qpNCA User Guide

Jan Huisman

v1.1.1, June, 2021

Table of Contents

List of Tables

1 Introduction

qpNCA is an R package for pharmacokinetic (PK) parameter calculations developed within qPharmetra. It consists of a set of functions for time deviation corrections, missing concentration imputations and parameter calculations. Parameter calculations are based on the NCA Parameter Guideline (*reference*). The output of the package comprises R datasets containing the final PK parameters, regression plots and information regarding the time corrections and concentration imputations applied.

2 qpNCA functions

2.1 Main functions

The main qpNCA functions perform the actual time deviation corrections, missing concentration imputations and parameter calculations. As most functions need the output from previous functions as input, the order of function calls is mandatory. Main qpNCA functions should be called in the following order:

1.	**correct.loq**	(LOQ handling)
2.	**est.thalf**	(estimate terminal half-life)
3.	**plot.reg**	(plot PK curve with regression line)
4.	**calc.ctmax**	(calculate C_{max} and t_{max} on uncorrected data)
5.	**correct.time**	(time deviation corrections)
6.	**correct.conc**	(missing concentration imputations)
7.	**tab.corr**	(tabulates applied time corrections and concentration imputations)
8.	**calc.par**	(PK parameter calculations that do not need half-life)
9.	**calc.par.th**	(PK parameter calculations that DO need half-life)

2.2 Internally used functions

Apart from the main qpNCA functions the package contains a number of functions that are called by the main functions internally and do not need intervention by the user:

-	**check_input**	(check input dataset and function arguments)
-	**interpol**	(interpolation rules)
-	**trap**	(trapezoidal rules for AUC calculations)
-	**trapm**	(trapezoidal rules for AUMC calculations)
-	**lag.lead**	(defines lag and lead values, needed by the main functions)
-	**filenamefun**	(derives plot filename from by variables)
-	**titlefun**	(derives plot title from by variables)

Detailed descriptions of the functions can be found in Appendix A.

2.3 Wrapper function qpNCA

To facilitate the execution of the individual functions a wrapper function (qpNCA) is available. The function calls the individual functions in the correct order.

3 Dataset requirements

For correct execution of the NCA analysis the input data should meet the following requirements:

Input dataset

The input dataset must contain:

- variables identifying a single concentration-time profile over one dosing interval (by-variables)
- a variable containing the nominal time after the last dose
- a variable containing the actual time after the last dose
- a variable containing the dependent variable
- a variable indicating whether an observation is below the LOQ (value=1) or not (value=0)
- a variable containing the LLOQ value

Covariates dataset

The covariates dataset should have only one record per by-group and contains variables which should be kept within the final parameter dataset (e.g. demographics, treatment codes). This dataset must contain a variable with the dose amount given for that particular by-group.

4 Execution

The wrapper function is called in the following way:

```
qpNCA (x, by=c("subject"), nomtimevar="ntad", timevar="time", depvar="dv",
        bloqvar="bloq", loqvar="loq", loqrule=1,
        includeCmax="Y", exclvar=NA, plotdir=NA, timelab="timevar", deplab="depvar",
        tau=NA, tstart=NA, tend=NA, teval=NA,
        covariates=NA, dose=”dose”, factor=1, reg="SD", ss="N", route="EV", method=1)
```

Arguments of the wrapper function are described in Table 1.

Table 1: Arguments of the qpNCA function

Argument	Description	Default value	Manda-tory?
x	Input dataset name		Y
by	by-variable(s), e.g. c("subject","day"). Each by-group should contain concentration-time information for one dosing interval	NA	Y
nomtimevar	variable name containing the nominal sampling time	“ntad”	Y
timevar	variable name containing the actual sampling time	“tad”	Y
depvar	variable name containing the dependent variable (e.g., concentration)	“dv”	Y
bloqvar	variable name containing the BLOQ flag (0: no, 1: yes)	“bloq”	Y
loqvar	variable name containing the LOQ value (e.g., 0.1)	“loq”	Y

Argument	Description	Default value	Manda-tory?
loqrule	rule number to be applied to the LOQ values in the curve: before first measurable concentration — after first measurable concentration: first consecutive LOQ — other consecutive LOQ ------------------ ---------------- ---------------- 1: 0 NA NA 2: 0 0 0 3: 0 0.5*LOQ NA 4: 0 0.5*LOQ 0	1	Y
includeCmax	Include Cmax in half-life estimation? (y/n)	“Y”	Y
exclvar	variable name indicating points to be excluded in half-life estimation (these should have <exclvar>=1)	NA	N
plotdir	folder where regression plots (.PNG) will be saved; ; NA gives default location, NULL skips regression plots	NA	N
timelab	X-axis label in regression plots	“timevar”	N
deplab	Y-axis label in regression plots	“depvar”	N
tau	dosing interval (for multiple dosing), NA (default) if single dose	NA	N
tstart	start time of partial AUC (start>0), NA (default) if not requested	NA	N
tend	end time of partial AUC, NA (default) if not requested	NA	N
teval	user selected AUC interval (starting at t=0), NA (default) if not requested	NA	N
covariates	co-variates dataset, containing variables to be merged with each curve (e.g. demographics). Must contain the dose variable	NA	Y
dose	variable containing the dose amount	“dose”	Y
factor	conversion factor for CL and V calculation (e.g. dose in mg, conc in ng/mL, factor=1000 to get CL in L/h and V in L)	1	N
reg	regimen, "SD" or "MD"	“SD”	Y
ss	is steady state reached (Y/N)	“N”	Y
route	route of drug administration ("EV", "IVB", ”IVI”)*	“EV”	Y
method	method for trapezoidal rule: 1: linear up - linear down 2: linear up - logarithmic down 3: linear before Tmax, logarithmic after Tmax	1	Y

* EV: Extravascular IVB: Intravenous bolus IVI: Intravenous infusion

5 Output

The qpNCA package produces two types of output: result datasets and regression plots.

5.1 Result datasets

The output generated during the execution are gathered in an R list object. The following output datasets will be generated:

half_life: contains all relevant information and results from the regression analysis

covariates: contains relevant covariates for each curve

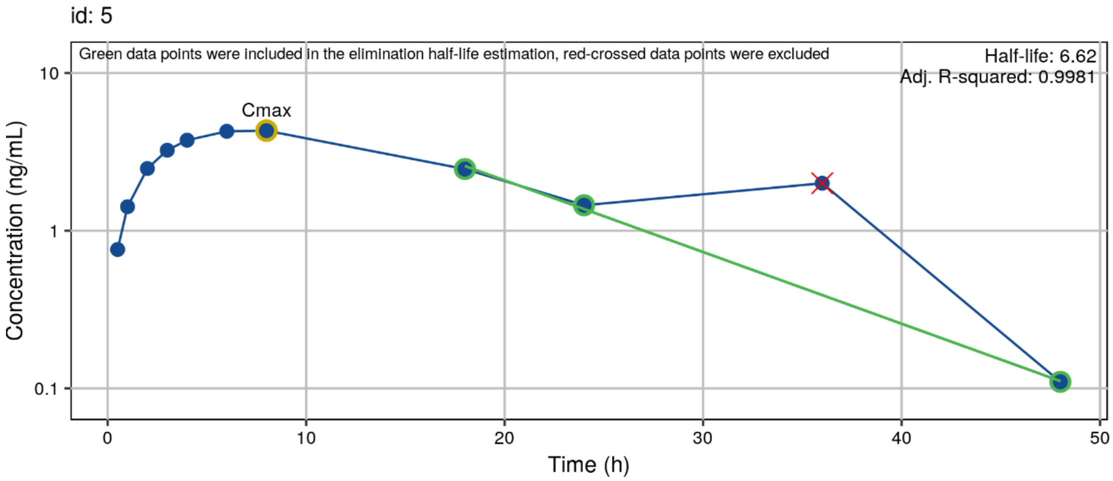
corrections: contains information about what deviation corrections have been applied to each individual concentration-time curve

ct_corr: contains corrected concentration-time curves

pkpar: contains individual pharmacokinetic parameters

5.2 Regression plots

Regression plots will be made after the half-life estimation step. The user can choose to plot to standard output or to save the plots as *.png files to a specific folder (see *plotdir* argument). The file name is automatically generated and reflects the by-variables for that particular curve (e.g. “subject_1_day_1.png”. An example of a regression plot (filename: id_5.png) is given below.



In this plot the following items are indicated:

- Cmax (Yellow, even if no half-life was estimated)

- points used in regression and resulting regression line (green)

- points excluded from regression (red crossed)

- estimate of elimination half-life and adjusted R-squared

6 Workflow

This paragraph describes the qpNCA main functions and their use in more detail. In the examples below, it is assumed that individual curves in the input file *df* are identified by the *subject* and *period* variables. Furthermore, the following values are used for the arguments:

Table 2: Argument values used in the examples

Argument	Description	Value
by	variable name containing the by-variables	c("subject","period")
nomtimevar	variable name containing the nominal time after last dose	"ntad"
timevar	variable name containing the actual time after last dose	"tad"
depvar	variable name containing the dependent variable	"dv"
bloqvar	variable name containing the BLOQ flag (0: no, 1: yes)	"bloq"
loqvar	variable name containing the LOQ value	"loq"
exclvar	variable name containing information about points to be excluded in the thalf estimation (these should have <*exclvar*>=1)	"excl"
plotdir	Folder where regression plots will be stored	"plots"

6.1 BLOQ corrections (correct.loq.r)

Imputes LOQ values according to the chosen LOQ substitution rule. Table 3 shows the available substitution rules and how LOQ values will be imputed.

Table 3: LOQ substitution rules

Rule nr.	Before first measurable concentration	After first measurable concentration	
		First consecutive BLOQ	Other consecutive BLOQs
1	0	NA	NA
2	0	0	0
3	0	0.5*LOQ	NA
4	0	0.5*LOQ	0

Function call:

```
loqed = <inputfile> %>%
        correct.loq(by= c("subject","period"), nomtimevar="ntad", timevar="tad", depvar="dv",
        bloqvar="bloq", loqvar="loq", loqrule=1)
```

Arguments of the correct.loq function are described in Table 4.

Table 4: Arguments for correct.loq function

Argument	Description	Default value	Mandatory?
by	variable name containing the by-variables	NA	Y
nomtimevar	variable name containing the nominal sampling time	"ntad"	Y
timevar	variable name containing the sampling time	"tad"	Y
depvar	variable name containing the dependent variable (e.g., concentration)	"dv"	Y
bloqvar	variable name containing the BLOQ flag (0: no, 1: yes)	"bloq"	Y
loqvar	variable name containing the LOQ value	"loq"	Y
loqrule	rule number to be applied to the LOQ values in the curve	1	Y

The result of the function is a dataset with imputed BLOQ concentrations using the chosen imputation rule.

6.2 Elimination half-life estimation (est.thalf.r)

Estimates terminal half-life and related parameters based on the procedure described in the NCA Parameter Guideline.

The function starts with the last three sample points and performs log-linear regression on it. It then adds one sampling point at a time (including and ending at tmax) and performs the regression again. The results of the regression with the highest adjusted R-squared are returned. Visual outliers can be excluded from the regression analysis.

Function call:

```
th = loqed %>%
        est.thalf(by=c(“subject”,”period”), timevar="tad", depvar=“dv", includeCmax="Y",
        exclvar=”excl”)
```

Arguments of the est.thalf function are described in Table 5. The result of the function is a dataset with regression parameters for each curve. The parameters are listed in Table 6.

Table 5: Arguments for est.thalf function

Argument	Description	Default value	Manda-tory?
by	variable name containing the by-variables	NA	Y
timevar	actual time after last dose	“tad”	Y
depvar	dependent variable	“dv”	Y
includeCmax	can Cmax be part of subset used for regression? (Y/N)	“Y”	Y
exclvar	variable name containing information about points to be excluded (these should have <*exclvar*>=1)	NA	N

Table 6: Parameters returned from est.thalf function

Parameter	Description
no.points	number of data points used in the regression analysis
intercept	estimated intercept
lambda_z	-1*estimated slope
r.squared	square of the correlation coefficient
adj.r.squared	adjusted square of the correlation coefficient
thalf	elimination half-life
start_th	time of first sample included in the thalf estimation
end_th	time of last sample included in the thalf estimation
includeCmax	include results of regression including Cmax in selection? (y/n)
points_excluded	are time points excluded from the half-life estimation? (y/n)

The output dataset will be used in both time deviation corrections and missing concentration imputations.

6.3 Regression plots

The user can choose to create plots (*.PNG) which depicts details regarding the estimation of the elimination rate constant.

Function call:

```
Loqed %>% plot.reg(by=c("subject",”period”), th=th, bloqvar="bloq", timevar="tad", depvar="dv",
            exclvar="excl", plotdir="plots", timelab="Time (h)", deplab="Concentration (ng/mL)")
```

Arguments of the plot.reg function are described in Table 7: Arguments for the plot.reg function. The result of the function is a graphical file (*.PNG) for each curve. See paragraph 5.2 for more details about the information in the graph.

Table 7: Arguments for the plot.reg function

Argument	**Description**	**Default value**	**Manda-tory?**
by	list of variables that identify a single PK curve	NA	Y
th	file name of file with lamdba_z information for each curve	NA	Y
bloqvar	variable name containing the BLOQ flag (0: no, 1: yes)	”bloq”	Y
timevar	variable name containing the sampling time	“tad”	Y
depvar	variable name containing the dependent variable (e.g., concentration)	“dv”	Y
exclvar	variable name containing information about points to be excluded (these should have <*exclvar*>=1)	NA	Y
plotdir	folder where plots will be saved; if “NA” plots will be sent to standard output, if “NULL” no plots will be generated. If the folder does not yet exist it will be created	NA	Y
timelab	label for the time axis of the plot	“timevar”	Y
deplab	label for the Y-axis (dependent variable) of the plot	“depvar”	Y

6.4 C_{max} and t_{max} calculation (calc.ctmax.r)

Calculates C_{max} and t_{max} from uncorrected data. When maximum concentrations occur at various time points in the same individual concentration vs. time profile, the first occurrence will be considered for C_{max} and t_{max}. If all concentrations are NA, both C_{max} and t_{max} are set to NA.

Function call:

```
ctmax = loqed %>%
        calc.ctmax(by=c("subject",”period”), timevar="tad", depvar="dv"))
```

Arguments of the calc.ctmax function are described in Table 8. The result of the function is a dataset containing the parameters described in Table 9.

Table 8: Arguments for the calc.ctmax function

Argument	**Description**	**Default value**	**Manda-tory?**
by	list of variables that identify a single PK curve	NA	Y
timevar	actual time after last dose	“tad”	Y

Argument	Description	Default value	Manda-tory?
depvar	dependent variable	“dv”	Y

Table 9: Parameters returned from calc.ctmax function

Parameter	Description
C_{max}	maximum concentration
t_{max}	time of first occurence of C_{max}

6.5 Time deviation corrections (correct.time.r)

Time deviation corrections are needed if actual time after dose deviates from nominal time after dose at critical time points. Critical time points can be t=0, t=tau, t=teval (where teval is the last time point of the partial area $AUC_{0\text{-}teval}$), tstart and tend (if $AUC_{tstart\text{-}tend}$ is requested).

qpNCA corrects time deviations by applying specific rules to the critical time points, as stated in Table 10.

Table 10: Time deviation correction rules

Rule nr.	Regimen	Description	Applied to
SDT-1	sd	Set actual time to 0	t=0
SDT-2	sd	Correct concentration at deviating time **by interpolation**	t=tau, tstart, tend, teval
SDT-3	sd	Correct concentration at deviating time **by extrapolation**	t=tau, tend, teval
MDT-1	md	if predose sample taken after dosing, set actual time to 0 and conc to NA	t=0
MDT-2	md	Correct concentration at deviating time **by interpolation**	t=tau, tstart, tend, teval
MDT-3	md	Correct concentration at deviating time **by extrapolation**	t=0, tau, tend, teval
MDT-3a	md	Set actual time to 0 if concentration is BLOQ	t=0

Function call:

```
df_tc = loqed %>%
        correct.time(by=c(”subject”,”period”), nomtimevar="ntad", timevar="tad", depvar=“dv",
                     tau=, tstart=, tend=, teval=, th=, reg=, method=)
```

Arguments of the correct.time function are described in Table 11.

Table 11: Arguments of the correct.time function

Argument	Description	Default value	Manda-tory?
by	list of variables that identify a single PK curve	NA	Y
nomtimevar	variable containing the nominal sampling time	“ntad”	Y
timevar	actual time after last dose	“tad”	Y
depvar	dependent variable	“dv”	Y

Argument	Description	Default value	Manda-tory?
tau	dosing interval (for multiple dosing)	NA	N
teval	user selected AUC interval, if not requested, leave empty	NA	N
tstart	start time of partial AUC (start>0)	NA	N
tend	end time of partial AUC	NA	N
th	file name of file with lamdba_z information for each curve	NA	N
reg	regimen, "SD" or "MD“	“SD”	Y
method	method of interpolation: 1: linear up - linear down 2: linear up - logarithmic down 3: linear before first Tmax, logarithmic after first Tmax	1	Y

The result of the function is a dataset with:

- Time deviation corrections applied to each requested critical time point
- Correction rule number and explanatory text

This dataset will be the input dataset for the missing concentration imputations.

6.6 Missing concentration imputations (correct.conc.r)

Missing concentrations at critical time points need to be imputed before corresponding AUCs can be calculated. This function will take the output dataset from correct.time.r as input dataset. qpNCA imputes missing concentrations by applying specific rules to the critical time points as described in Table 12.

Table 12: Missing concentration imputation rules

Rule nr.	Regimen	Description	Applied to
SDC-1	sd	Set concentration to 0 (only non-endogenous compounds)	t=0
SDC-2	sd	impute missing concentration **by interpolation**	t=tau, tstart, tend, teval
SDC-3	sd	impute missing concentration **by extrapolation**	t=tau, tend, teval
SDC-4	sd (IVB)	impute missing concentration by **back-extrapolation**	t=0
MDC-1	md	impute missing concentration by existing conc at t=0 or t=tau*	t=0, tau
MDC-2	md	impute missing concentration **by interpolation**	t=tau, tstart, tend, teval
MDC-3	md	impute missing concentration **by extrapolation**	t=tau, tend, teval
MDC-4	md (IVB)	impute missing concentration by **back-extrapolation**	t=0

* *only if steady state has been reached*

Function call:

```
df_tc_cc = df_tc %>%
            correct.conc(by=c(“subject”,”period”), nomtimevar="ntad", tau=,
                           tstart=, tend=, teval=, th=, reg=,ss=, method=, route=)
```

Arguments of the correct.conc function are described in Table 13.

Table 13: Arguments of the correct.conc function

Argument	**Description**	**Default value**	**Manda-tory?**
x	input dataset name (likely the output of correct.time	NA	Y
by	list of variables that identify a single PK curve	NA	Y
nomtimevar	variable containing the nominal sampling time	“ntad”	Y
tau	dosing interval (for multiple dosing)	NA	N
teval	user selected AUC interval (start: t=0)	NA	N
tstart	start time of partial AUC (start: t>0)	NA	N
tend	end time of partial AUC	NA	N
th	file name of file with lamdba_z information for each curve	NA	N
reg	regimen, "SD" or "MD“	“SD”	Y
ss	is steady state reached? (y/n)	“N”	Y
method	method of interpolation: 1: linear up - linear down 2: linear up - logarithmic down 3: linear before first Tmax, logarithmic after first Tmax	1	Y
route	route of drug administration ("EV", "IVB", ”IVI”)	“EV”	Y

The result of the function is a dataset with:

- Concentration imputations applied to each requested critical time point
- Correction rule number and explanatory text

This dataset will be the input dataset for both the overview of time corrections and imputations (tab.corr.r) and the parameter calculation (calc.par.r).

6.7 Overview of time corrections and imputations (tab.corr.r)

This function summarizes the time deviation corrections and concentration imputations applied to the entire dataset. If records are added because a critical time point record is missing, this will also be documented. It takes the resulting dataset after time deviation correction and concentration imputation as input dataset.

Function call:

```
Corrtab = df_tc_cc %>%
           tab.corr(nomtimevar="ntad",by=c(“subject","period"))
```

Arguments of the tab,corr function are described in Table 14.

Table 14: Arguments of the tab.corr function

Argument	Description	Default value	Manda-tory?
nomtimevar	variable containing the nominal time	“ntad”	Y
by	list of variables that identify a single PK curve	NA	Y

The result of the function is a dataset with:

- A record for each deviating time point containing:

  - Time deviation correction rule applied
  - Concentration imputation rule applied
  - Textual explanation
  - Info about the critical time point(s) involved
  - If records are added because a critical time point record is missing, this will also be documented

6.8 Calculation of PK parameters, no half-life needed (calc.par.r)

This function produces all requested parameters for which half-life is not needed in the calculation. It takes the resulting dataset after time deviation correction and concentration imputation as input dataset.

Function call:

```
par= df_tc_cc %>% calc.par(by=c(“subject","period"), tau=, tstart=, tend=, teval=,
                            route=, method=)
```

Arguments of the calc.par function are described in Table 15. The result of the function is a dataset containing the parameters described in Table 16.

Table 15: Arguments of the calc.par function

Argument	Description	Default value	Manda-tory?
by	list of variables that identify a single PK curve	NA	Y
tau	dosing interval (for multiple dosing)	NA	N
teval	user selected AUC interval	NA	N
tstart	start time of partial AUC (start>0)	NA	N
tend	end time of partial AUC	NA	N
route	route of drug administration ("EV", "IVB", ”IVI”)	“EV”	Y
method	method for trapezoidal rule: 1: linear up - linear down 2: linear up - logarithmic down 3: linear before first t_{max}, logarithmic after first t_{max}	1	Y

Table 16: Parameters returned from the calc.par function

Parameter	Description
t0.ok	flags if t=0 concentration could be corrected/imputes. If not, no AUCs starting at t=0 are calculated

Parameter	Description
tlast.ok	flags if there is at least one measurable concentration. If not, no AUClast can be calculated
tlast	time of last sample with measurable concentration
clast.obs	observed concentration at tlast
aucall	auc calculated over all observations, including values below LOQ (which are set to 0)
auclast	auc calculated using all observations up to and including the last measurable concentration (clast.obs at tlast)
aumcall	aumc calculated over all observations, including values below LOQ (which are set to 0)
aumclast	aumc calculated using all observations up to and including the last measurable concentration (clast.obs at tlast)
mrtall	mean residence time based on aucall
mrtlast	mean residence time based on auclast
tau	the dosing interval (if specified)
calc.tau	flags if AUCtau could be calculated
auctau	auc calculated over the dosing interval, only calculated if tau is specified
aumctau	aumc calculated over the dosing interval, only calculated if tau is specified
teval	user selected AUC interval starting at t=0 (if specified)
calc.teval	flags if AUCteval could be calculated
aucxx	auc calculated from t=0 up to/including teval, only calculated if teval is specified (xx is substituted by teval)
calc.part	flags if AUCpart could be calculated
tstart	start time of partial AUC (if specified)
tend	end time of partial AUC (if specified)
aucx_y	partial auc from time=x up to/including time=y, where x>0, only calculated if tstart and tend are specified
c0	back-extrapolated concentration at t=0 for IV bolus administration
area.back.extr	area back-extrapolated to 0

This dataset will be the input dataset for the functions that calculates PK parameters that do need half-life for calculation.

6.9 Calculation of Parameters that do need thalf (calc.par.th.r)

This function calculates the PK parameters for which a half-life is needed for calculation. It takes the output dataset from the calc.par.r function as input dataset. Besides a dataset containing the half-life information it needs a covariate dataset that at least contains the by-variables and a variable that contains the dose amount associated with the curve.

Function call:

```
par_th = par %>%
        calc.par.th(by=c(“subject”,”period”), th=th,
        covariates=, dose=, factor=, reg=, ss=,route=)
```

Arguments of the calc.par function are described in Table 17. The result of the function is a dataset containing the parameters calculated by the *est.thalf* and *calc.par* functions and the extra parameters described in Table 18.

Table 17: Arguments of the calc.par.th function

Argument	Description	Default value	Manda-tory?
by	by-variable(s), e.g. c("subject","day"). Each by-group should contain concentration-time information for one dosing interval	NA	Y
th	result dataset from est.thalf	th	Y
covariates	co-variates dataset	NA	Y
dose	variable containing the dose amount	“dose”	Y
factor	conversion factor for CL and V calculation (e.g. dose in mg, conc in ng/mL, factor=1000)	1	Y
reg	regimen, "sd" or "md"	“SD”	Y
ss	is steady state reached (y/n)	“N”	Y
route	route of drug administration ("EV", "IVB", ”IVI”)	“EV”	Y

Table 18: Parameters returned from the calc.par.th function

Parameter	Description
t0.ok	flags if t=0 concentration could be corrected/imputes. If not, no AUCs starting at t=0 are calculated
tlast.ok	flags if there is at least one measurable concentration. If not, no AUClast can be calculated
tlast	time of last sample with measurable concentration
clast.obs	observed concentration at tlast
aucall	auc calculated over all observations, including values below LOQ (which are set to 0)
auclast	auc calculated using all observations up to and including the last measurable concentration (clast.obs at tlast)
clast.pred	predicted concentration at tlast
aucinf.obs	aucinf based on observed concentration at tlast
aucinf.pred	aucinf based on predicted concentration at tlast
aumcinf.obs	area under the first moment curve extrapolated to infinity, based on observed concentration at tlast
aumcinf.pred	area under the first moment curve extrapolated to infinity, based on predicted concentration at tlast
cl.obs, cl.f.obs, cl.ss, cl.f.ss	clearance based on aucinf.obs, at steady state based on auctau
cl.pred, cl.f.pred	clearance based on aucinf.pred
mrt.obs	mean residence time based on aumcinf.obs and aucinf.obs
mrt.pred	mean residence time based on aumcinf.pred and aucinf.pred
vz.obs, vz.f.obs	distribution volume based on cl.obs/cl.f.obs, at steady state based on auctau
vz.pred, vz.f.pred	distribution based on cl.pred/cl.f.pred
vss.obs	steady-state volume based on cl.obs and mrt.obs
vss.pred	steady-state volume based on cl.pred and mrt.pred
pctextr.pred	percentage of AUC extrapolated to infinity, based on aucinf.pred
pctextr.obs	percentage of AUC extrapolated to infinity, based on aucinf.obs
pctback.pred	percentage of AUC extrapolated back to 0, based on aucinf.pred
pctback.obs	percentage of AUC extrapolated back to 0, based on aucinf.obs

7 References

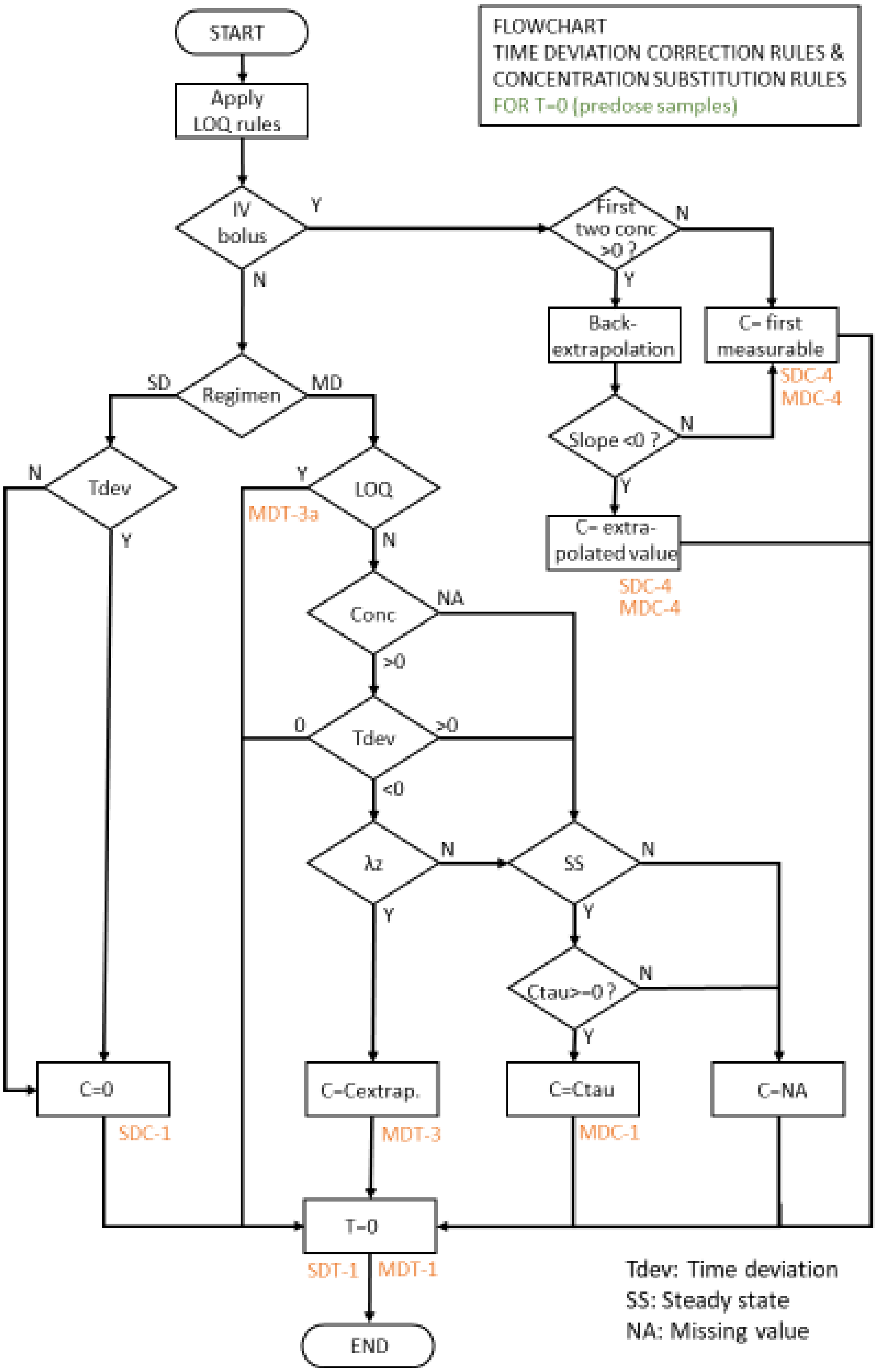
Rowland and Tozer (2011). *Clinical Pharmacokinetics and Pharmacodynamics: Concepts and applications*, 4th ed. Wolters Kluwer, Philadelphia.

Gabrielsson and Weiner (1997). *Pharmacokinetic and Pharmacodynamic Data Analysis: Concepts and Applications*, 2nd ed. Swedish Pharmaceutical Press, Stockholm.

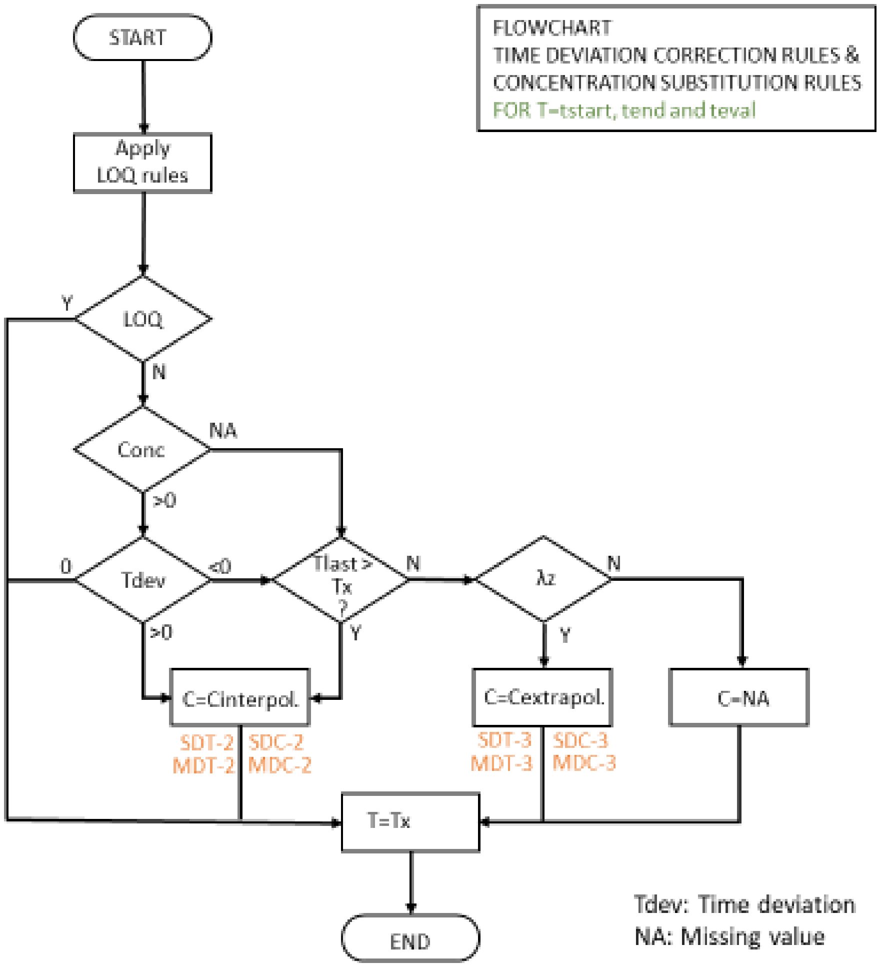
Gibaldi and Perrier (1982). *Pharmacokinetics*, 2nd ed. Marcel Dekker, New York.

8 Flowcharts

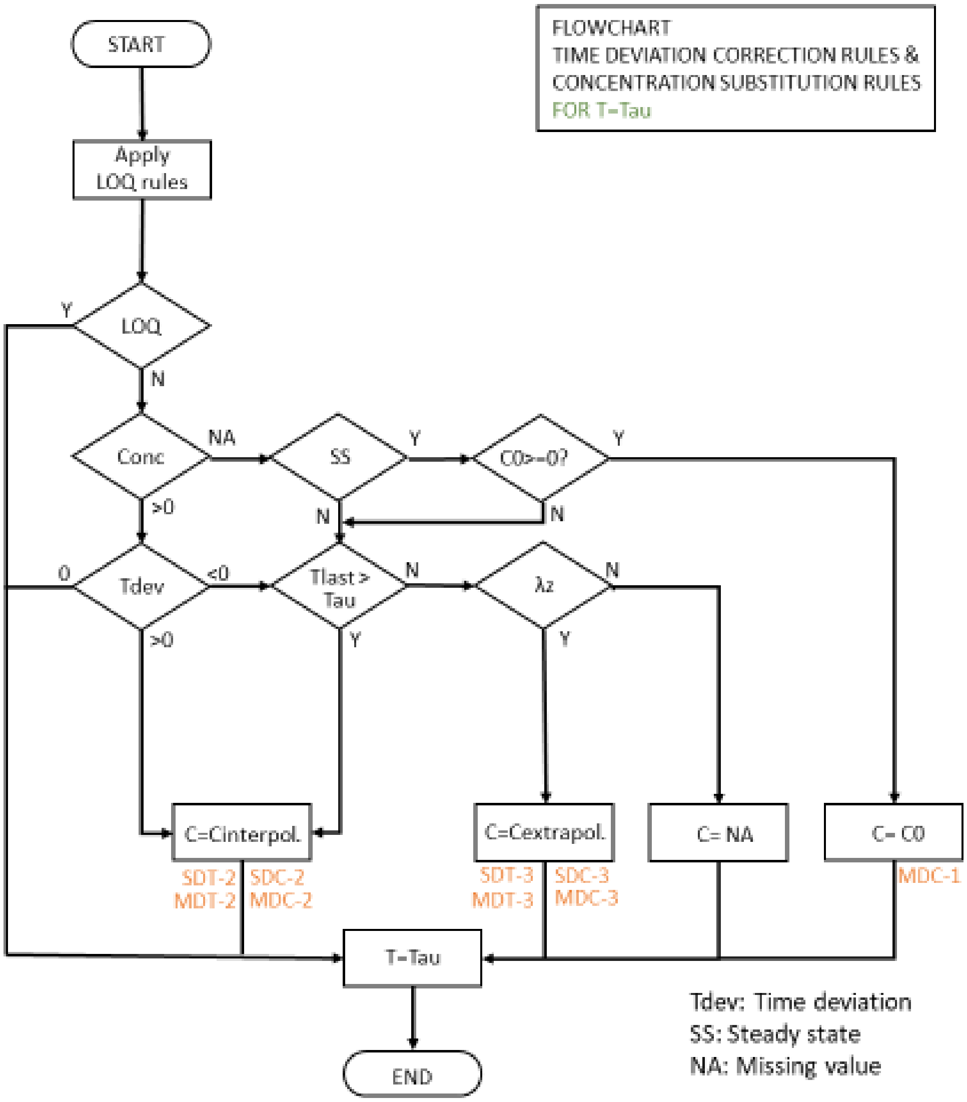
8.1 Flowchart of corrections and imputations for t = 0 (predose)

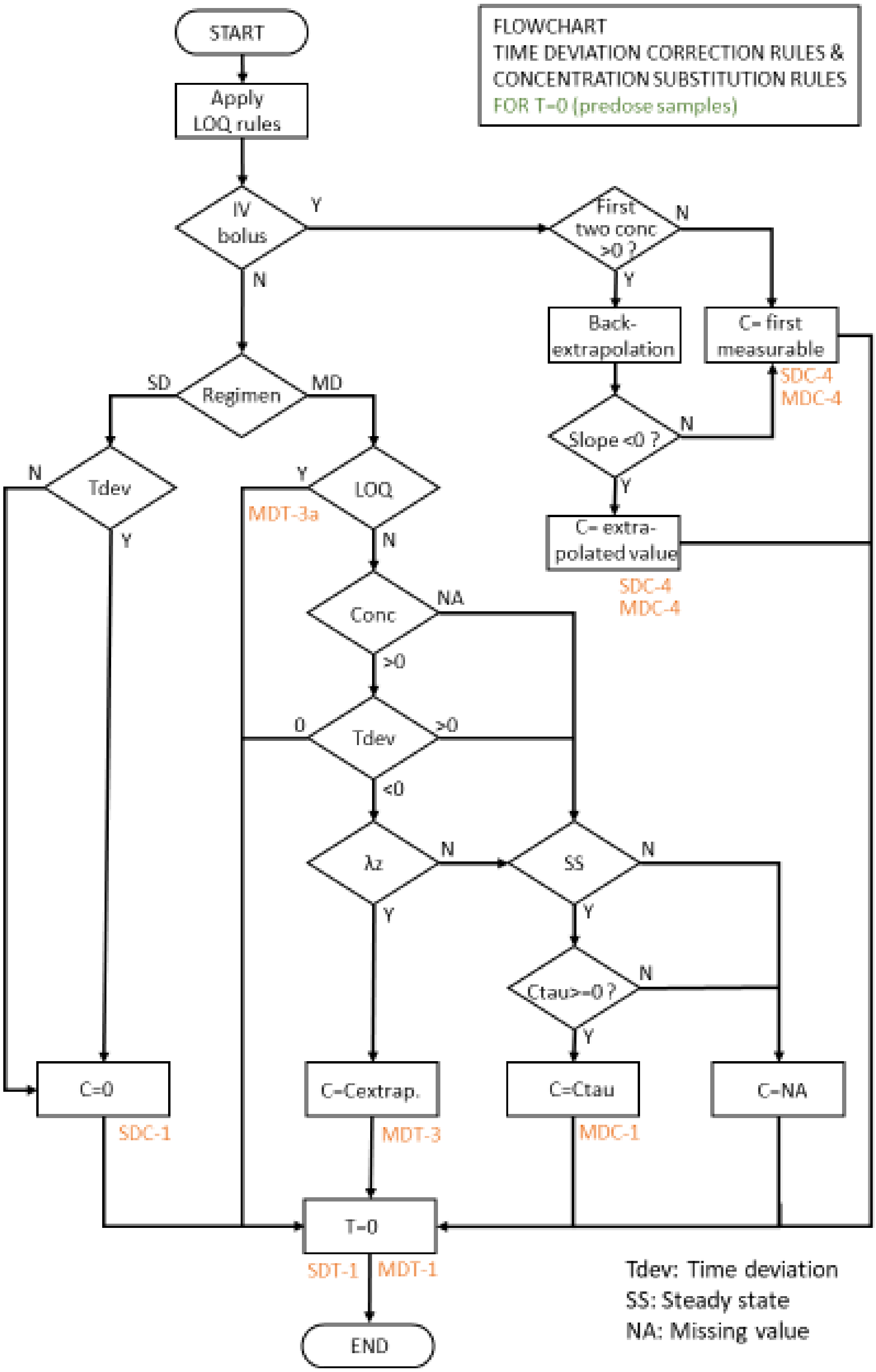


8.2 Flowchart of corrections and imputations for t = tstart, tend and teval



8.3 Flowchart of corrections and imputations for t = tau





FLOWCHART
TIME DEVIATION CORRECTION RULES &
CONCENTRATION SUBSTITUTION RULES
FOR T=0 (predose samples)

Tdev: Time deviation
SS: Steady state
NA: Missing value