Faraday simulation physics
(how it's implemented, how it works)

Chris Malley (cmalley@pixelzoom.com)

This document describes the physics-related bits of the Faraday simulation. Other implementation details should be obvious from reading the code or javadoc.

Terminology

B-field is another name for magnetic field.

Some quantities are referred to as **amplitude** (for example "voltage amplitude"). Amplitude represents how the value relates to the maximum value. For example if the voltage of an AC Power Supply is 20v, and its maximum voltage is 100v, then its voltage amplitude is 20/100 = 0.2. Throughout the implementation, amplitude is expressed as a value from 0...1 or -1...+1, depending on the quantity.

There are two basic types of "players" in the simulation: B-field **Producers** and B-field **Consumers**.

B-Field Producers

The simulation contains 3 B-field producers: **BarMagnet, Electromagnet and Turbine**.

All magnets (**BarMagnet**, **Electromagnet** and **Turbine**) can provide the B-field vector at a point of interest, relative to the magnet's location. In reality, the B-field decreases as a function of the distance cubed (exponent=3). But to make things look better in the simulation, you have the option of specify the exponent. The compass grid and field meter both uses exponent=3. The PickupCoil uses exponent=2.

The **BarMagnet** is based on a **dipole magnet**. See the javadoc in DipoleMagnet for details.

The **Electromagnet** is based on a **coil magnet** model. See the javadoc in CoilMagnet for details. It's voltage source can be either a **Battery** or an **ACPowerSupply**. The strength of the B-field produced by the electromagnet is proportional to the amplitude of the voltage in the voltage source and the number of loop in the coil. The current in the coil is proportional to the amplitude of the voltage source. Note that there is no model of resistance for the coil or voltage source.

The **Battery** is rather straightforward. It has a maximum voltage, and its voltage amplitude is varies by the user via a slider control.

The **AC Power Supply** has a configurable maximum voltage. The user varies the maximum voltage amplitude and frequency using sliders. The voltage amplitude varies over time.

The **Turbine** is based on the same dipole magnet model as the BarMagnet, and is in fact graphically represented as a rotating bar magnet, attached to a water wheel. The B-field produced by the turbine is proportional to the rotational speed of the water wheel.

B-Field Consumers

The simulation contains 4 B-field consumers: **Compass, CompassGrid, FieldMeter and PickupCoil**. All of these things can be influenced by only one magnet; there is no support for multiple magnets.

The **Compass** asks the magnet for the B-field vector at the Compass' location. The Compass' "behavior" determines how it reacts to the B-field. There are 3 types of behavior, implemented as inner classes of the Compass class. "Simple" behavior causes the compass needle to immediately align itself with the B-field direction. "Incremental" behavior causes the compass needle to animate the rotation required to align itself with the B-field direction. "Kinematic" behavior is the most "real" looking and uses a **Verlet algorithm for rotational kinematics**. This causes the compass needle to rotate and exhibit inertia, angular acceleration, and angular velocity; the needle will appear to wobble as it reaches equilibrium.

The **CompassGrid** is a representation of a magnet's B-field. It is composed of a bunch of compass needles, located on an equally-space grid of points in 2D space. The grid asks the magnet for the B-field vector at each of the grid points, and immediately aligns the corresponding needles with the B-field direction. Points with field strength below a configured threshold are ignored, and their corresponding needles are not rendered.

The **FieldMeter** asks the magnet for the B-field vector at the meter's location. It then displays the vector components: magnitude, X, Y, angle.

The **PickupCoil** is the most compilcated B-field consumer, and the place where

Faraday's Law is implemented. (We won't be describing Faraday's Law here; consult your physics textbook.) The B-field is calculated using a set of sample points across the center loop of the coil. (The number of sample points is configurable, but 9 seems to be a good number.) The B-field values at the sample points are averaged to compute the flux in one loop of the coil, then multiplied by the number of loops. If there is a change in flux, than an emf is induced. The current in the coil (magnitude and direction) is a function of the induced emf. The PickupCoil can have one of two indicators attached (Lightbulb or Voltmeter) which display the current in the coil. Both of these indicators can be scaled, so that they react nicely in situations with various magnet strengths.

The **Lightbulb**'s intensity is proportional to the amplitude of the current in the pickup coil.

The **Voltmeter**'s needle deflection is proportial to the amplitude of the current in the pickup coil. The Voltmeter uses an ah hoc algorithm that makes the needle wobble around the zero point.

Miscellaneous

All parameterization of the simulation is done via Java constant definitions. FaradayConfig contains all of the global parameters for the simulation. Parameters that are specific to a module are specified in that module.