Microbial-ENzyme Decomposition (MEND) Model
MANUAL

Gangsheng Wang
wanggs@ou.edu
Institute for Environmental Genomics
University of Oklahoma

Updates:

- 9/2/2015; 1/10/2017; 2/17/2017
- 4/6/2017: use MEND_namelist.nml to replace 'MEND.ini' & 'MENDcn.ini'
- 8/29/2018; 10/26/2018
- 11/1/2018:
  1) add a new parameter "fINP" to scale litter input rates
  2) set half-saturation constants VM=VP2=VP1, KP2=0.1*KP1, KM=10*KP1
- 12/1/2018: add 'Q10' method for temperature response function
- 1/31/2019: add MCMC and change OPT outputs
- Please ignore the Line numbers in 'MEND_namelist.nml', as I don't update the Line numbers any more.

References:

Wang G, Huang W, Mayes MA *et al.* (2019) Soil moisture drives microbial controls on carbon decomposition in two subtropical forests. *Soil Biology and Biochemistry,* **130**: 185-194.

Wang G, Jagadamma S, Mayes MA, Schadt CW, Steinweg JM, Gu L, Post WM. (2015) Microbial dormancy improves development and experimental validation of ecosystem model. *The ISME Journal*. **9**: 226-237. doi:10.1038/ismej.2014.120.

Wang G, Post WM & Mayes MA (2013) Development of microbial-enzyme-mediated decomposition model parameters through steady-state and dynamic analyses. *Ecological Applications* **23**: 255-272. doi: 10.1890/12-0681.1.

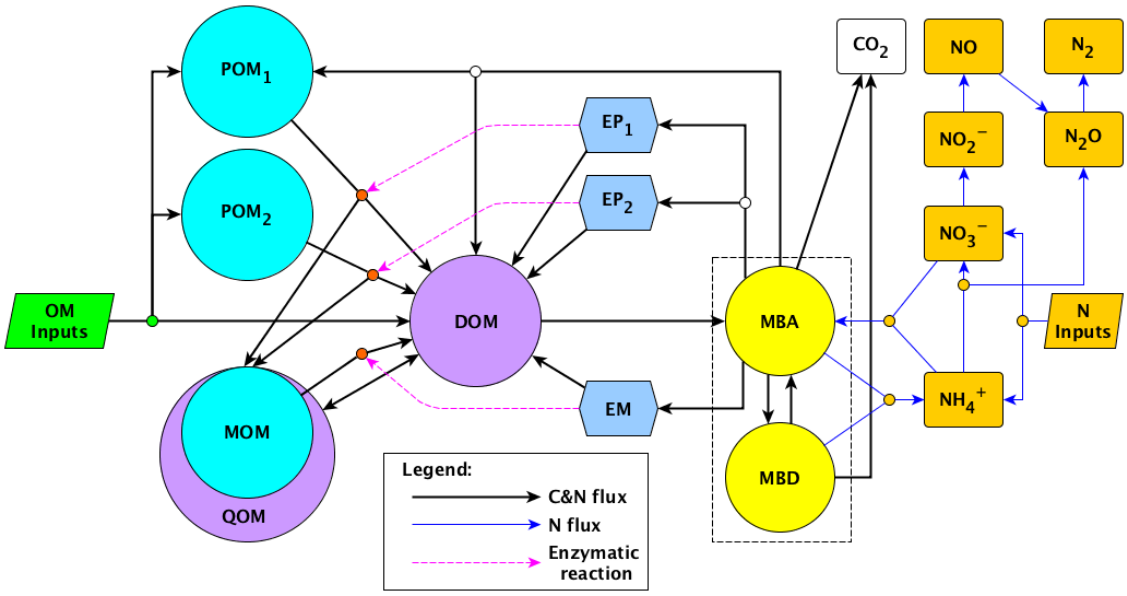


Fig. 1 MEND Diagram

OM: Organic Matter; **POM**: Particulate OM; **POM_1**: decomposed by oxidative enzymes (**EP_1**); **POM_2**: decomposed by hydrolytic enzymes (**EP_2**); **MOM**: Mineral-associated OM, decomposed by oxidative & hydrolytic enzymes (**EM**); **DOM**: Dissolved OM; **QOM**: adsorbed DOM; **MBA** & **MBD**: Active & Dormant Microbial Biomass.

Table of Contents

1. Model Repositories

Model Version	Notes
MEND	/Users/wg4/Dropbox (ORNL)/ORNL/MEND Carbon-Nitrogen coupled version: **sINI%Carbon_only = .FALSE.** https://wanggangsheng@bitbucket.org/mend_ornl/mend.git
MEND_mult	Multiple-Case Version of MEND: run multiple cases in one-run https://wanggangsheng@bitbucket.org/wanggangsheng/mend_mult

2. Directories

ID	Directory	Notes
0	model dir	Control Files: **MEND_namelist.nml**
1	src	Source code
2	Userio/inp	Input data **MEND_mult**: includes sub-folder for each case
3	Userio/out	Output data **MEND_mult**: includes sub-folder for each case

3. Major Subroutine Calls

3.1. MEND & MENDcn

0	1	2	3	4
MEND_main	MENDIN			
	SCEUA	fMEND_OBJ	Par: sINI%LCI0	
			Par: sINI%r0	
			subMEND_INI	
			subMEND_RUN	subMEND_PAR
				subMEND
				subMEND_output_rate subMEND_output
				sOUT_OPT_h
				sOUT_Day2Mon

3.2. MEND_mult

0	1	2	3	4
MEND_main	MENDIN			
	SCEUA	fMEND_OBJ	Par: sINI%LCI0	
			Par: sINI%r0	
			DO iCase=1,nCase	
			MENDIN_CASE	
			subMEND_INI	
			subMEND_RUN	subMEND_PAR
				subMEND
				subMEND_output_rate subMEND_output
				sOUT_OPT_h
				sOUT_Day2Mon
			END DO	

- '**SIM_obs.out**' combines all '*_SIM_obs.out' for all cases into 1 output file.
- After each MEND run, the total objective function (**fMEND_OBJ**) for multiple cases is calculated by reading data in this file.
- MEND_mult allows different calibration-variables used in various cases. e.g., case1 with 2 variables (CO2 & MBC), case2 with 2 variables (CO2 & DOC). Under this condition, the total OBJ (**fMEND_OBJ**) will include 3 single objectives (CO2, MBC, & DOC).
- 'MEND_namelist.nml' Cali_OBJ_Weight(·) defines OBJ weighting factors for these objectives combined from multiple cases; whereas the OBJ weighting factors for each case ('case.ini' Line 48-57) are NOT used for calibration/optimization.
- MEND_mult runs much slower than MEND. Because MEND_mult REPEATEDLY reads input data for each case (subroutine 'MENDIN_CASE') during each model run. However, MEND only read input data ONCE for a single case.

4. Source Code Files & Major Functions & Subroutines

ID	F90 file	Notes	Major Functions & Subroutines
1	MEND_main	Main program	
2.1	MEND_IN	Control file	
2.2	MENDIN_CASE	Input data for each case	
3	MOD_MEND_TYPE	Data structure for MEND	
4	MOD_MEND	MEND model; Depends on ▪ MOD_MEND_TYPE ▪ MOD_USRFS	○ subMEND: MEND model ○ subMEND_PAR: MEND parameters modified by temperature, moisture, etc. ○ subMEND_RUN: run model continuously with multiple time-steps ○ subMEND_INI: model initialization ○ fMEND_OBJ: objective function for model evaluation & optimization ○ sINP_Read: read input, soil temp & moisture ○ sOUT_OPT_h: extract HOURLY outputs for response variables used for optimization ○ sOUT_OPT: convert HOURLY data (sOUT_OPT_h) to DAILY or MONTHLY data ○ subMEND_output: HOURLY outputs for all state variables & fluxes ○ sOUT_tscale: convert HOURLY outputs (subMEND_output) to DAILY, MONTHLY & YEARLY outputs ○ sOUT_ALL_tscale: convert HOURLY outputs (subMEND_output) to DAILY, MONTHLY & YEARLY outputs, called by sOUT_tscale
5	MOD_OPT_TYPE	Data structure for model optimization	
6	MOD_OPT	Optimization algorithm	Depends on ▪ MOD_OPT_TYPE ▪ MOD_MEND
7	MOD_STRING	String utility	
8	MOD_USRFS	User Functions and Subroutines	
9	MOD_MCMC	MCMC algorithm	

5. Control & Output files

ID	FILE	Notes
	CONTROL FILE:	
1	MEND_namelist.nml	(1) MEND CONTROL file, stored in the model root dir ○ iModel = '0'-run MEND model with parameter sets 'Pinitial'; '1'-model calibration/optimization '2'- uncertainty quantification (UQ) using COFI method, output UQpar.out '3'-generate data for Sobol Sensitivity analysis '4'-UQ using MCMC '5'-UQ with COFI, output both UQpar.out & UQvar.out (see Table 7) ○ sSite: site name (2) MEND_mult CONTROL file ▪ Line 11-14: define multiple cases, case names are dir names in inp & out
2	**.ini	CONTROL/INITIAL file for each case, in inp/casedir
	INITIAL FILE:	
1	SOIL_INI.dat	Initialization of SOC pools, stored in 'userio/inp'
	OUTPUT FILES:	**MEND_mult**: Outputs of optimization for all-case are saved in out; Outputs for each case are saved in the out/casedir
1	*_SIM_obs.out	Outputs for SIM vs. OBS for those specific days with OBS. VAR: corresponds to VARid in 'MEND_namelist.nml'. Note: The 'best' parameter set among those 'best' from multiple opt-runs is shown at the last line of this file. You may copy this line to 'MEND_namelist.nml' for future run.
2	*_SIM_day.out	Continuous daily SIM results for those variables used for calibration
3	*_SIM_mon.out	Continuous monthly SIM results for those variables used for calibration
4	*_VAR_hour.out	All state variables: hourly
5	*_VAR_day.out	All state variables: daily
6	*_VAR_mon.out	All state variables: monthly
7	*_VAR_year.out	All state variables: yearly
8	*_FLX_hour.out	All fluxes: hourly
9	*_FLX_day.out	All fluxes:daily
10	*_FLX_mon.out	All fluxes: monthly
11	*_FLX_year.out	All fluxes: yearly
12	*_ITW_hour.dat	External input (litter), T, SWC, SWP: hourly
13	*_ITW_day.dat	External input (litter), T, SWC, SWP: daily
14	*_ITW_mon.dat	External input (litter), T, SWC, SWP: monthly
15	*_RATE_hour.out	Derived rates: hourly, e.g., active fraction
16	*_RATE_day.out	Derived rates: daily
17	*_RATE_mon.out	Derived rates: monthly
18	*_RATE_year.out	Derived rates: yearly
19	*_PAR_hour.out	Parameters modified by T, SWP, pH, etc: hourly
20	*_PAR_day.out	Parameters modified by T, SWP, pH, etc: daily
21	*_PAR_mon.out	Parameters modified by T, SWP, pH, etc: monthly
22	*_PAR_year.out	Parameters modified by T, SWP, pH, etc: yearly
23	*_OPT_end.out	'best' parameter sets from multiple independent optimizations
24	*_OPT_ini.out	Initial parameters for optimization, best parameter sets in each loop
25	*_OPT_all.out	All parameter sets during optimization, used for UQ
26	*_Ndep_hour.out	Mineral N input: hourly

6. Derived Rates: *_*RATE*_hour.out

Col	Name	Units	Notes
1	Hour		Hourly time-step
2	kPOC1	h^{-1}	Equivalent 1st-order decomposition rate; k=VP1*EP1/(POC1 + KP1)
3	kPOC2	h^{-1}	Equivalent 1st-order decomposition rate; k=VP2*EP2/(POC2+ KP2)
4	kMOC	h^{-1}	Equivalent 1st-order decomposition rate; k=VM*EM/(MOC+ KM)
5	kDOC	h^{-1}	Equivalent 1st-order turnover rate; k=[(Vg+Vmt)/Yg]*MBa/(DOC + KD)
6	kMBa	h^{-1}	Equivalent 1st-order turnover rate; k=[(Vg+Vmt)*(1/Yg−1)]*phi + rMORT + pEP+ pEM) * Vmt
7	kMBa_in	h^{-1}	Equivalent 1st-order microbial uptake rate, k=[(Vg+Vmt)/Yg]*phi + Resuscitation/MBa
8	kMBd	h^{-1}	Output rate of dormant microbes, k=(Resuscitation+Maintenance)/MBCd
9	kMBd_in	h^{-1}	Input rate for dormant microbes, k=Dormancy/MBCd
10	kMB	h^{-1}	Turnover rate of total MBC, k=(CO2_gm+ENZ_prod+Mortality)/MBC
11	kMB_in	h^{-1}	Assimilation rate of total MBC, k=uptake/MBC
12	Phi	–	DOC saturation level, = DOC/(DOC + KD)
13	Active_Fraction	–	Fraction of active microbes
14	CUE	–	Apparent carbon use efficiency = [uptake - CO2_gmo]/uptake
15	NUE	–	Apparent nitrogen use efficiency = $[CN_{MB} - CN_{MB\_min}]/(CN_{MB\_max} - CN_{MB\_min})$
16	Balance_Error	mg C g^{-1} soil or mg C cm^{-3} soil	Error for balance check, RE=(TOCend − TOCbeg) − (TOCinp − TOCout)*dt
17	TOCbeg	mg C g^{-1} soil mg C cm^{-3} soil	Total organic carbon at the beginning of the time-step
18	TOCend	Same as above	Total organic carbon at the end of the time-step
19	TOCinp	Same as above	TOC input during the time-step
20	TOCout	Same as above	TOC output during the time-step
21	STP	°C	Soil temperature
22	SWC	$cm^3\ cm^{-3}$	Soil water content
23	SWP	MPa	Soil water potential
24	pH	–	Soil pH

7. Uncertainty Quantification (UQ): Inputs & Outputs

iModel = 2

Category	File Name	Notes
Input File	UQpar.dat	Copy '*_OPT_all.out', add "OBJ_critical= J_{cr}" to 1st line, where J_{cr} denotes the critical OBJ for UQ: $J_{cr} = J_{opt} \cdot \eta = J_{opt} \cdot \left(1 + \frac{p}{n-p} F_{\alpha,p,n-p}\right)$ J_{opt} is the optimum (min) OBJ, n is the number of OBS, p is the number of PARs, and $F_{\alpha,p,n-p}$ is the value of the F-distribution for α, p, and $n-p$.
Output Files	*_UQpar.out	Save parameters that result in fObj <= J_{cr}
	*_UQvar.out	Save variables on those observational time steps predicted by the PARs in '*_UQpar.out'

8. sINI%iScenario

Scenario design: data used to conduct scenario analysis beyond calibration period

See 'MEND_namelist.nml' siScenario

iScenario	Notes
1	Repeatedly use 1-yr mean hourly data derived from observed data
2	Repeatedly use multiple-year hourly (observed) data

9. Scenario Design

Parameters controlling the changes in Temperature, Water content & Litterfall Input; specifically, for the subtropical Dinghushan case study, may not be applied to other case studies

See 'MEND_namelist.nml' Scenario design

Parameter	Note
STP_delta	Gradually annual change in Temperature, °C/a
SWC_logis(3)	$$\frac{\theta(t)}{\theta(t_0)} = \frac{p}{1-(1-p)\cdot e^{-r(t-t_0)}}$$ Soil Water Content (SWC, θ) parameters in logistic equation; SWC DECREASES with time (1) $p \in (0,1)$: lower bound for the ratio of $\frac{\theta(t\to\infty)}{\theta(t_0)}$ (2) r > 0: steepness; r = 0 means no changes (3) t_0: reference year
SIN_logis(4)	$$\frac{L(t)}{L_{max}} = \frac{1}{1+\exp\left[\beta_0 - \beta_1\cdot(t-t_0)\right]}$$ Litterfall input parameters in logistic equation; Litterfall INCREASES with time (1) $\beta_0 > 0$: intercept (2) $\beta_1 > 0$: steepness; $\beta_1 = 0$ means no changes (3) t_0: reference year (4) fDOC_delta (NOT included in the logistic equation): gradually annual change of DOC fraction in SOC input

10.sINI%iKenetics

Decomposition Kinetics for POC/MOC

See 'MEND_namelist.nml' siKinetics (**MEND_mult**: Line22-23)

iKenetics	Mic-Enz	Kinetics	Equation	Sample MEND_namelist.nml
0	Both	Michaelis-Menten	$F_{dec} = \frac{V_M \cdot E \cdot S}{K_M + S}$	MEND_Enz-Mic.ini
1	No Enz	First Order	$F_{dec} = k_M \cdot S$	MEND_No-Enz.ini
2	Both	Second Order	$F_{dec} = k_M \cdot E \cdot S$	
11	None	First Order	$F_{dec} = k_M \cdot S$	MEND_No-Enz-Mic.ini

11.sINI%iHR

Calculation Method for Growth & Maintenance Respiration from Active Microbes

iHR	Method	Notes
0	$HR_g = \max\left[D, \left(\frac{1}{Y_G} - 1\right)\frac{V_g \cdot BA \cdot D}{K_D + D}\right]$ $HR_m = \max\left[D, \left(\frac{1}{Y_G} - 1\right)\frac{V_m \cdot BA \cdot D}{K_D + D}\right]$	HR_g or HR_m is constrained by DOC
1	$HR_{gm} = \max\left[D, \frac{1}{Y_G} \times \frac{\left(V_g + V_m\right) \cdot BA \cdot D}{K_D + D}\right] \times \left(1 - Y_g\right)$ $HR_g = HR_{gm} \times \frac{V_g}{V_g + V_m}, HR_m = HR_{gm} \times \frac{V_m}{V_g + V_m}$	Total microbial uptake is constrained by DOC, HR_g or HR_m is a fraction of total uptake

12.sINI%iTmp_Func

Temperature Response Function

iTmp_Func	fT	Notes
0	fTArh: Arrhenius Equation	See function in MOD_MEND
1	fTQ10: Q10 method	See function in MOD_MEND

13.Input Data

Note: all input data will be converted to hourly rate; see input data format in dir: userio/inp (**MEND_mult**: see casedir in inp)

Input Data	Notes/Sample data	Time scale
Litter input	1) Continuous input, e.g., litter fall, 'MEND_namelist.nml' Type-1 Input	Monthly, daily, or hourly
	2) Other constant input, e.g., only annual amount available, 'MEND_namelist.nml' Type-2 Input	Convert annual amount to hourly rate
	3) Other constant input during a specific period, e.g., dead roots, 'MEND_namelist.nml' Type-3 Input	Specify total amount & the period (beginning & ending dates)
Soil temperature	'MEND_namelist.nml' preferred units: °C STP.dat	Hourly, daily, or monthly
Soil moisture	'MEND_namelist.nml' preferred units: % will be converted to soil water potential (MPa) using the retention curve parameters given in soil retention curve See 'fSWC2SWP' in 'MOD_MEND.F90' SWC.dat	Hourly, daily, or monthly
Mineral N input	NH4dep.dat; NO3dep.dat	Monthly
C:N ratios	1) Input C:N, see 'MEND_namelist.nml' 2) SOM C:N, see SOIL_INI.dat	

14.MEND Parameters

('MEND_namelist.nml' INITIAL PARAMETERS) (**MEND_mult**: Line 28-57)

14.1 MEND_Carbon_only Parameters

ID	Parameter	Description	Apriori range	Units
1	LF_0	Initial fraction of P_1, $LF_0 = P_1/(P_1+P_2)$	(0.1, 1.0)	—
2	r_0	Initial active fraction of microbes, $r_0 = BA/(BA+BD)$	(0.01, 1)	—
3	$fINP$	Scaling factor for litter input rate	(0.1, 1)	—
4	V_{P1}	Maximum specific decomposition rate for P_1	(0.1, 100)	mg C mg^{-1} C h^{-1}
5	V_{P2}	Maximum specific decomposition rate for P_2	(0.1, 100)	mg C mg^{-1} C h^{-1}
6	V_M	Maximum specific decomposition rate for M	(0.1, 100)	mg C mg^{-1} C h^{-1}
7	K_{P1}	Half-saturation constant for P_1 decomposition	(10, 100)	mg C g^{-1} soil
8	K_{P2}	Half-saturation constant for P_2 decomposition	(1, 10)	mg C g^{-1} soil
9	K_M	Half-saturation constant for M decomposition	(100, 1000)	mg C g^{-1} soil
10	Q_{max}	Maximum sorption capacity	(0.5, 5)	mg C g^{-1} soil
11	K_{ba}	Binding affinity	(1, 16)	(mg C g^{-1} soil) $^{-1}$
12	K_{des}	Desorption rate, Sorption rate $K_{ads} = K_{des} \times K_{ba}$	(0.0001, 0.01)	mg C g^{-1} soil h^{-1}
13	r_E	Turnover rate of EP_1, EP_2, and EM	(0.0001, 0.01)	mg C mg^{-1} C h^{-1}
14	p_{EP}	$[V_{mt} \times p_{EP}]$ is the production rate of EP (EP_1 + EP_2), V_{mt} is the specific maintenance rate for active microbes	(0.0001, 0.05)	—
15	fp_{EM}	$fp_{EM} = p_{EM}/p_{EP}$, $[V_{mt} \times p_{EM}]$ is the EM production rate	(0.1, 2.0)	—
16	f_D	Fraction of decomposed P_1 and P_2 allocated to D	(0.05, 1)	—
17	g_D	Fraction of dead BA allocated to D	(0.01, 1)	—
18	V_g	Maximum specific uptake rate of D for growth	(0.001, 0.1)	mg C mg^{-1} C h^{-1}
19	$alpha$	$= V_{mt} / (V_g + V_{mt})$	(0.01, 0.5)	—
20	K_D	Half-saturation constant for microbial uptake of D	(0.0001, 0.5)	mg C g^{-1} soil
21	Y_g	True growth yield	(0.1, 0.64)	—
22	Y_{g_sl}	Slope for Y_G dependence of temperature, see function 'fT_CUE' in 'MOD_MEND.F90'	(0.001,0.016)	1/°C
23	Q_{10}	Q10 for T response function. See function 'fTQ10' in 'MOD_MEND.F90'	(1.1, 5)	—
24	$gamma$	Max microbial mortality rate = $V_{mt} \times gamma$	(0.01, 20)	—
25	$beta$	Ratio of dormant maintenance rate to V_{mt}	(0.0005,0.05)	—
26	WP_{A2D}	Soil water potential (SWP) threshold for microbial dormancy; note that WP_{A2D} & WP_{D2A} are |SWP|, since SWP <0. See functions 'fSWP_A2D' & 'fSWP_D2A' in 'MOD_MEND.F90'	(0.2, 0.6)	–MPa
27	tau	$WP_{D2A} = WP_{A2D} \times tau$, WP_{D2A} is the SWP threshold for microbial resuscitation	(0.1, 0.8)	—
28	w_{dorm}	Exponential in SWP function for microbial dormancy or resuscitation,	(0.5, 6)	—

14.2 Additional parameters for Nitrogen dynamics

ID	Parameter	Description	Apriori range	Units
29	*VNup_MB*	Mineral N uptake rate by MBA	(0.001, 0.05)	mg N mg^{-1} N h^{-1}
30	*KsNH4_MB*	half-saturation constant for NH_4 uptake by MBA	(0.001, 0.1)	mg N g^{-1} soil
31	*KsNO3_MB*	half-saturation constant for NO_3 uptake by MBA	(0.001, 0.1)	mg N g^{-1} soil
32	V_{nit}	Maximum nitrification rate	(0.001, 0.1)	h^{-1}
33	V_{denit}	Maximum denitrification rate	(0.001, 0.1)	h^{-1}
34	*YgN*	Max nitrogen use efficiency	(0.5, 1.0)	—

Note1: 'MEND_namelist.nml' Cali_Calibrate (**MEND_mult**: Line 31-57), Column 'Calibrate': '1' means the parameter will be calibrated, '0' mean no-calibration.
Note2: 'MEND_namelist.nml' Pintital (**MEND_mult**: Line 61), initial parameter values, will override the values in Column 'Initial' at Line 78-104.

14.3 MEND Parameters that may be pre-determined

Parameter	Regression Equation	R^2	p-value
K_{P1} (mg C g^{-1} soil): half-saturation constant for POC decomposition	K_{P1} = 41.58 × DOC + 44.95 K_{P1} = 0.62 × (%Sand) + 45.56 K_{P1} = 1.07 × POC + 50.13	0.97 0.82 0.64	0.01 0.10 0.20
K_M (mg C g^{-1} soil): half-saturation constant for MOC decomposition	K_M = 455.11 × DOC + 400.42 K_M = 4.13 × MOC + 425.75	0.98 0.73	0.01 0.15
K_D (mg C g^{-1} soil): half-saturation constant for microbial uptake of DOC	K_D = 0.33 × MBC + 0.21	0.63	0.21
Q_{max} (mg C g^{-1} soil): maximum sorption capacity	Q_{max} = 0.012 × MOC + 3.43 Q_{max} = 0.011 × SOC + 3.39	0.98 0.91	0.01 0.04
K_{ba} ([mg C g^{-1} soil] $^{-1}$): binding affinity	K_{ba} = −0.082 × (%Sand) + 11.23	0.84	0.08
K_{des} (mg C g^{-1} soil h^{-1}): desorption rate	K_{des} = −0.0059 × DOC + 0.0059	0.95	0.02
$V_{death} = V_{mt} \times gamma$: microbial death rate	[Hansen, 1990, DAISY] 0.001 – 0.01 d^{-1}, 4×10^{-5} – 4×10^{-4} h^{-1}		

Notes:
SOC: soil organic carbon; POC: particulate organic carbon; MOC: mineral-associated organic carbon; DOC: dissolved organic carbon; MBC: microbial organic carbon. Units: mg C g^{-1} soil.
%Sand: percentage of sand in soil.

15.Response Variables for Model Calibration/Optimization

Note: 'MEND_namelist.nml' CALIBRATION DATA; see data format in dir: userio/inp
MEND_mult: Line 45-57 in '*.ini' in 'userio/inp/casedir'

Column	Column Name	Notes
1	VARid	Variable ID
2	VAR	Variable Name
3	Units	Units
4	Calibrate	'1'-data available for calibration, '0'-unavailable
5	tstep	Time-step: 0-hourly, 1-daily, 2-monthly,5-mean. Usually the observations are regarded as data measured during an hourly-scale, the simulation results will be averaged during 1 day (24-h) to compare with the observations
6	Obs_file	File with observation data if available
7	Obs_file_column	Column id for the data, currently not used
8	OBJ	Type of objective function (minimization): ▪ NSEC: Nash-Sutcliffe Efficiency Coefficient (Coefficient of Determination), see 'f1NSE' in 'MOD_USRFS.F90' ▪ MARE: Mean Absolute Relative Error, see 'fMARE' in 'MOD_USRFS.F90' ▪ CORR: correlation coefficient, see 'f1CORR' ▪ CORl: CORR with log10-tranformed data, see 'f1CORR' ▪ MARt: MARE with tolerance, see 'fMARE_tolerance' ▪ NSEn: NSEC for normalized data, see 'f1NSE_norm' ▪ MARn: MARE for normalized data, see 'fMARE_norm' ▪ AVGr: |ratio – AVGsim/AVGobs|, see 'f1RAVG_ratio'
9	OBJ_Weight	Weighting factor for each OBJ, will be normalized in the code

16.Parameters for the Optimization Algorithm

Note: '**MEND_namelist.nml**' SCE parameters, (**MEND_mult**: Line 16-20)
only 3 parameter '**nrun**' , '**iniflg**' and '**iprint**' are required to edit

Parameter	Notes
nrun	# of runs for optimization, these opt-runs are independent of each other. There is a different random seed for each opt-run. nrun <= 200
iniflg	flag on whether to include the initial point (Line 108) in population = 0, not included = 1, included
iprint	Flag on whether to print all points generated during optimization = 0, no printing =1, print to *_OPT_all.out for COFI UQ analysis

17. sINI%VAROpt_int

sINI%VAROpt_int(sINI%nVAROpt, 3)

Column1: i	Column2: n	Column3: t: tstep
Index of calibrated output VARiable	# of observations	0(hourly) 1(daily) 2(monthly) 3(seasonal): to_do 4(yearly) 5(mean, excluding the first 10% data)

18.CN Ratios in 3 Litter Pools

3 litter pools: Lignin, Cellulose, and Labile
Assume their fractions in Carbon mass = f_Lig, f_Cel, f_Lab
f_Lig + f_Cel + f_Lab = 1
Assume CN_Cel = 500
Assume CN_Lig/CN_Lab = rCN
Assume overall CN in litter pool = CN
N mass balance:

$$N = \frac{C}{CN} = \frac{C \cdot f_{Lig}}{CN_{Lab} \cdot rCN} + \frac{C \cdot f_{Cel}}{CN_{Cel}} + \frac{C \cdot f_{Lab}}{CN_{Lab}}$$

thus

$$CN_{Lab} = \left[\frac{f_{Lig}}{rCN} + f_{Lab}\right] \Big/ \left[\frac{1}{CN} - \frac{f_{Cel}}{CN_{Cel}}\right]$$

$$CN_{Lig} = CN_{Lab} \times rCN$$

19.Configuration/Compiling on OSX & Cygwin/Unix

19.1 Dependency of source code

- Edit 'nbproject/**Configurations.xml**' to define dependency of source code for compilation, e.g.,

MOD_MEND depends on 2 modules (MOD_MEND_TYPE & MOD_USRFS);
MOD_OPT depends on 3 modules (MOD_OPT_TYPE, MOD_MEND, & MOD_USRFS).

- See './**Configurations.xml**' for example:

```
<item path="src/MEND_IN.F90" ex="false" tool="2" flavor2="0">
 <fortranCompilerTool>
  <additionalDep>${OBJECTDIR}/src/MOD_STRING.o
${OBJECTDIR}/src/MOD_USRFS.o ${OBJECTDIR}/src/MOD_MEND_TYPE.o
${OBJECTDIR}/src/MOD_MEND.o
${OBJECTDIR}/src/MOD_OPT_TYPE.o</additionalDep>
 </fortranCompilerTool>
</item>
<item path="src/MEND_main.F90" ex="false" tool="2" flavor2="0">
 <fortranCompilerTool>
  <additionalDep>${OBJECTDIR}/src/MOD_USRFS.o
${OBJECTDIR}/src/MOD_MEND_TYPE.o ${OBJECTDIR}/src/MOD_MEND.o
${OBJECTDIR}/src/MOD_OPT_TYPE.o
${OBJECTDIR}/src/MOD_OPT.o</additionalDep>
 </fortranCompilerTool>
</item>
<item path="src/MOD_MEND.F90" ex="false" tool="2" flavor2="0">
 <fortranCompilerTool>
  <additionalDep>${OBJECTDIR}/src/MOD_USRFS.o
${OBJECTDIR}/src/MOD_MEND_TYPE.o</additionalDep>
 </fortranCompilerTool>
</item>
<item path="src/MOD_MEND_TYPE.F90" ex="false" tool="2" flavor2="0">
</item>
<item path="src/MOD_OPT.F90" ex="false" tool="2" flavor2="0">
 <fortranCompilerTool>
  <additionalDep>${OBJECTDIR}/src/MOD_USRFS.o
${OBJECTDIR}/src/MOD_MEND.o
${OBJECTDIR}/src/MOD_OPT_TYPE.o</additionalDep>
 </fortranCompilerTool>
</item>
<item path="src/MOD_OPT_TYPE.F90" ex="false" tool="2" flavor2="0">
</item>
<item path="src/MOD_STRING.F90" ex="false" tool="2" flavor2="0">
</item>
<item path="src/MOD_USRFS.F90" ex="false" tool="2" flavor2="0">
</item>
```

19.2 Install gcc/gfortran

(1) OSX
https://github.com/fxcoudert/gfortran-for-macOS/releases

(2) Windows (install gcc/gfortran/make)
Cygwin, https://www.cygwin.com/

19.3 NetBeans Configuration on OSX

(3) Install NetBeans with JDK

- NetBeans 9.0

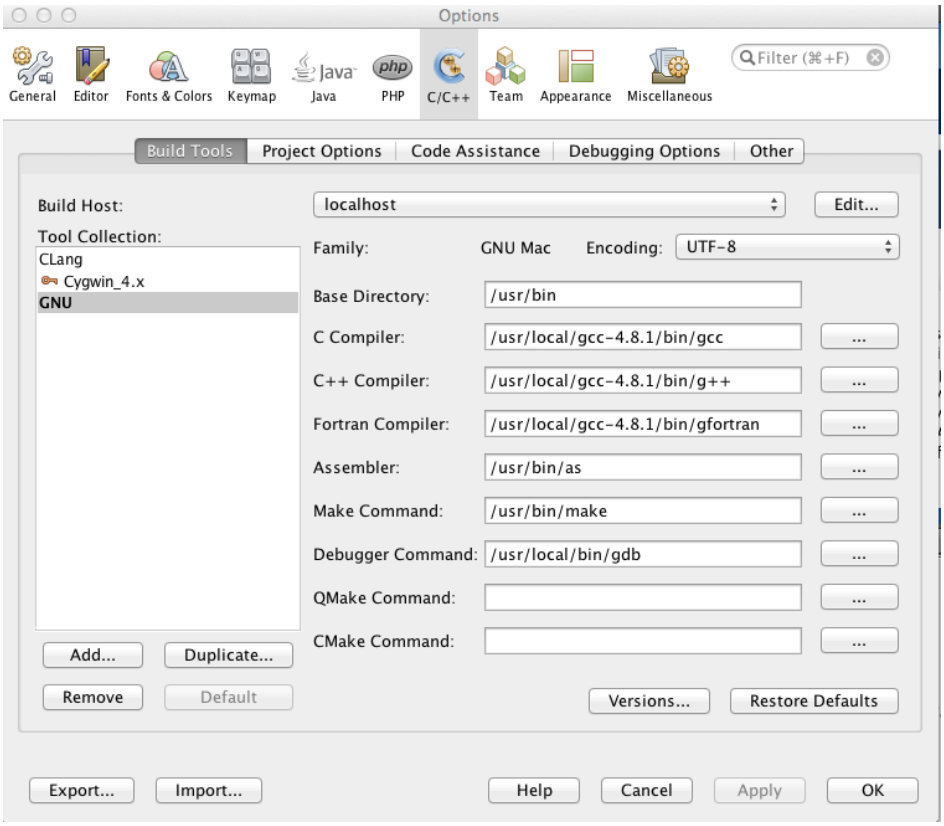
https://github.com/carljmosca/netbeans-macos-bundle
./install.sh

- C/C++ plugin for NetBeans 9.0
  - **Tools > Plugins > Settings** tab **>** click the **Add** button.
  - On the **Update Center Customizer** screen:
    - Enter some value in the **Name** field (e.g. *"My plugins"*),
    - Enter http://updates.netbeans.org/netbeans/updates/8.2/uc/final/distribution/catalog.xml.gzin the **URL** field
    - Click the **OK** button.
  - This should create a new entry in the **Configuration of Update Centers** list in the **Settings**tab.
  - Checking that new entry should instantly add plugins to the **Available Plugins** tab.
  - Click the **Available Plugins** tab, then click the **Category** column to sort the entries by category.
  - The **Name** of the entry at the top of the list should be **C/C++**. If so, you have successfully made the plugin available:

- JDK 11

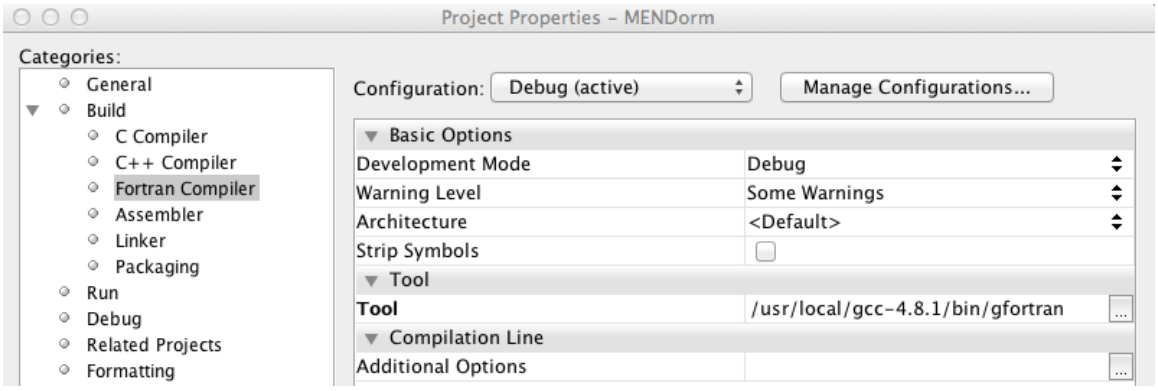
https://www.oracle.com/technetwork/java/javase/downloads/jdk11-downloads-5066655.html

(4) gcc/gfortran Compiler configuration
(i) preference->Options->C/C++->Build Tools
If something is missing (e.g., gfortran), we need to install it first, then click the button "…" to locate the package.
If we want to debug the code, please install "gdb" first.



(5) Project Properties->Build->Fortran Compiler->Tool
Specify the absolute path for "gfortran"
Additional Options:
-ffpe-trap=invalid,zero,overflow (check floating point exceptions)
-finit-local-zero (initialize local INTEGER, REAL,and COMPLEX variables to zero, LOGICAL variables to false, and CHARACTER variables to a string of null bytes)

/Users/wgs/Dropbox_job/Model/MEND/nbproject/Makefile-Debug.mk:
# Fortran Compiler Flags
FFLAGS=-ffpe-trap=invalid,zero,overflow



(6) We can compile a single FORTRAN file by right-clicking the file name and click "Compile File (F9)"
(7) If a module is modified, e.g., a new variable is declared in the module, suggest to "Clean" the project then re-do "build": right-click project name->More Build Commands->Clean Project.
(8) Run model in NetBeans: click icon 

19.4 Compiling on Cygwin/Unix

(9) Copy the NetBeans-configured MEND folder to another system (Windows-Cygwin or Unix)
(10) Edit file "**/nbproject/Makefile-Debug.mk**": replace "**FC=/**/gfortran**" with the full path of "gfortran" on the new machine system
(11) "cd" into the MEND model root-dir
(12) make clean
(13) **make -k**
(14) The "**mendcn.exe**" file will be generated if there's no errors after n times
(15) If the path for "mendorm.exe" is "/dist/Debug/…/mendorm.exe", type "**./dist/Debug/…/mendcn.exe**" to run model
(16) Another option to run model: copy "/dist/Debug/…/mendorm.exe" to root-dir of MEND model, type "**./mendcn.exe**"

19.5 Install gdb on Mac OS High Sierra

(1) sudo port install gdb@8.0.1
Note: version after gdb8.0.1 does NOT compatible with High Sierra
(2) ln -sf /opt/local/bin/ggdb /usr/local/bin/gdb
(3) create gdb certificate (e.g., gdbcert)
1) Open Applications/Utilities/Keychain Access
2) In menu, open **Keychain Access > Certificate Assistant > Create a Certificate**
3) Give it a name (e.g. `gdbcert`)
Identity type: Self Signed Root
Certificate type: Code Signing
Check: Let Me Override Defaults
4) Continue until "Specify a Location For"
5) Set Keychain location to System. If this yields the following error: `Certificate Error: Unknown Error =-2,147,414,007` Set Location to Login, Unlock System by click on the lock at the top left corner and drag and drop the certificate gdbcert to the System Keychain.
6) Create certificate and close Certificate Assistant.
7) Find the certificate in System keychain.
8) Double click certificate.
9) Expand **Trust**, set **Code signing** to `Always Trust`
(4) Restart taskgated in terminal: sudo killall taskgated
(5) (Option) Enable root account:
- Open System Preferences

- Go to User & Groups > Unlock
- Login Options > "Join" (next to Network Account Server)
- Click "Open Directory Utility"
- Go up to **Edit > Enable Root User**

(6) Codesign gdb certificate: sudo codesign -fs gdb801 /usr/local/bin/gdb
sudo codesign -fs gdbcompile /usr/local/gdb-8.0.1/bin/gdb
sudo codesign -fs gdbnew /usr/local/gdb-8.0.1/bin/gdb

(7) (Option) Disable root account (see #10)

(8) codesign -v /usr/local/gdb-8.0.1/bin/gdb
if this command doesnot print anything, it means the binary is correctly signed.

(9) gdb does NOT work in NetBeans. However, it will work if we run NetBeans with sudo:
sudo '/Applications/NetBeans/NetBeans 8.2.app/Contents/Resources/NetBeans/bin/netbeans'

(10) gdb works in command line: sudo gdb ./mend

First, download `gdb-8.0.1` sources from mirrors. Then add to `gdb-8.0.1/bfd/mach-o.c` the following code at line `4649` :

```
case BFD_MACH_O_LC_BUILD_VERSION:
break;
```

And finally add int `gdb-8.0.1/include/mach-o/loader.h` :

```
  BFD_MACH_O_LC_BUILD_VERSION = 0x32
```

at line `189` (don't forget to add a `,` at the end of the line 188
after `BFD_MACH_O_LC_VERSION_MIN_WATCHOS = 0x30`).
After these instructions you can follow a classic `gdb` compilation as indicate inside the README :

```
run the ``configure'' script here, e.g.:

    ./configure
    make

To install them (by default in /usr/local/bin, /usr/local/lib, etc),
then do:
    make install
```