CONTROL System

Signal Flow Graph

Er. Yastuti Rao Gautam
Asst. Prof.
Mech. Engg. Dept.
U.I.E.T. Kanpur
C.S.J.M.U. Kanpur

Signal Flow Graph

Signal flow graph is a graphical representation of algebraic equations .
Nodes and branches are the basic elements of signal flow graph.

When applying the signal flow graph method to analysis of control systems, we must first transform linear differential equations into algebraic equations.

Basic Elements of Signal Flow Graph
Nodes and branches are the basic elements of signal flow graph

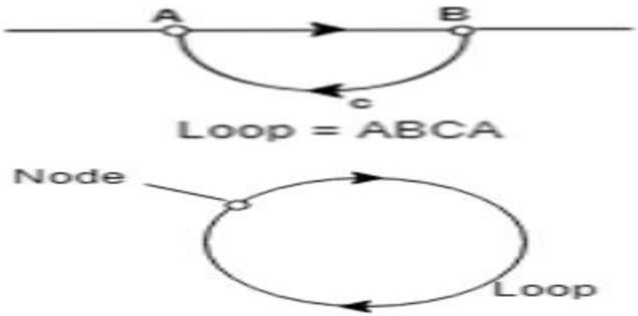
Node is a point which represents either a variable or a signal. There are three types of nodes — input node, output node and mixed node.

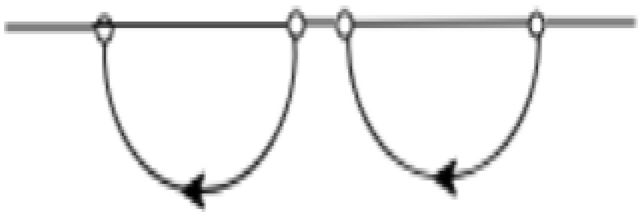
- **Input Node –** It is a node, which has only outgoing branches.
- **Output Node –** It is a node, which has only incoming branches.
- **Mixed Node –** It is a node, which has both incoming and outgoing branches.

Path: a path is a traversal of connected branches in the direction of the branch arrows. The path should not cross a node more than once

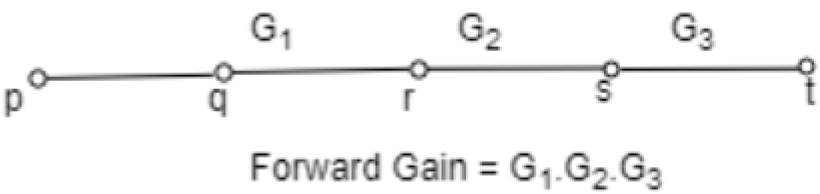
Input node or source: It is the node which have only outgoing branches.
Output node or sink: It is a node which has only incoming branches.

Forward Path: It is a path from an input node to an output node in the direction of branch arrow.
Loop: It is a path that starts and ends at the same node.

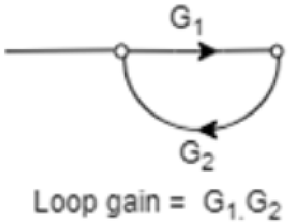




Non-touching loop: Loop is said to be non-touching if they do not have any common node.
Forward path gain: A product of all branches gain along the forward path is called Forward path gain.



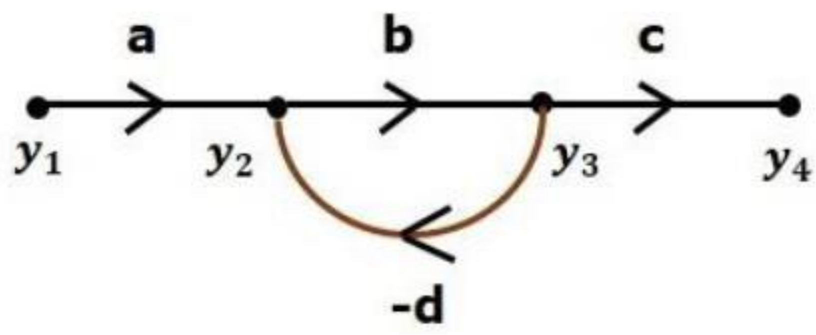
Loop Gain: Loop gain is the product of branch gain which travels in the loop.



Individual loop: it is a closed path starting from a node and after passing through a certain part of a graph arrives at same node without crossing any node more than once

Example

Let us consider the following signal flow graph to identify these nodes.



- The **nodes** present in this signal flow graph are **y1, y2, y3** and **y4**.
- **y1** and **y4** are the **input node** and **output node** respectively.
- **y2** and **y3** are **mixed nodes**

Branch

Branch is a line segment which joins two nodes. It has both **gain** and **direction**. For example, there are four branches in the above signal flow graph. These branches have **gains** of **a, b, c** .

Comparison of block diagram and signal flow graph method

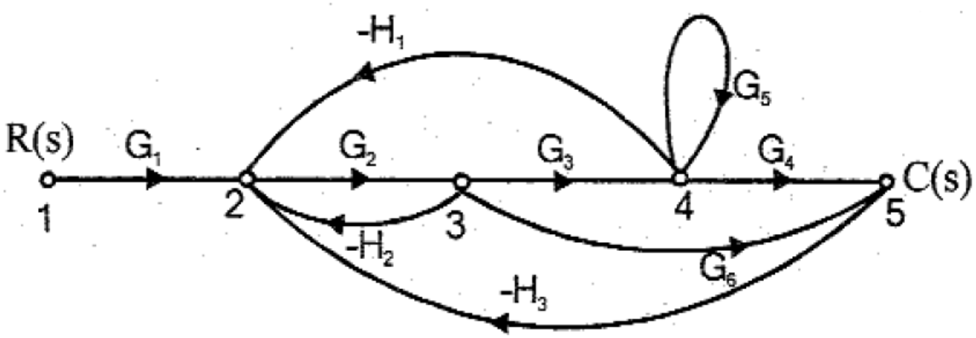
Sl.No	Block diagram	SFG
1	applicable to Linear time invariant systems	applicable to Linear time invariant systems
2	each element is represented by block	each variable is represented by node
3	summing point and take off points are separate	summing and take off points are absent
4	self-loop do not exist	self-loop can be exist
5	it is time consuming method	require less time by using Mason gain formula
6	block diagram is required at each and every step	at each step it is not necessary to draw SFG
7	Only transfer function of the element is shone inside the corresponding block	transfer function is shown along the branches connecting the nodes
8	feedback path is present	feedback loops are used

SIGNAL FLOW GRAPH (SFG)

MASON'S GAIN FORMULA

- The signal flow graph is used to represent the control system graphically and it was developed by S J mason .

- The advantage in signal flow graph method is that, using **Mason's gain formula** the overall gain of the system can be computed easily.



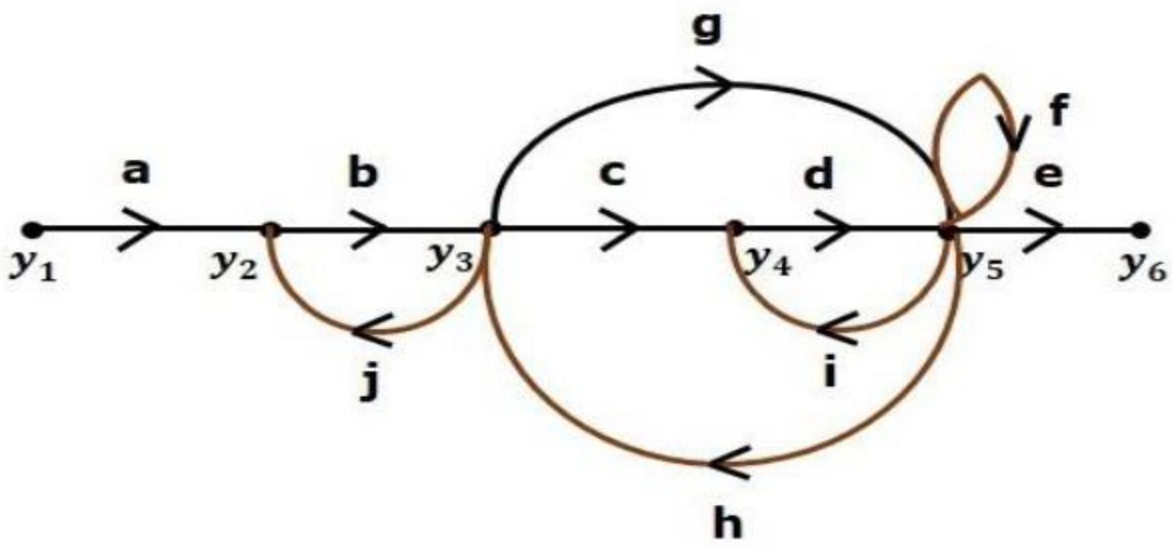
Mason's Gain formula

Suppose there are 'N' forward paths in a signal flow graph. The gain between the input and the output nodes of a signal flow graph is nothing but the **transfer function** of the system. It can be calculated by using Mason's gain formula.

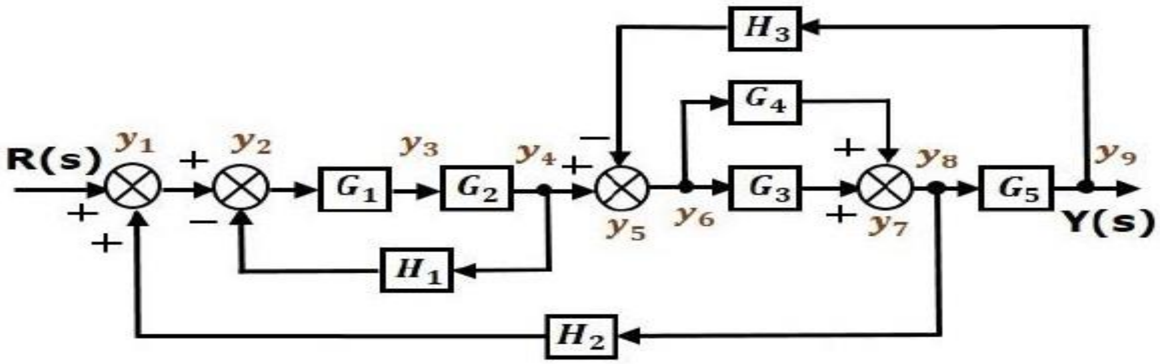
$$T = \frac{C(s)}{R(s)} = \frac{\sum_{K=1}^{N} P_k \Delta_k}{\Delta}$$

Where,

- **C(s)** is the output node
- **R(s)** is the input node
- **T** is the transfer function or gain between R(s) and C(s)
- $\boldsymbol{P_k}$ is the k^{th} forward path gain
- Δ=1−(sum of all individual loop gains) +(sum of gain products of all possible two non-touching loops) −(sum of gain products o fall possible three non-touching loops)+...
- $\Delta_{\boldsymbol{k}}$ *is obtained from Δ by removing the loops which are touching the* k^{th}*forward path (Associated path factor depend on forward path and isolated loop)*



Path
It is a traversal of branches from one node to any other node in the direction of branch arrows. It should not traverse any node more than once.
Examples − y2→y3→y4→y5 and y5→y3→y2
Forward Path
The path that exists from the input node to the output node is known as **forward path**.
Examples − y1→y2→y3→y4→y5→y6 and y1→y2→y3→y5→y6
Forward Path Gain
It is obtained by calculating the product of all branch gains of the forward path.
Examples − a-b-c-d-e is the forward path gain of y1→y2→y3→y4→y5→y6 and a-b-g-e is the forward path gain of y1→y2→y3→y5→y6
Loop
The path that starts from one node and ends at the same node is known as **loop**. Hence, it is a closed path.
Examples − y2→y3→y2 and y3→y5→y3
Loop Gain
It is obtained by calculating the product of all branch gains of a loop.
Examples − bj is the loop gain of y2→y3→y2 and gh is the loop gain of y3→y5→y3
Non-touching Loops
These are the loops, which should not have any common node.
Examples − The loops, y2→y3→y2 and y4→y5→y4 are non-touching.


R(s)
y1
y2
y3
y4
y5
y6
y7
y8
y9
G1
G2
G3
G4
G5
H1
H2
H3
Y(s)