

# telegraph equations solved with MFEM library using MFEM

Denis Lachapelle

May 2025

## 1 Introduction

This document explains the telegraph equations simulation using finite difference method based on MFEM library.

## 2 Theory

### 2.1 From Telegraph PDE to Matrix form

The telegraph equations are:

$$\frac{\partial V}{\partial x} + L \frac{\partial I}{\partial t} + RI = 0 \quad (1)$$

$$\frac{\partial I}{\partial x} + C \frac{\partial V}{\partial t} + GV = 0 \quad (2)$$

For now assumes a single line from points a to b.

Going to weak form using different space for V and I,  $\phi$  for V and  $\psi$  for I. Note the test functions span the entire domain and is zero on the border and the equations below have to be true for all test functions. Look at "element\_finis.pdf" p. 42.

$$\int_{\Omega} \left( \frac{\partial V}{\partial x} + L \frac{\partial I}{\partial t} + RI \right) \phi = 0 \quad (3)$$

$$\int_{\Omega} \left( \frac{\partial I}{\partial x} + C \frac{\partial V}{\partial t} + GV \right) \psi = 0 \quad (4)$$

and then

$$\int_{\Omega} \frac{\partial V}{\partial x} \phi + \int_{\Omega} \left( L \frac{\partial I}{\partial t} + RI \right) \phi = 0 \quad (5)$$

$$\int_{\Omega} \frac{\partial I}{\partial x} \psi + \int_{\Omega} \left( C \frac{\partial V}{\partial t} + GV \right) \psi = 0 \quad (6)$$

Using integration by part on the first term...

$$\int_a^b u(x) v'(x) dx = \left[ u(x) v(x) \right]_a^b - \int_a^b u'(x) v(x) dx \quad (7)$$

Each equations becomes ...

$$\left[ \phi V \right]_a^b - \int_{\Omega} \left( V \frac{\partial \phi}{\partial x} \right) + \int_{\Omega} \left( L \frac{\partial I}{\partial t} + RI \right) \phi = 0 \quad (8)$$

$$\left[ \psi I \right]_a^b - \int_{\Omega} \left( I \frac{\partial \psi}{\partial x} \right) + \int_{\Omega} \left( C \frac{\partial V}{\partial t} + GV \right) \psi = 0 \quad (9)$$

Then we approximate by different basis function for V and I, V(x) by

$$V(x) = \sum_j V_j \phi(x) \quad (10)$$

and I(x) by

$$I(x) = \sum_j I_j \psi(x) \quad (11)$$

Where  $I_j$  and  $V_j$  are constants and notice that  $\left[ \phi V \right]_a^b$  and  $\left[ \psi I \right]_a^b$  are zero since  $\phi$  and  $\psi$  are zero on the border.

Equations (8) becomes ...

$$\int_{\Omega_i} \sum_j V_j \phi_j \frac{\partial \phi_i}{\partial x} - L \int_{\Omega_i} \frac{\partial}{\partial t} \left[ \sum_j I_j \psi_j \right] \phi_i - R \int_{\Omega_i} \sum_j I_j \psi_j \phi_i = 0 \quad (12)$$

Assume  $\frac{\partial I}{\partial t}$  is constant on each element so we can exclude  $\frac{\partial I}{\partial t}$  from inside the integral and interchange the integral and summation.

$$\sum_j V_j \int_{\Omega_i} \phi_j \frac{\partial \phi_i}{\partial x} - L \sum_j \frac{\partial I_j}{\partial t} \int_{\Omega_i} \psi_j \phi_i - R \sum_j I_j \int_{\Omega_i} \psi_j \phi_i = 0 \quad (13)$$

and equation 9 becomes ...

$$\sum_j I_j \int_{\Omega} \psi_j \frac{\partial \psi_i}{\partial x} - C \sum_j \frac{\partial V_j}{\partial t} \int_{\Omega} \phi_j \psi_i - G \sum_j V_j \int_{\Omega} \phi_j \psi_i = 0 \quad (14)$$

So now we have two equations (13) and (14) with unknown  $V_j$  and  $I_j$ .

The  $j$  span the trial functions (Approximating Functions) and the  $i$  span the elements. The equations can be converted in matrix form...

$$S_V V - L M_V \frac{\partial I}{\partial t} - R M_V I == 0 \quad (15)$$

$$S_I I - C M_I \frac{\partial V}{\partial t} - G M_I V == 0 \quad (16)$$

Where ...

$$S_V = \int_{\Omega} \phi_j \frac{\partial \phi_i}{\partial x}$$

dimensions VDOFxVDOF

$$M_V = \int_{\Omega} \psi_j \phi_i$$

dimension VDOFxIDOF

$$S_I = \int_{\Omega} \psi_j \frac{\partial \psi_i}{\partial x}$$

dimension IDOFxIDOF

$$M_I = \int_{\Omega} \phi_j \psi_i$$

dimension IDOFxVDOF

Note that  $S_V = S_I^T$ .

In equations 13 and 14 we have  $\frac{\partial V}{\partial t}$  and  $\frac{\partial I}{\partial t}$  that can each be approximated by  $\frac{I_{n+1}-I_n}{\Delta t}$ . This is the forward Euler method (explicit Euler method).

$$S_V V_n - LM_V \left( \frac{I_{n+1} - I_n}{\Delta t} \right) - RM_V I_n = 0 \quad (17)$$

$$S_I I_n - CM_I \left( \frac{V_{n+1} - V_n}{\Delta t} \right) - GM_I V_n == 0 \quad (18)$$

Now with  $V_{n+1}$  and  $I_{n+1}$  on the left side.

$$V_{n+1} = V_n + \frac{1}{C} M_I^{-1} S_I I_n \Delta t - \frac{G}{C} V_n \Delta t \quad (19)$$

$$I_{n+1} = I_n + \frac{1}{L} M_V^{-1} S_V V_n \Delta t - \frac{R}{L} I_n \Delta t \quad (20)$$

The above solution is not appropriated since  $M_V$  and  $M_I$  are not square and thus cannot be inverted.

Note they are not square because the trial and test functions are not the same (not so sure here).

## 2.2 another way without matrix inversion

Starting from equation 17 and 18.

$$S_V V_n - \frac{LM_V}{\Delta t} I_{n+1} + \frac{LM_V}{\Delta t} I_n - RM_V I_n = 0 \quad (21)$$

$$S_I I_n - \frac{CM_I}{\Delta t} V_{n+1} + \frac{CM_I}{\Delta t} V_n - GM_I V_n == 0 \quad (22)$$

$$\frac{LM_V}{\Delta t} I_{n+1} = \frac{LM_V}{\Delta t} I_n - RM_V I_n + S_V V_n \quad (23)$$

$$\frac{CM_I}{\Delta t} V_{n+1} = \frac{CM_I}{\Delta t} V_n - GM_I V_n + S_I I_n \quad (24)$$

$$M_V I_{n+1} = M_V I_n - \frac{R\Delta t}{L} M_V I_n + \frac{\Delta t}{L} S_V V_n \quad (25)$$

$$M_IV_{n+1} = M_IV_n - \frac{G\Delta t}{C}M_IV_n + \frac{\Delta t}{C}S_II_n \quad (26)$$

$$M_VI_{n+1} = (1 - \frac{R\Delta t}{L})M_VI_n + \frac{\Delta t}{L}S_VV_n \quad (27)$$

$$M_IV_{n+1} = (1 - \frac{G\Delta t}{C})M_IV_n + \frac{\Delta t}{C}S_II_n \quad (28)$$

### 2.3 Boundary

Now we have a voltage source  $V_S$  driving transmission line point a through a resistor of  $R_S$  ohm.

$$R_S I_a + V_a = V_S \quad (29)$$

The transmission line point b is loaded with a resistor  $R_L$ .

$$V_b - R_L I_b = 0 \quad (30)$$

### 2.4 The complete Matrix System

$$V_{a,n+1} = V_S - R_S I_{a,n} \quad (31)$$

$$V_{b,n+1} = R_L I_{b,n} \quad (32)$$

The system is a block matrix

$$\begin{bmatrix} M_I & 0 \\ 0 & M_V \end{bmatrix} \begin{bmatrix} V_{n+1} \\ I_{n+1} \end{bmatrix} = \begin{bmatrix} K3M_I & K4S_I \\ K1S_V & K2M_V \end{bmatrix} \begin{bmatrix} V_n \\ I_n \end{bmatrix} \quad (33)$$

Where ...

$$\begin{aligned} K1 &= \frac{\Delta t}{L} \\ K2 &= 1 - \frac{R\Delta t}{L} \\ K3 &= 1 - \frac{G\Delta t}{C} \end{aligned}$$

$$K4 = \frac{\Delta t}{C}$$

Here we still have to include the Vs and load.

To include  $V_S$  and  $R_S$  we should add a column  $\{1 \ 0 \ 0 \ 0 \dots 0\}^T$  on the rhs matrix and  $V_S$  to the rhs column vector and replace the first row of  $K4S_I$  by  $\{-R_S \ 0 \ 0 \ 0 \dots 0\}$  and zeroed the first row of  $K3M_I$  to match equation 31.

$$\begin{bmatrix} M_I & 0 \\ 0 & M_V \end{bmatrix} \begin{bmatrix} V_{n+1} \\ I_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & K3M_I & K4S_I \\ 0 & K1S_V & K2M_V \end{bmatrix} \begin{bmatrix} V_S \\ V_n \\ I_n \end{bmatrix} \quad (34)$$

Now to include the load  $R_L$  zeroed the last row of  $K3M_I$  and replaced the last row of  $K4S_I$  by  $\{0 \ 0 \ 0 \ 0 \dots R_L\}$ .

Now check if the size of each matrix make sense.

$$\begin{bmatrix} IxV & 0 \\ 0 & VxI \end{bmatrix} \begin{bmatrix} Vx1 \\ Ix1 \end{bmatrix} = \begin{bmatrix} 1 & IxV & IxI \\ 0 & VxV & VxI \end{bmatrix} \begin{bmatrix} 1x1 \\ Vx1 \\ Ix1 \end{bmatrix} \quad (35)$$

Once the matrix multiplication done

$$\begin{bmatrix} Ix1 \\ Vx1 \end{bmatrix} = \begin{bmatrix} Ix1 \\ Vx1 \end{bmatrix} \quad (36)$$

So it look fine since the dimensions match.

### 3 MFEM Implementation

With the theory developed above we can start implementing the program using MFEM<sup>1</sup> library.

The program is named stlt.cpp for single transmission line transient.

The program will run this way:

1. Compute the two block matrix, the lhs and the rhs.

- (a) matrice  $M_V$ ,  $M_I$ ,  $S_V$  and  $S_I$  computed.

---

<sup>1</sup><https://mfem.org/>

- (b) rhs and lhs matrices assembled as block operator.
- (c) RS and RL included in the system.
- (d) VS ???

2. compute  $V_S$  at the time  $n\Delta t$
3. compute the rhs by matrix multiplication.
4. solve the system of equations

$$[lhsOp] [xL] = [rhsOp] [xR] \quad (37)$$

$$[lhsOp] [xL] = [y] \quad (38)$$

The system is unstable... try with  $\Delta T = 0.01e-9$  instead of  $0.1 e-9$ .  
Change L, C, R and G for RG-58.

```
// Constants for the telegrapher's equation for RG-58, 50 ohm.
double L = 250e-9; // Inductance per unit length
double C = 100.0e-12; // Capacitance per unit length
double R = 10e-3; // Resistance per unit length
double G = 1.0e-9; // Conductance per unit length.
```

I find some problem in the matrix system theory, so I change the structure.

So i need to adjust the software.

The determinant of the  $[lhsOp]$  is zero and the condition number is very high, so it wont converge.

## 4 Theory, a second trial

Assuming both V and I in H1 space, this will make  $M_I$  and  $M_V$  square, it will then be possible to invert them.

## 4.1 From Telegraph PDE to Matrix form

The telegraph equations are:

$$\frac{\partial V}{\partial x} + L \frac{\partial I}{\partial t} + RI = 0 \quad (39)$$

$$\frac{\partial I}{\partial x} + C \frac{\partial V}{\partial t} + GV = 0 \quad (40)$$

For now assumes a single line from points a to b.

Going to weak form using same space for V and I. Note the test functions span the entire domain and is zero on the border and the equations below have to be true for all test functions. Look at "element\_finis.pdf" p. 42.

$$\int_{\Omega} \left( \frac{\partial V}{\partial x} + L \frac{\partial I}{\partial t} + RI \right) \phi = 0 \quad (41)$$

$$\int_{\Omega} \left( \frac{\partial I}{\partial x} + C \frac{\partial V}{\partial t} + GV \right) \phi = 0 \quad (42)$$

and then

$$\int_{\Omega} \frac{\partial V}{\partial x} \phi + \int_{\Omega} \left( L \frac{\partial I}{\partial t} + RI \right) \phi = 0 \quad (43)$$

$$\int_{\Omega} \frac{\partial I}{\partial x} \phi + \int_{\Omega} \left( C \frac{\partial V}{\partial t} + GV \right) \phi = 0 \quad (44)$$

Using integration by part on the first term...

$$\int_a^b u(x)v'(x) dx = \left[ u(x)v(x) \right]_a^b - \int_a^b u'(x)v(x) dx \quad (45)$$

Each equations becomes ...

$$\left[ \phi V \right]_a^b - \int_{\Omega} \left( V \frac{\partial \phi}{\partial x} \right) + \int_{\Omega} \left( L \frac{\partial I}{\partial t} + RI \right) \phi = 0 \quad (46)$$

$$\left[ \phi I \right]_a^b - \int_{\Omega} \left( I \frac{\partial \phi}{\partial x} \right) + \int_{\Omega} \left( C \frac{\partial V}{\partial t} + GV \right) \phi = 0 \quad (47)$$

Then we approximate V and I, V(x) by



$$V(x, t) = \sum_j V_j(t) \phi(x) \quad (48)$$

and  $I(x)$  by

$$I(x, t) = \sum_j I_j(t) \phi(x) \quad (49)$$

Notice that  $\left[\phi V\right]_a^b$  and  $\left[\phi I\right]_a^b$  are zero since  $\phi$  is zero on the border.

Equations (46) becomes ...

$$\int_{\Omega_i} \sum_j V_j \phi_j \frac{\partial \phi_i}{\partial x} - L \int_{\Omega_i} \frac{\partial}{\partial t} \left[ \sum_j I_j \phi_j \right] \phi_i - R \int_{\Omega_i} \sum_j I_j \phi_j \phi_i = 0 \quad (50)$$

Assume  $\frac{\partial I}{\partial t}$  is constant on each element so we can exclude  $\frac{\partial I}{\partial t}$  from inside the integral and interchange the integral and summation.

$$\sum_j V_j \int_{\Omega_i} \phi_j \frac{\partial \phi_i}{\partial x} - L \sum_j \frac{\partial I_j}{\partial t} \int_{\Omega_i} \phi_j \phi_i - R \sum_j I_j \int_{\Omega_i} \phi_j \phi_i = 0 \quad (51)$$

and equation 9 becomes ...

$$\sum_j I_j \int_{\Omega} \phi_j \frac{\partial \phi_i}{\partial x} - C \sum_j \frac{\partial V_j}{\partial t} \int_{\Omega} \phi_j \phi_i - G \sum_j V_j \int_{\Omega} \phi_j \phi_i = 0 \quad (52)$$

So now we have two equations (51) and (52) with unknown  $V_j(t)$  and  $I_j(t)$ .

The  $j$  span the trial functions (Approximating Functions) and the  $i$  span the elements. The equations can be converted in matrix form...

$$S_V V - L M_V \frac{\partial I}{\partial t} - R M_V I == 0 \quad (53)$$

$$S_I I - C M_I \frac{\partial V}{\partial t} - G M_I V == 0 \quad (54)$$

Where ...

$$S_V = \int_{\Omega} \phi_j \frac{\partial \phi_i}{\partial x}$$

dimensions VDOFxVDOF

$$M_V = \int_{\Omega} \phi_j \phi_i$$

dimension VDOFxIDOF

$$S_I = \int_{\Omega} \phi_j \frac{\partial \phi_i}{\partial x}$$

dimension IDOFxIDOF

$$M_I = \int_{\Omega} \phi_j \phi_i$$

dimension IDOFxVDOF

In equations 53 and 54 we have  $\frac{\partial V}{\partial t}$  and  $\frac{\partial I}{\partial t}$  that can each be approximated by  $\frac{I_{n+1}-I_n}{\Delta t}$ . This is the forward Euler method (explicit Euler method).

$$S_V V_n - LM_V \left( \frac{I_{n+1} - I_n}{\Delta t} \right) - RM_V I_n = 0 \quad (55)$$

$$S_I I_n - CM_I \left( \frac{V_{n+1} - V_n}{\Delta t} \right) - GM_I V_n == 0 \quad (56)$$

Now with  $V_{n+1}$  and  $I_{n+1}$  on the left side.

$$V_{n+1} = V_n + \frac{1}{C} M_I^{-1} S_I I_n \Delta t - \frac{G}{C} V_n \Delta t \quad (57)$$

$$I_{n+1} = I_n + \frac{1}{L} M_V^{-1} S_V V_n \Delta t - \frac{R}{L} I_n \Delta t \quad (58)$$

## 4.2 another way without matrix inversion

Starting from equation 17 and 18.

$$S_V V_n - \frac{LM_V}{\Delta t} I_{n+1} + \frac{LM_V}{\Delta t} I_n - RM_V I_n = 0 \quad (59)$$

$$S_I I_n - \frac{CM_I}{\Delta t} V_{n+1} + \frac{CM_I}{\Delta t} V_n - GM_I V_n == 0 \quad (60)$$

$$\frac{LM_V}{\Delta t}I_{n+1} = \frac{LM_V}{\Delta t}I_n - RM_VI_n + S_VV_n \quad (61)$$

$$\frac{CM_I}{\Delta t}V_{n+1} = \frac{CM_I}{\Delta t}V_n - GM_IV_n + S_II_n \quad (62)$$

$$M_VI_{n+1} = M_VI_n - \frac{R\Delta t}{L}M_VI_n + \frac{\Delta t}{L}S_VV_n \quad (63)$$

$$M_IV_{n+1} = M_IV_n - \frac{G\Delta t}{C}M_IV_n + \frac{\Delta t}{C}S_II_n \quad (64)$$

$$M_VI_{n+1} = (1 - \frac{R\Delta t}{L})M_VI_n + \frac{\Delta t}{L}S_VV_n \quad (65)$$

$$M_IV_{n+1} = (1 - \frac{G\Delta t}{C})M_IV_n + \frac{\Delta t}{C}S_II_n \quad (66)$$

These two equations are more appropriated since we will use a solver.

### 4.3 Boundary

Now we have a voltage source  $V_S$  driving transmission line point a through a resistor of  $R_S$  ohm.

$$R_S I_a + V_a = V_S \quad (67)$$

The transmission line point b is loaded with a resistor  $R_L$ .

$$V_b - R_L I_b = 0 \quad (68)$$

### 4.4 The complete Matrix System

$$V_{a,n+1} = V_S - R_S I_{a,n} \quad (69)$$

$$V_{b,n+1} = R_L I_{b,n} \quad (70)$$

The system is a block matrix

$$\begin{bmatrix} M_I & 0 \\ 0 & M_V \end{bmatrix} \begin{bmatrix} V_{n+1} \\ I_{n+1} \end{bmatrix} = \begin{bmatrix} K3M_I & K4S_I \\ K1S_V & K2M_V \end{bmatrix} \begin{bmatrix} V_n \\ I_n \end{bmatrix} \quad (71)$$

Where ...

$$\begin{aligned}
K1 &= \frac{\Delta t}{L} \\
K2 &= 1 - \frac{R\Delta t}{L} \\
K3 &= 1 - \frac{G\Delta t}{C} \\
K4 &= \frac{\Delta t}{C}
\end{aligned}$$

Here we still have to include the Vs and load.

To include  $V_S$  and  $R_S$  we should add a column  $\{1 \ 0 \ 0 \ 0 \dots 0\}^T$  on the rhs matrix and  $V_S$  to the rhs column vector and replace the first row of  $K4S_I$  by  $\{-R_S \ 0 \ 0 \ 0 \dots 0\}$  and zeroed the first row of  $K3M_I$  to match equation 31.

$$\begin{bmatrix} M_I & 0 \\ 0 & M_V \end{bmatrix} \begin{bmatrix} V_{n+1} \\ I_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & K3M_I & K4S_I \\ 0 & K1S_V & K2M_V \end{bmatrix} \begin{bmatrix} V_S \\ V_n \\ I_n \end{bmatrix} \quad (72)$$

Now to include the load  $R_L$  zeroed the last row of  $K3M_I$  and replaced the last row of  $K4S_I$  by  $\{0 \ 0 \ 0 \ 0 \dots R_L\}$ .

## 5 MFEM Implementation

With the theory developed above we can start implementing the program using MFEM<sup>2</sup> library.

The program is named stltfe.cpp for single transmission line transient forward euler.

The program will run this way:

1. Compute the two block matrix, the lhs and the rhs.
  - (a) matrice  $M_V$ ,  $M_I$ ,  $S_V$  and  $S_I$  computed.
  - (b) rhs and lhs matrices assembled as block operator.

---

<sup>2</sup><https://mfem.org/>

(c) RS and RL included in the system.

(d) VS ???

2. compute  $V_S$  at the time  $n\Delta t$
3. compute the rhs by matrix multiplication.
4. solve the system of equations

$$[lhsOp] [xL] = [rhsOp] [xR] \quad (73)$$

$$[lhsOp] [xL] = [b] \quad (74)$$

The system is unstable even if i remove the source and apply an initial condition of 1,

## 6 Theory, Finite Differences

The telegraph equations are:

$$\frac{\partial V}{\partial x} + L \frac{\partial I}{\partial t} + RI = 0 \quad (75)$$

$$\frac{\partial I}{\partial x} + C \frac{\partial V}{\partial t} + GV = 0 \quad (76)$$

1. Assume the TL is divided in N segments with node 0 to N. There is N+1 nodes.
2. Assume the  $\frac{\partial V(t,x)}{\partial x}$  and  $\frac{\partial I(t,x)}{\partial x}$  are constant over each segment.

So at a given time t we can express the  $\frac{\partial V(t,x)}{\partial x}$  and  $\frac{\partial I(t,x)}{\partial x}$  as difference  $\frac{V(t,nh)-V(t,(n-1)h)}{h}$  and  $\frac{I(t,nh)-I(t,(n-1)h)}{h}$ .

We can write matrix equations....

$$\frac{\partial I}{\partial t} = \begin{bmatrix} Dv & Ri \end{bmatrix} \begin{bmatrix} V^k \\ I^k \end{bmatrix} \quad (77)$$

$$\frac{\partial V}{\partial t} = \begin{bmatrix} Gv & Di \end{bmatrix} \begin{bmatrix} V^k \\ I^k \end{bmatrix} \quad (78)$$

The two equations above can be written as a single matrix equation...

$$\begin{bmatrix} \frac{\partial V}{\partial t} \\ \frac{\partial I}{\partial t} \end{bmatrix} = \begin{bmatrix} Gv & Di \\ Dv & Ri \end{bmatrix} \begin{bmatrix} V^k \\ I^k \end{bmatrix} \quad (79)$$

$$\begin{bmatrix} V^{k+1} \\ I^{k+1} \end{bmatrix} = \begin{bmatrix} V^k \\ I^k \end{bmatrix} + \Delta t \begin{bmatrix} \frac{\partial V}{\partial t} \\ \frac{\partial I}{\partial t} \end{bmatrix} \quad (80)$$

$$\begin{bmatrix} V^{k+1} \\ I^{k+1} \end{bmatrix} = \begin{bmatrix} V^k \\ I^k \end{bmatrix} + \Delta t \begin{bmatrix} Gv & Di \\ Dv & Ri \end{bmatrix} \begin{bmatrix} V^k \\ I^k \end{bmatrix} \quad (81)$$

Where Dv and Di are the same derivative matrix multiplied by different terms -1/L for Dv and -1/C for Di.

$$\begin{bmatrix} -1/h & 1/h & 0 & 0 & \dots & 0 & 0 \\ 0 & -1/h & 1/h & 0 & \dots & 0 & 0 \\ 0 & 0 & -1/h & 1/h & \dots & 0 & 0 \\ 0 & 0 & 0 & -1/h & 1/h & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 & -1/h & 1/h \\ 0 & 0 & 0 & \dots & 0 & -1/h & 1/h \end{bmatrix} \quad (82)$$

Gv and Ri are identity matrix scale by either -G/C and -R/L.

For Dv the multiplier value will be  $\frac{\Delta t}{Lh}$  which is in our case  $1e-12/(250e-9 \cdot 1e-3) = 0.004$

## 6.1 MFEM Implementation

The program is named stltfd.cpp for Single Transmission Line Transient Finite Difference.

No way to get the stable, signal do not make sense. Try with a step, impulse and a sine wave 1 Ghz.

I try with central difference and same problem.

I read that backward Euler and Crank-Nicolson are more stable.

Should I try with python? I have doubt about MFEM.

I made some test and the Dv and Di matrix within the block matrix works fine.

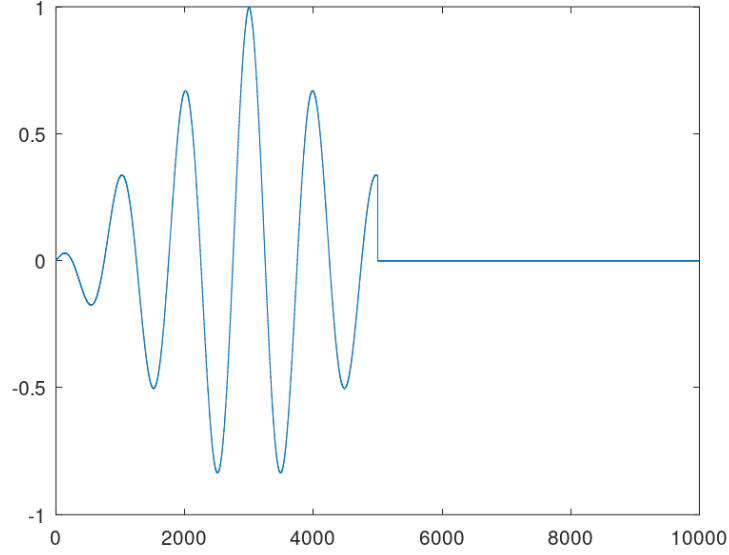


Figure 1: Test pulse

For  $Dv$  the multiplier value will be  $\frac{\Delta t}{Lh}$  which is in our case  $1e-12/(250e-9 \cdot 1e-3) = 0.004$ .

For  $Di$  the multiplier value will be  $\frac{\Delta t}{Ch}$  which is in our case  $1e-12/(100e-12 \cdot 1e-3) = 10$ .

## 7 Another way, Finite Difference Discretization

### 7.1 Discretizing the first equation

$V_i^n$  is meaning  $V_{ih}^{n\Delta t}$

$$\frac{V_{i+1}^n - V_i^n}{\Delta x} + L \frac{I_i^{n+1} - I_i^n}{\Delta t} + RI_i^n = 0 \quad (83)$$

Rearranged for  $I_i^{n+1}$ :

$$I_i^{n+1} = I_i^n - \frac{\Delta t}{L} \left( \frac{V_{i+1}^n - V_i^n}{\Delta x} + RI_i^n \right) \quad (84)$$

## 7.2 Discretizing the second equation

$$\frac{I_{i+1}^n - I_i^n}{\Delta x} + C \frac{V_i^{n+1} - V_i^n}{\Delta t} + G V_i^n = 0 \quad (85)$$

Rearranged for  $V_i^{n+1}$ :

$$V_i^{n+1} = V_i^n - \frac{\Delta t}{C} \left( \frac{I_{i+1}^n - I_i^n}{\Delta x} + G V_i^n \right) \quad (86)$$

## 8 Conclusion (for now: April 6, 2025)

I never succeed in any way; I may be confused with the expected result or the stability. I start trying with finite method and then finite difference, none succeeded.

Given my many reading, the finite element method is one of the best with the finite volume method. I suspect the code I made for finite element method using MFEM is mostly right, but I do not provide valid initial and boundary conditions and my understanding of the output was not good.

I shall also understand the notion of elliptic, parabolic and hyperbolic equation because each seems to call for a different approach.

## 9 Finite Element Method with RK4

The file name is stltferk4.cpp for single transmission transient line finite fe runge kutta 4. stltferk4.cpp is the continuation of stltfe.cpp.

MFEM provides a class for RK4 solver I'll try to use it. odesolver need like rk4solver need a time dependent operator that is used to compute the various slopes required by RK4.

The telegraph equations are:

$$\frac{\partial V}{\partial x} + L \frac{\partial I}{\partial t} + R I = 0 \quad (87)$$



$$\frac{\partial I}{\partial x} + C \frac{\partial V}{\partial t} + GV = 0 \quad (88)$$

with the  $\frac{\partial}{\partial t}$  on the left side.

$$\frac{\partial I}{\partial t} = -\frac{RI}{L} - \frac{1}{L} \frac{\partial V}{\partial x} \quad (89)$$

$$\frac{\partial V}{\partial t} = -\frac{GV}{C} - \frac{1}{C} \frac{\partial I}{\partial x} \quad (90)$$

With this formulation it look easier to use finite difference....let give it a try in the same file using option.

The following search on google "transmission line simulation with finite differences" returns many interesting results.

```
int TransmissionLineTransient::CreateFiniteDiffMatrix()
```