Explaining the error in the IT calculations

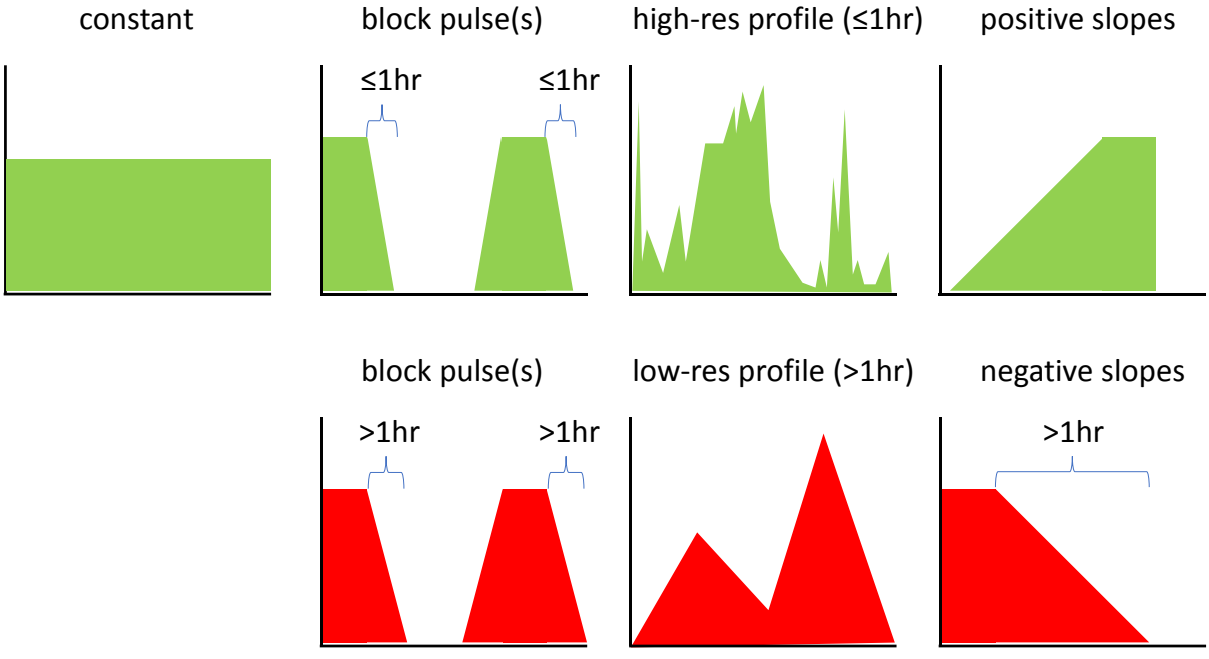
Tjalling Jager (DEBtox Research), document version 19 May 2020

The essentials

The IT calculations for time-varying exposure used a shortcut to increase speed. This shortcut, however, was a mistake as it does not work in all situations. This error only affects IT, and in most practical cases, the error made will fortunately be small. However, it is important to identify the cases where this error becomes relevant. First the cases where this error always has *little* or *no* effect:

- The error does not affect SD at all.
- The error does not affect IT for constant exposure.
- The error does not affect IT for scenarios with on/off pulse exposure.
- The error does not affect IT for scenarios where exposure only *increases* in time.
- The error has very little effect when the exposure scenario is specified with high temporal resolution (as is for example the case with FOCUS profiles).
- The error has very little effect on validation.
- The error has no effect on the calculated damage, as this is calculated analytically.
- The error has no effect on the plots for damage and survival; these will correctly reflect the model parameters.
- The error has little effect when there are many observations on survival over time in calibration.

The error will thus influence IT calibration with time-varying exposure, for specific types of exposure scenarios with low temporal resolution. Further, it affects LPx predictions for IT, for specific exposure profiles with low temporal resolution. Temporal resolution is the key issue here. Validation and plotting are hardly affected since the underlying calculations always use a default high-resolution time vector (hourly resolution, and a minimum of 100 points), so these results will always reflect the model parameters correctly. Exposure scenarios specified with high temporal resolution will also minimise the error. For example, FOCUS profiles are specified on an hourly basis; testing so far identified all errors on LPx <0.5%. As illustration, below a summary of the types of scenarios that lead to no or small errors (green) and those that could lead to appreciable errors (red):



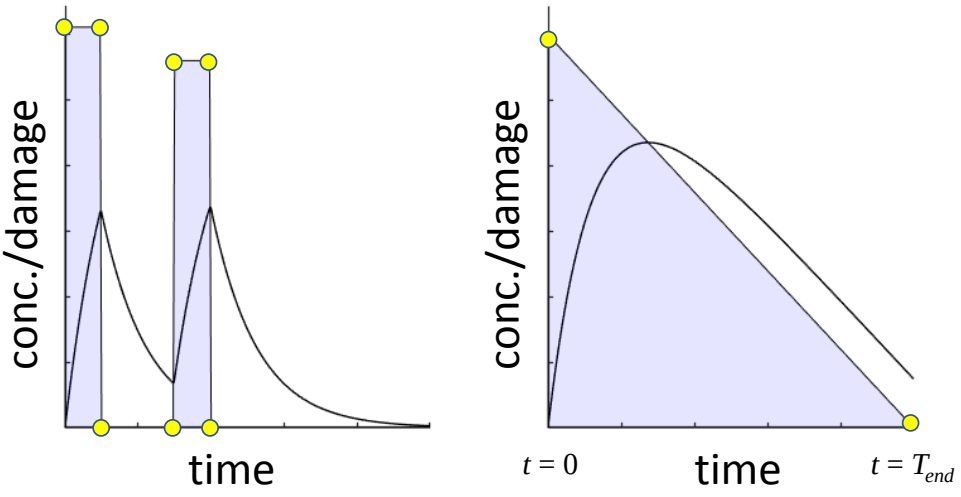
The primary concern is thus for IT calibrations and predictions with time-varying exposure, for specific low-resolution scenarios. It is difficult to be more precise, as the degree of error will depend on details of the exposure profile and the value of the model parameters. However, given that in testing, an hourly resolution always led to very small errors, a good rule-of-thumb will be to scrutinise the model output when the exposure scenario contains episodes with a linear decreases, with a courser than hourly resolution. In affected calibrations, the resulting model fit will not be the true best fit, and the model parameters will be biased. In affected predictions, the LPx estimates will not be correct. The plots will correctly reflect the model parameters and the LPx, but they would be biased as well since the underlying parameters/metrics will be biased themselves.

Note that the exposure profiles in the validation stage can be of any type or shape; the calculations in this stage are always performed on a fine time scale (hourly resolution) so they are hardly affected.

More details

To better understand this issue, and the possible workarounds, some more explanation of the code is needed. For SD, a fine time vector is needed for all calculations, since hazard needs to be numerically integrated. However, for IT, survival at a certain time point is fully determined by the maximum amount of damage reached until that time point. The code was simplified based on the assumption that the maximum level of damage always occurs at a peak in the exposure concentration (i.e., one of the time points used to define the exposure scenario). This is true for block pulses, but unfortunately not in general. Since users of openGUTS can enter any scenario that they like, this can lead to errors.

This is illustrated in the plot below. For instant block pulses (left plot), maximum damage occurs at a point that is part of the scenario definition (yellow points). The right plot shows an exposure profile with only two points: a high concentration at t=0 and a zero concentration at t=T_{end}. Damage follows one-compartment kinetics, and the damage peak clearly does not coincide with either point in the exposure scenario. The IT code only calculates survival at the yellow points (and the points at which there are observations on survival), so that leads to a large error on the maximum damage achieved at the relevant time points (also depending on the values of the GUTS model parameters).

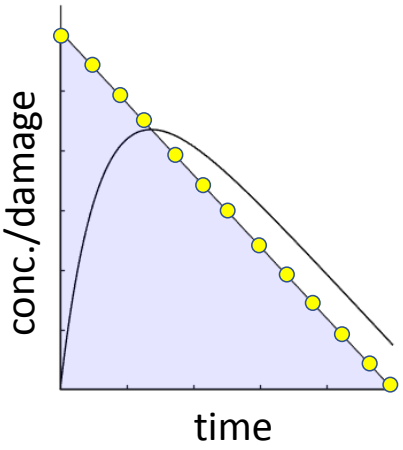


Workarounds

For block pulses, it is best to specify them as instantaneous: define the exposure scenario with two exposure concentrations at the same time point. An example can be found in the example file `propiconazole_pulsed_renewals.txt` distributed with the software. This is better than adding time points around the point of transfer of the animals (which is done in the other example

file for the same data set: `propiconazole_pulsed_linear.txt`). In the latter case, the error is small (as the negative slopes are occurring over the course of 1 hour) but noticeable.

For scenarios that include linear decreases with low resolution, it is possible to add more points to the exposure scenario definition as in the example below. This forces damage to be explicitly calculated at these extra time points as well. This is rather annoying, but can be done with help of Excel. Note that damage is also calculated on the time points where there are survival observations.



In the next update of the Matlab version of openGUTS (to version 1.1), this error will be repaired. When in doubt, the Matlab version can thus be used, in the near future. The repair is rather simple, and therefore this will be taken care of in the next update of the standalone as well. However, this repair will lead to a substantial increase in the calculation time for IT. Note that the BYOM package for GUTS does *not* use this shortcut, so its calculations are correct.