Short program computes response of RLC networks

by Werner A. Schnider
Swiss Federal Institute of Technology, Zurich

When designing filters, amplifiers, and other circuits, the frequency response can be determined with a Fortran program that is very short and requires much less computation time than many common circuit-design programs such as ECAP-1. By introducing current-controlled current sources, the program can analyze transistor circuits directly, while with a little modification it works as well with circuits containing operational amplifiers. The program stores three matrixes—basically of resistance, capacitance, and reciprocal inductance—during the whole frequency-response computation, and from them builds the node-admittance matrix for each frequency.

The computational procedure for each frequency requires the set-up of a node-admittance matrix, [H], and the computation of two determinants, $|H_{io}|$ and $|H_{ii}|$ and the gain V_o/V_i. The gain is the quotient of the two determinants—a consequence of the equivalent current vector having only one non-zero element, corresponding to the input node. In the abstract, each of the two voltages that determine gain is the matrix product of the current vector and the inverse of the node-admittance matrix. Computing that inverse usually involves a great many determinants. But when the current vector has only one non-zero element, the matrix product becomes a scalar product, and, because the gain is a quotient of two such products, everything cancels out

INPUT DATA CARDS

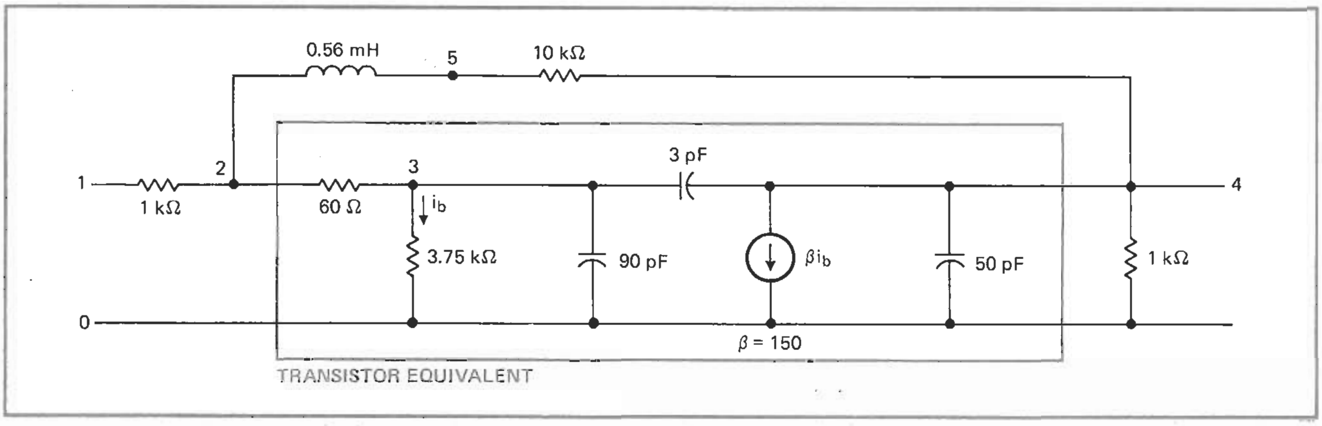
```
5,
AR,1,2,1.E3
AR,2,3,60.,
AR,3,0,3.75E3,
AC,3,0,90.E-12,
AC,3,4,3.E-12,
AR,4,0,1.E3,
AC,4,0,50.E-12,
AR,4,5,10.E3,
AL,5,2,0.56E-3,
AB,3,6,150.,
A ,,,,
1,4,
2.E5,7.E6,35.
```

PROGRAM

```
      PROGRAM AC(INPUT,OUTPUT)
      DIMENSION DC(20,20),BC(20,20),GM(30),NKA(30),NKB(30),EC(20,20)
      COMPLEX CO,CE,Q(20,20)
      DIMENSION A(100),P(100)
      INTEGER R,C,B,BLANK,TYPE
      DATA R,C,B,BLANK,LQ/1HR,1HC,1HB,1H ,1HL/
C INPUT-ROUTINE.
      PRINT 7
7     FORMAT(1H1,10(1H*),20H FREQUENCY RESPONSE ,10(1H*),/)
      DO 8 I=1,20
      DO 8 J=1,20
8     DC(I,J)=BC(I,J)=EC(I,J)=0.
      NZ=0
      CALL NREAD(NK)
2     CALL NREAD(TYPE,K1,K2,WERT)
      IF(TYPE.EQ.BLANK) GOTO 6
      PRINT 10,TYPE,K1,K2,WERT
10    FORMAT(1H ,A1,2(I4),4X,E12.4)
      IF(TYPE.NE.B) GOTO 3
      WERT=WERT*GM(K1)
      NA=NKA(K1)
      NB=NKB(K1)
      NBB=NKB(K2)
      NAA=NKA(K2)
      IF(NA.NE.0.AND.NAA.NE.0) BC(NA,NAA)=BC(NA,NAA)+WERT
      IF(NA.NE.0.AND.NBB.NE.0) BC(NA,NBB)=BC(NA,NBB)-WERT
      IF(NB.NE.0.AND.NBB.NE.0) BC(NB,NBB)=BC(NB,NBB)+WERT
      IF(NB.NE.0.AND.NAA.NE.0) BC(NB,NAA)=BC(NB,NAA)-WERT
      GOTO 2
3     CONTINUE
      NZ=NZ+1
      NKA(NZ)=K1
      NKB(NZ)=K2
      IF(TYPE.EQ.R) GM(NZ)=1./WERT
      IF(TYPE.EQ.R) CALL MATR(K1,K2,BC,GM(NZ))
      IF(TYPE.EQ.C) CALL MATR(K1,K2,DC,WERT)
      IF(TYPE.EQ.LQ) H=1./WERT
      IF(TYPE.EQ.LQ) CALL MATR(K1,K2,EC,H)
      GOTO 2
6     CALL NREAD(KA,KB)
      CALL NREAD(F1,F2,N)
      PRINT 20,KA,KB,F1,F2,N
20    FORMAT(/,11H I/O-PORTS:,2I4,/,1H ,8HF1-F2-N=,2E14.4,I8)
      PRINT 11,NZ,NK
11    FORMAT(/ ,1H ,I4,9H BRANCHES,10X,I4,6H NODES,//)
C COMPUTATION OF FREQUENCY RESPONSE.
      DEL=(F2-F1)/(N-1)
      DO 50 I=1,N
      FREQ=F1+(I-1)*DEL
      OMEGA=6.2831852*FREQ
      DO 51 K=1,NK
      DO 51 L=1,NK
51    Q(K,L)=CMPLX(BC(K,L),OMEGA*DC(K,L)-EC(K,L)/OMEGA)
      CALL CCDET(Q,KA,KA,NK,CO)
      CALL CCDET(Q,KB,KA,NK,CE)
      CO=CE/CO
      AMPL=CABS(CO)
      IF(REAL(CO).EQ.0.) CO=CO+1.E-16
      PHASE=ATAN2(AIMAG(CO),REAL(CO))
      PHASE=PHASE*180./3.1415926
      PRINT 60,FREQ,AMPL,PHASE
60    FORMAT(6H FREQ=,E10.3,5X,5HAMPL=,E12.4,5X,6HPHASE=,E12.4)
      A(I)=AMPL
      P(I)=PHASE
50    CONTINUE
      CALL FPLOT(A,P,N,F1,DEL)
      END


      SUBROUTINE MATR(K1,K2,DC,WERT)
      DIMENSION DC(20,20)
      IF(K1.NE.0) DC(K1,K1)=DC(K1,K1)+WERT
      IF(K2.NE.0) DC(K2,K2)=DC(K2,K2)+WERT
      IF(K1.EQ.0.OR.K2.EQ.0) GOTO 9
      DC(K1,K2)=DC(K1,K2)-WERT
      DC(K2,K1)=DC(K2,K1)-WERT
9     RETURN
      END


      SUBROUTINE CCDET(Q,IZ,IS,NK,VALUE)
      COMPLEX Q(20,20),CA(20,20),VALUE
      NKM=NK-1
      DO 10 K=1,NK
      DO 10 L=1,NK
10    CA(K,L)=Q(K,L)
      IF(IZ.EQ.NK) GOTO 3
      DO 2 K=IZ,NKM
      DO 2 L=1,NK
2     CA(K,L)=Q(K+1,L)
3     IF(IS.EQ.NK) GOTO 5
      DO 4 K=1,NK
      DO 4 L=IS,NKM
4     CA(K,L)=CA(K,L+1)
5     CONTINUE
      CALL CDET(CA,NKM,VALUE)
      VALUE=VALUE*(-1.)**(IZ+IS)
      RETURN
      END
```



OUTPUT LISTING

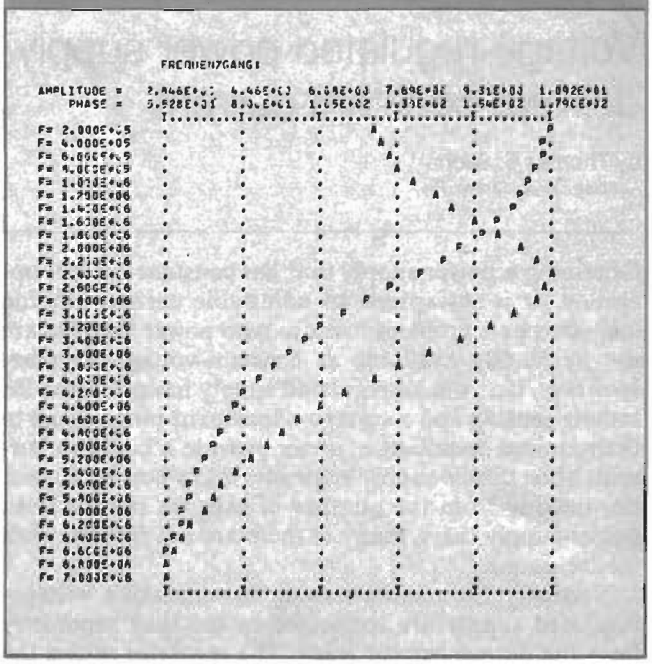
```
********** FREQUENCY RESPONSE **********

R   1   2     1.0000E+03
R   2   3     6.0000E+01
R   3   0     3.7500E+03
C   3   0     9.0000E-11
C   3   4     3.0000E-12
R   4   0     1.0000E+03
C   4   0     5.0000E-11
R   4   5     1.0000E+04
L   5   2     5.6000E-04
B   3   6     1.5000E+02
I/O-PORTS:   1   4
F1-F2-N=    2.0000E+05    7.0000E+06      35

   9 BRANCHES              5 NODES

FREQ= 2.000E+05    AMPL=  7.2734E+00    PHASE=  1.7903E+02
FREQ= 4.000E+05    AMPL=  7.3804E+00    PHASE=  1.7795E+02
FREQ= 6.000E+05    AMPL=  7.5593E+00    PHASE=  1.7664E+02
FREQ= 8.000E+05    AMPL=  7.8091E+00    PHASE=  1.7499E+02
FREQ= 1.000E+06    AMPL=  8.1295E+00    PHASE=  1.7287E+02
FREQ= 1.200E+06    AMPL=  8.5163E+00    PHASE=  1.7017E+02
FREQ= 1.400E+06    AMPL=  8.9594E+00    PHASE=  1.6677E+02
FREQ= 1.600E+06    AMPL=  9.4415E+00    PHASE=  1.6256E+02
FREQ= 1.800E+06    AMPL=  9.9313E+00    PHASE=  1.5748E+02
FREQ= 2.000E+06    AMPL=  1.0382E+01    PHASE=  1.5148E+02
FREQ= 2.200E+06    AMPL=  1.0732E+01    PHASE=  1.4464E+02
FREQ= 2.400E+06    AMPL=  1.0920E+01    PHASE=  1.3713E+02
FREQ= 2.600E+06    AMPL=  1.0898E+01    PHASE=  1.2925E+02
FREQ= 2.800E+06    AMPL=  1.0658E+01    PHASE=  1.2135E+02
FREQ= 3.000E+06    AMPL=  1.0231E+01    PHASE=  1.1378E+02
FREQ= 3.200E+06    AMPL=  9.6741E+00    PHASE=  1.0677E+02
FREQ= 3.400E+06    AMPL=  9.0501E+00    PHASE=  1.0047E+02
FREQ= 3.600E+06    AMPL=  8.4103E+00    PHASE=  9.4893E+01
FREQ= 3.800E+06    AMPL=  7.7896E+00    PHASE=  8.9997E+01
FREQ= 4.000E+06    AMPL=  7.2076E+00    PHASE=  8.5714E+01
FREQ= 4.200E+06    AMPL=  6.6730E+00    PHASE=  8.1960E+01
FREQ= 4.400E+06    AMPL=  6.1879E+00    PHASE=  7.8653E+01
FREQ= 4.600E+06    AMPL=  5.7503E+00    PHASE=  7.5721E+01
FREQ= 4.800E+06    AMPL=  5.3567E+00    PHASE=  7.3104E+01
FREQ= 5.000E+06    AMPL=  5.0026E+00    PHASE=  7.0749E+01
FREQ= 5.200E+06    AMPL=  4.6837E+00    PHASE=  6.8614E+01
FREQ= 5.400E+06    AMPL=  4.3958E+00    PHASE=  6.6666E+01
FREQ= 5.600E+06    AMPL=  4.1352E+00    PHASE=  6.4876E+01
FREQ= 5.800E+06    AMPL=  3.8986E+00    PHASE=  6.3221E+01
FREQ= 6.000E+06    AMPL=  3.6831E+00    PHASE=  6.1683E+01
FREQ= 6.200E+06    AMPL=  3.4863E+00    PHASE=  6.0245E+01
FREQ= 6.400E+06    AMPL=  3.3059E+00    PHASE=  5.8896E+01
FREQ= 6.600E+06    AMPL=  3.1441E+00    PHASE=  5.7624E+01
FREQ= 6.800E+06    AMPL=  2.9874E+00    PHASE=  5.6421E+01
FREQ= 7.000E+06    AMPL=  2.8462E+00    PHASE=  5.5280E+01
```

OUTPUT PLOT



except the two determinants mentioned.

So, to establish the node-admittance matrix, number all the nodes, beginning with zero for the common-input-output port. Place the data cards—either punched cards or an equivalent data-entry medium—at the beginning of the program. The program includes a subroutine, NREAD, that eliminates all requirements for specific formats. All data elements are separated by commas.

The first data card specifies the number of nodes minus 1. It is followed by cards listing the branches with their elements. The first two characters on each branch card are alphabetics that specify the type of the passive element in that branch: AR for a resistor, AC for a capacitor, AL for an inductor. These are followed by two integers that denote the beginning and ending nodes of this branch, and a floating-point number that gives the value of the element in ohms, farads, or henries. The program numbers the branches automatically in the sequence the cards are read.

The branch cards are followed by cards giving the current-controlled sources with their amplification factors. Each of these begins with the letters AB, which identify this type of card. Then two integers specify the number of the controlling branch and the number of the controlled branch, and a floating-point number denotes the amplification factor. This is followed by the end card, which has only the letter A, one blank, and a series of commas, to signify that all the branches and sources have been listed.

Next comes an input/output card, with two integers that denote the numbers of the input and output nodes. The last data card carries two floating-point numbers that specify the frequency range and an integer that denotes the number of frequency steps—up to 100—to be calculated within the specified range.

The program uses five subroutines, three of which are available in most scientific computer centers, so are not listed here. They are NREAD (format-free reading of data cards), CDET (to compute the determinant of a complex matrix), and FPLOT (which plots the amplitude and phase of V_o/V_i. FPLOT is optional and if it is used, two arrays called A and R are required for storing the data to be plotted.

Two other subroutines are MATR, for setting up the matrixes, and CCDET, which computes the determinant $|H_{kj}|$ from the matrix [H] and the specified row and column. The program can handle any circuit with up to 20 nodes and 30 branches.

Here is an example. In a compensated rf amplifier (see drawing) the transistor is replaced by a current-controlled current source and by passive elements. The new program computes the frequency response of this circuit in 35 steps between 200 kilohertz and 7 megahertz on the Control Data 6400/6500 in less than 1.4 seconds. This includes use of the FPLOT subroutine and computation of the plotting values, but not the actual plotting itself. The same circuit computed with a modified ECAP took 3.4 seconds. (The output listing and the plot, prepared on a 1,200-line/min printer, comprise 94 lines, which take another 5 seconds or so to produce.)

The data cards that describe the circuit in this example are listed in the table. □