Road Map Representation of *s*-Expanded Symbolic Network Functions

Marian Pierzchała
Institute of Telecommunication, Teleinformatics and Acoustics, Wrocław University of Technology (WUT), Wrocław, Poland. E-mail: marian.pierzchala@pwr.wroc.pl

Benedykt Rodanski
University of Technology, Sydney (UTS), Faculty of Engineering, P.O. Box 123, Broadway, NSW 2007, Australia.
E-mail: ben.rodanski@uts.edu.au

***Abstract*—A new two-graph approach is presented for the generation of exact symbolic network functions in the form of rational polynomials of the complex frequency variable s for analogue circuits. The closed-loop circuit representation is used, but the numerator and denominator of the desired transfer function are obtained separately without using any sorting process. In this method each symbolic expression for a coefficient at particular power of *s* is obtained immediately without any operation on polynomials. For compact representation of terms in symbolic network functions a new notation of expressions called the road map is presented. This notation has also the advantage that in the process of the repetitive evaluation of transfer function for the analyzed network we do not have to calculate the whole expression but only to update the terms which are changed. We consider R,C,L,E,I,cs networks (R,C,L,E,I,cs stand for resistors, capacitors, inductors, independent voltage and current sources, and all types of controlled sources, respectively).**

I. Introduction

Comparison of the efficiencies of the improved determinant decision diagram (DDD) based and two-graph based term generation techniques [1] leads to the conclusion that the DDD based algorithm performs better than the two-graph based algorithm, reported in [2]. However, even this improved DDD method suffers from the term cancellation problem. In order to avoid generating the canceling terms a special technique, called *just-in-time,* is needed.

The important advantage of the two-graph method is that, intrinsically, no canceling terms are generated. We must only improve its efficiencies. Following are the major sources of inefficiencies in the two-graph method implementation:

- topological procedures for enumeration of common spanning trees and for determining the sign of the term,
- using a sorting process for separating the numerator and denominator terms in closed-loop circuit representation,
- the lack of compact representation of terms,
- the need of recalculation of the entire formula, even if we change the value of a single element in the circuit,
- in the classical form, this method works only for the RLCg_m networks (for networks containing other elements, some preliminary circuit transformation might be required [3]).

This paper presents a modification of the classical two-graph method which removes these deficiencies. It is based on a modification of the Binet-Cauchy theorem [4] and a new method of representation of terms in the symbolic transfer function of a circuit, called the *road map* [5].

II. Symbolic Formulae for Networks Modelled with R,C,L,E,I,cs Elements

The networks to be considered are connected, linear, time-invariant active or passive networks modeled with R,C,L,E,I,cs (R,C,L,E,I,cs stand for resistors, capacitors, inductors, independent voltage and current sources, and all four types of controlled sources, respectively). Such network will be represented topologically by a linear graph G consisting of a voltage graph G_V and a current graph G_I. For a common tree T we can construct the fundamental loop and cut-set matrices, $\mathbf{B}_T^V$ and $\mathbf{Q}_C^I$, and two component matrices $\mathbf{Z}_T$ and $\mathbf{Y}_C$. If the two-graph element stamps, developed in [6], are used, matrices $\mathbf{Z}_T$ and $\mathbf{Y}_C$ are diagonal for all dependent sources. The hybrid system of equations for such network (assuming the closed system, i.e., the input source replaced by a source controlled by the output variable) is

$$\mathbf{H}\,\mathbf{x}=\begin{bmatrix}-\mathbf{I} & \mathbf{0} & \mathbf{Z}_T & \mathbf{0}\\ \mathbf{B}_T^V & -\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{Q}_C^I\\ \mathbf{0} & \mathbf{Y}_C & \mathbf{0} & -\mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{v}_T\\ \mathbf{v}_C\\ \mathbf{i}_T\\ \mathbf{i}_C\end{bmatrix}=\mathbf{0} \qquad (1)$$

It was shown in [5] that the determinant of $\mathbf{H}$ can be written as:

$$\begin{aligned}\det(\mathbf{H}) &= 1+\sum_{i=1}^{n_1}\mathrm{sgn}(i)\,b_i+\sum_{i=1}^{n_2}\mathrm{sgn}(i)\prod_{k=1}^{2}b_{i_k}+\ldots+\sum_{i=1}^{n_z}\mathrm{sgn}(i)\prod_{k=1}^{z}b_{i_k}\\ &= 1+\sum_{i=1}^{m_1}\mathrm{sgn}(i)\,t_i+\sum_{i=1}^{m_2}\mathrm{sgn}(i)\prod_{k=1}^{2}t_{i_k}+\ldots+\sum_{i=1}^{m_z}\mathrm{sgn}(i)\prod_{k=1}^{z}t_{i_k}\end{aligned} \qquad (2)$$

where:

n_1 (m_1) – number of non-zero elements in $\mathbf{B}_\mathrm{T}^\mathrm{V}$ ($\mathbf{T}_\mathrm{C}^\mathrm{I}$),

n_2 (m_2) – number of representative combinations of two non-zero elements,

n_z (m_z) – number of representative combinations of z non-zero elements,

$z = \min(m,n)$; m,n – number of rows and columns in $\mathbf{B}_\mathrm{T}^\mathrm{V}$ ($\mathbf{T}_\mathrm{C}^\mathrm{I}$), respectively,

b_i (t_i) – ith non-zero element in the product matrices $\mathbf{PB}_\mathrm{T}^\mathrm{V}$ ($\mathbf{PT}_\mathrm{C}^\mathrm{I}$),

b_{ik} (t_{ik}) – kth component of the product of non-zero elements,

$\mathrm{sgn}(i) = \det[\mathbf{B}_\mathrm{T}^\mathrm{V}(I_l, J_l)]\det[\mathbf{T}_\mathrm{C}^\mathrm{I}(I_l, J_l)]$,

$\mathbf{T}_\mathrm{C}^\mathrm{I}(I_u, J_v) = \left[-\mathbf{Q}_\mathrm{C}^\mathrm{I}(J_v, I_u)\right]^t$,

I_u, J_v – the sets of row and column indices, specifying how the submatrices are selected from $\mathbf{B}_\mathrm{T}^\mathrm{V}$ and $\mathbf{T}_\mathrm{C}^\mathrm{I}$.

By representative combination we mean a single combination of non-zero elements in a submatrix $\mathbf{PB}_\mathrm{T}^\mathrm{V}(I_u, J_v)\left[\mathbf{PT}_\mathrm{C}^\mathrm{I}(I_u, J_v)\right]$ which represents the determinant of this submatrix (in general, we can have a number of such combinations).

The product matrices $\mathbf{PB}_\mathrm{T}^\mathrm{V}$ and $\mathbf{PT}_\mathrm{C}^\mathrm{I}$ are closely related to the fundamental loop and cut-set matrices $\mathbf{B}_\mathrm{T}^\mathrm{V}$ and $\mathbf{T}_\mathrm{C}^\mathrm{I}$: their non-zero elements are the (symbolic) products of the respective row and column labels (with the complex frequency s omitted). Both matrices are formulated as follows:

i) Label the ith row and the jth column of $\mathbf{B}_\mathrm{T}^\mathrm{V}$ ($\mathbf{T}_\mathrm{C}^\mathrm{I}$) with symbols $S_\mathrm{C}(i)$, $S_\mathrm{T}(j)$ (i=1,2,…,m, j=1,2,…,n), representing the corresponding circuit elements,

ii) Substitute each non-zero element of $\mathbf{B}_\mathrm{T}^\mathrm{V}$ ($\mathbf{T}_\mathrm{C}^\mathrm{I}$) by: b_{ij} (t_{ij}) ← $S_\mathrm{C}(i)S_\mathrm{T}(j)$.

Although (1) is obtained topologically, the procedure for its determinant evaluation is purely numerical [all submatrices $\mathbf{B}_\mathrm{T}^\mathrm{V}(I_u, J_v)$ and $\mathbf{T}_\mathrm{C}^\mathrm{I}(I_u, J_v)$ are unimodular] and no sign rule is required. The network in this method is described by a closed signal-flow graph, thus the numerator and denominator of the desired transfer function are obtained separately, without employing a sorting process.

A. Obtaining the terms at different powers of s

If capacitors and inductors are in both the tree and the cotree, the fundamental circuit matrix $\mathbf{B}_\mathrm{T}^\mathrm{V}$ and the cut-set matrix $\mathbf{T}_\mathrm{C}^\mathrm{I} = \left(-\mathbf{Q}_\mathrm{C}^\mathrm{I}\right)^t$, and the associated product matrices $\mathbf{PB}_\mathrm{T}^\mathrm{V}$ and $\mathbf{PT}_\mathrm{C}^\mathrm{I}$ can be partitioned as shown in Tab. I (only $\mathbf{PB}_\mathrm{T}^\mathrm{V}$ is shown, as both matrices have identical structure at this level of detail).

TABLE I

$\mathbf{PB}_\mathrm{T}^\mathrm{V}$	E	$1/s\mathrm{C_T}$	$s\mathrm{L_T}$	$\mathrm{R_T}$
I	$\mathbf{PB}_{\mathrm{I,E}}$	$\mathbf{PB}_{\mathrm{I,C_T}}$	$\mathbf{PB}_{\mathrm{I,L_T}}$	$\mathbf{PB}_{\mathrm{I,R_T}}$
$s\mathrm{C_C}$	$\mathbf{PB}_{\mathrm{C_C,E}}$	$\mathbf{PB}_{\mathrm{C_C,C_T}}$	$\mathbf{PB}_{\mathrm{C_C,L_T}}$	$\mathbf{PB}_{\mathrm{C_C,R_T}}$
$1/s\mathrm{L_C}$	$\mathbf{PB}_{\mathrm{L_C,E}}$	$\mathbf{PB}_{\mathrm{L_C,C_T}}$	$\mathbf{PB}_{\mathrm{L_C,L_T}}$	$\mathbf{PB}_{\mathrm{L_C,R_T}}$
$\mathrm{G_C}$	$\mathbf{PB}_{\mathrm{G_C,E}}$	$\mathbf{PB}_{\mathrm{G_C,C_T}}$	$\mathbf{PB}_{\mathrm{G_C,L_T}}$	$\mathbf{PB}_{\mathrm{G_C,R_T}}$

The form of individual elements of the product matrices is determined by the symbolic product of their respective row and column labels. For example: elements of $\mathbf{PB}_{\mathrm{C_C,L_T}}$ have the form $s^2C_\mathrm{x}L_\mathrm{y}$, elements of $\mathbf{PT}_{\mathrm{L_C,R_T}}$ have the form $s^{-1}R_\mathrm{y}/L_\mathrm{x}$, etc. (In practical implementation there is no need to carry the s^x symbol, as it is obvious from the position of the element in the matrix.)

Thus, as we generate each product term in (2), it is possible to designate its character; no additional sorting is needed. This process yields the network functions in the form

$$T\left(s,s^{-1}\right)=\frac{N\left(s,s^{-1}\right)}{D\left(s,s^{-1}\right)}=\frac{n_n\, s^n+\ldots+n_1\, s+n_0+n_1^*\, s^{-1}+\ldots+n_m^*\, s^{-m}}{d_d\, s^d+\ldots+d_1\, s+d_0\, +d_1^*\, s^{-1}\ldots+d_e^*\, s^{-e}} \tag{3}$$

If one prefers the classical form of the network function (with only nonnegative powers of s), it can be obtained from (3) trivially.

The proposed method is best illustrated with an example.

Example 1: Consider a simple circuit shown in Fig. 1 [7]. It is required to find the symbolic expression for the input impedance $Z_\mathrm{in} = v_1/I_\mathrm{in}$. The independent current source I_in is replaced by the current source controlled by the output variable v_1: $I_\mathrm{in} = 1v_1$, thus forming the closed system [3].

If tree T is chosen to consist of branches (R_1, C_2, C_3), the matrices $\mathbf{B}_\mathrm{T}^\mathrm{V}$ and $\mathbf{T}_\mathrm{C}^\mathrm{I} = \left(-\mathbf{Q}_\mathrm{C}^\mathrm{I}\right)^t$ for this circuit are identical; also the product matrices $\mathbf{PB}_\mathrm{T}^\mathrm{V}$ and $\mathbf{PT}_\mathrm{C}^\mathrm{I}$ are identical. Table II shows the fundamental loop matrix $\mathbf{B}_\mathrm{T}^\mathrm{V}$ and its associated product matrix $\mathbf{PB}_\mathrm{T}^\mathrm{V}$.

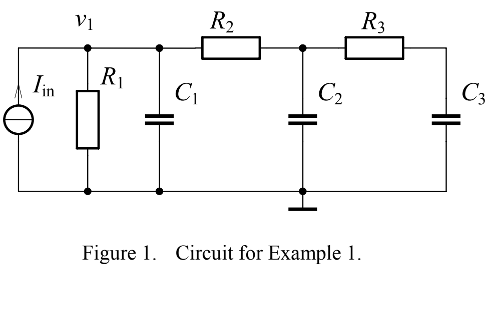

Figure 1. Circuit for Example 1.

TABLE II

$\mathbf{B}_{\mathrm{T}}^{\mathrm{V}}$	$1/sC_2$	$1/sC_3$	R_1
I_{in}	0	0	-1
sC_1	0	0	-1
G_2	1	0	-1
G_3	-1	1	0

$\mathbf{PB}_{\mathrm{T}}^{\mathrm{V}}$	$1/sC_2$	$1/sC_3$	R_1
I_{in}	0	0	R_1
sC_1	0	0	C_1R_1
G_2	G_2/C_2	0	G_2R_1
G_3	G_3/C_2	G_3/C_3	0

Applying the modification of the Binet-Cauchy theorem [5], we can now write directly the coefficients of numerator and denominator polynomials in (3):

$n_0 = b_{13}$; $n_1^* = b_{13}b_{31} + b_{13}b_{41} + b_{13}b_{42}$; $n_2^* = b_{13}b_{31}b_{42}$;
$d_1 = b_{23}$; $d_0 = b_{23}b_{31} + b_{23}b_{41} + b_{23}b_{42} + b_{33} + 1$;
$d_1^* = b_{23}b_{31}b_{42} + b_{31} + b_{33}b_{41} + b_{33}b_{42} + b_{41} + b_{42}$; $d_2^* = b_{31}b_{42}$;

where:

$b_{13}=R_1$; $b_{23}=R_1C_1$; $b_{31}=G_2/C_3$; $b_{33}=G_2R_1$; $b_{41}=G_3/C_2$; $b_{42}=G_3/C_3$.

Multiplying the numerator and the denominator of the transfer function by s^2 we obtain the input impedance as

$$Z_{\mathrm{in}} = \frac{n_0 s^2 + n_1^* s + n_2^*}{d_1 s^3 + d_0 s^2 + d_1^* s + d_2^*}.$$

B. Time complexity of the term generation phase

The maximum number of combinations we must examine in this method is equal to

$$\sum_{i=1}^{z} \binom{a}{i} - b = t + c + d, \tag{4}$$

where: a is the number of non-zero elements in the fundamental loop (cut-set) matrix,
b is the number of combinations of the non-zero elements which are in the same rows or columns,
t is the total number of terms in the transfer function,
c is the number of the non-representative combinations,
d is the number of combinations whose corresponding determinant is zero.

Thus, the time complexity of the term generation phase in this method is O(t+c+d). In many practical networks $c+d << t$ and the time complexity can be estimated by O(t). In Example 1: $s = 3$, $a = 6$, $b = 24$, $t = 17$, $c = 0$, $d = 0$.

C. Compact storage of the product terms

For large networks the number of product terms in matrix determinant is very large. We need therefore a compact method to simply represent the determinants of network functions.

Many methods for compact representation of terms in network determinants have been reported in the literature (see, for example, [1]). In this paper we propose a method for representation of terms in the circuit determinant in the form of a road map. Formally, a road map is a signed, directed, acyclic graph with parallel edges. The graph contains n_{v} vertices, the root (the start vertex) and the end vertex. The number of vertices n_{v} is equal to the number of non-zero elements in the product matrix $\mathbf{PB}_{\mathrm{T}}^{\mathrm{V}}$ or $\mathbf{PT}_{\mathrm{C}}^{\mathrm{I}}$ [we choose the matrix with fewer non-zero elements, so $n_{\mathrm{v}} = \min(n_1,m_1)$]. Each vertex k ($1 \le k \le n_{\mathrm{v}}$) represents a symbolic expression in the product matrix (e.g., in Example 1: $b_{33} = G_2R_1$). A symbolic expression for a coefficient at particular power of s contains a sum of products of these elements (e.g., in Example 1: $d_1^* = b_{23}b_{31}b_{42} + b_{31} + b_{33}b_{41} + b_{41} + b_{42}$; the first product term may be thought of as a road from b_{23} via b_{31} to b_{42}). Each road number is complemented by a symbol: N for the roads which are related to the numerator and D for the roads which are related to the denominator. Moreover, each road element X is given a label $L = L_{\mathrm{a}};L_{\mathrm{b}}$, representing the status of this element as follows:

- If X is a terminal vertex of the road, then L_{a} is denoted by E and L_{b} is equal to the value of the road determinant (+1 or -1),
- If X is a non-terminal vertex of the road, then L_{a} represents the next vertex and L_{b} is denoted by P.

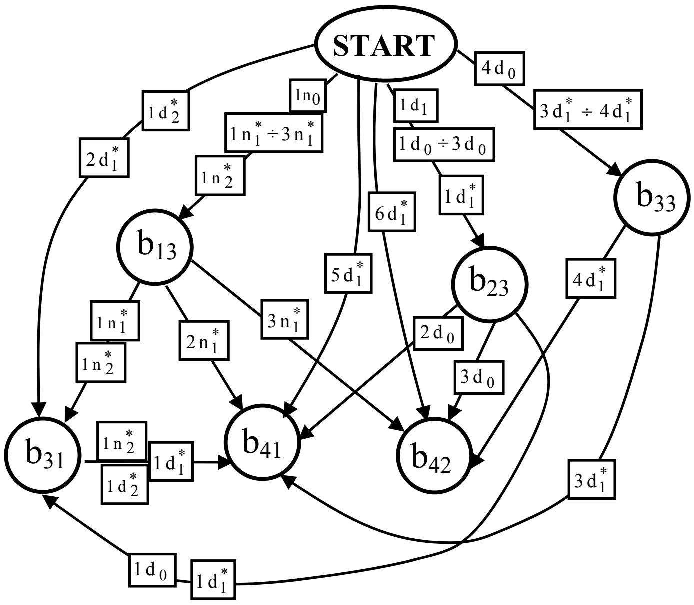

Figure 2. The road map of the example circuit (without the end vertex).

TABLE III

Start		1,3		2,3		3,1		3,3		4,1		4,2	
RN	L	RN	L	RN	L	RN	L	RN	L	RN	L	RN	L
$1n_0$	1,3;P	$1n_0$	E;1	$1d_0$	3,1;P	$1n_1^*$	E;1	$4d_0$	E;1	$2n_1^*$	E;1	$3n_1^*$	E;1
$1n_1^*\div 3n_1^*$	1,3;P	$1n_1^*$	3,2;P	$2d_0$	4,1;P	$1n_2^*$	4,1;P	$3d_1^*$	4,1;P	$1n_2^*$	E;1	$3d_0$	E;1
$1n_2^*$	1,3;P	$2n_1^*$	4,1;P	$3d_0$	4,2;P	$1d_0$	E;1	$4d_1^*$	4,3;P	$2d_0$	E;1	$4d_1^*$	E;1
$1d_0\div 3d_0$	2,3;P	$3n_1^*$	4,2;P	$1d_1^*$	3,1;P	$1d_1^*$	4,1;P			$1d_1^*$	E;1	$6d_1^*$	E;1
$4d_0$	3,3;P	$1n_2^*$	3,1;P			$2d_1^*$	E;1			$3d_1^*$	E;1		
$5d_0$	E;1					$1d_2^*$	4,1;P			$5d_1^*$	E;1		
$1d_1^*$	2,3;P									$1d_2^*$	E;1		
$2d_1^*$	3,1;P												
$3d_1^*\div 4d_1^*$	3,3;P												
$5d_1^*$	4,1:P												
$6d_1^*$	4,2;P												
$1d_2^*$	3,1;P												

Now, we can illustrate the network function of Example 1 in the form of the road map (Fig. 2). The graph has six vertices (nodes), corresponding to the non-zero elements of the product matrix $\mathbf{PB}_{\mathrm{T}}^{\mathrm{V}}$. Each road starts at the root node. Every link is labeled with all road numbers that pass between the two nodes. By following particular road label we can easily construct the product term that this road represents.

In this case, the graph representing the network function is simpler and more compact than the corresponding *s*-expanded DDD graph, presented in [1] and [7].

For the sake of notational simplicity the road map may be represented by a list of destinations (Table III). Each ‘road’ is simply a term in the transfer function. Every vertex has a list of roads that pass through it.

To illustrate the process of obtaining a coefficient at particular power of s, suppose we want to obtain the coefficient d_0 in Example 1, using the road map. In the list of roads, associated with the start vertex, we find that the first road of interest, $1d_0$, begins at b_{23}. In the list of roads passing through b_{23} we find that $1d_0$ continues to b_{31}. There, we find that the road ends and its sign is +1 (label E;1). Thus, the first term of d_0 is $+b_{23}b_{31}$. Repeating this process for roads $2d_0$ – $5d_0$, we obtain the symbolic value of the required coefficient.

The road map has additional advantage: in the repetitive evaluation of transfer function we do not have to calculate the entire determinant but only to update the terms which are changed. For example, when we change the value of C_1 (b_{23}) we must recalculate only the terms related to the following roads: $1d_0$, $2d_0$, $3d_0$ and $1d_1^*$.

This form of linked list representation could be easily stored and processed in any list language (LISP – for example).

III. CONCLUSIONS

This paper presents a new method for generation of exact symbolic network functions in the form of rational polynomials in the complex frequency variable s. The method is applicable to analogue circuits, modeled with resistors, inductors, capacitors, independent voltage and current sources and all types of controlled sources. In this method each symbolic expression for a coefficient at particular power of s is obtained separately, without any operation on polynomials. The procedures for their evaluation are purely numerical, involving only integer arithmetic. The method guarantees that no term cancellation occurs.

For the sake of notational simplicity a new method of representation of terms in network function, called the *road map*, is introduced. The road map is a linked list – a structure easy to store and process – and very efficient in repetitive calculations.

REFERENCES

[1] W.Verhagen and G.Gielen, “Efficient DDD-based symbolic analysis of linear analog circuits,” *IEEE Trans. CAS II*, vol. 49, pp. 474-487, July 2002.

[2] P.Wambacq, G.G.E.Gielen and W.Sansen, “Symbolic network analysis methods for practical analog integrated circuits: A survey,” *IEEE Trans. CAS II*, vol. 45, pp. 1331-1341, Oct. 1998.

[3] P.M.Lin, *Symbolic Network Analysis.* New York, Elsevier, 1991.

[4] M.Pierzchała, “Modification of the Binet-Cauchy theorem,” *Applicationes Mathematicae*, vol. XVII, no. 1(1980), pp. 131-141, (in Russian).

[5] M.Pierzchała and B.Rodanski , ”Numerical method for calculating the terms of the closed system determinant in two-graph representation of electronic circuits,” *Proc. SMACD’06*, Oct. 2006, Firenze, Italy.

[6] M.Pierzchała and B.Rodanski, “Two-graph stamps for linear controlled sources,” *Proc. SMACD’04*, Sep. 2004, Wrocław, Poland.

[7] C-J.R.Shi and X-D.Tan, “Compact representation and efficient generation of *s*-expanded symbolic network function for computer-sided snalog circuits design,” *IEEE Trans. CAD*, vol. 20, pp. 813-827, July 2001.