How could our ancestors live without computers?!
The world they lived in must have been horrible! – or – WONDERFUL!

Simulation and Analysis of Power System Transients

If you are going to print this file
use the two- side printing option!
Save your money – protect the Nature.

My "whereabouts":

Dr Marek Michalik
D-20 room 419
tel.: +48 (0)71 320 2153
e-mail: michalik@pwr.wroc.pl

Wrocław, February, 2008

References

Basic reading

[1] Watson N., Arrillaga J., *Power systems electromagnetic transients simulation*. The Institution of Electrical Engineers, London 2003.(available in our library)

[2] Dommel H.W., *Digital computer solution of electromagnetic transients in single- and multiphase networks*. EEE Transactions on Power Apparatus and Systems. Vol. PAS-88, April 1969, s. 388-399(also available)

[3] Kizilcay M., *Review of solution methods in ATP-EMTP*. EEUG News, Feb-May 2001, p. 25-36 (available on request)

Useful Web pages:

1. http://www.rose.pwr.wroc.pl/
   a lot of useful material and interesting examples – in Polish

2. http://zas.ie.pwr.wroc.pl/dyd.htm. Outline of the lecture in English, Lab work topics, Sample problems to solve.

3. http://www.pes-psrc.org/Apublications.html
   EMTP Tutorial sponsored by the Power System Relaying Committee IEEE PES .

4. http://www.ipst.org/

5. http://www.mathworks.com/

The list is open - any interesting reader's finding will immediately be included

ATP – EMTP Program

Fundamentals

The EMTP (*Electromagnetic Transient Program)* is probably the most popular computer software package used for simulation analysis of power systems.
The program is meant for solving of such problems like:

- switching transients and overvoltages
- short term analysis of disturbances
- overcurrent calculations
- protection performance analysis
- power quality estimation
- control of electric drives,
- FACTS and many others.

Historical background

EMTP was developed in early 60-ties by a group of programmers directed by Prof. Herman Dommel at sponsorship of BPA (Bonneville Power Administration, Canada). The first version was based on Fortran source code and data format. Some of these features are used up till now. On the basis of EMTP experience many professional programs for similar applications were developed.

ATP-EMTP version

Alternative Transient Program is the version available in public domain and being continuously developed by regional groups of users on all continents. The ATP – EMTP is now used on all up-to-date computers and is provided with handy graphical interface ATPDraw.The ATP-EMTP program can be run under all nowadays-used operational systems, in particular:

- MS – DOS (version Salford ATP)
- MS – Windows (version Watcom ATP)
- Linux (version GNU – Mingw32)The ATP-EMTP program capacity:
- Number of nodes - 6000
- Number of branches - 10000
- Number of switches – 1200
- Number of sources – 900
- Number of non-linear elements – 2250
- Number of synchronous machines - 90

The basic ATP-EMTP program comprises of 3 files:

- TPBIGx.exe – basic module, x – identifies the compiler used and: x=G: GNU-Mingw32, x=S: Salford, x=W: Watcom; LISTSIZE.DAT – text file available to user and meant for setting of some parameters, for example – max. number of nodes.
- STARTUP – text file available to user for control of some parameters like: input data file extension, input data format, e.c.t...

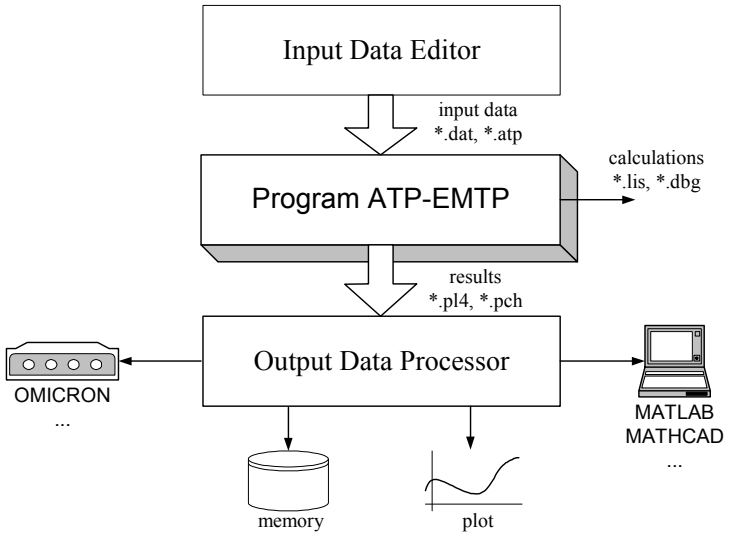


Fig.1. Functional diagram of ATP-EMTP package

The TPBIGx.exe has no embodied graphical interface so the use of external ATPDraw program is essentially useful. The basic module (Fig.2) also contains some auxiliary procedures meant for calculation of input file data for some more complex models.

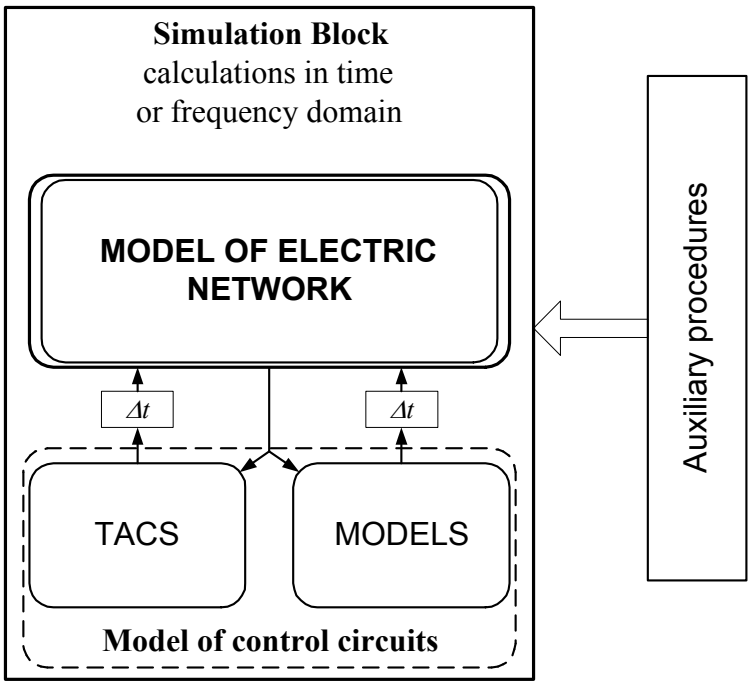


Fig.2 The functional structure of the basic program block.

Models of control circuits are grouped in 2 units (Fig.2):

- TACS – Transient Analysis of Control Circuits
- MODELS – Programmed modeling of control and other circuits.

Both units provide facilities for modeling of: elements described by transfer functions, counters, advanced voltage and time controlled switches, quantizers, non-linear elements, e.c.t.

MODELS are more versatile since the unique models can be implemented by use of symbolic programming language. The results of simulation are yielded in form of OUTPUT FILES:

- *.pl4 file that contains results as a function of time or frequency in form of text or binary file (by selection)
- *.pch file with auxiliary data
- *.lis, *.dbg text files with the record of program execution.

Auxiliary Procedures are dedicated supporting programs which enable calculation of parameters for Data Input File for power network elements of complex models. Most frequently used (out of many other available) are:

- LINE CONSTANTS – for calculation of power lineout of geometrical and material data for lumped and distributed parameter line models,
- CABLE CONSTANTS – like Line Constants but for cable model,
- CABLE PARAMETERS – extension of Cable Constants for distributed transverse parameter model
- JMARTI SETUP - calculates parameters for distributed frequency dependent line model,
- NETWORK EQUIVALENT – calculates equivalent parameters of the complex network fragment,
- XFORMER – procedure that, on basis of measurement data calculates the π-cell parameters representing 2 or 3-windingsingle phase transformer
- HISTERESIS – calculates the magnetizing characteristic with hysteresis for a given material
- ZNOFITTER – calculates analytical parameters of a varistors or a group of varistors basing on few point *v-i* characteristic
- DATA BASE MODULE – compilation of data for selected part of the network that can be included into the investigated model using the control instruction $INCLUDE.
- BCTRAN – calculation of linear multi-port circuit on the basis of measurement data for representation of a single or multi-phase multi-winding transformer

The simulation results can be processed and used in the following way:

- The results of auxiliary procedure (*.pch) can be applied to the network model by use of $INCLUDE in DATA BASE MODULE
- The frequency characteristic of the simulated network for specified variables can be obtained by use of FREQUENCY SCAN procedure
- The time characteristics for the selected variables are stored in the file *.pl4. Format of the file can be set in STARTUP file.
- The *.pl4 files can be plotted directly using PLOTXY or TOP programs. If the simulation results are to be used
- in MATLAB environment, the ATP-EMTP package offers special filters: PL42MAT, PL4TOMAT.

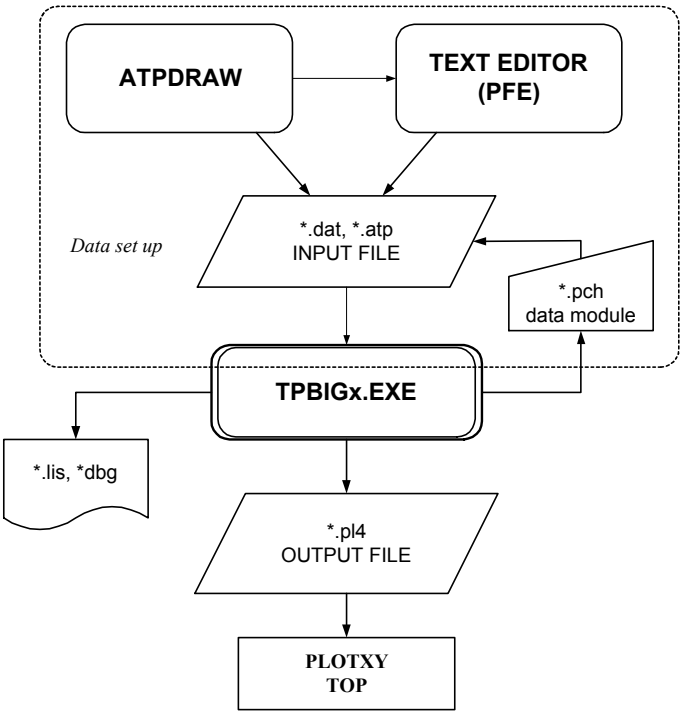


Fig.3 The input and output data of the ATP-EMTP package

The input data file set up by use of a text

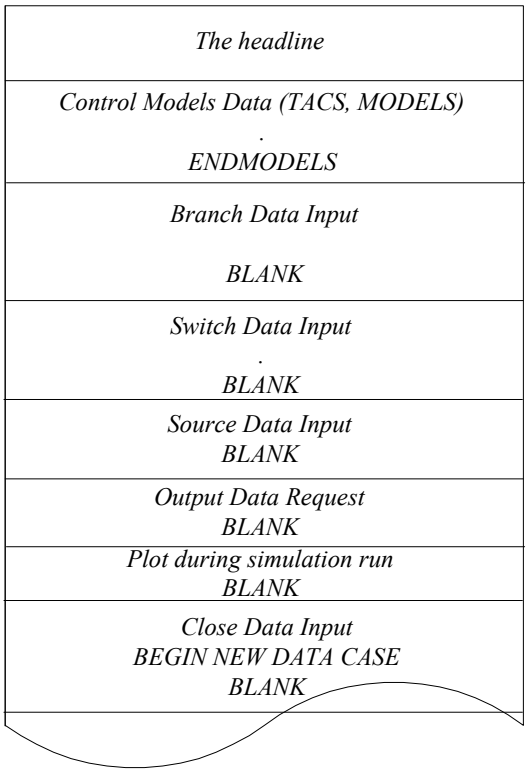


Fig.4 The basic structure of the user's input data file

The input data file headline

```
C DELTAT    TMAX    XOPT    COPT  EPSILN  TOLMAT  TSTART
C -----><------><------><------><------><------><------>
2.439E-5     .20    50.0    50.0
C   IOUT   IPLOT  IDOUBL  KSSOUT  MAXOUT    IPUN  MEMSAV    ICAT  NENERG  IPRSUP
C -----><------><------><------><------><------><------><------><------><------>
     200      41                                            1
```

The data entries must be in Fortran format, e.i.:

345.00=0.345E3=.345E3; 0.0234=.0234=.234E-1

The description of most important entry fields:

DELTAT – simulation step [s],

XOPT:

1. XOPT=0 – enter inductances L in [mH]
2. XOPT>0 – enter X_L in [Ω] and $X_L = 2\pi * XOPT * L$

TMAX – simulation time [s],

COPT:

1. COPT=0 – enter capacitance C in [μF]
2. COPT>0 – enter C as B [μS]; $B = 2\pi * COPT * C$

IOUT – determines frequency of simulation output display on the screen. IOUT=200 means that each 200th line will be displayed. To save the simulation duration this parameter should not be too large.

IPLOT – determines the number of the simulation results entries in the output file (.pl4).

- IPLOT=1 results in max. size output file (record in each calculation step T (DELTAT)
- IPLOT=101 means that the output is sampled at sampling period $T_{out} = 101T$

ICAT – controls the output file set up:

- 0 – no output file is made
- 1 – the output file record is enabled, on screen graphic display is off.
- 2 – as for 1 and on screen plotting during calculation is on

Branch Data Input

This is the program block which most of the network model parameters are entered for:

- lumped parameter models
- distributed parameter models
- distributed and frequency dependent parameter models
- non-linear time-variant models
- non-linear (pseudo-linear) models
- true non-linear models
- transformer models

A two-digit number in the first two columns of the line always indicates the type of model which the data are input for.

RLC branch data input for linear lumped parameter models

```
C   NOD1  NOD2 NOD01 NOD02      R      L      C                                  0
C <    ><    ><    ><    ><     ><     ><     >                                  <
  LIN1  LIN2                0.25   3.75   29.                                   1
```

The first 2 columns are empty, the next 24 ones are reserved for the node names on the branch.
Then the R L C parameters values are input into the next 3, six columns each, entry fields (in NORMAL PRECISION format). The last (80^{th}) column entry indicates the electric requested to be stored in the output file according to the code:

- 1 – current in the branch
- 2 – voltage across the branch
- 3 – 1and 2
- 4 – power and energy dissipated in the branch

Data for *RLC* element represented by a π-cell

The data have the form of matrixes that correspond to the circuit diagram shown in Fig.5:

$$\mathbf{R}=\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{12} & R_{22} & R_{23} \\ R_{13} & R_{23} & R_{33} \end{bmatrix} \quad \mathbf{L}=\begin{bmatrix} L_{11} & L_{12} & L_{13} \\ L_{12} & L_{22} & L_{23} \\ L_{13} & L_{23} & L_{33} \end{bmatrix} \quad \mathbf{C}=\begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{12} & C_{22} & C_{23} \\ C_{13} & C_{23} & C_{33} \end{bmatrix}$$

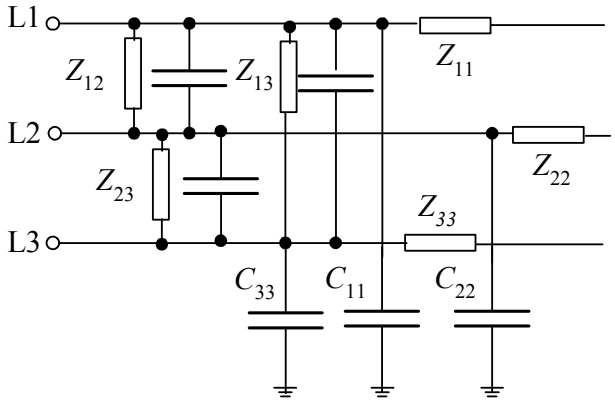


Fig.5. *RLC* element in for of a π-cell; Z_{jk} – series connection of R_{jk} and L_{jk}

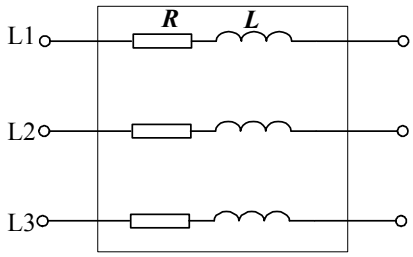
Data format layout for this type of element is as below:

```
C   NODA  NODAA              R11    L11    C11
C <    ><    >           <     ><     ><     >
 1LIN1A LIN2A              1.42   8.37   .158
C NODB  NODB                R12    L12    C12    R22    L22    C22
 2LIN1B LIN2B              0.968  .837 -.078   1.38   .142   .165
C NODC  NODC                R13    L13    C13    R23    L23    C23    R33    L33    C33
 3LIN1C LIN2C              0.955  .842 -.079   .968   .837 -.078   1.38   .142  .165
```

Since the matrixes are symmetrical the data for upper triangular sub matrix are entered only. In the similar way the multi-phase element data can be input into the data file. Maximum size of the data matrix is 40X40.

Coupled *RL* elements (Fig.6)

The elements are very frequently used to model overhead transmission lines, in particular, the transposed ones. In this model the transverse parameters (C) are neglected. The data matrix for the transposed line contains 2 parameters only, as it is shown below:



Rys.6. Coupled *RL* element

$$Z = \begin{bmatrix} Z_s & Z_m & Z_m \\ Z_m & Z_s & Z_m \\ Z_m & Z_m & Z_m \end{bmatrix} \Leftrightarrow \begin{bmatrix} Z_0 & 0 & 0 \\ 0 & Z_1 & 0 \\ 0 & 0 & Z_2 \end{bmatrix},$$

For symmetrical components the matrix takes the diagonal form for which:

$$Z_0 = Z_s + 2Z_m,\ Z_1 = Z_2 = Z_s - Z_m,$$

$$Z_s = \frac{Z_0 - Z_1}{3},\ Z_m = \frac{Z_0 + 2Z_1}{3}.$$

The assigned data format for this element is as shown below:

```
C   WEZ1  WEZ2 WEZ01 WEZ02     R     L
C <     ><    ><    ><    ><    ><    >
C   WEZ1  WEZ2 WEZ01 WEZ02    R0    L0
51GEN1A GEN2A               12.5  3.85
C   WEZ1  WEZ2 WEZ01 WEZ02    R1    L1
52GEN1B GEN2B                2.5  1.75
53GEN1C GEN2C
```

Switch Data Format

The simplest example of a switch is the time –controlled unit. The input data format for this switch is as follows:

```
C <BUS1><BUS2><T_CLOSE >< T_OPEN ><C_LIM   ><V_FLESH ><SPECIAL >
    WEZ1  WEZ2      -1.0      0.02      1.E5
```

BUS1, BUS2 – branch node names.
T_CLOSE – switch on time; negative value implies the closed state for steady state calculations. T_OPEN – the switch off time [s],condition: T_CLOSE<T_OPEN;
C_LIM – maximum value of current flowing through the switch below which it can be opened if t>T_OPEN.V_FLESH – minimum value of voltage across the switch at which it can be closed if t>T_CLOSE.
SPECIAL – if the word MEASUREMENT is entered the switch remains permanently closed and the current data are recorded in the output file.

Source Data Format

ATP program offers many sources of different types. The most popular one used in power system transient analysis is the source no.14. The source is the ideal (no source resistance) generator of cosine wave of pre-set parameters. The relevant data format is as shown below:

```
C NODE->i?<-Amplit-><--Freq--><-Phase -><---A1---><---T1---><-Tstart-><-Tstop-->
  GENER 1       100.       50.      -45.
```

Where:

NODE– node name

i? – if 0 current source option, 1- voltage source

Amplit – the magnitude of the cosine wave

Freq – fundamental frequency in[Hz]

Phase – initial phase displacement in deg(A1=0) or in [s] (A1>0)

Tstart, Tstop – beginning and the end of the source activation [s]

Discrete models of power networks

The simulation of power networks is aimed at detailed analysis of many problems and the most important of them are:

- determination of power and currents flow in normal operating conditions of the network,
- examination of system stability in normal and abnormal operating conditions,
- determination of transients during disturbances that may occur in the network,
- determination of frequency characteristics in selected nodes of the network.

The network model is derived from differential equations that relate currents and voltages in network nodes according to Kirchoffs law.
The simulation models are usually based upon equivalent network diagrams derived under simplified assumptions (which sometimes can be significant) that are applied to the network elements representation. In this respect models can be divided into two basic groups:

- Lumped parameter models
  3D properties of elements are neglected and sophisticated electromagnetic relations that include space geometry of the network are not taken into account.

- Distributed parameter models
  Some geometrical parameters are used in the model describing equations (usually the line length).

In classic theory relations between currents and voltages on the network elements are continuous functions of time. In digital simulations the numerical approach must be applied. Two ways are applied for this purpose:

- Transformation of continuous differential relations into discrete (difference) ones,
- State variable representation in continuous domain and its solution by use of numerical methods.

Consequences of transformation from continuous to discrete domain:

- Problem of accuracy - discrete representations are always certain (better or worse) approximation of continuous reality,
- Frequency characteristics become periodic according to Shannon's theorem,
- Problem of numerical stability - numerical instability may appear even though the continuous representation of the network is absolutely stable.

Numerical solution of differential equations - basic algorithms

In electric networks with lumped parameters the basic differential equation that describes dynamic relation between physical quantities observed in branches with linear elements (*R, L, C*) takes the form:

$$\frac{dy(t)}{dt} + \lambda y(t) = bw(t) \tag{1.1}$$

Where *y(t), w(t)* denotes electric quantities (current, voltage) and l, *b* are the network parameters. In case of a single network component (inductor, capacitor) eqn. (1.1) simplifies into:

$$\frac{dy(t)}{dt} = bw(t) \tag{1.2}$$

Laplace transformation of (2) results in:

$$sY(s) = bW(s) \tag{1.3}$$

To obtain discrete representation of (1.2) the continuous operator in s-domain must be replaced by the discrete operator z in z-domain ('shifting operator'). The basic and accurate relation between those two domain is given by the fundamental formula:

$$z = e^{sT} \tag{1.4}$$

T - sampling period

Approximate rational relations between *z* and *s* can be obtained from expansions of (1.4) into power series. Let's consider the following most obvious cases:

1.
$$z = e^{Ts} = 1 + Ts + \frac{(Ts)^2}{2!} + \ldots + \frac{(Ts)^n}{n!} + \ldots\ldots$$

Neglecting terms of powers higher than 1 results in approximation:

$$z \cong 1 + Ts$$

or

$$s \cong \frac{z-1}{T} \tag{1.5}$$

2.

$$z = e^{Ts} \approx 1 + Ts + (Ts)^2 + ... + (Ts)^n + ... = \frac{1}{1-Ts}$$

or

$$z \cong \frac{1}{1-Ts}$$

Again, higher power terms have been neglected so:

$$s \cong \frac{z-1}{Tz} \tag{1.6}$$

3.

$$z = e^{Ts}$$

$$s = \frac{1}{T}\ln z = \frac{2}{T}\left[\frac{z-1}{z+1} + \frac{(z-1)^3}{3(z+1)^3} + ...\right]$$

and

$$s \cong \frac{2(z-1)}{T(z+1)} \tag{1.7}$$

The approximation (1.7) is the well known Bilinear Transformation or Tustin's operator. Applying the derived approximations of *s* to differential equation (1.3) three different discrete algorithms for calculation of *w*(k) integral can be obtained.

1a.

$$\frac{z-1}{T} Y(z) = bW(z)$$

In discrete domain:

$$\frac{y(k+1) - y(k)}{T} = bw(k)$$

So the applied approximation is the *Euler's forward approximation* of a continuous derivative. The resulting integration algorithm takes the form:

$$\begin{aligned} Y(z) &= z^{-1}Y(z) + z^{-1}bTW(z) \\ y(k) &= y(k-1) + bTw(k-1) \end{aligned} \tag{1.8}$$

The algorithm (1.8) realizes iteration that within a single step T can be written as:

$$y(t_k) = y(t_{k-1}) + bT \int_{t_{k-1}}^{t_k} w(\tau)d\tau$$

The algorithm is of ***explicit*** type since the current output in k-th instant depends only on past values of the input and output in (k-1) instant.

2a.

Using the second approximation of z (1.6):

$$\frac{z-1}{zT}Y(z) = bW(z)$$

In discrete time domain:

$$\frac{y(k)-y(k-1)}{T} = bw(k)$$

So the applied approximation is the *Euler's backward approximation* of a continuous derivative. The resulting integration algorithm takes the form:

$$\begin{aligned} Y(z) &= z^{-1}Y(z) + bTW(z) \\ y(k) &= y(k-1) + bTw(k) \end{aligned} \tag{1.9}$$

This algorithm is of ***implicit*** type since the current output in k-th instant depends on present value of the input in the same instant.

The algorithm (1.9) realizes integration which, within a single step T, can be written as:

$$y(t_k) = y(t_{k-1}) + bT \int_{t_k}^{t_{k+1}} w(\tau)d\tau$$

3a.

Using the third approximation of z (1.7):

$$\begin{aligned} &\frac{2(z-1)}{T(z+1)}Y(z) = bW(z) \\ &Y(z) = z^{-1}Y(z) + \frac{Tb\left[W(z) + z^{-1}W(z)\right]}{2} \\ &y(k) = y(k-1) + \frac{Tb\left[w(k) + w(k-1)\right]}{2} \end{aligned} \tag{1.10}$$

This algorithm (1.10) realizes numerical integration based upon trapezoidal approximation of the input function w(k).

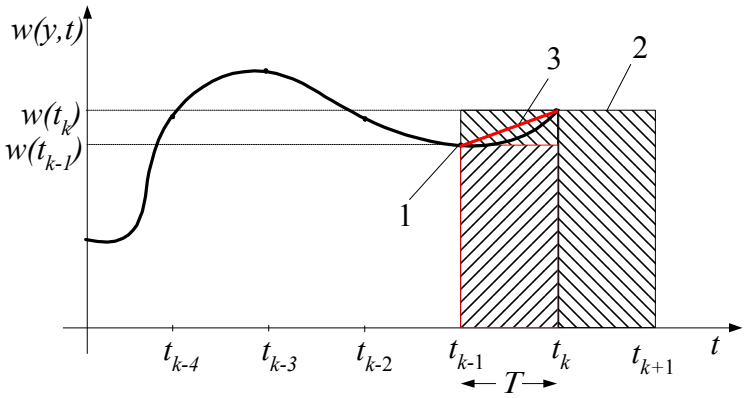


Fig.1.1. Numerical integration; 1 - Euler's 'step back' (explicit) approximation.; 2 - Euler's 'step forward' (implicit) approx.; 3 - trapezoidal approximation

Examination of Fig.1.1 leads to the following conclusions:

- Forward approximation of derivative results in 'step backward' (explicit) integrating algorithm and vice versa.
- The explicit algorithm tends to underestimate while the implicit one overestimates the integration result.
- The algorithm based on trapezoidal approx. reduces the integration error since its output y_{TR}(k) (1.10) is an average of outputs of both aforementioned algorithms y_E(k) (1.8), y_I(k) (1.10) at any instant k, e.i.

$$y_{TR}(k) = \frac{y_E(k) + y_I(k)}{2}$$

In general, the numerical integration methods depend on approximations of continuous derivative (or integral) and can be divided into two groups, namely:

- single step integration methods (self-starting)
- multi-step methods.

All algorithms considered belong to the first group. As an example of a multi-step numerical integrator the 2-nd order Gear algorithm can be shown:

$$y(k) = \frac{4y(k-1) - y(k-2) + Tbw(k)}{3} \tag{1.11}$$

The algorithm is not self-starting one and must be started by use of a single step algorithm; reveals stiff stability properties.

Accuracy of operation

Accuracy of numerical integration for the algorithms considered can be estimated from homogenous from of the eqn.(1.1), e.i:

$$\frac{dy(t)}{dt} - \lambda y(t) = 0 \tag{1.12}$$

which yields the accurate solution:

$$y(t) = y(t_0)e^{-\lambda t}$$

$y(t_0)$ – initial condition at t_0 ; $\lambda > 0$

Applying *s* approximations (1.5, 1.6, 1.7) to (12) the following numerical expressions are obtained:

$$y(k) = (1-\lambda T)y(k-1) \qquad \text{Explicit Euler's method ('step backward')} \qquad (1.13)$$

$$y(k) = \frac{y(k-1)}{1+\lambda T} \qquad \text{Implicit Euler's method ('step forward')} \qquad (1.14)$$

$$y(k) = \frac{2-\lambda T}{2+\lambda T}y(k-1) \qquad \text{Trapezoidal approximation} \qquad (1.14)$$

Accurate result of integration at the instant $t_k=kT$ is:

$$y_{aL}(k) = y(k-1)e^{-\lambda T}$$

Thus the local integration error for one interval $T=t_k - t_{k-1}$ can be defined as:

$$\Delta_L = y_{aL}(k) - y(k) \qquad (1.15)$$

This local error can easily be determined in detail for each algorithm considered. Let's take for example the method (1.13):

$$\Delta_L = y(k-1)e^{-\lambda T} - (1-\lambda T)y(k-1) = y(k-1)(e^{-\lambda T} - 1 + \lambda T)$$

Expansion of the exponential term into power series yields:

$$\Delta_L = y(k-1)(\frac{(\lambda T)^2}{2} - \frac{(\lambda T)^3}{3!} + ...)$$

Putting the constraint $\lambda T<1$ and using some mathematics the local error can be estimated by the approximate formula:

$$\Delta_L = \left| \frac{(\lambda T)^{p+1}}{(2)^{p-1}(2+\lambda T)} \right| \qquad (1.16)$$

p – the order of the algorithm(in this case *p*=1)

The global error ΔG is defined as the difference between accurate and approximate integration result in a longer time span e.i. from the the first step (k=1) to the arbitrary step k, so:

$$\Delta_G = y_0 e^{-k\lambda T} - y(k) \tag{1.17}$$

The respective integration results for the algorithms considered are (order of presentation as in previous case):

$$y(k) = (1-\lambda T)^k y_0 \qquad \text{step backward} \tag{1.18}$$

$$y(k) = \frac{y_0}{(1+\lambda T)^k} \qquad \text{step forward} \tag{1.19}$$

$$y(k) = \left[\frac{2-\lambda T}{2+\lambda T}\right]^k y_0 \qquad \text{trapezoidal approximation} \tag{1.20}$$

Discussion of eqns.(1.18, 1.19, 1.20)

- Algorithm (1.18)
  The integration method is convergent and the algorithm remains stable if

$$|1-\lambda T| < 1$$

- That implies:

$$T < \frac{2}{\lambda} \tag{1.21}$$

- Algorithms (1.19, 1.20) remain stable regardless of T
- if the algorithm is stable the global error tends to zero even though the local error may attain significant values

Illustration of the errors discussed is shown in Fig.1.2. The plots presented have been calculated for: y_0=10; l=2; T=0.987.

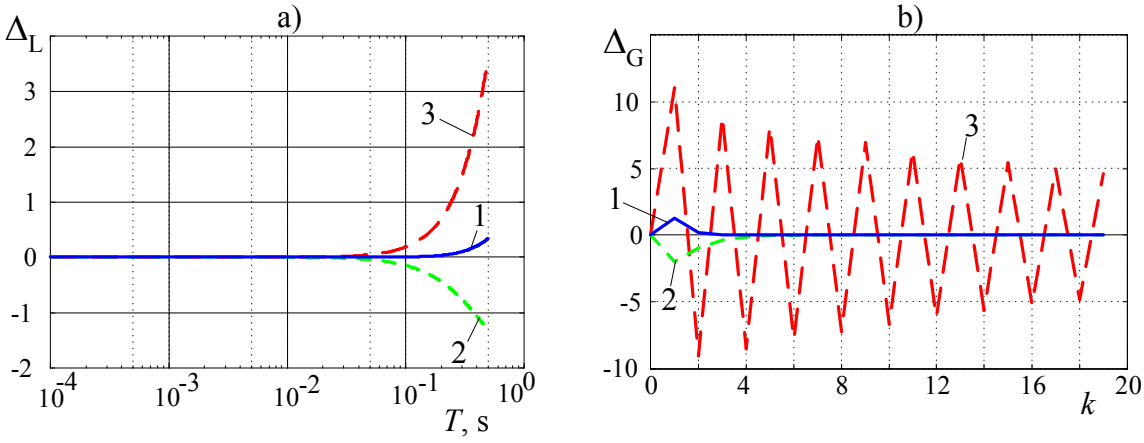


Fig. 1. 2. Local Δ_L and global Δ_G error values for the algorithms considered:

1 – trapezoidal approx.; 2 – Euler's 'step forward' ; 3 – Euler's 'step back'

Resistance

As the resistive elements do not have the energy storing capacity the discrete relation between current and voltage drop across resistance *R* can be obtained directly from the continuous relation and:

$$i(k) = \frac{1}{R}u(k) = Gu(k) \tag{2.1}$$

Inductance

The energy stored in magnetic field produced by current has the impact on voltage across the element so its continuous model is described by the equation:

$$\frac{di(t)}{dt} = \frac{1}{L}u(t) \tag{2.1a}$$

Using the transformation (1.6) or eqn.(1.9) the Euler's implicit discrete model of the element is obtained:

$$i(k) = i(k-1) + \frac{T}{L}u(k) = i(k-1) + Gu(k), \quad G = \frac{T}{L} \tag{2.2}$$

Note that *T/L* has the conductance unit.

For the trapezoidal transformation (1.7) or eqn.(1.10) the discrete model takes the form:

$$\begin{aligned} &i(k) = i(k-1) + \frac{T}{2L}\left[u(k-1) + u(k)\right] \\ &or \\ &i(k) = Gu(k) + i(k-1) + Gu(k-1), \quad G = \frac{T}{2L} \end{aligned} \tag{2.3}$$

The eqn.(2.3) can be rearranged in the following way:

$$\begin{aligned} &i(k) = Gu(k) + i(k-1) + Gu(k-1) \\ &or \\ &i(k) = Gu(k) + j(k-1) \\ &where \\ &\qquad j(k-1) = i(k-1) + Gu(k-1) \end{aligned} \tag{2.4}$$

Since the calculations in step *k* employ the values calculated in step *k*-1 which are constant and can be considered as the constant current sources *j*(*k*-1). Thus the inductance can be represented by equivalent companion model corresponding to (2.4) which is shown in Fig.2.1.

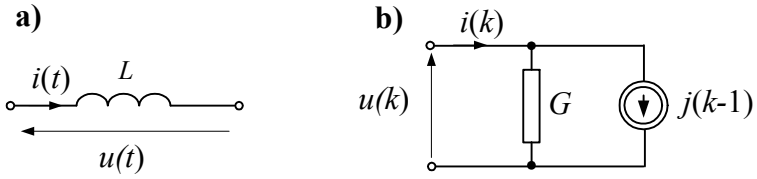


Fig. 2.1. Discrete model of inductance; a) symbol; b) companion discrete model

Capacitance

This element also reveals the energy storing capacity in form of electric charge so the relation between voltage and current in the element is given by the formula:

$$\frac{du(t)}{dt} = \frac{1}{C} i(t)$$

Using the same transformations as for the inductance the discrete models of capacitance can be derived:

$$u(k) = u(k-1) + \frac{T}{C} i(k) \tag{2.5}$$

Introducing the conductance notation (2.5) takes the form:

$$i(k) = Gu(k) - Gu(k-1), \quad G = \frac{C}{T}$$

and

$$i(k) = Gu(k) + j(k-1), \qquad j(k-1) = -Gu(k-1) \tag{2.6}$$

Using the trapezoidal integration method the discrete model of capacitance takes the similar form:

$$u(k) = u(k-1) + \frac{T}{2C}\left[i(k-1) + i(k)\right] \tag{2.7}$$

The companion discrete model for capacitance can be derived as:

$$i(k) = Gu(k) + j(k-1)$$

in which:

$$j(k-1) = -i(k-1) + Gu(k-1), \quad G = \frac{2C}{T} \tag{2.8}$$

The respective representation is shown in Fig.2.2:

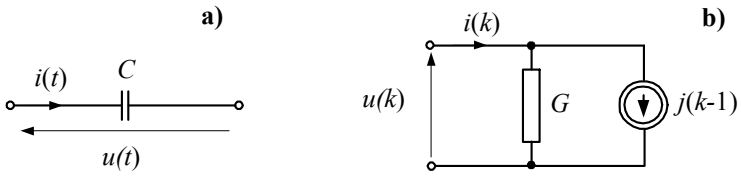


Fig. 2.2. Discrete model of capacitance; a) symbol; b) companion discrete model

In the very similar way the parameters of companion circuits can be derived for any integration method used. In Table 1.1 the example of those parameters for three selected methods are shown:

Table 1.1 Companion circuits parameters for selected numerical integration methods.

Integration method	Model of inductance L	Model of capacitance C
Euler's implicit method	$j(k-1)=i(k-1)\,,\ G=\frac{T}{L}$	$j(k-1)=-Gu(k-1)\,,\ G=\frac{C}{T}$
Trapezoidal approximation	$j(k-1)=i(k-1)+Gu(k-1)\,,$ $G=\frac{T}{2L}$	$j(k-1)=-\left(i(k-1)+Gu(k-1)\right),$ $G=\frac{2C}{T}$
Gear's 2nd order	$j(k-1)=\frac{1}{3}\left(4i(k-1)-i(k-2)\right),$ $G=\frac{2T}{3L}$	$j(k-1)=-G\left(2u(k-1)+\frac{1}{3}u(k-2)\right),$ $G=\frac{3C}{2T}$
Basic numerical algorithm: $i(k)=Gu(k)+j(k-1)$		

Series *R L C* branch

The equivalent discrete model of in series connected *R L C* branch can be obtained by series connection of basic models of each particular element in the branch as it is shown in Fig.7b.

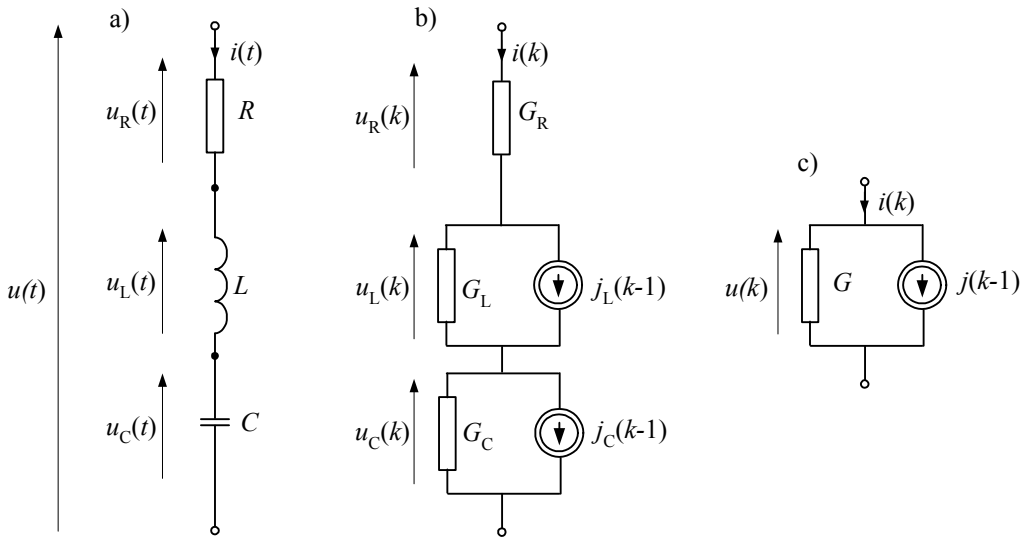


Fig.7. Discrete model of *RLC* branch; a0 the continuous model; b) discrete models of particular elements; b) the equivalent companion model of the branch.

To derive the equivalent companion circuit model (Fig.7c) consider the basic equation for voltage across the branch (Fig.7b):

$$u(k)=u_R(k)+u_L(k)+u_C(k) \tag{2.9}$$

in which the particular terms can be expressed by their basic models:

$$u_R(k)=\frac{1}{G_R}i(k),$$
$$u_L(k)=\frac{1}{G_L}\big(i(k)-j_L(k-1)\big),\ u_C(k)=\frac{1}{G_C}\big(i(k)-j_C(k-1)\big). \tag{2.10}$$

After substitution and appropriate rearrangement of (2.9) the equivalent model equation is obtained:

$$i(k)=Gu(k)+j(k-1) \tag{2.11}$$

in which, for trapezoidal approximation,

$$G=\frac{G_R G_L G_C}{G_R G_L+G_R G_C+G_L G_C}=\frac{2CT}{4LC+2RCT+T^2}$$

$$j(k-1)=\frac{G_R G_C j_L(k-1)+G_R G_L j_C(k-1)}{G_R G_L+G_R G_C+G_L G_C}=\frac{G}{G_L}j_L(k-1)+\frac{G}{G_C}j_C(k-1),$$

and

$$G_R=\frac{1}{R},\ G_L=\frac{T}{2L},\ G_C=\frac{2C}{T}.$$

If capacitance C is not present in a branch then $C\to\infty$ must be put into the above equations. For missing R or L, $R = 0$ or $L = 0$ must be used. For example, the $R\,L$ branch is the respective relations are:

$$G=\frac{T}{2L+RT}\qquad j(k-1)=\frac{2L}{2L+RT}j_L(k-1)=\frac{1-RG_L}{1+RG_L}i(k-1)+Gu(k-1) \tag{2.12}$$

Controlled sources

Controlled sources are used very often in electronic and electric network models. Generally there are four basic types of such sources (Fig.8):

- Voltage controlled current sources $j=ku_x$ controlled by voltage u_x applied to control terminals.
- Current controlled current sources $j=ki_x$ controlled by current i_x injected into control terminals.
- Voltage controlled voltage sources $u=ku_x$.
- Current controlled voltage sources $u=ki_x$.

Models of controlled sources are very simple, however, their inclusion into equivalent model of the network considered may sometimes be cumbersome.

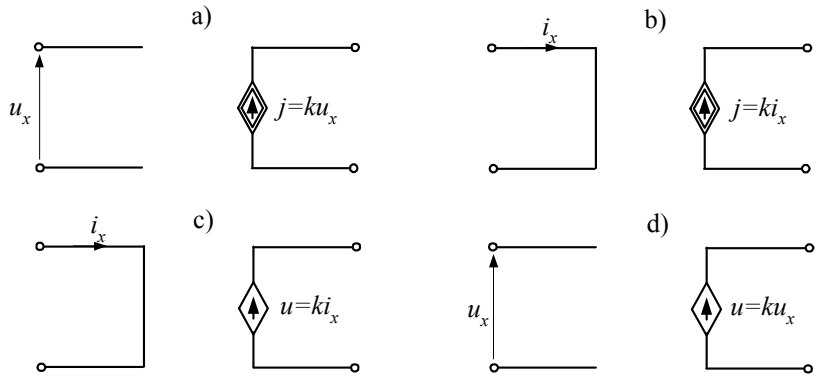


Fig.8. Diagrams of controlled sources; a) voltage controlled current source; b) current controlled current source; c) current controlled voltage source; d) voltage controlled voltage network.

Frequency properties of discrete models

The frequency properties of discrete models are uniquely determined by the method used for approximation of derivatives that appear in the continuous model of a given element. Comparison of the continuous and the discrete models frequency properties provides very useful information on how to select the calculation period T in order to obtain the accurate enough transient component waveform of frequency $f_{\max}$ – if such a component is present in the continuous transient voltages or currents.

As an example let's consider the discrete model of inductance obtained by use of trapezoidal approximation. Using the already known relations (2.1a, 1.10, 1.7) we get:

$$\frac{2(z-1)}{T(z+1)}i(z)=\frac{1}{L}u(z) \tag{2.13}$$

$$i(z)=\frac{T}{2L}\frac{z+1}{z-1}u(z) \tag{2.14}$$

Now using (1.4) and remembering that in frequency domain $s=j\omega$:

$$i(j\omega)=\frac{T}{2L}\frac{e^{j\omega T}+1}{e^{j\omega T}-1}u(j\omega) \tag{2.15}$$

Applying rudimentary trigonometry knowledge the equation (2.15) can be written in the simplified form:.

$$\left|i(j\omega)\right|=\frac{\frac{T}{2}}{Ltg\,\frac{\omega T}{2}}\left|u(j\omega)\right| \tag{2.16}$$

Introducing the complex discrete admittances $Y_d(j\omega)$ and $Y_c(j\omega)$ we get:

$$|Y_d(j\omega)| = \frac{|i(j\omega)|}{|u(j\omega)|} = \frac{\frac{T}{2}}{Ltg\frac{\omega T}{2}} = \frac{\frac{T\omega}{2}}{L\omega tg\frac{\omega T}{2}} = \frac{\frac{T\omega}{2}}{tg\frac{\omega T}{2}}|Y_c(j\omega)| \tag{2.17}$$

where $Y_c(j\omega) = 1/jL\omega$ is the admittance of the continuous model of inductance .

The ratio of the discrete and continuous admittances vary with frequency as it is shown in Fig. 9.

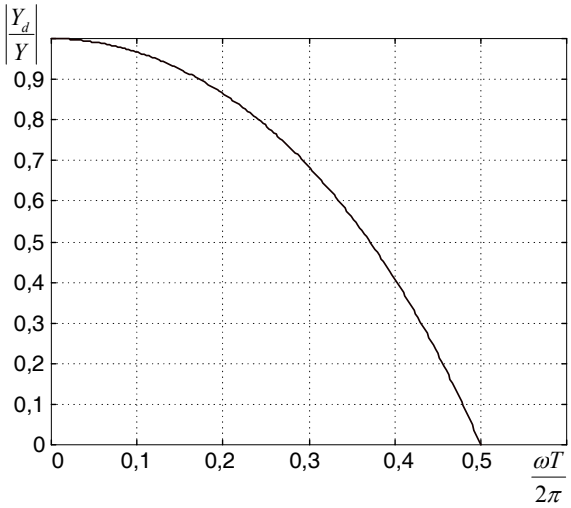


Fig.9. Discrete inductance model frequency response.

From eqn. (2.17) and from Fig. 9 one can notice that $Y_d(j\omega)$ reaches zero if $tg\frac{\omega T}{2} \to \infty$ This limit is reached when:

$$\frac{\omega T}{2} = \frac{\pi}{2} \quad or \quad T = \frac{\pi}{\omega} = \frac{\pi}{2\pi f} = \frac{1}{2f}$$

So if f_{max} is the frequency of the highest current or voltage harmonic expected then the following condition must be met:

$$T << \frac{1}{2f_{max}} \tag{2.18}$$

Practically, if requested number of data output within the period $T_{max} = \frac{1}{f_{max}}$ is N (usually $N > 20$) then (2.18) implies:

$$T_{max} \le \frac{1}{2Nf_{max}} \tag{2.19}$$

Distributed parameters model (long line model)

Distinction between lumped and distributed models of electric elements is made on the basis of mutual relation between three basic parameters of the environment in which the electromagnetic wave is propagated. These parameters are:

- specific electric conductivity γ
- relative magnetic permeability μ
- relative electric permitivity ε

In case of lumped elements it is assumed that only one of the above listed parameters is dominant and the remaining ones can be neglected. Thus particular elements are deemed as lumped under following conditions:

- μ = ε = 0 – lumped resistance
- γ = ε = 0 – lumped inductance
- γ = μ = 0 – lumped capacitance

Additionally in case of lumped parameters model of an electric network the electromagnetic field must be quasi-stationary; it means that in each point of the network the electromagnetic field is practically the same or the differences are negligibly small. In this respect the length of the electric conductor l is considered as the distinctive parameter. As the boundary value the length l_c equal to ¼ of the electromagnetic wavelength propagated is assumed.

Thus, if the frequency of the propagated wave is f, than the l_c can be estimated as:

$$l_{gr} = \frac{\lambda}{4} = \frac{c}{4f} \tag{3.1}$$

where c is the velocity of light and $\lambda = \frac{c}{f}$ is the wavelength.

If $l << l_{gr}$ then the length of the line can be neglected and can be modeled as the lumped parameter element. Otherwise ($l \approx l_{gr}$) the line should be considered as the long one.

For example, if the transient harmonics of frequency $f = 1000\,\text{Hz}$ (the 20th harmonic) may appear in the line during faults then $l_{gr} = c/(4f) = 3 \cdot 10^5/(4 \cdot 1000) = 75\,\text{km}$. The lightning stroke may induce much higher harmonics in the line so in such case even a few kilometers long line should be represented by distributed parameters model.

The long line representation

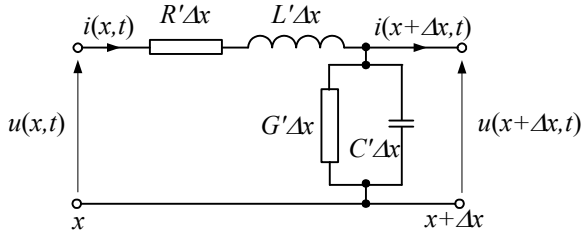


Fig.10. Elementary segment of a long line

The basic equations that describe the elementary cell in Fig.10 are:

$$\begin{aligned} u(x,t) &= R'\Delta x \cdot i(x,t) + L'\Delta x \frac{\partial i(x,t)}{\partial t} + u(x+\Delta x,t), \\ i(x,t) &= G'\Delta x \cdot u(x+\Delta x,t) + C'\Delta x \frac{\partial u(x+\Delta x,t)}{\partial t} + i(x+\Delta x,t), \end{aligned} \tag{3.2}$$

where R', L', G', C' denote 'unit/ length ' values of resistance, inductance and capacitance of the line, respectively.

Dividing both equations by Δx and taking the limes ($\Delta x \to 0$) the following relations are obtained:

$$\begin{aligned} -\frac{\partial u(x,t)}{\partial x} &= R'i(x,t) + L'\frac{\partial i(x,t)}{\partial t}, \\ -\frac{\partial i(x,t)}{\partial x} &= G'u(x,t) + C'\frac{\partial u(x,t)}{\partial t}. \end{aligned} \tag{3.3}$$

If the line is homogenous then (3.3) can be separated with respect to current and voltage (for simplicity: $u = u(x,t)$, $i = i(x,t)$):

$$-\frac{\partial u^2}{\partial x^2} = -R'G'u - R'C'\frac{\partial u}{\partial t} + L'\frac{\partial i^2}{\partial x \partial t}$$

and

$$\frac{\partial u^2}{\partial x^2} = R'G'u + \left(R'C' + G'L'\right)\frac{\partial u}{\partial t} + L'C'\frac{\partial u^2}{\partial t^2} \tag{3.4}$$

Applying the same simplifying procedure to the second equation in (3.3) the respective relation for current can be obtained:

$$\frac{\partial i^2}{\partial x^2} = R'G'i + \left(R'C' + G'L'\right)\frac{\partial i}{\partial t} + L'C'\frac{\partial i^2}{\partial t^2} \tag{3.5}$$

Both (3.4) and (3.5) are the second order hyperbolic partial differential equations known as "telegraph equation".

Lossless (non-dissipating) long line

This case is obtained under assumption that $R' = 0$ and $G' = 0$ and the resulting simplification of (3.4) and (3.5) is:

$$\begin{aligned} \frac{\partial u^2}{\partial x^2} - \frac{1}{v^2}\frac{\partial u^2}{\partial t^2} &= 0, \\ \frac{\partial i^2}{\partial x^2} - \frac{1}{v^2}\frac{\partial i^2}{\partial t^2} &= 0. \end{aligned} \tag{3.6}$$

in which:

$$v = \frac{1}{\sqrt{L'C'}}$$

The general solution of (3.6) has been found by d'Alembert [24], [17]. For the following boundary conditions:

$$u(x,t)\big|_{x=0} = \varphi(t) \quad , \quad \left.\frac{\partial u(x,t)}{\partial x}\right|_{x=0} = \psi(t)$$

the solution of (3.6) takes the form:

$$u(x,t) = \frac{1}{2}\big(\varphi(t + x/v) + \varphi(t - x/v)\big) + \frac{v}{2}\int_{t-x/v}^{t+x/v} \psi(\alpha)\mathrm{d}\alpha \tag{3.7}$$

The loci of points $(t - x/v) = \text{const}$ and $(t + x/v) = \text{const}$ known as propagation characteristics of (3.7) [9], [12] show the propagation mechanism of $\varphi(x,t)$ waves in a long line.

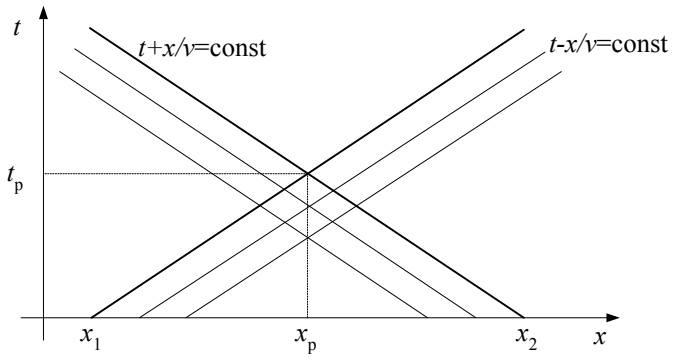


Fig.11. Propagation characteristics of a lossless long line

The boundary conditions expressed in terms of voltage $u_1(t)$ and current $i_1(t)$ at the beginning of the line yields (from (3.3)):

$$\varphi(t) = u(0,t) = u_1(t) \ , \ \psi(t) = \frac{\partial u(0,t)}{\partial x} = -L'\frac{\partial i(0,t)}{\partial t} = -L'\frac{\mathrm{d}i_1(t)}{\mathrm{d}t}$$

and the solution (3.7) takes the form:

$$u(x,t) = \frac{1}{2}\big(u_1(t + x/v) + u_1(t - x/v)\big) - \frac{1}{2}Z_f\int_{t-x/v}^{t+x/v} \mathrm{d}i_i(t) \tag{3.8}$$

where $Z_f = \sqrt{\frac{L'}{C'}}$ is the wave impedance of the line.

For $x = l$ (end of the line) (3.8) is:

$$u_2(t) = \frac{1}{2}\left(u_1(t+\tau) + u_1(t-\tau)\right) - \frac{1}{2} Z_f \left(i_1(t+\tau) - i_1(t-\tau)\right) \tag{3.9}$$

where $\tau = l/v$ is the line propagation time.

Similarly, the wave equation for current can be obtained and

$$i_2(t) = -\frac{1}{2}\left(i_1(t+\tau) + i_1(t-\tau)\right) + \frac{1}{2Z_f}\left(u_1(t+\tau) - u_1(t-\tau)\right) \tag{3.10}$$

Subtracting (3.9) from (3.10) the model of the long line takes the form:

$$i_2(t) = G_f u_2(t) - G_f u_1(t-\tau) - i_1(t-\tau) \tag{3.11}$$

where $G_f = \frac{1}{Z_f}$.

When the boundary conditions are assigned to the beginning and to the end of the line, the solution concerns these two points only. The propagation characteristics also comprise of 2 points: $x_1 = 0$ and $x_2 = l$. This simple model is call the Bergerons model [12], [17].
The continuous model (3.11) of the lossless line can easily be converted into the discrete one .
Assuming that wave propagation time is $mT = \tau$ then :

$$m = \frac{\tau}{T} = \frac{l}{vT} \tag{3.12}$$

and

$$i_2(k) = G_f u_2(k) - G_f u_1(k-m) - i_i(k-m) \tag{3.13}$$

By analogy the discrete model for the current at the beginning of the line can be derived, so:

$$\begin{aligned} i_1(k) &= G_f u_1(k) + j_1(k-m), \\ i_2(k) &= G_f u_2(k) + j_2(k-m), \end{aligned} \tag{3.14}$$

where

$$\begin{aligned} j_1(k-m) &= -G_f u_2(k-m) - i_2(k-m), \\ j_2(k-m) &= -G_f u_1(k-m) - i_1(k-m), \end{aligned} \tag{3.15}$$

The equivalent circuits corresponding to (3.14) and (3.15) are shown in Fig. 12.

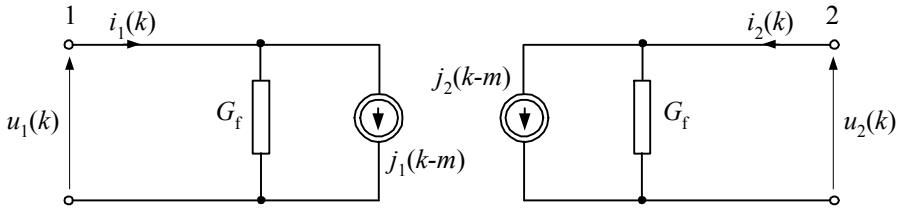


Fig.12. Equivalent circuit of the long line discrete model

The long line model with dissipation losses

The dissipation losses appear when resistance appears in the line model. The inclusion of the resistance is based upon assumption that its value is relatively small with respect to the line reactance. This assumption justifies the inclusion of the lumped resistance at both ends of the line as it is shown in Fig. 13.

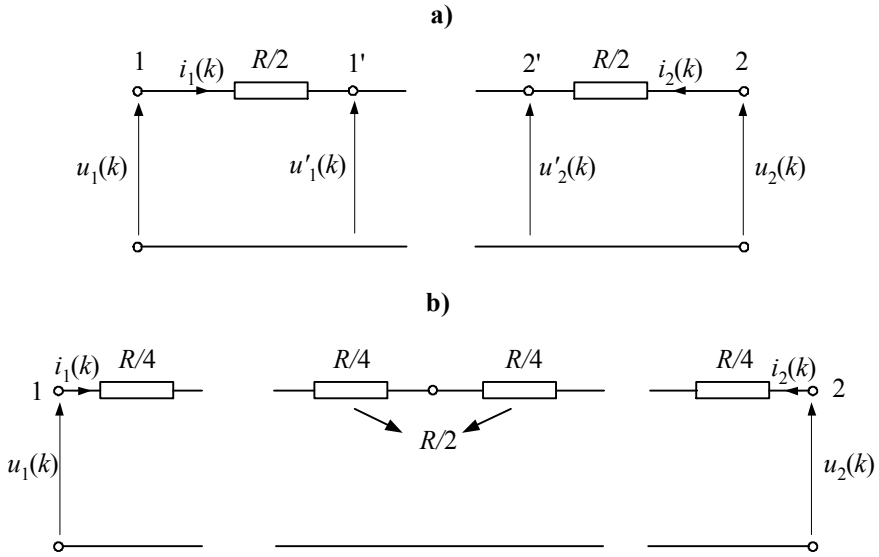


Fig.13 Inclusion of resistance into the long line model

When the resistance is connected as shown in Fig.13a. the equations (3.14), (3.15) take the form:

$$\begin{aligned} u'_1(k) &= u_1(k) - \frac{R}{2} i_1(k), \\ u'_2(k) &= u_2(k) - \frac{R}{2} i_2(k), \end{aligned} \tag{3.16}$$

where:

$$R = lR'$$

As the result the conductance G_f and history calculation changes so that:

$$\begin{aligned} j_1(k-m) &= -G_f u_2(k-m) - h_f i_2(k-m), \\ j_2(k-m) &= -G_f u_1(k-m) - h_f i_1(k-m), \end{aligned} \tag{3.17}$$

where:

$$G_f = \frac{1}{Z_f + R/2} \ , \ h_f = \frac{2Z_f - R}{2Z_f + R}$$

More accurate model can be obtained when the resistance is connected into the line model as it is shown in Fig.13b. In this case all the line parameters connected to the middle node of the line can be eliminated and the resulting equations obtained are:

$$\begin{aligned} i_1(k) &= G_f u_1(k) + h_{fa} j_1(k-m) + h_{fb} j_2(k-m), \\ i_2(k) &= G_f u_2(k) + h_{fa} j_2(k-m) + h_{fb} j_1(k-m), \end{aligned} \tag{3.18}$$

where: $h_{fa} = Z_f G_f$, $h_{fb} = \frac{R}{4} G_f$, and $G_f = \frac{1}{Z_f + R/4}$.

In general the dissipating long line models can be written in the compact matrix form so that:

$$\begin{bmatrix} i_1(k) \\ i_2(k) \end{bmatrix} = \begin{bmatrix} G_f & \\ & G_f \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} + \begin{bmatrix} h_{fa} & h_{fb} \\ h_{fb} & h_{fa} \end{bmatrix} \begin{bmatrix} j_1(k-m) \\ j_2(k-m) \end{bmatrix} \tag{3.19}$$

and the matrixes $\mathbf{G}_f = \{G_f\}$ and $\mathbf{h}_f = \{h_f\}$.

The form of the matrixes depends upon the considered representation of the dissipating long line (as in Fig. 13a or as in Fig. 13.b).

Numerical stability of digital models

Numerical models used for simulation of transient processes in power networks can be deemed as satisfactory if the simulation results are adequate processes observed in real networks. There are two basic sources of errors that can make the simulation results inadequate, namely,

- omission of the elements which are essential for the network operation
- application of numerical methods that are inadequate to calculation of analyzed effects.

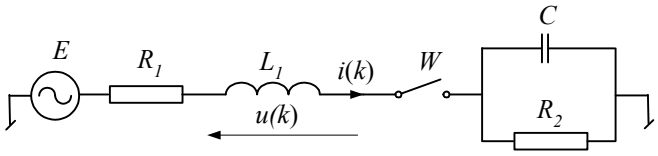
The problems concerned may appear in some specific situations only. For example, the ideal switch that is represented by two limit values of conductance (0 and ∞) can be used as a circuit breaker if the values of the current to be broken are relatively low. Similar problems may occur due to application of inadequate numerical methods resulting in numerical instability.
Numerical instability appears when the errors caused by numerical round up of calculation results sum up in each calculation step.

Practically, the both considered types of errors are related very closely as the further analysis shows .

Numerical oscillations in transient state simulations

As the typical illustration of the problem let's consider the following example:

The task: *simulate the transient effects that appear in the network shown in Fig.14 when the switch opens at* t_{open} *=0.012s. Assume that the models of elements used are companion to trapezoidal approximation method.*



The element parameters: R_1=1Ω, L_1=100mH, R_2=1000Ω, C =4,7μF, E=100cos(100πt).

Fig.14. The simulated network

The respective waveforms of the current flowing through the switch and the voltage drop across the inductance L_1 are shown in Fig.15.

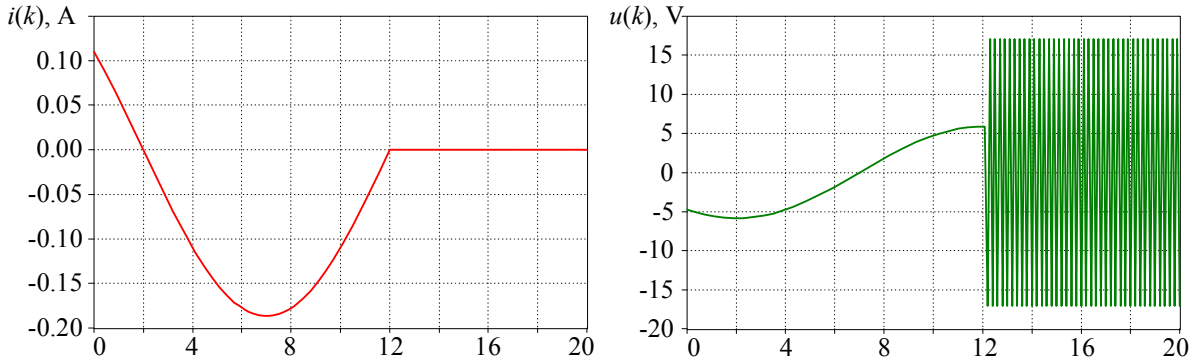


Fig 15. The results of simulation; a) the current in the switch; b) the voltage across the inductance L_1

As one can see the network current drops to zero when the switch opens but the voltage across inductance oscillates with constant non-decaying amplitude of relatively small value since the value of the current at the breaking moment is also very small. A closer look at the oscillating voltage (Fig.16) reveals that it changes its sign in each calculation step.
The oscillations appear since the energy stored in the coil cannot be dissipated (the circuit is broken). Thus the observed error in simulation result can be credited to inadequate model applied. Such errors may appear in less obvious situations (some model parameters drastically change their values within one calculation step).
To analyze the described numerical effect let's consider the voltage drop across the inductance which, in case of numerical model derived for trapezoidal approximation, can be expressed as (*derive this equation)*:

$$u(k) = \frac{1+RG_L}{G_L} i(k) - \frac{1-RG_L}{G_L} i(k-1) - u(k-1) \qquad (3.20)$$

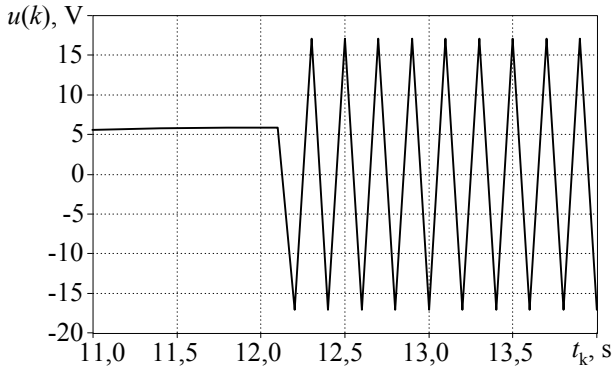


Fig.16. Oscillating inductance voltage

When the switch opens at k-1 instant the current attains zero in two consecutive steps ($i(k) = i(k-1) = 0$). Thus $u(k) = -u(k-1)$ for all further calculation steps.

There are many methods that can be applied to damp such oscillations; they are known as critical damping adjustment methods (CDA) [27], [29].

Suppression of oscillations by use of a damping resistance.

The most obvious way of oscillation suppression is the use of nonlinear model that matches reality. However, sometimes this approach may be very difficult or even impossible. In such cases the use of linear resistance can bring the satisfactory effects.

The analysis of the network in Fig. 14. immediately brings to the conclusion that the use of resistance connected in parallel with the coil should result in suppression of voltage oscillations. In such case the modified inductance model takes the form (Fig.17):

$$i(k) = \frac{T}{2L}\big(u(k) + u(k-1)\big) + i(k-1) + \frac{1}{R}\big(u(k) - u(k-1)\big) \tag{3.21}$$

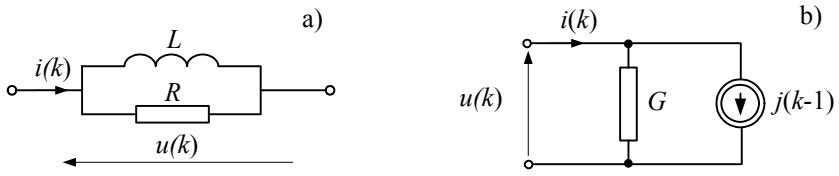


Fig.17. Modified inductance model

In standard notation it is:

$$i(k) = Gu(k) + j(k-1) \tag{3.22}$$

where:

$$G = \frac{TR + 2L}{2LR},\ j(k-1) = i(k-1) + \frac{TR - 2L}{2LR}u(k-1).$$

Voltage across the modified inductance is:

$$u(k)=\frac{1}{G}\big(i(k)-i(k-1)\big)-\alpha u(k-1) \tag{3.23}$$

where:

$$\alpha=\frac{R-\dfrac{2L}{T}}{R+\dfrac{2L}{T}}$$

The coefficient α is responsible for damping of oscillations. If $R=\infty$, $\alpha=1$. The lower value of R the lower value of α. The oscillations on inductance in the example circuit for different values of α are shown in Fig. 18.

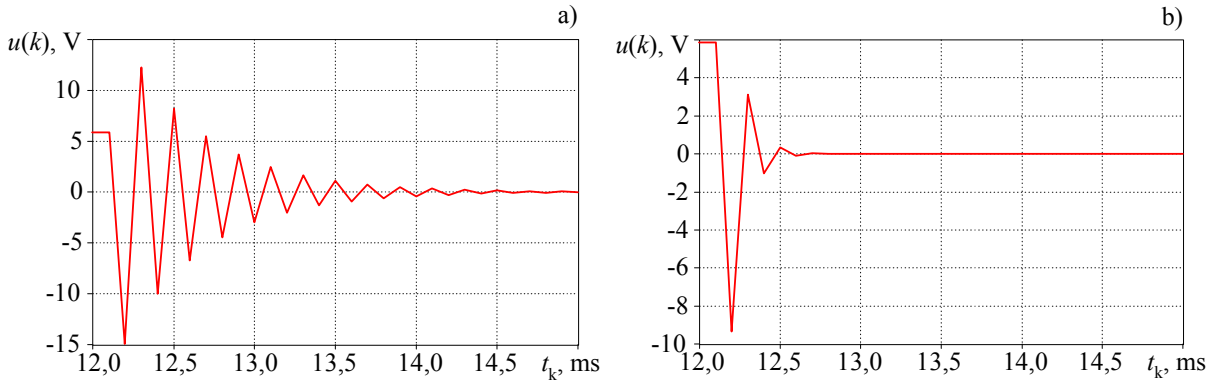


Fig.18. Oscillations on the inductor for different values of α . α=0,818 (a) and α=0,333 (b)

The similar effects can be observed on capacitances in case of rapid decrease of the capacitance voltage.

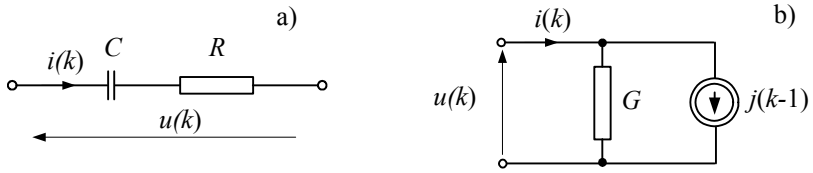


Fig.19. Series *RC* model.

The respective relations are:

$$i(k)=\frac{2C}{T}\big(u(k)-Ri(k)\big)-i(k-1)-\frac{2C}{T}\big(u(k-1)-Ri(k-1)\big) \tag{3.24}$$

$$i(k)=G\big(u(k)-u(k-1)\big)-\alpha i(k-1) \tag{3.25}$$

where:

$$G = \frac{2C}{T + 2RC}, \quad \alpha = \frac{\frac{T}{2C} - R}{\frac{T}{2C} + R}$$

In this case the oscillations of current occur for $\alpha = 1$ (R=0) at the moment when $u(k)=u(k-1)=0$.
It must be noted that the damping resistor changes the frequency response of the model considered. For example, in case of inductance the eqn.(2.15) is now:

$$i(j\omega) = G\frac{e^{j\omega T} + \alpha}{e^{j\omega T} - 1} u(j\omega) = \underline{Y}_d \underline{U}_d(j\omega) \tag{3.26}$$

and G and α are as in (3.22, 3.23).

The relation between the digital Y_d and continuous Y_c admittances for different values of α are shown in Fig.20.

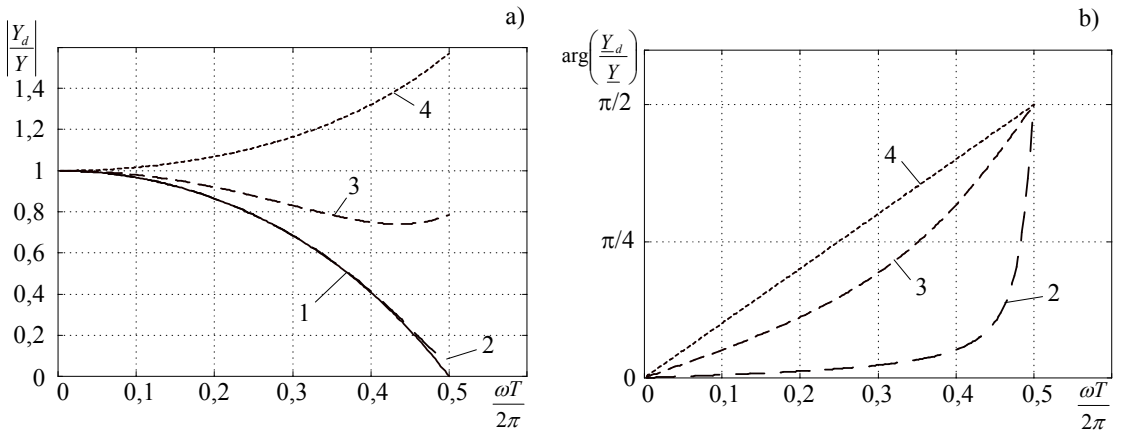


Fig.20. Frequency response for magnitude and argument of the relation $\underline{Y}_d / \underline{Y}_c$;
1 - α = 1, 2 - α = 0.818, 3 - α = 0.333, 4 - α = 0.

Suppression of numerical oscillations by change of integration method

The analysis carried out above shows that numerical oscillations are related directly to the method of continuous derivative approximation.
Using the three different approximations, namely:

$$u(k) = \frac{L}{T}\big(i(k) - i(k-1)\big) \qquad \text{implicit Euler's method}$$

$$u(k) = \frac{2L}{T}\big(i(k) - i(k-1)\big) - u(k-1) \qquad \text{trapezoidal approximation}$$

$$u(k) = \frac{L}{2T}\big(3i(k) - 4i(k-1) + i(k-2)\big) \qquad \text{Gear's } 2^{\text{nd}} \text{ order}$$

for the same network model (example) different intensity of numerical oscillations can be observed. It is shown in Fig.21.

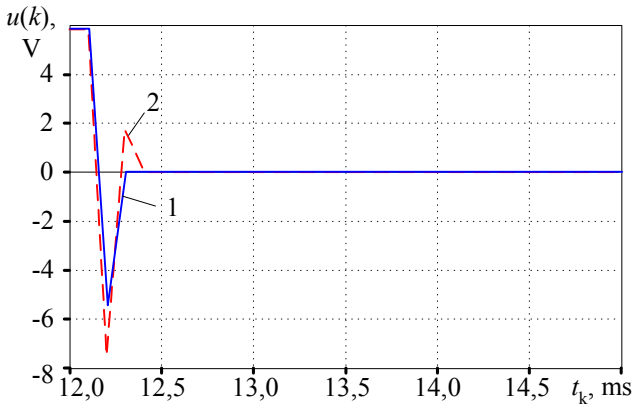


Fig.21. Oscillations at the inductor (sample network);
1 – implicit Euler's method,: 2 – Gear's 2nd order

The Euler's method reveals the best oscillation damping property since they are suppressed in one calculation step (critical damping). The Gear's method is slightly worse. On the other hand the trapezoidal method that is least stable offers simplicity and good accuracy of calculations in steady state (no rapid changes of the network parameters).
Thus, in practice, the combination of Euler's and trapezoidal methods are applied in the following way:

- if step there are no rapid changes of the network parameters in the current calculation the trapezoidal method is used,
- otherwise the Euler's implicit method takes the calculations over for 2 consecutive steps but of twice shorter duration ($T/2$) to avoid the model parameter change.

The root matching technique

Another approach to the numerical network representation that results in suppression of numerical oscillation is the use of the root matching technique. In this approach the network model is based on the continuous transfer function relating current and voltage in the network considered. In general such a transfer function has the form:

$$H(s) = \frac{(s - s_{z1})(s - s_{z2})...(s - s_{zM})}{(s - s_{p1})(s - s_{p2})...(s - s_{pN})} \tag{3.27}$$

Transformation to the discrete domain is obtained by replacement of continuous zeros and poles by their discrete counterparts:

$$z_i = e^{s_i T} \quad i \text{ number of respective zero and pole.} \tag{3.28}$$

so that:

$$H_d(z) = \frac{D(z - e^{s_{z1}T})(z - e^{s_{z2}T})...(z - e^{s_{zM}T})}{(z - e^{s_{p1}T})(z - e^{s_{p2}T})...(z - e^{s_{pN}T})} \tag{3.29}$$

This operation is called a *matched* Z *transform* [20].

The constant *D* is determined by comparison of steady state response for specified excitation that should be the same for the continuous and the discrete system.
Since the calculations are carried out in off-line mode the input signal can be represented (sampled) in many different ways. Some of them are shown in Fig.22.

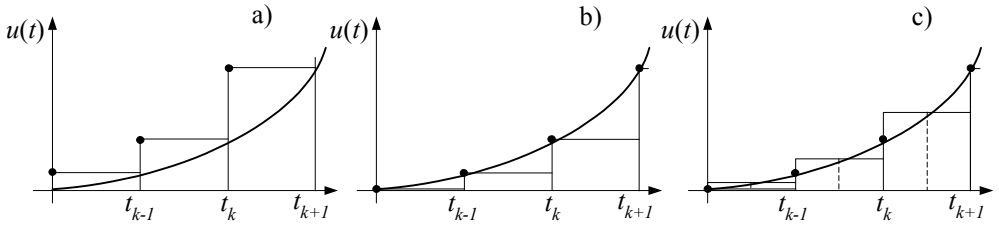


Fig.22. The ways of the continuous input signal representation (sampling).

Thus the final discrete transfer function *H(z)* must be corrected accordingly by use of the sampling function $H_s(z)$ related to the applied signal sampling operation:

$$H(z) = H_d(z)H_s(z) \tag{3.30}$$

and

$H_s(z) = z$ for Fig 22a

$H_s(z) = 1$ for Fig.22b

$H_s(z) = \frac{1}{2}(z+1)$ for Fig.22c

Thus the algorithm of the *matched* Z *transform* application can be summarized as follows:

- determine the continuous transfer function *H(s)* of the network considered.
- transform *H(s)* into *H(z)* replacing all continuous zeros and poles by use of $z = e^{sT}$.
- determine the constant *D* so that $\mathsf{L}\{y(t)\}_{t\to\infty} = \mathsf{Z}\{y(k)\}_{k\to\infty}$. for specified input signal - *y* stands for the output variable (current or voltage).

Example

Consider the network shown in Fig.23a which has the transfer function block diagram representation like in Fig.23b. The resulting continuous transfer function of the network is:

$$U(s) = (R + sL)I(s) \quad \text{and} \quad H(s) = \frac{I(s)}{U(s)} = \frac{\frac{1}{R}}{1 + s\frac{L}{R}} = \frac{K}{1 + s\tau}$$

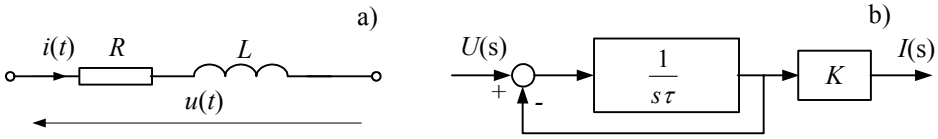


Fig.23. The network considered a) and its equivalent block diagram b)

The transfer function obtained has no zeros and only one pole $s_{p1} = -1/\tau = -R/L$.
Assuming that the input signal (voltage) is sampled as in Fig.22a and using the transformation (3.28) the resulting discrete transfer function is obtained:

$$H(z) = \frac{D}{\left(z - e^{-TR/L}\right)}z = \frac{Dz}{\left(z - e^{-TR/L}\right)}$$

To determine *D* let's assume that *u(t)* is a unit step function *u(t)* = 1*(t)* for which *U(s)=1/s* and:

$$i(s) = \frac{K}{s(1 + s\tau)}.$$

The steady state response is now:

$$i(t)\big|_{t\to\infty} = \lim_{s\to 0} sI(s) = \lim_{s\to 0} \frac{K}{1 + s\tau} = K.$$

Similarly for *u(k)*=1*(k)* and $Z\{1(k)\} = z/(z-1)$:

$$I(z) = \frac{Dz^2}{\left(z - e^{-TR/L}\right)(z-1)}$$

and

$$i(k)\big|_{k\to\infty} = \lim_{z\to 1}(z-1)I(z) = \lim_{z\to 1} \frac{Dz^2}{z - e^{-TR/L}} = \frac{D}{1 - e^{-TR/L}}.$$

By comparison of steady state responses we get:

$$D = \left(1 - e^{-TR.L}\right)K.$$

So, finally:

$$H(z)=\frac{I(z)}{U(z)}=\frac{\left(1-e^{-TR/L}\right)z}{R\left(z-e^{-TR/L}\right)}.$$

Now in few steps the numerical algorithm for calculation of *i(k)* can be written:

$$RI(z)\left(1-e^{-TR/L}z^{-1}\right)=U(z)\left(1-e^{-TR/L}\right)$$

$$i(k)=Gu(k)+j(k-1),$$

where: $G=\frac{\left(1-e^{-TR/L}\right)}{R}$, $j(k-1)=e^{-TR/L}i(k-1)$.

The past history of the algorithm depends upon the current only so the voltage oscillations on inductance due to the rapid change of current will not occur.
In the similar way the numerical algorithms corresponding to the typical first and the higher order transfer functions and related to them electrical elements can be developed
It must be noted that the method considered can be applied to the transfer functions that have at least one zero or pole located outside the origin of the *s* plane. Thus the single R, C elements must be modeled using the trapezoidal method.

Another example (Fig.24) shows the comparison of oscillation suppression properties for different numerical methods.

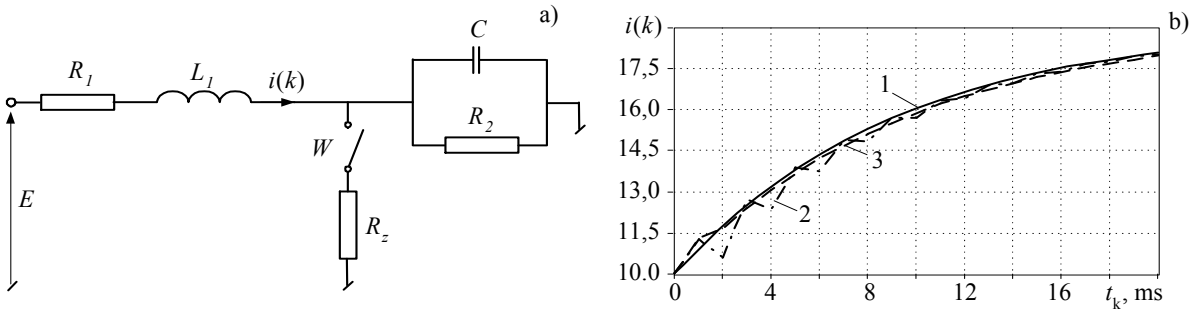


Fig.24. The current *I(k)* waveform after closing of W; 1- accurate values; 2- trapezoidal method; 3 - the root matching method.

The obtained current waveforms are evidently in favor of the root-matching algorithm.

The state variable method

This method is based on the mathematical model of a network in which the vector of input variables $\mathbf{f}(t)$ (sources) is related with the output variable vector $\mathbf{y}(t)$ (selected currents and voltages) by the fundamental equations:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{f}(t),\\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{f}(t),\end{aligned} \tag{4.1}$$

where: $\mathbf{A}_{n\times n}$, $\mathbf{B}_{n\times r}$, $\mathbf{C}_{m\times n}$, $\mathbf{D}_{m\times r}$ - matrixes of parameters

$\mathbf{x}(t)$ - state vector

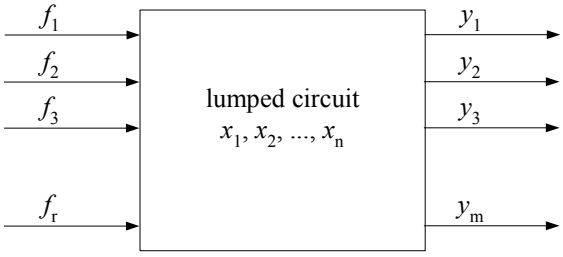


Fig. 25. Network state variable representation

In general the matrixes of parameters can change in time (time-variant network) or can be functions of the state variables (non-linear network). The state variables are related with energy preserving elements (inductances capacitances) for which:

$$\frac{d}{dt}\left(Li_L(t)\right) = u_L(t) \tag{4.2}$$

and

$$\frac{d}{dt}\left(Cu_C(t)\right) = i_C(t) \tag{4.3}$$

so the voltages on inductances and currents flowing through capacitances are always selected as state variables. Thus any network can be represented as it is shown in Fig.26.

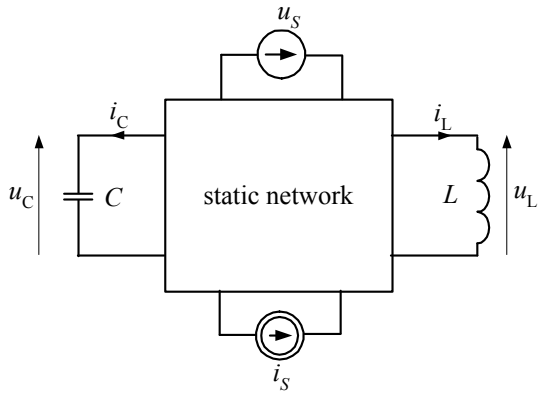


Fig.26. Network represented by state variables.

The inertial elements (Fig.26) can be described by the following general equations:

$$\begin{aligned} \mathbf{i}_C(t) &= \mathbf{g}_C\left(\mathbf{u}_C, \mathbf{i}_L, \mathbf{u}_S, \mathbf{i}_S, t\right), \\ \mathbf{u}_L(t) &= \mathbf{g}_L\left(\mathbf{u}_C, \mathbf{i}_L, \mathbf{u}_S, \mathbf{i}_S, t\right), \end{aligned} \tag{4.4}$$

which lead to the state equation (4.1).

The number of state variable doesn't always be equal to the sum of inductors and coils in the network. One condenser in an independent circuit closed loop *CE* or one coil from the independent node *LJ* can always be eliminated as it is shown in Fig.27.

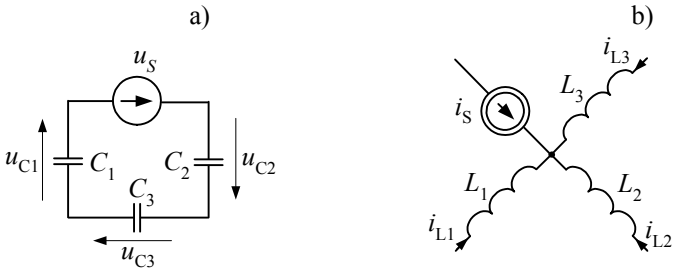


Fig.27. Example of independent loop*CE* (a) and independent node *LJ*(b)

For the circuit loop in Fig.27a there is:

$$u_S(t) + u_{C1}(t) + u_{C2}(t) + u_{C3}(t) = 0 \tag{4.5}$$

To make the circuit loop equation independent one of the variables should be eliminated.
The same concerns the independent node (Fig.27b) for which:

$$i_S(t) + i_{L1}(t) + i_{L2}(t) + i_{L3}(t) = 0 \tag{4.6}$$

In case of linear, time invariant lumped networks the number of indispensable state variables n can be determined as:

$$n = n_{LC} - \left(n_{CE} + n_{LJ}\right) \tag{4.7}$$

where:

n_{LC} - the number of all condensers and coils in the network
n_{CE} - the number of independent circuit loops *CE* in the network
n_{LJ} - the number of independent circuit nodes *LJ* in the network

Examples of independent nodes *LJ* and loops *CE* are shown in Fig.28.
Reduction of state variable number according to (4.7) results in the presence of input sources derivatives in the input vector.

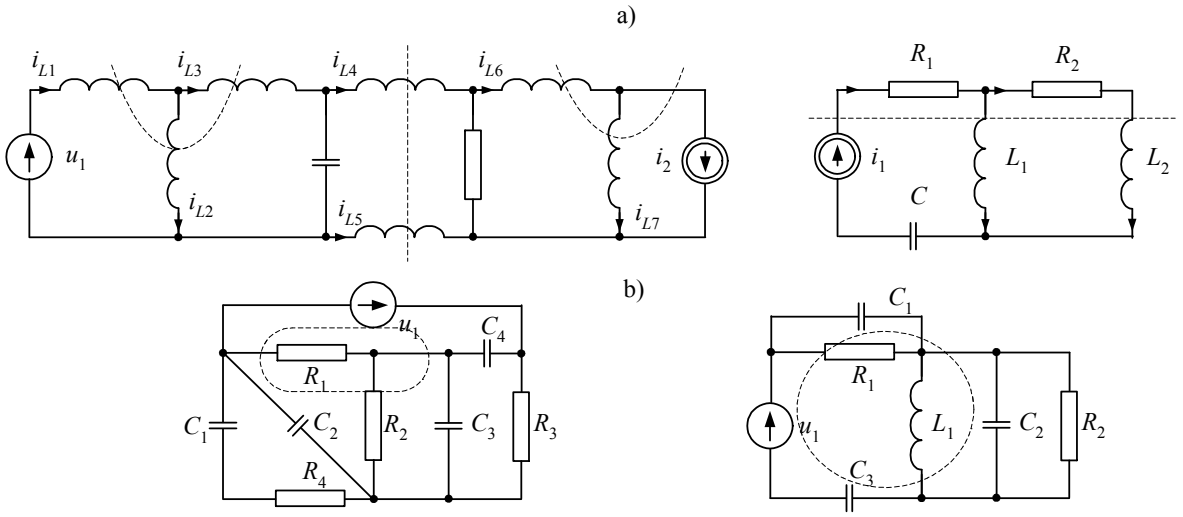


Fig.28. Examples of independent nodes *LJ* (a) and circuit loops *CE* (b)

State space representation of an electric network

State space representation of an electric network is obtained by application of Kirchoff's rules to the network layout for state variables assigned to reactive elements (condensers, coils) in the number selected according to (4.7).

Example

Determine the state space representation of the circuit shown in Fig.29.

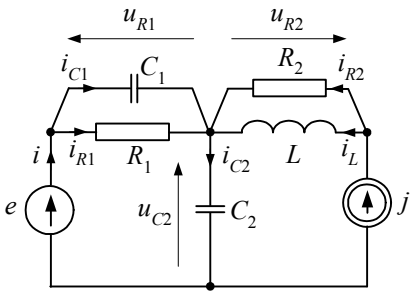


Fig.29. The sample circuit digram

The circuit contains 3 *LC* (n_{LC} =3) components but n_{CE} =1 since one loop (*e*, C_1, C_2) is linearly dependent and n_{LJ} =0. Thus the number of independent equations in overall state equation is 2. The selected state variables are the current i_L and the voltage u_{c2} . As the output variables the currents i_L , i_{c1} and i_{c2} are selected.

The respective circuit equations are:

$$j - i_L - i_{R2} = 0\,,\; C_1 \frac{du_{R1}}{dt} + i_{R1} - i = 0$$

$$e - u_{R1} - u_{C2} = 0\,,\; L\frac{di_L}{dt} - R_2 i_{R2} = 0\,,\; R_1 i_1 - u_{R1} = 0$$

Eliminating all variables, except for selected state variables and excitations (sources), we get:

$$\frac{di_L}{dt} = -\frac{R_2}{L} i_L + \frac{R_2}{L} j$$

$$\frac{du_{C2}}{dt} = -\frac{1}{R_1(C_1 + C_2)} u_{C2} + \frac{1}{C_1 + C_2} j + \frac{1}{R_1(C_1 + C_2)} e + \frac{C_1}{C_1 + C_2}\frac{de}{dt}$$

The output variables (aside of i_L) are:

$$i_{C1} = \frac{C_1}{R_1(C_1 + C_2)} u_{C2} - \frac{C_1}{C_1 + C_2} j - \frac{C_1}{R_1(C_1 + C_2)} e + \frac{C_1^2}{C_1 + C_2}\frac{de}{dt}$$

$$i_{C2} = -\frac{C_2}{R_1(C_1 + C_2)} u_{C2} + \frac{C_2}{C_1 + C_2} j + \frac{C_2}{R_1(C_1 + C_2)} e + \frac{C_1 C_2}{C_1 + C_2}\frac{de}{dt}$$

In matrix notation :

$$\frac{d}{dt}\begin{bmatrix} i_L \\ u_{C2} \end{bmatrix} = \begin{bmatrix} -\frac{R_2}{L} & 0 \\ 0 & -\frac{1}{R_1(C_1 + C_2)} \end{bmatrix}\begin{bmatrix} i_L \\ u_{C2} \end{bmatrix} + \begin{bmatrix} \frac{R_2}{L} & 0 \\ \frac{1}{C_1 + C_2} & \frac{1}{R_1(C_1 + C_2)} \end{bmatrix}\begin{bmatrix} j \\ e \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{C_1}{C_1 + C_2} \end{bmatrix}\frac{d}{dt}\begin{bmatrix} j \\ e \end{bmatrix}$$

$= \mathbf{Ax} + \mathbf{Bf} + \mathbf{B}_1 \frac{d}{dt}\mathbf{f}$ (the state equation)

$$\begin{bmatrix} i_L \\ i_{C1} \\ i_{C2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{C_1}{R_1(C_1 + C_2)} \\ 0 & -\frac{C_2}{R_1(C_1 + C_2)} \end{bmatrix}\begin{bmatrix} i_L \\ u_{C2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{C_1}{C_1 + C_2} & -\frac{C_1}{R_1(C_1 + C_2)^2} \\ \frac{C_2}{C_1 + C_2} & \frac{C_2}{R_1(C_1 + C_2)^2} \end{bmatrix}\begin{bmatrix} j \\ e \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{C_1^2}{C_1 + C_2} \\ 0 & \frac{C_1 C_2}{C_1 + C_2} \end{bmatrix}\frac{d}{dt}\begin{bmatrix} j \\ e \end{bmatrix}$$

$= \mathbf{Cx} + \mathbf{Df} + \mathbf{D}_1 \frac{d}{dt}\mathbf{f}$ (the output equation)

Elimination of source variable derivatives from the state equation can be obtained by substitution:

$$u_e = u_{C2} - \frac{C_1}{C_1 + C_2} e$$

As a result the state equations take the form (4.1) in which:

$$\mathbf{x} = \begin{bmatrix} i_L \\ u_e \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} -\dfrac{R_2}{L} & 0 \\ 0 & -\dfrac{1}{R_1(C_1+C_2)} \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} \dfrac{R_2}{L} & 0 \\ \dfrac{1}{C_1+C_2} & \dfrac{C_2}{R_1(C_1+C_2)^2} \end{bmatrix},\ \mathbf{f} = \begin{bmatrix} j \\ e \end{bmatrix}$$

and

$$\mathbf{y} = \begin{bmatrix} i_L \\ i_{C1} \\ i_{C2} \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & \dfrac{C_1}{R_1(C_1+C_2)} \\ 0 & -\dfrac{C_2}{R_1(C_1+C_2)} \end{bmatrix},\ \mathbf{D} = \begin{bmatrix} 0 & 0 \\ -\dfrac{C_1}{C_1+C_2} & -\dfrac{C_1C_2}{R_1(C_1+C_2)^2} \\ \dfrac{C_2}{C_1+C_2} & \dfrac{C_2^2}{R_1(C_1+C_2)^2} \end{bmatrix},\ \mathbf{D}_1 = \begin{bmatrix} 0 & 0 \\ 0 & \dfrac{C_1^2}{C_1+C_2} \\ 0 & \dfrac{C_1C_2}{C_1+C_2} \end{bmatrix}$$

In general, the derivatives of excitation signals can be eliminated from the state equation by use of suitable substitutions but not from the output one.

References

[1] Alvarado F.L., Lasseter R.H., Sanchez J.J., *Testing of trapezoidal integration with damping for the solution of power transient problems*. IEEE Transactions on Power Apparatus and Systems. Vol. PAS-102, No.12, December 1983, s. 3783-3790.

[2] Ametani A. Baba Y., *Frequency-dependent line and cable modelling*. EEUG Course, 27-th September 2000, Wrocław, Poland.

[3] *ATP-EMTP Rule Book*, Canadian/American EMTP User Group, 1987-92.

[4] Bernas S., Ciok Z., *Modele matematyczne elementów systemu elektroenergetycznego*. WNT, Warszawa 1977.

[5] Białko M., *Analiza układów elektronicznych wspomagana komputerem*. WNT, Warszawa 1989.

[6] Cholewicki T., *Elektrotechnika teoretyczna*. T. II. WNT, Warszawa, 1971.

[7] Chua L.O., Lin P.M., *Komputerowa analiza układów elektronicznych. Algorytmy i metody obliczeniowe*. WNT, Warszawa 1981.

[8] Долгинов А.И., Ступель А.И., Левина Л.С., *Алгоритмы и программа расчета на ЭВМ электромагнитных переходных процессов в электрических системах*. Электричество, 1966, № 8, s. 23-29.

[9] Dommel H.W., *Digital computer solution of electromagnetic transients in singl- and multiphase networks*. IEEE Transactions on Power Apparatus and Systems. Vol. PAS-88, April 1969, s. 388-399.

[10] Dommel H.W., *Electromagnetic Transients Program. Reference Manual (EMTP theory book)*. Bonneville Power Administration, Portland 1986.

[11] Dommel H.W., *Introduction to the use of MicroTran® and other EMTP versions*. Notes used in Graduate Course ELEC 553 'Advanced Power Systems Analysis': http://www.ece.ubc.ca/power/e553data/e553.pdf.

[12] Джуварлы Ч.М., Дмитриев Е.В., *Применение метода характеристик к решению т-проводной линии электропередач*. Изв. АН СССР. Энергетика и Транспорт, 1967, № 1.

[13] *Electromagnetic Transients Program (EMTP) Workbook*. Electric Power Research Institute, Palo Alto 1986. EL-4651, Research Project 2149-6.

[14] Forsythe G.E., Malcolm M.A., Moler C.B., *Computer methods for mathematical computations*, Englewood Cliffs, N.J., Prentice-Hall, Inc., 1977.

[15] Fortuna Z., Macukow B., Wąsowski J., *Metody numeryczne*, Warszawa, WNT, 1993.

[16] Haginomori E., *Highly Sophisticated Electric Power Systems*. Japanese ATP User Group: http://www.jaug.gr.jp/atp/index.shtml.

[17] Henschel S., Ibrahim A.I., Dommel H.W., *Transmission line model for variable step size simulation algorithms*. Electrical Power and Energy Systems. 21 (1991), s. 191-198.

[18] Ibrahim A.I., *Frequency dependent network equivalents for electromagnetic transients studies: a bibliographical survey*. Electrical Power and Energy Systems. 25 (2003), s. 193-199.

[19] Ibrahim A.I., Henschel S., Lima A.C., Dommel H.W., *Application of a new EMTP line model for short overhead lines and cables*. Electrical Power and Energy Systems. 24 (2002), s. 639-645.

[20] Jackson L.B., *Digital filters and signal processing*. Kluwer Academic Publisher, Boston 1986.

[21] Kiełbasiński A., Schwetlick H., *Numeryczna algebra liniowa*, Warszawa, WNT, 1992.

[22] Kacejko P., Machowski J., *Zwarcia w systemach elektroenergetycznych.* WNT, Warszawa 2002.

[23] Kaczorek T., *Wektory i macierze w automatyce i elektrotechnice*, Warszawa, WNT, 1998.

[24] Kącki R., *Równania różniczkowe cząstkowe w zagadnieniach fizyki i techniki*. WNT, Warszawa 1989.

[25] Kizilcay M., *Review of solution methods in ATP-EMTP.* EEUG News, Feb-May 2001, s. 25-36.

[26] Kudrewicz J., *Nieliniowe obwody elektryczne*. WNT, Warszawa 1996.

[27] Lin J., Marti J.,R., *Implementation of the CDA procedures in the EMTP*. IEEE Transactions on Power Systems. Vol. 5, No. 2, May 1990, s. 394-401.

[28] Marti J.,R., *Accurate modelling of frequency-dependent transmission lines in electromagnetic transient simulations*. IEEE Transactions on Power Apparatus and Systems. Vol. PAS-101, No.1, January 1982, s. 147-155.

[29] Marti J.,R., Lin J., *Suppression of numerical oscillations in the EMTP*. IEEE Transactions on Power Systems. Vol. 4, No.2, May 1989, s. 739-745.

[30] Marti L., *Low-order approximation of transmission line parameters for frequency-dependent models*. IEEE Transactions on Power Apparatus and Systems. Vol. PAS-102, No.11, November 1983, s. 3582-3589.

[31] Marti L., *Simulation of transients in underground cables with frequency-dependent modal transformation matrices*. IEEE Transactions on Power Delivery. Vol. 3, No.3, July 1988, s. 1099-1106.

[32] Meyer W.S., Dommel H.W., *Numerical modelling of frequency-dependent transmission line parameters in an electromagnetic transients program*. IEEE Transactions on Power Apparatus and Systems. Vol. PAS-93, Sept./Oct. 1974, s. 1401-1409.

[33] Ogrodzki J., *Komputerowa analiza układów elektronicznych*. Wydawnictwo Naukowe PWN, Warszawa 1994.

[34] Osiowski J., Szabatin J., *Podstawy teorii obwodów. T. I.* WNT, Warszawa 1992.

[35] Osiowski J., Szabatin J., *Podstawy teorii obwodów. T. II.* WNT, Warszawa 1993.

[36] Osiowski J., Szabatin J., *Podstawy teorii obwodów. T. III*. WNT, Warszawa 1995.

[37] Osowski S., *Modelowanie układów dynamicznych z zastosowaniem języka SIMULINK*. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 1999.

[38] Press W.H., Teukolsky S.A., Vetterling W.T., Flannery B.P., *Numerical recipes in C. The art of scientific computing*. Cambridge University Press, Cambridge 1997.

[39] Rosołowski E., *Cyfrowe przetwarzanie sygnałów w automatyce elektroenergetycznej*. Oficyna Wydawnicza EXIT, Warszawa 2002.

[40] Сигорский В.П., Петренко А.И., *Алгоритмы анализа электронных схем*. СОВЕТСКОЕ РАДИО, Москва 1976.

[41] Stoer J., Bulrisch R., *Wstęp do analizy numerycznej*, Warszawa, PWN, 1987.

[42] Tadeusiewicz M., *Metody komputerowej analizy stałoprądowej nieliniowych układów elektronicznych*. WNT, Warszawa 1991.

[43] Tewarson R.P., *Sparse matrices*. Academic Press, New York 1973.

[44] Ульянов С.А., *Электромагнитные переходные процессы в электрических системах*. ЭНЕРГИЯ, Москва 1970.

[45] Watson N., Arrillaga J., *Power systems electromagnetic transients simulation*. The Institution of Electrical Engineers, London 2003.

[46] Watson N.R., Irwin G.D,. *Electromagnetic transient simulation of power systems using root-matching techniques*. IEE Proc.-Gener. Transm. Distrib. Vol. 145, No. 5, September 1998, s. 481-486.

[47] Watson N.R., Irwin G.D,. *Comparison of root-matching techniques for electromagnetic transient simulation.* IEEE Transactions on Power Delivery. Vol. 15, No. 2, April 2000, s. 629-634.

[48] Wedepohl L.M., *Application of matrix methods to the solution of travelling wave phenomena in polyphase systems*. Proc. IEE, vol. 110 (12) 1963, s. 2200-2212.

[49] Wojtkiewicz A., *Elementy syntezy filtrów cyfrowych*. WNT, Warszawa 1984.