JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Application of Signal Flow Graph and Mason's Gain Formula in Analysis of Unilateral Electrical Networks

B.C. Phukan

Faculty, Department of Mathematics, Jorhat Engineering College, Jorhat 785007, Assam, India. Email: bcphukan.jec@gmail.com

ABSTRACT

We can calculate the transfer function of an electrical network in several ways, such as using Kirchhoff's laws, two-point network theory, nodal analysis, or the time constant method. Another way to find the transfer function algebraically is by using a Signal Flow Graph (SFG). An SFG is a directed, weighted graph with nodes and branches. Here, we will make a signal flow graph for a unilateral electrical network and analyze it using Mason's gain formula and the reduction rules for Signal Flow Graphs.

(Keywords: Transfer function, Mason's formula, Signal Flow Graph, Unilateral Electrical Network.)

1. Introduction:

Graph theory is an important part of mathematics that studies networks made of points connected by lines [12]. It started from number games and fun math puzzles, and has grown into a large field used in computer science, social science, operational research, chemistry, and network analysis

A graph is described as a pair G = (V, E), where V is a [12] finite, non-empty set of points and E is the set of connections. On paper, these points appear as dots, and the connections show up as lines linking them. By studying the basic properties of graphs, we gain strong tools to understand complex networks and solve many problems more effectively.

An SFG, or signal flow graph [4,13], is a directed, weighted graph where signals are assigned to nodes and functions to edges to show how signals move within a system. We can find the transfer function of electrical circuits using several methods, including Kirchhoff's rules [10], two-port network theory [17], nodal analysis [14], and the time constant method [15]. These methods usually require extensive calculations and take a lot of time. Also, the signal flow graph offers a single visual model that connects state variables (parameters) to the transfer function, making it easier to understand complex network behaviour. We can build a signal flow graph using mesh rules, node rules, and Ohm's law. This method gives more accurate results. Furthermore, the signal flow graph acts as a single visual model that links state variables (parameters) to the transfer function, helping to explain the complex behaviour of a network. Using mesh rules, node rules, and Ohm's law makes it possible to build a more accurate signal flow graph.

Signal Flow Graphs (SFG) have evolved through the contributions of numerous scholars. Samuel Jefferson Mason first used the term to describe a specialized flow graph, while Claude Shannon later developed what is now commonly called the Signal Flow Graph, or Mason graph [11,12]. In 1942, Shannon applied the concept of Signal Flow Graphs to address problems in analog computers [16,18]. Mason, in his works from 1953 and 1956, is widely recognized for advancing signal flow graphs by demonstrating their application to complex electronic problems [1,3,5]. The term "signal flow graph" was selected because it was initially used in the context of electronic problems involving electronic signals and system flow charts [6,8,9]

The Signal Flow Graph (SFG) method, developed by J.A. Brzozowski and E.J. McCluskey Jr. in 1963, was designed to characterize sequential circuits using regular expressions [2]. State diagrams of sequential circuits can be accurately interpreted using methodologies derived from signal flow graph theory. In 2017, Feim Ridvan Rasim and Sebastian M. Sattler assessed the reliability of the SFG method [19, 20,21]. We can validate a signal flow graph by generating the graph's identity.

2. Preliminary

Electrical Network: A closed electrical network comprises various electrical components, such as transformers, resistors, inductors, capacitors, and both voltage and current sources.

Circuit: A **circuit** is a complete loop that lets electric current or other signals move from a source through parts and back to the source. A circuit is also called a closed electric network.

Graph: A graph, in its most precise form, is an ordered pair G = (V, E): here, V represents the collection of vertices or nodes, while E captures the set of edges or branches that connect them.

Node: An electrical node is any point, junction, or continuous conducting wire in a circuit where two or more parts, like resistors, capacitors, or power sources, connect.

Branch: A branch is a single path in an electrical circuit that links two nodes, or junctions, and contains one or more circuit elements, such as a resistor, source, capacitor, or inductor.

Walk: A walk is a finite sequence where vertices and edges alternate, starting and ending with vertices. Each edge connects the vertex before it to the one after it.

Path: A path refers to a sequence of connected points in which each point occurs exactly once.

Loop: A loop is a closed path that begins at a vertex and returns to the same vertex or node, without passing through any node more than once.

Self-loop: A **self-loop, also called a loop,** is an edge in a graph that connects a vertex to itself.

Loop gain: Loop gain is defined as the gain obtained by traversing a path that begins and ends at the same node.

Non-touching loops: Non-touching loops are loops that do not share any nodes.

Forward path: A path that begins at an initial node, passes through other nodes, and ends at an output node is called a forward path.

Feedback path: A "feedback path" refers to a path that originates and terminates at the same node.

Forward path gain: The forward path gain is the sum of gains along a path from an input node to an output node in a signal flow graph.

Digraph: A directed graph, also known as a digraph, is a graph in which each edge has a specific direction.

Transfer function: The system's transfer function, T(s), is defined as C(s)/R(s). Here, R(s) is the system's input shown in a signal flow graph, and C(s) is the output shown in the signal flow graph.

3. METHODOLOGY

3.1. Signal Flow Graph: A Signal Flow Graph (SFG) is a directed, weighted graph that represents signal flow within a system. An SFG graphically represents the relationships among system variables.

3.1.1. Reduction Rules for Signal Flow Graphs (SFG), also referred to as Modifications of Signal Flow Graphs:

Rule 1:

(a)

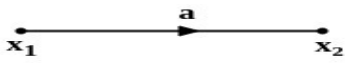


Fig.1

Let "input" be represented as "x_1" and "output" as "x_2" where "a" denotes the gain. Thus, $x_2 = ax_1$

(b)

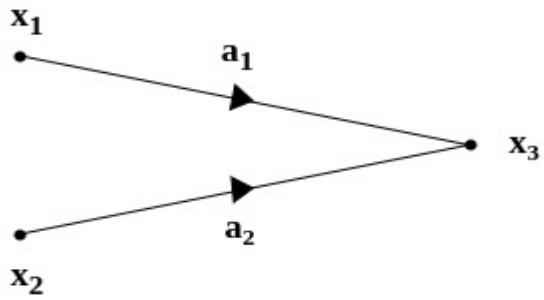


Fig.2

Let x_1 and x_2 denote input signals with respective gains a_1 and a_2 . The output signal is then expressed as $x_3 = a_1x_1 + a_2x_2$

Rule2:



Fig.3

Suppose x_1, x_2, x_3 represents nodes, with x_1 as the input and x_3 as the output, and the gains are a and b. Once the adjustments are made, we get $x_2 = ax_1,\ x_3 = bx_2 = abx_1$

Rule3:

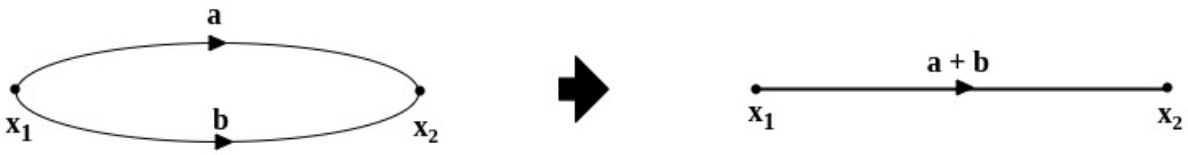


Fig.4

Let the input be denoted by x_1 and the output byx_2 with gains a and b assigned to the respective branches. The resulting output is then $x_2 = (a + b)x_1 = ax_1 + bx_1$

.

Rule4:

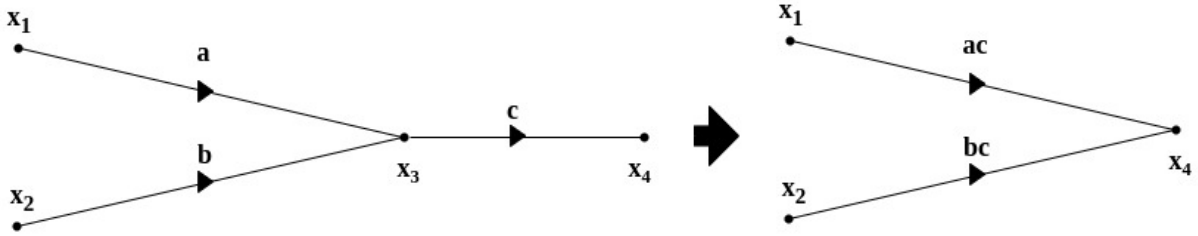


Fig.5

Assuming $x_1, x_2, x_3 \; and \; x_4$ denote nodes and a, b, and c represent the branch gains, the following relationships can be established. Consequently, it can be concluded that $x_3 = ax_1 + bx_2$ and $x_4 = cx_3 = c(ax_1 + bx_2)$. . Therefore, $x_4 = acx_1 + bcx_2$

Rule5:

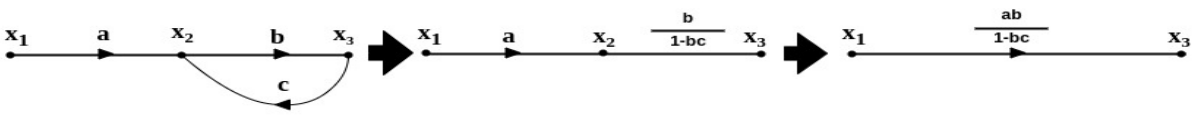
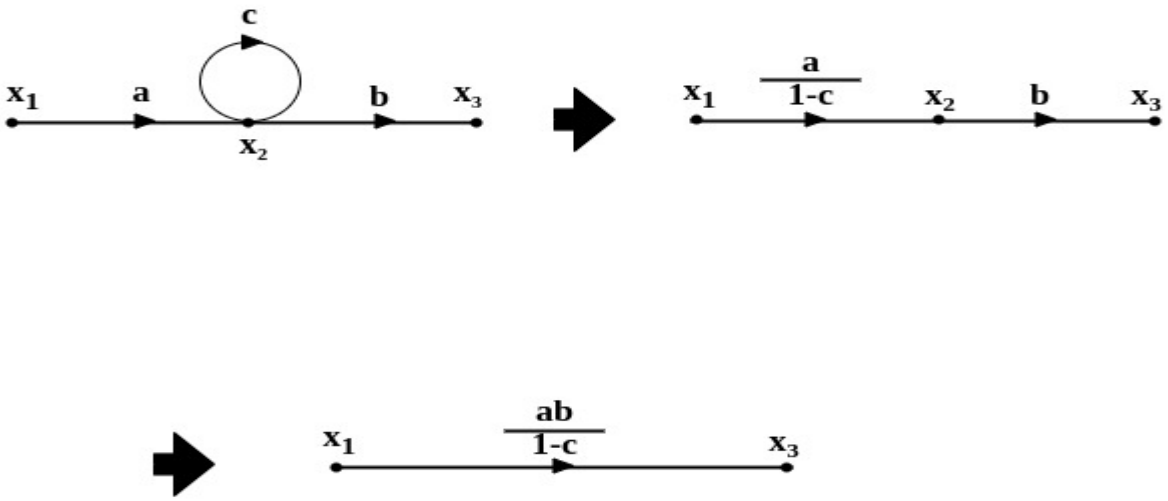


Fig.6

Let,x_1, x_2 and x_3 denote nodes with respective branch gains a, b, and c, as illustrated in the figure. Then, the following relationship holds:

$x_2 = ax_1$ and $x_3 = \left(\frac{b}{1-bc}\right)x_2 = \left(\frac{b}{1-bc}\right)ax_1 = \frac{ab}{1-bc}x_1$



Rule6:

Fig.7

If x_1, x_2 and x_3 denote nodes, where a, b (reflexive or self-referential), and c represent the gains of the respective branches as illustrated in the figure, then $x_2 = \frac{a}{1-c}x_1$ and $x_3 = bx_2 = b\frac{a}{1-c}x_1 = \frac{ab}{1-c}x_1$

Rule 7: Loop reduction:

(a) **Signal loop:** The transmission (T_{ij}) from an independent variable x_i (source node) to a dependent variable x_j (internal or sink node) in a signal flow graph that contains only one loop and one path is given by

$$T_{ij} = \frac{P_{ij}}{1-L} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (1)$$

where P_{ij} represents the forward-path transmission from x_i and x_j, and L denotes the loop transmission.

(b) Multi-loop with non-touching loops: In a system comprising a cascade of non-touching loops that do not share any common nodes, the overall transmission equals the product of the individual transmissions, as indicated by equation (1) for a single loop.

$$T = \frac{P_{ij}}{1-L_j}.\frac{P_{jk}}{1-L_k} \ldots\ldots\ldots\ldots\ldots\ldots.\frac{P_{(n-1)n}}{1-L_n} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2)$$

(c) Multiloop with touching loops:

In cases where the two loops share at least one node, as illustrated in the figure below, the overall transmission is expressed as

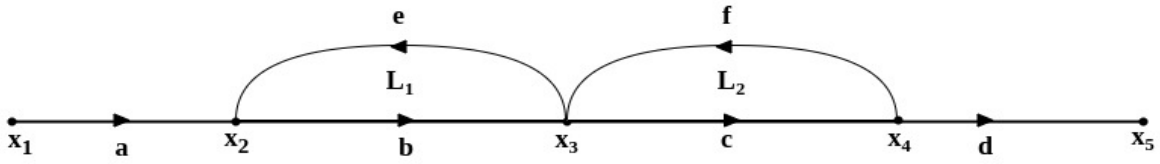


Fig.8

In this context x_1, x_2, x_3, x_4 *and* x_5 denotes nodes, while a, b, c, d, e, and f represent the gains of the branches as illustrated in the figure. The transfer function between ix_1 *and* x_5 is given as follows:

$$T_{15} = \frac{x_5}{x_1} = \frac{abcd}{1-be-cf} = \frac{P_{15}}{1-L_1-L_2} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

$$x_5 = \frac{abcd}{1-be-cf} x_1$$

3.2. Mason's Gain Formula (MGF): The Mason's Gain Formula (MGF) is used to find the transfer function of a linear signal flow graph. Here, R(S) represents the system input, and C(S) represents the output, as shown in the signal flow graph.

The system's transfer function is given by $T(S) = \frac{C(S)}{R(S)}$

Overall gain $T = T(S) = \sum_{k=1}^{N} \frac{P_k \Delta_k}{\Delta}$

where,

N Number of paths from input to output

P_k Forward path gain of the kth forward path

Δ = 1 - (Sum of individual loop gains) + (sum of two non touching loops gain)

- (sum of three non-touching loops gain) + (sum of four non-touching loops gain)

- ……..

Δ_k = 1 - (loops gain which are not touching the kth forward path)

4. An electrical network illustration

Graphs are a helpful way to represent electrical networks. In this approach, each node in the network is a vertex, and each branch is an edge. For simple circuits, finding loop currents and node voltages is usually easy. But as circuits become more complex, other methods are needed. Using graphs can make it easier to solve problems in complicated electrical networks [7]. Network graphs also help us understand how circuits work. Here is an example of a signal flow graph for an electrical network:

4.1 Unilateral circuit

The following unilateral electrical circuit will be analysed by calculating its transfer function using the Signal Flow Graph (SFG) method.

4.1.1 Consider a Unilateral Electrical Network

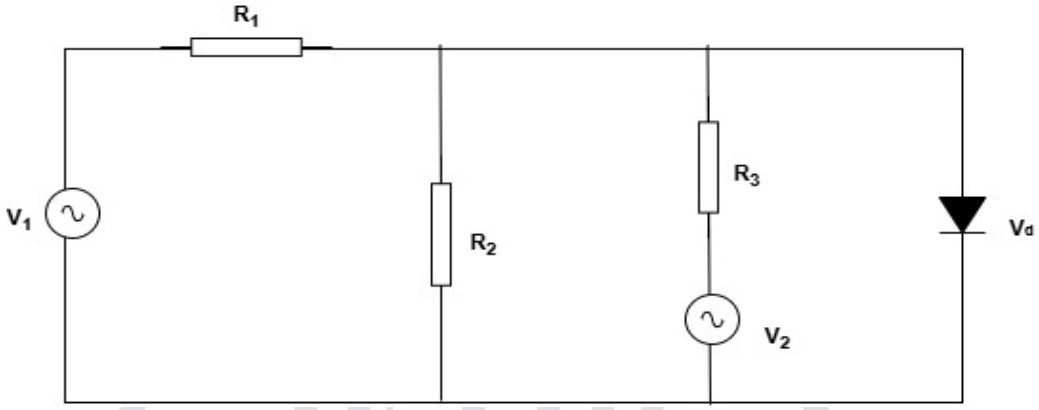


Fig.9 Electrical network

4.1.2 A graph model representation

A graph model represents a network as a set of vertices and edges that cover the entire network. Given below is the graph model of the circuit shown above

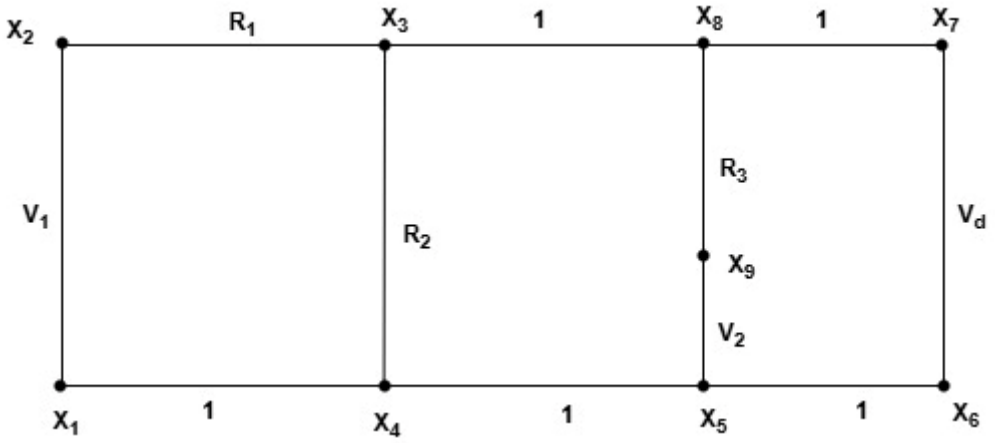


Fig.10 Graphical representation of the electrical network

4.1.3 Digraph Representation

A digraph model represents a graph using vertices and directed, weighted edges that cover the entire network. The digraph model of the circuit shown above is given below.

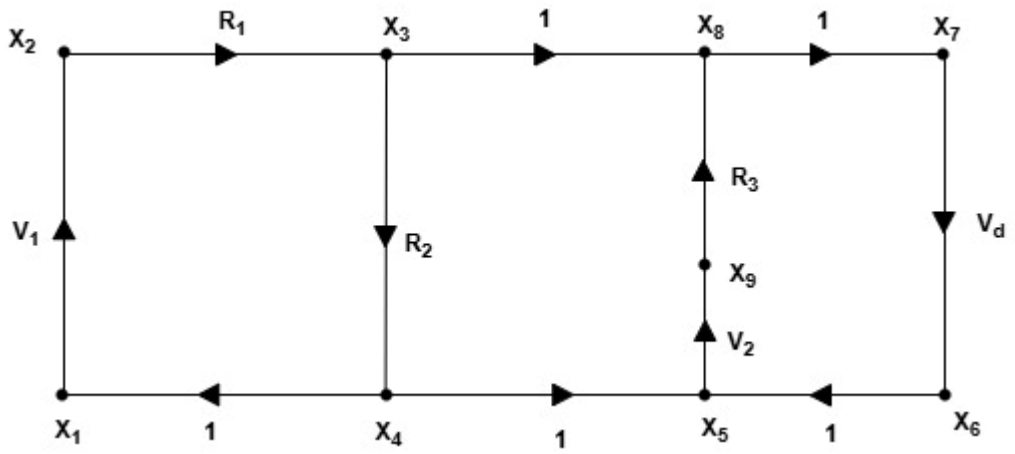


Fig.11 Digraph representation of the electrical network

4.1.4 Signal Flow Graph (SFG) representation

A signal flow graph (SFG) model represents a digraph as a directed, weighted graph encompassing the entire network. The SFG model for the network illustrated above is shown below.

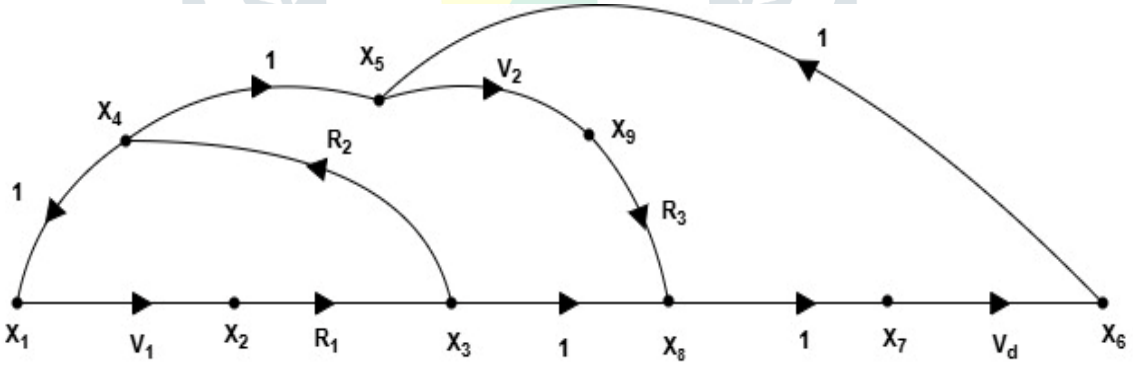


Fig.12 Signal Flow Graph representation

4.1.5 Calculation of Transfer Function by Reduction of the Signal Flow Graph

Node $x_7, x_{2,}$ and x_9 are taken out using Rule-

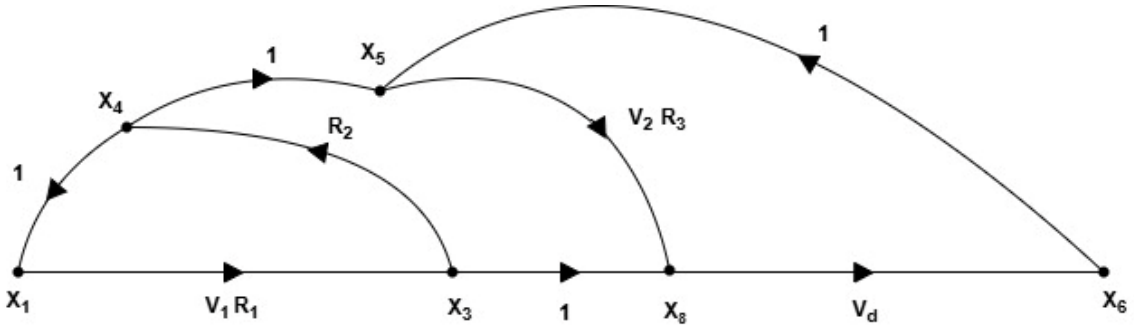


Fig.13 Reduced Signal Flow Graph

By using Rule 4, we have the following modification

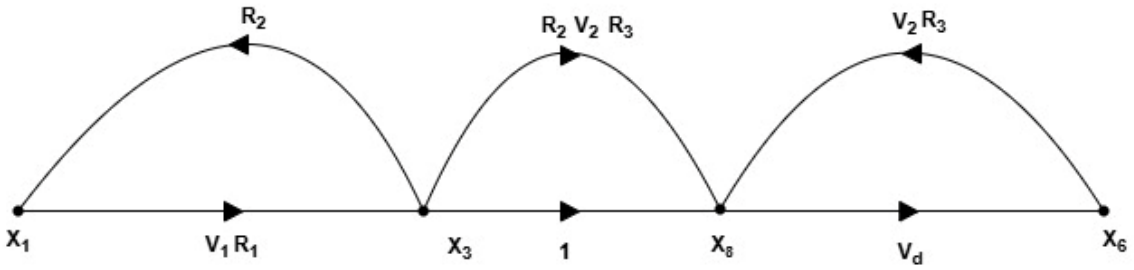


Fig.14 Reduced Signal Flow Graph

Using Rule 3, we have the following modification

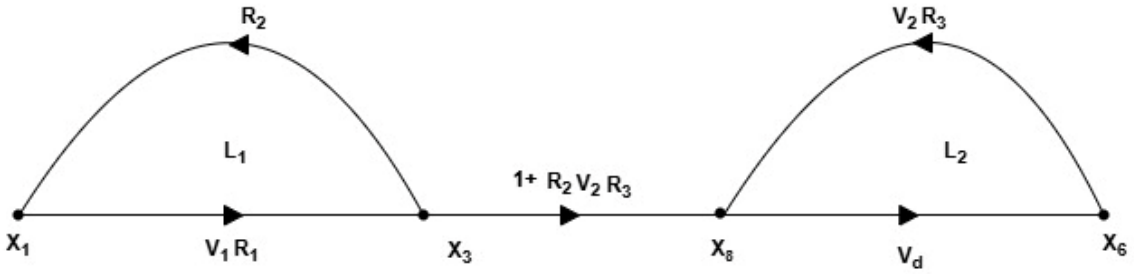


Fig.15 Reduced Signal Flow Graph

4.1.6 Calculation of Transfer Function by Reduction of the Signal Flow Graph

Now, the loop paths are as follows:

$$L_1 = x_1 V_1 R_1 x_3 R_2 x_1$$

$$L_2 = x_3 V_d x_6 V_2 R_3 x_3$$

Now, the loop gains are as follows:

$$L_1 = V_1 R_1 R_2$$

$$L_2 = V_d V_2 R_3$$

Both two loops are non-touching loops without any common nodes in the above Fig.5.7

Transmittance between the nodes x_1 and x_6. By using Rule 7(b) of loop reduction, i.e., multiloops with touching loops, we have $T_{1,6}$

Where, $P_{1,6}$ = Forward path gains

Here, there is only one forward path, as follows:

$$F_1 = V_1 R_1 R_2 V_d (1 + V_2 R_2 R_3)$$

Therefore, transmittance between the nodes x_1 and x_6 is

$$T_{1,6} = \frac{V_1 R_1 R_2 V_d (1+V_2 R_2 R_3)}{1-V_1 R_1 R_2 - V_2 R_3 V_d + V_1 R_1 R_{2_2} R_3 V_d} \quad \ldots\ldots\ldots\

P_k Forward path gain of the kth forward path

Δ = 1 - (Sum of individual loop gains) + (sum of two non touching loops gain)

- (sum of three non-touching loops gain) + (sum of four non-touching loops gain)

- ……..

Δ_k = 1 - (loops gain which are not touching the kth forward path)

From Fig.15**,** we first need to identify the forward paths and path gains. There are four forward paths in this particular problem.

$$F_1 = V_1R_1R_2V_d(1 + V_2R_2R_3)$$

Secondly, we have to find all individual loops and their gains, and here the individual loops' gains are

$$L_1 = V_1R_1R_2$$

$$L_2 = V_2R_3V_d$$

Thirdly, we have to identify the non-touching loops' gains. There are two non-touching loops in this problem.

Now, $\Delta = 1 - (V_1R_1R_2 + V_2R_3V_d) + (V_1R_1R_2V_2R_3V_d)$

$\Delta_1 = 1 -$(loop gain, which does not touch the forward path F_1)

=1-0

=1

The transfer function between the nodes x_1 and x_6 is

$$T(S) = \frac{C(S)}{R(S)} = T_{1,6}$$

$$T(S) = T_{1,6} = \frac{F_1\Delta_1}{\Delta}$$

$T_{1,6} = \frac{V_1R_1R_2V_d(1+V_2R_2R_3)}{1-V_1R_1R_2-V_2R_3V_d+V_1R_1R_{2_2}R_3V_d}$ …… (4.2)

We observe that equations (4.1) and (4.2) are equal.

For a given electrical circuit, the transfer function (i.e., the transmittance between any two nodes) can be calculated using the reduction rules of signal flow graphs and Mason's gain formula. In this paper, for the same electrical circuit shown in Fig.15, the transfer function between two nodes x_1 and $x_6$is determined using both the reduction of the signal flow graph equation (4.1) and Mason's gain formula equation (4.2), where both equations yield equivalent results.

In an electrical system, the Transfer Function (TF) is the ratio of the output to the input. The Transfer Function (TF) provides the circuit's gain, frequency response, and stability. It defines the range of input values for which the output remains stable, its behavior for different input types (AC or DC), and its key features. We have found the Transfer Function (TF) of any system (circuit) and can use it to determine many of its parameters. One can use this Transfer Function (TF) to analyze its behavior for any input. Another advantage is that it represents the whole system as **a single edge, i.e., a one-edge (I-edge)** between any two nodes. It makes studying a complex system easier.

5. Conclusion:

The Signal Flow Graph (SFG) is one of the best tools to analyze an electrical network. These methods are an alternative approach to traditional methods such as Kirchhoff's rules and nodal analysis. Sometimes calculating the transfer function using signal-flow graph reduction can be somewhat difficult, but Mason's Gain Formula is useful for analyzing complex network circuits. In particular, Mason's Gain Formula provides a systematic, algebraic method for calculating the transfer function, which can be especially useful when traditional techniques are impractical. Signal flow graphs and their reductions provide a graphical representation of the analysis, enhancing visualization and deepening understanding of the network's behavior.

In this paper, we aimed to advance electrical network analysis by showing how Mason's Gain formula and the Signal Flow Graph (SFG) can help find a network's transfer function. The signal flow graph is an important tool for this analysis. We started with a graphical representation of the network and built a Signal Flow Graph to find the transfer function between points. This process involves identifying forward-path gains and the different loops. A one-edge, or I-edge, can represent the transfer function between two points in the network. Mason's Gain Formula gives a clear, algebraic way to calculate the transfer function. Using the Reduction Rules of the Signal Flow Graph makes network analysis faster and more efficient. This method has strong potential and many academic uses, but more research is still needed.

6. References:

[1] Mason, Samul J. (1953). Feedback Theory-Some Properties of Signal Flow Graphs. Research Laboratory of Electronics, Massachusetts Institute of Technology.

[2] Brzozowski J.A. and McCluskey E. J. JR (1963). Signal Flow Graph Techniques for Sequential Circuit State Diagram. IEEE Transactions on Electronic Computers.

[3] Dostal Tomas and Mikula Jan (1992). Signal-Flow Graph Network Analysis. Radio Engineering, Vol.1, No.1.

[4] Karmakar, A., & Sinha, A. (2011). A novel architecture of a reconfigurable radio processor for implementing different modulation schemes. https://doi.org/10.1109/iccrd.2011.5763985

[5] Ridvan Feim and Sattler Sebastian M. (2017) Analysis of Electronic Circuits with the Signal Flow Graph Method. Circuit and System, 2017, 8, 261-274. http:// www. scirp.org/journal/cs

[6] Málek, M. (2017). Stochastic Optimization of Network Flows. https://core.ac.uk/download/83618168.pdf

[7] Velammal S and Ram Arun Muthu (2018). Application of Graph Theory in Network Analysis. LJSTE-International Journal of Science Technology & Engineering, Volume 5, Issue 1.

[8] Majeed, A. and Rauf, I., 2020. Graph theory: A comprehensive survey about graph theory applications in computer science and social networks. *Inventions*, *5*(1), p.10.

[9] Werner, F., 2020. Graph-theoretic problems and their new applications. Mathematics, 8(3), p.445.

[10] Mostafaie, T., Khiyabani, F.M. and Navimipour, N.J., 2020. A systematic study on meta-heuristic approaches for solving the graph coloring problem. *Computers & Operations Research*, *120*, p.104850.

[11] Wu, K.J., Hong, Y.W.P. and Sheu, J.P., 2020. Coloring-based channel allocation for multiple coexisting wireless body area networks: A game-theoretic approach. *IEEE Transactions on Mobile Computing*, *21*(1), pp.63-75.

[12] Jeng, S.L., Roy, R. and Chieng, W.H., 2020. A Matrix Approach for Analyzing Signal Flow Graph. *Information*, *11*(12), p.562.

[13] Beisegel, J., Chudnovsky, M., Gurvich, V., Milanič, M. and Servatius, M., 2022. Avoidable vertices and edges in graphs: Existence, characterization, and applications. *Discrete Applied Mathematics*, *309*, pp.285-300.

[14] Bravyi, S., Kliesch, A., Koenig, R. and Tang, E., 2022. Hybrid quantum-classical algorithms for approximate graph coloring. *Quantum*, *6*, p.678.

[15] Gheorghe, A.G. and Marin, M.E., 2022. Electrical Circuits as Dynamical Systems. In *Qualitative and Computational Aspects of Dynamical Systems*. Intech Open.

[16] Etzion, T., 2023. Puzzles and Tilings—A Fascinating Kingdom of Solomon Golomb. In *THE WISDOM OF SOLOMON: The Genius and Legacy of Solomon Golomb* (pp. 145-160).

[17] Etzion, T., 2023. Puzzles and Tilings—A Fascinating Kingdom of Solomon Golomb. In *THE WISDOM OF SOLOMON: The Genius and Legacy of Solomon Golomb* (pp. 145-160).

[18] Mezghani, A., Akrout, M., Castellanos, M.R., Saab, S., Hochwald, B., Heath, R.W. and Nossek, J.A., 2023. Reincorporating Circuit Theory into Information Theory. *IEEE BITS: The Information Theory Magazine*.

[19] FDA Approves Zepbound for Obstructive Sleep Apnoea: A Breakthrough in OSA Treatment https://vajiramandravi.com/current-affairs/fda-approves-zepbound-for-obstructive-sleep-apnoea/

[20] Biolek Dalibor and Biolkova Viera. Analysis of circuits containing active elements by using T- Dalibor.biolek@vabo.cz http://www.vabo.cz/stranky/biolek

[21] Biolek Dalibor and Biolkova Viera. Signal Flow Graphs Suitable for Teaching Circuit Analysis. This work is supported by the Grant Agency of Czech Republic by the research programme of BUT, Research of Electronic communication system and technologies.