



## Research Article

## Open Access

Petr Veigend\*, Gabriela Nečasová, and Václav Šátek

# Model of the telegraph line and its numerical solution

DOI: <https://doi.org/10.1515/comp-2018-0002>

Received February 28, 2018; accepted April 25, 2018

**Abstract:** This paper deals with a model of the telegraph line that consists of system of ordinary differential equations, rather than partial differential telegraph equation. Numerical solution is then based on an original mathematical method. This method uses the Taylor series for solving ordinary differential equations with initial condition – initial value problems in a non-traditional way. Systems of ordinary differential equations are solved using variable order, variable step-size Modern Taylor Series Method. The Modern Taylor Series Method is based on a recurrent calculation of the Taylor series terms for each time interval.

The second part of paper presents the solution of linear problems which comes from the model of telegraph line. All experiments were performed using MATLAB software, the newly developed linear solver that uses Modern Taylor Series Method. Linear solver was compared with the state of the art solvers in MATLAB and SPICE software.

**Keywords:** Telegraph line; ordinary differential equations; Taylor series method; MATLAB; SPICE

## 1 Introduction

The second order partial differential equations (PDEs) and especially the linear ones (elliptic, hyperbolic and parabolic) are very important in practical applications [1, 2]. One of these applications includes the telegraph equation, which is a PDE, that describes a telegraph line (a long wire that serves as a transmission medium for a signal). The PDE can describe the behavior of the signal, however, this description does not contain any specific information about the conditions on the wire and it is very complicated. This paper presents a numerical model of the telegraph line, that

consists only of ordinary differential equations (ODEs), it is relatively simple, easy to configure and the results match real output precisely. The system of ODEs is solved using the Modern Taylor Series Method (MTSM), which is introduced in one of the sections of the paper. To show the suitability of the MTSM to solve this kind of problem and its advantages over the other commonly used methods (Runge-Kutta, Gear), the set of experiments is performed and the results are analyzed.

This work extends the article presented at the 2017 IEEE 14th International Scientific Conference on Informatics [3].

## 2 Solution of telegraph equation

Voltage and current change along the telegraph line continuously in time and they can be expressed using equations

$$u = u(x, t), \quad (1)$$

$$i = i(x, t), \quad (2)$$

where  $x$  is a distance from the beginning of the line and  $t$  is the time. Voltage and current in the distance  $x + dx$  can be expressed using Taylor series with second and higher derivatives omitted

$$u(x + dx) = u(x, t) + \frac{\partial u}{\partial x} dx, \quad (3)$$

$$i(x + dx) = i(x, t) + \frac{\partial i}{\partial x} dx. \quad (4)$$

Basic Line Equations (5), (6) describe the change of voltage and current on the line

$$-\frac{\partial u}{\partial x} = Ri + L \frac{\partial i}{\partial t}, \quad (5)$$

$$-\frac{\partial i}{\partial x} = Gu + C \frac{\partial u}{\partial t}, \quad (6)$$

where constants  $R, G, L, C$  are parameters of line: the resistance of the wire, the conductance between wires, inductance of the wire (e.g. due to the magnetic field around the wires) and capacitance between two wires, respectively. Using (5) and (6) it is possible to construct a model of the segment. The entire line is then a series of infinite number of connected segments (Fig. 1). The line chained using this segment is lossy.

\*Corresponding Author: **Petr Veigend**: Brno University of Technology, Faculty of Information Technology, Božetěchova 2, 612 66, Brno, Czech Republic, E-mail: [iveigend@fit.vutbr.cz](mailto:iveigend@fit.vutbr.cz)

**Gabriela Nečasová, Václav Šátek**: Brno University of Technology, Faculty of Information Technology, Božetěchova 2, 612 66, Brno, Czech Republic

**Václav Šátek**: IT4Innovations, VŠB Technical University of Ostrava, 17. listopadu 15/2172, 708 33, Ostrava-Poruba, Czech Republic

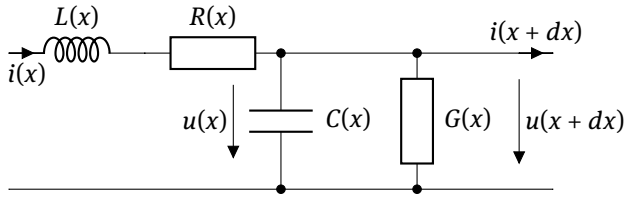


Fig. 1. Modeling a segment of the telegraph line – complex model.

The model in the Fig. 1 can be simplified by removing the terms  $R(x)$  and  $G(x)$ . The simplified model is in the Fig. 2. The line then becomes lossless.

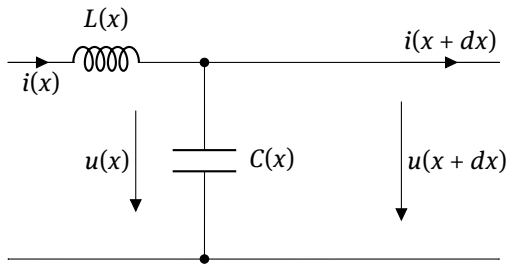


Fig. 2. Modeling a segment of the wire – simplified model.

Based on the simplified model, partial differential equations for voltage and current can be derived

$$L \cdot C \frac{\partial^2 u(x, t)}{\partial t^2} - \frac{\partial^2 u(x, t)}{\partial x^2} = 0, \quad (7)$$

$$L \cdot C \frac{\partial^2 i(x, t)}{\partial t^2} - \frac{\partial^2 i(x, t)}{\partial x^2} = 0. \quad (8)$$

The equations (7) and (8) can be solved analytically using for example the method of Separation of Variables, however the solution of large systems of PDEs is very complicated. Therefore, the simple model of telegraph equation was created. The model is described in Section 4.

### 3 Modern Taylor series method

The Modern Taylor Series Method (MTSM) is based on recurrent calculation of the Taylor series terms for each time step [4]. Therefore the complicated calculation of higher order derivatives (much criticized in the literature) does not need to be performed as the value of each term of the Taylor series is calculated numerically. An important part of the method is an automatic integration order setting, i.e. using as many Taylor series terms as the defined accuracy requires. Thus it is usual that the computation uses different numbers of Taylor series terms for different time steps of constant length.

Several papers focus on computer implementations of the Taylor series method in a variable order and variable step context, see for instance TIDES software [5], TAYLOR [6] including detailed description of a variable step size version, ATOMF [7], COSY INFINITY [8], DAETS [9]. Variable-stepsize variable-order scheme is also described in [10] and [11], where simulations on a parallel computer are shown.

The best-known and the most accurate method of calculating a new value of the numerical solution of ODE [12]

$$y' = f(t, y), \quad y(t_0) = y_0, \quad (9)$$

is to construct the Taylor series in the form

$$y_{i+1} = y_i + h \cdot f(t_i, y_i) + \frac{h^2}{2!} \cdot f'(t_i, y_i) + \dots + \frac{h^n}{n!} \cdot f^{[n-1]}(t_i, y_i), \quad (10)$$

where  $h$  is the integration step,  $y_i = y(t_i)$  is the previous value and  $y_{i+1} = y(t_i + h)$  is the next value of the function  $y(t)$ . Theoretically, it is possible to compute the solution of homogeneous linear differential equations with constant coefficients with arbitrary order and with arbitrary accuracy. Let us denote the *ORD* as the function which changes during the computation and defines the number of Taylor series terms in the current integration step ( $ORD_{i+1} = n$ ). The resulting system of linear equations can be effectively solved either sequentially or in parallel.

For linear systems of ODEs, the equation (10) can be rewritten in matrix/vector notation

$$y_{i+1} = y_i + h(\mathbf{A}y_i + \mathbf{b}) + \frac{h^2}{2!}\mathbf{A}(\mathbf{A}y_i + \mathbf{b}) + \dots + \frac{h^n}{n!}\mathbf{A}^{(n-1)}(\mathbf{A}y_i + \mathbf{b}), \quad (11)$$

where  $\mathbf{A}$  is the constant Jacobian matrix and  $\mathbf{b}$  is the constant right-hand side. Moreover, (11) can be rewritten in the form

$$y_{i+1} = DY_0 + DY_1 + DY_2 + \dots + DY_n, \quad (12)$$

where Taylor series terms could be computed recurrently using (13).

$$DY_0 = y_i, \quad DY_1 = h(\mathbf{A}y_i + \mathbf{b}), \quad (13)$$

$$DY_j = \frac{h}{j} \mathbf{A}DY_{j-1}, \quad j = 2, \dots, n$$

The model of the telegraph line presented in this article can be effectively solved by a system of linear ODEs.

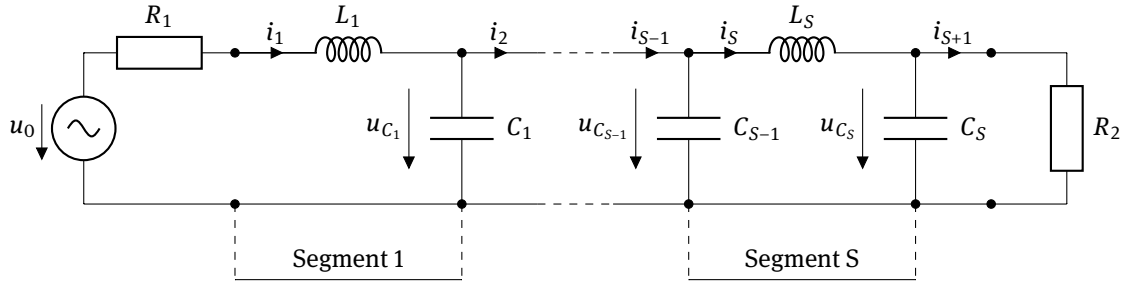


Fig. 3. Model of the line – series of  $S$  segments.

## 4 Telegraph equation model

As stated in Section 2, the model used in the experiments is the simplified model of the telegraph line (segment in Fig. 2). By chaining these segments together, we can create a lossless model of the line (Fig. 3). Let us denote the number of segments of telegraph line as  $S$ .

The equations describing the model are below. For the first segment

$$\begin{aligned} u'_{C_1} &= \frac{1}{C_1}(i_1 - i_2), \\ i'_1 &= \frac{1}{L_1}(u_0 - u_{C_1} - R_1 \cdot i_1), \end{aligned} \quad (14)$$

where  $u_0$  is the input voltage of the system,  $u_{C_1}$  is the voltage on the first capacitor and  $i_1$  is the current that flows through the first inductor. Resistor  $R_1$  represents input load of the transmission line. Equations for the following segments are very similar to one another. For the second segment

$$\begin{aligned} u'_{C_2} &= \frac{1}{C_2}(i_2 - i_3), \\ i'_2 &= \frac{1}{L_2}(u_{C_1} - u_{C_2}), \end{aligned} \quad (15)$$

for the next segments

$$\begin{aligned} u'_{C_k} &= \frac{1}{C_k}(i_k - i_{k+1}), \\ i'_k &= \frac{1}{L_k}(u_{C_{k-1}} - u_{C_k}), \end{aligned} \quad (16)$$

where  $k \in \langle 3, S \rangle$ . The last segment of the line ends with an output load, simulated by the resistor  $R_2$

$$i_{S+1} = \frac{1}{R_2} u_{C_S}. \quad (17)$$

Note that all differential equations have initial conditions equal to zero.

The model can be represented as the linear system of ODEs in the matrix/vector notation

$$\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}, \quad \mathbf{y}(0) = \mathbf{y}_0, \quad (18)$$

where  $\mathbf{A}$  is the sparse matrix,  $\mathbf{y}$  is the vector of voltages and currents and  $\mathbf{b}$  is the vector of constants. The block structure of matrix  $\mathbf{A}$ , vectors  $\mathbf{y}$  and  $\mathbf{b}$  is

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} u_{C_1} \\ \vdots \\ u_{C_S} \\ i_1 \\ \vdots \\ i_S \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{u_0}{L_1} \\ \vdots \\ 0 \end{pmatrix}, \quad (19)$$

where  $\mathbf{A}_{11}$ ,  $\mathbf{A}_{12}$ ,  $\mathbf{A}_{21}$  and  $\mathbf{A}_{22}$  are individual block matrices with size  $S \times S$

$$\begin{aligned} \mathbf{A}_{11} &= \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{-1}{R_2 C_S} \end{pmatrix} \\ \mathbf{A}_{12} &= \begin{pmatrix} \frac{1}{C_1} & \frac{-1}{C_1} & 0 & \cdots & \cdots & 0 \\ 0 & \frac{1}{C_2} & \frac{-1}{C_2} & 0 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \frac{1}{C_S} \end{pmatrix} \\ \mathbf{A}_{21} &= \begin{pmatrix} \frac{-1}{L_1} & 0 & 0 & \cdots & \cdots & 0 \\ \frac{1}{L_2} & \frac{-1}{L_2} & 0 & 0 & \cdots & \vdots \\ 0 & \frac{1}{L_3} & \frac{-1}{L_3} & 0 & \cdots & \vdots \\ 0 & \cdots & \cdots & \cdots & \frac{1}{L_S} & \frac{-1}{L_S} \end{pmatrix} \\ \mathbf{A}_{22} &= \begin{pmatrix} \frac{-R_1}{L_1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}. \end{aligned}$$

For our simulation experiments, the capacitances and inductances are the same,  $C_1 = C_2 = \cdots = C_S = 1$  pF and  $L_1 = L_2 = \cdots = L_S = 10$  nH (homogeneous lossless telegraph line).

The input voltage  $u_0$  should be generally constant (DC circuit) or harmonic (AC circuit) signal. In the case of DC circuit the input voltage  $u_0$  is hidden in constant right hand

side **b**, see (19). In the case of AC circuit the input voltage  $u_0 = U_0 \sin(\omega t)$  can be computed using auxiliary system of coupled linear ODEs

$$\begin{aligned} u_0' &= \omega x, & u_0(0) &= 0 \\ x' &= -\omega u_0, & x(0) &= U_0. \end{aligned} \tag{20}$$

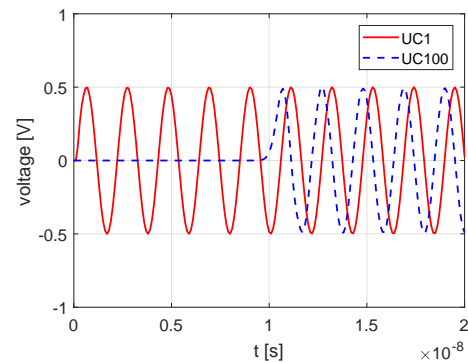
For our experiments, we used firstly harmonic input signal  $u_0 = \sin(\omega t)$ , where the angular velocity  $\omega = 3 \cdot 10^9$  rad/s, and secondly impulse input voltage which is defined as

$$\text{if}(t < \frac{\pi}{\omega}) \quad u_0 = \sin(\omega t) \quad \text{else} \quad u_0 = 0.$$

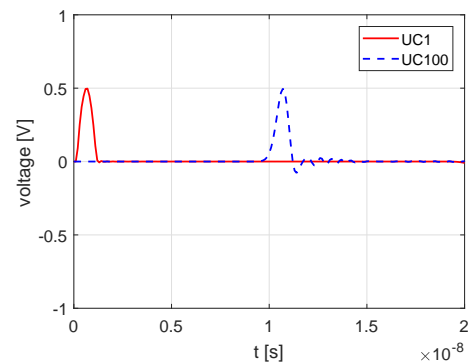
The behavior of the transmission on the line was studied. This behavior is based on the values of the input ( $R_1$ ) and the output ( $R_2$ ) loads. If the condition

$$R_1 = R_2 = \sqrt{L/C} \tag{21}$$

holds, the line is lossless and signal on the line is transmitted without change. For our simulation experiments, the line is adjusted for  $R_1 = R_2 = 100 \Omega$ . The behavior of the signal for the harmonic input and the impulse input on the line comprised of 100 segments ( $S = 100$ ) can be seen in the Fig. 4.



(a) Harmonic input

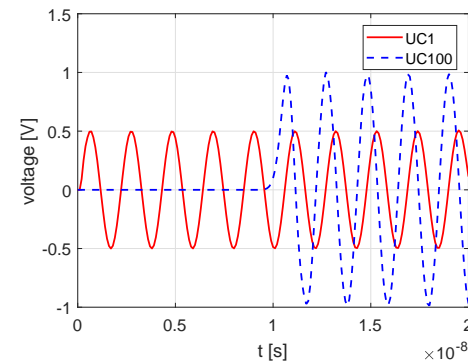


(b) Impulse input

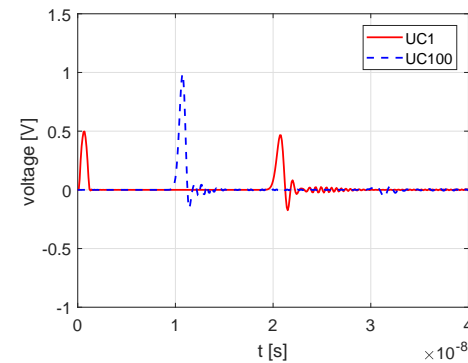
Fig. 4. Adjusted line – output just delayed.

The propagation constant per unit length of one segment for the used model is known  $t_{LC} = \sqrt{LC}$ . Then the total delay of input signal can be computed as  $t_{delay} = S t_{LC}$ . The simulation time for all experiments was set  $t_{max} = 2 t_{delay}$ . For  $S = 100$ , and used  $L$  and  $C$ , the output signal is delayed by  $100 \cdot \sqrt{10^{-8} \cdot 10^{-12}} = 10^{-8}$  seconds.

The next experiment was done for not adjusted line, where (21) doesn't hold (for example open-circuit with parameters  $R_1 = 100 \Omega$ ,  $R_2 = 10^{12} \Omega$ ). The behavior of such telegraph line (with  $S = 100$ ) could be seen in the Fig. 5.



(a) Harmonic input



(b) Impulse input

Fig. 5. Not adjusted line – output delayed and amplified.

Notice, that in the Fig. 5b, the input signal is amplified on the output and then repeated back. This can be useful to determine where the line is cut.

## 5 Numerical experiments

This section presents a set of experiments with the model from the Section 4. The experiments are performed using the traditional (MATLAB and SPICE software) and non-traditional approaches (new implementation of MTSM in

**Table 1.** MATLAB time of solutions: MTSM with fixed integration time step  $h = 5 \cdot 10^{-10}$ ; harmonic input.

| <i>S</i> | Adjusted line         |                       |                        |               | Not adjusted line     |                       |                        |               |
|----------|-----------------------|-----------------------|------------------------|---------------|-----------------------|-----------------------|------------------------|---------------|
|          | ode23<br><i>ratio</i> | ode45<br><i>ratio</i> | ode113<br><i>ratio</i> | expTay<br>[s] | ode23<br><i>ratio</i> | ode45<br><i>ratio</i> | ode113<br><i>ratio</i> | expTay<br>[s] |
| 200      | 29.5                  | 12.1                  | 9.7                    | 0.035         | 28.6                  | 10.8                  | 9.4                    | 0.035         |
| 600      | 17.3                  | 9.3                   | 6.5                    | 0.211         | 18.2                  | 9.4                   | 6.5                    | 0.21          |
| 1000     | 10.3                  | 5.1                   | 2.8                    | 0.895         | 10.3                  | 4.7                   | 2.8                    | 0.892         |
| 1400     | 7.9                   | 3.9                   | 2.1                    | 2.049         | 8.1                   | 3.8                   | 2.1                    | 2.051         |
| 1800     | 5.9                   | 3.2                   | 1.8                    | 3.808         | 6.5                   | 3.8                   | 1.8                    | 3.833         |

**Table 2.** MATLAB time of solutions: MTSM with fixed number of steps  $t_{max}/h = 200$ ; harmonic input.

| <i>S</i> | Adjusted line         |                       |                        |               | Not adjusted line     |                       |                        |               | expTay<br><i>ORD</i> |
|----------|-----------------------|-----------------------|------------------------|---------------|-----------------------|-----------------------|------------------------|---------------|----------------------|
|          | ode23<br><i>ratio</i> | ode45<br><i>ratio</i> | ode113<br><i>ratio</i> | expTay<br>[s] | ode23<br><i>ratio</i> | ode45<br><i>ratio</i> | ode113<br><i>ratio</i> | expTay<br>[s] |                      |
| 200      | 18.6                  | 7.3                   | 6.4                    | 0.053         | 19.5                  | 7.4                   | 6.4                    | 0.053         | 20                   |
| 600      | 19.03                 | 10.7                  | 7.1                    | 0.196         | 19.8                  | 10.8                  | 7.1                    | 0.196         | 41                   |
| 1000     | 16.3                  | 7.9                   | 4.5                    | 0.565         | 17.0                  | 7.9                   | 4.4                    | 0.567         | 62                   |
| 1400     | 15.9                  | 7.6                   | 4.2                    | 1.052         | 16.4                  | 7.7                   | 4.1                    | 1.056         | 84                   |
| 1800     | 9.1                   | 6.1                   | 2.7                    | 2.596         | 9.9                   | 7.8                   | 2.8                    | 2.596         | 104                  |

MATLAB). The aim is to compare the efficiency of MTSM and other available methods and highlight the suitability of MTSM for solving these kinds of problems coming from technical practice [13, 14].

All codes are implemented in MATLAB 2015a and the computations are performed by the SALOMON supercomputer at VŠB-TU Ostrava [15]. Experiments in SPICE software were performed on 4 core CPU clocked at 3.2 GHz with 8 GB of RAM using LTSpice XVII [16].

Maximum time of all numerical experiments was set to  $t_{max} = 2 t_{delay}$ , numerical error for all calculations was set to  $10^{-7}$ .

## 5.1 MATLAB

MATLAB is the language of scientific computing [17]. It contains the full suite of tools that can be used to solve various engineering and mathematical problems. Experiments in this paper use the ODE solvers, that are bundled with MATLAB. The solvers used in the paper are based on the Runge-Kutta method (solvers ode23 and ode45) [18]. Solver ode45 is based on an explicit Runge-Kutta (4,5) formula, solver ode23 on an explicit Runge-Kutta (2,3) formula. Both are single-step solvers. Solver ode113 is a variable-step, variable-order (VSVO) Adams-Bashforth-Moulton PECE solver of orders 1 to 13. The highest order used appears to be 12, however, a formula of order 13 is

used to form the error estimate and the function does local extrapolation to advance the integration at order 13.

Vectorized MATLAB code of explicit Taylor series **expTay** with a variable order and variable step size scheme for linear systems of ODEs (18) has been implemented. This algorithm was tested on a set of examples of telegraph line with different number of segments *S*. The MTSM was compared with vectorized MATLAB explicit **ode** solvers. Both relative and absolute tolerances for all solvers were set to  $10^{-7}$  for all experiments.

Benchmark results for MTSM with fixed integration step size  $h = 5 \cdot 10^{-10}$  are shown in Tables 1, 3. The numerical results for all experiments are similar and shows the stability of MTSM during the different computations. The results of computations for line with harmonic input and fixed number of integration steps  $t_{max}/h = 200$  are shown in Table 2. Other experiments (line with impulse input) will be again similar. Ratios of computation times ( $ratio := \mathbf{ode}/\mathbf{expTay} > 1$ ) indicate faster computation of the MTSM in all cases. Each reported runtime is taken as a median value of 100 computations. The fastest solution from MATLAB solvers is obtained by ode113 solvers that use higher order method.

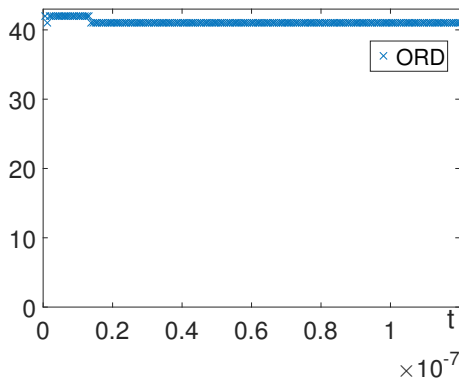
For the linearity and non-stiffness of the problem the *ORD* function was oscillating  $35 \pm 2$  during the computation (for experiments with fixed integration step size  $h = 5 \cdot 10^{-5}$ ). The average order of the MTSM ( $mean(ORD)$ ) for experiments with fixed number of integration steps ( $t_{max}/h = 200$ ) is shown in the last column of Table 2. The

**Table 3.** MATLAB time of solutions: MTSM with fixed integration time step  $h = 5 \cdot 10^{-10}$ ; impulse input.

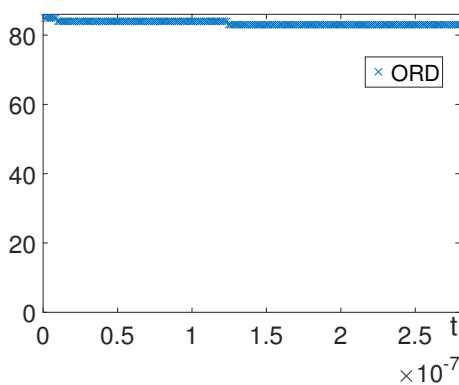
| <i>S</i> | Adjusted line         |                       |                        |               | Not adjusted line     |                       |                        |               |
|----------|-----------------------|-----------------------|------------------------|---------------|-----------------------|-----------------------|------------------------|---------------|
|          | ode23<br><i>ratio</i> | ode45<br><i>ratio</i> | ode113<br><i>ratio</i> | expTay<br>[s] | ode23<br><i>ratio</i> | ode45<br><i>ratio</i> | ode113<br><i>ratio</i> | expTay<br>[s] |
| 200      | 28.1                  | 11.6                  | 9.3                    | 0.036         | 30.3                  | 11.7                  | 9.3                    | 0.036         |
| 600      | 16.5                  | 9.3                   | 6.4                    | 0.214         | 17.6                  | 9.5                   | 6.4                    | 0.213         |
| 1000     | 8.7                   | 4.5                   | 2.9                    | 0.907         | 9.6                   | 4.7                   | 3.0                    | 0.907         |
| 1400     | 6.9                   | 3.6                   | 2.3                    | 2.075         | 7.4                   | 3.7                   | 2.2                    | 2.072         |
| 1800     | 5.12                  | 2.9                   | 1.9                    | 3.833         | 5.4                   | 2.9                   | 1.9                    | 3.847         |

**Table 4.** MATLAB number of steps: harmonic input.

| <i>S</i> | Adjusted line |       |        | Not adjusted line |       |        | expTay |
|----------|---------------|-------|--------|-------------------|-------|--------|--------|
|          | ode23         | ode45 | ode113 | ode23             | ode45 | ode113 |        |
| 200      | 9553          | 6924  | 2685   | 10017             | 6984  | 2685   | 80     |
| 600      | 26262         | 19944 | 8031   | 27479             | 20036 | 8031   | 240    |
| 1000     | 42260         | 32836 | 11804  | 44248             | 32964 | 11804  | 400    |
| 1400     | 57972         | 45656 | 15878  | 60755             | 45824 | 15878  | 560    |
| 1800     | 73465         | 58452 | 19952  | 77061             | 58592 | 19952  | 720    |



(a)  $S = 600$



(b)  $S = 1400$

**Fig. 6.** Adjusted line with harmonic input: MTSM order.

order of the MTSM for adjusted line with harmonic input and 600 segments is depicted in the Fig. 6a and for 1400 segments in Fig. 6b.

The number of integration steps needed during the computation could be seen in Tables 4, 5 for harmonic and impulse input signal, respectively. The number of integration steps is strictly linear with growing size of the ODEs (the number of segments  $S$  in transmission line).

More comparisons of MTSM numerical solutions of linear ODEs systems and some ideas of parallel computation could be found in [19].

## 5.2 SPICE

Content of this section is mostly from [20]. SPICE is used for analogue circuit simulation because it can compute the full large signal behavior of arbitrary circuits. SPICE uses three numerical methods for numerical integration. Newton integration is suitable to find the solution of circuits with non-linear elements. The sparse matrix method is used to save memory by storing only non-zero elements of the matrices. Implicit integration method is used to integrate the differential equations that describe the circuit.

Numerical integration of differential equations is necessary for analogue circuit simulation. SPICE uses second order integration. Most SPICE implementations follow Berkeley SPICE and provide two forms of second order implicit integration: Gear and Trapezoidal (trap). Trap integration is both faster and more accurate than Gear. However trap inte-



**Table 5.** MATLAB number of steps: impulse input.

| S    | Adjusted line |       |        | Not adjusted line |       |        | expTay |
|------|---------------|-------|--------|-------------------|-------|--------|--------|
|      | ode23         | ode45 | ode113 | ode23             | ode45 | ode113 |        |
| 200  | 9943          | 7552  | 2688   | 10589             | 7656  | 2688   | 80     |
| 600  | 25232         | 20404 | 8017   | 26983             | 20716 | 8017   | 240    |
| 1000 | 38668         | 32340 | 12603  | 41449             | 32860 | 12878  | 400    |
| 1400 | 51063         | 43776 | 17367  | 54861             | 44508 | 17367  | 560    |
| 1800 | 62696         | 54868 | 21879  | 67533             | 55804 | 21879  | 720    |

**Table 6.** SPICE time of solutions and number of steps: not adjusted line.

| S    | Time of solutions [s] |       |               |       | Number of steps |       |               |       |
|------|-----------------------|-------|---------------|-------|-----------------|-------|---------------|-------|
|      | Harmonic input        |       | Impulse input |       | Harmonic input  |       | Impulse input |       |
|      | modTrap               | gear  | modTrap       | gear  | modTrap         | gear  | modTrap       | gear  |
| 200  | 1.547                 | 2.427 | 2.593         | 3.812 | 1903            | 1903  | 2974          | 2921  |
| 600  | 15.0                  | 21.7  | 16.29         | 26.58 | 6045            | 6089  | 6866          | 6842  |
| 1000 | 39.33                 | 70.92 | 40.99         | 65.63 | 9543            | 9634  | 10122         | 10146 |
| 1400 | 141.9                 | 157.7 | 81.92         | 166.5 | 13287           | 13340 | 13176         | 13172 |
| 1800 | 125.3                 | 291.3 | 180.4         | 298.5 | 17903           | 17997 | 16163         | 16025 |

gration can cause a numerical artifacts. These artifacts manifest like the oscillation around the precise solution in each time step. LTSpice, which was used for the experiments, also provides Modified Trap method, which solves the problem with numerical artifacts. All of the experiments were performed using Gear and modified Trapezoidal method (denoted as modTrap in the tables).

The model for SPICE consists of components (models) for resistors, capacitors, inductors and input voltage (sine wave). The input voltage has the same characteristics as in the previous sections of the paper. For the harmonic input, the SINE component has the following parameters (0  $U_0 \omega/(2\pi)$  0 0 0 0) and for the impulse input (0  $U_0 \omega/(2\pi)$  0 0 0 0.5).

The description of the rest of the components is straight forward. The abbreviated version of the netlist (the description of the circuit) for the model with 100 segments ( $S = 100$ ) for not adjusted line with  $R_1 = 100 \Omega$ ,  $R_2 = 10^{12} \Omega$  and with the harmonic input where  $\omega = 3 \cdot 10^9$  rad/s could be seen in Listing 1. The model corresponds with the electrical circuit in Fig. 3.

```
V1 1 0 SINE(0 1 477464829.275686 0 0 0 0)
R1 1 2 100.0
C1 2 0 1e-12
L1 2 3 1e-08
C2 3 0 1e-12
L2 3 4 1e-08
...
C99 100 0 1e-12
L99 100 101 1e-08
C100 101 0 1e-12
L100 101 102 1e-08
R100 102 0 1000000000000.0
```

**Listing 1.** SPICE source code.

The tolerances of all solvers were set to  $10^{-7}$ . The result of the simulation for not adjusted line with 100 segments is shown in Fig. 7. The output matches the result obtained by the MATLAB solvers (see Fig. 5).

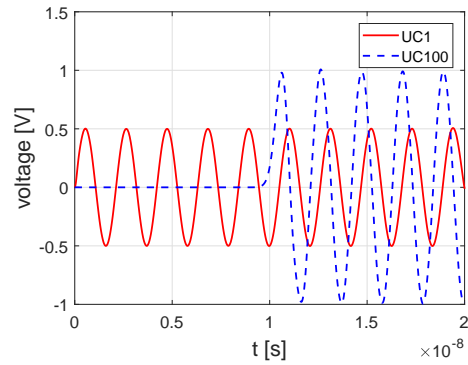
The results of the performed experiments for not adjusted line are summarized in Table 6.

The experiments for adjusted line show similar behavior. When comparing to the MTSM solver implemented in MATLAB, the time of solution is much longer. It is caused by the lack of vectorization, that is used in MATLAB or implicit schemes of methods implemented in SPICE. The time of transformation the set problem from netlist scheme (see Listing 1) to inner matrix/vector representation is also time consuming.

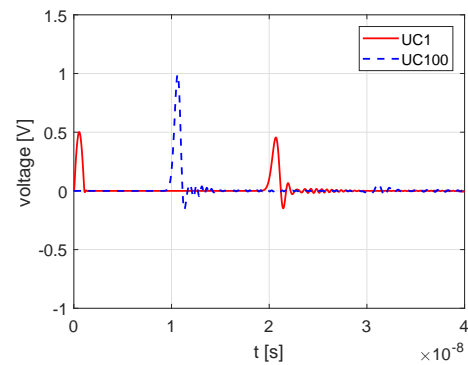
## 6 Conclusion

This article dealt with the numerical analysis of the second order partial differential equation – Telegraph equation. This equation was replaced by the segment model and described by the system of ordinary differential equations (ODEs). The system has been solved numerically using MTSM and other methods for comparison.

As an example, the model with different number of segments (different length of the line) was presented. The experiments showed the behavior of the signal transmission for the specific combination of input/output loads (parameters  $R_1$  and  $R_2$ ) and different input signal (harmonic



(a) Harmonic input



(b) Impulse input

Fig. 7. SPICE not adjusted line – output delayed and amplified.

or impulse). The experiments are summarized and clearly show the suitability of the MTSM for this kind of problem. The MTSM was compared with the state of the art numerical solvers implemented in MATLAB and SPICE software.

Due to the fact that the Telegraph equation is the special case of the wave equation, it might be possible to slightly change the presented model to represent much wider class of problems, which is an interesting topic for future research. The problem is easily scalable so different approaches to parallelization can also be researched especially for larger systems of ODEs.

**Acknowledgment:** This work was supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPS II) project “IT4Innovations excellence in science - LQ1602”. The paper also includes the solution results of the internal BUT project FIT-S-17-4014.

We would like to dedicate this article to our friend and mentor Jiří Kunovský, that passed away recently.

## References

- [1] Farlow S. J., Partial differential equations for scientists and engineers, Courier Corporation, 2012, ISBN 860-1234581253
- [2] Evans L. C., Partial differential equations; 2nd ed., Graduate Studies in Mathematics, American Mathematical Society, Providence, RI, 2010, ISBN 978-0821849743
- [3] Nečasová G., Kocina F., Veigend P., Šátek V., Kunovský J., Model of the Telegraph Line, In: Informatics 2017 - 14th International Scientific Conference on Informatics, Institute of Electrical and Electronics Engineers, 2017, 1–10
- [4] Kunovský J., Modern Taylor Series Method, Habilitation thesis, FEKT VUT Brno, 1995
- [5] Rodríguez M., Medina A. A., Gil R. B., Blesa F., Tides: A free software based on the Taylor series method, Monografías de la Real Academia de Ciencias Exactas, Físicas, Químicas y Naturales de Zaragoza, 2011, 35, 83–95
- [6] Jorba A., Zou M., A software package for the numerical integration of ODE by means of high-order Taylor methods, Experimental Mathematics, 2005, 14, 99–117
- [7] Chang Y., Corliss G., ATOMFT: solving ODEs and DAEs using Taylor series, Computers & Mathematics with Applications, 1994, 28(10-12), 209–233
- [8] Berz M., Makino K., COSY INFINITY Version 8.1. User’s Guide and Reference Manual, Department of Physics and Astronomy MSUHEP-20704, Michigan State University, 2002
- [9] Nedialkov N. S., Pryce J. D., Solving differential-algebraic equations by Taylor series (III): The DAETS code, Journal of Numerical Analysis, Industrial and Applied Mathematics, 2008, 3, 61–80
- [10] Barrio R., Blesa F., Lara M., VSVO Formulation of the Taylor Method for the Numerical Solution of ODEs, Computers and Mathematics with Applications, 2005, 50, 93–111
- [11] Barrio R., Performance of the Taylor series method for ODEs/-DAEs, In Applied Mathematics and Computation, 2005, 163, 525–545, ISSN 00963003
- [12] Hairer E., Wanner G., Solving ordinary differential equations, In: Nonstiff problems, Springer-Verlag, 2000
- [13] Kunovský J., Šátek V., Szöllös A., Telegraph equation and corresponding wave forms, In: Proceedings of the 12th International Scientific Conference Electric Power Engineering (EPE 2011), 2011, 561–564
- [14] Kunovský J., Šátek V., Topolánek D., Vopěnka V., Telegraph equation and its application in medium voltage line, In: Proceedings of the 13th International Scientific Conference Electric Power Engineering (EPE 2012), 2012, 1, 175–180
- [15] MATLAB, IT4Innovations documentation, 2018
- [16] LINEAR, LTSPICE, 2018
- [17] MathWorks, MATLAB, 2018
- [18] Butcher J.C., Numerical Methods for Ordinary Differential Equations, John Wiley & Sons, 2008
- [19] Šátek V., Kocina F., Kunovský J., Schirrer A., Taylor Series Based Solution of Linear ODE Systems and MATLAB Solvers Comparison, In: MATHMOD VIENNA 2015 - 8th Vienna Conference on Mathematical Modelling, ARGESIM Report No. 44, ARGE Simulation News, 2015, 693–694
- [20] Engelhardt M., SPICE Differentiation, Electronics World, 2015, 121(1946), 16–21