SNAP – program with symbolic core for educational purposes

DALIBOR BIOLEK
Dept. of Telecommunications
Brno University of Technology
Purkynova 118, 612 00 Brno
CZECH REPUBLIC
dalibor.biolek@vabo.cz, http://www.fee.vutbr.cz/UREL/snap

Abstract: - **SNAP** (**S**ymbolic **N**etwork **A**nalysis **P**rogram) is a program for the symbolic, semisymbolic, and numerical analyses of linearized circuits. It serves as a supporting tool of teaching electrical engineering and electronics, as a student's tool for supporting their individual work and discovering phenomena in electronic circuits, and as an important program for designers.

Key-Words: - Symbolic analysis, semisymbolic analysis, numerical analysis, SNAP.

1 Introduction

The symbolic analysis enables us to obtain the desired circuit function (gain, impedance,...) as an equation with the symbols of circuit components and the Laplace operator s. The semisymbolic form is obtained from the symbolic one by substituting the numerical values of circuit parameters in place of their symbols. In this way, numerical values are obtained for all coefficients of the numerator and denominator polynomials in terms of the Laplace operator s. Then the zeros and poles, equations of step and pulse responses and their graphs, as well as graphs of the corresponding frequency responses are derived from the semisymbolic form. SNAP also enables the computation of all known two-port parameters of the analyzed circuit and the sensitivities of all circuit functions to any arbitrary parameters. These results are available in the symbolic, semisymbolic or numerical form.

The matrix analysis method based on the modified nodal approach is implemented. The required circuit functions are solved using the matrix subdeterminants. To speed up the symbolic analysis, the classical algorithm of determinant expansion has been modified. Thanks to this approach, the implemented method is not memory- and time- expensive, and the symbolic analysis is then very fast.

2 Program Conception

In the first step of analysis, a circuit scheme is created using the schematic editor. The editor transforms it into a SNAP netlist (a file with the extension *.snn*). Then the SNAP program is executed which reads the data from the netlist and then the required analysis is performed. The schematic editor, EDITOR.EXE, is currently supplied with the SNAP program**.** However, it is also possible to use an arbitrary schematic editor which generates a standard SPICE-format netlist. The program EDITOR.EXE utilizes the ASCII text file SNAP.LIB. In this file, the library of schematics and parameters of electrical components are defined by a special language. Editing this file, we can easily extend the set of components and modify the existing one.

The circuit schematic can be saved to the file with the standard extension .CIR.

The SNAP.EXE analyzer exploits an auxiliary ASCII text file SNAP.CDL. In this file, the mathematical models of the electrical components are defined. The modeling philosophy is based on the well-known modified nodal approach. The text in the SNAP.CDL file is in close relation to the definition of the circuit

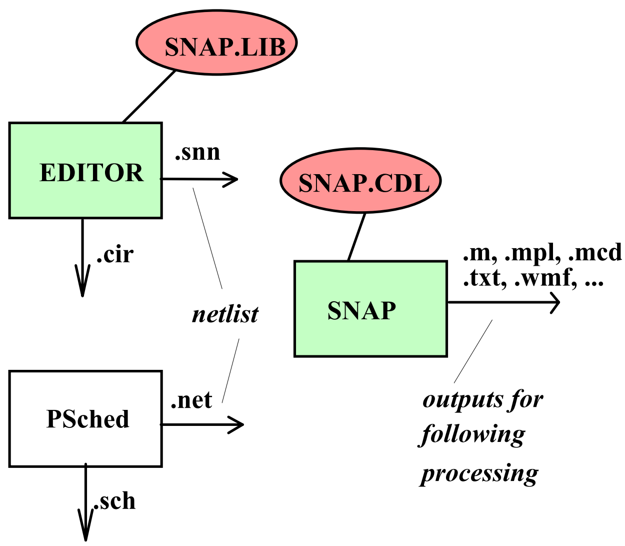


Fig. 1. Program conception.

elements in the SNAP.LIB file. In this way, the user can extend the SNAP performance without stint only by editing two text files.
The analysis results can be exported to the text (*.*txt*) or graphic (*.*wmf*) files, to the clipboard or directly to the printer, or to the corresponding Matlab (*.m), MathCad (*.*mcd*) or Maple (*.*mpl*) functions for additional processing.

3 Schematic Editor

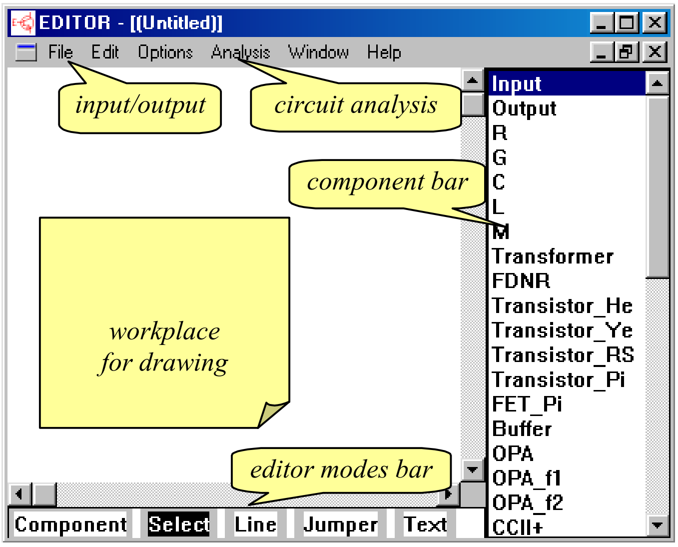


Fig. 2. Schematic editor.

Editor modes bar:

Component: Placing components on the workplace.

Select: Editing the already laid components, wires, and texts (deleting, rotation, moving, modification).

Line: Drawing the wires.

Jumper: Placing „jumpers“ for non-conducting wire crossing.

Text: Placing texts on the workplace (headings, notices,…).

Schematic editor has to contain elements of the “input” and “output” type. These elements are two-poles whose position in the schematic defines the input and the output gates of an analyzed circuit. The character of both the input and the output signal, i.e. voltage or current, is only determined when specifying the analyzed circuit functions in the SNAP program.
After positioning the schematic symbol on the schematic editor screen, clicking on the symbol in the SELECT regime will open the window for editing component parameters in the *Parameters* window. The item Name comprises the component symbol, which will appear in the resulting symbolic equation after the analysis. In the items for element parameters, the vicarious symbol % is normally placed.
For example, Clicking resistor R1 accesses the window in Fig. 3.

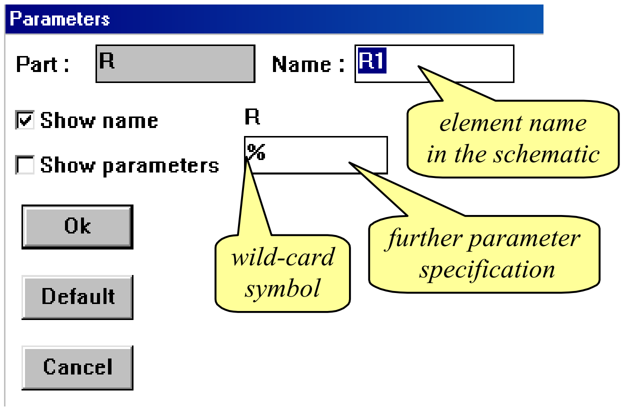


Fig. 3. The Parameters window.

<u>Denotation of the individual items:</u>
Part: Denotes the element type. R means resistor. The dark box background means that its content cannot be modified.
Name: Element denotation in the schematic. This denotation can be changed by the user.
R .. further parameter specification, in our case resistance of resistor R1. The wild-card symbol % has the following meaning:
It substitutes the contents of the **Name** item, i.e. R1 in our case. This term is then used in the equation of the symbolic result. For example, if we want to replace R1 with the shorter R in this equation, we have 2 possibilities:

- we write down directly R instead of %,
- we do not change the wild-card symbol, but we re-write the **Name** contents from R1 to R. However, the resistor designation in the schematic will be now R instead of R1.

If we want to specify the element parameter more closely, e.g. setting its numerical value R1=1kΩ, we can do it by ascribing %=1k (details can be found in [1]).

The following rule holds: *If in the schematic we enter all parameters of all circuit elements also numerically, then the SNAP performs not only the symbolic but also the semisymbolic and numerical analyses. In the opposite case, only the symbolic analysis is performed.*

After the netlist generation, the program SNAP can be started, which reads the netlist data necessary for a subsequent analysis. In case of SNAP executing directly from the EDITOR.EXE program, the sequence of netlist generation, SNAP starting, and data reading is performed automatically.

4 SNAP Analyzer

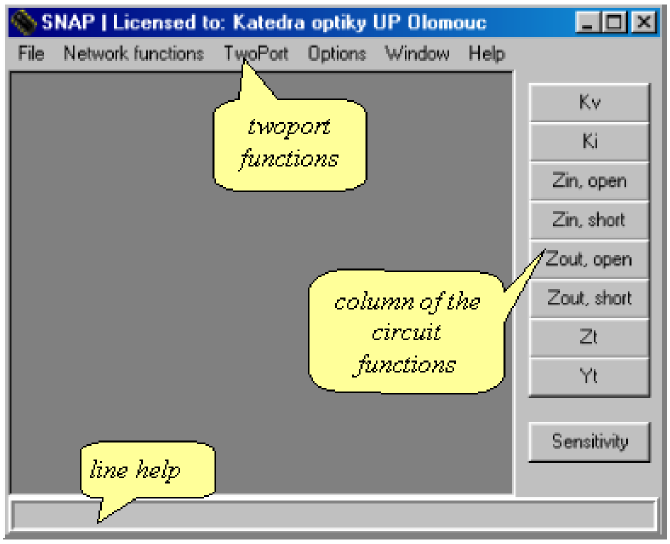


Fig. 4. Analyzer main window.

Column of the circuit functions consists a selection of parts from "Network functions" menu. The circuit input and output gates are defined by corresponding schematic elements in the circuit schematic and by the respective symbols *I* and *O* in the netlist.

The following *circuit functions* are available:

Voltage gain [-], open output
Current gain [-], shorted output
Input impedance [Ω], open output
Input impedance [Ω], shorted output
Output impedance [Ω], open input
Output impedance [Ω], shorted input
Transimpedance [Ω] Vout/Iin, Iout=0
Transconductance [s] Iout/Vin, Vout=0

The following *twoport functions* are available:

all parameters of type a,b,z,y,h,k
determinants of all twoport matrices
input and output wave impedances

In addition, students can compute the absolute and relative sensitivity of all circuit functions to arbitrary element parameter. All types of analyses are available in the symbolic, semisymbolic, and graphical representations.

5 Analysis example: Simple OpAmp circuit

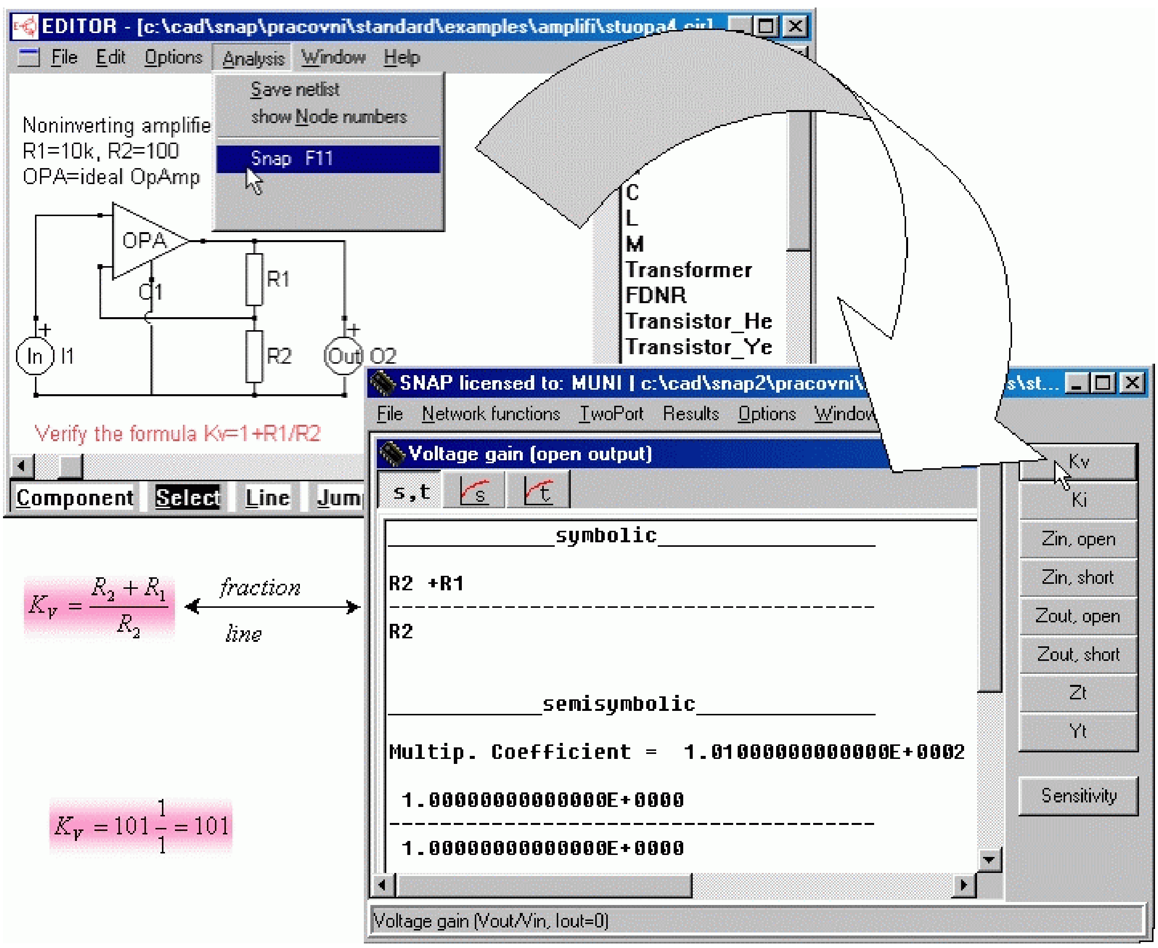
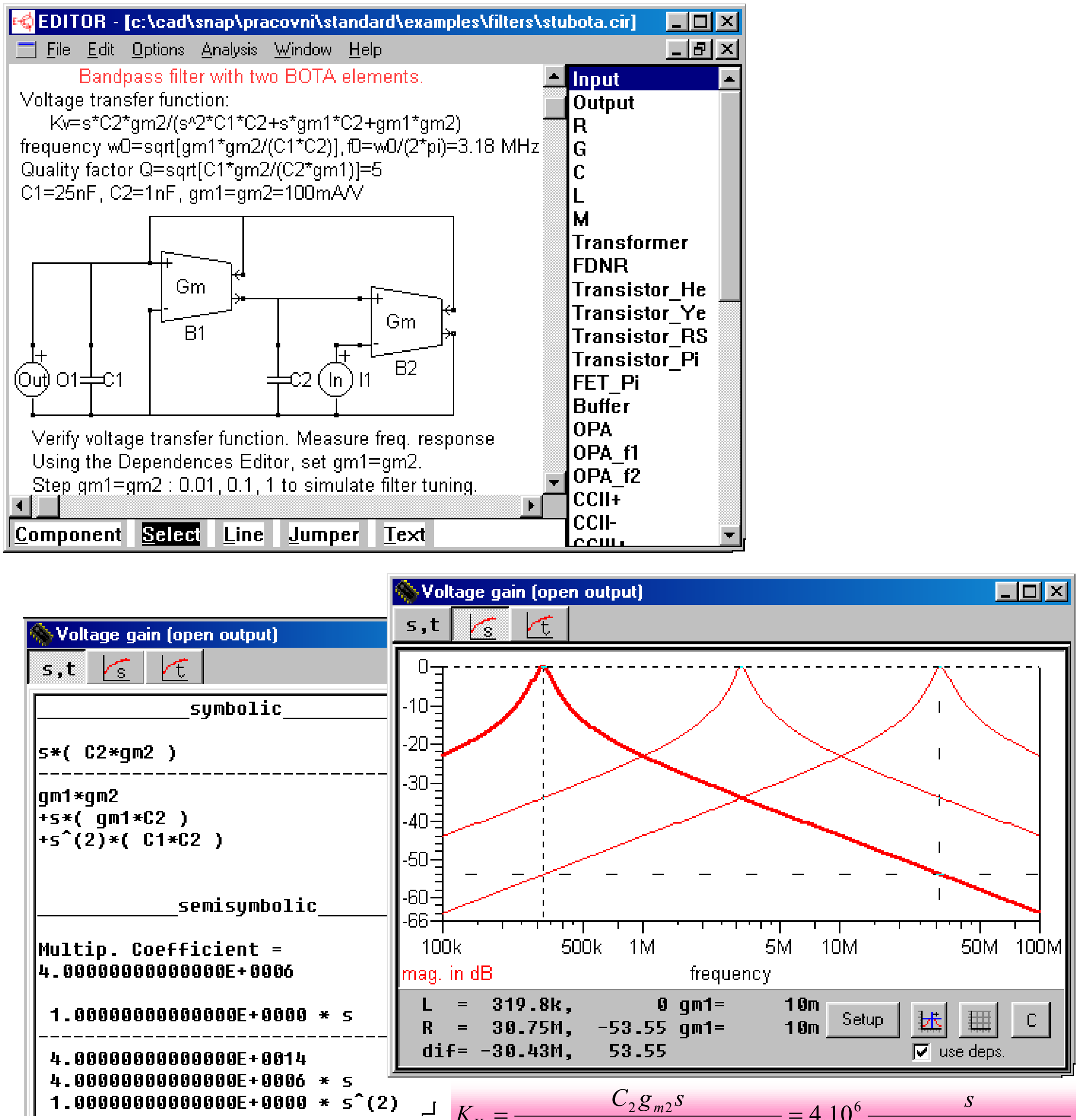


Fig. 5. From the schematic editor to SNAP analyzer: symbolic and semisymbolic results and their interpretation.

6 Analysis example: Bandpass filter with BOTA elements



$$K_V = \frac{C_2 g_{m2} s}{g_{m1} g_{m2} + g_{m1} C_2 s + C_1 C_2 s^2} = 4.10^6 \frac{s}{4.10^{14} + 4.10^6 s + s^2}$$

Fig. 6. Bandpass filter: Symbolic and semisymbolic analyses offer the s-domain coefficients of transfer function (and equations of impulse and step responses, which are not seen here). Numerical analysis yields various graphs in the frequency and time domains. Here the frequency responses are plotted for a set of transconductances to simulate the tuning ability.

7 Conclusion

In this paper, the basic features of SNAP program are introduced. This program combines the facilities that are important both for students and teachers: the simplicity, the universality of result forms, and the unlimited extensibility of the types of analyzed components.

References:

[1] BIOLEK,D.: Snap v. 2.6. Getting started. 40 p., *ATMOCOM*, Brno 1999, Czech Republic.

[2] BIOLEK,D.-SVIEZENY,B.: SNAP – analyse symbolique des circuits linéaires. Manuel utilisateur. S.E.E.R. (*Société d' Etudes d' Exploitation et de Recherches*), Paris, France, 1999, 63p.

[3] BIOLEK,D.-KOLKA,Z.: SNAP Program Extension for Supporting User-Based Analysis and Optimisation of Analog Filters. *AMTEE'99 Plzen*, 1999, pp. G01-G04. ISBN 80-7082-541-3

[4] BIOLEK,D.-KOLKA,Z.: SNAP program v. 2.6: New Opportunities for teaching and research. Proceedings of the Conference on Circuit Theory STO-7, „*Modern Trends in Teaching Electrical Engineering and Electronics*", VA Brno, 1999, pp. 66-69. Czech Republic. ISBN 80-214-1392-1.

[5] http://www.fee.vutbr.cz/UREL/snap