

Listing 10.3 The FieldLine class computes an electric field line using a Thread.

```
package org.opensourcephysics.sip.ch10;
import java.awt.Graphics;
import java.util.*;
import org.opensourcephysics.display.*;
import org.opensourcephysics.numerics.*;

public class FieldLine implements Drawable, ODE, Runnable {
    DrawingFrame frame;
    double[] state = new double[2]; // Ex and Ey for ODE
    ODESolver odeSolver = new RK45MultiStep(this);
    ArrayList chargeList; // list of charged particles
    Trail trail;
    double stepSize;
    boolean done = false;

    public FieldLine(DrawingFrame frame, double x0, double y0,
                    double stepSize) {
        this.stepSize = stepSize;
        this.frame = frame;
        odeSolver.setStepSize(stepSize);
        state[0] = x0;
        state[1] = y0;
        chargeList = frame.getDrawables(charge.class);
        trail = new Trail();
        trail.addPoint(x0, y0);
        Thread thread = new Thread(this);
        thread.start();
    }

    public double[] getState() {
        return state;
    }

    public void getRate(double[] state, double[] rate) {
        double ex = 0;
        double ey = 0;
        for(Iterator it = chargeList.iterator(); it.hasNext(); ) {
            Charge charge = (Charge) it.next();
            double dx = (charge.getX()-state[0]);
            double dy = (charge.getY()-state[1]);
            double r2 = dx*dx+dy*dy;
            double r = Math.sqrt(r2);
            if((r<2*stepSize)|| (r>100)) { // done if too close or too far
                done = true;
            }
            ex += (r==0) ? 0 : charge.q*dx/r2/r;
            ey += (r==0) ? 0 : charge.q*dy/r2/r;
        }
        double mag = Math.sqrt(ex*ex+ey*ey);
        rate[0] = (mag==0) ? 0 : ex/mag;
        rate[1] = (mag==0) ? 0 : ey/mag;
    }

    public void run() {
```

```
        int counter = 0;
        while(((counter<1000)&&!done)) {
            odeSolver.step();
            trail.addPoint(state[0], state[1]);
            if(counter%50==0) { // repaint every 50th step
                frame.repaint();
                try {
                    Thread.sleep(20); // give the event queue a chance
                } catch(InterruptedException ex) {}
            }
            counter++;
            Thread.yield();
        }
        frame.repaint();
    }

    public void draw(DrawingPanel panel, Graphics g) {
        trail.draw(panel, g);
    }
}
```

The FieldLine constructor saves a reference to the list of charges to calculate the electric field using (10.1). The loop in the run method solves the differential equation and stores the solution in a drawable trail. The loop is exited when the field line is close to a charge or when the magnitude of the field becomes too small. Because there are situations where the field line will never stop, this loop is executed no more than 1000 times.

The FieldLineApp program instantiates a field line when the user double clicks within the panel. Adding a charge or moving a charge removes all field lines from the panel. Study how the handleMouseAction allows the user to drag charges and to initiate the drawing of field lines. You are asked to modify this program in Problem 10.2.

Listing 10.4 The FieldLineApp program computes an electric field line when the user clicks within the panel.

```
package org.opensourcephysics.sip.ch10;
import java.awt.event.MouseEvent;
import org.opensourcephysics.controls.*;
import org.opensourcephysics.display.*;
import org.opensourcephysics.frames.DisplayFrame;

public class FieldLineApp extends AbstractCalculation implements
    InteractiveMouseHandler {
    DisplayFrame frame = new DisplayFrame("x", "y", "Field lines");

    public FieldLineApp() {
        frame.setInteractiveMouseHandler(this);
        frame.setPreferredMinMax(-10, 10, -10, 10);
    }

    public void calculate() {
        // remove old field lines
        frame.removeObjectsOfClass(FieldLine.class);
        double x = control.getDouble("x");
        double y = control.getDouble("y");
        double q = control.getDouble("q");
        Charge charge = new Charge(x, y, q);
```