COMPUTER JUSTIFICATION OF THE UPWARD TOPOLOGICAL ANALYSIS OF FLOW-GRAPHS

A. Kończykowska and J. Starzyk
Institute of Electronics Fundamentals
Technical University of Warsaw
Poland

Abstract

A computer program of symbolic analysis of decomposed signal flowgraph is presented. The program uses the modeification of the downward method of hierarchical decomposition. New method permits the condensed form of storage of results in the memory of the computer, which makes possible fully symbolic analysis of large networks and gives more convenient form for further computations.

1. INTRODUCTION

When the hierarchical decomposition method has been proposed for the analysis of large electrical networks although remarkable reduction of computation time has been achieved some inconveniences could be notified.

Firstly some operations were repeated during computations and better organization of analysis could result in reducing the time consumption from involution toward linear one.

The second consisted in the form of results obtained. Although all results were generated in fully symbolic way the fully symbolic storage of characteristic polynomials in the form of list of terms was impossible for big network. It resulted in the semisymbolic forms where only the most important elements were leaved as parameters.

The modified method of hierarchical analysis permits to resolve above problems. And so the first approach was called downward and the second upward analysis. The name is connected with the direction of exploration of the decomposition tree when represented in the most common graphical way /see fig. 1./.

2. HIERARCHICAL DECOMPOSITION

Let us consider a hierarchical decomposition as in [3, 4]. We shall concentrate on the node bisection of a signal flowgraph on each level of hierarchical structure.

We shall precise some terminology used for the description of the hierarchical structure.

We shall call a hierarchical decomposition a connected graph H with N nodes and N-1 edges such that: $\forall (x,y) \in H \quad \exists! z, \ (x,z) \in H$. In a graph H we have one initial node, denoted by x_0, such that $\not\forall\, x (x, x_0) \notin H$. We shall call a middle node such a node x that $\exists y\, (x,y) \in H$. All other nodes are terminal nodes. If $(x,y) \in H$ y is called a descendant of x and x a ascendant of y.

In our case every middle node has two descendants. If necessary we can call them arbitrary left and right descendant. Each node is situated on a certain level of the hierarchical structure. The level number of a node x equals the lenght of the path from x_0 to x. We say the hierarchical decomposition is the k-level one if the maximal level number of all nodes is equal k. We deal with the case of uniform hierarchical decomposition if all terminal nodes have the same level number.

In the aim to facilitate the further analysis of the hierarchical structure we shall number the nodes from 1 to N in the following way: if $(x,y) \in H$ then number of x < number of y. In this numeration the initial node has always the number 1. N.B. that in our case N is always odd. In the Fig. 1 we have an example

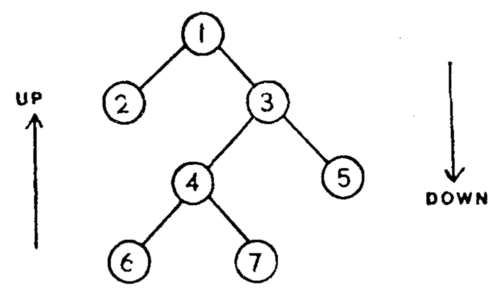

Fig. 1. Decomposition tree.

of the 3-level decomposition. The middle nodes are: 1, 3, 4 and the terminal: 2, 5, 6, 7. For example the descendants of 3 are 4 and 5 and the ascendant of 7 is 4.

3. UPWARD HIERARCHICAL ANALYSIS

To show what is the difference of the method of hierarchical analysis presented now in comparision with this presented in [3] we shall briefly outline the main concepts of the downward hierarchical analysis.

The nodes of the hierarchical decomposition have the following representation:

a/ the terminal nodel are the subgraphs of the initial graph /i.e. the model of an electrical network/ with distinguished block nodes the middle nodes represent the node association of two blocks from the next level.

In the downward method the analysis has been started on the 0-level and proceeded down to next levels with the analysis of the substitute graphs in the places of the middle nodes. On each level the type of necessary function for the next level has been determined. Arriving to the terminal node the analysis of the subgraph has been executed in due to get the necessary function of this block. Afterwards the way up has been executed as the multiplication of two functions on each middle node. When all the way back had been acomplished we have got the group of 'terms describing the function of our initial graph.

In the upward analysis we move only in one direction from terminal nodel to the top level 0. Such attitude gives us following advantages:

a/ we pass along the hierarchical structure only once and additionally we avoid the multiple analysis which occurs in the case of downward analysis

b/ this approach permits the application of the condensed form of results and finally better possibility of the storage of fully symbolic results of big networks. The proposition of such storage of results will be presented in the next parts of this paper.

If blocks of decomposition are numbered in the previously defined way, the process of analysis can be performed according to the flow-chart presented in the Fig. 2.

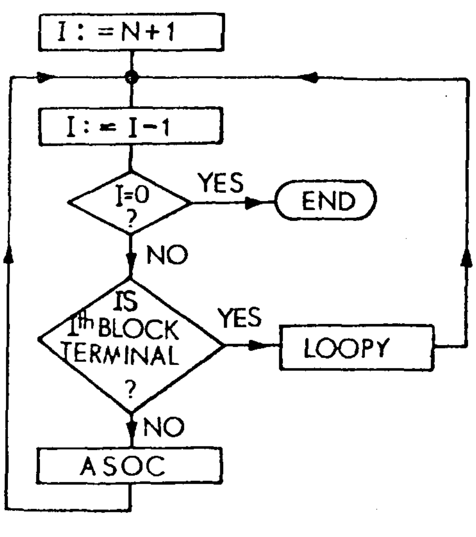

Fig. 2. Flow-chart of the program

The algorithm contains two main procedures:

a/ LOOPY - the analysis of the terminal block

b/ ASOC - the association of two previously analysed blocks.

4. ANALYSIS OF TERMINAL BLOCK

The analysis of the flow-graph is performed in the aim to obtain the function of the electrical networks. The formulas for these function have been presented for example in [3]. Generally speaking we are obliged to generate the sets of multiconnections [3] of different kinds with the respect of the input and output variable nodes. We must decide what kinds of sets we have to determine during the analysis of the subgraphs of the initial graph. They have to be general enough to reduce the number of different kinds of sets and on the other side the products of these sets should determine new sets of multiconnections without duplications.

Let us consider a subgraph with n nodes, nb block nodes, NB set of block nodes. The following kind of multiconnections is considered

Definition 1.

$C(B;E)$ where $B=\{b_1,\dots,b_m\}$, $E=\{e_1,\dots,e_m\}$

$B \cup E \subset NB$, is a set of multiconnections which have the following properties:

a/ the only incidence of block nodes is defined by sets B, E, nodes of B are origins and E are ends of multiconnections edges.

b/ all other nodes have full incidence.

Definition 2.

Every multiconnection has assigned a sign with respect to the node numeration

$$\text{sign } C = (-1)^{n+l+m}\,\text{ord}(b_1,b_2,\dots,b_m)\,\text{ord}(e_1,e_2,\dots,e_m) \qquad /1/$$

where $\text{ord}(x_1,x_2,\dots,x_m) =$

$$\begin{cases} 1 & \text{when the number of permutations ordering the set is even} \\ -1 & \text{in the opposite case} \end{cases}$$

l - number of loops

The number of different C for the subgraph with nb block nodes is equal to

$\sum_{i=0}^{nb} \binom{nb}{i}^2$ which is great reduction in comparision with the number of sets of multiconnections determined by paths and loops they have to form. This number is equal $\sum_{i=0}^{nb} \binom{nb}{i}^2 i!$

The procedure LOOPY generates all sets of defined multiconnections and stores them in a list with their identification code. Generation is based on the inspection of the node to node incidence matrix.

5. ASSOCIATION OF TWO BLOCKS

Let us consider an elementary node association of two blocks. This kind of operation we are obliged to perform every time when in the hierarchical analysis we deal with middle nodes.

Let us denote: NB_1, NB_2, NB set of block nodes for both blocks and the resulting block. Some subsets of these sets we denoted as follows:

COM = $NB_1 \cap NB_2$ - the set of common nodes

RED = COM - NB - the set of reduced nodes

Theorem

Any set of multiconnections C(B, E) can be obtained according to the following rule

$$C(B,E) = \bigcup C_1(B_1, E_1) \times C_2(B_2, E_2) \qquad /2/$$

where $B_1 \cap B_2 = \emptyset$, $E_1 \cap E_2 = \emptyset$,

$RED \subset (B_1 \cup B_2) \cap (E_1 \cup E_2)$

$B = B_1 \cup B_2$ - RED, $E = E_1 \cup E_2$ - RED

summation is performed on all such sets of multiconnections C_1, C_2.

For every multiconnection $c \in C$ the sign of c can be calculated as follows

$$\text{sign } c = \text{sign } c_1 \text{ sign } c_2 (-1)^k \qquad /3/$$

$$k = \min(\text{card } E_1 \cap B_2 \cap COM, \text{card } E_2 \cap B_1 \cap COM) + \text{card } COM \qquad /4/$$

where $c = c_1 \cup c_2$ $c_1 \in C_1$ and $c_2 \in C_2$

Example

In the Fig. 3 an example of block association is presented.

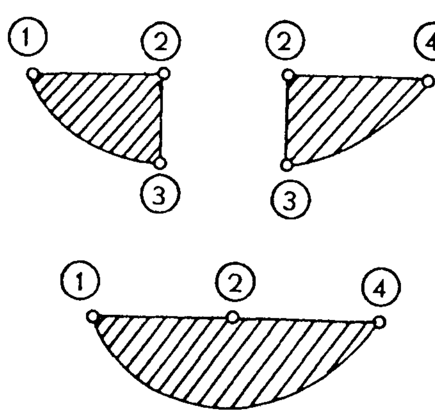

Fig. 3. Association of two blocks

For this case we have:

$NB_1 = \{1, 2, 3\}$, $NB_2 = \{2, 3, 4\}$, COM $= \{2, 3\}$

NB $= \{1, 2, 4\}$, RED $= \{3\}$.

6. FORM OF RESULTS

The condensed form of results was proposed in[1]. In the domain of signal-flow graph analysis it can be adopted as follows.

We store all results sequentially on one list. If these are results of final block different sets of multiconnections are memorized. Each of them has code, which is the bit representation of its type. For example two bits for every node - if the first bit equals 1 tne node $n \in B$, if the second bit equals 1 then the node $n \in E$. So every multiconnection can be coded on the bits of one word.

For the middle blocks results are slightly different. We remember /2/ that every set of multiconnections is the sum of combinations of sets from previous levels. If we want to conserve the condensed form on this level we do not develop the formulae but store only adresses of sets to be multiplied. Computations of numerical values for given values of parameters can be performed in the same manner.

Computer test of this method has been done for the fully symbolic ladder.

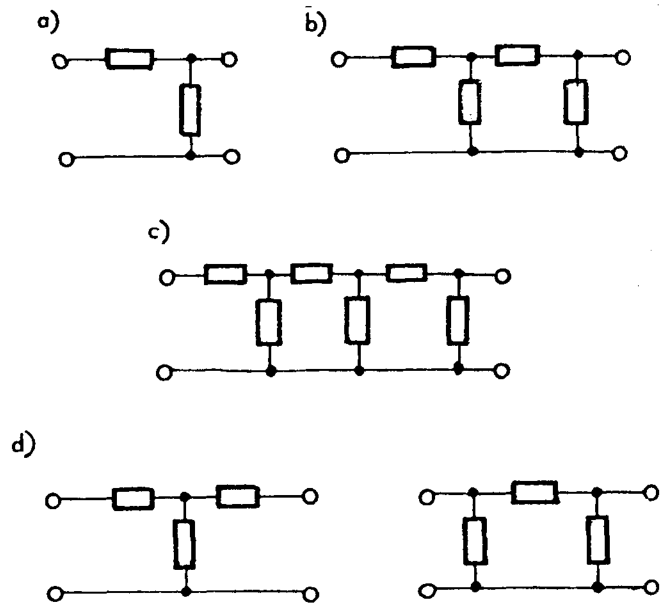

Fig. 4. Terminal blocks of ladder decomposition.

Fig. 5, 6 show that both time of analysis and memory needed for storage of results increase linearly with the number of nodes. Fig. 4 presents four types of terminal blocks analysed and results obtained show that both analysis time and memory depends on the kind of partition performed. We observe that the optimal terminal blocks are rather small, having only few internal nodes and as less as possible block nodes.

7. CONCLUSIONS

The presented method can be used for the fully symbolic analysis of any linear system of equations with practically non limited number of parameters. The method is very effective as time consumption and memory demands are concerned. If structurally identical subnetworks are isolated as subgraphs then analysis can be performed for one of them only and the result of analysis stored only once.

The impact of a single parameter on the final results can easily be observed and one can take advantage of it to solve different problems of circuit design, for example, sensitivity and tolerance analysis.

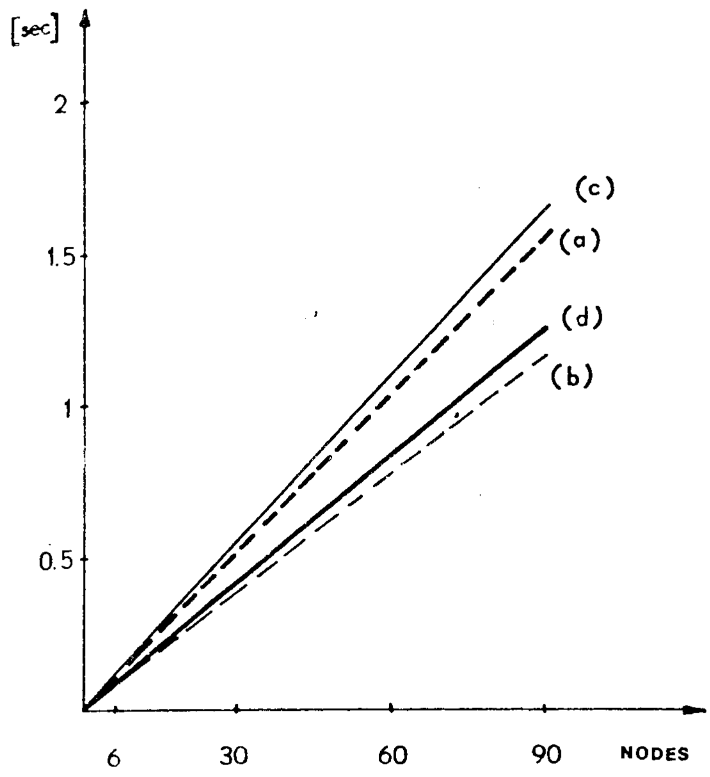

Fig. 5. Relationship between the analysis time and the number of nodes.

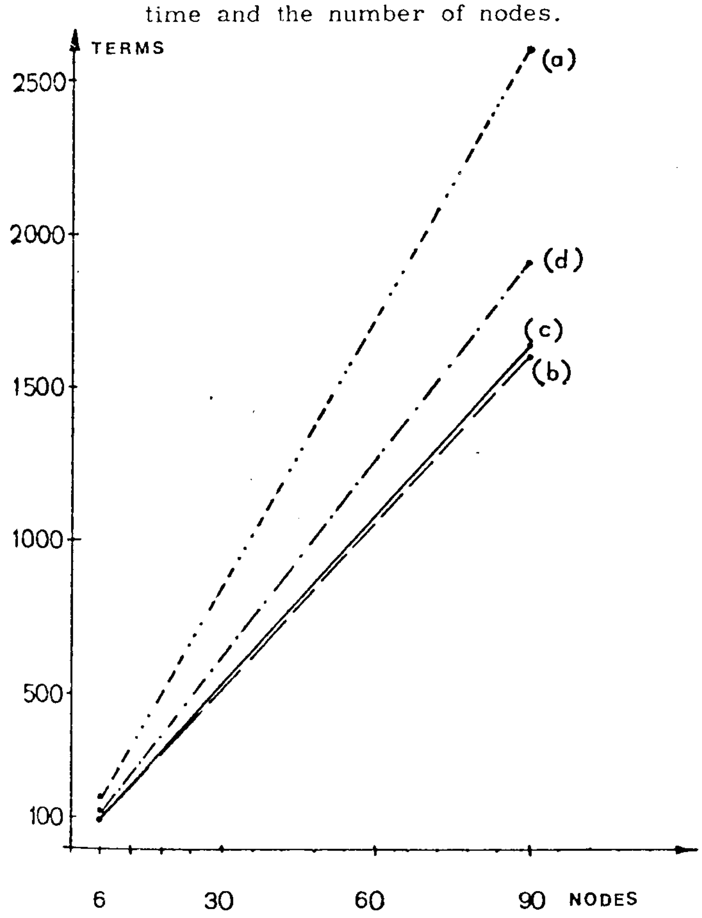

Fig. 6. Relationship between the number of terms and the number of nodes.

8. REFERENCES

[1] Centkowski G., J. Starzyk and E. Sliwa, "Symbolic analysis of large LLS networks by means of upward hierarchical analysis". submitted to ECCTD'81.

[2] Chen W.K., "Applied graph theory-graphs and electrical networks" North-Holland P.C. 1976.

[3] Kończykowska A., J. Starzyk, "Computer analysis of large signal flowgraphs by hierarchical decomposition method" ECCTD'80, Warsaw.

[4] Starzyk J., "Signal flow-graph analysis by decomposition method", IEE Journal on Electronic Circuits and System no 2, Apr. 1980.