s – Domain Design Procedure

- *Design Task Description*
- *s-Domain Design Philosophy*
- *Root Locus Concept*

Design Task Description

The **design task**, in a broad sense, is to **arrive at G_c,** which **meets** the stated closed loop **requirements.**

In the context of control **actions** and elements discussed previously, this **involves** determining the values of **three gains**, K_P, K_D, K_I as well as controller **type** 'k'

Thus, a **general** G_c is the **PID** controller, as given below.

$$G_c(s) = K_P + K_D s + \frac{K_I}{s}$$

Design Task Description

In many cases, the **controller** is reformulated in **terms** of zeros, type and a single gain, as **described** below.

$$G_c(s) = \frac{K(s+z_1)(s+z_2)}{s} = K(z_1+z_2) + Ks + \frac{Kz_1z_2}{s}$$

We see that either of the **forms** can create, P, PD, PI or PID **controllers** by suitably choosing the **values** of the gains or zeros.

s – Domain Design Task

The design task in **s-domain** is to arrive at the controller **gains** so as to achieve the **desired** dominant poles.

This can be done with **Routh's** method by invoking the **relative** stability procedures, if only '**σ**' is to be ensured.

However, as our **benchmark** closed loop requires both **'σ' and 'ω_d',** we need a **strategy** which provides **both** these for the dominant **poles.**

Root locus is such a procedure for **extracting** dominant closed loop **poles**, by varying 'K'.

Root Locus Concept

Root Locus is defined as **paths** of all closed loop **poles,** as gain increases from **0 to +∞.**

In this **method**, we solve the **closed** loop characteristic **equation** for extracting the **dominant** poles, though **non-dominant** poles are also **extracted** as a result.

Root locus works on **gain – zeros** form of the **controller,** in order to **admit** all controllers.

The **method** was proposed by W.R. **Evans** around 1955**.**

Root Locus Concept

Consider the **characteristic** equation of a unity feedback **proportional control** system as given below.

$$\boxed{D'(s) = 1 + KG_c(s)G(s) = 0 \rightarrow KG_c(s)G(s) = -1 = -1 + j0}$$

Here, $\mathbf{G_c}$ is the **zero-pole** part, without gain, **K.**

We see that values of '**s**' satisfying the above **constraint**, trace out a **path** in s-plane denoting closed loop **poles**.

However, in order to **minimize** computational **effort,** above path is generated using **geometric** conditions.

Summary

s – domain design procedure **aims** to arrive at **controller** to achieve the **dominant** closed loop poles.