User’s Manual

The Matlab software package *BaSaL* 1.0 was developed to conduct Bayesian inference of the pattern and rate of past sea-level changes from parse and error-prone observational data in terms of sea-level index points (SLIPs). As formulated in the main text, the functionality of the package essentially comprises two components: (1) age modeling, and (2) altitudinal modeling. The Markov chain Monte Carlo (MCMC) approach was adopted to infer these variables and other model parameters. Running the package is quite straightforward (Fig. 1).

```
Begin Bayesian sea-level modeling...
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 6).
Generating chain 1...
Generating chain 2...
Generating chain 3...
Parallel pool using the 'local' profile is shutting down.
Bayesian sea-level modeling completed successfully!
Monitor the convergence of chains...
R_hat = 1.0067 and the chain of T converged!
R_hat = 1.005 and the chain of S converged!
R_hat = 0.9999 and the chain of phi converged!
Post processing and plotting the results...
```

Fig. 1 Prompt of the BSL model run on the command window screen

A tab delimited spread sheet template was provided, which allow users to enter and organize their data (Fig. 2). The SLIPs should be arranged from the earliest to the latest age. The first column is “laboratory number”, the second column is “relative sea level (RSL) position”, the third column is “1σ RSL error”, the fourth column is “unit (m, or cm)”, the fifth column is “laboratory age”, the sixth column is “1σ age error”, the seventh column is “age type”, the eighth column is “reservoir age”, the ninth column is “1σ reservoir age error”, the 10th column is “calibration curve”, and the 11th column is “year the sample was dated”. BSL supports multisource data, including ^{14}C and non-^{14}C (e.g., OSL, U-Th) ages. For ^{14}C ages, users need to provide the reservoir age and reservoir age error if any and specify the calibration curve, while for non-^{14}C ages, a calibration curve is not needed. A total of five versions of the calibration curves (IntCal98, 04, 09, 13, 20) for the terrestrial and marine settings as well as the Southern Hemisphere were provided. Regarding the reservoir effect, users can supply both the reservoir age offset (ΔR) and local reservoir age (R). If ΔR is supplied, a calibration curve for the marine setting (e.g., marine20) should be applied; otherwise, a calibration curve for the terrestrial setting (e.g., intcal20) should be used.

	A	B	C	D	E	F	G	H	I	J	K
1	Lab_numbe	RSL	RSL_error	Unit	Age	Age_error	Age_type	Reservoir_a	R_error	Calibration	Year_dated
2	Lab-1241	400	20	cm	7700	110	OSL	NaN	NaN	None	2019
3	Lab-1240	300	36	cm	6400	80	14C	NaN	NaN	intcal20	2019
4	Lab-1239	-200	27	cm	5900	60	OSL	NaN	NaN	None	2019
5	Lab-1238	-100	32	cm	5200	80	14C	NaN	NaN	intcal20	2020
6	Lab-1230	100	28	cm	4560	30	U/Th	NaN	NaN	None	2020
7	Lab-1234	190	32	cm	4260	20	OSL	NaN	NaN	None	2019
8	Lab-1237	260	35	cm	3700	80	14C	300	200	marine20	2020
9	Lab-1236	190	32	cm	3000	40	14C	250	150	marine20	2019
10	Lab-1235	100	30	cm	2000	30	14C	150	100	intcal20	2019
11	Lab-1234	40	25	cm	1500	30	14C	100	50	marine20	2018

Fig. 2 Tab delimited spread sheet template for data entry

The only file users need to modify is *BaSaL_main.m,* which provides a textual interface enabling users to read their data, specify parameters for MCMC simulations, and run the model to obtain, analyze, and visualize the results. It contains the following steps.

Step 1. Setting up model parameters

In the Matlab environment, users need to open the main program *BaSaL_main* first and then specify a number of parameters (Fig. 3).

```
%%
data_file = 'SLIPs_template.txt';
age_scale = 'BP';
DT = 50;
delta = 5;
alpha = 0.05;
nsamples = 500;
nchains = 3;
burn_in = 1000;
thin = 20;
```

Fig. 3 An example of model set up

Here, the parameters are defined as followings:

data_file = the name of data file

age_scale = age scale

DT = size of the time grid

delta = the nearest years the modeled calendar ages to be rounded

alpha = the significance level for estimating confidence interval

nsamples = the number of samples to be kept for each chain

nchains = the number of chains to be run for convergence diagnosis
burn_in = the burn-in period of the Markov chains
thin = thinning interval of the Markov chains

Note that *BaSaL* supports three age scales: "BP", "B2K", and "BC/AD", which use different origins ("present"). For the "BP" scale, the present is AD 1950, while for the "B2K" scale, the present is AD 2000. The "BC/AD" scale is universal, but it is more convenient for archaeologists. If the data contain only non-^{14}C ages, the "B2K" scale is recommended; otherwise, the "BP" scale is preferred. Regarding convergence diagnosis, at least two chains for each variable or parameter should be run in parallel.

Step 2. Data preprocessing

This module comprises several functions (Table 1). The data are first read into two structures, *X*, and *Y*, which contain the chronological information and the altitudinal information of the data, respectively. Then, *BaSaL* corrects for the reservoir effect if any and coverts the ^{14}C ages to the F^{14}C space. Note that ^{14}C ages are always reported on the "BP" scale, while non-^{14}C ages are reported as "years before the year that the sample was dated." Therefore, *BaSaL* converts the non-^{14}C ages to the specified age scale. The F^{14}C values and non-^{14}C ages on the common age scale are stored in the augmented structure *X* for subsequent analyses.

BaSaL provides an automatic estimate of the two floating parameters, *A* and *B*, which define the beginning and the ending boundaries of the model time domain, respectively. Users can reset these parameters. For example, for the Holocene epoch, *B* can be replaced with 0 if the "BP" scale is chosen and data contain ^{14}C ages, and $B = 1950$ if the "BC/AD" scale is used. For the "B2K" scale, *B* can be reset to 0 if and only if the data do not contain ^{14}C ages; otherwise, *B* is set to 50. *BaSaL* then makes the temporal grid points by equally dividing the time domain into a number of grids with a width of *DT* years. Finally, *BaSaL* reads and extracts the distinct calibration curves associated with the ^{14}C ages from the data files to a structure.

Table 1 Matlab functions used for data preprocessing

Name	Functionality
read_data.m	Reading the SLIPs from a data file to two structures
data_process.m	Correcting for the reservoir effect and converting ^{14}C ages to F^{14}C as well as converting non-^{14}C ages to the designated age scale
age_bound.m	Estimating the early and late boundary of the model time domain
make_grid.m	Discretizing the time domain [*A*, *B*] and generating temporal grid points
read_curves.m	Reading calibration curves associated with the ^{14}C ages to a structure

Step 3. MCMC simulations

This is the core of the model, which consists of several functions (Tables 2). *BaSaL* first obtains the initial state of the variables and the model parameters automatically and then updates the state iteratively using a random walk method for the ages to be modeled and the Gibbs sampler for other variables and model parameters. Several (at least two) chains are run in parallel to allow convergence diagnosis. If the chains are converged, they are mixed by calculating the mean of these chains for subsequent analyses. For each variable and model parameter, random samples generated from the MCMC simulations are also saved to files.

Table 2 Matlab functions and dependence used for MCMC simulations

Name	Functionality
Initial_T.m	Initializing calendar ages, T
mcmc.m	Sampling T, S, ϕ, and hyperparameters using the MCMC method
update_T.m	Updating calendar ages, T, using the Metropolis-Hastings algorithm
age_prior.m	Calculating the prior probability of calendar ages, T
age_like.m	Calculating the likelihood given calendar ages, T
Initials.m	Initializing true sea level, S, and rate, ϕ
Gibbs.m	Updating S, ϕ, and hyperparameters using the Gibbs samplers
update_sigma2.m	Sampling σ^2 from the inverse Gamma distribution
update_kappa2.m	Sampling κ^2 from the inverse Gamma distribution
update_phi.m	Sampling ϕ from a truncated normal distribution
update_S.m	Sampling S from a truncated normal distribution
dtrandn_MH.m	Sampling a truncated normal distribution. Called by *update_phi.m* and *update_S.m*
convergence.m	Monitoring the convergence of Markov chains
save_mcmc.m	Saving the results from MCMC run to files

Step 4. Post processing and visualization

The last step is post processing and visualization, which invokes several functions (Table 3). *BaSaL* first generates descriptive statistics of the modeled calendar ages (T), relative sea-level curve (S), and rate of sea-level changes (ϕ) from MCMC runs. Then, *BaSaL* plots the modeled calendar ages, the relative sea-level curve, and the rate of sea-level changes, respectively. For the modeled calendar ages, *BaSaL* first estimates the empirical probability density function (pdf) of the modeled calendar ages through the numerical difference of the empirical cumulative distribution function, and then calculates the 95.4% (2σ) and the 68.2% (1σ) highest posterior density (HPD) regions of the modeled ages as well as the age at median probability. For the relative sea level (S) and the rate of sea-level changes (ϕ), *BaSaL* calculates the posterior mean and the 95% confidence interval from the MCMC runs. The results are saved automatically to files. Users can plot the SLIPs against either the unmodeled or modeled calendar ages (Fig. 4). The probability density function of both the modeled and unmodeled ages is scaled up to the 1σ altitudinal error of the SLIPs. Also, users can choose to plot the inferred sea-level curve on the backdrop of the SLIPs for comparison (Fig. 5) or plot the curve along with the rate of sea-level changes (Fig. 6).

Table 3 Matlab functions used for post processing

Name	Functionality
post_process.m	Generating descriptive statistics of T, S, and ϕ from MCMC runs
mcmc2pdf.m	Estimating the empirical pdf of the modeled calendar ages, T
pdf2hpd.m	Calculating the HPD regions of the modeled calendar ages, T

plot_slips0.m	Plotting SLIPs against unmodeled calendar ages
plot_slips1.m	Plotting SLIPs against modeled calendar ages, T
plot_rsl.m	Plotting modeled relative sea-level curve, S
plot_rate.m	Plotting modeled rate of relative sea-level changes, ϕ

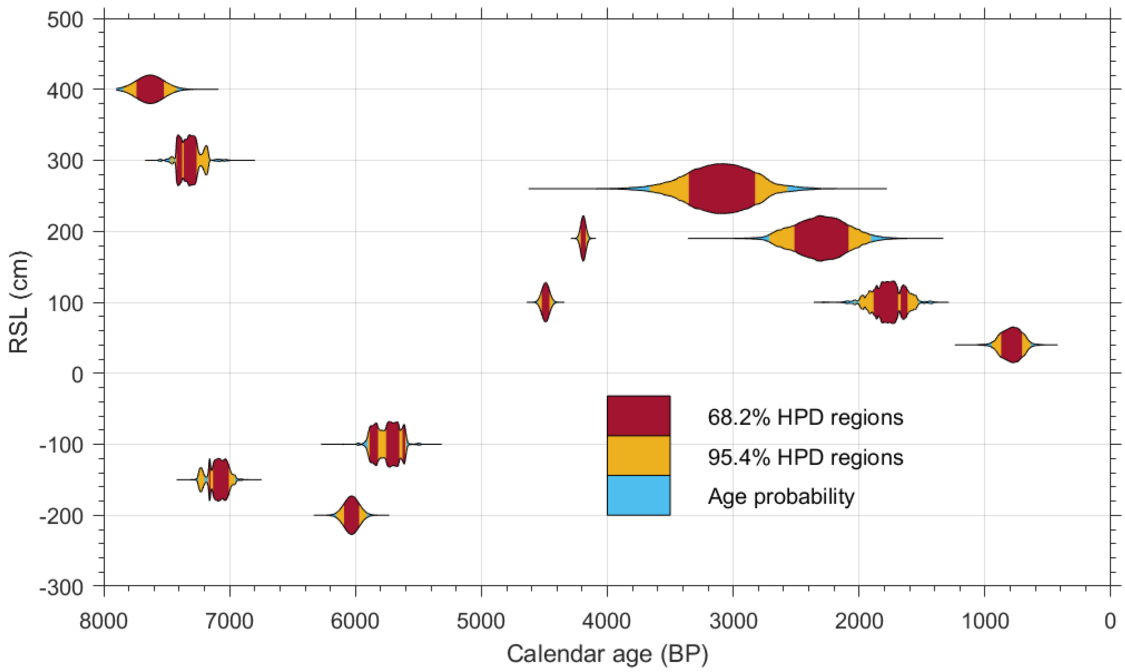

Fig. 4 Plot of SLIPs against unmodeled calendar ages

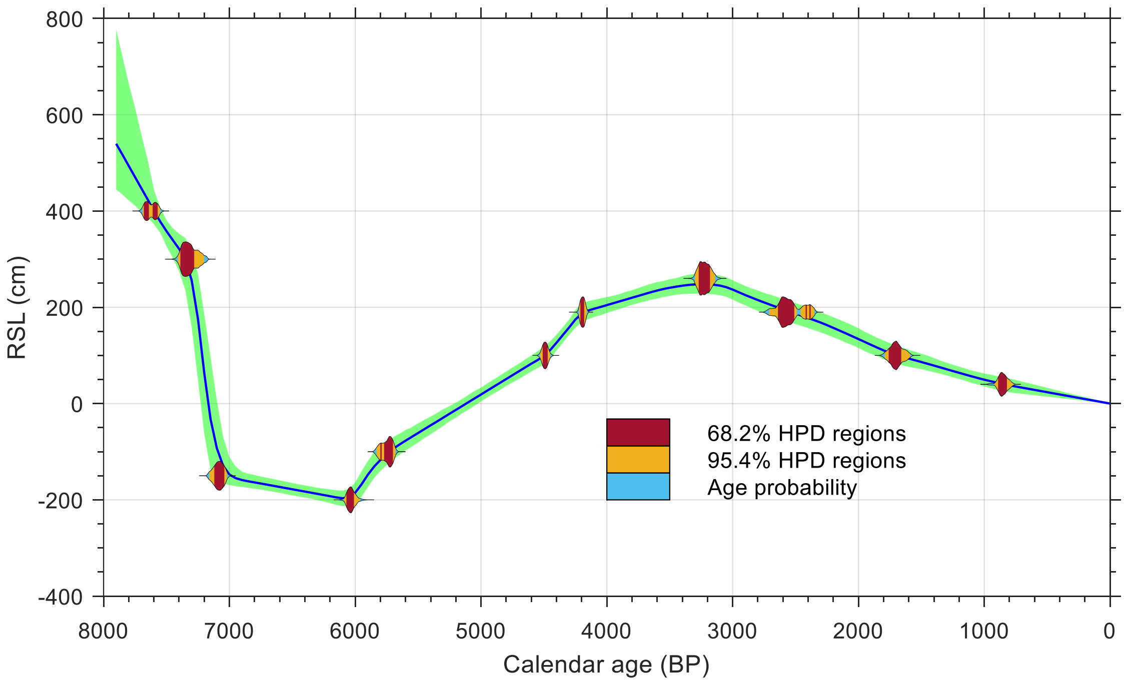

Fig. 5 Plot of inferred relative sea-level (RSL) curve along with SLIPs against modeled calendar ages. Solid blue line is the posterior mean and green envelope is the 95% confidence interval.

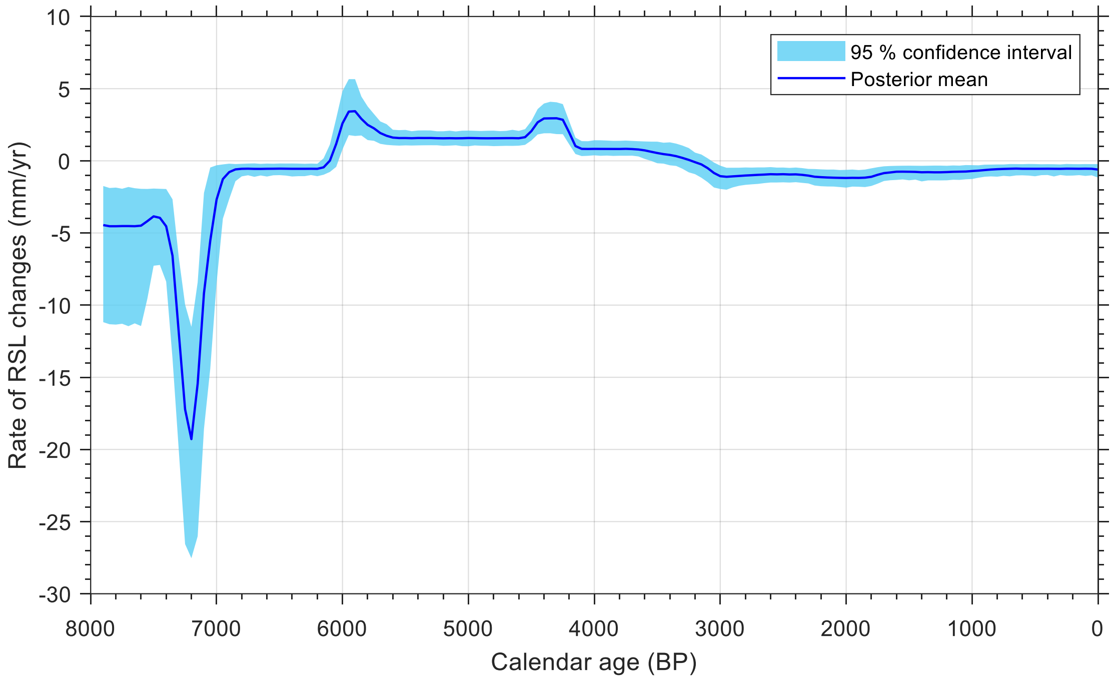

Fig. 6 Plot of inferred rate of relative sea-level (RSL) changes