More on the guessing technique for the damped harmonic oscillator

Last class we guessed that solutions of the form \( y(t) = e^{\lambda t} \) might be solutions to the damped harmonic oscillator

\[
m \frac{d^2 y}{dt^2} + b \frac{dy}{dt} + ky = 0.
\]

In fact, we saw that \( y(t) = e^{\lambda t} \) is a solution if and only if

\[
m\lambda^2 + b\lambda + k = 0.
\]

This equation is called the characteristic equation of the harmonic oscillator, and the polynomial \( p(\lambda) = m\lambda^2 + b\lambda + k \) is its characteristic polynomial.

**Example.** Consider the harmonic oscillator

\[
\frac{d^2 y}{dt^2} + 3 \frac{dy}{dt} + 2y = 0.
\]

Its characteristic equation is
Let’s plot these solutions with HPGSystemSolver. What are the corresponding solution curves and component graphs?

Euler’s method for a system

We can use the vector field for a system to produce numerical approximations for the solutions.

Example. Consider the IVP

\[
\begin{align*}
\frac{dx}{dt} &= -y \\
\frac{dy}{dt} &= x - y
\end{align*}
\]

\((x_0, y_0) = (2, 0)\).

The EulersMethodForSystems tool demonstrates the method. We pick a large step size \(\Delta t = 0.5\) so that we can see the method in action.
Now let’s derive the general equations for Euler’s method for an autonomous IVP of the form

\[
\frac{dx}{dt} = f(x, y) \\
\frac{dy}{dt} = g(x, y) \\
(x(t_0), y(t_0)) = (x_0, y_0).
\]
Euler’s method for systems is just as easy to program as Euler’s method for equations. Once again here’s how we can program it with a spreadsheet.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

There are two spreadsheets posted on the course web site—one for the example above and one for the following example.

**Example.** Consider the predator-prey system

\[
\frac{dR}{dt} = R - 0.2RF \\
\frac{dF}{dt} = -0.3F + 0.1RF
\]

along with the initial condition \((R_0, F_0) = (1, 2)\). Using the spreadsheet on the web site, we see that Euler’s method has trouble approximating periodic solutions.

**HPGSystemSolver** uses a more sophisticated fixed-step-size algorithm called the Runge-Kutta method. It usually works better than Euler’s method, but there are equations for which any fixed-step-size algorithm is not appropriate.
Existence and Uniqueness Theory for Systems

There is an existence and uniqueness theorem for systems just like the theorem for equations.

**Existence and Uniqueness Theorem.** Let

\[
\frac{dY}{dt} = F(t, Y)
\]

be a system of differential equations. Suppose that \( t_0 \) is an initial time and \( Y_0 \) is an initial value. Suppose also that the function \( F \) is continuously differentiable. Then there is an \( \epsilon > 0 \) and a function \( Y(t) \) defined for \( t_0 - \epsilon < t < t_0 + \epsilon \) such that

\[
\frac{dY}{dt} = F(t, Y(t)) \quad \text{and} \quad Y(t_0) = Y_0.
\]

In other words, \( Y(t) \) satisfies the initial-value problem. Moreover, for \( t \) in this interval, this solution is unique.

There is an important consequence of the Uniqueness Theorem for autonomous systems: Consider the metaphor of the parking lot.

Given the autonomous system

\[
\frac{dY}{dt} = F(Y).
\]

Let \( Y_0 \) be an initial condition such that \( Y_1(t) \) is a solution that satisfies \( Y(t_1) = Y_0 \) and \( Y_2(t) \) is another solution that satisfies \( Y(t_2) = Y_0 \). Then

\[
Y_2(t) = Y_1(t - (t_2 - t_1)).
\]
Example. Consider the second-order equation

\[ \frac{d^2y}{dt^2} + y = 0, \]

which is equivalent to the system

\[ \frac{dy}{dt} = v \]
\[ \frac{dv}{dt} = -y. \]

Note that

\[ Y_1(t) = \begin{pmatrix} \cos t \\ -\sin t \end{pmatrix} \quad \text{and} \quad Y_2(t) = \begin{pmatrix} \sin t \\ \cos t \end{pmatrix} \]

are both solutions to the system. How are \( Y_1(t) \) and \( Y_2(t) \) related?

There is an animation on the web site that illustrates this phenomenon.

Here is an informal restatement of this consequence of uniqueness:

For an autonomous system, if two solution curves in the phase plane touch, then they are identical.