Documentation of a Simple Land Data Assimilation System (SLDAS)

Jean-François MAHFOUF

19 January 2024

1. Introduction

A simple land data assimilation system (SLDAS) has been designed to evaluate the behavior of various assimilation techniques at local scale using the land surface scheme ISBA described in Noilhan and Mahfouf (1996) and simulated observations (so-called "twin experminents"). Such experimental set-up allows to understand rather easily the results obtained since the solution of the inversion problem is known.

2. Main features of the system

The experimental set-up consists of three main items:

- A *reference simulation* (`REF`) is performed to generate "simulated observations" and to define the truth that the assimilation runs should reach. This corresponds to a standard integration of the ISBA scheme that needs to be initialized for its prognostic variables and forced by prescribed atmospheric quantities (wind, temperature, humidity, radiation fluxes, precipitation fluxes). The land surface characteristics for the soil and the vegetation have also to be specified.
- An *open loop run* (`OL`) : this run is similar to the `REF` run except that the initial conditons in the soil are modified and the forcing is perturbed. The response of the land surface scheme is expected to be significantly different from the `REF` integration.
- *Assimilation runs* where observations are taken from the `REF` run whereas the initial conditions and the perturbed forcing correspond to the `OL` run. If the SLDAS works as expected it should progressively reach a state close to the `REF` run.

For the experiments proposed in this package, the atmospheric forcing has been generated over a two-month period in July-August 2002 over Central US (adapted from ERA40). The surface specifications (soil/vegetation) are given in Mahfouf (2007).

3. Practical aspects

The main directory is `LDAS_ISBA` that contains six sub-directories :

- `data_in`: contains the input data for the atmospheric forcing (already available for you) and the simulated observations (to be generated by you as explained in section 4).
- `data_out`: contains the ouput files produced by the various runs
- `proc`: contains the scripts to run the ISBA scheme and the assimilation experiments. It also contains scripts to generate graphics with the GNUPLOT software (check that it is available on your PC by typing the command :`gnuplot`). It contains a `namelist` to define the most important features of each run.
- `src1`: contains the sources, the executables and a `Makefile`. Experiments to be run from this directory correspond to screen-level observations over a 6 hour assimilation window (similar to operational configurations)
- `src2`: contains the sources, the executables and a `Makefile`. Experiments to be run from this directory correspond to assimilations of both screen-level observations and L-band brightness temperatures with temporal frequencies of 6-h and 3 days respectively.
- `graphs`: contains postscript files generated by GNUPLOT.

4. Practical aspects

a. *How to generate the executable ?*

In the `src1` and `src2` directories, edit the `Makefile` and change the compilation options if needed. Then, type the command `make`. It will generate an executable `main-ref` that allows to run the various experiments suggested in the practical exercises.

b. *How to launch the model and assimilation runs ?*

i. Set-up the namelist for the reference run Edit the file `proc/namelist`.

`&ASSIM` : In this block, set the following logicals to `.FALSE.` to indicate that at this stage you don't want to perform a data assimilation experiment :

```
L_OI = .FALSE.

L_EC = .FALSE.

L_2DVAR = .FALSE.

L_EKF = .FALSE.

L_ENKF = .FALSE.
```

The other logicals in this block are not used.

&SOILINIT : In this block, define the initial soil conditions of the reference run for the two water reservoirs SWI1 (w_g) and SWI2 (w_2)[1] and the two soil temperatures TG1 (T_s) and TG2 (T_2). These are the four elements of the control variable $\mathbf{x}$. We suggest a value of 4 for the soil reservoirs (high value to produce saturated soils) and of 295 K for the soil temperatures (see Mahfouf, 2007):

```
SWI1 = 4.0

SWI2 = 4.0

TG1 = 295.

TG2 = 295.
```

&PERTRAIN : In this block set the value of SCALE_RAIN to one (it means that the precipitation forcing is not modified).

ii. Perform a reference run: Under the sub-directory proc, type the following command to create the reference simulation :

```
submit.bat REF 2c
```

The experiment type is REF and the experiment identifier 2c (imposed for the REF and OL experiments for reasons explained in the footnote [2]). For all assimilation experiments there is no constraint on what the experiment identifier should be (it could be useful to have a log file that makes the correspondence with the experimental set-up)

iii. Perform an open loop run: This run has the same namelist options as the reference run except for different initial soil moisture conditions and a modified precipitation forcing. We suggest the following values in the namelist file: SWI1 = 0, SWI2 = 0 and SCALE_RAIN = 0.5. The soil moisture content is at the wilting point (dry soils) and the precipitation forcing is reduced by a factor of two. Then, type the following command:

```
submit.bat OL 2c
```

iv. Do it again ! The above steps allows to perform the reference and open loop runs that will be used for the assimilation of screen-level observations every 6 hours (in the src1 sub-directory). For the same runs corresponding to the assimilation of L-band brightness temperature experiments and screen-level parameters (in the src2 sub-directory), you should modify the namelist file as suggested before but type the following execution commands:

```
submit.bat REF 2f TB

submit.bat OL 2f TB
```

[1] defined in terms of *soil wetness index* $SWI = (w - w_{wilt})/(w_{fc} - w_{wilt})$ that is the "equivalent" of relative humidity in the atmosphere, expect that negative values and values above one can exist.

[2] We force the user to give 2c for REF and OL experiments in src1 and 2f for REF and OL experiments in src2, to avoid code modifications for the simulated observation file name as well as in plotting script.

The additional key TB allows to select the executable file in the src2 sub-directory.

These two types of simulations are basically done once and they are used to compare the behaviour of the various assimilations that you will run.

c. *What are the various assimilations that can be run ?*

i. Assimilation of screen-level observations - directory src1 The possible choicees are given in the &ASSIM block of the namelist. When observations are screen-level temperature and relative humidty and a 6-hour assimilation cycle (using executable in src1). As before, you have to modify the namelist file in order to define the various set-ups of for each experiment (given below). Then you have to type the following command :

```
submit.bat exptype expid
```

The exptype is given below (not an user option) whereas the expid can be chosen at your convenience.

OPTIMAL INTERPOLATION USING MÉTÉO-FRANCE FORMULATION (Giard and Bazile, 2000): L_OI = .TRUE. and all other logical set to .FALSE.
The exptype is OI_MF

OPTIMAL INTERPOLATION USING ECMWF FORMULATION (Douville et al., 2000): L_OI = .TRUE. and L_EC = .TRUE. with all other logical set to .FALSE.
The exptype is OI_EC

SIMPLIFIED 2D VARIATIONAL ASSIMILATION (Balsamo et al. 2004): L_2DVAR = .TRUE. and all the other logical are set to .FALSE.
The exptype is 2DVAR

SIMPLIFIED EXTENDED KALMAN FILTER (Seuffert et al., 2004) L_EKF = .TRUE. and all the other logical are set to .FALSE.. The simplication lies in the fact that the $\mathbf{B}$ matrix remains constant at each cycle.
The exptype is EKF

ENSEMBLE KALMAN FILTER (Lorenc, 2003; Evensen, 2002): L_ENKF = .TRUE.. All other logical are set to .FALSE. except the value L_NOISE that can be set to .TRUE. in order to include noise in the atmospheric forcing (but not recommended as the default, since results are not particularly good).
The exptype is ENKF

ii. Assimilation of L-band brightness temperatures and screen-level observations - directory src2 When observations are 2m temperature and humidity as well as L-band brightness temperatures (using executables in src2), two additional logicals have to be defined. They indicate which type of observations has to be considered : L_WG = .TRUE. for the assimilation of brightness temperatures (every 3 days) and L_2M = .TRUE. for the assimilation of screen-level observations (every 6 hours). As before you have to modify the namelist file in order to define the various set-ups of each experiment (given below). Then, you have to type the following command :

```
submit.bat exptype expid TB
```

The exptype is given below (not an user option) whereas the expid has to chosen at your convenience.

SIMPLIFIED 2D VARIATIONAL ASSIMILATION (Balsamo et al. 2004): L_2DVAR = .TRUE. the other logical are set to .FALSE.. The Jacobian of the observation operator is evolved over a 3-day period to allow the assimilation to produce analyses every 6 hours instead of every 3 days (see Mahfouf (2007) for details).
The exptype is 2DVAR

EXTENDED KALMAN FILTER (Seuffert et al., 2004) L_EKF = .TRUE. and the other logical are set to .FALSE.. The the $\mathbf{B}$ matrix evolves according the Kalman filter equations but is set back to the diagonal matrix every 3-days (to account for the fact that the specification of the $\mathbf{Q}$ matrix is rather arbitrary).
The exptype is EKF

ENSEMBLE KALMAN FILTER (Lorenc, 2003; Evensen, 2002): L_ENKF = .TRUE.. All other logical are set to .FALSE. except L_NOISE that can be set to .TRUE. in order to include noise in the atmospheric forcing (but not recommended as the default, since results are not particularly good).
The exptype is ENKF

5. Tunable parameters of the assimilations

All these parameters have to be defined in the namelist

&OBSERR : The observation errors (standard deviations defining the $\mathbf{R}$ matrix) for each element of the observation vector $\mathbf{y}_o$

&SIZEJAC : The size of the perturbation of the control vector to get the Jacobians in finite differences

&BKGERR : The background errors (standard deviations defining a diagonal $\mathbf{B}$ matrix) for each element of the control vector $\mathbf{x}$

&BKGERR : The model errors (standard deviations defining a diagonal $\mathbf{Q}$ matrix) for each element of the control vector $\mathbf{x}$ (same units as $\mathbf{B}$). This specification is only required for the EKF option.

&SETENKF: The ensemble size NDIM and the inflation factor XINFL (every 6 hours) for the Ensemble Kalman filter (ENKF)

6. Graphics

By typing the command:

```
proc/plot_model_variables.bat exptype expid
```

a postscript figure is generated :

```
graphs/FIGURE_exptypeexpid.ps
```

In the case of the ENKF the spread of the ensemble (diagonal elements of $\mathbf{B}$) can be plotted by the command:

```
proc/plot_enkf.bat exptype expid
```

The following postcript file is generated:

```
graphs/FC_ERRORS_ENKF_expid.ps
```

You should be able to recover most of the figures presented in Mahfouf (2007) and examine the main features of each assimilation system.

7. The subroutines

a. *ISBA*

The ISBA scheme is considered as a black box that evolves (over a specified time window) the model state $\mathbf{x}$ from an initial state `XI` to a final state `XF`, and produces the simulated observations $\mathbf{y}$ (`YF`).

- `isba.f90` : main driver of the ISBA scheme that performs initialisations (except the atmospheric forcing), runs the scheme and post-process the ouputs. The projection of the model state in the observation space is done in this subroutine (i.e. that defines the observation operators).
- `drag_coeff_z0h.f90`: computation of the surface aerodynamic resistance using Louis et al. (1982) formulation (stability function depending on the Richarsdon number), and generalized by Mascart et al. (1995) to the situation of different roughness lengths for heat and momentum.
- `energy_budget.f90`: resolution of the prognostic equations for T_s and T_2 (implicit resolution for T_s by a linearization of the various terms of the surface energy balance).
- `fluxes.f90`: diagnostic of the fluxes entering the surface energy balance
- `init_surf.f90`: definition of the initial conditions characterizing the soil and the vegetation (except for the soil texture that is defined in `isba.f90`).
- `interpol_forcing.f90`: temporal interpolation of the forcing including an option for noise generation in the EnKF
- `rs_soil.f90`: computation of the soil resistance for bare soil evaporation (ECMWF formulation)
- `rs_veg.f90`: computation of the canopy resistance (important dependency with w_2)
- `soil_prop.f90`: computation of the thermal and hydraulic coefficients of the force-restore scheme (see Appendix A of Noilhan and Mahfouf, 1996) (C_1, C_2, C_G and w_{geq}).
- `soil_def.f90`: computation of the soil constants depending on soil texture only
- `water_budget.f90`: resolution of the prognostic equations for w_g and w_2 A red noise is added in the equations (parameters defined in this routine) when the EnKF is used (to describe model errors as proposed by Evensen (2002)).

b. *Inputs/ouputs*

- `read_forcing.f90` : reads the forcing data set (the file name to be read is explicitely given in this routine)
- `print_output_enkf.f90` : computation of means and standard deviations from the EnKF outputs for comparison with deterministic assimilations
- `print_output_oi.f90`: computation of accumulated fluxes and store the various diagnostic and prognostic variables produced by ISBA in files (see Section 8 for file naming and content)
- `clean_exit.f90` : deallocation of arrays

c. *Analyses*

- `master.f90` : main driver program that selects the proper simulation according the logicals defined in `&ASSIM` of the `namelist`. Does nothing if inconsistencies are identified.
- `main.f90` : driver to produce the reference (`REF`) and open loop (`OL`) runs of ISBA
- `main_oi.f90`: driver to produce analyses from optimum interpolations, simplified 2D-Var, and Extended Kalman Filter
- `main_enkf.f90`: driver to produce analyses from the Ensemble Kalman filter
- `oi_coeffs_ec.f90`: computation of the optimum interpolation coefficients to analyse soil variables from T_{2m} and HU_{2m} using the ECMWF formulation (Douville et al., 2000).
- `oi_coeffs_mf.f90`: computation of the optimum interpolation coefficients to analyse soil variables from T_{2m} and HU_{2m} using the Météo-France formulation (Giard and Bazile, 2000).

d. *Tools : algebra and others*

`chlodc.f90` : Cholesky factorization (part I) for inversion of a symmetric matrix

`chlosl.f90` : Cholesky factorization (part II) for inversion of a symmetric matrix

`inverse_matrix.f90`: produce the inverse matrix that has been implicitely obtained in the Cholesky decomposition.

`gasdev.f90`: Generation of a random number following a Gaussian distribution of zero mean and variance one.

`out_product.f90`: Get a matrix $\mathbf{x}\mathbf{y}^T$ formed by the outer product of two vectors $\mathbf{x}$ and $\mathbf{y}$ (used in the EnKF approach)

`solar_angle.f90`: Computation of the solar zenith angle knowing the location in space and time of a given point.

`thermo_functions.f90`: Computation of the saturation water vapour pressure, specific humidity at saturation and their derivatives with respect to temperature.

e. *Observation operators*

- `lsmem.f90`: simple microwave radiative transfer model to simulate L-band brightness temperatures from the superficial moisture content.
- `cls_interpol.f90`: vertical interpolation at observation level (2m and 10m) using Monin-Obukhov similarity theory and Businger-Dyer stability functions.
- `vdfppcfls.f90`: vertical interpolation at observation level (2m and 10m) using Monin-Obukhov similarity theory and Geleyn (1988) formulation.

8. The main output files

In the directory `data_out` the following files are created for each assimilation run:

`FIC22_exptypeexpid.dat` (simulated observations) : day , T_{2m} , HU_{2m}

`FIC23_exptypeexpid.dat` (prognostic variables) : day , T_s , T_2 , w_g , w_2

`FIC24_exptypeexpid.dat` (instantaneous surface energy fluxes) : day , R_n , H , LE, G

`FIC25_exptypeexpid.dat` (accumulated water fluxes) : day , ΣLE, $\Sigma Precip$, $\Sigma Runoff$

`FIC26_exptypeexpid.dat` (accumulated energy fluxes) : day , ΣR_n, ΣLE , ΣH , ΣG

`FIC27_exptypeexpid.dat` (standard-deviations from the ensemble mean : EnKF only) : day , σ_{wg}, σ_{w2} , σ_{Ts} , σ_{T2} , σ_{LE} , σ_H

`FIC55_exptypeexpid.dat` (Jacobians of observation operator) : day , $\partial T_{2m}/\partial w_g$, $\partial T_{2m}/\partial w_2$, $\partial T_{2m}/\partial T_s$, $\partial T_{2m}/\partial T_2$

`FIC56_exptypeexpid.dat` (Jacobians of observation operator) : day , $\partial HU_{2m}/\partial w_g$, $\partial HU_{2m}/\partial w_2$, $\partial HU_{2m}/\partial T_s$, $\partial HU_{2m}/\partial T_2$

9. The ISBA land surface scheme

The ISBA scheme evolves four prognostic variables:

$$\frac{\partial T_s}{\partial t} = C_T(R_n - H - LE) - \frac{2\pi}{\tau}(T_s - T_2) \quad (1)$$

$$\frac{\partial T_2}{\partial t} = \frac{1}{\tau}(T_s - T_2) \quad (2)$$

$$\frac{\partial w_g}{\partial t} = \frac{C_1}{\rho_w d_1}(P_g - E_g) - \frac{C_2}{\tau}(w_g - w_{geq}) \quad (3)$$

$$\frac{\partial w_2}{\partial t} = \frac{1}{\rho_w d_1}(P_g - E_g - E_{tr}) - \frac{C_3}{\tau}\max\left[0., (w_2 - w_{fc})\right] \quad (4)$$

10. The main analysis equations

a. *The Extended Kalman filter (*`EKF`*)*

We consider a control vector $\mathbf{x}$ (dimension N_x) that represents the prognostic equations of the land surface scheme ISBA $\mathcal{M}$ (`isba.f90`) that evolves with time as:

$$\mathbf{x}^t = \mathcal{M}(\mathbf{x}^0)$$

Therefore $N_x = 4$ and $\mathbf{x} = (w_g, w_2, T_s, T_2)$
At a given time t a vector of observation is available $\mathbf{y}_o$ (with a dimension N_y) characterized by an error covariance matrix $\mathbf{R}$. An observation operator $\mathcal{H}$ allows to get the model counterpart of the observations :

$$\mathbf{y}^t = \mathcal{H}(\mathbf{x}^t)$$

In our case, $\mathcal{H}$ will be a vertical interpolation scheme for T_{2m} and HU_{2m} (`vdfppcfls.f90`) and microwave radiative transfer model for brightness temperatures (2 polarizations) T_{bV} and T_{bH} (`lsmem.f90`). The dimension of the observation vector is $N_y = 4$. The forecast $\mathbf{x}$ at time t (written $\mathbf{x_f}^t$) is characterized by an background error covariance matrix $\mathbf{B}$. The implicit hypotheses for the errors is that they are gaussian and unbiased.

A new value of $\mathbf{x}$ written $\mathbf{x_a}^t$ (the analysis), obtained by an optimal combination the observations and the background (short-range forecast), is given by :

$$\mathbf{x_a}^t = \mathbf{x_f}^t + \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1}(\mathbf{y}_o^t - \mathcal{H}(\mathbf{x_f}^t))$$

Since the observation operator can be non-linear, a new operator appears in this analysis equation : $\mathbf{H}$ (together with its transpose $\mathbf{H}^T$). It corresponds to the Jacobian matrix of $\mathcal{H}$ defined as :

$$\mathbf{H}_{ij} = \frac{\partial \mathbf{y}_j}{\partial \mathbf{x}_i}$$

This matrix has N_x raws and N_y columns. In SLDAS we use a finite difference approach where the input vector $\mathbf{x}$ is perturbed N_x times to get for each integration a column of the matrix $\mathbf{H}$, that is :

$$\mathbf{H}_{ij} \simeq \frac{\mathbf{y}_i(\mathbf{x} + \delta x_j) - \mathbf{y}_i(\mathbf{x})}{\delta x_j}$$

where δx_j is a small increment value added to the j-th component of the $\mathbf{x}$ vector (defined in the block `&SIZEJAC` of the `namelist` file by the values `EPS_W1` (for w_g), `EPS_W2` (for w_2), `EPS_T1` (for T_s), `EPS_T2` (for T_2)). This is an interesting alternative to the adjoint coding when the size of the control vector is not too large.
The analysis state in characterized by an analysis error covariance matrix:

$$\mathbf{A} = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{B}$$

where $\mathbf{K}$ is the gain matrix defined in the analysis equation by:

$$\mathbf{K} = \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1}$$

it relates linearly the *analysis increments*

$$[\mathbf{x_a} - \mathbf{x_f}]$$

to the *innovation vector* :

$$[\mathbf{y}_o - \mathcal{H}(\mathbf{x_f})]$$

The analysis is cycled by propagating the time the two quantities $\mathbf{x_a}$ et $\mathbf{A}$ up to next time where observations are available :

$$\mathbf{x_f}^{t+1} = \mathcal{M}(\mathbf{x_a}^t)$$
$$\mathbf{B}^{t+1} = \mathbf{M}\mathbf{A}^t\mathbf{M}^T + \mathbf{Q}$$

This equation requires the Jacobian matrix $\mathbf{M}$ of the model $\mathcal{M}$, that is defined as (between time t and time t_0):

$$\mathbf{M}_{ij} = \frac{\partial x_i^t}{\partial x_j^0}$$

A new matrix $\mathbf{Q}$ representing the model error covariance matrix needs to be defined. This sequential approach is the *Extended Kalman filter* as coded in the `src2` directory.

b. *Variational assimilation*

It is possible to show that when $\mathcal{H}$ is linear, the EKF solution is identical to the one provided by a variational approach where a state $\mathbf{x}$ minimizes a cost-function measuring the departure between a bacground information (short-range forecast) $\mathbf{x_b}$ and available observations $\mathbf{y}_o$:

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x_b})^T\mathbf{B}^{-1}(\mathbf{x} - \mathbf{x_b}) + \frac{1}{2}(\mathbf{y}_o - \mathcal{H}(\mathbf{x}))^T\mathbf{R}^{-1}(\mathbf{y}_o - \mathcal{H}(\mathbf{x}))$$

The minimum of J is found by computing its gradient that is given to a descent algorithm :

$$\nabla J(\mathbf{x}) = \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x_b}) + \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}_o - \mathcal{H}(\mathbf{x}))$$

where appears the transpose (adjoint) of the observator operator $\mathbf{H}^T$

The above cost function does not include the time dimension explicitely. It can be applied at a given time like a Kalman filter (3D-Var), but also over a temporal interval : a state $\mathbf{x}^0$ at the beginning of the assimilation window is searched so that it leads to a model state $\mathbf{x}^t$ fitting all the available observations at several times over the period (4D-Var). In that case, the observation operator is the combination of a first operator (forecast model) evolving the initial state $\mathbf{x}^0$ up to a time t where a vector of observations is available $\mathbf{y}_o^t$:

$$\mathbf{x}^t = \mathcal{M}(\mathbf{x}^0)$$

and a "true" observation operator (as defined in the EKF) that gives the model equivalent of the observation at the same time :

$$\mathbf{y}^t = \mathcal{H}(\mathbf{x}^t)$$

Finally, the relation between the variable to be analyzed and the available observation is :

$$\mathbf{y}^t = \mathcal{H}_1(\mathbf{x}^0)$$

with $\mathcal{H}_1 = \mathcal{H}\mathcal{M}$. This approach allows modification of the $\mathbf{B}$ matrix over the assimilation window by the linearized versions of the forecast model: at a given time t, the effective $\mathbf{B}$ matrix is $\mathbf{M}\mathbf{B}\mathbf{M}^T$ (which is similar to the propagation of the $\mathbf{A}$ matrix in the Kalman filter without the error model term and without explicit matrix evolution).

The variational method is rather expensive since the linearized operator $\mathbf{H}_1$ is needed at each iteration of a minimization algorithm.

c. *Simplified variational assimilation (*`2DVAR`*)*

The cost of the variational assimilation can be significantly reduced (i.e. no minimization), if the observation operator can be linearized as:

$$\mathcal{H}(\mathbf{x}^t) = \mathcal{H}\left[\mathcal{M}(\mathbf{x_b}^0)\right] + \mathbf{HM}(\mathbf{x}^0 - \mathbf{x_b}^0)$$

The analysis state at the beginning of the assimilation window can be written as:

$$\mathbf{x_a}^0 = \mathbf{x_f}^0 + \mathbf{BH}^T\mathbf{M}^T(\mathbf{HMBM}^T\mathbf{H}^T + \mathbf{R})^{-1}(\mathbf{y}_o^t - \mathcal{H}\left[\mathcal{M}(\mathbf{x_f}^0)\right])$$

This state in then progated by the model $\mathcal{M}$ to define the initial state of the next cycle. In the case of a linear model this equation is equivalent to an Extended Kalman Filter but where the model error term $\mathbf{Q}$ is set to zero (perfect model assumption). This approach is cheaper than a real variational assimilation and takes into account simultaneous observations at various times (implicit evolution of the $\mathbf{B}$ matrix over the assimilation window). The length of the window depends upon the linearity of the combined operator : $\mathcal{HM}$. In a EKF (or EnKF) approach the time window of the assimilation corresponds to the interval between the most frequent observations.

d. *The ensemble Kalman filter (*`ENKF`*)*

This approach has been developed widely during the last ten years as it allows to by-pass the problem of the linearized observation operators (Jacobians) $\mathbf{H}$, $\mathbf{M}$, $\mathbf{H}^T$, and $\mathbf{M}^T$ as well as the specification of the $\mathbf{B}$ matrix. It is rather popular in hydrology (Crow et al., 2001; Reichle et al., 2002). The same equation as for the EKF is solved, but the matrices $\mathbf{BH}^T$ and $\mathbf{HBH}^T$ are obtained from an ensemble of predictions $(\mathbf{x_f})_i$ with i varying between 1 et N. The value of N has been chosen to 100 in LSDAS (`NDIM`). One writes :

$$\mathbf{BH}^T \approx \overline{(\mathbf{x_f} - \overline{\mathbf{x_f}})(\mathcal{H}(\mathbf{x_f}) - \overline{\mathcal{H}(\mathbf{x_f})})^T}$$

$$\mathbf{HBH}^T \approx \overline{(\mathcal{H}(\mathbf{x_f}) - \overline{\mathcal{H}(\mathbf{x_f})})(\mathcal{H}(\mathbf{x_f}) - \overline{\mathcal{H}(\mathbf{x_f})})^T}$$

with the average operators :

$$\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

$$\overline{\mathbf{xy}^T} = \frac{1}{N-1}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{y}_i^T$$

An advantage of this approach is that there is no need to get explicily the various matrices entering in the computation of the Kalman gain $\mathbf{K}$. The spread of the ensemble is obtained by perturbing randomly the observations according to the $\mathbf{R}$ matrix. Small values of N for sampling the error statistics lead to an underestimation of the analysis error covariance matrix, that leads to a divergence of the filter (the background is becoming more and more accurate and observations are becoming useless). To remedy this problem, the spread of the ensemble can be artificially enhanced at each analysis cycle (inflation factor) and by introducing model errors in the prognostic equations (that are however rather difficult to tune).

In practice, each analysis $\mathbf{x}_a^i$ is given by the filter from the following equation:

$$\mathbf{x}_a^i = \mathbf{x}_b^i + \mathbf{K}[\mathbf{y}_0 - \mathcal{H}(\mathbf{x}_b^i) + \varepsilon_i]$$

where ε is a random noise with gaussian distribution having a zero mean and a variance given by $\mathbf{R}$. The gain matrix is computed using the approximated values of $\mathbf{BH}^T$ and $\mathbf{HBH}^T$ given by the above equations.

The prognostic variables for soil water contents w have been modified to add a model error term φ defined by :

$$\varphi^{t+1} = \nu\varphi^t + \varepsilon_w\sqrt{1-\nu^2}$$
$$w^{t+1} = \mathcal{F}(w^t) + \varphi^{t+1}\Delta t$$

with $\nu = 1/(1+\Delta t/\tau)$. It is a first order autoregressive model with temporal correlation τ of 3 days and a gaussian noise amplitude ε_w set to à 0.001 m^3/m^3/day. These quantities are hardcoded in the subroutine `water_budget.f90` and could be modified for sensitivity studies.

In order to avoid an underestimation of the analysis error covariance matrix $\mathbf{A}$, after each analysis cycle an inflation factor α (`XINFL` in `namelist`) leads to a redefinition of the various analysed members as:

$$\mathbf{x}_a = \overline{\mathbf{x}_a} + \alpha(\mathbf{x}_a - \overline{\mathbf{x}_a})$$

In practice a value of 1.015 has been chosen for a 6 hour cycling.

11. Namelist default values

`&ASSIM` [options for experiment type]
`L_OI = .FALSE.`
`L_EC = .FALSE.`
`L_2DVAR = .FALSE.`
`L_EKF = .FALSE.`
`L_ENKF = .FALSE.`
`L_NOISE = .FALSE.` [option to add noise in the forcing - works only with ENKF]
L_WG = .TRUE.[option when assimilation of combined obs types]
L_2M = .TRUE.[option when assimilation of combined obs types]
/
&SETENKF [options for ensemble kalman filter]
NDIM = 100 [ensemble size N]
XINFL = 1.015 [inflation factor α]
/
&SOILINIT [options for initialisation of soil prognostic variables]
SWI1 = 4.0 [soil wetness index for w_g]
SWI2 = 4.0 [soil wetness index for w_2]
TG1 = 295. [surface temperature T_s]
TG2 = 295. [deep temperature T_2]
/
&PERTRAIN [option for rescaling the precipitation forcing]
SCALE_RAIN = 1.0
/
&SIZEJAC [option to define the perturbation size for the Jacobian estimation]
EPS_W1 = 0.0001 [perturbation for w_g in soil wetness index (SWI)]
EPS_W2 = 0.0001 [perturbation for w_2 in soil wetness index (SWI)]
EPS_T1 = 0.001 [perturbation for T_s in K]
EPS_T2 = 0.001 [perturbation for T_2 in K]
/

&OBSERR [option to define the observation errors]
ER_T2M = 1.0 [standard deviation error for 2-m temperature (σ_{T2m}) in K]
ER_HU2M = 0.1 [standard deviation error for 2-m relative humidity (σ_{HU2m}) no units]
ER_TB = 2.0 [standard deviation error for L-band brightness temperature (σ_{Tb}) in K]
ER_WG = 0.1 [standard deviation error for surface soil moisture (σ_{w_g}) in SWI (not used)]
/
&BKGERR [option to define the background errors]
ER_W1 = 0.1 [standard deviation error for surface soil moisture (σ_{w_g}) in SWI]
ER_W2 = 0.1 [standard deviation error for deep soil moisture (σ_{w_2}) in SWI]
ER_T1 = 1.0 [standard deviation error for surface temperature (σ_{T_s}) in K]
ER_T2 = 1.0 [standard deviation error for deep soil temperature (σ_{T_s}) in K]
/
&MODERR [option to define the model errors]
Q_W1 = 0.02 [standard deviation error for surface soil moisture (σ_{w_g}) in SWI]
Q_W2 = 0.02 [standard deviation error for deep soil moisture (σ_{w_2}) in SWI]
Q_T1 = 0.5 [standard deviation error for surface temperature (σ_{T_s}) in K]
Q_T2 = 0.5 [standard deviation error for deep soil temperature (σ_{T_s}) in K]
/

12. Summary of experimental set-ups

Experiment	exptype	L_OI	L_EC	L_EKF	L_2DVAR	L_ENKF
Reference run	REF	F	F	F	F	F
Open loop run	OL	F	F	F	F	F
OI Météo-France	OI_MF	T	F	F	F	F
OI ECMWF	OI_EC	T	F	F	F	F
Extended Kalman filter	EKF	F	F	T	F	F
Simplified 2D-Var	2DVAR	F	F	F	T	F
Ensemble Kalman filter	ENKF	F	F	F	F	T

Table 1: Summary of possible experiments

13. Exercises

14. References

Balsamo, G., F. Bouyssel, and. J. Noilhan, 2004: A simplified bi-dimensional variational analysis of soil moisture from screen-level observations in a mesoscale numerical weather prediction model. *Quart. J. Roy. Meteor. Soc.*, **130**, 895-915.

Balsamo, G., J.-F. Mahfouf, S. Bélair, and G. Deblonde, 2007: A land data assimilation system for soil moisture and temperature : an information content study. *J. Hydrometeor.* (accepté pour publication)

Bouttier, F., J.-F. Mahfouf, and J. Noilhan, 1993: Sequential assimilation of soil moisture from atmospheric low-level parameters. Part I: Sensitivity and calibration studies. *J. Appl. Meteor.*, **32**, 1335-1351.

Douville, H., P. Viterbo, J.-F. Mahfouf, and A.C.M. Beljaars, 2000: Evaluation of optimal interpolation and nudging techniques for soil moisture analysis using FIFE data. *Mon. Wea. Rev.*, **128**, 1733-1756.

Evensen, G., 2003 : The ensemble Kalman Filter: Theoretical formulation and practical implementation. *Ocean Dyn.*, **53**, 343-367.

Geleyn, J.-F., 1988: Interpolation of wind, temperature, humidity values from model levels to the height of measurement. *Tellus*, **40A**, 347-351.

Giard, D., and E. Bazile, 2000: Implementation of a new assimilation scheme for soil and surface variables in a global NWP model. *Mon. Wea. Rev.*, **128**, 997-1015.

Hess, R., 2001: Assimilation of screen-level observations by variational soil moisture analysis. *Meteor. Atmos. Phy.*, **77**, 145-154.

Houtekamer, P. and H.L. Mitchell, 1998: Data assimilation using an Ensemble Kalman filter technique. *Mon. Wea. Rev.*, **126**, 796-811.

Mahfouf, J.-F., 1991: Analysis of soil moisture from near-surface parameters: A feasibility study. *J. Appl. Meteor.*, **30**, 506-526.

Mahfouf, J.-F., and J. Noilhan, 1991: Comparative study of various formulations of evaporation from bare soil using in-situ data. *J. Appl. Meteor.*, **30**, 1354-1365.

Mahfouf, 2007: The Météo-France soil analysis. Part I: Evaluation and perspectives at local scale (in French). Note de Centre CNRM/GMME No 84. Available at *https://hal.science/hal-04390802*

Mascart, P., J. Noilhan, and H. Giordani, 1995: A modified parameterization of the surface layer flux-profile relationships using different roughness length values for heat and momentum. *Bound. Layer. Meteor.*, **72**, 331-344.

Muñoz-Sabater, J., L. Jarlan, J.-C. Calvet, F. Bouyssel, and P. De Rosnay, 2007: From near-surface to root-zone soil moisture using different assimilation techniques. *J. Hydrometeor.*, **8**, 194-206.

Noilhan, J. , and J.-F. Mahfouf, 1996: The ISBA land surface parameterization scheme. *Global Planet. Change*, **13**, 145-159.

Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, 1992: Numerical recipes in FORTRAN. The art of scientific computing. Cambridge University Press, 963 pp.

Seuffert, G., H. Wilker, P. Viterbo, M. Drusch, and J.-F. Mahfouf, 2004: The usage of screen-level parameters and microwave brightness temperature for soil moisture analysis. *J. Hydrometeor.*, **5**, 516-531.