Use of Distributed Delay in PML

Michael Dunlavey, Shuhua Hu
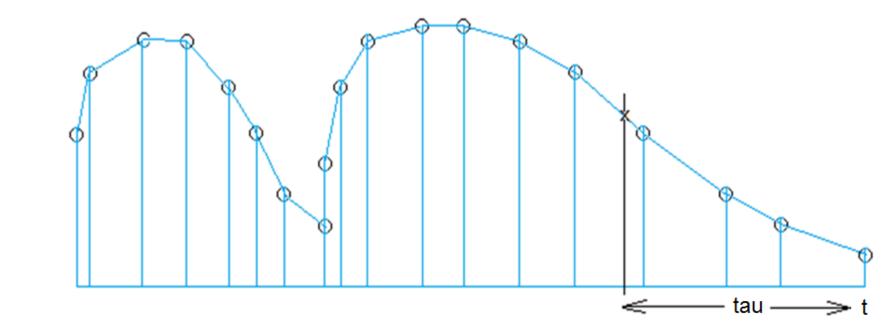Certara/Pharsight® Corporation

Pharsight

Objective

ODEs incorporating delay are called Delay Differential Equations (DDEs). They can be thought of as convolving an input function with a probability density function(PDF). In the case where the PDF is a delta function, we call it "discrete delay". This is to distinguish it from "distributed delay"(e.g. see [1]), in which the convolution is not with a delta function, but with a continuous probability density function. This can be used to model such things as absorption delay and delayed drug effect in a realistic manner.

Methods

A compartment-modeling statement was added to the PML language, and a previously existing function was extended, to incorporate distributed delay for the common case of a Gamma distribution. The Gamma distribution is useful in that it has a scale parameter and a shape parameter. It models absorption and delayed drug effect well,and it has Exponential and Erlang distributions as special cases. If the optional shape parameter is not given, or if it exceeds a threshold, discrete delay is assumed.

The `delay` function in PML was extended to the following syntax:

```
delay(y, tau [, shape=c][, hist=h])
```

where *y* is the input to be delayed, and *tau* is the non-negative mean delay time. If the optional argument shape=*c* is given, *c* is the positive shape parameter of the Gamma distribution. For the given delay time, the larger *c* is, the narrower is the distribution. Below is a plot of the Gamma distribution for various values of the shape parameter:*

At values of 100 or more, discrete delay is assumed. If the optional argument hist=*h* is given *h* is the value to be given to *y* when *t* is less than 0.

It was desired to have PK models with delayed behavior, such as to model absorption. The simple delay function given above cannot do this because the expression *y* cannot contain dosing information. So a statement was added to the language:

```
delayInfCpt(A, tau
            [, shape=n]
            [, in=inflow]
            [, out=outflow])
```

meaning "delayed inflow compartment". *A* is the name of the compartment, and the above is *roughly* equivalent to this differential equation:

```
deriv(A = delay(inflow + infusionrate, tau, shape=n+1)
        + outflow)
```

In other words, the *inflow* is delayed, as is the dosing infusion rate (which is a variable not available in the PML language). *A* can receive doses, such as boluses and infusions, and all of those are subject to discrete or distributed delay. If a shape of 0 is given, a shape of 1 is used, so it has identical semantics to the transit statement, with *tau* meaning the mean transit time, and *n* meaning the number of transit stages. (The transit statement has accuracy problems when *n* is small and fractional and doses are close together. It also has performance problems when *n* becomes large.)

Implementation

Wherever delay appears in the model code, a structure called a `delayTable` is created. It contains two tabular functions of time. The first is a history of the *y* values, recorded at points during the execution of the ODE solver. The second is a record of doses, either bolus doses of a given amount, or infusions of a given amount and rate. The delay function convolves input *y* with the distribution. The `delayInfCpt` statement does

* Images courtesy michael.tomashevskiy@ontarget-group.com

that, plus it convolves the dosing history with the distribution, adding that to the first convolution.

In the case of discrete delay, the prior value of *y* is simply looked up by linear interpolation:
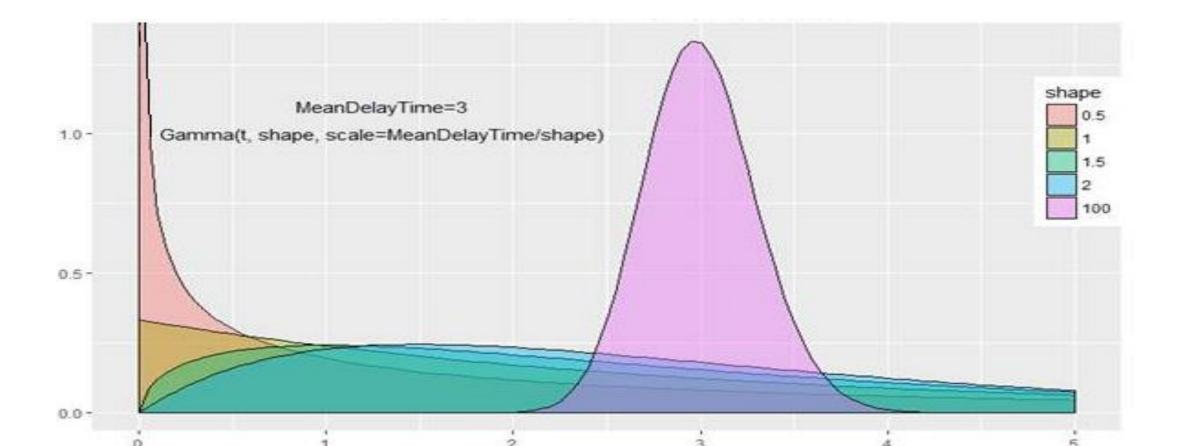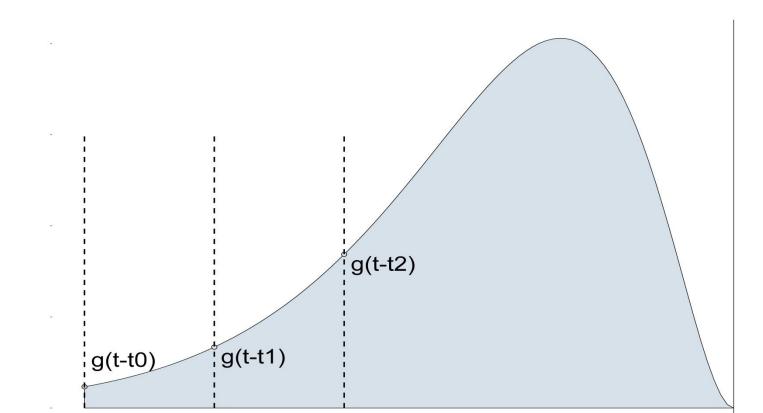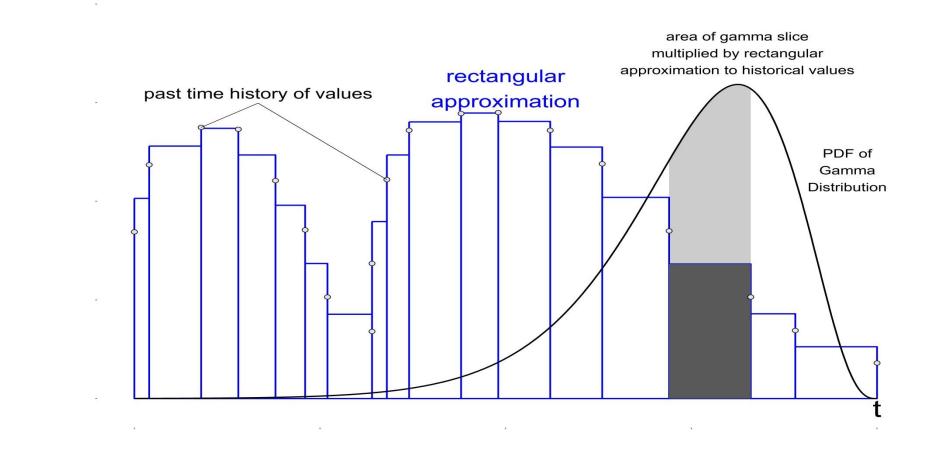

(Higher-order interpolation is not used because the recorded points are fairly close together, and because there are discontinuities at change-points such as doses.) Infusions are handled the same way. Bolus doses are handled as short infusions.

In the case of distributed delay, convolution is performed.

$$\int_{-\infty}^{t} g(t-x)S(x)\,dx$$

Here t is the current time, *S(x)* is the function of time to be delayed, and *g* is the probability density function (PDF) of the distribution. If *S* consists of multiple components, they can each be considered individually and added together.

In the case of bolus doses of amounts D_i at times t_i, *S* is a Dirac delta function, and the convolution is the sum of $D_i g(t-t_i)$.

In the case of an infusion, *S* is a boxcar (rectangular) function $R\ boxcar(t_0, t_1)$, and the convolution is $R\big(G(t-t_0) - G(t-t_1)\big)$ where *G* is the cumulative distribution function (CDF) of *g*.

The history table *y* can be approximated as a series of boxcar functions:

Another method is to use trapezoidal approximations.

Results

Predictions by the delay functions were compared against predictions obtained by superposition, with agreement within four or more decimal digits, over a variety of dosing histories and time scales.

Conclusions

Further work will include extension to other distributions such as Weibull, Inverse Gaussian, and log-normal distributions, and it will include processing to accomplish steady-state.

References

[1] Smith, H., An Introduction of Delay Differential Equations with Applications to the Life Sciences, Springer, New York, 2011

Contact

Michael R. Dunlavey, PhD, Certara/Pharsight Corporation, Ph: +1 781.449.2719, Email: Mike.Dunlavey@Certara.com