EXPERIMENT No. 6

Signal Flow Graph Analysis using Mason's Gain Formula and Verification through Simulink

1. Aim

To determine the overall transfer function C/R of a given Signal Flow Graph (SFG) analytically using Mason's Gain Formula, and to verify the result by constructing an equivalent block diagram of the same system in MATLAB/Simulink.

2. Software Required

MATLAB with Simulink toolbox

3. Theory

A Signal Flow Graph is a graphical representation of a set of linear algebraic equations describing a system, in which nodes represent variables and directed branches (arrows) represent the gain relating two variables. Mason's Gain Formula provides a systematic, non-iterative way of obtaining the overall transfer function between an input (source) node and an output (sink) node directly from the topology of the graph, without needing to reduce the graph block by block.

Mason's Gain Formula is stated as: $T = C/R = (1/\Delta)\ \Sigma_k\ P_k\ \Delta_k$

where:

- P_k = gain of the k-th forward path (a path from input to output that does not touch any node more than once)
- Δ = graph determinant = 1 − (Σ individual loop gains) + (Σ gain products of two non-touching loops) − (Σ gain products of three non-touching loops) + …
- Δ_k = the cofactor for forward path k, obtained from Δ by deleting all loops that touch (share a node with) that forward path

Two loops are said to be non-touching if they do not share any common node. Similarly, a loop and a forward path are non-touching if they share no common node.

4. Given Signal Flow Graph

The signal flow graph to be analysed is shown below. The input is R and the output is C.

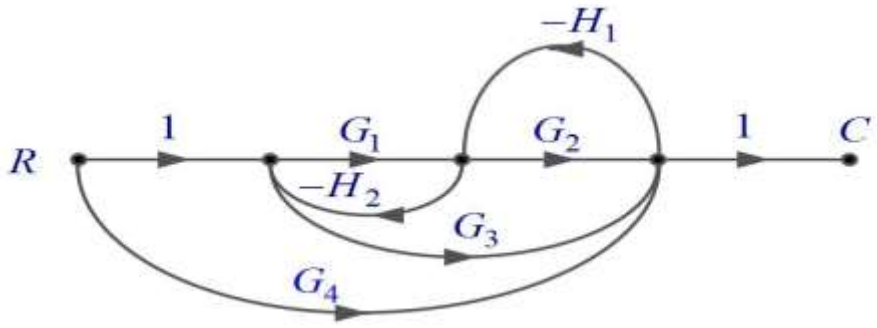

Fig. 1: Given Signal Flow Graph

For analysis, the internal nodes are labelled N1 (junction after the unity gain from R and after the −H2 feedback), N2 (junction after G1, where −H1 feedback arrives and −H2 / G3 branches leave), and N3 (junction after G2/G3, where −H1 leaves, G4 arrives, and the path proceeds to C).

5. Procedure

Part A: Analytical Solution using Mason's Gain Formula

1. Identify all forward paths from R to C (paths that touch every node at most once).

Forward paths:

$P_1 = 1 \cdot G_1 \cdot G_2 \cdot 1 = G_1G_2$ *(R → N1 → N2 → N3 → C)*

$P_2 = 1 \cdot G_1 \cdot G_3 \cdot 1 = G_1G_3$ *(R → N1 → N2 → N3 → C, via G3)*

$P_3 = G_4 \cdot 1 = G_4$ *(R → N3 → C, direct)*

2. Identify all individual loops in the graph.

Loops:

$L_1 = G_2 \cdot (-H_1) = -G_2H_1$ *(N2 → N3 → N2)*

$L_2 = G_1 \cdot (-H_2) = -G_1H_2$ *(N1 → N2 → N1)*

$L_3 = G_3 \cdot (-H_1) = -G_3H_1$ *(N2 → N3 → N2, via G3)*

3. Check every pair of loops for non-touching status.

L1 and L2 both include node N2 — touching. L2 and L3 both include node N2 — touching. L1 and L3 both include nodes N2 and N3 — touching. Every pair of loops shares node N2, so there are no non-touching loop combinations.

4. Compute the graph determinant Δ.

$$\Delta = 1 - (L_1 + L_2 + L_3) = 1 - (-G_2H_1 - G_1H_2 - G_3H_1)$$

$$\Delta = 1 + G_2H_1 + G_1H_2 + G_3H_1$$

5. Compute the cofactor Δ_k for each forward path (delete loops touching that path).

P1 passes through N1, N2, N3 — it touches every loop, so $\Delta_1 = 1$.

P2 passes through N1, N2, N3 — it also touches every loop, so $\Delta_2 = 1$.

P3 passes through only R and N3 (it does not touch N1 or N2). L2 does not touch N3, so L2 is non-touching with P3. L1 and L3 do touch N3, so they are removed.

$$\Delta_3 = 1 - L_2 = 1 - (-G_1H_2) = 1 + G_1H_2$$

6. Apply Mason's Gain Formula to obtain the overall transfer function.

$$\frac{C}{R} = \frac{P_1\Delta_1 + P_2\Delta_2 + P_3\Delta_3}{\Delta}$$

$$\frac{C}{R} = \frac{G_1G_2 + G_1G_3 + G_4(1 + G_1H_2)}{1 + H_1(G_2 + G_3) + G_1H_2}$$

This is the final closed-form transfer function to be verified through simulation.

Part B: Constructing the Equivalent Model in Simulink

Since Simulink does not have a direct "node" primitive, every SFG node that receives more than one incoming signal is realised using a Sum block, and every branch gain is realised using a Gain block. The equivalent block diagram is shown in Fig. 2.

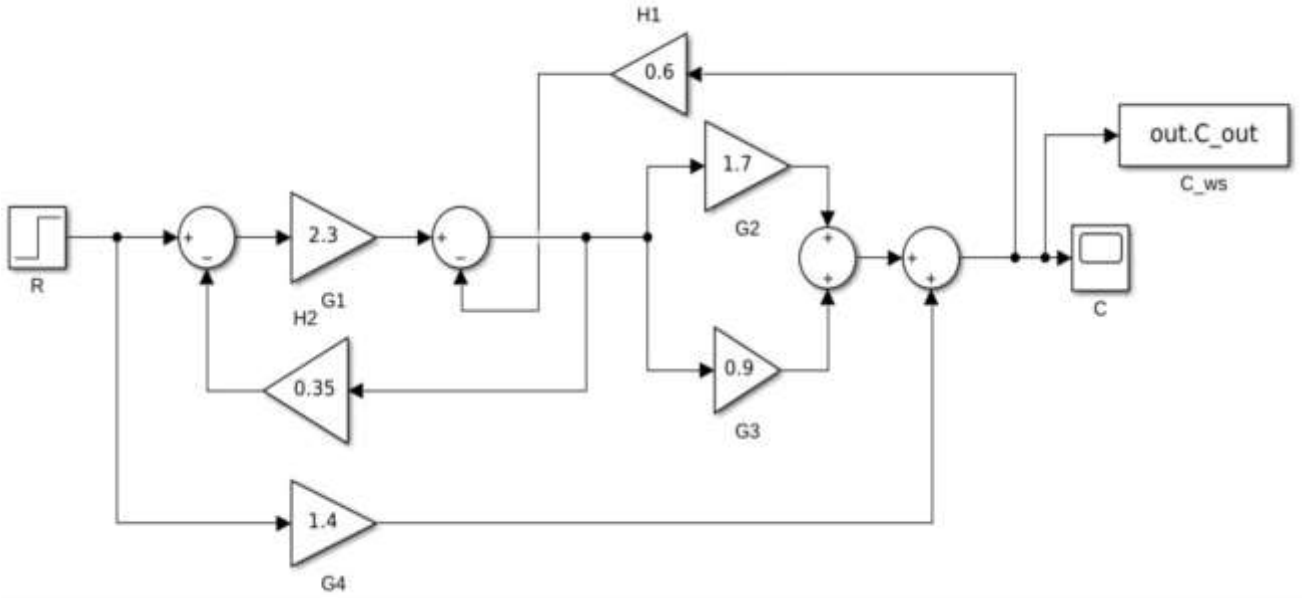

Fig. 2: Simulink block-diagram equivalent of the given SFG

1. Open Simulink and create a new blank model.
2. Place a Step block (Simulink → Sources) to represent input R.
3. Place three Sum blocks: Sum1 with signs [+ −] for node N1, Sum2 with signs [+ −] for node N2, and two more Sum blocks (Sum_G23 with [+ +] and Sum3 with [+ +]) to combine the parallel G2/G3 paths and the G4 path into node N3.
4. Place six Gain blocks (Simulink → Math Operations) and set their gain values to G1, G2, G3, G4, H1 and H2 respectively.
5. Wire the model as follows: R → Sum1(+); Sum1 output → G1 → Sum2(+); Sum2 output (N2) branches into G2, G3 and H2; G2 and G3 outputs → Sum_G23 → Sum3(+); R → G4 → Sum3(+) (second input); H2 output → Sum1(−) (second input, closing the −H2 loop); Sum3 output (N3) → H1 → Sum2(−) (second input, closing the −H1 loop); Sum3 output → Scope and → To Workspace block, which is taken as output C.
6. Assign numerical values to G1, G2, G3, G4, H1, H2 (either directly in each Gain block, or from MATLAB workspace variables), run the simulation, and read the steady output value of C for a unit step input R = 1. Since the system has no dynamics, C reaches its final value immediately at t = 0+ and remains constant.
7. Compare the simulated value of C (with R = 1) against the value predicted by the Mason's Gain Formula expression derived in Part A.

6. Results

Sample gain values were used to numerically verify the derived transfer function:

G1	G2	G3	G4	H1	H2
2.3	1.7	0.9	1.4	0.6	0.35

Method	Computed value of C/R
Analytical: Mason's Gain Formula	2.528083
Simulation: Simulink model	2.528083

The transfer function value obtained from the Simulink model exactly matches the value computed analytically using Mason's Gain Formula.

7. Conclusion

The overall transfer function of the given signal flow graph was successfully obtained using Mason's Gain Formula as:

$$\frac{C}{R} = \frac{G_1G_2 + G_1G_3 + G_4(1 + G_1H_2)}{1 + H_1(G_2 + G_3) + G_1H_2}$$

This result was verified by constructing an equivalent Simulink model and comparing the simulated input–output gain against the analytically computed value; both were found to be identical, confirming the correctness of the analysis and the validity of Mason's Gain Formula as a systematic method for reducing signal flow graphs.