

Lecture II

1D FDTD - transmission line

In this lecture, we continue to study the Finite-Difference Time-Domain (FDTD) method. We apply this method to a *distributed* electric circuit. We intend to solve a (linear or not) transmission line - a circuit with a distributed inductance/capacitance that supports the propagating EM waves.

1. Transmission line - a distributed circuit

A typical transmission line is shown in Fig. 1. It is characterized by a distributed capacitance per unit length, C , as well as by a distributed inductance, L , and a resistance per unit length, R .

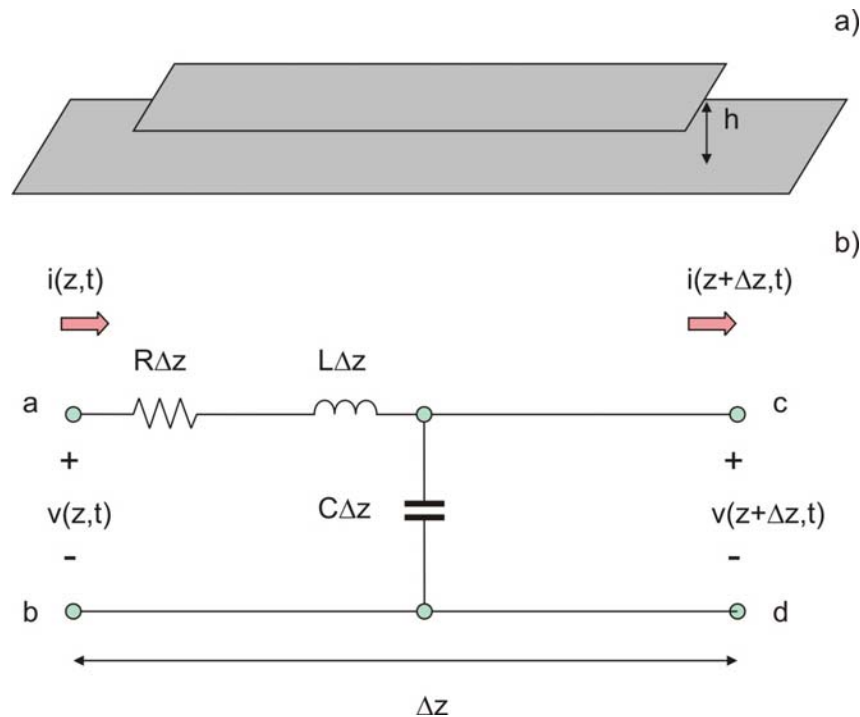


Fig. 1. A transmission line and its equivalent circuit. Line conductance or dielectric loss is ignored. Line resistance or metal loss is included into consideration.

The problem now involves not only the temporal dependence (studied in lecture #1), but also a spatial dependence since both voltage and current change along the transmission line. As a result, we have a *distributed* electric or RF circuit.

2. Transmission line equations

The corresponding transmission line equations for the line voltage $v(z,t)$ (voltage between the top conductor and the ground plane) and the line current $i(z,t)$ (current

through the top conductor) are obtained by applying the KVL and KCL to the circuit shown in Fig. 1b. By KVL for the *abcd* loop,

$$-v(z, t) + R\Delta z i(z, t) + L\Delta z \frac{\partial i}{\partial t} + v(z + \Delta z, t) = 0 \quad (1)$$

By KCL for the top node,

$$-i(z + \Delta z, t) + i(z, t) = C\Delta z \frac{\partial v}{\partial t} \quad (2)$$

Dividing these two equations by Δz and grouping the terms, we obtain,

$$\frac{v(z + \Delta z, t) - v(z, t)}{\Delta z} = -R i(z, t) - L \frac{\partial i}{\partial t} \quad (3)$$

$$\frac{i(z + \Delta z, t) - i(z, t)}{\Delta z} = -C \frac{\partial v}{\partial t} \quad (4)$$

or, in terms of spatial derivatives,

$$\frac{\partial v}{\partial z} = -R i(z, t) - L \frac{\partial i}{\partial t} \quad (5)$$

$$\frac{\partial i}{\partial z} = -C \frac{\partial v}{\partial t} \quad (6)$$

Eqs. (5), (6) are the celebrated partial differential equations for the transmission line. The line resistance per unit length, R , is often ignored. Therefore, one finally obtains quite a symmetric result, i.e.

$$\frac{\partial v}{\partial t} = -C^{-1} \frac{\partial i}{\partial z} \quad (7)$$

$$\frac{\partial i}{\partial t} = -L^{-1} \frac{\partial v}{\partial z} \quad (8)$$

Our goal is to solve partial differential Eqs. (7, 8) (a system of partial differential linear equations) numerically and to establish signal propagation along the transmission line at any initial and/or boundary conditions. This task may require two separate lectures.

Note: Eqs. (7, 8) may be reduced to one wave equation (for voltage or current) in the form (see below)

$$\frac{\partial^2 v}{\partial t^2} - \frac{1}{LC} \frac{\partial^2 v}{\partial z^2} = 0 \quad (9)$$

The wave propagation speed is $1/\sqrt{LC}$; it coincides with the speed of light in a air-filled transmission line. However, we will rather solve Eqs. (7, 8) instead of Eq. (9).

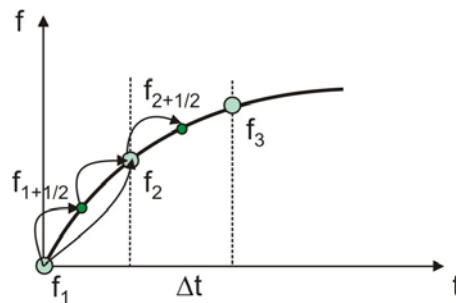
3. Numerical method - time derivatives

The Runge-Kutta method studied in lecture#1 is *not* used for the problems involving the spatial dependence. Instead, we use the central differences, which have the same accuracy, but require knowledge of *two* starting points instead of one physical point. Fig. 2 shows the concept and highlights the difference between two methods for a differential equation

$$\frac{df}{dt} = F(f, t) \quad (10)$$

To start the central differences, we thus need two initial conditions: $f_1, f_{1+1/2}$. One of them should be calculated separately. Otherwise, both methods are equivalent in accuracy.

a) 2nd order Runge-Kutta



b) 2nd order central difference

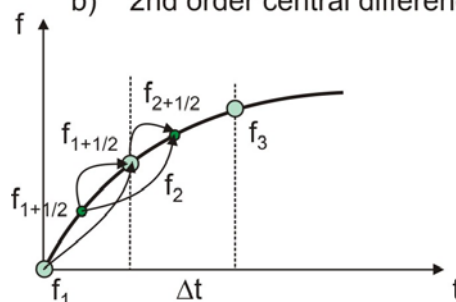


Fig. 2. A difference between the second-order Runge-Kutta method (top) and the central differences (bottom). In both, the right-hand side is calculated at non-integer nodes.

4. Numerical method - staggered grid and “marching on in time”

Eqs. (7, 8) could then be solved using the standard second-order finite-difference approximation in both time and space domain. However, a staggered grid is routinely used here. It allows us to reduce the number of nodes by a factor of four!

Fig. 3 shows the concept of the staggered grid. This concept is two-dimensional in nature and requires some explanation.

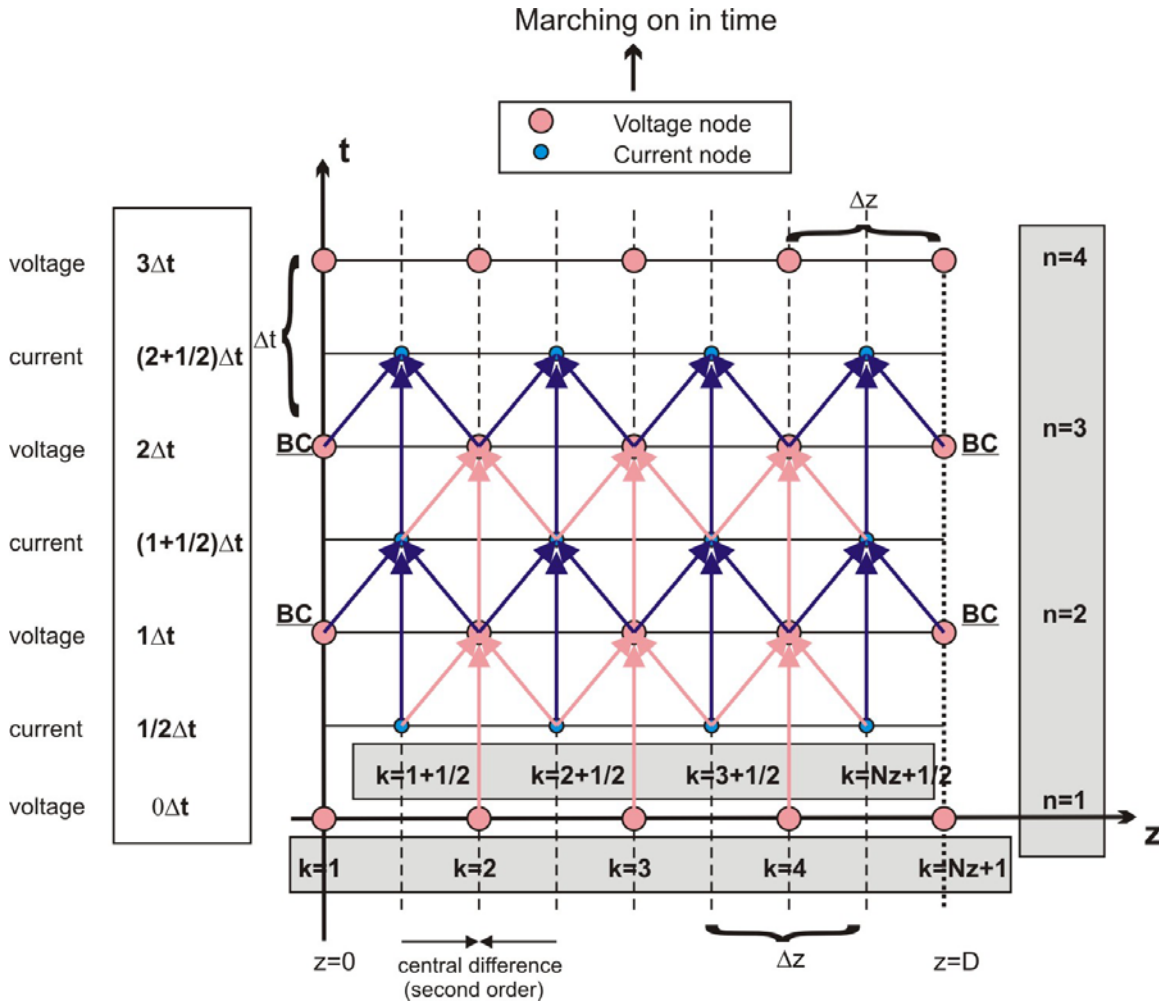


Fig. 3. The concept of the staggered grid and marching on in time for transmission line Eqs. (7, 8). The transmission line length is D .

First and foremost, the voltage and current are defined at *different* nodal discretization points, for both t and z . They never coincide. The nodal points for voltage and current are interleaving each other as shown in Fig. 3.

Second, the initial conditions (bottom of Fig. 3) are formulated in terms of voltage at $t = 0$ and in terms of current at $t = \frac{1}{2}\Delta t$. This discrepancy is an intrinsic feature of the staggered grid. However, it usually has a minor influence on the solution.

The staggered grid implies the use of central differences both in space and in time, and is therefore second-order accurate both in space and in time. The realization of the staggered grid in Fig. 3 applied to Eqs. (7, 8) yields

$$\begin{aligned}\frac{\partial v(t, z)}{\partial t} &= -C^{-1} \frac{\partial i(t, z)}{\partial z} \Rightarrow \frac{V_k^{n+1} - V_k^n}{\Delta t} = -C^{-1} \frac{I_{k+1/2}^{n+1/2} - I_{k-1/2}^{n+1/2}}{\Delta z} \\ \frac{\partial i(t, z)}{\partial t} &= -L^{-1} \frac{\partial v(t, z)}{\partial z} \Rightarrow \frac{I_{k+1/2}^{n+3/2} - I_{k+1/2}^{n+1/2}}{\Delta t} = -L^{-1} \frac{V_{k+1}^{n+1} - V_k^{n+1}}{\Delta z}\end{aligned}\quad (11)$$

Here, the capital letters denote numerical values of voltage/current, which are in general different from the exact values.

According to Eqs. (11), the line voltage is always found at integer nodes whereas the line current – at non-integer nodes.

Eqs. (11) say that the marching on in time operates in the following way:

-
1. We find all voltages in space at time step $n = 2$, $t = \Delta t$ using voltage values at the previous time step $n = 1$, $t = 0$ and current values at the intermediate time step $n = 1/2$, $t = 1/2\Delta t$.
 2. We find all currents in space at the intermediate time step $n = 1 + 1/2$, $t = \Delta t + \Delta t/2$ using current values at the previous intermediate time step $n = 1/2$, $t = \Delta t/2$ and voltage values at the previous time step $n = 2$, $t = \Delta t$.
 3. We find all voltages in space at time step $n = 3$, $t = 2\Delta t$ using voltage values at the previous time step $n = 2$, $t = \Delta t$ and current values at the intermediate time step $n = 1 + 1/2$, $t = \Delta t + \Delta t/2$.
 4. We find all currents in space at the intermediate time step $n = 2 + 1/2$, $t = 2\Delta t + \Delta t/2$ using current values at the previous intermediate time step $n = 1 + 1/2$, $t = \Delta t + \Delta t/2$ and voltage values at the previous time step $n = 3$, $t = 2\Delta t$.
 5. We find all voltages ..., etc.
-

The method thus remain "bootstrapping" - see Fig. 3. It is sometimes called exactly in the same way.

5. Numerical method - simple boundary conditions

Yet another point of concern are the boundary conditions that have to be defined at $z = 0, D$, i.e. at the ends of the transmission line -see Fig. 3. Fig. 3 indicates that,

formally, we only need to know the voltage values at these nodes. No current knowledge is therefore formally necessary at the boundary.

The simplest boundary condition here is $v(t, z = 0, D) = 0$. This condition corresponds to a short-short transmission line.

Note: The bootstrapping method could be rewritten by swapping voltage and current integration nodes. In that way, the line voltage is always found at non-integer nodes whereas the line current – at integer nodes.

The simplest boundary condition now would be $i(t, z = 0, D) = 0$. This condition corresponds to an open-open transmission line.

6. Realization

Marching on in time also implies that the solution at all time moments does not have to be stored. In particular, four realizations - the past voltage and the past current arrays, and the next voltage and the next current arrays - are quite sufficient - see Fig. 3. We could even work with two arrays in a recursive fashion.

We will work here with four row arrays: two past rows and two next rows. The arrays with integer and non-integer indexes in Eqs. (11) are thus replaced by MATLAB arrays

```
Inext = zeros(1, Nz+0); % fourth layer in the series of four (n+3/2)
Vnext = zeros(1, Nz+1); % third layer in the series of four (n+1)
Ipast = zeros(1, Nz+0); % second layer in the series of four (n+1/2)
Vpast = zeros(1, Nz+1); % first layer in the series of four (n+0)

% Z-coordinates for the voltage nodes
zkv = [0:dz:D]; % Nz+1 total
% Z-coordinates for the current nodes
zki = [dz/2:dz:D-dz/2]; % Nz total
```

where N_z is the number of spatial discretization nodes - see Fig. 3. Note the argument values corresponding to those arrays.

Eqs. (11) are then programmed in a very simple form (a batch mode in MATLAB)

```
indv = 2:Nz; % batch process in space (voltage)
indi = 1:Nz; % batch process in space (current)
delta = dt/dz; % standard ratio

Vnext(indv) = Vpast(indv) - delta*(1/C)*(Ipast(indv+0) - Ipast(indv-1));
Inext(indi) = Ipast(indi) - delta*(1/L)*(Vnext(indi+1) - Vnext(indi-0));
Vpast = Vnext;
Ipast = Inext;
```

The FDTD method for a distributed circuit is done!

Example 1: A short-short transmission line with the total length of $D=300\text{mm}$, with the capacitance per unit length of 122 pF/m and inductance per unit length of $0.302\text{ }\mu\text{H/m}$ (a 50Ω transmission line on 62mil FR4) is subject to an initial voltage excitation in the form

$$v(t=0, z) = 1\text{V} \exp\left(-\frac{(D/2-z)^2}{\sigma^2}\right), \quad \sigma = 200\text{mm} \quad (12)$$

The initial current is zero. Using the bootstrapping method (the second-order differences) find the evolution of the initial voltage pulse along the line.

Solution: A relatively simple MATLAB script given below solves the problem and outputs the solution behavior when time progresses.

```
clear all;
%-----
D      = 300.0e-3;          % Trace trace length, m
Nz     = 400; dz = D/Nz;   % Number of spatial cells and spatial step
zkv    = [0:dz:D];         % Z-coordinates for the voltage nodes, Nz+1 total
zki    = [dz/2:dz:D-dz/2]; % Z-coordinates for the current nodes, Nz total
%-----
C      = 1.22e-010;        % Line capacitance per unit length
L      = 3.02e-007;        % Line inductance per unit length
%-----
cph    = 1/sqrt(L*C);      % phase speed on the main line
dt     = (1/cph)*dz;       % magic time step
T      = 5e-9;             % observation time
NT     = round(T/dt); t = [0: dt: NT*dt]; % Number of time steps and time vector
%-----
% FDTD - marching on in time
% Initialization
Inext   = zeros(1, Nz+0); % fourth layer in the series of four (n+3/2)
Vnext   = zeros(1, Nz+1); % third layer in the series of four (n+1)
Ipast   = zeros(1, Nz+0); % second layer in the series of four (n+1/2)
Vpast   = zeros(1, Nz+1); % first layer in the series of four (n+0)
indv    = 2:Nz;           % batch process in space (voltage)
indi    = 1:Nz;           % batch process in space (current)
delta   = dt/dz;          % standard ratio
% Initial condition (voltage)
sigma   = 0.02; Vpast = exp(-(D/2 - zkv).^2/sigma^2);
% Main loop - "bootstrapping"
n = 1;
while n <= NT-1
    Vnext(indv) = Vpast(indv) + delta*(-1/C)*(Ipast(indv+0) - Ipast(indv-1));
    Inext(indi) = Ipast(indi) + delta*(-1/L)*(Vnext(indi+1) - Vnext(indi-0));
    Vpast      = Vnext;
    Ipast      = Inext;
    n          = n + 1;
%-----
    string = strcat(num2str(1e9*t(n)), ' ns');
    x = [min(zkv) max(zkv)]; y = [0.5 0.5];
    plot(zkv, Vnext(1, :), 'r', 'LineWidth', 2); grid on;
    line(x, y, 'LineWidth', 2); line(x, -y, 'LineWidth', 2);
    axis([min(zkv), max(zkv) -1 1]); title(strcat('Line voltage at t=', string));
    xlabel('TL length, m'); ylabel('Voltage instant. profile'); drawnow;
%-----
end
```

Question: Could we model a simple *traveling electromagnetic wave* on a transmission line? Let's say a wave traveling to the right?

Answer: In an electromagnetic wave traveling to the right,

$$v = v(z - c_{ph}t), \quad i = i(z - c_{ph}t) \quad (13)$$

where

$$c_{ph} = 1/\sqrt{LC} \leq c_0 \quad (14)$$

is the wave *propagation speed (phase velocity)*. Substitution of Eq. (13) into Eq. (7) yields

$$-c_{ph}v'(z - c_{ph}t) = -C^{-1}i'(z - c_{ph}t) \quad (15)$$

Integration of Eq. (15) and setting the corresponding integration constant equal to zero gives

$$i = c_{ph}Cv \quad (16)$$

Thus, in order to model the traveling wave, we should give voltage and current as two initial conditions:

$$v(t = 0, z) = 1V \exp\left(-\frac{(D/2 - z)^2}{\sigma^2}\right), \quad \sigma = 200\text{mm} \quad (17a)$$

$$i(t = 1/2\Delta t, z) = c_{ph}C \exp\left(-\frac{(D/2 - z + \Delta z/2)^2}{\sigma^2}\right), \quad \sigma = 200\text{mm} \quad (17b)$$

where

$$\Delta z = c_{ph}\Delta t \quad (18)$$

The MATLAB code to example 1 could be changed accordingly:

```
sigma = 0.02; vpast = exp(-(D/2 - zkv).^2/sigma^2);
Ipast = cph*C*exp(-(D/2 - zki + dz/2).^2/sigma^2);
```

This new code will give us an evolution of the traveling wave.

7. Choosing spatial resolution (Δz)

Based on the previous examples, some general observations can be made, with regard to Δz and Δt . A common rule here is

- i. at least eight spatial nodes either for voltage or for current per wavelength;
- ii. at least four spatial nodes per ramp for a bit form;
- iii. at least eight spatial nodes for a monopolar pulse.

There are some quite impressive special schemes developed for shock waves, which track a discontinuity explicitly and require only one or a few points for a step function.

8. Choosing temporal resolution (Δt). Stability

To implement a finite-difference scheme, the size of the time step must be chosen small enough, in order to avoid numerical instability. The stability criterion (the so-called Courant (or Courant-Friedrichs-Levy) condition) also implies convergence to the true solution in the limit $\Delta t \rightarrow 0$. The standard criterion for hyperbolic equations (Maxwell's equations, wave equations, telegrapher equations (7), (8) for the transmission line, aerodynamic equations, acoustics, etc.) reads

$$\Delta t \leq \frac{\Delta z}{c_{ph}} \quad (19)$$

where c_{ph} is phase propagation speed on the line. Physically, Eq. (19) simply means that the FDTD should be able to track *fast* wave motion, with the speed c_{ph} . One might try to ignore the condition (19) in the above exercises, i.e. choose $\Delta t = 1.1 \frac{\Delta z}{c_{ph}}$, for example.

The result will speak for itself!

Unfortunately, usually we *do not know* the phase speed in advance. Therefore, Eq. (19) could be written in a "weak" form

$$\Delta t \leq \frac{\Delta z}{c_0} \quad (20)$$

which is perhaps more practical for our basic purposes. Here, c_0 is the speed of light in vacuum.

9. Numerical dispersion and the magic time step

On the other hand, we have seen in the previous example that the time step given by the limit of Eq. (19), i.e.

$$\Delta t = \frac{\Delta z}{c_{ph}} \quad (21)$$

gives an excellent performance if the phase speed on the line is known exactly. This time step is commonly called the "magic" time step. Why is this time step so special? To answer this question we will replicate a simple and elegant derivation given in the class textbook.. The system of 1D telegrapher equations (7), (8) is converted to the wave equation first, by differentiating the first equation with respect to time and substituting the time derivative of the current from the second equation. This yields

$$\frac{\partial^2 \mathcal{U}(t, z)}{\partial t^2} - c_{ph}^2 \frac{\partial^2 \mathcal{U}(t, z)}{\partial z^2} = 0, \quad c_{ph} = \frac{1}{\sqrt{LC}} \quad (22)$$

The standard central finite-difference approximation to this equation takes the form

$$\frac{V_k^{n+1} - 2V_k^n + V_k^{n-1}}{\Delta t^2} - c_{ph}^2 \frac{V_{k+1}^n - 2V_k^n + V_{k-1}^n}{\Delta z^2} = 0 \quad (23)$$

An exact solution to Eq. (23) is a sinusoidal traveling wave,

$$\mathcal{U}(t, z) = V_0 \exp(i\omega t - jk_{ph} z), \quad k_{ph} = \omega / c_{ph} \quad (24)$$

We therefore expect an exact solution to Eq. (23) to be

$$V_k^n = V_0 \exp(i\omega n \Delta t - jk_{ph} k \Delta z) \quad (25)$$

We plug Eq. (25) into Eq. (23), divide the result by $\exp(i\omega n \Delta t - jk_{ph} k \Delta z)$, convert the complex exponentials to the sine functions, and finally obtain,

$$\sin \frac{\omega \Delta t}{2} = \frac{c_{ph} \Delta t}{\Delta z} \sin \frac{k_{ph} \Delta z}{2} \quad (26)$$

When $\Delta t = \Delta z / c_{ph}$, Eq. (26) gives $k_{ph} = \omega / c_{ph}$, and the FDTD solution is exact! When $\Delta t \neq \Delta z / c_{ph}$,

$$k_{ph} = \frac{\omega}{c_{ph}^*}, \quad c_{ph}^* = c_{ph}^*(\omega, \Delta t / \Delta z) \neq c_{ph} \quad (27)$$

This means that the particular harmonic wave will move with the speed different from that of the true phase speed, i.e. the solution will be wrong. Moreover, harmonic waves with different frequencies will now move with different speeds. A voltage signal (e.g. a bit form), which is a sum of these harmonic waves, may be deformed or destroyed when time progresses. This effect is the so called *numerical dispersion*.

Fortunately, the situation is not as bad as might appear at first sight. Let us assume that $\Delta t \rightarrow 0, \Delta z \rightarrow 0$ in Eq. (26). Then, using the Taylor expansion $\sin x \rightarrow x$, we obtain that $k_{ph} = \omega / c_{ph}$ again. In other words, small discretization steps will generally lead to a smaller dispersion. And yet, since the dispersion accumulates over time, its effect for a large number of small

steps might be exactly as profound as for a smaller number of larger steps. This interesting question is beyond the scope of this lecture. We note that many special dispersion-preserving FDTD schemes exist.

HW#2 exercise

1. Modify the finite-difference scheme (11) by including into consideration the line resistance per unit length, R , as described by Eq. (5):

$$\frac{\partial v}{\partial z} = -Ri(z,t) - L \frac{\partial i}{\partial t}$$

Present the corresponding modified finite-difference equations.

2. Modify the code to example #1 by programming your new findings.
3. Observe the solution for the line voltage to example #1 when $R = 100\Omega/\text{m}$.
4. Plot this solution at $t = 7.3\text{ns}$.