Technical background LGUTS and LPop

Hans Baveco

LEffectModel version 20211111, documentation version 20211111

*Models*

LGUTS : reduced GUTS at catchment scale. GUTS-RED-IT: reduced GUTS, Individual Tolerance version; GUTS-RED-SD: reduced GUTS Stochastic Death version.

LPop: DEB-based population model at catchment scale, parameterized for *Asellus aquaticus*. With GUTS-RED-IT or GUTS-RED-SD. Abj DEB version with population regulation through density-dependent mortality.

*Installation*

Unpack LEffectModel.zip at the desired location (zip contains all content under a LEffectModel directory). Directory structure should resemble the one in figure 1.

```
LEffectModel
+---ecotalk
|   \---LPopSDModelSystem
|       \---x1
|           \---x1s1
+---ETInput
|   +---AmP
|   +---CatchmentGUTSITModelSystem
|   |   +---data
|   |   +---maps
|   |   |   \---shapes
|   |   |       \---reachlist_shp
|   |   \---parameters
|   +---CatchmentModelSystem
|   |   \---data
|   +---LPopITModelSystem
|   |   +---data
|   |   +---maps
|   |   |   \---shapes
|   |   |       \---reachlist_shp
|   |   \---parameters
+---LGUTS
+---LPOP
```

*Figure 1. The main directory tree structure, showing the input and output for some of the models (obtained by executing CMD /c "Tree /A > dirtree.txt" in the file explorer, where the /A is for 8 bit ascii characters. NB adding /F prints all the files as well).*

Test concentration files are provided only for one year, in \ETInput\CatchmentModelSystem\data\ directory. To be able to run a test over multiple years, simply copy the rummen_2007.csv or the rummen_2007.msgpack files to other years (rummen_2008...rummen_2010, etc). Make sure concentration data files are defined for each year with applications (for 'applicationYear' with LGUTS and for 'startApplicationYear' to 'endApplicationYear' for LPop).

*Running*

The GUTS-IT (individual tolerance) model can be run with the supplied concentration data simply by starting the batch file LGUTSIT.bat; the GUTS-SD (stochastic death) version by LGUTSSD.bat.

A margin of safety analysis in which concentrations are multiplied by a series of multiplication factors, can be run be starting the batch file LGUTSIT_MoS.bat or LGUTSSD_MoS.bat.

The LPop population model with individual GUTS-IT submodels can be run by starting the batch file LPopIT.bat; the population model with GUTS-SD submodels by LPopSD.bat.

A margin of safety analysis in which concentrations are multiplied by a series of multiplication factors, can be run be starting the batch file LPopIT_MoS.bat or LPopSD_MoS.bat.

*How does it work*

A Smalltalk image (something similar to an R image) is run by starting a VM (virtual machine) with the name of the image provided as an argument, e.g. by

```
Squeak.exe LEffectModel.image
```

This opens the image (LEffectModel), with the models contained in the programming environment. In the context of the Landscape Model, interactive use of the Smalltalk environment via the GUI is not required. Instead the environment is used as a scripting engine, executing the Smalltalk code in a small user-defined text file. The name of this file is supplied to the VM as a command-line argument. As it contains Smalltalk code, its extension is by convention set to *st*. Its content (code) will be processed immediately after starting up the image. Thus with

```
Squeak.exe LEffectModel.image LGUTSIT.st
```

(as found in file LGUTSIT.bat) the content of LGUTSIT.st is executed:

```
ModelIO invalidateRootDirectories.
CatchmentConcDataBase removeAllDataBases.
(ModelProject onModel: #CatchmentGUTSITModelSystem) runModelProjectForeground.
Smalltalk quitPrimitive
```

which basically makes sure that (line 1) the current directories for input and output are used (see below), (line 2) no left-over concentration data files are used, (line 3) the model runs in the foreground (although the GUI opens it will not be accessible as it is running in a lower-priority process), and (line 4) the environment quits when finished.

*Common requirements*

LGUTS and LPop expect:

- geo-data in ESRI shapefile format to build up the spatial structure,
- hourly concentration data in a csv or msgpack file,
- model coefficients model in a csv file,
- simulation control parameters in a csv file.

All input data, except concentration data, are expected under a generic \ETInput\ directory in a sub-directory defined by the model name (e.g. \ETInput\CatchmentGUTSSDModelSystem).

Geo data

In \maps\shapes\ the (polylineZ) shapefile ReachList_shp is used to define spatial structure of the Rummen catchment. The shapefile's dbf should have an attribute named 'key' that defines the id of the reach (an integer). This id is also expected in the concentration input files as an identifier of the reach. Output for the reaches is always in the order of the features in the shapefile (dbf). As an extra check, a *reach_order.txt* file is produced with the actual order.

Coefficients

Model coefficients, for LGUTS mainly the set of GUTS coefficients, are defined in a file in csv format in the \parameters\ sub-directory. The default file name is model name with *_coefs* appended to it. See table 1 for an example of a coefficients file for the `CatchmentGUTSSDModelSystem` and table 2 for one for `LPopSDModelSystem`. As one can see, the LGUTS coefficients are also required in the LPop model. The `conversionToGutsFactor` parameter is needed only when concentration data are based on a different unit than the GUTS coefficients, e.g. a factor 10^6 is needed when concentration is supplied in mg/l while GUTS expects concentrations in ng/l. Note that the first term in each row with a coefficient value should not be modified. It refers to a method selector used in the model (a case-sensitive symbol).

Simulation control parameters are defined in a similar csv file, named *_control.csv*. In a following section with model-specific details, the coefficients and control parameters for LGUTS and LPop separately are explained in more detail.

Concentration data

Concentration data files are given per calendar year, and are e.g. named Rummen_2007, Rummen_2008 etc. They contain the hourly concentrations per reach (row). First value in the row is the Reach id. Then 24 * 365 (or 366 in case of a leap year) concentration values follow. Two formats can be used: csv or Message Pack (msgpack) format. The option is set in the control parameters file (Table 3 and 4). Concentrations are expected in nanogram per liter. If this is not the case, a conversion factor can be defined in the coefficients file. Note that, to avoid copying the very large concentration files, they are not in the specific model's input directory, but instead in a common one: \ETInput\CatchmentModelSystem\data\ where they can be used by all models and model versions.

Output

Model results are written to the standard 'ecotalk' directory, in a sub-directory defined by the model (script) name, in a hierarchy of directories that reflects the applied structure of 'experiments' (x) and 'scenarios' (s). When we run the model only once, for the provided concentration data and model coefficients, there will be only 1 experiment consisting of 1 scenario, and the output for the single scenario ends up in \x1\x1s1\. In a Monte Carlo approach, were several replicate runs (r) of the same scenario are performed, output files are prefixed by experiment, scenario and run id, e.g. x1s1r1, x1s1r2, etc.

In a margin of safety analysis, the simulations with the different multiplication factors are represented by different scenarios, and their output ends up in the scenario directories, e.g., \x1\x1s1\, \x1\x1s2\, etc. For LGUTS the results for all scenarios are collected in the experiment directory, e.g., \x1\.

*Table 1. Example of a parameter definition file (CatchmentGUTSSDModelSystem_coefs.csv), as it appears in a plain text editor.*

```
Component,model-dependent,inhabitantClass
kd:,0.28512,dominant rate constant [1/d]
```

```
hb:,0,background hazard rate [1/d]
z:,0.05507,threshold concentration [ng/L]
b:,0.06276,killing rate [L/(ng*d)]
Component,model-dependent,landscapeClass
conversionToGutsFactor:,1000000.0,mg/L to nanogram/L
```

*Table 2. Example of a parameter definition file (LPopSDModelSystem_coefs.csv), as it appears in a plain text editor.*

```
Component,model-dependent,inhabitantClass
minClutchSize:,14,minimum clutch size [ind]
useAgeing:,0,switch DEB aging mortality on (1) or off (0)
backgroundMortality:,0.00500,background mortality rate [d-1]
muDD:,0.000025,(default 0.000010) density-dependent mortality rate [m2 ind-1 d-1]
kd:,0.28512,dominant rate constant [1/d]
hb:,0,background hazard rate [1/d]
z:,0.05507,threshold concentration [ng/L]
b:,0.06276,killing rate [L/(ng*d)]
Component,model-dependent,landscapeClass
envTav:,10.0,average temperature parameter of forcing function [oC]
envTamp:,10.0,amplitude temperature fluctuations parameter [oC]
envTminShift:,31.0,shift forward of daynr with lowest temperature [d]
conversionToGutsFactor:,1000000.0,mg/L to nanogram/L
migrationProb:,0.05,per individual probability of migration [d-1]
downStreamProb:,0.667,probability of a migrating individual to move downstream
```

*Table 3. Contents of the control parameter file (CatchmentGUTSSDModelSystem.csv).*

```
applicationYear:,2007,year of pesticide application
verbose:,0,survival output per day (1) or end of the year only (0)
useCSV:,0,use the slow csv input format (1) or much faster msgpack format(0)
stepsInHr:,2,the number of steps within 1 hourly time step for the GUTS simulation
```

*Table 4. Contents of the control parameter file (LPopSDModelSystem.csv).*

```
startYear:,2005,start year of the simulation
endYear:,2007,last year of the simulation
startApplicationYear:,2007,start year of pesticide application
endApplicationYear:,2007,last year of pesticide application
useCSV:,0,use the slow csv input format (1) or much faster msgpack format(0)
stepsInHr:,2,the number of steps within 1 hourly time step for the GUTS simulation
useTemperatureData:,0,define water temperature from data (1) or forcing functions (0)
```

*LGUTS*

Concept

LGUTS simulates per spatial unit (a reach in the catchment) a toxico-kinetic toxico-dynamic (TKTD) model, over a single year of exposure to a pesticide in the water. Concentrations are provided on an hourly base. If needed, the simulation may apply a smaller sub-hourly timestep. The applied TKTD model is the reduced GUTS model in either the stochastic death (SD) or the individual tolerance (IT) version. No life-history of a particular species in taken into account. The set of GUTS coefficients is species- and chemical-specific. Outcome of the simulation is the survival over time, in the particular the survival up to the end of the simulation period. A deterministic approach is followed, so only a single run is required.

Control

Control parameters for LGUTS are defined in the *_*control.csv* input file (Table 3). LGUTS is run for a specific year (`applicationYear`). Concentration input is provided in csv or msgpack (MessagePack) format, defined by the value of `useCSV`. Survival is reported per day (`verbose` = 1) or only at the end of the year (`verbose` = 0). Hourly concentration values are expected, but the GUTS model can be simulated with a smaller sub-hourly timestep. Parameter `stepsInHr` sets the number of sub-hourly steps.

Output

Current output file contains individual survival over time (x1s1r1_guts_survival_reaches.txt), once per day (at the end of the day) for all the reaches in the catchment. Elements are tab-separated. When the option `verbose` is set to 0 (table 3), survival is only recorded at the end of the simulated year. The LGUTS model is deterministic and requires only a single simulation run. No Monte Carlo approach is needed, though of course the model can be part of such an approach running multiple stochastic efate simulations.

Margin of Safety analysis

To determine LP50 values, the concentration multiplication factor leading to a 50% reduction of final survival in the GUTS model, simulations for all reaches are run applying a series of multiplication factors to the hourly concentration time series.

Ideally, the full range from 0 to 100% effect (reduction of survival) should be covered, to ensure a reliable LP50 estimation by fitting a dose-response relationship. As a kind of brute-force approach, we set default multiplication factors according to a power function, e.g., ranging from 2^-10 to 2^15 (1/512 to 16384). The range can easily be redefined by the user in the script file.

To run a margin of safety analysis, the content of the `st` file with Smalltalk code that will be processed as a first step after starting up the image needs to be modified. Thus when starting (via LGUTSSD_MoS.bat)

```
Squeak.exe LEffectModel.image LGUTSSD_MoS.st
```

the content of the default `LGUTSSD_MoS.st` will be executed (Table 5).

*Table 5. Contents of the script file* `LGUTSSD_MoS.st`*).*

```
| mfs scriptFile |
ModelIO invalidateRootDirectories.
CatchmentConcDataBase removeAllDataBases.
RInterface rDirectory: nil. "'C:\Progra~1\R\R-3.5.3\bin\x64'."
mfs := (-10 to: 15) collect: [:p | (2 raisedTo: p) asFloat].
scriptFile := LGUTSProject scriptMoSAnalysisGUTSSDMultiplicationFactors: mfs.
(ModelProject fromScriptFile: scriptFile) runModelProjectForeground.
Smalltalk quitPrimitive
```

This Smalltalk script defines a modelscript, writes it in the \ecotalk output directory, and starts executing it. The `st` file can be modified by the user and optionally saved under another name – in which case the batch file needs to be modified as well. In particular the set of multiplication factors to use (line 5 defining temporary variable `mfs`) may be adjusted. E.g., the following statement leads to exactly the same series of multiplication factors:

```
mfs := #( 0.03125 0.0625 0.125 0.25 0.5 1.0 2.0 4.0 8.0 16.0 32.0 64.0 128.0 256.0 512.0
1024.0 2048.0 4096.0 8192.0 16384.0 32768.0).
```

Note that when editing, care should be taken that no smart (curly) quotes creep in (this is likely to happen when using Word as a text editor). NB in Smalltalk single quotes define a String; double quotes define a comment; names starting with a capital refer to object classes.

With a margin of safety analysis, results end up in a sub-directory defined by the model name with *_mos* appended to it. Each simulation with a specific multiplication factor is considered to be a separate scenario. Thus the complete analysis involves 1 experiment with multiple scenarios, and the output of the analysis ends up in the experiment directory \x1\. At the end of a margin of safety analysis, when all scenarios have finished, the results per scenario are collected in a single file (guts_survival_reaches.txt_mfactors.txt) in the experiment directory. Rows represent the reaches in the order defined by the geo-data (features in the shape file). Columns represent the different multiplication factors. Further analysis is coordinated by the Landscape Model's XRisk component.

Stand-alone

For a stand-alone application of LGUTS, LP50 values can be obtained by running the R script created by LGUTS. When the path to *Rscript.exe* of a working R environment is defined in the `st` script file (Table 5), the R script for the margin of safety analysis will directly run in a DOS box. The package *drc* should be installed in the R environment. The R script generates a file in csv format (*guts_marginOfSafety_LP50.csv*) with the final results: for each reach the estimated LP50 value after fitting dose-response curves as shown in figure 2. The columns in the file guts_marginOfSafety_LP50.csv refer to the estimate, the standard error, lower and upper value of the LP50. The spatial distribution of LP50 values is also reported in an ESRI shapefile similar to the one used for catchment spatial definition. This *guts_marginOfSafety_LP50.shp* (*.shx*, *.prj*, *.dbf*) contains the LP50 results in the dbf. It can directly be loaded in any GIS to produce a map or used for further spatial analyses. The shapefile is of the same format as the input shapefile from which the landscape was constructed (now a polylineZ shape).

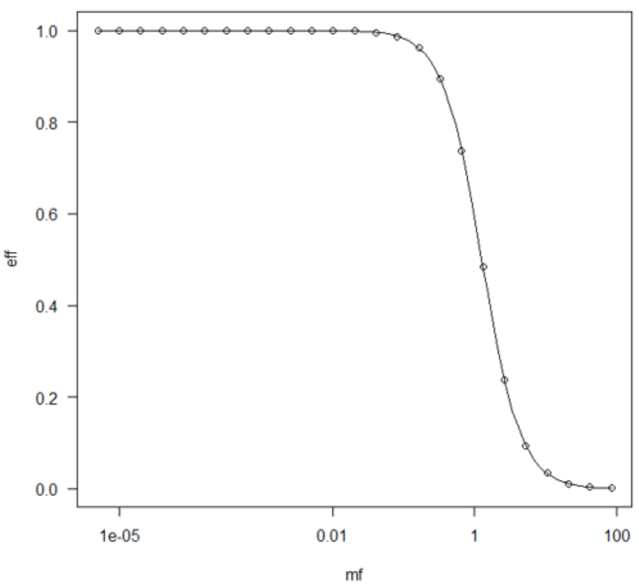


*Figure 2. With R package drc the multiplication factor (mf) required for 50% effect (reduction in survival) can be estimated. Note that all calculations are strictly deterministic in this version of the model, resulting in a perfect fit.*

*LPop*

Concept

LPop simulates the population dynamics of Asellus aquaticus on a network of connected waterbodies. In LPop individuals have a dynamic energy budget (DEB) and a reduced GUTS sub-model. The *abj* version of DEB (with metabolic acceleration) is used, parameterized from the Add-my-pet database. Individual growth and development (DEB) are simulated on a daily base; GUTS dynamics on an hourly base. The model accounts for fluctuating temperatures, from a catchment-wide forcing function, modifying the DEB rate coefficients. LPop is a stochastic model with probabilistic processes and individual variation in some of the DEB coefficients. Optionally, constant background mortality is assumed and/or the aging mortality from DEB model is used. Population regulation occurs through the mechanism of density-dependent mortality.

Control

Control parameters for LPop are defined in the **_control.csv* input file (Table 4). LPop is run for a series of years, from `startYear` to `endYear`. A subset of these years may have applications of pesticides, from `startApplicationYear` to `endApplicationYear`. As for LGUTS, hourly concentration values are expected, but the GUTS model can be simulated with a smaller sub-hourly timestep. Parameter `stepsInHr` sets the number of sub-hourly steps. LPop may use daily water temperature timeseries read from file. If the parameter `useTemperatureData` is set to 1, the file *water_temperature_101096_1979-2020.csv* will be used. This file is expected in the concentration data input directory.

Output

For a sinlge scenario (multiplication factor) an example of the output is given in figure 3. Number of individuals in the 3 stage classes (embryos, juveniles, adults) are given per day, per reach or over the whole catchment population (metapopulation). The metapopulation results are also combined in a single file when multiple runs are performed (the column_3 file is the one with the relevant information).

Another endpoint on the metapopulation level is observed: the number of extant local populations. This endpoint becomes more relevant in a margin of safety approach, where it constitutes an easy to collect and to interpret measure.

```
LPopSDModelSystem
|   ecotalk.ok
|   LPopSDModelSystem.modelscript
|
\---x1
    \---x1s1
            extantLocalPopsMetapop.txt_column_1.txt
            extantLocalPopsMetapop.txt_column_2.txt
            extantLocalPopsMetapop.txt_column_3.txt
            juvAndAdultMetapop.txt_column_1.txt
            juvAndAdultMetapop.txt_column_2.txt
            juvAndAdultMetapop.txt_column_3.txt
            LPopSDModelSystem_coefs.csv
            LPopSDModelSystem_control.csv
            x1s1r1_adultMetapop.txt
            x1s1r1_adultPopByReach.txt
            x1s1r1_embryoMetapop.txt
            x1s1r1_embryoPopByReach.txt
```

```
x1s1r1_extantLocalPopsMetapop.txt
x1s1r1_juvAndAdultMetapop.txt
x1s1r1_juvAndAdultPopByReach.txt
x1s1r1_juvenileMetapop.txt
x1s1r1_juvenilePopByReach.txt
```

*Figure 3. Example of output for a modelscript named LPopSDModelSystem.*

Visualization of dynamics is easily done for the metapopulation, in plots of population size against time. Local dynamics on the reaches is more easily visualized on maps, either static or dynamic.

Margin of safety

A margin of safety approach can be followed, running a set of scenarios, where each scenario applies a different multiplication factor. The number of extant local populations is a straightforward indicator of impact on the catchment level. More sensitive measures based on population density (e.g. distance to the normal operating range, cq difference with the behaviour of a reference population) will have to be defined.