



**HAL**  
open science

# Ensemble Integrations of Telemac-Mascaret for the Optimal Model Calibration

Fabrice Zaoui, Cédric Goeury, Yoann Audouin

► **To cite this version:**

Fabrice Zaoui, Cédric Goeury, Yoann Audouin. Ensemble Integrations of Telemac-Mascaret for the Optimal Model Calibration. XXVth Telemac & Mascaret User Club, Oct 2018, Norwich, United Kingdom. hal-01908756

**HAL Id: hal-01908756**

**<https://hal.science/hal-01908756>**

Submitted on 30 Oct 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Ensemble Integrations of Telemac-Mascaret for the Optimal Model Calibration

Fabrice Zaoui<sup>1</sup>, Cédric Goeury<sup>1</sup>, Yoann Audouin<sup>1</sup>

<sup>1</sup> EDF R&D – National Laboratory for Hydraulics and Environment (LNHE)

6 quai Watier, 78401 Chatou, France

E-mail: [fabrice.zaoui@edf.fr](mailto:fabrice.zaoui@edf.fr)

**Abstract**— This paper is focused on the use of a filtering method with Telemac-Mascaret in order to calibrate the most influent modelling parameters to their optimal values when the observation data is acquired. Optimality remains an objective that tends to reduce uncertainty a posteriori by finding trade-offs between model results and observations. This can be done by explicitly considering the uncertainties associated with the modelling process and the measurement chain. Two main families of mathematical methods are developed by the community of Data Assimilation to achieve this task. The first is a variational approach based on the mathematical minimization of a two parts error function. The second is a filtering approach where only linear algebra formula are used. The paper presents two real test cases with the Ensemble Kalman Filtering (EnKF) method for the error reduction of parameter estimation of the Telemac-Mascaret system.

## I. INTRODUCTION

Data Assimilation (DA) is a family of mathematical methods for the state estimation of a physical dynamic system commonly used in geosciences [1][2]. Observations and prior information are optimally combined with the results of a numerical model in order to minimize the uncertainty on the state of a system. Different techniques can help to compensate for the errors. The two most popular are the variational approach and the filtering methods. In this paper, the DA-based estimation relies on the Ensemble Kalman Filter (EnKF) [3]. This filter is close to a Monte Carlo sampling method but it does not require in practice a large number of state vectors to define a converging ensemble. It will apply to find the best estimations for some physical parameters of Telemac-Mascaret during the calibration phase.

The application of EnKF to Telemac-Mascaret is not new, see for instance [4] for a Telemac 2D application in the late 2000s on twin experiments or more recently [5] for a Mascaret-based operational application. The objective of this paper is to show on two real test cases how the same EnKF algorithm [6] can successfully calibrate Telemac-Mascaret models with little implementation effort and a reasonable computational cost. After a short presentation of DA and EnKF, the paper introduces the Python tools [7] used for the implementation of this automatic filter-based calibration. These tools are open-source and easy to learn.

A Mascaret test case is firstly presented and calibrated on the value of the friction coefficients of the Rhône River in France. As this test has a low computational cost, it is used for a small study of sensitivity in order to better understand the convergence of the filter with respect to some of its parameters. A Telemac 2D maritime case is also filtered with

EnKF in order to calibrate some coefficients of a tidal wave. This case allowed us to compare the method with a variational approach and clearly demonstrates the interest of the filter as competitor for future real test cases using a complex representation.

## II. DATA ASSIMILATION

### A. Background

The aim of a Data Assimilation (DA) method is to find the best estimation of a state using two sources of information, model results and observational data. These information sources are supposed to be complementary and recorded over time. Both have errors that can be seen as random variables and modeled as probability density functions. Applying the Bayes' theorem:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \quad (1)$$

the estimation of an unknown state  $x$  is the distribution  $p(x|y)$  given a prior knowledge  $p(x)$  on the state, the observations  $p(y)$  not depending on  $x$  and the probability  $p(y|x)$  of having  $y$  if  $x$  was known, i.e. the likelihood. The uncertainty on  $x$  is reduced with rule (1) compared to the observations and a-priori estimation, see Fig. 1.

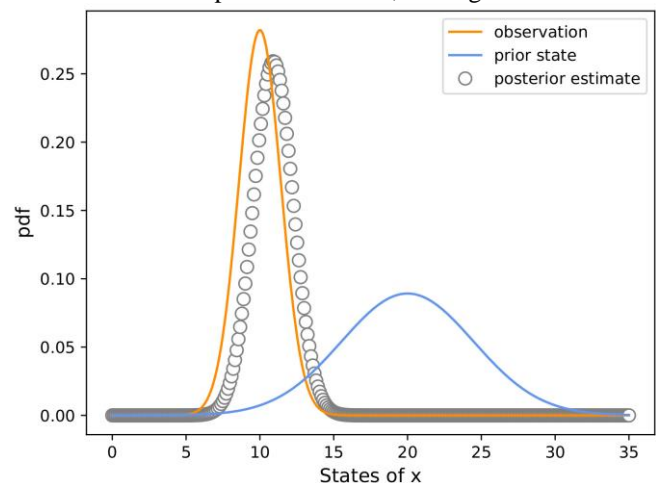


Figure 1. Coupling prediction and measurement

In most applications involving DA, a Gaussian distribution is used for the probability density functions of errors of each information source. This hypothesis, although it may be far from reality, simplifies the definition of uncertainties and opens many perspectives in the theory of estimation. A discrete state space model can be written as:

$$x_{k+1} = M_k(x_k) + w_k \quad w_k \sim \mathcal{N}(0, Q_k) \quad (2)$$

where  $x_{k+1}$  is the predicted system state at time step  $k + 1$ ,  $M_k$  is a linear or non-linear model and  $w_k$  a Gaussian noise representing the modelling errors with a covariance matrix  $Q_k$ . If  $x_k$  stands for the parameters to calibrate then it is convenient to take  $M_k = I$ . Similarly, the observations can be modelled as:

$$y_k = H_k(x_k) + v_k \quad v_k \sim \mathcal{N}(0, R_k) \quad (3)$$

where  $H_k$  is the observation model (shallow water) that relates the values of parameters  $x_k$  to the measurements  $y_k$  and  $v_k$  the observational noise with the covariance  $R_k$ .

DA aims to estimate  $\hat{x}$  as a solution of a mismatch problem to the observations. If the observations are simultaneously considered for a prescribed time window, the solution is a global model trajectory  $\hat{x}_{0:N} = (\hat{x}_0, \hat{x}_1, \dots, \hat{x}_N)^T$  where  $\hat{x}_k$  is for the time  $t_k \in [t_0, t_N]$  a local solution influenced by all the observations  $y_{0:N}$ . By supposing identical time steps for the model and the availability of observations, the DA problem in a variational approach can be written as the minimization of the error function  $J(\hat{x}_{0:N})$ :

$$J(\hat{x}_{0:N}) = \frac{1}{2} \|\hat{x}_0 - x_b\|_{B^{-1}}^2 + \frac{1}{2} \sum_{k=1}^N \|\hat{x}_k - M_{k-1}(\hat{x}_{k-1})\|_{Q_k^{-1}}^2 + \frac{1}{2} \sum_{k=0}^N \|y_k - H_k(\hat{x}_k)\|_{R_k^{-1}}^2 \quad (4)$$

The first two terms in (4) penalize the gaps to the prior (background) solution  $x_b$  at  $t_0$  and to modelling solutions from  $t_1$  to  $t_N$ . The third term is the penalization to observations. All terms are weighted by the error covariance matrices  $B$ ,  $Q$  and  $R$ .

### B. Ensemble Kalman Filtering

The Kalman filters are not global but sequential DA techniques where observations are gradually assimilated by evolving the uncertainty in the state from one observation time to the next. The original version of Kalman filter [8] assumes that operators  $M$  and  $H$  are linear. To work around this problem, the extended Kalman filter (EKF) has been developed but it still needs a linearization part that can pose computational cost and/or convergence problems (tangent linear and adjoint models required).

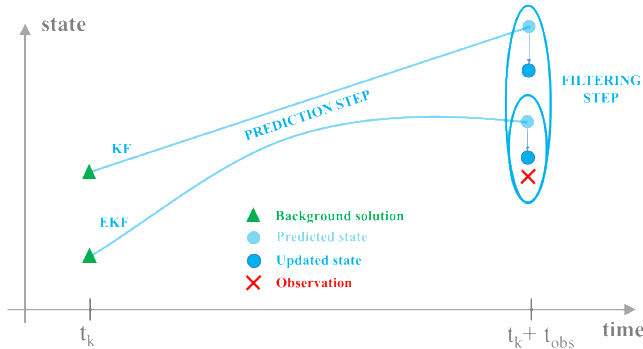


Figure 2. Sketch of KF and EKF algorithms

EnKF is an alternative method to estimate the covariance matrix using as a first approximation a sample of random

states (ensemble). Each state (member) evolves forward in time using  $M_k$  and the covariance is approximated with the new ensemble (prediction step). Then the ensemble is updated using perturbed observations (filtering step). In practice for most applications EnKF does not require a large number of members to converge.

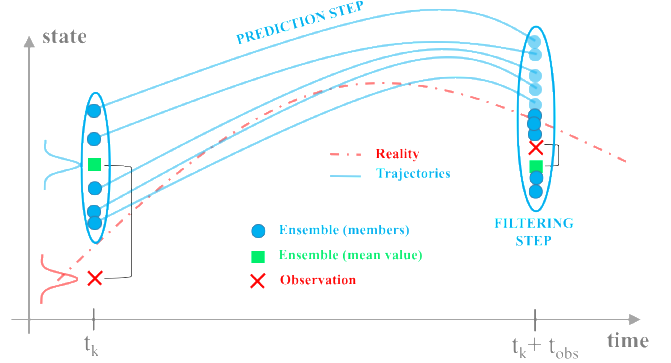


Figure 3. Sketch of EnKF algorithm

### Prediction step

- Propagation of each member  $i$  of ensemble  $N_e$  from time  $t_k$  to  $t_k + t_{obs}$  where  $t_{obs}$  is the period of observations:

$$x_{k+t_{obs}}^i = M_{k:k+t_{obs}}(x_k^i) + w_k^i \quad (5)$$

- Estimation of the covariance matrix:

$$P_{k+t_{obs}} = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (x_{k+t_{obs}}^i - \bar{x}_{k+t_{obs}})(x_{k+t_{obs}}^i - \bar{x}_{k+t_{obs}})^T$$

$$\text{with the mean ensemble: } \bar{x}_{k+t_{obs}} = \frac{1}{N_e} \sum_{i=1}^{N_e} x_{k+t_{obs}}^i \quad (6)$$

### Filtering step

- Computation of the Kalman gain:

$$K_{k+t_{obs}} = P_{k+t_{obs}} H^T (H P_{k+t_{obs}} H^T + O_{k+t_{obs}})^{-1} \quad (7)$$

where  $O_{k+t_{obs}}$  is the covariance computed from the perturbed observations.

- Update the ensemble:

$$x_{k+t_{obs}}^{i*} = x_{k+t_{obs}}^i + K_{k+t_{obs}}(y_{k+t_{obs}} + v_k^i - H x_{k+t_{obs}}^i) \quad (8)$$

## III. IMPLEMENTATION

### A. FilterPy

FilterPy is an open source (MIT License) Python module that implements some Kalman and Bayesian filters [9]. The module is rather oriented for engineering studies and offers documentation and basic examples for each filter. See [6] for the EnKF algorithm implemented in FilterPy as there exists many versions of this filter. In this program there is no particular improvement of the classic algorithm based on (5)-(8). Most advanced algorithms can implement for instance methods like localisation and/or inflation to make EnKF working in high dimension or being less cost computational. For the time being, these techniques have not been considered for solving data assimilation problems with the Telemac-Mascaret examples of this paper.

## B. TelApy

All the computations with the modules of the Telemac-Mascaret system can now be done directly in Python with TelApy [7]. TelApy is a package for using APIs of the system. It allows a fine interaction with the code while running. For instance it is possible to ask for values of variables or set some new values for parameters at any time of the simulation. The list of variable names and functions that can be used with TelApy are documented and tutorials exist in the source of the system [10].

TelApy is generic enough to not restrict the use of Telemac-Mascaret to some use cases or simulation platform. In particular it allows to easily couple Telemac-Mascaret with an optimizer or a filter like FilterPy for the DA.

## C. Coupling

The coupling between FilterPy and TelApy is quite easy. The user has only to write a supervisor script to firstly initialize the two modules and then call TelApy for  $t \in [t_k, t_k + t_{obs}]$  followed by a call to FilterPy.

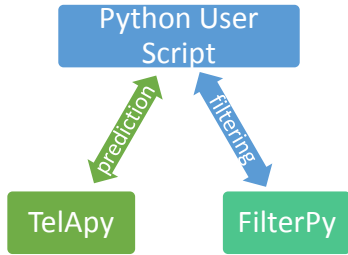


Figure 4. Python example for EnKF with Telemac-Mascaret

## Pseudo code

### algorithm telemac-enkf is

**input:** the size of the ensemble  $N_e$   
 the Telemac steering file  $cname$   
 initial values  $p_0$  for the parameters  
 model covariance matrix  $Q$   
 observation covariance matrix  $R$   
 observation operator  $H$   
**output:** ensemble of optimal Telemac states  $T_{opt}$   
 optimal values for parameters  $P_{opt}$

define an ensemble of parameters  $p_f$  from  $(p_0, N_e)$   
 define an ensemble of Telemac states  $S$  from  $(cname, p_f)$   
**while** the time  $t$  is **in** the assimilation window **do**  
 if observation is available:  
 $p_f \leftarrow \text{FilterPy}(S, p_f, H, R)$  # filtering  
**else:**  
 $S \leftarrow \text{TelApy}(S, p_f)$  # prediction  
 $p_f \leftarrow p_f + Q$   
 $t \leftarrow t + 1$

**return**  $(T_{opt} = S, P_{opt} = p_f)$

## D. Parallelism

Since each member  $i$  of  $N_e$  is an independent hydraulic state, the ensemble can be predicted in parallel. As Telemac is also a MPI-based parallel code, different configurations for parallelism are possible. TelApy requires the mpi4py module for solving the physics and one can add in the user script the support of the multiprocessing if needed for predictions. For the filtering equations (7)-(8), FilterPy is sequential. This is not an issue when  $N_e$  and  $dim(x)$  are low values.

## IV. RESULTS AND DISCUSSION

### A. Mascaret

EnKF is used to calibrate nine friction coefficients of the Rhône river in France on a reach of 22 km long. The geometry is not described as it is not the purpose of the present study. The mesh is made of 1,259 1D nodes. The boundary conditions are an imposed flow rate of 156 m<sup>3</sup>/s at the upstream and an imposed water level of 187.3 m at downstream. One observation of the water level per friction zone is available for this non-overtopping flow regime.

The background  $x_b$  is a vector of nine Strickler coefficients of 40 m<sup>1/3</sup>/s. The initial ensemble will be a normal distribution around this mean value. The initial error ( $L^2$ -norm) on the water levels computed by Mascaret and corresponding to  $x_b$  is equal to 0.6125 m.

As a reference for the appreciation of EnKF results, a variational approach has been tested for the minimization of (4) by an L-BFGS algorithm [11]. This has given a solution  $\hat{x}_{L-BFGS}$  corresponding to an error of 0.0894 m.

Many configurations with EnKF have been tested for different values of  $N_e$  repeated 100 times over 10 assimilation cycles (filtering). The choice for 100 repetitive runs of EnKF is done for trying to get some basic but confident statistics on the behaviour of the filter. The choice to stop EnKF after only 10 assimilation cycles is arbitrary but identical for all the values of  $N_e$ .

Fig. 5 shows the statistics of these errors. The best solution (minimal error) was found for  $N_e = 64$  with a value equal to 0.0159 m. This is an improvement corresponding to a ratio  $\approx 38$  in comparison with the initial error (background solution) and to a ratio  $\approx 5$  in comparison with the reference solution (variational solution). It is noticeable that the reference solution in terms of error value can be reached by EnKF with a low value of  $N_e$ , typically  $N_e = 8$ . To be certain to get the same performance as the reference solution it would be better to work with  $N_e \geq 64$ .

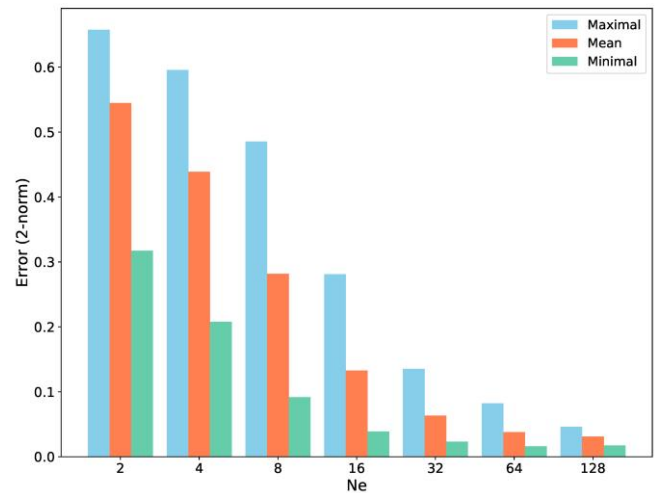


Figure 5. Statistic errors for EnKF

The longitudinal profile of the river with the water levels corresponding to the background solution  $x_b$  and to the EnKF best solution are depicted in Fig. 6.

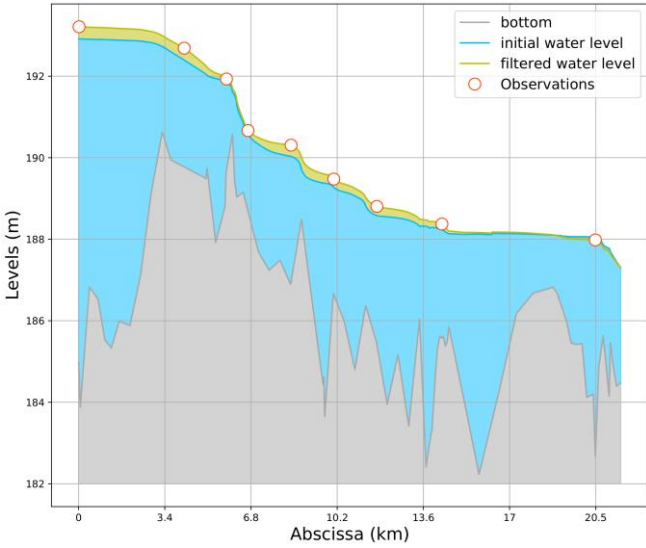


Figure 6. Results for the water levels (background and EnKF-based calibration)

Tab. 1 and Fig. 7 presents the optimal values for the friction parameters in the different zones of the river for the variational and filtering approaches. Results in Tab. 1 are rounded with no decimal part for reading convenience.

TABLE 1. FRICTION COEFFICIENTS RESULTS ( $m^{1/3}/s$ )

Zone	1	2	3	4	5	6	7	8	9
$x_b$	40	40	40	40	40	40	40	40	40
$\hat{x}_{L-BFGS}$	32	25	36	39	29	32	33	27	40
$\hat{x}_{EnKF}$	31	26	35	42	28	34	32	25	45

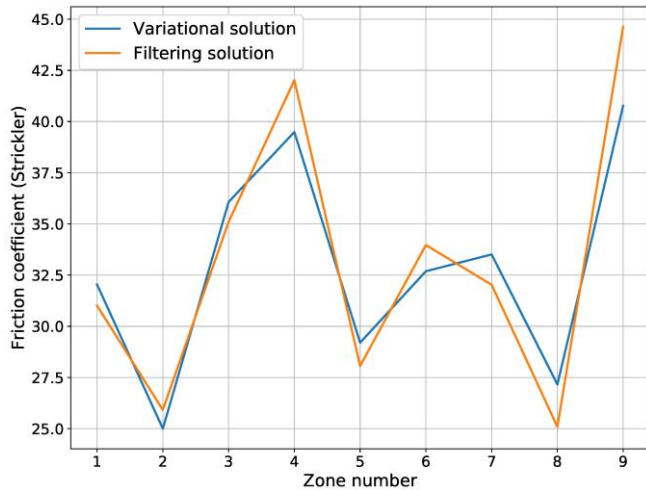


Figure 7. Optimal parameters with two DA methods

Calibrated parameters (Fig. 7) with variational and filtering approaches are not so far apart, although the resulting errors differ (0.0894 m vs 0.0159 m). Both results are in the same space where the filtering method was able to achieve a better accuracy. It is noticeable that if the convergence

tolerance of the L-BFGS algorithm is decreased, the differences with EnKF vanish.

Fig. 8 presents some convergences of EnKF for 100 assimilation cycles and different values of  $N_e$ . Since EnKF is a stochastic algorithm, the curves in Fig. 8 may slightly change but not so much in practice for this Mascaret calibration problem.

Anyway the EnKF convergence is clearly correlated with the value of  $N_e$ . It is not surprising to see how the errors are sensitive to the ensemble size. Too low values can lead to an undersampling of the search space of the parameters and consequently to the divergence of the filter. As expected best results are obtained for the largest ensemble sizes. The difference between  $N_e = 64$  and  $N_e = 128$  is small and here the best ratio between the calculation cost and the convergence error lies between these two values.

Whatever the tests here, the convergences have always been quite regular and monotonic.

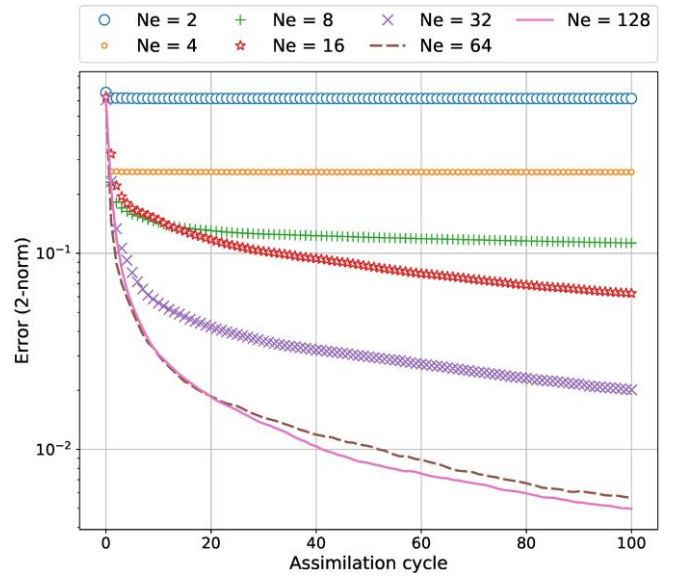


Figure 8. Typical convergence of EnKF for different values of  $N_e$

In this calibration problem the computational cost is extremely low as it is the steady state computational kernel of Mascaret that is used by TelApy. Consequently, the filtering with FilterPy is systematically done:  $t_{k+obs} = t_{k+1}$  for all  $k$  as Mascaret is able to compute the steady state in one step. Hence the number of Mascaret simulations needed by the EnKF calibration is equal to  $N_e$  multiplied by the number of assimilation cycles.

In order to see the relative importance of covariance matrices  $Q$  (2) and  $R$  (3) on the convergence of EnKF, many calibrations have been tested considering those matrices as diagonal with an identical variance value for all the terms and ranging from  $10^{-2}$  to  $10^{-8} m^2$ . Tests have been performed for  $N_e = 64$  and the convergence was stopped after 10 assimilation cycles. Fig. 9 shows the corresponding results with a representation of the error function in the variance space  $[10^{-2}, 10^{-8}] \times [10^{-2}, 10^{-8}]$  for two levels of fine and coarse discretization.

The two graphs in Fig. 9 give concordant information. One striking feature is that EnKF is almost independent on the values of  $diag(Q)$  and  $diag(R)$  in a wide range extending from  $10^{-3}$  to  $10^{-8}$ . The interaction effects between  $Q$  and  $R$  on the error function are practically non-existent.  $Q$  only increases the error function for a large variance value and only when  $diag(R)$  is small. As complement, large values for  $R$  have always the same impact whatever  $Q$  covariance is.

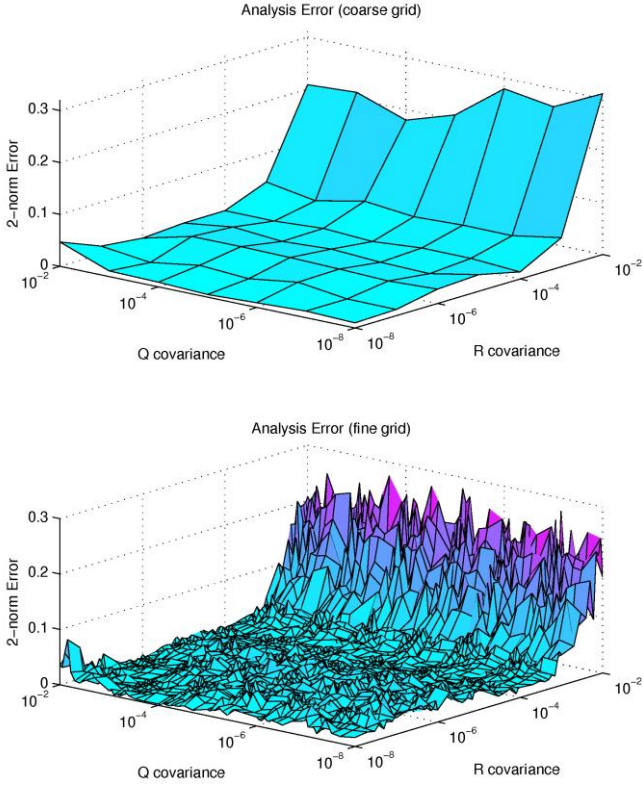


Figure 9. EnKF errors as a function of diagonal elements of  $Q$  and  $R$

The rapid convergence and the relatively low importance of the prediction model error indicate that the physics of the case while being non-linear is not so difficult to capture with the basic version of the EnKF algorithm. However, as the best solutions are only reached for the largest  $N_e$  values, the ensemble spread might be increased by a covariance inflation to mitigate the undersampling of low  $N_e$  values. Inflation or localisation techniques are not implemented in the actual version of FilterPy but they could be tested in the future with minimal changes in the module.

The covariance matrix for the convergence of the ensemble  $N_e = 128$  after 100 assimilation cycles is presented in Fig. 10. As expected the matrix is diagonal dominant with few interactions except for the three last zones where a negative covariance appears. Actually, this part of the river is the most difficult one to calibrate as the flow regime is sub-critical and constrained by an imposed water level at the end of the reach. Other off-diagonal elements are negligible indicating that there is no spurious correlation.

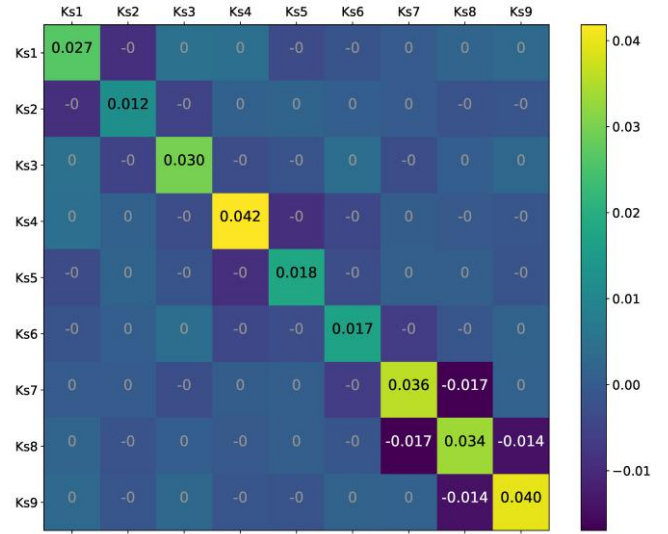


Figure 10. Covariance matrix for the Strickler's coefficients  $K_s$

### B. Telemac

Any model using the Telemac-Mascaret system can be calibrated according to Fig. 4. The implementation is relatively easy. The reader is invited to test the approach on a simple Telemac 2D case if interested [12].

To perform on a real test case, EnKF is used to calibrate three tidal parameters of a maritime configuration. The study concerns the Alderney Race that is located between the island of Alderney, UK, and the western tip of the Cotentin peninsula in Normandy, France. Details of this Telemac study are presented in [13]. The modelling covers an area of roughly  $55 \text{ km} \times 35 \text{ km}$  and the finite element discretization is composed of a triangular mesh with 17,983 nodes and 35,361 elements. The mesh size varies from 100 m on the shoreline and within the areas of interest to 1.8 km offshore, western and northern sectors.

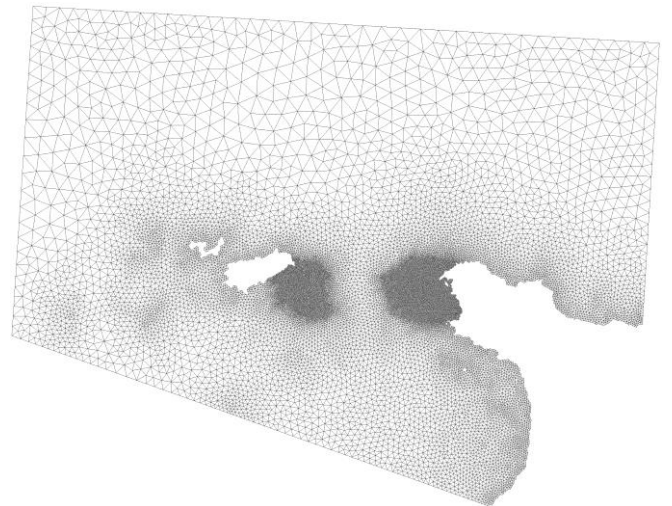


Figure 11. Model mesh

The boundary conditions of the model have been set up using depth-averaged velocities and water levels from the TPXO dataset (8 primary, 2 long-period and 3 nonlinear

constituents). The TPXO dataset is an accurate global models of ocean tides based on a best-fit of tidal levels measured along remote sensing tracks from the TOPEX/POSEIDON satellite project in operation since 2002. Moreover, velocities and water depths are imposed along the marine borders of the model using Thompson-type boundary conditions that allows internal waves to leave the domain with little or no reflection.

Several measurement campaigns were carried out to the west of Cap de la Hague. For this study, only the results of one of these (a campaign lasting six months) are used over a 5 days period from October 15th to 20th. This six-month campaign was carried out during summer of 2009 (from the end of July to the end of January). Two ADCPs were deployed to measure flow velocity (magnitude and direction) and water depth with 1 measurement every 10 minutes and one hour respectively.

The model calibration is done by estimating three tidal parameters that were proved to be the most influent ones on the Telemac results after a sensitivity analysis [13]. Tidal characteristics are imposed using a database of harmonic constituents to force the open boundary conditions. For each harmonic constituent, the water depth  $h$  and horizontal components of velocity  $u$  and  $v$  are calculated, at point  $M$  and time  $t$  by:

$$\begin{cases} F(M, t) = \sum_i F_i(M, t) \\ F_i(M, t) = f_i(t)A_{F_i}(M) \cos\left(\frac{2\pi t}{T_i} - \phi_{F_i}(M) + u_i^0 + v_i(t)\right) \end{cases} \quad (9)$$

where  $F$  is either the water level (referenced to mean sea level)  $z_s$  or one of the horizontal components of velocity  $u$  or  $v$ ,  $i$  refers to the considered constituent,  $T_i$  is the period of the constituent,  $A_{F_i}$  is the amplitude of the water level or one of the horizontal components of velocity,  $\phi_{F_i}$  is the phase,  $f_i(t)$  and  $v_i(t)$  are the nodal factors and  $u_i^0$  is the phase at the original time of the simulation.

The water level and velocities of each constituent are then summed to obtain the water depths and velocities for the open boundary conditions:

$$\begin{cases} h = \alpha \sum_i z_{s_i} - z_f + z_{mean} \\ u = \beta \sum_i u_i \\ v = \beta \sum_i v_i \end{cases} \quad (10)$$

where  $z_f$  is the bottom elevation and  $z_{mean}$  the mean reference level.

In (10), the multiplier coefficients of the tidal range and velocity, respectively  $\alpha$  and  $\beta$ , at boundary locations and the sea level  $z_{mean}$  are the tidal parameters to be calibrated with EnKF.

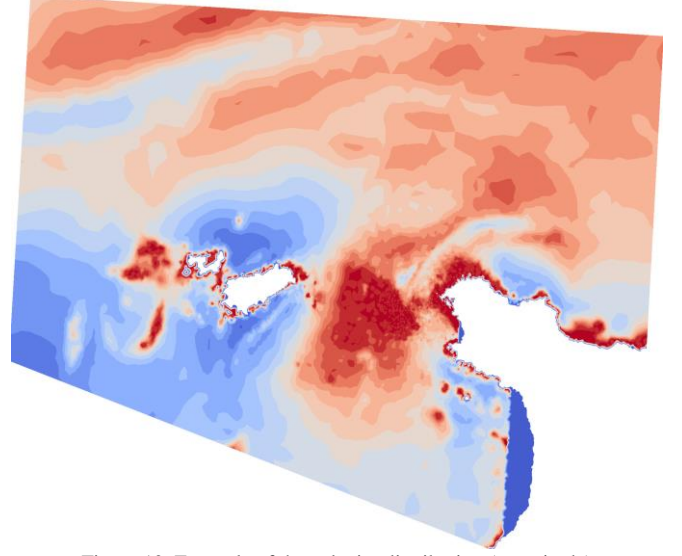


Figure 12. Example of the velocity distribution (magnitude)

The initial mean values for the ensemble of tidal parameters are set to  $z_{mean} = -1.6$ ,  $\alpha = 1$  and  $\beta = 1$  as prescribed by the original study of this model [13]. Fig. 13 shows the water depths on 24 hours (the first day of a five-day period) and at a point of observation, calculated before and after the calibration of tidal parameters by EnKF.

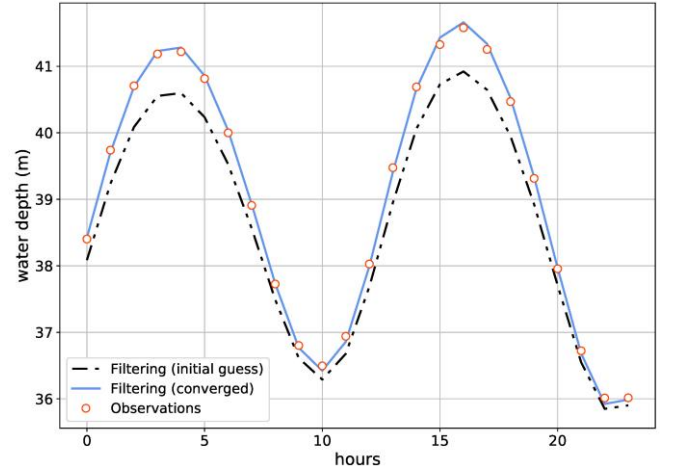


Figure 13. Water depths before and after EnKF convergence

The results on the water depth in Fig. 13 match very well with the observations. The error between the simulation and the observation is approximately four times lower than that obtained with a variational approach [13]. Indeed, the error 2-norm for the optimal solution of 3D-VAR is 0.83 m ( $z_{mean} = -0.995$ ,  $\alpha = 1.112$ ,  $\beta = 1.106$ ) and 0.22 m for the EnKF ( $z_{mean} = -1.149$ ,  $\alpha = 1.112$ ,  $\beta = 1.106$ ). Except for the sea level  $z_{mean}$  the two others solutions on the tidal range and velocity are identical.

The number of members  $N_e$  required to obtain the best EnKF estimate was only 5. In practice, the solution was not found the first time but it required few tests. In total the computational cost was less important than working with a much larger ensemble.

The filtering step was applied every hour all along the five-day period. Fig. 14 presents the convergence for the three tidal parameters (mean values) together with the decrease of the error norm between observations and estimations

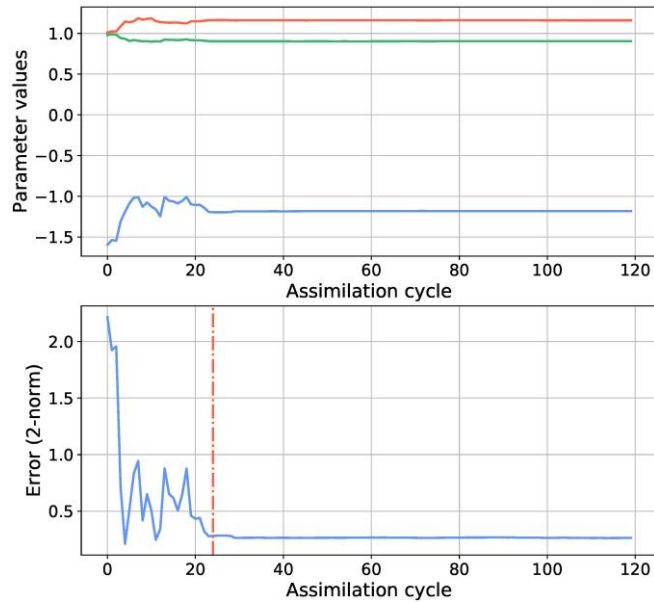


Figure 14. Convergence reached in 24 hours

It should be noted that only 24 assimilations (24 hours with one assimilation per hour) are necessary to reach the best estimates. For the remainder of the period, no further changes are identified. The entire five-day window is not necessary in this example, but that does not mean that the parameters can not change over time beyond this period. This will surely depend on the time of year and weather conditions.

The computational cost here is very acceptable and less important than those of the variational approach [13]. For example on a standard PC workstation of four cores (Intel® Xeon® CPU E3-1240@3.4 GHz), the time required to lead the assimilation in Fig. 14 to its end was about 40 minutes.

## V. CONCLUSIONS

This article presents a Python-based framework for data assimilation with the Telemac system. The data assimilation technique studied is the Ensemble Kalman Filter. It is shown how this filter can work well with the Telemac system for optimal model calibration with a small number of parameters. The filter is also considered competitive compared to other methods for its ease of implementation and its quick convergence on fairly simple problems.

In perspective, this first use of EnKF with the Telemac system at EDF R&D will certainly be continued on more complex models, that is to say with a greater dimensionality or a different physics. This will be an opportunity to test filter improvement techniques such as localisation and inflation, to use HPC as EnKF is easily parallelisable, or to compare the performance on surrogate models.

## ACKNOWLEDGEMENT

The authors gratefully acknowledge the open source community and especially that of Python. Many thanks to Dr. Sophie Ricci (CERFACS) and Vanessya Laborie (CEREMA) for the motivating discussions on EnKF and the Telemac-Mascaret system.

## REFERENCES

- [1] E. Blayo, M. Bocquet, E. Cosme, and L.F. Cugliandolo, "Advanced Data Assimilation for Geosciences," Lecture Notes of the Les Houches School of Physics: Special Issue, Oxford University Press, June 2012
- [2] A. Carrassi, M. Bocquet, L. Bertino, and G. Evensen, "Data Assimilation in the Geosciences - An overview on methods, issues and perspectives," unpublished
- [3] G. Evensen, "Sequential data assimilation with a nonlinear quasi-geographic model using Monte Carlo methods to forecast error statistics," *J. Geophys. Res.*, vol. 99, pp. 10143-10162, 1994
- [4] I.S. Strub, J. Percelay, O.-P. Tossavainen, and A.M. Bayen, "Comparison of two data assimilation algorithms for shallow water flows," *Networks and Heterogeneous Media*, ©American Institute of Mathematical Sciences, vol. 4, Num. 2, pp. 409-430, June 2009.
- [5] S. Barthélémy, S. Ricci, M.C. Rochoux, E. Le Pape, and O. Thual, "Ensemble-based data assimilation for operational flood forecasting – On the merits of state estimation for 1D hydrodynamic forecasting through the example of the 'Adour Maritime' river," *Journal of Hydrology*, vol. 552., pp. 210-224, 2017.
- [6] J.L. Crassidis and J.L. Junkins, "Optimal Estimation of Dynamic Systems," CRC Press, pp. 257-9, Second Edition 2012
- [7] Y. Audouin, C. Gœury, F. Zaoui, R. Ata, S. El Idrissi Essebtey, A. Torossian, and D. Rouge, "Interoperability Applications of TELEMAC-MASCARET System," 24<sup>th</sup> Telemac-Mascaret User Conference, Graz, Austria, pp. 57- 64, October 2017
- [8] R.E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, pp. 35-45, 1960
- [9] R. Labbe, "Kalman and Bayesian Filters in Python," <https://github.com/rlabbe/Kalman-and-Bayesian-Filters-in-Python>
- [10] C. Gœury, Y. Audouin, and F. Zaoui, "User documentation v7p3 of TelApy module," 2017
- [11] F. Demangeon, C. Gœury, F. Zaoui, N. Goutal, V. Pascual, and L. Hascoët, "Algorithmic differentiation for the optimal calibration of a shallow water model," *La Houille Blanche*, vol. 4, pp. 53-61, 2016
- [12] F. Zaoui, Ensemble Kalman filter example for Telemac-Mascaret, <https://github.com/fzao/telenkf>
- [13] C. Gœury, A. Ponçot, J.-P. Argaud, F. Zaoui, R. Ata, and Y. Audouin, "Optimal calibration of TELEMAC-2D models based on a data assimilation algorithm", 24<sup>th</sup> Telemac-Mascaret User Conference, Graz, Austria, pp. 73- 80, October 2017



