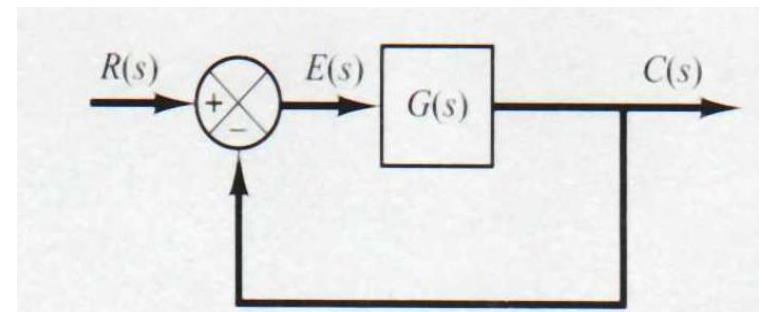


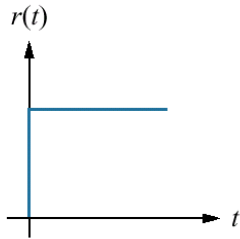
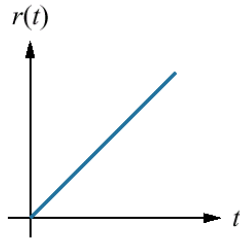
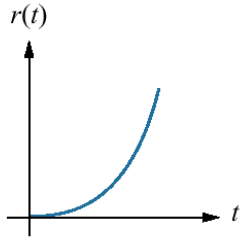
Figure 5-17 Unity-feedback control system.



**Centurion
UNIVERSITY**

*Shaping Lives...
Empowering Communities...*

Table 5-3 Test waveforms for evaluating steady-state errors of position control systems

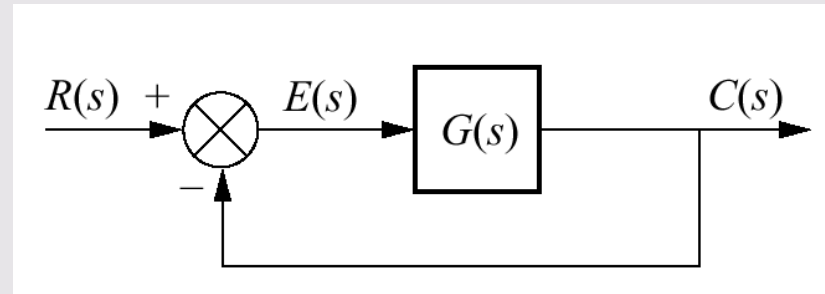
Waveform	Name	Physical interpretation	Time function	Laplace transform
	Step	Constant position	1	$\frac{1}{s}$
	Ramp	Constant velocity	t	$\frac{1}{s^2}$
	Parabola	Constant acceleration	$\frac{1}{2}t^2$	$\frac{1}{s^3}$



Centurion
UNIVERSITY

*Shaping Lives...
Empowering Communities...*

Figure 5-18 Unity-feedback control system.



$$E(s) = R(s) - C(s)$$

$$C(s) = R(s)T(s)$$

$$E(s) = R(s)[1 - T(s)] = R(s) \left[1 - \frac{G(s)}{1 + G(s)} \right] = R(s) \frac{1}{1 + G(s)}$$

Applying the final value theorem, which allows us to use the final value of $e(t)$ without taking the inverse Laplace transform of $E(s)$, and then letting t approach infinity, we obtain

$$e(\infty) = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s) = \lim_{s \rightarrow 0} sR(s)[1 - T(s)] = \lim_{s \rightarrow 0} \frac{sR(s)}{1 + G(s)}$$

Static Position Error Constant

The steady-state error of the system for a unit-step input is

$$e_{ss} = \lim_{s \rightarrow 0} \frac{s}{1 + G(s)} \cdot \frac{1}{s} = \frac{1}{1 + G(0)}$$

The static position error constant K_p is defined by

$$K_p = \lim_{s \rightarrow 0} G(s) = G(0)$$

Thus, the steady-state error in terms of the static position error constant is given by

$$e_{ss} = \frac{1}{1 + K_p}$$

For a type 0 system,

$$K_p = \lim_{s \rightarrow 0} \frac{K(T_a s + 1)(T_b s + 1) \cdots}{(T_1 s + 1)(T_2 s + 1) \cdots} = K$$



Centurion
UNIVERSITY

Shaping Lives...
Empowering Communities

For a type 1 or higher system,

$$K_p = \lim_{s \rightarrow 0} \frac{K(T_a s + 1)(T_b s + 1) \cdots}{s^N (T_1 s + 1)(T_2 s + 1) \cdots} = \infty \quad \text{for } N \geq 1$$

Hence, for a type 0 system, the static position error constant K_p is finite, while for a type 1 or higher system, K_p is infinite.

For a unit-step input, the steady-state error e_{ss} may be summarized as follows:

$$e_{ss} = \frac{1}{1 + K}, \quad \text{for type 0 system}$$

$$e_{ss} = 0, \quad \text{for type 1 or higher systems}$$

Static Velocity Error Constant

The steady-state error of the system with a unit-ramp input is given by

$$e_{ss} = \lim_{s \rightarrow 0} \frac{s}{1 + G(s)} \cdot \frac{1}{s^2} = \lim_{s \rightarrow 0} \frac{1}{sG(s)}$$

The static velocity error constant K_v is defined by

$$K_v = \lim_{s \rightarrow 0} sG(s)$$

Thus, the steady-state error in terms of the static velocity error constant is given by

$$e_{ss} = \frac{1}{K_v}$$

For a type 0 system,

$$K_v = \lim_{s \rightarrow 0} \frac{sK(T_a s + 1)(T_b s + 1) \cdots}{(T_1 s + 1)(T_2 s + 1) \cdots} = 0$$

For a type 1 system,

$$K_v = \lim_{s \rightarrow 0} \frac{sK(T_a s + 1)(T_b s + 1) \cdots}{s(T_1 s + 1)(T_2 s + 1) \cdots} = K$$

For a type 2 or higher system,

$$K_v = \lim_{s \rightarrow 0} \frac{sK(T_a s + 1)(T_b s + 1) \cdots}{s^N (T_1 s + 1)(T_2 s + 1) \cdots} = \infty, \quad \text{for } N \geq 2$$

The steady-state error e_{ss} for the unit-ramp input can be summarized as follows:

$$e_{ss} = \frac{1}{K_v} = \infty, \quad \text{for type 0 system}$$

$$e_{ss} = \frac{1}{K_v} = \frac{1}{K}, \quad \text{for type 1 system}$$

$$e_{ss} = \frac{1}{K_v} = 0, \quad \text{for type 2 or higher systems}$$



Centurion
UNIVERSITY

Shaping Lives...
Empowering Communities

Static Acceleration Error Constant

The steady-state error of the system with a unit-parabolic input, which is defined by

$$r(t) = \frac{t^2}{2}, \quad \text{for } t \geq 0$$
$$= 0, \quad \text{for } t < 0$$

is given by

$$e_{ss} = \lim_{s \rightarrow 0} \frac{s}{1 + G(s)} \cdot \frac{1}{s^3} = \lim_{s \rightarrow 0} \frac{1}{s^2 G(s)}$$

The static acceleration error constant K_a is defined by the equation

$$K_a = \lim_{s \rightarrow 0} s^2 G(s)$$

The steady-state error is then

$$e_{ss} = \frac{1}{K_a}$$



Centurion
UNIVERSITY

*Shaping Lives...
Empowering Communities...*

For a type 0 system,

$$K_a = \lim_{s \rightarrow 0} \frac{s^2 K (T_a s + 1)(T_b s + 1) \cdots}{(T_1 s + 1)(T_2 s + 1) \cdots} = 0$$

For a type 1 system,

$$K_a = \lim_{s \rightarrow 0} \frac{s^2 K (T_a s + 1)(T_b s + 1) \cdots}{s (T_1 s + 1)(T_2 s + 1) \cdots} = 0$$

For a type 2 system,

$$K_a = \lim_{s \rightarrow 0} \frac{s^2 K (T_a s + 1)(T_b s + 1) \cdots}{s^2 (T_1 s + 1)(T_2 s + 1) \cdots} = K$$

For a type 3 or higher system,

$$K_v = \lim_{s \rightarrow 0} \frac{s^2 K (T_a s + 1)(T_b s + 1) \cdots}{s^N (T_1 s + 1)(T_2 s + 1) \cdots} = \infty, \quad \text{for } N \geq 3$$



Centurion
UNIVERSITY

*Shaping Lives...
Empowering Communities...*

Thus, the steady-state error for the unit parabolic input is

$$e_{ss} = \frac{1}{K_a} = \infty, \quad \text{for type 0 and type 1 systems}$$

$$e_{ss} = \frac{1}{K_a} = \frac{1}{K}, \quad \text{for type 2 system}$$

$$e_{ss} = \frac{1}{K_a} = 0, \quad \text{for type 3 or higher systems}$$

Table 5-3 Steady-state error in terms of gain K.

	Step Input $r(t) = 1$	Ramp Input $r(t) = t$	Acceleration Input $r(t) = \frac{1}{2}t^2$
Type 0 system	$\frac{1}{1 + K}$	∞	∞
Type 1 system	0	$\frac{1}{K}$	∞
Type 2 system	0	0	$\frac{1}{K}$



Centurion
UNIVERSITY

Shaping Lives...
Empowering Communities...

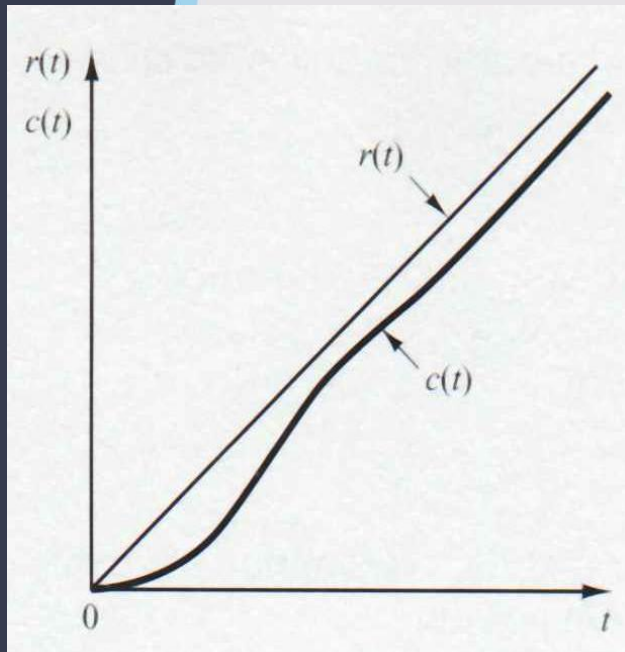


Figure 5-19 Response of a type 1 unity-feedback system to a ramp input.

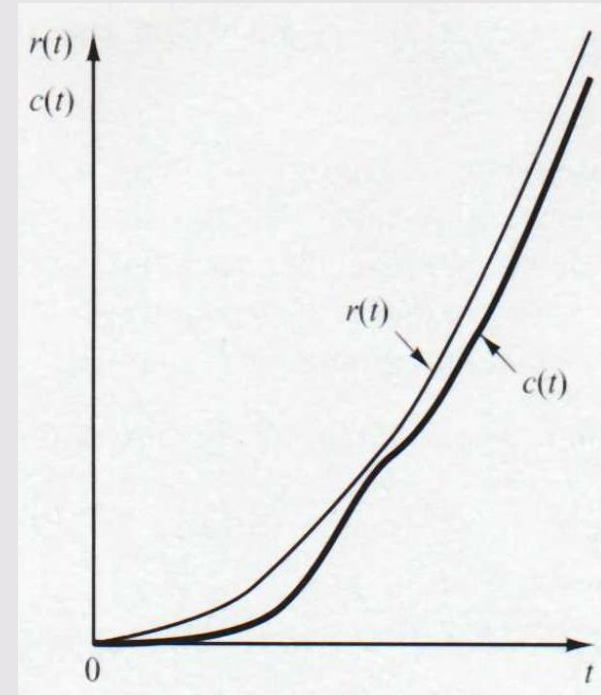


Figure 5-20 Response of a type 2 unity-feedback system to a parabolic input.