Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

Symbolic analysis of electric circuits using the program SALEC

D.V. Tošić, B.D. Reljin
Faculty of Electrical Engineering, Belgrade, Yugoslavia

Abstract

A computer tool, the original program SALEC, for symbolic analysis of linear, lumped, time-invariant electric circuits is presented. The analysis is based on the reduced modified nodal analysis and is carried out in the complex domain of the Laplace transform. The library of circuit components, the basic circuit elements and general multiport networks specified by their matrix parameters, is introduced. The initial conditions for capacitors and inductors are taken into account. An algorithm for analysis of circuits with disconnected graphs is proposed. The program operation is demonstrated by an example.

1 Introduction

Symbolic analysis can be described as a technique utilized at the circuit level to calculate the behavior or a characteristic of a circuit with some or all of the circuit elements represented by symbols. It is a method used to obtain insight into the circuit's behavior, to generate analytic models for automated circuit sizing, and to evaluate and determine circuits characteristics, such as input impedance, voltage feedback ratio, current gain, output admittance, and power supply rejection ratio. This is complementary to numerical analysis (where the variables and the circuit elements are represented by numbers) and qualitative analysis (where only qualitative values are used for voltages and currents, such as increase, decrease or no change) Chua[1], Lin[2], Gielen[3], Huelsman[4], Gielen[5].

A *symbolic simulator* is a computer program that receives the circuit description as input and can automatically carry out the symbolic analysis and generate the symbolic expression (closed form, analytic) for the desired circuit characteristic or response (voltage or current) Gielen[3,5]. In recent years

Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

several symbolic simulators were reported: ISAAC Gielen[6], ASAP Fernandez[7], SYNAP Seda[8], SAPEC Manetti[9], SSPICE Wierzba[10], SCYMBAL Konczykowska[11], SCAPP Hassoun[12], GASAP Hulesman[13] and SAPWIN Liberatore[14]. Also, due to the enormous increase in computing power of the present computers, symbolic analysis gained a renewed and growing interest Wambacq[15], Chang[16], Hassoun[17].

The above mentioned simulators were written in FORTRAN, C or Lisp. Their functionality was determined primarily by the tasks they were targeted at.

This paper presents a new symbolic simulator, SALEC, intended for symbolic evaluation of linear, lumped, time-invariant electric circuits Tošić[18,19,20].

2 SALEC - A New Symbolic Simulator

The program SALEC is intended to be a computer-aided tool for calculating the complete circuit response and characteristics. Except analysis, it comprises a set of functions for symbolic synthesis and design of a class of linear networks Hribšek[21,22], Tošić[23]. It takes into account initial conditions (initial energy) of inductors and capacitors. Also, it handles circuits with disconnected graphs. The analysis algorithm inside SALEC is based on nodal approach: Compacted Modified Nodal Analysis (CMNA) Gielen[3] and Reduced MNA (RMNA) Lee[24]. SALEC is developed in *Mathematica* Wolfram[25].

Prior to symbolic analysis energized capacitors and inductors are replaced with parallel connections of empty (zero initial condition) elements and the corresponding current sources. The current sources are preferred because they are well suited for the nodal approach - they do not increase the number of nodes Reljin[26,27]. Next, the circuit graph is examined. If it is disconnected, auxiliary grounded voltage sources are added to make it connected. This is exemplified by Figs. 1 and 2 showing a simple circuit containing a floating linear inductive transformer. The two voltage sources are introduced, E_1 and E_2, to make the circuit graph connected and to enable formulation of nodal equations for CMNA/RMNA.

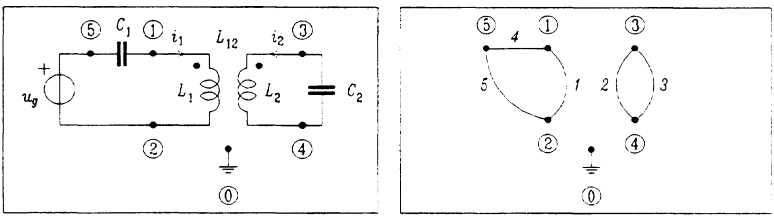

Figure 1. A simple circuit with the disconnected graph.

Since only branch voltages and currents are important the auxiliary sources cancel out in the final expression for the circuit response. It means that the voltages of these sources can have arbitrary symbolic values. If required, these

Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

voltages can model prescribed potential values of the circuit floating parts. Since their currents are always equal to zero, they do not increase the number of MNA equations.

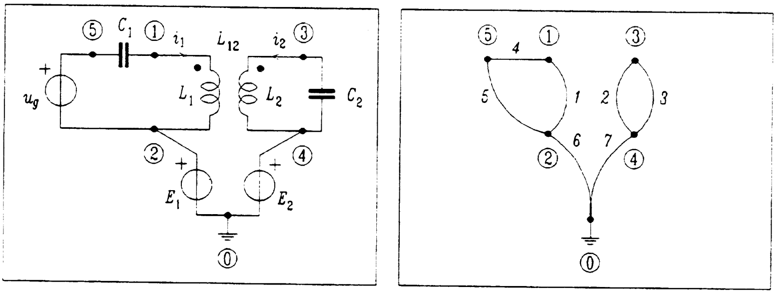

Figure 2. Adding voltage sources, E_1, E_2, to connect the graph of the circuit from Fig 1.

The circuit to be analyzed is described by its *net list* - the list of its *components*. A component can be a standard circuit *element*, a multiport *network* whose matrix parameters are known, or a special *functional block*.

The standard (basic) circuit elements are assumed to be *ideal* (pure resistors, inductors, capacitors, ideal operational amplifiers, etc.). The circuit nodes are consecutively numbered from 0 to n, the components are numbered from 1 to m. The zero node, 0, is the *reference node* (ground). The symbol s is an identifier designating the complex frequency and is a reserved name/symbol.

Standard oneport (2-terminal) elements are specified by a list of the form

component[k] = {"*type*", "*name*", $n1$, $n2$, *parameter*, *energy*}

where k represents the component unique sequence number and *type* identifies the component as: **R** resistor, **L** inductor, **C** capacitor, **Z** impedance, **Y** admittance, **G** conductance, **V** voltage source, **I** current source and **SHORT** short circuit (ideal ammeter). Each component can have a unique individual name, specified as the second item, *name*. The element terminals are $n1$ and $n2$, assuming the current direction from $n1$ to $n2$. The voltage and current reference directions are standard "associated" Chua[1]. The *parameter* is a symbolic expression that stands for the element value. Depending on the *type* it can be: resistance, inductance, capacitance, impedance, admittance, conductance, voltage or current. A **SHORT** component does not have this item. The initial conditions are taken into account by the last item, *energy*. For a capacitor it is the initial voltage, for an inductor it is the initial current. Only **C** and **L** components can have *energy*. For zero initial conditions this item can be dropped out.

Standard twoport (4-terminal) elements are defined by the list

component[k] = {"*type*", "*name*", {$n1$,$n2$}, {$n3$,$n4$}, *parameter*, *energy*}.

The identifiers k and *name* retain their meaning. The *type* can be: **OPAMP** operational amplifier (nullor), **VCVS** voltage-controlled voltage source (voltage amplifier), **VCCS** voltage-controlled current source (trans-conductance amplifier), **CCCS** current-controlled current source (current amplifier), **CCVS**

Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

current-controlled voltage source, **T** resistive transformer, **GYR** gyrator, **NIC** negative impedance converter, **NIV** negative impedance invertor, **ILT** inductive linear transformer (a system of two coupled inductors) and **LINE** a section of a uniform, homogeneous, lossless TEM transmission line. The first port of the component, $\{n1,n2\}$, refers to **OPAMP** outputs and to controlled ports of dependent sources. Except **OPAMP** components, twoport elements are characterized by a symbolic expression *parameter* (resistive transformer turn ratio, gyrator resistance, amplifier gain, electrical length of a line, etc.). For inductive transformers, **ILT**, it is a triplet of the form $\{L1,L2,M\}$, specifying the primary, secondary and mutual inductance. **ILT** components can optionally have *energy*. It is a pair $\{I01,I02\}$ taking into account the initial conditions for the transformer currents.

A SALEC component can be a *multiport network* characterized by matrix parameters. It is specified by a list of the form

component[k] = {"*type*", "*name*", *pterm*, *nterm*, *parameters*}.

The *type* can be: **YNET**, **ZNET**, **ANET**, **HNET**, **KNET** and **SNET**, referring to networks characterized by y-, z-, $ABCD$-, h-, k- and S-parameters, respectively. Except **SNET**, the matrix parameters are given in the form of nested lists $\{\{p11,p12,p13...\},\{p21,p22,p23...\},\{p31,p32,p33,...\}...\}$ and assume passive networks. The identifier *parameter* stands for such a list whose items can be arbitrary symbolic expressions. For **SNET** components the *parameter* is a triplet $\{S,Z,b\}$ describing the most general case: S-matrix, S, port reference impedances, Z, and equivalent independent sources, b Tošić[20]. It is the preferred characterization of a network, for the most general case, even when independent sources are contained in the network Tošić[20]. The number of ports for these components (networks) is arbitrary for **YNET**, **ZNET** and **SNET**, while **ANET**, **HNET** and **KNET** are restricted to 2-ports only. The network terminals are specified by *pterm* and *nterm*. Both are the lists: *pterm*=$\{p1,p2,p3,...\}$ contains the "positive" nodes, and *nterm*=$\{n1,n2,n3,...\}$ the "negative" ones. The ports are: ($p1,n1$), ($p2,n2$), ..., etc. The current reference direction is from $p1$ to $n1$, from $p2$ to $n2$, ... and so on. For grounded networks *nterm* is a list of zeros.

A special group of components are *functional blocks* intended for *behavioral* modeling of devices. They are generalized, grounded VCVSs and are specified by a list of the form

component[k] = {"*type*", "*name*", *output*, *input*(s), *parameter*(s)}.

The *type* can be: **SUM** single-output multiple-input summer, **DIF** differentiator, and **INT** integrator. All components have one output terminal, *output*, and the input currents are equal to zero. For **DIF/INT** components the output voltage is proportional to the derivative/integral of the input voltage. The *parameter* item defines the corresponding multiplicative constant of a differentiator/integrator. For **SUM** components the output voltage is a weighted sum of the input node voltages, *input*(s)=$\{n1,n2,n3,...\}$, with weights given by the *parameter*= $\{w1,w2,w3,...\}$. The weights are arbitrary symbolic expressions.

The circuit to be analyzed is described by an ASCII file structured as:

Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

numberofnodes = *n*
component[1] = {"*type*_1", "*name*_1", ...}
component[2] = {"*type*_2", "*name*_2", ...}
component[3] = {"*type*_3", "*name*_3", ...}
...
component[*m*] = {"*type*_*m*", "*name*_*m*", ...}
numberofcomponents = *m*

It is the *native* circuit specification for SALEC and can be created by a text editor or visually, by a schematic editor (currently under development).

The program SALEC receives this file as input, scans the component list and formulates MNA (**M**odified **N**odal **A**nalysis) system of circuit equations in the matrix form. The equivalent current sources due to the initial conditions, and auxiliary voltage sources to make the circuit graph connected are automatically included if/when necessary. When calculating characteristics or steady-state response the initial conditions are ignored. The analysis is performed in the complex domain (the one-sided *Laplace* transform) for the frequency variable *s*. The initial conditions are defined at t=0, and the analysis finds the response valid for $t \geq 0$. The MNA matrices can be displayed if required, for example in education Tošić[19]. To compact the MNA matrix of a circuit, and reduce the number of equations and variables, the methodology of RMNA (**R**educed **MNA**) Lee[24] and CMNA (**C**ompacted **MNA**) Gielen[3] is applied.

The symbolic solution of MNA system equations is carried out by the *Mathematica* `LinearSolve` function. The symbolic response is prepared for presentation by the use of the `Simplify`, `Together` and `Cancel` functions. For simpler expressions the Inverse Laplace transform, provided by *Mathematica*, is applied to find the time-domain response Wolfram[25]. Optionally, the response preparation for presentation can be avoided to minimize the SALEC processing time. The solution for the MNA variables is contained in the list `response` containing symbolic, closed-form (analytic) expressions for the node voltages and the additional currents required for MNA.

3 Symbolic Simulation Examples

To illustrate the symbolic analysis capabilities of SALEC consider a lumped, linear, time-invariant circuit shown in Fig. 3. It is the well known Wien-bridge oscillator. Let us assume that the resistances equal *R*, and that the capacitances equal *C*. The circuit is described in the text file `WIEN.S23`:

(* WIEN.S23 Wien Oscillator 13:49 28/12/95 *)
numberofnodes = **2**
component[1] = **{"C", "C1", 1, 0, C, U0}**
component[2] = **{"R", "R1", 1, 0, R}**
component[3] = **{"VCVS", "A", {2,0}, {1,0}, A}**
component[4] = **{"Z", "ZR2C2", 1, 2, R+1/(C*s)}**
numberofcomponents = **4**
(* WIEN.S23 eof *)

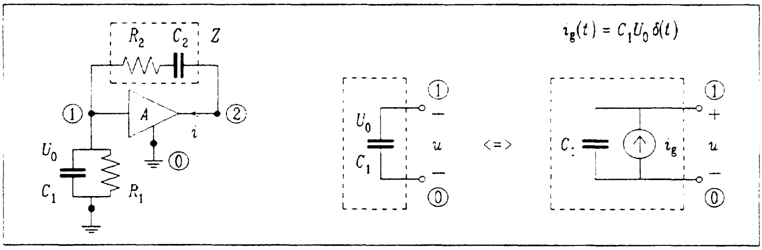

Figure 3. Wien-bridge oscillator and the current source for the initial condition.

To run SALEC issue the command

SALEC["WIEN.S23", MNAmode->All, MNAvariables->All].

It directs the simulator to exclude MNA equation reduction and to prepare all MNA variables (the node voltages and the required currents) for presentation. The simulation results follows.

$$V1 = \frac{C\,R\,(1 + C\,R\,s)\,U0}{1 + 3\,C\,R\,s - A\,C\,R\,s + C^2\,R^2\,s^2}$$

$$V2 = \frac{A\,C\,R\,(1 + C\,R\,s)\,U0}{1 + 3\,C\,R\,s - A\,C\,R\,s + C^2\,R^2\,s^2}$$

$$J1A = \frac{(-1 + A)\,C^2\,R\,s\,U0}{-1 - 3\,C\,R\,s + A\,C\,R\,s - C^2\,R^2\,s^2}$$

(SALEC execution: 3.02 Seconds, PC-i486/DX50)

Inspecting the response it is obvious that the well known condition A=3 must be met if sinusoidal waveforms are wanted.

The amplifier output voltage can be visualized, Fig. 4, as a function of the amplifier gain, A, and the imaginary part of the normalized complex frequency, $w=\omega RC$, $\omega=\mathrm{Im}(s)$. Again, it is clearly seen when the circuit will oscillate and at what frequency.

Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

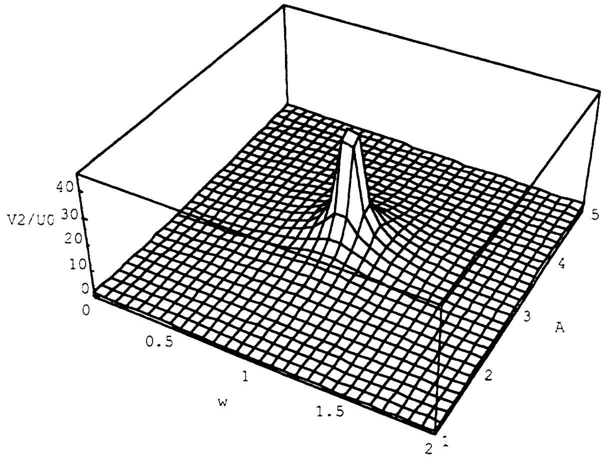

Figure 4. Modul of the output amplifier voltage as a function of frequency and gain.

The next example evaluates the Riordan's gyrator Reljin[26] used to simulate an inductor, Fig. 5. The only response required is the voltage across the current source.

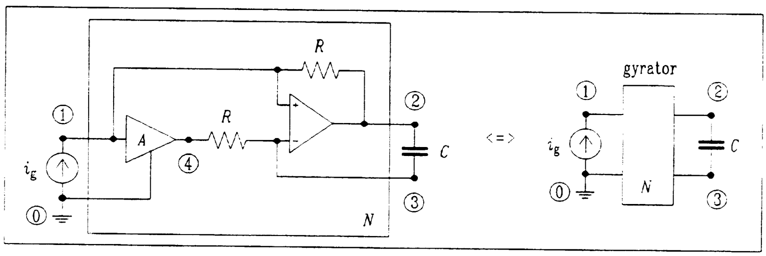

Figure 5. Riordan's gyrator simulating an inductor.

The SALEC command to analyze the circuit in Fig. 2, and the corresponding simulation output follows. In this run SALEC will work with the RMNA system of equations. The currents of the grounded voltage sources, normally involved in MNA, are excluded. This applies to both, independent and dependent sources (operational amplifiers, voltage amplifiers and current-controlled voltage sources)

SALEC["RIORDAN.S23", MNAmode->RMNA, MNAvariables->{1}]

(* RIORDAN.S23 Riordan's gyrator 10:29 29/12/95 *)
numberofnodes = 4
component[1] = {"I", "Ig", 0, 1, Ig}
component[2] = {"VCVS", "A", {4,0}, {1,0}, A}

```
component[3] = {"R", "R1", 4, 3, R}
component[4] = {"OPAMP", "OA", {2,0}, {1,3}}
component[5] = {"R", "R2", 1, 2, R}
component[6] = {"C", "C", 2, 3, C}
numberofcomponents = 6
(* RIORDAN.S23 eof *)
```

```
               2
        C Ig R   s
V1  =  ---------
         -1 + A
```

It is evident that the current source "sees" an inductor of inductance $L=CR^2/(A-1)$. For the particular value, $A=2$, the network N turns into a gyrator characterized by the gyrator resistance $r=R$. In this simulation only the voltage of the first node was prepared for presentation (the SALEC option, MNAvariables->{1}). In general, the preparation of the simulation results for the presentation of the specific form can be time-consuming, especially when complex symbolic expressions are involved.

4 Conclusion

A program for symbolic analysis of lumped, linear, time-invariant electric circuits (SALEC) is presented. It has a built-in library of all standard circuit elements, including several non-standard elements for behavioral and macro-modeling. It can analyze circuits containing networks specified by matrix parameters. All the circuit parameters can be given by symbolic expression. In the special case these expressions can be symbols or specific numerical values.

The analysis relies on MNA in the complex domain of the one-sided *Laplace* Transform, including the periodic steady-state (sinusoidal or non sinusoidal) analysis as a special case. Optionally, RMNA/CMNA can be involved to reduce the number of equations and variables.

If the circuit graph is not connected, a grounded auxiliary voltage source can be added to each floating subgraph. The voltages of these generators are arbitrary, they serve to enable the MNA formulation and do not affect the currents and voltages across the circuit branches. The program SALEC will check/examine the circuit graph, identify floating subgraphs (if any) and automatically add these generators.

The initial conditions are taken into account by equivalent current sources.

Functional blocks are provided as the built-in SALEC components to help behavioral and macro-modeling of devices. In conjunction with the SALEC matrix specified components they constitute a powerful basis for hierarchical circuit decomposition - breaking a large circuit into a smaller one consisting of the blocks whose matrix parameters are known Hassoun[17].

The above features (handling of disconnected graphs, initial conditions, functional blocks and matrix based components/networks) are not found in the existing symbolic simulators reported in the open literature.

Transactions on Engineering Sciences vol 11, © 1996 WIT Press, www.witpress.com, ISSN 1743-3533

Except analysis, SALEC has a set of internal modules for synthesis of simple filter structures, pole-zero extraction, symbolic approximation, response graphing and for calculation of matrix parameters Tošić[20].

The symbolic simulations are illustrated by two examples and carried out on a standard PC-i486 platform. For the example circuits it takes less than 10 seconds (50MHz CPU clock) to obtain the complete response, all MNA variables, including the preparation for the presentation.

Numerical simulators like SPICE Nagel[28] can accurately evaluate the circuit, for a given set of numerical parameter values, but there is no way to identify the contributions of each particular parameter itself. If repetitive calculations of the response are required, the numerical simulation can be time-consuming, and sometimes error-prone, especially if lossless circuits with pronounced resonant effects are analyzed Djordjević[29].

Using the computer program SALEC as a symbolic analysis tool novice (practicing) or experienced designers and researchers can examine different circuit topologies and design alternatives in a minute, which is virtually impossible by hand (in manual circuit response derivation).

References

[1] Chua, L.O. & Lin P.M. *Computer-Aided Analysis of Electronic Circuits*, Prentice Hall, Englwood Cliffs NJ, 1975.

[2] Lin, P.M. *Symbolic Network Analysis*, Elsevier, Amsterdam, 1991.

[3] Gielen, G. & Sansen, W. *Symbolic Analysis for Automated Design of Analog Integrated Circuits*, Kluwer, Norwell MA, 1991.

[4] Huelsman, L. & Gielen, G. (ed). *Symbolic Analysis of Analog Circuits: Techniques and Applications*, Kluwer, Norwell MA, 1993.

[5] Gielen, G. *et al.* Symbolic Analysis Methods and Applications for Analog Circuits: A Tutorial Overview, *Proc. of the IEEE*, 1994, **82**, 286-304.

[6] Gielen, G., Walscharts, H. & Sansen, W. ISAAC: A symbolic simulator for analog integrated circuits, *IEEE J. Solid-State Circ.*, 1989, **24**, 1587-1597.

[7] Fernandez, F., Rodrigez-Vazquez, A. & Huertas, J. A tool for symbolic analysis of analog integrated circuits including pole-zero extraction, pp. 752-761, *Proc. ECCTD*, 1991.

[8] Seda, S., Degrauwe, M. & Fichtner, W. A symbolic analysis tool for analog circuit design automation, pp. 488-491, *Proc. ICCAD*, 1988.

[9] Manetti, S. New approaches to automatic symbolic analysis of electric circuits, *Proc. IEE part G*, 1991, **138**, 22-28.

[10] Wierzba, G. *et al.* SSPICE - A symbolic SPICE program for linear active circuits, pp. 1197-1201, *Proc. Midwest Symp. on Circuits and Syst.*, 1989.

[11] Konczykowska, A. & Bon, M. Automated design software for switched-capacitor IC's with symbolic simulator SCYMBAL, pp. 363-368, *Proc. DAC*, 1988.

[12] Hassoun, M. & Lin, P.M. A new network approach to symbolic simulation of large scale networks, pp. 806-809, *Proc. ISCAS*, 1989.

[13] Hulesman, L. Personal computer symbolic analysis programs for undergraduate engineering courses, pp. 798-801, *Proc. ISCAS*, 1989.
[14] Liberatore, A., *et al.* A new symbolic program package for the interactive design of analog circuits, pp. 2209-2212, *Proc. ISCAS*, 1995.
[15] Wambacq, P., *et al.* Algorithm for efficient symbolic analysis of large analog circuits, *Electronic Letters*, 1994, **30**, 1108-1109.
[16] Chang, S.M. & Wierzba, G.M. Circuit Level Decomposition of networks with nullors for symbolic analysis, *IEEE Trans. CAS-I*, 1994, **41**, 699-711.
[17] Hassoun, M. & Lin, P.M. A hierarchical network approach to symbolic analysis of large-scale networks, *IEEE Trans. CAS-I*, 1995, **42**, 201-211.
[18] Tošić, D.V., Markoski, I.M. & Reljin, B.D. Symbolic analysis of electrical circuits, pp. III.3-III.8, *Proc. XXXVII Yugoslav conference ETAN*, Belgrade, Yugoslavia, 1993 (in Serbian).
[19] Tošić, D.V., Reljin, B.D. & Reljin, I.S. Educational aspect in symbolic analysis of linear electric circuits, pp. 533-536, *Proc. ISTET'95*, Thessaloniki, Greece, 1995.
[20] Tošić, D.V. & Reljin, B.D. Unified treatment of subnetworks in symbolic analysis of linear electric circuits, *Proc. BCOR'95*, Thessaloniki, Greece, 1995.
[21] Hribšek, M.F., Tošić, D.V. & Jokić, N. High-frequency continuos-time filters design using a single operational transconductance amplifiers, pp. 747-751, *Proc. MIEL'95*, Niš, Yugoslavia, 1995.
[22] Hribšek, M.F., Tošić, D.V. & Reljin, B.D. Simple generation and design method of second order filters with complex poles, pp. 497-499, *Proc. SIP-95*, Las Vegas, USA, 1995.
[23] Tošić, D.V., Hribšek, M.F. & Reljin, B.D. Generation and design of new continuous-time second order gain equalizers using program SALEC, *Archiv für Elektronik und Übertragungstechnik* (accepted, Mar. 1996).
[24] Lee, K. & Park, S.B. Reduced Modified Nodal Approach to Circuit Analysis, *IEEE Trans. Circuits and Syst.*, 1985, **32**, 1056-1060.
[25] Wolfram, S. *Mathematica: A System for Doing Mathematics by Computer*, Addison-Wesley, Redwood City CA, 2nd ed., 1991.
[26] Reljin, B.D. *Circuit Theory I: Time-Domain Analysis*, Nauka, Belgrade, 4th ed. 1995 (in Serbian).
[27] Reljin, B.D. *Circuit Theory II: Frequency-Domain Analysis*, Nauka, Belgrade, 3rd ed. 1995 (in Serbian).
[28] Nagel, L. SPICE2: A Computer Program to Simulate Semiconductor Circuits, *Memorandum No. M520*, University of California, Berkeley, California, May 1975.
[29] Djordjević, A.R., Reljin, B.D., Tošić, D.V. & Sarkar, T.K. Transmission-line theory approach to solution of state equations for linear lumped circuits, *IEEE Trans.* MTT (accepted, Mar. 1996).