optpar.txt

The file holds additional model settings to info.txt and is therefore located in the same folder as info.txt. The file is used to define what kind of optimisation to be done if `calibration` is set in info.txt. There are several different methods to chose from, each with their settings. Which model parameters to calibrate and within which boundaries are information also given in *optpar.txt*.

Maximum 100 model parameters may be optimised simultaneously. To optimise more parameters, the code needs to be changed (set *maxoptpar* to a higher value). All parameters are described in the section on par.txt, but not all of them can be calibrated. The objective function of the optimization is defined in info.txt as the combination of criteria chosen, see Performance criteria options.

There are eight methods of optimisation implemented in HYPE as detailed in the table below (read more about them in the tutorial). Additionally, there are two other tasks for output generation, `WA` and `WS`, which produce detailed performance and simulation results for all runs performed during optimisation. Tasks `WA` and `WS` are compatible with selected optimisation methods only, a denoted in the table. The task of organized scanning `SC` is a parameter investigation method.

The optpar.txt file may also be used for parameter ensemble simulation (`parensemble` in info.txt), to know which model parameters to use. The parameter ensemble may use the parameters earlier found by calibration.

Task	Description
`MC`	Monte Carlo (MC) simulation with parameter values randomly distributed over the intervals (basic MC-method)
`BP`	progressive Monte Carlo simulation with parameter space limited by best found so far (alternative MC-method)
`SM`	progressive Monte Carlo simulation with parameter space reduced in stages (alternative MC-method)
`DE`	Differential Evolution Markov Chain method (alternative MC-method)
`BN`	optimisation with Brent method
`Q1`	optimisation with QuasiNewton DFP gradient-based method
`Q2`	optimisation with QuasiNewton BFGS gradient-based method
`SD`	optimisation with QuasiNewton steepest descent method
`SC`	organised scanning of two parameters
`WA`	write performance result for all simulations (`MC`, `SM` or `DE`)
`WS`	write simulation results (basin-, time-, map-, , regional-, or , class-files) for all ensembles in Monte Carlo simulation (`MC`, `BP`, or `DE`)
`AS`	run parameter ensemble simulations with parameters found in allsim.txt
`BS`	run parameter ensemble simulations with parameters found in bestsims.txt

File content

The first row is for general comments. It is ignored by the program when reading the file. Next comes a section with calibration settings. It reaches from second to 21st row, and are used to define tasks and other settings. Last comes a section defining the parameters to be calibrated. These occupy row 22 and onward.

In the calibration setting section a row starts with a code indicating a task or other settings. Argument of the code is listed from position 12 and forward on each row. The following options are available for the calibration setting section:

Code	Argument	Default value	Description
task	*two letter word*	FALSE	define what kind of optimisation to do (see methods above), and if additional results are to be written for the MC-methods
cal_log	*Y/N*	YES	flag for writing a calibration.log file
scan_numx	*integer*	1	number of steps taken for the first parameter (SC method)
scan_numy	*integer*	1	number of steps taken for the second parameter (SC method)
num_mc	*integer*	1000	number of Monte Carlo simulations (per centre point and stage for progressive MC)
num_ens	*integer*	1	number of best Monte Carlo simulations to keep and print results from (and use as centre points for next stage of progressive MC) (maximum 999)
num_bpmc	*integer*	200	number of simulations per reduced parameter space which the best simulations shall be selected from (BP MC-method)
num_bpmax	*integer*	100	number of reductions of the parameter space for MC simulation (BP MC-method)
num_stages	*integer*	1	number of stages for progressive Monte Carlo (SM MC-method)
num_zoom	*real*	0.9	reduction of parameter space (0-1) for each stage of progressive MC (BP MC-method)
DEMC_ngen	*integer*	100	number of generations for DEMC method
DEMC_npop	*integer*	25	number of populations for DEMC method
DEMC_gammascale	*real*	1	scaling of the mutation strength for DEMC method. A new (next generation) parameter candidate is proposed as a mutation of the parent parameter value based on the difference between two random members of the parent population. DEMC_gammascale is a scaling factor for the resulting parameter jump width. Small values will cause smaller mutations, which potentially stabilises the search through large parameter spaces at the cost of convergence speed. A value of 1 will result in no scaling.
DEMC_crossover	*real*	1	crossover probability for DEMC method. Probability that the proposed candidate is chosen instead of the parent parameter. Large DEMC_crossover values mean larger probability that the proposal is chosen. Set to 1, all proposals are accepted. This makes it harder to find an acceptable overall proposal because all parameters are changed in every generation. Set to 0.5, each parameter candidate has only a 50% chance to be accepted into the next proposal.

Code	Argument	Default value	Description
DEMC_sigma	*real*	0.1	sample error standard deviation for DEMC method. Base for the standard deviation of the random perturbation, which adds random noise to the proposed parameter in addition to the gamma-mutation. This value is multiplied with 3rd-row value for each parameter (see description of parameter rows below).
DEMC_accprob	*integer*	0	scaling factor for probabilistic acceptance for DEMC method (0 = off (default); >0 = on). If set to off, parameter proposals will only (and always) be accepted if the objective function decreases (= better performance). If turned on, also proposals with higher value of the objective function can be accepted; better performance will give higher probability of acceptance. High values of the scaling factor will also increase the probability of acceptance.
BR_diagStp	*Y/N*	YES	flag for taking a diagonal step at the end of each iteration (BN method)
num_maxItr	*integer*	500	max amount of iterations (interrupt non-MonteCarlo methods)
num_maxTim	*integer*	72	max calibration time (hours) (interrupt non-MonteCarlo methods)
num_parItr	*integer*	10	number of iterations taken into account for parameter change tolerance (interrupt non-MonteCarlo methods)
num_criItr	*integer*	10	number of iterations taken into account for criterium change tolerance (interrupt non-MonteCarlo methods)
num_criTol	*real*	0.001	tolerance for criteria relative change over last iterations (interrupt non-MonteCarlo methods)
lnS_maxItr	*integer*	500	max amount of line search iterations (per line) (non-Monte Carlo methods)
lnS_tol	*real*	0.001	general relative tolerance for line search (non-Monte Carlo methods)
QN_nrmTol	*real*	0.001	tolerance for gradient norm to be considered zero (QN methods)
QN_pctDerv	*real*	0.02	factor to offset current parameter values for numerical derivative (QN methods)
QN_stencil	*integer*	2	numerical derivative stencil type (2, 4, 6 and 8 allowed) (QN methods)
QN_lambMax	*real*	0.9	factor of parameter interval, used to limit the step length of the line search within given parameter intervals (QN methods)
QN_lambAcc	*real*	1.618	factor increasing the step length of the line search (QN methods)

From row 22 and onward, model parameters to be calibrated are listed. The parameter is given as it is or with single quotation marks (e.g. 'cevp') followed by its values. For non-general parameters, values for all soil types/land uses/subbasins/parameterregions/etc have to be provided.

Each parameter is defined on three rows:

- **Row 1** specifies lower boundaries of the parameter range

- **Row 2** specifies upper boundaries of the parameter range (the model actually accepts lower and upper boundaries in any order)
- **Row 3** specifies either a minimum step width for parameter change *or*, in case of the DE method, a parameter specific factor to scale the random noise added to the proposed next-generation parameter, see description of DEMC settings code `DEMC_sigma` in table above.

NOTE: If lower and upper boundaries are identical, the parameter is omitted. This allows to calibrate a selection of the values for dependent parameters. The omitted parameters will not be included in the result in *bestsims.txt* and *allsim.txt*. The omitted parameters will be included in the result *respar.txt*.

Example of parameter rows in *optpar.txt*:

```
wcfc    0.100 0.020 0.120 0.050 0.250 0.250 0.150 0.050 0.500 0.500 0.050
wcfc    0.100 0.120 0.120 0.050 0.250 0.250 0.150 0.050 0.500 0.500 0.050
wcfc    0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001
ttmp  0.0   0.0
ttmp  2.0   2.0
ttmp  0.01  0.01
```

Explaination of example: The rows starting with wcfc are representing field capacity for 11 soil types, where the second soil type's wcfc is calibrated. Parameter ttmp is threshold temperature for 2 land uses, which are both calibrated.