

Simulation and Animation in Optical Fiber Communication

Brian Jenkins
United States Naval Academy

ABSTRACT

Students find certain concepts in fiber optic communication theory difficult to grasp. A classic example is electromagnetic mode theory in cylindrical coordinates. The solutions of the differential equation which describe the modes of the fiber are Bessel functions, with which most students are unfamiliar, and the vectorial nature of the analysis only complicates the situation. A second difficult concept is dispersion. The fact that different frequencies of light travel at different speeds in the fiber is not confusing. It is in the implications where the difficulty arises. Specifically, the concept of group velocity is physically counterintuitive, and an analysis of the pulse spreading that results because of dispersion often includes a rigorous Fourier analysis. This paper describes how computer simulation and animation can provide a visual means of simplifying these concepts so that they are easier to understand.

INTRODUCTION

Computer animation has become popular in academia because the common student is visually oriented. Furthermore, some concepts are simply easier to understand visually. Modern computing tools provide the visual means to educate students more efficiently in concepts that are traditionally difficult to teach at a blackboard using mathematical derivation. In engineering, ideas also may be simple mathematically but physically counterintuitive. Computer simulation can enable a student to jump over the hurdle that an abstract physical concept presents. High levels of abstraction are especially prevalent in electromagnetic field theory and Fourier analysis, two disciplines which are fundamental in fiber optic communication theory. We use animation and simulation in MATLAB to help students grasp some of the more complex topics in these disciplines.

MATLAB is a matrix driven language which integrates numerics and graphics in a single package.¹ It provides a computing environment which is relatively easy for the students to understand. This can be true even for students that have not been formally educated in higher-level programming languages. MATLAB programs are written using mathematical expressions similar to those that most engineering students are familiar with. Furthermore, it provides a comprehensive graphical environment, which includes color, surface plots in three-dimensions, etc., that enables an engineering educator to create relevant animations that provide the student with greater insight into the underlying physical processes.

Three MATLAB animations have been used in the fiber optic communications course at the Naval Academy, a senior elective in the electrical engineering department. In the first of these, the temporal and spatial behavior of various electromagnetic modes of the fiber may be observed. The second simulation illustrates why the group velocity of a pulse differs from the phase velocity of the

optical carrier in a dispersive media. In the last animation, students observe how the energy in an optical pulse spreads because of group velocity dispersion. The last two animations help students understand the complex relationship between the amplitude and phase of the frequency components which are integral to pulse formation and degradation in the fiber.

ANIMATION OF MODES IN AN OPTICAL FIBER WAVEGUIDE

An optical fiber guides light because of total internal reflection; energy in the optical wave is reflected at the interface between the core and the cladding of the fiber and remains in the core. The energy propagates down the fiber core in modes; animation helps students understand what modes are and how they behave in time and space. The differential wave equation which describes the propagation of the light in the core of the fiber is²

$$\nabla(\nabla \cdot \mathbf{E}) - \nabla \times \nabla \times \mathbf{E} - k^2 \frac{\partial^2 \mathbf{E}}{\partial t^2} = 0.$$

The solutions of the electric field \mathbf{E} are written in cylindrical coordinates r , ϕ , and z . We use MATLAB to analyze these solutions. The spatial frequency of the optical wave is $k = \omega n/c = 2\pi n/\lambda_0$, where ω is the angular frequency of the wave, $c = 3 \times 10^8$ m/sec is the speed of light in the vacuum, n is the index of refraction in the fiber core and λ_0 is the wavelength of the light.

The modes that are simulated are derived from the real part of the complex components of \mathbf{E} ,

$$E_z(r, \phi, z, t) = J_\nu(\kappa r) e^{j\nu\phi} e^{-j\beta z} e^{j\omega t}$$

$$E_\phi(r, \phi, z, t) = \frac{-j}{\kappa^2} \left(\frac{\beta}{r} \frac{\partial E_z}{\partial \phi} - \omega \mu_0 \frac{\partial H_z}{\partial r} \right)$$

$$E_r(r, \phi, z, t) = \frac{-j}{\kappa^2} \left(\frac{\mu_0 \omega}{r} \frac{\partial H_z}{\partial \phi} + \beta \frac{\partial E_z}{\partial r} \right)$$

where μ_0 is the magnetic permeability and H_z (the z -component of the magnetic field vector) is similar in form to E_z . J_ν is a Bessel function of the first kind of order ν . The roles ν , κ and β play are especially troublesome for students. Computer animation allows a student to directly observe the impact of these parameters. While ν is the order of the Bessel function, it is also an integer that represents the frequency of sinusoidal oscillation azimuthally in ϕ around the fiber core. The Bessel functions themselves are oscillatory with frequency κ and look like damped sinusoidal waves radiating outward in r from the center of the fiber. β is the frequency of oscillation of the wave as it propagates down the fiber longitudinally in z . κ and β are related to k by $k^2 = \kappa^2 + \beta^2$.

Ultimately, κ and β take on only discrete values, as determined by applying the electromagnetic boundary conditions to the field components at the interface between the fiber core and cladding. The resultant solutions for every integer value of ν are the modes of the fiber. Unfortunately, many textbooks do not relate the math that describes \mathbf{E} to what an observer sees at the output of the fiber. Hence, students quickly lose sight of the physical meaning of the math. Since each mode may be viewed as a different intensity distribution $|\mathbf{E}|^2$ at the fiber output, graphics and animation provide

a visual path to the meaning of modes. Simply put, each mode looks different at the end of the fiber. As ν and κ vary, changes in the field distribution in ϕ and r can be graphically represented in a three-dimensional surface plot, and β can be described through the animation.

To introduce the concept of modes to the students, we begin with the fundamental mode of the fiber since it is the simplest. The field amplitude is plotted versus $x = r\cos\phi$ and $y = r\sin\phi$ in Figure 1(a), where the radius of the fiber is a ; we are plotting the fundamental mode in the cross-section of the fiber. The strength of the field is highest in the middle of the fiber; an observer would see a single bright spot at the end of the fiber. There is no azimuthal variation, so $\nu = 0$, and the field is approximated by the $J_0(\kappa r)$ function,³ which is maximum at $r = 0$. This plot is made with the standard three-dimensional surface plotting functions of MATLAB.

We then consider a mode with $\nu = 1$ in Figure 1(b). The azimuthal variation in the field is $\text{Re}\{e^{j\phi}\}$, so there is one period rotationally about the fiber as ϕ varies between 0 and 2π . In addition, the radial variation is $J_1(\kappa r)$, which is zero at the center of the fiber. The intensity distribution appears as two bright lobes. (The students also observe this mode experimentally in the lab associated with the course.) The lobe with negative field amplitudes indicates that the polarization is in the opposite direction for $\pi/2 < \phi < 3\pi/2$. Since the azimuthal variation tends to be the most difficult to grasp, another mode with $\nu = 2$ is shown in Figure 1(c), where there are four lobes in a singular ring about the center of the fiber.

Figure 1(d) illustrates what happens to the mode in Figure 1(c) as κ increases. Specifically, there are more cycles in the radial direction. In the intensity distribution, an observer sees two rings, with four bright lobes in each. Furthermore, the Bessel functions are damped as r increases, so the amplitude in the outer ring is less than the amplitude in the inner ring. In intensity, the four lobes in the outer ring are less bright than the four lobes in the inner ring.

When we include animation, we see how β impacts each mode. Visually, the modes oscillate about the origin of the E -axis at a frequency that corresponds to the value of β . Physically, this corresponds to fixing time at a specific instant and watching the field vary as the position z along the length of the fiber is varied. Since the longitudinal variation in the field is $e^{-j\beta z}$, the oscillatory behavior speeds up as β increases. Higher order modes, such as the one in Figure 1(d), have smaller values of β than the lower order modes, so they oscillate less rapidly.

To create the animation, the movie functions of MATLAB are used in combination with the surface plotting functions. MATLAB stores successive frames of the animation and plays them back as a movie. The MATLAB file which creates the animation of Figure 1(d) is included in Appendix A.

DISPERSIVE EFFECTS IN PULSE PROPAGATION

We also use animation to demonstrate how dispersion affects pulse propagation in the fiber. Dispersion is a fundamental property of the glass from which a fiber is formed. A fiber is dispersive because the index of refraction n and the propagation constant β (which is proportional to n) are different at each optical frequency. The phase velocity of a wave with frequency ω is $v_p = c/n = \omega/\beta$, so the velocity also varies with frequency in a dispersive material. Since any data pulse is formed through the superposition of energy at different frequencies, digital pulses carried on an optical wave eventually break apart after propagating a sufficient distance.

To simulate dispersive effects in a pulse, we analyze the Taylor expansion of β ,

$$b(\omega) = b_0 + b_1(\omega - \omega_0) + b_2(\omega - \omega_0)^2/2 + \dots,$$

which describes how the propagation constant (and the index) varies about the optical frequency ω_0 . The coefficients $\beta_m = (d^m\beta/d\omega^m)$ are defined at ω_0 . β_0 is the parameter that was described through the animation of fiber modes in Figure 1, and the meaning of the terms β_1 and β_2 are best described through simulation as well. β_1 is related to the group velocity v_g of the pulse in the fiber, and β_2 is the dispersion coefficient that defines how rapidly a pulse spreads because of dispersion.

To teach the concept of group velocity, we simulate the behavior of two waves with different frequencies,⁴ as seen in Figure 2(a). The superposition of the waves

$$\begin{aligned} E' + E'' &= \cos(\omega' t - \beta' z) + \cos(\omega'' t - \beta'' z) \\ &= 2 \cos(\omega t - \beta z) \cos(\Delta\omega t - \Delta\beta z), \end{aligned}$$

is depicted in Figure 2(b) at time $t = 0$. The frequency ω is the average of ω' and ω'' . If the frequencies of the two waves are similar, the difference (or beat) frequency $\Delta\omega$ is small, and the superposition consists of a higher frequency carrier wave $\cos(\omega t - \beta z)$ that is modulated by a low frequency pulse envelope $\cos(\Delta\omega t - \Delta\beta z)$. The peak of the pulse envelope results where the two waves constructively interfere. In Figure 2 this is at $z = 0$, as indicated by the \otimes .

The velocity of the carrier wave is the phase velocity $v_p = \omega/\beta$. The velocity of the pulse envelope is the group velocity, $v_g = \Delta\omega/\Delta\beta$. For a pulse composed of many frequency components,

$$\lim_{\Delta\omega \rightarrow 0} \frac{\Delta\omega}{\Delta\beta} = \frac{d\omega}{d\beta} = v_g = \frac{1}{\beta_1},$$

where β_1 is the second term in the Taylor expansion of β . If ω/β is different from $d\omega/d\beta$, the phase velocity of the carrier wave is different from the group velocity of the pulse envelope. Since $\beta = n\omega/c$, whenever the index n varies with frequency, as it does in a dispersive material, v_g will be different from v_p .

This is rather easy to describe mathematically to students. However, it is not obvious to a student why one would physically expect a pulse to move at a different velocity than the carrier wave it is riding on, regardless of the mathematical argument. To physically see what is happening, we

animate the two waves shown in Figure 2(a), each at a different phase velocity, $v_p' = 0.19 \mu\text{m}/\text{fsec}$ and $v_p'' = 0.2 \mu\text{m}/\text{fsec}$, respectively. The group velocity of the pulse envelope is observed directly in the animation, but we can only show the final result here in Figure 3. Initially at $t = 0$ in Figure 2, the crests of the waves coincide at $z = 0$; in Figure 3(a), the two waves have moved $1.9 \mu\text{m}$ and $2.0 \mu\text{m}$ after 10 fsec, respectively, as indicated by the \circ and the \times . The phase velocity of the carrier wave is close to $v_p = 0.195 \mu\text{m}/\text{fsec}$, between v_p' and v_p'' , so the carrier moves $1.95 \mu\text{m}$ in 10 fsec.

In contrast, the pulse envelope in Figure 3(b) has shifted only slightly more than $1 \mu\text{m}$, so it moves a shorter distance in the same amount of time. Clearly, the pulse envelope moves at a slower velocity than the carrier wave. The reason lies in the fact that the envelope results from constructive interference between the two waves. As the two waves move at different velocities, the phase position at which they constructively interfere moves at a velocity different from the individual phase velocities. Hence, in the animation, the carrier wave appears to move through the pulse envelope as time progresses. Real data pulses generally consist of more than two frequency components, but the effect is precisely the same.

To create the animation in MATLAB, the handle graphics functions are used. This simulation is markedly different from that used to animate the modes. For simple waveforms such as those shown in Figures 2 and 3, MATLAB processing is fast enough to calculate and update the waveform on the screen in real time. The program to simulate group velocity is included in Appendix B.

The final animation demonstrates group velocity dispersion in a pulse. When $\beta_2 = 0$, β is linear with respect to frequency ω , so the group velocity v_g is the same for every frequency component of the pulse. The phase velocity of the carrier may be different from the group velocity of the pulse, but there is no distortion of the pulse as it propagates down the fiber. This is essentially what happens in Figure 3. In contrast, group velocity dispersion results when $\beta_2 \neq 0$. As the pulse propagates, its amplitude drops and its energy spreads. If digital information is transmitted on the pulse and too much energy spreads into neighboring bit slots, the information may be distorted. Hence, β_2 is the term in the Taylor expansion of β which causes degradation in a communication system.

The animation is based on the equation⁵

$$A(z, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} A(0, \omega) \exp\left(\frac{j}{2} \beta_2 \omega^2 z - j\alpha\right) d\omega,$$

where the final pulse envelope $A(z, t)$ at arbitrary z is related to the spectrum $A(0, \omega)$ of the initial pulse envelope $A(0, t)$ at $z = 0$. Essentially, each spectral component of the pulse experiences a phase shift of $\exp(j\beta_2 \omega^2 z / 2)$ upon propagation. Students sometimes find this equation intimidating because of the Fourier analysis. For many pulse shapes, though, the integral equation can be solved only through numerical simulation, so it is easiest to convey the physical meaning to the student this way. In fact students use discrete Fourier transforms in MATLAB to simulate this equation for a variety of pulse shapes during the course.

In the animation, we use a pulse with a Gaussian shape, although other pulse shapes could be chosen.

Figure 4 depicts a pulse at $z = 0$, and Figure 5 depicts the same pulse at the end of the animation. During the simulation, the pulse energy gradually spreads as the envelope moves down the fiber. A more realistic pulse would have thousands of cycles of the optical wave under the envelope, but the physical process is most easily seen by using a smaller carrier frequency.

The simulation assists in teaching the following concepts: 1) As pulse width increases, the amplitude drops, so energy is conserved. 2) The carrier moves through the pulse because the phase velocity and group velocity are different, as before. 3) The pulse is chirped as it propagates, meaning that the instantaneous frequency varies across the envelope. Lower frequency waves are in the front of the pulse in Figure 5 and higher frequencies are at the back. In the simulation, the initial pulse is green. As the pulse evolves, the color gradually changes to emphasize the chirp as the pulse propagates, so that the final pulse is red-shifted on the rising edge and blue-shifted on the trailing edge. 4) The sign of β_2 can be reversed so that higher frequencies move more quickly down the fiber; the colors are reversed in the animation. 5) A pulse really does consist of more than one frequency component; the multiple components emerge from the pulse as the energy disperses. 6) The concept of moving reference frames can be introduced since, in effect, the student moves along the fiber with the pulse envelope, watching what happens.

The handle graphics features of MATLAB were used again in this animation and colormaps were manipulated to simulate chirp. The program is included in Appendix C. MATLAB does not normally include a function which automatically scales the colormap in a two-dimensional line plot. The coloring is achieved using a MATLAB m-file called plotcol.m available on the MATLAB help screens on the World-Wide-Web.⁶

CONCLUSION

Three simulations have been described which assist in teaching fiber optic communication theory to undergraduate electrical engineering students. Thus far, students have been very receptive. Other simulations are planned which will simulate the effects of loss, amplification, noise and filtering that have become important in high-performance fiber communication systems.

REFERENCES

1. MATLAB User's Guide, The MathWorks, Inc., Natick, MA 01760, 1992.
2. Pollock, C.R., Fundamentals of Optoelectronics, Irwin, 1995, Ch. 5.
3. Gloge, D., "Weakly Guiding Fibers", Applied Optics, Vol. 10, No. 10, 10/71, pp. 2252-2258.
4. Hecht, E., Optics, Addison-Wesley, 1987, pp. 252-253.
5. Agrawal, G.P., Fiber-Optic Communication Systems, Wiley & Sons, 1992, p. 47.
6. See technical note 1205 at <http://www.mathworks.com/support/tech-notes/>.

BRIAN JENKINS received the B.S. and M.S. degrees from The Ohio State University in 1983 and 1991 and the Ph.D degree from the University of Colorado in 1995. From 1983-1989 he worked as a development engineer for IBM Corporation in Endicott, NY. Since 1996, he has been an assistant professor in the Department of Electrical Engineering at the U.S. Naval Academy where he teaches and does research in fiber optic communications.

APPENDIX A

```
%Matlab m-file used to simulate the LP22 mode of an optical fiber.
%The animation approximates the field in the fiber core. An exact
%analysis requires the use of Bessel functions of the second kind,
%K_nu, and matched boundary conditions at the core cladding interface.
%The movie command runs the frames. The speed at which the frames
%run may be controlled here.
%Written by B. Jenkins; 9/97
nu=2;      %Define the order of the Bessel function
a=4;      %Fiber radius
pts=50;    %Number of discretization points
r=[0:a/pts:a]; %Define cylindrical coordinate system
phi=[0:2*pi/pts:2*pi];
x=r*cos(phi);
y=r*sin(phi);
z=(bessel(1,1.75*r))*cos(nu*phi);
surf(x,y,z,abs(z)); %Plot the initial waveform in 3D
axis([-a a -a -2*max(max(z)) 2*max(max(z))])
lim=axis;
view(-10,30)
pcolor(z); %Scale color using field amplitude
colormap(hot) %Define the colormap
LP22m=moviein(20); %Allocate memory for movie frames
%Store successive frames
for j=1:20
    surf(x,y,sin(2*pi*j/20)*z,abs(z));
    axis(lim);
    view(-10,30);
    LP22m(:,j) = getframe;
end
%Run the movie one time
%movie(LP22m,1)
```

APPENDIX B

```
%Matlab m-file used to simulate group velocity in an optical fiber.
%The program gives a choice of using a dispersive or a non-dispersive
%material. Time is updated in the upper right corner as the
%simulation progresses.
%Written by B. Jenkins; 8/97
clear
clf
medium=input('Is this a dispersive(d) or non-dispersive(n) media?','s')
kb=12; %Define spatial and radial frequencies k and w for blue
kr=10.5; %and red waves...units are 1/um and 1/fs respectively
if medium=='d'
    wb=(kb/5)*.95; %Multiply wb by .95 to demonstrate group velocity
end
if medium=='n'
    wb=(kb/5);
```

```

end
wr=kr/5;
zinit=-4; %Define initial/final positions and step in um
zstep=.02;
zfinal=4;
z=zinit:zstep:zfinal;
t=0; %Initial time
bwave=cos(wb*t-kb*z); %Define blue and red waves
rwave=cos(wr*t-kr*z);
gwave=bwave+rwave; %Green wave = blue + red
beat=abs(2*cos((wb-wr)*t/2-(kb-kr)*z/2)); %The envelope of the green wave
subplot(2,1,1); %Start first plot
b=plot(z,bwave,'EraseMode','xor'); %Plot blue wave
axis([zinit zfinal -3 3]);
xlabel('Position \it{z}\rm (\mum)');
ylabel('\it{b}\rm, \it{r}\rm');
title('\it{b}\rm = cos(\omega_1\it{t} - \it{k}_1\it{z}\rm) \it{r}\rm = cos(\omega_2\it{t} - \it{k}_2\it{z}\rm)');
%set(gca,'Color',[0 0 0]);
hold on; %Hold axes
r=plot(z,rwave,'r','EraseMode','xor'); %Plot red wave
%legend here if desired
time=text(2,2,['\it{t} = \rm ',num2str(t),' fs']); %Give time and velocities
text(-3.5,2.5,['\it{v}_b = \rm ',num2str(wb/kb),' \mum/fs']);
text(-3.5,1.75,['\it{v}_r = \rm ',num2str(wr/kr),' \mum/fs']);
hold off; %Prepare for next plot
subplot(2,1,2); %Start second plot
g=plot(z,gwave,'g','EraseMode','xor'); %Plot green wave
axis([zinit zfinal -3 3]);
xlabel('Position \it{z}\rm (\mum)');
ylabel('\it{g}\rm = \it{b}\rm + \it{r}\rm');
%set(gca,'Color',[0 0 0]);
hold on;
envg=plot(z,beat,'g','EraseMode','xor'); %Plot envelope of green
hold off;
'Press a key to continue'
pause;
%Now update plots for advancing time
for j=1:400,
    t=j*.025;
    bwave=cos(wb*t-kb*z);
    rwave=cos(wr*t-kr*z);
    gwave=bwave+rwave;
    beat=abs(2*cos((wb-wr)*t/2-(kb-kr)*z/2));
    set(b,'YData',bwave);
    set(r,'YData',rwave);
    set(g,'YData',gwave);
    set(envg,'YData',beat);
    if rem(j,40)==0;
        set(time,'String',['\it{t} = \rm ',num2str(t),' fs']);
    end; drawnow; end;

```

APPENDIX C

```
%Matlab m-file used to simulate pulse dispersion in an optical fiber.
%The file creates a Gaussian pulse in a stationary reference frame
%with a scaled optical frequency and plots the new pulse as position
%z in the fiber changes. The pulse is chirped and the colormap
%is successively adapted to simulate chirping of the pulse. Program
%requires Matlab function file plotcol.m as well to allow coloring
%of a parametric surface line.
%Written by B. Jenkins; 10/97
clear
clf
t=-512:1:511; %Define time scale
T0=40; %Pulse width
B0=.2; %Assume a phase velocity for the carrier
B2=38; %Dispersion coefficient in ps^2/km
z=0; %Initial Position
f=.020; %Optical Frequency
%Define the complex pulse (magnitude and phase)
T1=sqrt(T0^2-j*B2*z);
twopift=2*pi*f*t;
phase=twopift+B0*z;
A=(T0/T1)*exp(-0.5*(t/T1).^2-j*phase);
n=4096; %Define the size of the colormap
x=jet(n); %Choose a colormap
%The next line is used to scale the intensity of the colormap at
%different time points so that the pulses fades to black on the
%edges where the wave amplitude decreases to zero.
absA=(abs(A(1:16:1024)).^(1/10))/max(abs(A).^(1/10));
%Choose the initial size of the colormap
colormap(diag(absA,0)*x(n/2-32:1:n/2+31,:));
p=plotcol(t,real(A),zeros(size(t)),-t);
axis([-500 500 -1 1])
set(gca,'Color',[0 0 0]);
xlabel('t (ps)')
ylabel('A(z,t)')
pause
%Begin the animation
for i=1:127,
    z=i*1.5;
    T1=sqrt(T0^2-j*B2*z);
    phase=twopift+B0*z;
    A=(T0/T1)*exp(-0.5*(t/T1).^2-j*phase);
    absA=(abs(A(1:16:1024)).^(1/10))/max(abs(A).^(1/10));
    set(p,'YData',[real(A);real(A)]);
    drawnow;
    %Update the colormap every so often
    if rem(i,4)==0;
        colormap(diag(absA,0)*x(n/2-32-(i/4)*64:2*(i/4)+1:n/2+31+(i/4)*62,:));
    end; end;
```

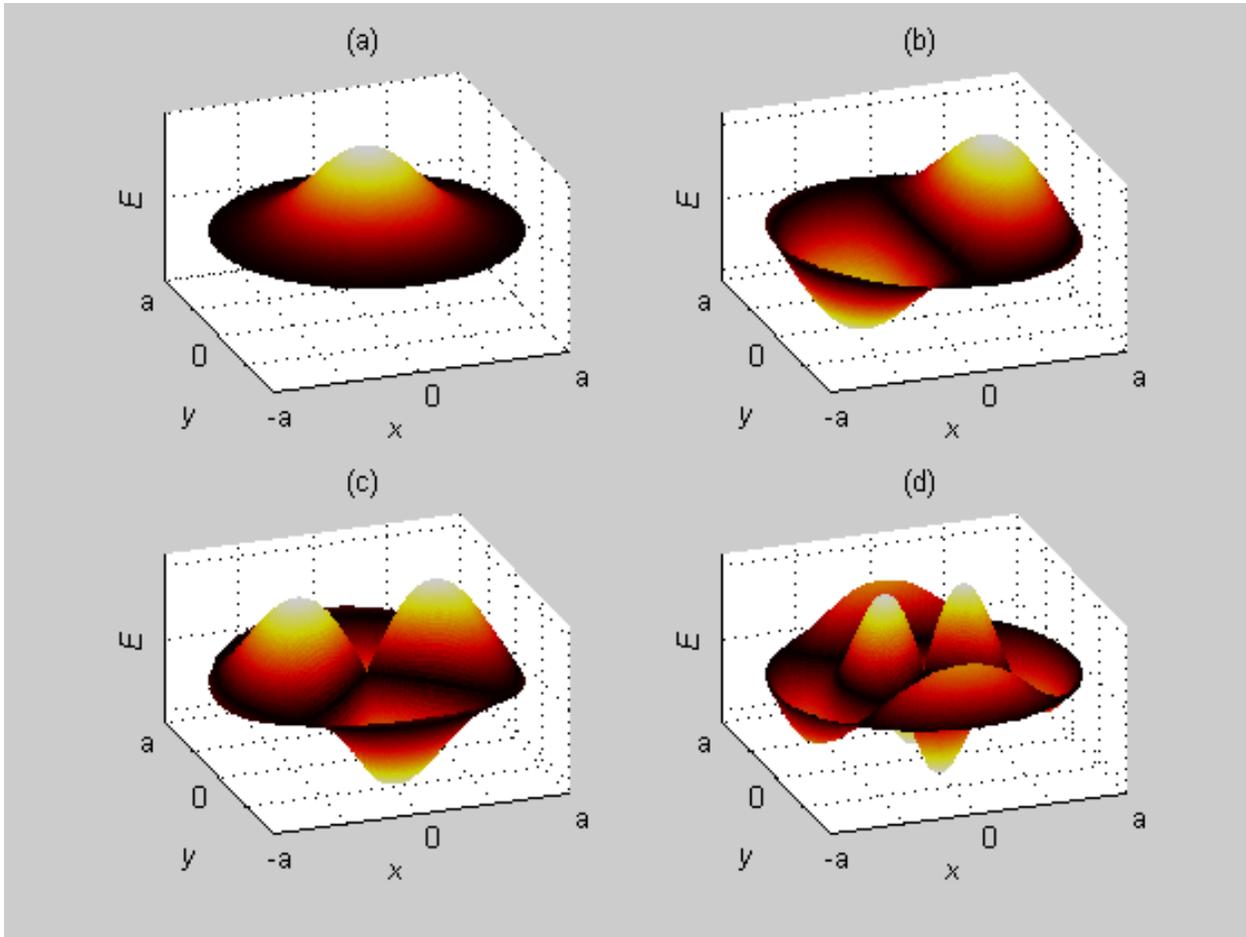


Figure 1: (a) Fundamental mode. (b) Mode with $v = 1$. (c) Mode with $v = 2$. (d) Mode with $v = 2$ and larger κ .

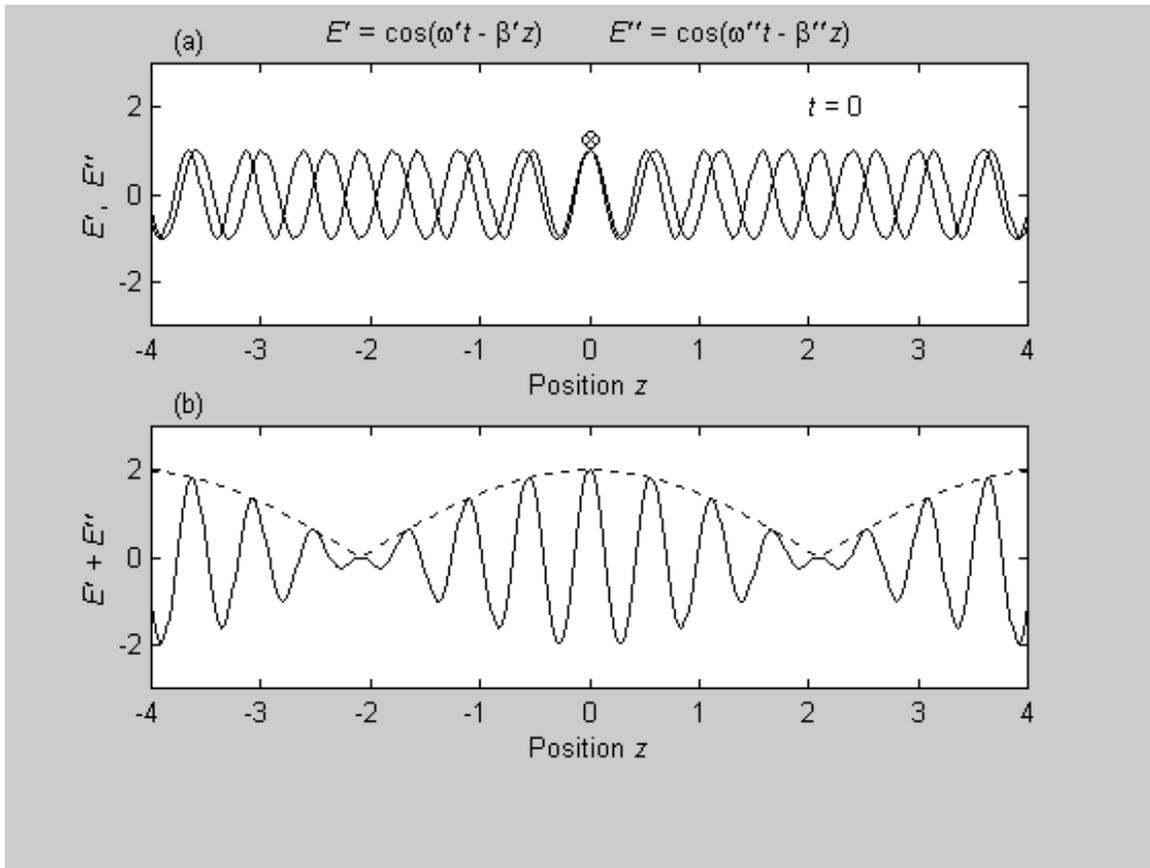


Figure 2: (a) Two waves E' and E'' with different frequencies at $t = 0$. (b) Superposition $E' + E''$ at $t = 0$.

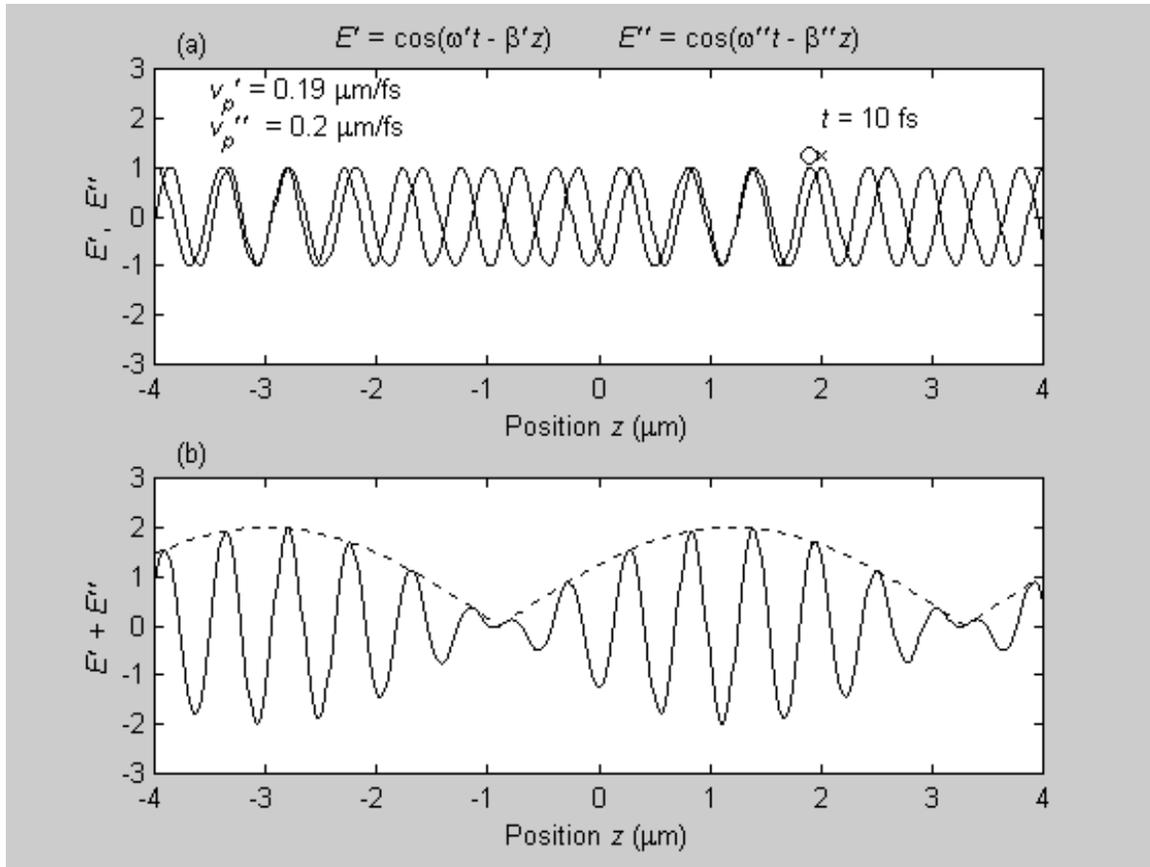


Figure 3: (a) Two waves E' and E'' with different frequencies at $t = 10$ fsec. (b) Superposition $E' + E''$ at $t = 10$ fsec.

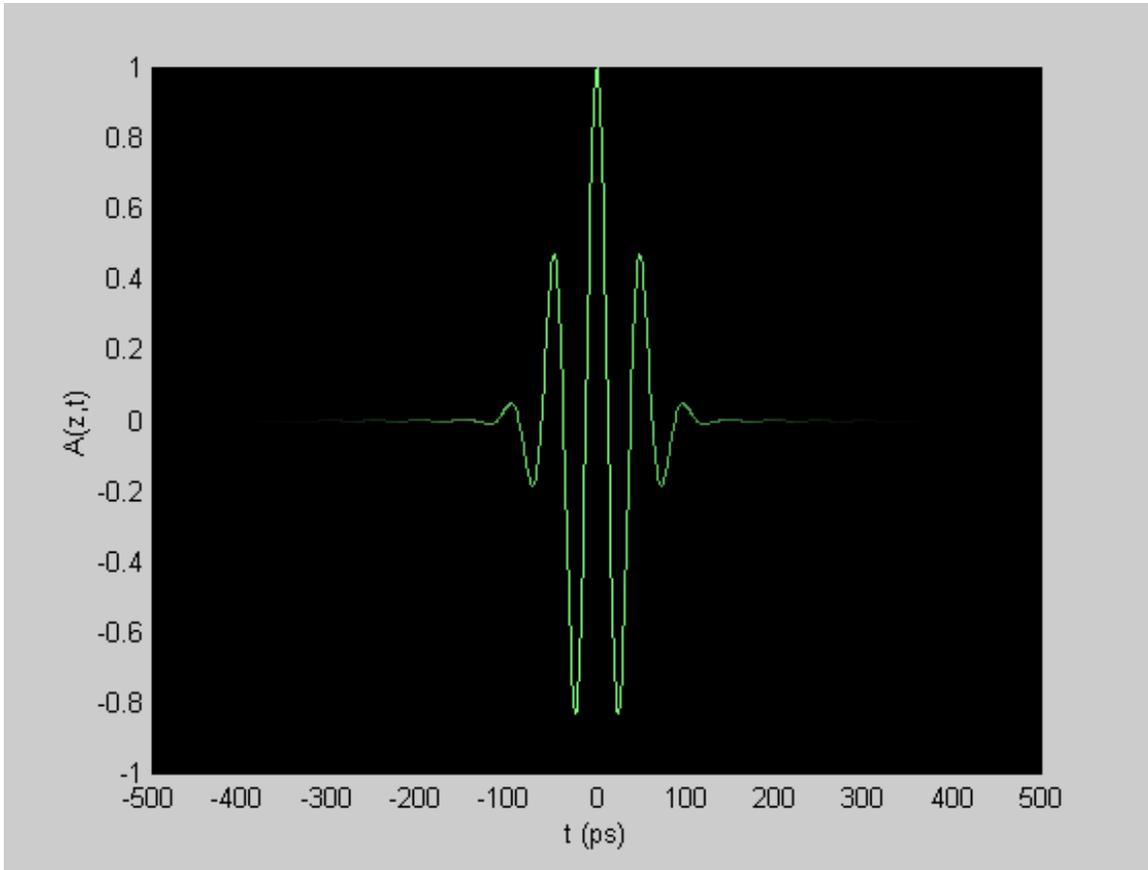


Figure 4: Initial pulse $A(0,t)$ before dispersion.

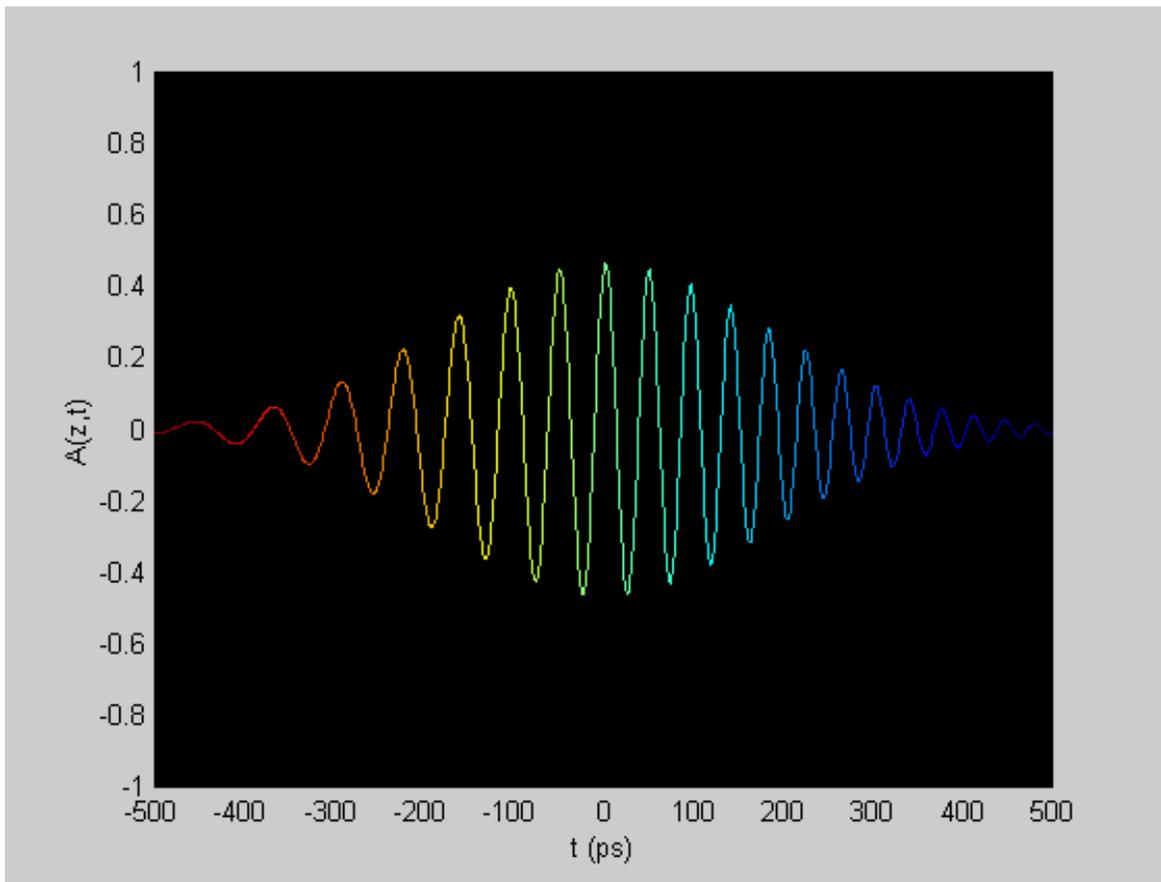


Figure 5: Final pulse $A(z,t)$ after dispersion. Pulse is chirped with lower frequencies shifted toward the front and higher frequencies shifted toward the back.