CS241 Automatic Control

SFG simulator

Prepared by

Mazen Elmesery (50)

Abd Elrahman Sami (38)

Problem statement

It is required to make a Signal flow graph representation of the system. Assume that total number of nodes and numeric branches gains are given it is required to implement :

1- Graphical interface.
2- Draw the signal flow graph showing nodes, branches, gains, ...
3- Listing all forward paths, individual loops, all combination of n non-touching loops.
4- The values of Δ , $\Delta 1$, ..., Δm where m is number of forward paths.
5- Overall system transfer function.

overview

This is a simulator that applies **Mason's** with signal flow graph and calculate the overall transfer function using the formula

$$T = \frac{C(s)}{R(s)} = \frac{\sum_{i=1}^{N} P_i \Delta_i}{\Delta}$$

Where,

- $C(s)$ is the output node
- $R(s)$ is the input node
- T is the transfer function or gain between $R(s)$ and $C(s)$
- P_i is the i^{th} forward path gain

$$\Delta = 1 - (sum\ of\ all\ individual\ loop\ gains) + (sum\ of\ gain\ products\ of\ all\ possible\ two\ nontouching\ loops) - (sum\ of\ gain\ products\ of\ all\ possible\ three\ nontouching\ loops) + \cdots$$

Δ_i *is obtained from* Δ *by removing the loops which are touching the* i^{th} *forward path.*

Main Modules

- **Linkclass**

  This is the base element of the program and it represents the link between any 2 nodes in the graph and it holds 2 fields :

  - **Destination :** and this is the node that the link arrow is pointing at
  - **gain:** and this is the link gain

- **SFG class**

  This is the main part of the program that holds most of the logic and the structure of the graph and it contains :

  - **Nodes list**
  - **Forward paths list**
  - **Forward paths gains list**
  - **Forward paths deltas list**
  - **Loops List**
  - **Loops gain list**
  - **Non touched loops list**

  also it contains a lot of methods that find the forward paths and the loops and applies operations on them to calculate the deltas and the gains in order to calculate the overall transfer function.

Algorithms used

Famous algorithms used

- **DFS** to find the forward paths and the loops.

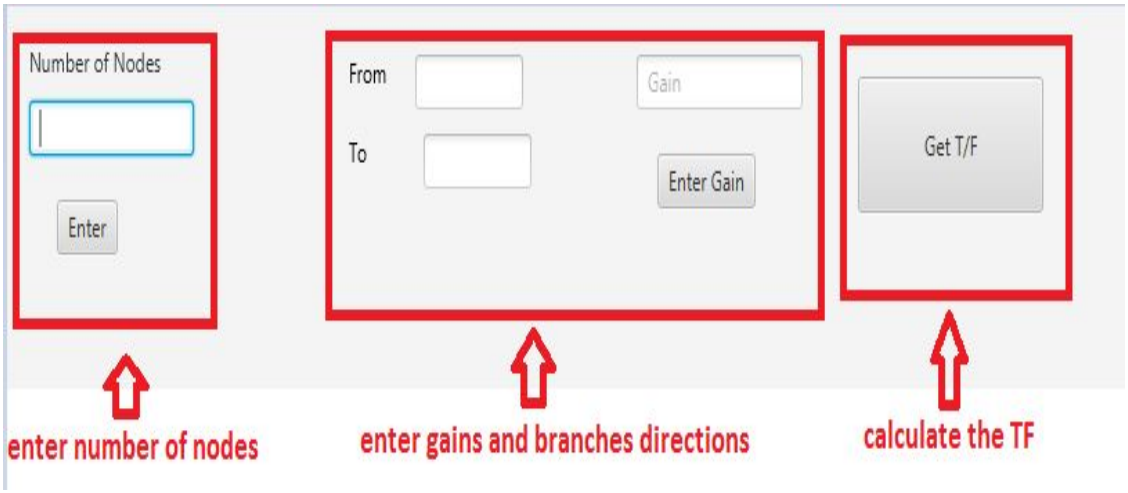
Implemented algorithms

- **addLink(source, destination, gain)**
- **findForwardPaths()**
- **findLoops()**
- **getAllNonTouchedLoops()**
- **getNonTouchedLoopsforFps()**
- **calculateDeltasFPS()**
- **calculateOverallTF()**

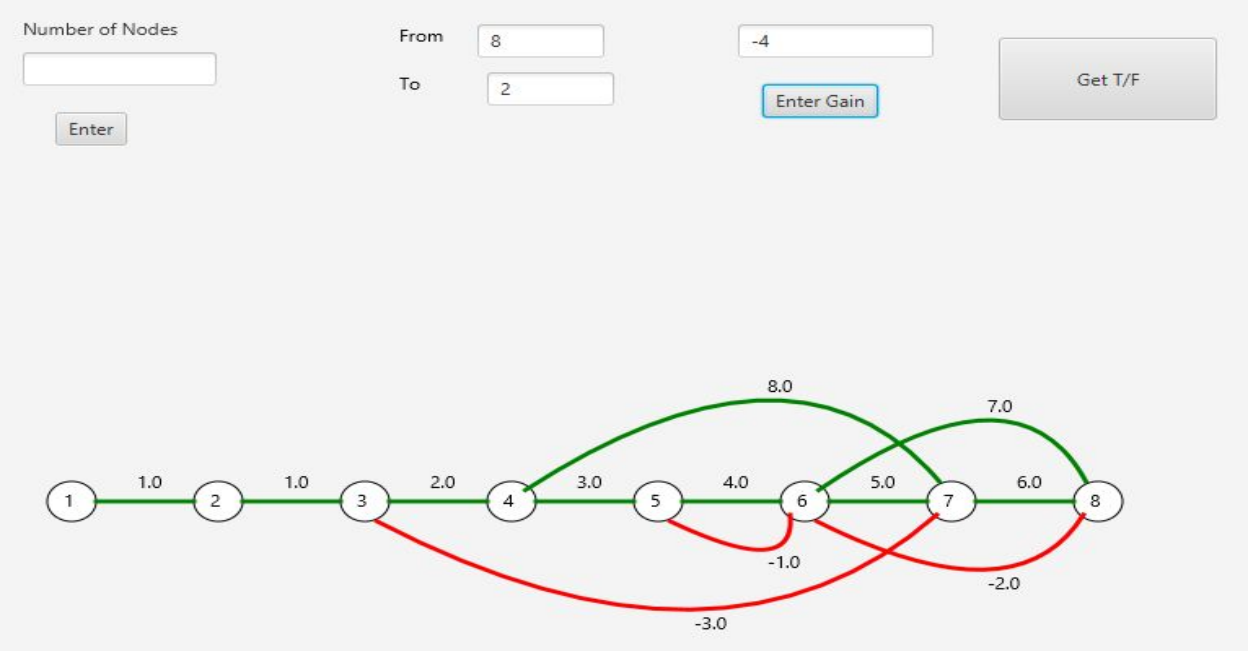
User guide and sample runs

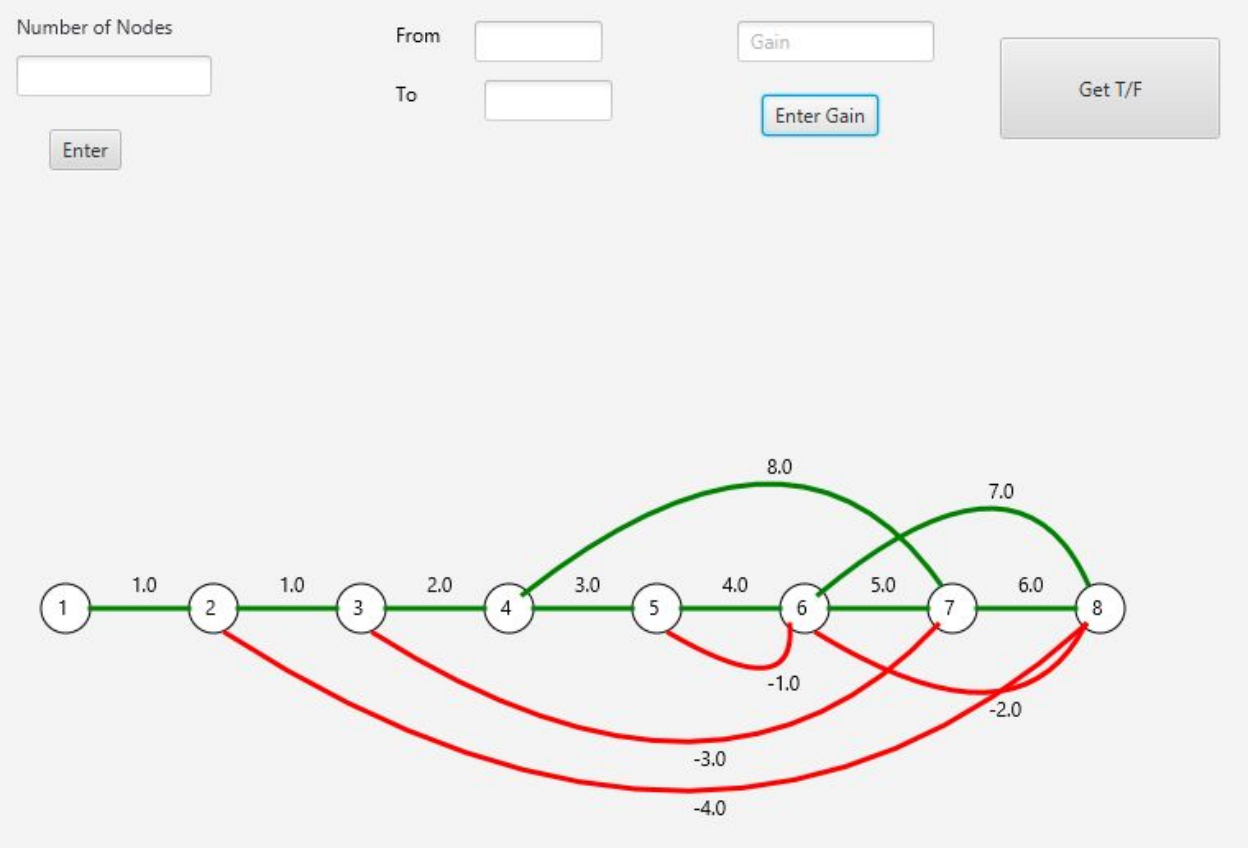
- **User interface**

  Our program supports a graphical user interface that user can interact with.



after user enters the required inputs the graph is drew dynamically with each input like the following

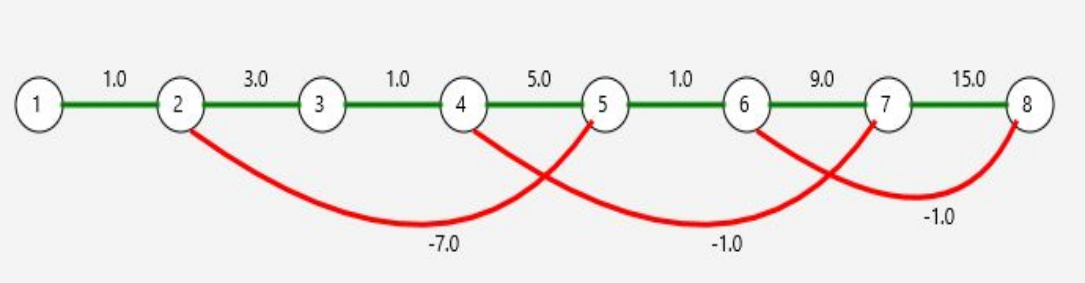


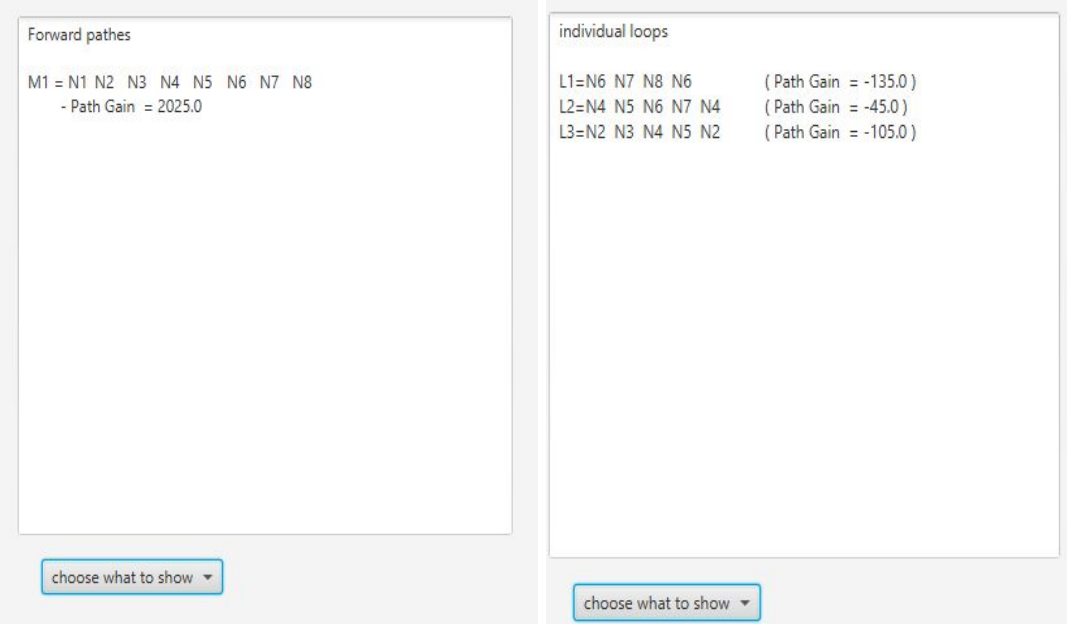


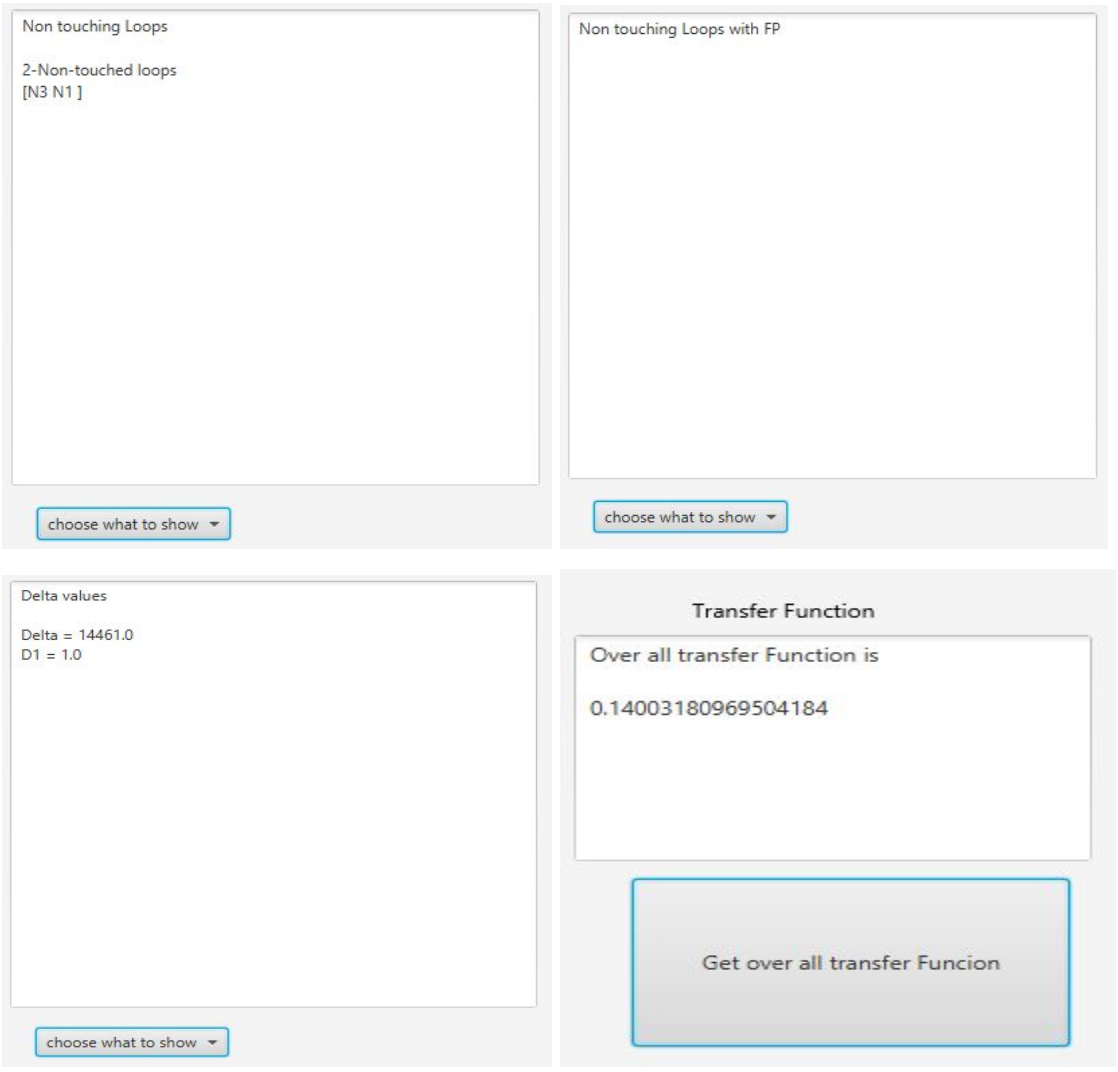
- **Samples runs**

Example 1 :

#nodes : 8	from :	to:	gain:
	1	2	1
	2	3	3
	3	4	1
	4	5	5
	5	6	1
	6	7	9
	7	8	15
	8	6	-1
	7	4	-1
	5	2	-7

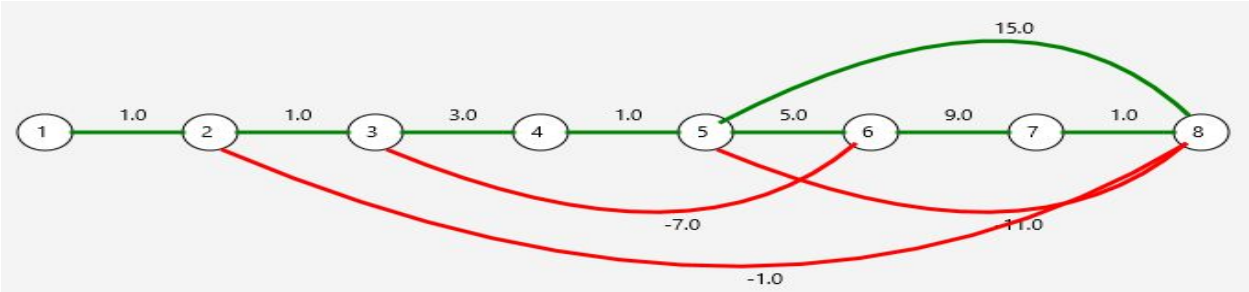






Example 2 :

#nodes : 8	**from :**	**to:**	**gain:**
	1	**2**	**1**
	2	**3**	**1**
	3	**4**	**3**
	4	**5**	**1**
	5	**6**	**5**
	6	**7**	**9**
	7	**8**	**1**
	8	**5**	**-11**
	8	**2**	**-1**
	6	**3**	**-7**
	5	**8**	**15**



Forward pathes

M1 = N1 N2 N3 N4 N5 N6 N7 N8
- Path Gain = 135.0

M2 = N1 N2 N3 N4 N5 N8
- Path Gain = 45.0

choose what to show

individual loops

L1=N5 N6 N7 N8 N5 (Path Gain = -495.0)
L2=N5 N8 N5 (Path Gain = -165.0)
L3=N2 N3 N4 N5 N6 N7 N8 N2 (Path Gain = -135.0)
L4=N2 N3 N4 N5 N8 N2 (Path Gain = -45.0)
L5=N3 N4 N5 N6 N3 (Path Gain = -105.0)

choose what to show

Non touching Loops

choose what to show

Non touching Loops with FP

choose what to show

Delta values

Delta = 946.0
D1 = 1.0
D2 = 1.0

choose what to show

Transfer Function

Over all transfer Function is

0.19027484143763213

Get over all transfer Funcion