Project 16.35 Evolution of a wave packet in two dimensions

Both the half-step and split-operator algorithms can be extended to model the evolution of two-dimensional systems with arbitrary potentials $V(x, y)$. (See *Numerical Recipes* for how the FFT algorithm is extended to more dimensions.) Implement either algorithm and model a wave packet scattering from a central barrier and a wave packet passing through a double slit.

A clever way to insure stability in the half-step algorithm is to use a boolean array to tag grid locations where the solution becomes unstable and to set the wave function to zero at these grid points.

```
double minV = -2/dt;
double maxVx = 2/dt-2/(dx*dx);
double maxVy = 2/dt-2/(dy*dy);
double maxV = Math.min(maxVx,maxVy);
for(int i = 0, n = potential.length; i <= n; i++) {
   for(int j = 0, m = potential[0].length; j <= m; j++) {
      if (potential[i][j] >= minV && potential[i][j] <= maxV)
         stable[i][j] = true;  // stable
      else
         stable[i][j] = false; // unstable, set wave function to zero
   }
}
```

■

Project 16.36 Two-particle system

Rubin Landau has studied the time dependence of two particles interacting in one dimension with a potential that depends on their relative separation:

$$V(x_1, x_2) = V_0 e^{-(x_1-x_2)^2/2\alpha^2}. \tag{16.109}$$

Model a scattering experiment for particles having momentum p_1 and p_2 by assuming the following (unnormalized) initial wave function:

$$\Psi(x_1, x_2) = e^{ip_1x_1}\, e^{-(x_1-a)^2/4w^2}\, e^{ip_2x_2}\, e^{-(x_2-a)^2/4w^2}, \tag{16.110}$$

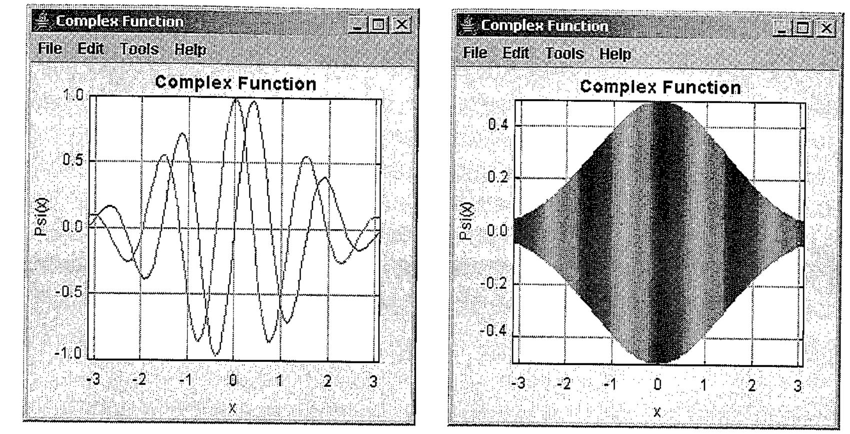
where $2a$ is the separation and w is the variance in each particle's position. Do the particles bounce off each other when the interaction is repulsive? What happens when the interaction is attractive? ■

APPENDIX 16A: VISUALIZING COMPLEX FUNCTIONS

Complex functions are essential in quantum mechanics and the frames package contains classes for displaying and analyzing these functions. Listing 16.12 uses `ComplexPlotFrame` to display a one-dimensional wave function.

Listing 16.12 The `ComplexPlotFrameApp` class displays a one-dimensional Gaussian wave packet with a momentum boost.

```
package org.opensourcephysics.sip.ch16;
import org.opensourcephysics.frames.ComplexPlotFrame;
```

(a) Real and imaginary. (b) Amplitude and phase.

Figure 16.2 Two representations of complex wave functions. (The actual output is in color.)

```
public class ComplexPlotFrameApp {
   public static void main(String[] args) {
      ComplexPlotFrame frame = new ComplexPlotFrame("x", "Psi(x)",
           "Complex function");
      int n = 128;
      double xmin = -Math.PI, xmax = Math.PI;
      double x = xmin, dx = (xmax-xmin)/n;
      double[] xdata = new double[n];
      // real and imaginary values alternate
      double[] zdata = new double[2*n];
      // test function is e^(-x*x/4)e^(i*mode*x) for x=[-pi,pi)
      int mode = 4;
      for(int i = 0;i<n;i++) {
         double a = Math.exp(-x*x/4);
         zdata[2*i] = a*Math.cos(mode*x);
         zdata[2*i+1] = a*Math.sin(mode*x);
         xdata[i] = x;
         x += dx;
      }
      frame.append(xdata, zdata);
      frame.setVisible(true);
      frame.setDefaultCloseOperation(javax.swing.JFrame.EXIT_ON_CLOSE);
   }
}
```

Figure 16.2 shows two representations of a quantum wave function. The real and imaginary representation displays the real and imaginary parts of the wave function $\Psi(x)$ by drawing two curves. In the amplitude and phase representation the vertical height represents the wave function magnitude and the color indicates phase. Note that the complex phase is oscillating, indicating that the wave function has a nonzero momentum expectation value, which is known as a *momentum boost*.