Second order ODEs - variation of parameters

Prof. Joyner, 9-3-2007

Consider an ordinary constant coefficient non-homogeneous 2nd order linear differential equation,

\[ ay'' + by' + cy = F(x) \]

where \( F(x) \) is a given function and \( a, b, \) and \( c \) are constants. (For the method below, \( a, b, \) and \( c \) may be allowed to depend on the independent variable \( x \) as well.) Let \( y_1(x), y_2(x) \) be fundamental solutions of the corresponding homogeneous equation

\[ ay'' + by' + cy = 0. \]

The method of variation of parameters is originally attributed to Joseph Louis Lagrange (1736-1813) \cite{L}. It starts by assuming that there is a particular solution in the form

\[ y_p(x) = u_1(x)y_1(x) + u_2(x)y_2(x), \]

where \( u_1(x), u_2(x) \) are unknown functions \cite{V}.

In general, the product rule gives

\[
(fg)' = f'g + fg',
\]

\[
(fg)'' = f''g + 2f'g' + fg'',
\]

\[
(fg)''' = f'''g + 3f''g' + 3f'g'' + fg''',
\]

and so on, following Pascal’s triangle,

\[
\begin{array}{c}
1 \\
1 1 \\
1 2 1 \\
1 3 3 1,
\end{array}
\]

and so on.

\footnote{These notes licensed under Attribution-ShareAlike Creative Commons license, \url{http://creativecommons.org/about/licenses/meet-the-licenses}.}
Using SAGE, this can be check as follows:

```python
sage: t = var('t')
sage: x = function('x', t)
sage: y = function('y', t)
sage: diff(x(t)*y(t),t)
x(t)*diff(y(t), t, 1) + y(t)*diff(x(t), t, 1)
sage: diff(x(t)*y(t),t,t)
x(t)*diff(y(t), t, 2) + 2*diff(x(t), t, 1)*diff(y(t), t, 1) + y(t)*diff(x(t), t, 2)
sage: diff(x(t)*y(t),t,t,t)
x(t)*diff(y(t), t, 3) + 3*diff(x(t), t, 1)*diff(y(t), t, 2) + 3*diff(x(t), t, 2)*diff(y(t), t, 1) + y(t)*diff(x(t), t, 3)
```

By assumption, \( y_p \) solves the ODE, so

\[ ay''_p + by'_p + cy_p = F(x). \]

After some algebra, this becomes:

\[ a(u'_1y_1 + u'_2y_2) + a(u'_1y'_1 + u'_2y'_2) + b(u'_1y_1 + u'_2y_2) = F. \]

If we assume

\[ u'_1y_1 + u'_2y_2 = 0 \]

then we get massive simplification:

\[ a(u'_1y'_1 + u'_2y'_2) = F. \]

Cramer’s rule says that the solution to this system is

\[
\begin{align*}
  u'_1 &= \frac{\det \begin{pmatrix} 0 & y_2 \\ F(x) & y'_2 \end{pmatrix}}{\det \begin{pmatrix} y_1 & y_2 \\ y'_1 & y'_2 \end{pmatrix}}, \\
  u'_2 &= \frac{\det \begin{pmatrix} y_1 & 0 \\ y'_1 & F(x) \end{pmatrix}}{\det \begin{pmatrix} y_1 & y_2 \\ y'_1 & y'_2 \end{pmatrix}}.
\end{align*}
\]

Note that the Wronskian of the fundamental solutions \( W(y_1, y_2) \) is in the denominator.

Solve these for \( u_1 \) and \( u_2 \) by integration and then plug them back into \( y_p \) to get your particular solution.
Example 1. Solve

\[ y'' + y = \tan(x). \]

\text{soln: The functions } y_1 = \cos(x) \text{ and } y_2 = \sin(x) \text{ are fundamental solutions with Wronskian } W(\cos(x), \sin(x)) = 1. \text{ The Cramer’s rule formulas above become:}\\

\[ u'_1 = \frac{\det \begin{pmatrix} 0 & \sin(x) \\ \tan(x) & \cos(x) \\ 1 \end{pmatrix}}{1}, \quad u'_2 = \frac{\det \begin{pmatrix} \cos(x) & 0 \\ -\sin(x) & \tan(x) \\ 1 \end{pmatrix}}{1}. \]

Therefore,

\[ u'_1 = -\frac{\sin^2(x)}{\cos(x)}, \quad u'_2 = \sin(x). \]

Therefore, using methods from integral calculus, \( u_1 = -\ln |\tan(x) + \sec(x)| + \sin(x) \) and \( u_2 = -\cos(x) \). Using SAGE, this can be check as follows:

```
sage: integral(-sin(t)^2/cos(t),t)
-log(sin(t) + 1)/2 + log(sin(t) - 1)/2 + sin(t)
sage: integral(cos(t)-sec(t),t)
-sin(t) - log(tan(t) + sec(t))
sage: integral(sin(t),t)
-cos(t)
```

As you can see, there are other forms the answer can take. The particular solution is

\[ y_p = (-\ln |\tan(x) + \sec(x)| + \sin(x)) \cos(x) + (-\cos(x)) \sin(x). \]

The homogeneous (or complementary) part of the solution is

\[ y_h = c_1 \cos(x) + c_2 \sin(x), \]

so the general solution is
\[ y = y_h + y_p = c_1 \cos(x) + c_2 \sin(x) \\
+(- \ln |\tan(x) + \sec(x)| + \sin(x)) \cos(x) + (-\cos(x)) \sin(x). \]

**Using SAGE**, this can be carried out as follows:

```python
sage: SR = SymbolicExpressionRing()
sage: MS = MatrixSpace(SR, 2, 2)
sage: W = MS([[\cos(t), \sin(t)],[\diff(\cos(t), t), \diff(\sin(t), t)]];
sage: W

[ cos(t) sin(t)]
[-sin(t) cos(t)]
sage: det(W)
\sin(t)^2 + \cos(t)^2
sage: U1 = MS([[0, \sin(t)],[\tan(t), \diff(\sin(t), t)]])
sage: U2 = MS([[\cos(t), 0],[\diff(\cos(t), t), \tan(t)]])
sage: integral(det(U1)/det(W), t)
-\log(\sin(t) + 1)/2 + \log(\sin(t) - 1)/2 + \sin(t)
sage: integral(det(U2)/det(W), t)
-\cos(t)
```

**Exercise**: Use SAGE to solve \( y'' + y = \cot(x) \).

**References**

