Laplace Transforms in Maple

Kurt Bryan and SIMIODE

To begin, load in the "inttrans" package that contains commands for computing Laplace transforms, inverse Laplace Transforms, and the definition of the Heaviside and Dirac delta functions.

> $with(inttrans):$

The Laplace Transform: To Laplace transform a function, e.g., f(t) = t^2, execute

> $laplace(t^2, t, s)$

More generally, to compute the Laplace transform of f(t) execute "laplace(f(t),t,s)". The second argument "t" indicates the independent variable in f(t) and the third argument "s" indicates that the transform should have independent variable "s".

The Inverse Laplace Transform: The command is "invlaplace", e.g.,

> $invlaplace\left(\frac{2}{s^3}, s, t\right)$

Example: Solving a first-order ODE: Consider the ODE u'(t) = -2*u(t) + t with initial data u(0) = 3.

Step 1: To solve using Laplace transforms (explicitly carrying out all the steps), first define the ODE

> $ode := u'(t) = -2 \cdot u(t) + t$

Step 2: Laplace transform both sides of the ODE, which can be done as

> $lapode := laplace(ode, t, s)$

Maple transformed both sides of the ODE, and knows the rule for transforming derivatives. Maple uses the notation "laplace(u(t),t,s)" for the Laplace transform of u(t). Let use replace this with the notation U (s) (not necessary, just more aesthetically pleasing) and also substitute in the initial data u(0) = 3.

> $lapode2 := subs(u(0) = 3, laplace(u(t), t, s) = U(s), lapode)$

Step 3: We can now solve for U(s), e.g.,

> $Usol := solve(lapode2, U(s))$

Step 4: The last step is to inverse Laplace transform "Usol" to find u(t), which we do as

> $usol := invlaplace(Usol, s, t)$

This is the solution to the ODE. An easy alternative is

> $usol := dsolve(\{ode, u(0) = 3\}, u(t), method = laplace)$

in which Maple uses the Laplace transform and does all the dirty work.

Example: A second-order ODE. Consider the second-order ODE

> $ode := u''(t) + 4 \cdot u'(t) + 3 \cdot u(t) = \sin(t)$

with u(0) = 1 and u'(0) = 2. To solve follows the steps below.

Step 1: Laplace transform both sided of the ODE

> $lapode := laplace(ode, t, s)$

Step 2: Substitute in the initial data and (optionally) use U(s) to replace "laplace(u(t),t,s)":

> $lapode2 := subs(u(0) = 1, u'(0) = 2, laplace(u(t), t, s) = U(s), lapode)$

Step 3: Solve for U(s) as

> $Usol := solve(lapode2, U(s))$

Step 4: Inverse transform to find u(t)

> $invlaplace(Usol, s, t)$

The Heaviside and Dirac Delta Functions: The Heaviside function in Maple is

> $\mathrm{Heaviside}(t)$

For example,

> $laplace(\mathrm{Heaviside}(t), t, s)$

A plot of the Heaviside function:

> $plot(\mathrm{Heaviside}(t), t = -2..2, thickness = 3, color = red)$

although a slightly more representative graph is obtained by telling Maple not to draw vertical lines at discontinuities:

> $plot(\mathrm{Heaviside}(t), t = -2..2, thickness = 3, color = red, discont = true)$

The Dirac delta function is given by

> $\mathrm{Dirac}(t)$

Example: The mass in a damped spring-mass system with mass 1 kg, damping constant 2 newtons per meter per second, and spring constant 5 newtons per meter is subjected to a constant force of 2 newtons for 0 < t < 4 seconds, at which point the force drops to zero. At time t = 6 seconds the mass is subjected to a hammer blow with total impulse 8 newton-meters. Find the motion of the mass if the mass starts at rest and at equilibrium, and plot this motion for 0 < t < 20 seconds.

The relevant ODE is

> $ode := u''(t) + 2\cdot u'(t) + 5\cdot u(t) = 2 - 2\cdot \mathrm{Heaviside}(t-4) + 8\cdot \mathrm{Dirac}(t-6)$

We can solve exactly as in the second order example above, or simple execute

> $usol := dsolve(\{ode, u(0) = 0, u'(0) = 0\}, u(t), method = laplace)$

The solution might return as appearing to be complex-valued, but we can make sure it is expressed in real-valued terms with

> $usolreal := evalc(usol)$

A plot (we need to use "rhs" to pick off the right side of "usol")

> $plot(rhs(usolreal), t = 0..20)$

>