UNIVERSITÀ DI PARMA

UNIVERSITA' DEGLI STUDI DI PARMA

Dottorato di Ricerca in Ingegneria Civile e Architettura
Ciclo XXXIII

in CO-TUTELA con

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Advanced techniques for solving groundwater and surface water problems in the context of inverse methods and climate change

Coordinatore:
Chiar.mo Prof. Sandro Longo

Tutori:
Chiar.ma Prof.ssa Maria Giovanna Tanda
Chiar.mo Prof. J. Jaime Gòmez-Hernàndez

Co-Tutore:
Dott. Marco D'Oria

Dottoranda: Valeria Todaro

Anni Accademici 2017/2018 – 2019/2020

Acknowledgements

I would first like to express my special thanks to Prof. Tanda for her attentive, continuous and thoughtful guidance throughout my studies.

Many thanks to Prof. Gomez-Hernandez for his invaluable advice, insightful feedback and the great hospitality in Valencia.

Also, my deep gratitude to Dr. Marco D'Oria for the constant support and the precious suggestions, without which this work would not have been the same.

I would also thank all the professors and colleagues of the Parma research group for the positive and stimulating work environment they create every day. Many thanks to Dr. Federico Prost for his patient collaboration in developing part of the research presented in chapter 4.

I am also very grateful to the Valencian research group for the helpful Kalman filter discussions, the shared paellas, all the good times.

Finally, particular thanks go to my old and new friends for making these three years memorable, to my sister for always being my biggest supporter and to my parents for everything they did and still do for me.

Abstract

This work focuses on the investigation of advanced techniques to handle groundwater and surface water problems in the framework of inverse methods and climate change. The Ensemble Kalman filter methods, with particular attention to the Ensemble Smoother with Multiple Data Assimilation (ES-MDA), are extensively analyzed and improved for the solution of different types of inverse problems. In particular, the main novelty is the application of these methods for the identification of time series function.

In the first part of the thesis, after the description of the ES-MDA method, the development of a Python software package for the application of the proposed methodology is presented. It is designed with a flexible workflow that can be easily adapted to implement different variants of the Ensemble Kalman filter and to be applied for the solution of various types of inverse problems. A complemented tool package provides several functionalities that allow to setup the algorithm configuration suiting the specific analyzed problem.

The first novelty application of the ES-MDA method aimed at solving the reverse flow routing problem. The objective of the inverse procedure is the estimation of an unknown inflow hydrograph to a hydraulic system on the basis of information collected downstream and a given forward routing model that relates inflow hydrograph and downstream observations. The procedure is tested by means of two synthetic examples and a real case study; the impact of ensemble

sizes and the application of covariance localization and inflation techniques are also investigated. The tests show the capability of the proposed method to solve this type of problem; the performance of ES-MDA improves, especially for small ensemble sizes, when covariance localization and inflation techniques are applied.

The second application, in the context of surface water, concerns the calibration of a hydrological-hydraulic model that simulates rainfall-runoff processes. The ES-MDA is coupled with the numerical model by parallel way for the estimation of roughness and infiltration coefficients based on the knowledge of a discharge hydrograph at the basin outlet. The results of two synthetic tests and a real case study demonstrate the capability of the proposed method to calibrate the hydrological-hydraulic model with a reasonable computational time.

In the groundwater field, ES-MDA is applied for the first time to simultaneously identify the source location and the release history of a contaminant spill in an aquifer from a sparse set of concentration data collected in few points of the aquifer. The impacts of the concentration sampling scheme, the ensemble size and the use of covariance localization and covariance inflation techniques are tested; furthermore, a new procedure to perform a spatiotemporal iterative localization is presented. The methodology is tested by means of an analytical example and a study case that uses real data collected in a laboratory sandbox. ES-MDA leads to a good estimation of the investigated parameters; a well-designed monitoring network and the use of covariance corrections improve the performance of the method and help to minimize ill-posedness and equifinality.

A part of the thesis investigates the impact of climate change on the groundwater availability. A surrogate model that describes the response of groundwater levels to meteorological variables up to 2100 is presented. It is a simple statistical approach based on the correlations between groundwater levels and two drought indices that depend on precipitation and temperature data. The presented method is used to evaluate the impact of climate change on groundwater resources in a

study area located in Northern Italy using historical and regional climate model data. The results denote a progressive increase of groundwater droughts in the investigated area.

Sommario

La tesi si sviluppa su diverse tematiche idrologiche relative alle acque sotterranee e superficiali nell'ambito dei problemi inversi e del cambiamento climatico. I metodi basati sul filtro di Kalman, con particolare attenzione al metodo Ensemble Smoother with Multiple Data Assimilation (ES-MDA), sono analizzati e migliorati per la soluzione di diversi tipi di problemi inversi. In particolare, una delle novità è l'applicazione di tali metodi per l'identificazione di serie temporali.

Uno degli obiettivi della tesi è lo sviluppo di un pacchetto software scritto in linguaggio Python per l'applicazione della metodologia presentata. Il software è progettato in modo da poter essere facilmente adattato a diverse varianti del filtro di Kalman e da poter essere applicato per la soluzione di differenti tipi di problemi. Sono forniti diversi strumenti che consentono di impostare una configurazione dell'algoritmo che meglio si adatta al caso specifico in esame.

La prima applicazione del metodo ES-MDA riguarda la determinazione dell'idrogramma delle portate in ingresso ad un sistema idraulico, sulla base di informazioni disponibili a valle (reverse flow routing) e un dato modello in avanti. Al fine di accertare le capacità del metodo sono stati sviluppati preliminarmente due esempi sintetici, per i quali viene valutata anche l'influenza delle dimensioni dell' "ensemble" e l'applicazione di alcune modifiche all'algoritmo, come la localizzazione e le tecniche di "inflation". Infine, ES-MDA è applicato a un caso studio reale. I risultati mostrano la capacità del metodo proposto di risolvere questo

tipo di problema; le prestazioni di ES-MDA migliorano, soprattutto per "ensemble" di piccole dimensioni, quando vengono applicate le tecniche di localizzazione e "inflation".

La seconda applicazione, nell'ambito delle acque superficiali, riguarda la calibrazione di un modello idrologico-idraulico che simula i meccanismi di formazione di eventi di piena a partire da sollecitazioni idrometeorologiche e la successiva propagazione. ES-MDA e il modello numerico sono accoppiati per la stima dei coefficienti di scabrezza e infiltrazione sulla base di un idrogramma delle portate noto in una sezione del dominio. I risultati di due test sintetici e un caso di studio reale dimostrano la capacità del metodo proposto di calibrare il modello idrologico-idraulico con un tempo di calcolo accettabile.

Nel campo delle acque sotterranee, ES-MDA viene applicato per la prima volta per identificare simultaneamente la posizione della sorgente e la storia di rilascio di un inquinante in una falda acquifera, noti alcuni dati di concentrazione rilevati in diversi punti del dominio. Numerosi test sono stati eseguiti per valutare l'influenza della distribuzione spaziale e temporale dei dati di concentrazione, la numerosità dell' "ensemble" e l'uso delle tecniche di localizzazione e "inflation"; inoltre, viene presentata una nuova procedura per eseguire una localizzazione iterativa spazio-temporale. La metodologia è validata mediante un esempio analitico e un caso di studio per il quale sono utilizzati dati ottenuti in laboratorio mediante una sandbox. ES-MDA porta ad una buona ricostruzione dei parametri investigati; una rete di monitoraggio ben progettata e l'applicazione delle modifiche sull'algoritmo (localizzazione e "inflation") migliorano le prestazioni del metodo e aiutano a mitigare il possibile problema della non univocità della soluzione.

Una parte della tesi riguarda lo studio dell'impatto del cambiamento climatico sulla disponibilità idrica delle falde acquifere. A tale scopo, viene sviluppato un modello surrogato capace di descrive la risposta dei livelli di falda alle variabili meteorologiche fino al 2100. Si tratta di un semplice approccio statistico basato

sulle correlazioni tra i livelli di falda e due indici di siccità che dipendono dai dati di precipitazioni e temperatura. Il metodo viene utilizzato per valutare l'impatto del cambiamento climatico sulle risorse idriche sotterranee in un'area di studio situata in Nord Italia, utilizzando i dati di serie storiche ed estratti da modelli climatici regionali. I risultati denotano un progressivo aumento della siccità delle acque sotterranee nell'area di studio.

Resumen

El tema de la investigación se centra en técnicas avanzadas para manejar problemas de aguas subterráneas y superficiales relacionados con métodos inversos y cambio climático. Los filtros de Kalman, con especial atención en Ensemble Smoother with Multiple Data Assimilation (ES-MDA), se analizan y mejoran para la solución de diferentes tipos de problemas inversos. En particular, la principal novedad es la aplicación de estos métodos para la identificación de series temporales.

La primera parte de la tesis, luego de la descripción del método ES-MDA, presenta el desarrollo de un software escrito en lenguaje Python para la aplicación de la metodología propuesta. El software cuenta con un flujo de trabajo flexible que puede adaptarse fácilmente para implementar diferentes variantes del filtro de Kalman y ser aplicado para la solución de varios tipos de problemas. Un paquete complementar de herramientas proporciona varias funcionalidades que permiten de configurar el algoritmo de acuerdo con el problema específico analizado.

La primera aplicación se refiere a un nuevo enfoque para la solución del problema inverso de flujo en ríos. Este es un procedimiento inverso destinado a estimar el flujo de entrada a un sistema hidráulico en función de información recopilada aguas abajo. El procedimiento se prueba mediante dos ejemplos sintéticos y un estudio de caso real; se investiga el impacto de los tamaños de los conjuntos y la aplicación de técnicas de localización e inflación de covarianzas. Los resultados muestran la capacidad del método propuesto de resolver este tipo de problemas;

el rendimiento de ES-MDA mejora, especialmente para tamaños de conjuntos pequeños, cuando se aplican técnicas de inflación y localización de covarianza.

La segunda aplicación en el campo de las aguas superficiales se refiere a la calibración de un modelo hidrológico-hidráulico que simula los mecanismos de formación de eventos de inundación a partir de tensiones hidrometeorológicas y su posterior propagación. ES-MDA se acopla al modelo numérico de forma paralela para la estimación de los coeficientes de rugosidad e infiltración en base al conocimiento de un hidrograma de flujo en una sección del dominio. Los resultados de dos casos sintéticos y un estudio de caso real demuestran la capacidad del método propuesto para calibrar el modelo hidrológico-hidráulico con un tiempo computacional razonable.

En el campo de aguas subterráneas, ES-MDA se aplica por primera vez para identificar simultáneamente la ubicación de la fuente y el historial de liberación de un contaminante en un acuífero a partir de un conjunto de datos de concentración detectados en diferentes puntos del dominio. Se realizaron numerosas pruebas para evaluar la influencia de la distribución espacial y temporal de los datos de concentración, el número del conjunto y el uso de técnicas de localización e inflación; además, se presenta un nuevo procedimiento para realizar una localización iterativa espacio-temporal. La metodología se valida mediante un ejemplo analítico y un estudio de caso para el que se utilizan datos obtenidos en el laboratorio mediante una caja de arena. ES-MDA conduce a una buena reconstrucción de los parámetros investigados; una red de monitoreo bien diseñada y la aplicación de correcciones de covarianza mejoran el rendimiento del método y ayudan a mitigar el posible problema de no unicidad de la solución.

Otro propósito de la investigación es investigar el efecto del cambio climático en las aguas subterráneas. Se presenta un modelo simplificado que describe la respuesta de los niveles de agua subterránea a las variables meteorológicas hasta 2100. Es un enfoque estadístico sencillo basado en las correlaciones entre los

niveles de agua subterránea y dos índices de sequía que dependen de los datos de precipitación y temperatura. El método se utiliza para evaluar el impacto del cambio climático en los recursos de agua subterránea en un área de estudio ubicada en el norte de Italia utilizando datos históricos y de modelos climáticos regionales. Los resultados muestran un aumento progresivo de la sequía de aguas subterráneas en el área de estudio.

Resum

El tema de la investigació se centra en tècniques avançades per a manejar problemes d'aigües subterrànies i superficials relacionats amb mètodes inversos i canvi climàtic. Els filtres de Kalman, amb especial atenció en Ensemble Smoother with Multiple Data Assimilation (ES-MDA), s'analitzen i milloren per a la solució de diferents tipus de problemes inversos. En particular, la principal novetat és l'aplicació d'aquests mètodes per a la identificació de sèries temporals.

La primera part de la tesi presenta el desenvolupament d'un programari escrit en llenguatge Python per l'aplicació de la metodologia presentada. El programari compta amb un flux de treball flexible que pot adaptar-se fàcilment per a implementar diferents variants del filtre de Kalman i ser aplicat per a la solució de diversos tipus de problemes. Un paquet complementar d'eines proporciona diverses funcionalitats que permeten de configurar l'algorisme d'acord amb el problema específic analitzat.

La primera aplicació es refereix a un nou enfocament per a la solució del problema invers de flux en rius. Aquest és un procediment invers destinat a estimar el flux d'entrada a un sistema hidràulic en funció d'informació recopilada aigües avall. El procediment es prova mitjançant dos exemples sintètics i un estudi de cas real; s'investiga l'impacte de les grandàries dels conjunts i l'aplicació de tècniques de localització i inflació de covariàncies. Els resultats mostren la capacitat del mètode proposat de resoldre aquest tipus de problemes; el rendiment de ES-

MDA millora, especialment per a grandàries de conjunts xicotets, quan s'apliquen tècniques d'inflació i localització de covariància.

La segona aplicació en el camp de les aigües superficials es refereix al calibratge d'un model hidrològic-hidràulic que simula els mecanismes de formació d'esdeveniments d'inundació a partir de sollicitació hidrometeorológicas i la seua posterior propagació. ES-MDA s'acobla al model numèric de manera paral·lela per a l'estimació dels coeficients de rugositat i infiltració sobre la base del coneixement d'un hidrograma de flux en una secció del domini. Els resultats de dos casos sintètics i un estudi de cas real demostren la capacitat del mètode proposat per a calibrar el model hidrològic-hidràulic amb un temps computacional raonable.

En el camp d'aigües subterrànies, ES-MDA s'aplica per primera vegada per a identificar simultàniament la ubicació de la font i l'historial d'alliberament d'un contaminant en un aqüífer a partir d'un conjunt de dades de concentració detectats en diferents punts del domini. Es van realitzar nombroses proves per a avaluar la influència de la distribució espacial i temporal de les dades de concentració, el número del conjunt i l'ús de tècniques de localització i inflació; a més, es presenta un nou procediment per a realitzar una localització iterativa espaciotemporal. La metodologia es valguda mitjançant un exemple analític i un estudi de cas per al qual s'utilitzen dades obtingudes en el laboratori mitjançant una caixa d'arena. ES-MDA condueix a una bona reconstrucció dels paràmetres investigats; una xarxa de monitoratge ben dissenyada i l'aplicació de correccions de covariància milloren el rendiment del mètode i ajuden a mitigar el possible problema de no unicitat de la solució.

Un altre propòsit de la investigació és investigar l'efecte del canvi climàtic en les aigües subterrànies. Es presenta un model simplificat que descriu la resposta dels nivells d'aigua subterrània a les variables meteorològiques fins a 2100. És un enfocament estadístic senzill basat en les correlacions entre els nivells d'aigua subterrània i dos índexs de sequera que depenen de les dades de precipitació i

temperatura. El mètode s'utilitza per a avaluar l'impacte del canvi climàtic en els recursos d'aigua subterrània en una àrea d'estudi situada en el nord d'Itàlia utilitzant dades històriques i de models climàtics regionals. Els resultats mostren un augment progressiu de la sequera d'aigües subterrànies en l'àrea d'estudi.

Contents

List of Figures

List of Tables

Introduction

The thesis presents innovative techniques for the solution of surface and subsurface hydrology problems in two main thematic areas: the inverse methods and climate change.

The inverse problem is one of the most important mathematical problem as it allows to estimate unknown parameters that can not be directly observed. It is the process of identifying input parameters using output measurements; it has found numerous applications in geophysics, communication theory, optics, radar, acoustics, medical imaging, meteorology, oceanography, astronomy, and other many scientific fields. A huge number of approaches have been proposed in the literature to handle this issue. This thesis focuses on the solution of inverse problems in the context of surface and subsurface hydrology using ensemble Kalman filter techniques.

The Ensemble Smoother with Multiple Data Assimilation (ES-MDA, Emerick & Reynolds (2013)) is extensively analyzed and improved for the solution of inverse problems. The first application deals with the solution of an inverse problem in the hydrology field: ES-MDA is used for the estimation of discharge hydrographs at ungauged sections using information collected downstream. The indirect estimation of the inflow to a river reach is often required since only few sections are equipped to record data; this is particularly challenging for sections that do not have reliable data upstream and the common forward flow routing cannot be used

at this purpose. The proposed method is a new approach for the solution of the reverse flow routing problem.

The second application in the hydrology field aims at the calibration to a hydrological-hydraulic model that simulates rainfall-runoff processes. The roughness and infiltration coefficients, which are input data required by the investigated numerical model, are estimated on the basis of a discharge hydrograph observed in the outlet section of the river.

Another novel application of ES-MDA is in the groundwater field and it deals with the simultaneous reconstruction of the release history and the identification of the source location of a groundwater contamination event from observed concentration data. This is an inverse problem whose solution is still open in the literature and for the first time is solved with the proposed approach.

The final objective of this work is to develop a software package for the solution of inverse problems based on ensemble Kalman methods. There are several open-source codes for the application of these approaches, but they are usually challenging to use. The innovation of the developed software is the easy implementation and the supply of useful tools to improve the performance of the method. The open-source codes are written in Python programming language and accompanied by supporting documentation. Python is one of the most popular programming languages of the last decade: it has a simple syntax, it is platform-independent, and it is free. The Ensemble Smoother with Multiple Data Assimilation is considered as reference for the development of the software, but it is kept as general as possible so that it can be easily adapted to other ensemble-based methods and extended to different tasks. A complete application example of the software is provided: the solution of the inverse problem for the identification of the source location and release history of a contaminant dispersion in groundwater.

Although this work focuses primarily on groundwater quality analysis, important issues of subsurface hydrology concerns quantitative aspects. A part of the

thesis focuses on the investigation of the impact of climate change on groundwater resources. Groundwater represents about 98% of the available fresh water on Earth and, in many cases, is the only resource of water in critical periods of the year, especially in arid and semi-arid regions, where surface water are almost absent for several months. The evaluation of the effect of climate change on groundwater resources in the future periods is thus a key issue. Despite its importance, only a few works are presented in the literature aimed at analyzing the impact of climate variability on groundwater resources, due to the difficulty to set up a complete subsurface model. In this work, a simple statistical approach to evaluate the impact of climate change on groundwater level, which is considered a good indicator of the aquifer condition, is proposed. The response of groundwater levels to projected meteorological variables is evaluated up to 2100 on the basis of precipitation and temperature data extracted from several Regional Climate Models under different climate scenarious.

The thesis is organized as follows: in the first chapter, the Ensemble Smoother with Multiple Data Assimilation methodology is presented together with the proposed modifications to the original algorithm. In Chapter 2, the software package for the application of ES-MDA is described. Chapter 3 describes the application of ES-MDA for the solution of the reverse flow routing problem; two synthetic cases and a real one are provided to demonstrate the capability of the proposed inverse procedure. Chapter 4 presents the analysis for the calibration of a hydrological-hydraulic model. In chapter 5, ES-MDA is used for the estimation of hydraulic and transport parameters of an aquifer and for the simultaneous identification of the source location and the release history of a groundwater pollutant. Chapter 6 is dedicated to the analysis of the impact of climate change on groundwater levels by means of a statistical approach. In the last chapter, the conclusions of the thesis and some suggestions for future research are outlined. Finally, the appendix reports an example of the Python codes written for the application pre-

sented in Chapter 5 for the simultaneous estimation of the source location and release history of a contaminant spill in an aquifer.

1

Ensemble smoother with multiple data assimilation

1.1. Introduction

The Ensemble Kalman-based methods are Monte Carlo implementations of the Kalman filter (KF), introduced by Kalman (1960). The KF is an optimal linear filter that allows to estimate unknown variables by using a series of measurements that are typically noisy. Linearity is a strong constraint of this method and makes it not applicable for many real cases, where complex systems can be nonlinear. Linearized versions of KF, such as the extended Kalman filter (EKF), have been proposed to overcome the linear assumption limitation. EKF uses the Kalman filter and a linear approximation of the nonlinear model, but its applications are restricted to small-scale problems and mild nonlinearities. Furthermore, the ad-

ditional cost of linearization make all the linearized versions of KF impratical in many cases.

The Ensemble Kalman-based methods derived from the ensemble Kalman filter (EnKF), initially proposed by Evensen (1994), allow to work with large-scale and nonlinear systems. Since the introduction of the EnKF, many variants of the method have been developed and widely applied in many scientific field for data assimilation and the estimations of system states and parameters. The present work focuses on the application of these methods for the solution of inverse problems. The investigated parameters are estimated based on the knowledge of observed measurements and a given forward model that relates parameters and observations. The main advantages of the ensemble Kalman-based methods, useful for this purpose, are its capability to be coupled with almost any forward models, the possibility of being implemented through parallel computing and assessing the uncertainty associated with the estimations, due to the generation of multiple alternative realizations. Moreover, they are more computationally efficient than other Monte Carlo inverse modeling methods due to the procedure used to compute the covariance matrices.

Among the ensemble-based methods, the ensemble smoother with multiple data assimilation (ES-MDA) has been analyzed in detail to solve the inverse problem. It is a valid alternative to the EnKF, for the case in which the time sequence of state observations is all available in full at the time of the analysis. ES-MDA, introduced by Emerick & Reynolds (2012, 2013), is a variant of the Ensemble Smoother, proposed by van Leeuwen & Evensen (1996). ES-MDA iteratively assimilates the same data multiple times in order to improve the results of the ES, which assimilates all data simultaneously in a single update step. The purpose of the multiple assimilation is to avoid the problem of overcorrection detected by by Evensen & van Leeuwen (2000) and Crestani et al. (2013) with the ES on its application to highly nonlinear problems with a single global update.

All the ensemble-based methods are affect to the fundamental undersampling problem, which arise when the size of the ensemble is so small that do not accurately reflect the statistics of the underlying population. Undersampling leads to two main problems: filter divergence and the appearance of long-range spurious correlations. Part of the thesis will focus on the analysis and improvement of the main techniques developed to overcome this problem: covariance localization and covariance inflation.

In this chapter, the implementation of ES-MDA and the corrections on the algorithm will be discussed.

1.2. ES-MDA

The ES-MDA is an iterative data assimilation method that updates the unknown parameters maintaining consistency with the observations. The inverse procedure requires that a reliable forward model is available:

$$\mathbf{Y} = g\left(\mathbf{X}\right). \tag{1.1}$$

The model operator $g\left(\cdot\right)$ predicts the system state at measurement locations, $\mathbf{Y} \in \Re^m$, given a realization of the model parameters $\mathbf{X} \in \Re^{N_p}$. Here, N_p is the number of parameters and m is the number of available observations. The parameter vector $\mathbf{X}$ is estimated on the basis of a set of observations $\mathbf{D} \in \Re^m$ of the system state $\mathbf{Y}$, which are assimilated N times.

The ES-MDA scheme consists of an initialization phase and two main iterative steps: a forecast step and an update step.

0. Initialization step

   Initially, the procedure requires to define an initial ensemble of parameters.

The ensemble realizations should be generated using all the available information, but often no prior data are available. It is suggested to generate the ensemble semi-randomly on the basis of expert knowledge. For instance, if the parameters represent a discretized function in time, imposing some degree of continuity in the prior information can lead to a smooth solution consistent with the available data. This can be achieved by generating the prior ensemble as random discretized time functions. Else, if the parameters to be estimated are discrete values, the ensemble can be generated using random values selected over a range that guarantees the consistency of all realizations with the considered problem.

The second preliminary step includes the choice of the number of iterations N, the generation of an ensemble of observed data measurement errors and the definition of inflated coefficients α_i required by the ES-MDA procedure. The observation errors are assumed to follow a Gaussian distribution of mean zero and covariance matrix $\mathbf{R} \in \Re^{m \times m}$. The coefficients α_i applies to the measurement error and its covariance matrix, at each iteration i, and help to avoid overcorrections. They must satisfy the condition:

$$\sum_{i=1}^{N} \frac{1}{\alpha_i} = 1, \tag{1.2}$$

which guarantees an exact equivalence between single and multiple data assimilation methods at least for linear models. The scheme proposed by Evensen (2018) is used for the computation of the α_i, which ensures that the constraint of Eq. 1.2 is satisfied. The procedure starts selecting any nonzero value for α'_1, then the following α'_i are computed as:

$$\alpha'_{i+1} = \alpha'_i / \alpha_{geo}. \tag{1.3}$$

where the constant α_{geo} controls the extent of the change of α_i from one iteration to the next. At the end, the values from Eq. 1.3 are scaled to obtain the final coefficients:

$$\alpha_i = \alpha'_i \left(\sum_{i=1}^{N} \frac{1}{\alpha'_i} \right). \tag{1.4}$$

The simplest choice is to consider α_{geo}=1 that leads to constant α_i=N. However, a gradual decrease of α_i, obtained with $\alpha_{geo} > 1$, can improve the performance of the method, since it reduces the magnitude of the initial updates in which the misfit between observations and model predictions is usually larger.

1. Forecast step

   Predictions are obtained, by means of the forward model, for each realization j of the parameter ensemble. For the first iteration, $\mathbf{Y}$ is generated using the initial ensemble of parameters; for the following iterations, the ensemble of predictions is generated using the updated parameters.

$$\mathbf{Y}_{j,i} = g\left(\mathbf{X}_{j,i}\right). \tag{1.5}$$

2. Update step

   Parameters are updated, for each realization of the ensemble j and iteration i according to the following equation, based on the misfit between observations $\mathbf{D}$ and corresponding model predictions $\mathbf{Y}$:

$$\mathbf{X}_{j,i+1} = \mathbf{X}_{j,i} + \frac{\mathbf{C}^i_{\mathbf{XY}}}{\mathbf{C}^i_{\mathbf{YY}} + \alpha_i \mathbf{R}} \left(\mathbf{D} + \sqrt{\alpha_i}\varepsilon_j - \mathbf{Y}_{j,i} \right), \tag{1.6}$$

   where $\mathbf{C}^i_{\mathbf{XY}}$ is the cross-covariance matrix between parameters and predic-

tions and $\mathbf{C}^i_{\mathbf{YY}}$is the autocovariance matrix of predictions. They are computed from the ensemble at each iteration i as:

$$\mathbf{C}^i_{\mathbf{XY}} = \frac{1}{N_e - 1} \sum_{j=1}^{N_e} \left(\mathbf{X}_{j,i} - \overline{\mathbf{X}}_i\right) \left(\mathbf{Y}_{j,i} - \overline{\mathbf{Y}}_i\right)^T, \tag{1.7}$$

$$\mathbf{C}^i_{\mathbf{YY}} = \frac{1}{N_e - 1} \sum_{j=1}^{N_e} \left(\mathbf{Y}_{j,i} - \overline{\mathbf{Y}}_i\right) \left(\mathbf{Y}_{j,i} - \overline{\mathbf{Y}}_i\right)^T, \tag{1.8}$$

where N_e is the total number of ensemble realizations and $\overline{\mathbf{X}}_i$ and $\overline{\mathbf{Y}}_i$ are the ensemble means, at iteration i, of parameters and predictions, respectively.

Then, return to the forecast step considering $\mathbf{X}_{j,i} = \mathbf{X}_{j,i-1}$ and repeat until the last iteration.

At the end of each update step, it is possible to apply a linear relaxation on the ensemble realizations. The linear relaxation, similarly to the effect of the α coefficients, reduces the changes made to the parameters from one iteration to the next and prevent the filter divergence The solution of Eq. 1.6 is modified as follows:

$$\widetilde{\mathbf{X}}_{j,i+1} = (1 - w)\, \mathbf{X}_{j,i+1} + w\mathbf{X}_{j,i} \tag{1.9}$$

where w is the relaxation coefficient selected in the range 0-1.

When necessary, the update step can be performed in a transformed space in order to prevent the appearance of unphysical negative values for some types of problems. In these cases, the vector of parameters is transformed before the update step and back transformed into the parameter space after the updating. Covariances and cross-covariances must be computed in the transformed space, too. The most common transformations used to handle non-negative data are the logarithmic and the square root. Sometimes, it is useful to constrain the vector of estimated parameters to a specific range; transformations can also be used

for this purpose. The logarithmic and the square root transformations can be modified to ensure the parameters are in the specific interval [a,b]. The modified log-transformation, is given by:

$$f(x) = y = \log\left(\frac{x-a}{b-x}\right). \tag{1.10}$$

The inversion of this expression gives the appropriate back-transformation:

$$f^{-1}(x) = \frac{(b-a)\,e^{y}}{1+e^{y}} + a \tag{1.11}$$

The modified square root transformation is defined as follows:

$$f(x) = y = \left(\frac{x-a}{b-x}\right)^{\frac{1}{2}} \tag{1.12}$$

and the corresponding back-transformation is:

$$f^{-1}(x) = \frac{(b-a)\,y^{2}}{1+y^{2}} + a \tag{1.13}$$

It is noteworthy that a different transformation can be applied for each parameter to be estimated.

1.3. Undersampling problems

Undersampling occurs when the size of the ensemble is so small that it is not statistically representative of the variability of the unknowns and can cause the appearance of long-range spurious correlations and filter divergence. The computational burden depends on the ensemble size since the computation of the prediction vectors $\mathbf{Y}_{j,i}$ requires N_e simulations at each iteration. Therefore, the

number of ensemble realizations should be kept as small as possible to reduce computational time. Covariance localization and covariance inflation techniques have been developed to overcome this problem.

1.3.1. Covariance localization

Covariance localization (CL) is a technique developed to mitigate the problem of long-range spurious correlations. CL, at the same time, expands the degrees of freedom available to assimilate data increasing the rank of the covariance matrices computed from the ensemble, which are usually rank deficient even more so when the ensemble size is lower than the number of unknown parameters or observations. Different covariance localization approaches have been proposed in literature (Houtekamer & Mitchell 1998, Hamill et al. 2001, Anderson 2007, Chen & Oliver 2009) considering correlations among spatial dependent variables. However, parameters may be time dependent or both time-space dependent. In this work, a new localization approach, which takes into account both spatial and temporal distance, is presented.

CL is done by element-wise multiplication (Schur product or Hadamard product) of the original covariance matrix and a distance dependent correlation function ρ that smoothly reduces the correlations between points for increasing distances and cuts off long-range correlations above a specific distance. The covariances in Eqs. 1.7 and 1.8, computed in the update step, are modified as follows:

$$\widetilde{\mathbf{C}}^i_{\mathbf{XY}} = \rho^i_{\mathbf{XY}} \circ \mathbf{C}^i_{\mathbf{XY}}, \tag{1.14}$$

$$\widetilde{\mathbf{C}}^i_{\mathbf{YY}} = \rho_{\mathbf{YY}} \circ \mathbf{C}^i_{\mathbf{YY}}. \tag{1.15}$$

where $\circ$ represents the elementwise multiplication and $\rho^i_{\mathbf{XY}}$ and $\rho_{\mathbf{YY}}$are correlation matrices based on spatial and temporal distances between parameters and

observations and between observations and observations, respectively. The correlations in space ($\rho^i_{\mathbf{XY},s}$, $\rho_{\mathbf{YY},s}$) and time ($\rho_{\mathbf{XY},t}$, $\rho_{\mathbf{YY},t}$) are computed independently and then coupled via a Schur product:

$$\rho^i_{\mathbf{XY}} = \rho^i_{\mathbf{XY},s} \circ \rho_{\mathbf{XY},t}, \tag{1.16}$$

$$\rho_{\mathbf{YY}} = \rho_{\mathbf{YY},s} \circ \rho_{\mathbf{YY},t}. \tag{1.17}$$

The fifth-order correlation function introduced by Gaspari & Cohn (1999) is used; it smoothly reduces the correlations between points for increasing distances and cuts off long-range correlations above a specific distance:

$$\rho = \begin{cases} -\frac{1}{4}\left(\frac{\delta}{b}\right)^5 + \frac{1}{2}\left(\frac{\delta}{b}\right)^4 + \frac{5}{8}\left(\frac{\delta}{b}\right)^3 - \frac{5}{3}\left(\frac{\delta}{b}\right)^2 + 1, & 0 \leq \delta \leq b; \\ \frac{1}{12}\left(\frac{\delta}{b}\right)^5 - \frac{1}{2}\left(\frac{\delta}{b}\right)^4 + \frac{5}{8}\left(\frac{\delta}{b}\right)^3 + \frac{5}{3}\left(\frac{\delta}{b}\right)^2 - 5\left(\frac{\delta}{b}\right) + 4 - \frac{2}{3}\left(\frac{\delta}{b}\right)^{-1}, & b \leq \delta \leq 2b; \\ 0 & \delta \geq 2b; \end{cases} \tag{1.18}$$

where δ represents the parameter-observation or observations-observation distances in space ($\delta^i_{XY,s}$, $\delta_{YY,s}$) or time ($\delta_{XY,t}$, $\delta_{YY,t}$). The spatial distances between parameters and observations may be unknown if the location of the parameters change over the iterative process, $\delta^i_{XY,s}$ and $\delta^i_{XY,t}$ can be updated at each iteration i considering the update parameters at the previous iteration. The coefficient b characterizes the space (b_s) or time (b_t) distance at which the covariances become zero.

1.3.2. Covariance inflation

Covariance inflation is a technique developed to overcome the problem of filter divergence. The filter divergence may occur when the variance is underestimated leading to overconfidence in prior estimates and, as a consequence, the ensemble

collapses into a set of too similar realizations, which could be different from the true solution. This reduces the weight given to subsequent updates and can lead to a divergence of the ensemble since the filter is not able to adjust an incorrect estimation. Covariance inflation can be achieved by different ways (see e.g. Anderson 2007, Li et al. 2009, Liang et al. 2011, Wang & Bishop 2003, Zheng 2009); in this work, the scheme introduced by Anderson & Anderson (1999) has been followed. Each realization of the ensemble at the end of each update step i, $\mathbf{X}_{ij}$, is linearly inflated around its mean, $\overline{\mathbf{X}}_i$, by an inflation factor (r) slightly larger than 1:

$$\widetilde{\mathbf{X}}_{j,i+1} = r\left(\mathbf{X}_{j,i+1} - \overline{\mathbf{X}}_{i+1}\right) + \overline{\mathbf{X}}_{i+1} \tag{1.19}$$

2

Python software package

2.1. Introduction

One of the objectives of this thesis is to develop a software package for the easy application of the proposed methodology to solve inverse problems. The codes are written in Python programming language, which is one of the most popular and used programming languages of the last decades. A Python software package with a flexible workflow is presented; the codes focus on the application of the Ensemble Smoother with Multiple Data Assimilation (ES-MDA), but they can be easily adapted to any ensemble Kalman filter methods.

A tool package with various functionalities is provided in order to give the possibility to implement different configurations of the algorithm that suit the investigated problem. In particular, this package presents useful tools for the solution of inverse problems aimed at identifying time series function, which is a

novelty for these methods.

In this chapter, the Python codes are reported and described in detail. The software consists of several modules which contain different functions. The Python codes of the general part of the software package, which are not specific to the study case, is presented in the following sections. The modules that are specific to the analyzed problem and required to be modified by the user are here described in a general way; then, the Appendix provides an example of the Python codes written for solving the inverse problem aimed at the simultaneous identification of the source location and release history of a contaminant spill in an 2D aquifer.

2.2. Software package structure

The software package structure is depicted in Fig. 2.1. The ESMDA.py is the main module of the package containing the script for the implementation of ES-MDA method; the codes are general and do not depend on the study case, but they can be easily edited to adapt the software to another ensemble Kalman filter method. Mod.py and InputSettings.py are subordinate modules that depend on the investigated problem and allow to change the algorithm settings consistently with the study case. The Tools package contains several modules that provide useful utilities for the application of ensemble-based methods. All these files are located in the same working directory, which must also include a folder called "Model" that contains the forward model and its related files and some external input files. Obs.txt and Par.txt are necessary input files holding information about observations and parameters. Furthermore, three optional external input files can be present: the ens.txt file that provides an initial ensemble of parameters and the eps.txt and R.txt files that contain the measurement errors matrix and its covariance matrix, respectively.

All parts of the software package are detailed in the following sections.

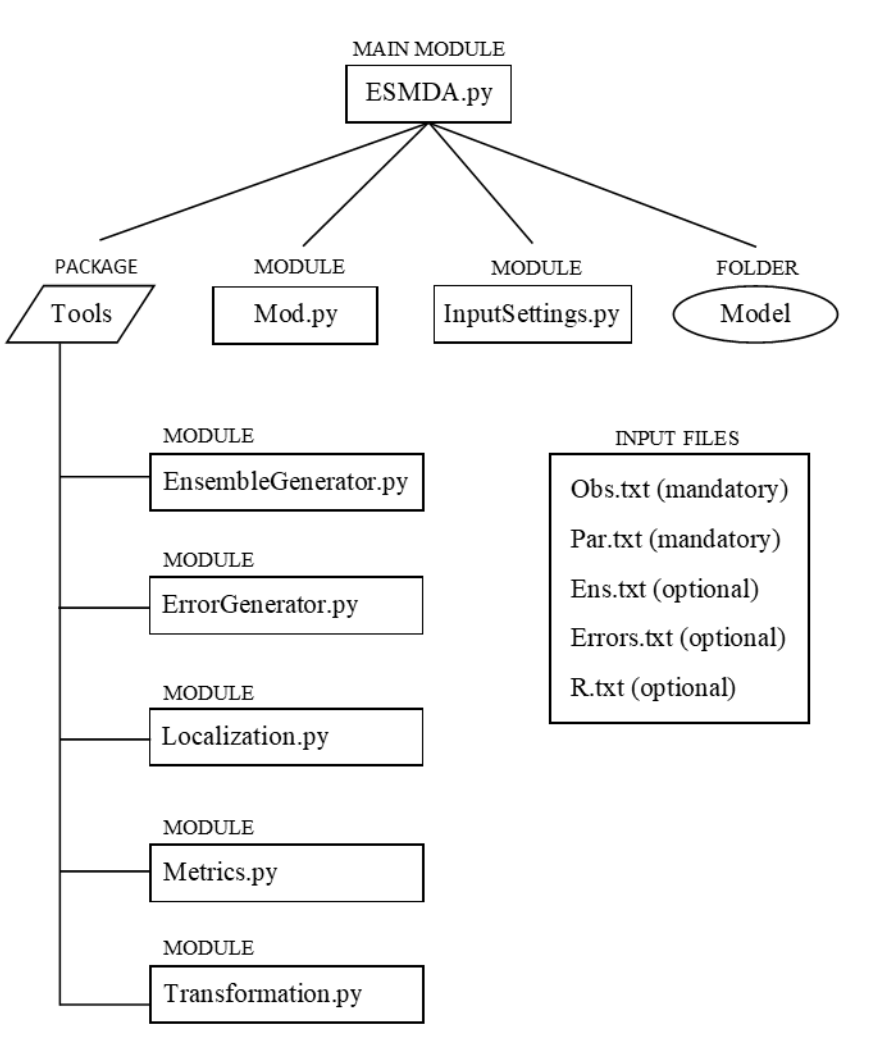

Figure 2.1. Software package structure

2.3. Input files

The input files must be present in the folder where ES-MDA is running. They are text file that can be edit with any text editor:

- **Obs.txt** (mandatory)

  The file contains the observation data with their spatial and temporal location. It must be written as follows:

  1st column: x-coordinates of the observed data
  2nd column: y-coordinates of the observed data
  3rd column: sampling times of the observed data
  4th column: values of the observed data

  If the space coordinates or the sampling time of the observations are not defined or available, the columns must be filled with NaN.

- **Par.txt** (mandatory)

  Par.txt file contains information about the parameter to be estimated; it includes actual parameters, in case the reference solution is known.

  1st column: x-coordinates of the parameters
  2nd column: y-coordinates of the parameters
  3rd column: sampling times of the parameters
  4th column: values of the actual parameters

  If the reference solution is not available, as in most cases, the 4th column must be filled with NaN. Also, if parameters do not depend on time or space the corresponding columns must be filled with NaN. It is noteworthy that the Par.txt can contains multiple parameters type (for example, spatial coordinates of a pollution source location, hydraulic conductivity and discretized time function)

- **Ens.txt** (optional) The file contains the initial ensemble of parameter realizations; each row corresponds to the realizations of one parameter. It is an optional file since the initial ensemble can be generated within the code package.

- **Errors.txt** (optional)

  The file contains the observation error ensemble ε; each row corresponds to the errors of one observation. It is an optional file since the measurement errors can be generated within the code package.

- **R.txt** (optional)

  The file contains the observation error covariance matrix; it is a diagonal matrix since the observation errors are assumed independent from one another. It is an optional file related to the previous one.

2.4. Modules

Mod.py

Mod.py is the module related to the forward model; it depends on the analyzed problem and must be changed according to each application. It must include 3 mandatory functions:

- *write_input*: it is the function that writes the parameters, unknowns of the inverse procedure, into their proper location in the input files of the forward model;

- *run*: it is the function that contains the command to run the forward model;

- *read_output*: it is the function that reads the output values from the forward model that correspond to the observation data.

An example of the Mod.py module is provided in Appendix; the codes written for the solution of the inverse problem that aims to simultaneously estimate the location and release history of a contaminant in groundwater are reported.

InputSettings.py

InputSettings.py is the module that includes all the input information, specific for the investigated inverse problem, required to perform the inverse procedure. It contains different functions, which must be modified by the user in order to adapt them to the case study:

- *Func_ens*: it is the function used to set up the the generation of the initial ensemble of parameters. The function makes use of the EnsembleGenerator.py module of the Tools package described in the next section.

- *Func_err*: it is the function used to set up the generation of the ensemble of observation errors and its covariance matrix. The function makes use of the ErrorGenerator.py module of the Tools package described in the next section.

- *forward_transf*: it is the function used to set up the parameter space transformation, if necessary during the estimation. It allows to apply a different type of transformation for each parameter. This function makes use of the Transformation.py module.

- *backward_transf*: it is the function, dependent on the previous one, used to back-transform the parameters in their physical space. Also the *backward_transf* makes use of the Transformation.py module of the Tools package.

- *localization*: it is the function used to set up a covariance localization suitable for the considered problem. It makes use of the Localization.py module of

the Tools package.

- *Metrics_ obs*: it is the function used to define the metrics for the evaluation of the method performance. *Metrics_ obs* is used to compute the metrics based on the comparison between actual and predicted values of the observations. It makes use of the Metrics.py module of the Tools package described in the next section.

- *Metrics_ obs_ par*: it is a function similar to the previous one, which is used when a reference solution is available. Performance metrics take into account both the comparison between actual and predicted observations and actual and estimated parameters. It uses the Metrics.py module.

An example of the InputSettings.py module, written for the inverse problem of the simultaneous identification of the source location and release history of a contaminant in groundwater, is provided in Appendix.

ESMDA.py

ESMDA.py is the main module of the software package; it contains the code block to perform the ensemble smoother with multiple data assimilation method. It does not require modifications by the user as it is written in a generic way and does not depend on the analyzed problem. Hereafter, the Python codes are shown in blue text.

The first part of the codes refers to the initialization step; after importing the numPy and OS Python libraries and the Mod.py module,

```
import numpy as np
import os
import Mod
```

the user is asked to choose the ensemble size, the number of iterations, the α_{geo} coefficient (Eq. 1.3) and the relaxation coefficient (Eq. 1.9). Moreover, it is asked

whether to apply the covariance inflation and if so, to choose the inflation factor (Eq. 1.19), and whether to apply the covariance localization and if so, to specify to perform it in the standard or iterative form.

```
    ens=int(input('ensemble size: '))
    maxit=int(input('number of iterations: '))
    alpha_geo=int(input('alpha_geo: '))
    w=float(input('Relaxation coefficient: '))
    inflation=(input('Do inflation? (y)yes or (n)not: '))
    if inflation=='y':
       rr=float(input('Inflation coefficient: '))
    localize=(input('Do localization? (y)yes or (n)not: '))
    if localize=='y':
        iter_loc=(input('Do iterative localization? (y)yes or (n)not: '))
    space_transform=(input('Work in transformed space:
(y)yes or (n)not: '))
```

Then, the observation information contained in the True_obs.txt file are loaded;

```
    Obs_file=np.loadtxt('Obs.txt', dtype=float)
    x_obs=Obs_file[:,0]
    y_obs=Obs_file[:,1]
    time_obs=Obs_file[:,2]
    Obs=np.atleast_2d(Obs_file[:,3]).T
    N_obs=Obs.shape[0]
```

it is asked if the reference solution is available and the parameters information contained in the Par.txt file are read;

```
    ref_solution=(input('Do you have the reference solution?\
(y)yes or (n)not: '))
    True_par_file=np.loadtxt('Par.txt', dtype=float)
    x_par=True_par_file[:,0]
    y_par=True_par_file[:,1]
    time_par=True_par_file[:,2]
    True_par=True_par_file[:,3]
```

Then, a new initial ensemble of parameters is generated or an available one is loaded from the X.txt file.

```
    new_ens=(input('Generate a new ensemble: (y)yes or (n)not '))
    if new_ens=='n':
```

```
    X=np.loadtxt('Ens.txt')
elif new_ens=='y':
    from InputSettings import Func_ens
    X=Func_ens(ens)
else:
    print('Error, invalid input')
N_par=X.shape[0]
```

The following code block generates the ensemble of measurements errors and its covariance matrix or upload this data from eps.txt and R.txt files.

```
new_err=(input('Generate new random errors: (y)yes or (n)not '))
if new_err=='n':
    R=np.loadtxt('R.txt')
    eps=np.loadtxt('Errors.txt')
elif new_err=='y':
    from InputSettings import Func_err
    eps,R=Func_err(N_obs, ens)
else:
    print('Error, invalid input')
```

The next step of the initialization phase is the definition of the coefficients α_i for each iteration; the scheme of Eqs 1.3 and 1.4 is followed, which ensure that the condition of Eq. 1.2 is satisfied.

```
al_i = np.ones((maxit, 1),float)
for i in range(1,maxit):
    al_i[i]=al_i[i-1]/alpha_geo
sum_al_i=sum(1./al_i)
alpha=al_i*sum_al_i
sum_alpha=sum(1./alpha)
```

If the covariance localization is not performed iteratively, the correlation matrices based on spatial or temporal distances between parameters and observations and between observations and observations are computed before the iterative steps.

```
if localize=='y':
    from InputSettings import localization
if localize=='y' and iter_loc=='n':
    (rho_yy,rho_xy,rho_xx)=localization(X,True_par_file[:,0:3],
                                        Obs_file[:,0:3],iter_loc)
```

Then, the iterative process starts. At the beginning of each iteration, the correlation matrices to apply the iterative localization are computed.

```
r=[]
pred=np.zeros((N_obs,ens))
Xprev=np.copy(X)
for i in range(0,maxit):
    R_corr=alpha[i]*R
    r.append(R_corr[i,i])
    if localize=='y' and iter_loc=='y':
        (rho_yy,rho_xy,rho_xx)=localization(Xprev,
                                            True_par_file[:,0:3],
                                            Obs_file[:,0:3],iter_loc)
```

The following code block describes the forecast step (Eq. 1.1).

```
os.chdir('Model')
for j in range(0,ens):
    Mod.write_input(Xprev[:,j])
    Mod.run()  #run forward model
    pred[:,j]=Mod.read_output()
os.chdir('..')
```

Next, the vector of parameters is transformed before the update step, if necessary.

```
if space_transform=='y':
    from InputSettings import forward_transf
    Xprev=forward_transf(Xprev)
```

In the following code block, the update step is performed. The covariance matrices are computed from the ensemble (Eqs. 1.7 and 1.8) and, if required, the covariance localization is applied.

```
xm=np.atleast_2d(Xprev.mean(1)).T
ym=np.atleast_2d(pred.mean(1)).T
Qx=Xprev-xm*np.ones((1,ens))
Qy=pred-ym*np.ones((1,ens))
Qxy=Qx@Qy.T/(ens-1)
Qyy=Qy@Qy.T/(ens-1)
Qxx=Qx@Qx.T/(ens-1)
```

```
if localize=='y':
    Qxy=rho_xy*Qxy
    Qyy=rho_yy*Qyy
    Qxx=rho_xx*Qxx
```

The parameters are updated based on the Kalman gain matrix and the misfit between observations and corresponding model predictions.

```
Gain=Qxy @ np.linalg.inv(Qyy+R_corr)
Xnew=Xprev+Gain@(Obs@np.ones((1,ens))+(alpha[i])**(1/2)*eps-pred)
```

Then, the linear relaxation is applied (Eq 1.9), if the relaxation coefficient w is nonzero.

```
Xnew=(1-w)*Xnew+w*Xprev
```

After the update, the parameters are back-transformed to their physical space, if the transformation was performed.

```
if space_transform=='y':
    from InputSettings import backward_transf
    Xnew=backward_transf(Xnew)
    Xprev=backward_transf(Xprev)
```

The covariance inflation is applied (Eq. 1.19), if required.

```
if inflation=='y':
     Xnew_mean=np.atleast_2d(np.mean(Xnew,axis=1)).T
     Xnew=Xnew_mean*np.ones((1,ens))+rr\
          (Xnew-Xnew_mean*np.ones((1,ens)));
```

At the end of each iteration, the performance metrics are computed and collected in a dictionary. Then the process repeats until the last iteration.

```
Xprev=np.copy(Xnew)
Xp=np.mean(Xprev,axis=1)
pred_mean=np.mean(pred,axis=1)

if ref_solution=='y':
     from InputSettings import Metrics_obs_par
     metrics_iter=Metrics_obs_par(Xprev,pred,True_par,Obs)
else:
```

```
        from InputSettings import Metrics_obs
        metrics_iter=Metrics_obs(Xprev,pred,True_par,Obs)

    if i==0:
        metrics_name=list(metrics_iter.keys())
        metrics_dict=metrics_iter.copy()
    else:
        for m in metrics_name:
            metrics_dict[m]=metrics_dict[m]+metrics_iter[m]
```

2.5. Tools package

The tools package includes different modules that provide the instrument to build the initial ensemble of parameters, generate the observation errors, apply the localization, perform the transformation of the parameter space and calculate the evaluation metrics.

EnsembleGenerator.py

The EnsembleGenerator.py module contains several functions that allow to generate the initial realizations of parameters in different ways consistently with the type of analyzed problem. All the functions return the initial ensemble matrix of dimensions $(N_p \times N_e)$ as output, where N_p and N_e are the number of parameters and the number of realizations, respectively. The numeric python library NumPy is used and imported at the beginning of EnsembleGenerator.py as:

```
import numpy as np
```

The available functions to generate the initial ensemble are:

- *Random*: the realizations of the parameters are uniformly distributed random values selected over a range of values. The function requires the following input arguments: the boundary values of the range, the number of

parameters and the number of ensemble realizations.

```
def Random(Min,Max,N_par,ens):
    X=np.random.uniform(low=Min, high=Max, size=(N_par,ens))
    return X
```

- *PdfGamma*: each realization of the parameters follow a gamma distribution; it is defined as:

$$f(t) = A + \frac{1}{k^n \Gamma(n)} t^{n-1} e^{-t/k}, \tag{2.1}$$

where t is time (or space), A represents a base amount of the considered variable, B is the volume under the Gamma function, n is the shape coefficient, k the scale coefficient and $\Gamma(n)$ is the gamma function. The coefficients are generated randomly from uniform distributions over ranges of values. The function requires the following input arguments: the range limit values for coefficients A, B, n and k, the vector of the variable t, the number of parameters and the number of ensemble realizations.

```
def PdfGamma(aMin,aMax,kMin,kMax,nMax,nMin,
             bMin,bMax,x,N_par,ens):
    from scipy.stats import gamma
    a=np.random.uniform(low=aMin, high=aMax, size=(ens))
    k=np.random.uniform(low=kMin, high=kMax, size=(ens))
    n=np.random.uniform(low=nMin, high=nMax, size=(ens))
    b=np.random.uniform(low=bMin, high=bMax, size=(ens))
    X=np.zeros((N_par,ens))
    for i in range(0,ens):
        X[:,i]=gamma.pdf(x, n[i], loc=0, scale=k[i])*b[i]+a[i]
    return X
```

- *PdfGammaNpeaks*: each realization of the parameters is given by the summation of gamma functions:

$$f(t) = A + \sum_{r=1}^{M} B_r \cdot \frac{1}{k_r^{n_r} \Gamma(n_r)} t^{n_r - 1} e^{-t/k_r}, \tag{2.2}$$

where M denotes the number of summed gamma functions and the other coefficients are the same as those defined for the function *PdfGamma*; each gamma function is generated using coefficients selected randomly over the same ranges of values. The function requires the following input arguments: the number of Gamma functions M, the range limit values for coefficients A, B, n and k, the vector of the variable t, the number of parameters and the number of ensemble realizations.

```
def PdfGammaNpeaks(aMin,aMax,kMin,kMax,nMax,nMin,
                   bMin,bMax,x,N_par,ens,N_peaks):
    from scipy.stats import gamma
    a=np.zeros((ens,N_peaks))
    k=np.zeros((ens,N_peaks))
    n=np.zeros((ens,N_peaks))
    b=np.zeros((ens,N_peaks))
    for i in range(0,N_peaks):
        a[:,i]=np.random.uniform(low=aMin, high=aMax, size=(ens))
        k[:,i]=np.random.uniform(low=kMin, high=kMax, size=(ens))
        n[:,i]=np.random.uniform(low=nMin, high=nMax, size=(ens))
        b[:,i]=np.random.uniform(low=bMin, high=bMax, size=(ens))
    X=np.zeros((N_par,ens))
    for j in range(0,ens):
        for i in range(0,N_peaks):
        X[:,j]+=gamma.pdf(x, n[j,i], loc=0, scale=k[j,i])*\
                b[j,i]+a[j,i]
    return X
```

- *PdfNormal*: The realizations of parameters are Gaussian functions described by the following expression:

$$f(t) = A + B \cdot \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)} \tag{2.3}$$

where t is time (or space), A is a base amount of the considered variable, B is the volume under the Gaussian function and μ and σ are the mean and variance used to define it. These coefficients are selected randomly, for each

realization of the ensemble, from a uniform distribution over fixed ranges. The function requires the following input arguments: the range limit values for coefficients A, B, μ and σ, the vector of the variable t, the number of parameters and the number of ensemble realizations.

```
def PdfNormal(aMin,aMax,muMin,muMax,sigmaMin,sigmaMax,
              bMin,bMax,x,N_par,ens):
    from scipy.stats import norm
    a=np.random.uniform(low=aMin, high=aMax, size=(ens))
    mu=np.random.uniform(low=muMin, high=muMax, size=(ens))
    sigma=np.random.uniform(low=sigmaMin, high=sigmaMax, size=(ens))
    b=np.random.uniform(low=bMin, high=bMax, size=(ens))
    X=np.zeros((N_par,ens))
    for i in range(0,ens):
        X[:,i]=norm.pdf(x, mu[i], sigma[i])*b[i]+a[i]
    return X
```

- *ConstantRandom*: each ensemble realization of parameters is made of a unique constant value, which is different for each realization and generated randomly from a uniform distributions over a fixed range. The function requires the following input arguments: the limit values of the coefficients range, the number of parameters and the number of ensemble realizations.

```
def ConstantRandom(Min,Max,N_par,ens):
    values=np.random.uniform(low=Min, high=Max, size=(ens))
    values=np.atleast_2d(values)
    vector=np.ones([N_par])
    vector=np.atleast_2d(vector).T
    X=vector@values
    return X
```

- *ConstantNormal*: The realizations of parameters are constant values, which are different for each realization and generated randomly from a Gaussian distribution. The function requires the following input arguments: the mean and the variance of the Gaussian function, the number of parameters and the number of ensemble realizations.

```
def ConstantNormal(mu,sigma,N_par,ens):
    values=np.random.normal(mu, sigma, size=(ens))
    values=np.atleast_2d(values)
    vector=np.ones([N_par])
    vector=np.atleast_2d(vector).T
    X=vector@values
    return X
```

The use of the functions *PdfGamma*, *PdfGammaNpeaks* and *PdfNormal* functions is suggested when the parameters to be estimated are discretized time series.

ErrorGenerator.py

ErrorGenerator.py is the module that deals with the observation errors. It contains two functions that returns as output the ensemble of observed data errors, which is a matrix of dimensions $(m \times N_e)$, and the covariance matrix of the observation errors of dimensions $(m \times m)$; where m denotes the number of observations. The NumPy library is imported at the beginning of the module as:

```
import numpy as np
```

The two available functions are:

- *NormalError*: it generates random observation errors with normal distribution, zero mean and fixed variance. The function requires the following input arguments: the variance, the number of parameters and the number of ensemble realizations.

```
def NormalError(var_err,N_obs,ens):
    eps=np.zeros((N_obs,ens))
    R=np.identity(N_obs)*var_err
    for i in range(N_obs):
        eps[i,:]=np.random.normal(0, var_err, size=(ens))
    return (eps,R)
```

- *PercNormalError*: it generates observation errors expressed as a percentage of the measurement data. The errors of each observation are normally

distributed with zero mean and variances defined so that the 99.7% of the errors lies within the range defined by the selected percentage of the observed value. It is possible to set a minimum variance threshold in order to avoid too small values given by the percentage of small observed data. The function requires the following input arguments: the vector of the observations, the percentage of the observed data, the lower limit of the variances, the number of parameters and the number of ensemble realizations.

```
def PercNormalError(obs,perc,var_err_lim,N_obs,ens):
    var_err=(perc/100*obs/3)**2
    var_err[var_err < var_err_lim]=var_err_lim
    R=np.diag(var_err)
    eps=np.zeros((N_obs,ens))
        for i in range(N_obs):
    eps[i,:]=np.random.normal(0, var_err[i], size=(ens))
    return (eps,R)
```

Localization.py

Localization.py is the module used to apply the covariance localization. It contains two functions that return as output the correlation matrices, computed following Eq. 1.18, based on spatial or temporal observations-observations, parameters-observations and parameters-parameters distances. The NumPy is imported at the beginning of Localization.py module and renamed as "np":

```
import numpy as np
```

The two functions are:

- *TimeLocal*: it generates the correlation matrices based on time distances. The functions requires the following input arguments: the correlation time length characterizing the time distance at which the covariances become zero (coefficient b in Eq. 1.18), the discrete time vector of parameters and the discrete time vector of observations.

```
def TimeLocal(a,time_par,time_obs):
    N_par=time_par.shape[0]
    N_obs=time_obs.shape[0]
    d_yy=np.zeros((N_obs,N_obs))
    d_xy=np.zeros((N_par,N_obs))
    d_xx=np.zeros((N_par,N_par))
    rho_yy=d_yy;
    rho_xy=d_xy;
    rho_xx=d_xx;
    for row in range(0,d_yy.shape[0]):
        for col in range(0,d_yy.shape[1]):
            d_yy[row,col]=abs(time_obs[row]-time_obs[col])
            if d_yy[row,col]<=a:
                rho_yy[row,col]=-1/4*(abs(d_yy[row,col])/a)**5+\
                                1/2*(abs(d_yy[row,col])/a)**4+\
                                5/8*(abs(d_yy[row,col])/a)**3-\
                                5/3*(abs(d_yy[row,col])/a)**2+\
                                1
            elif d_yy[row,col]>a  and d_yy[row,col]<=2*a:
                rho_yy[row,col]=1/12*(abs(d_yy[row,col])/a)**5-\
                                1/2*(abs(d_yy[row,col])/a)**4+\
                                5/8*(abs(d_yy[row,col])/a)**3+\
                                5/3*(abs(d_yy[row,col])/a)**2-\
                                5*(abs(d_yy[row,col])/a)+\
                                4-\
                                2/3*a/abs(d_yy[row,col])
            elif d_yy[row,col]>2*a:
                rho_yy[row,col]=0
    for row in range(0,d_xy.shape[0]):
        for col in range(0,d_xy.shape[1]):
            d_xy[row,col]=abs(time_par[row]-time_obs[col])
            if d_xy[row,col]<=a:
                rho_xy[row,col]=-1/4*(abs(d_xy[row,col])/a)**5+\
                                1/2*(abs(d_xy[row,col])/a)**4+\
                                5/8*(abs(d_xy[row,col])/a)**3-\
                                5/3*(abs(d_xy[row,col])/a)**2+\
                                1
            elif d_xy[row,col]>a  and d_xy[row,col]<=2*a:
                rho_xy[row,col]=1/12*(abs(d_xy[row,col])/a)**5-\
                                1/2*(abs(d_xy[row,col])/a)**4+\
                                5/8*(abs(d_xy[row,col])/a)**3+\
```

```
                                    5/3*(abs(d_xy[row,col])/a)**2-\
                                    5*(abs(d_xy[row,col])/a)+\
                                    4-\
                                    2/3*a/abs(d_xy[row,col])
            elif d_xy[row,col]>2*a:
                rho_xy[row,col]=0
    for row in range(0,d_xx.shape[0]):
        for col in range(0,d_xx.shape[1]):
            d_xx[row,col]=abs(time_par[row]-time_par[col])
            if d_xx[row,col]<=a:
                rho_xx[row,col]=-1/4*(abs(d_xx[row,col])/a)**5+\
                                 1/2*(abs(d_xx[row,col])/a)**4+\
                                 5/8*(abs(d_xx[row,col])/a)**3-\
                                 5/3*(abs(d_xx[row,col])/a)**2+\
                                 1
            elif d_xx[row,col]>a  and d_xx[row,col]<=2*a:
                rho_xx[row,col]=1/12*(abs(d_xx[row,col])/a)**5-\
                                1/2*(abs(d_xx[row,col])/a)**4+\
                                5/8*(abs(d_xx[row,col])/a)**3+\
                                5/3*(abs(d_xx[row,col])/a)**2-\
                                5*(abs(d_xx[row,col])/a)+\
                                4-\
                                2/3*a/abs(d_xx[row,col])
            elif d_xx[row,col]>2*a:
                rho_xx[row,col]=0
    return (rho_yy,rho_xy,rho_xx)
```

- *SpaceLocal*: it generates the correlation matrices based on spatial distances. The functions requires the following input arguments: the correlation length characterizing the spatial distance at which the covariances become zero (coefficient b in Eq. 1.18), the location of parameters and the location of observations.

```
def SpaceLocal(a,pos_par,pos_obs):
    x_par=pos_par[:,0]
    y_par=pos_par[:,1]
    x_obs=pos_obs[:,0]
    y_obs=pos_obs[:,1]
    N_par=pos_par.shape[0]
```

```
N_obs=pos_obs.shape[0]
d_yy=np.zeros((N_obs,N_obs))
d_xy=np.zeros((N_par,N_obs))
d_xx=np.zeros((N_par,N_par))
rho_yy=d_yy;
rho_xy=d_xy;
rho_xx=d_xx;
for row in range(0,d_yy.shape[0]):
    for col in range(0,d_yy.shape[1]):
        d_yy[row,col]=np.sqrt((x_obs[row]-x_obs[col])**2+\
                              (y_obs[row]-y_obs[col])**2)
        if d_yy[row,col]<=a:
            rho_yy[row,col]=-1/4*(abs(d_yy[row,col])/a)**5+\
                            1/2*(abs(d_yy[row,col])/a)**4+\
                            5/8*(abs(d_yy[row,col])/a)**3-\
                            5/3*(abs(d_yy[row,col])/a)**2+\
                            1
        elif d_yy[row,col]>a  and d_yy[row,col]<=2*a:
            rho_yy[row,col]=1/12*(abs(d_yy[row,col])/a)**5-\
                            1/2*(abs(d_yy[row,col])/a)**4+\
                            5/8*(abs(d_yy[row,col])/a)**3+\
                            5/3*(abs(d_yy[row,col])/a)**2-\
                            5*(abs(d_yy[row,col])/a)+4-\
                            2/3*a/abs(d_yy[row,col])
        elif d_yy[row,col]>2*a:
            rho_yy[row,col]=0
for row in range(0,d_xy.shape[0]):
    for col in range(0,d_xy.shape[1]):
        d_xy[row,col]=np.sqrt((x_par[row]-x_obs[col])**2+\
                              (y_par[row]-y_obs[col])**2)
        if d_xy[row,col]<=a:
            rho_xy[row,col]=-1/4*(abs(d_xy[row,col])/a)**5+\
                            1/2*(abs(d_xy[row,col])/a)**4+\
                            5/8*(abs(d_xy[row,col])/a)**3-\
                            5/3*(abs(d_xy[row,col])/a)**2+\
                            1
        elif d_xy[row,col]>a  and d_xy[row,col]<=2*a:
            rho_xy[row,col]=1/12*(abs(d_xy[row,col])/a)**5-\
                            1/2*(abs(d_xy[row,col])/a)**4+\
                            5/8*(abs(d_xy[row,col])/a)**3+\
                            5/3*(abs(d_xy[row,col])/a)**2-\
```

```
                                5*(abs(d_xy[row,col])/a)+4-\
                                2/3*a/abs(d_xy[row,col])
            elif d_xy[row,col]>2*a:
                rho_xy[row,col]=0
    for row in range(0,d_xx.shape[0]):
        for col in range(0,d_xx.shape[1]):
            np.sqrt((x_par[row]-x_par[col])**2+\
                    (y_par[row]-y_par[col])**2)
            if d_xx[row,col]<=a:
                rho_xx[row,col]=-1/4*(abs(d_xx[row,col])/a)**5+\
                                1/2*(abs(d_xx[row,col])/a)**4+\
                                5/8*(abs(d_xx[row,col])/a)**3-\
                                5/3*(abs(d_xx[row,col])/a)**2+\
                                1
            elif d_xx[row,col]>a  and d_xx[row,col]<=2*a:
                rho_xx[row,col]=1/12*(abs(d_xx[row,col])/a)**5-\
                                1/2*(abs(d_xx[row,col])/a)**4+\
                                5/8*(abs(d_xx[row,col])/a)**3+\
                                5/3*(abs(d_xx[row,col])/a)**2-\
                                5*(abs(d_xx[row,col])/a)+4-\
                                2/3*a/abs(d_xx[row,col])
            elif d_xx[row,col]>2*a:
                rho_xx[row,col]=0
    return (rho_yy,rho_xy,rho_xx)
```

Transformation.py

Transformation.py is the module used to perform the transformation and back-transformation of the parameters space in different ways. The NumPy is imported at the beginning of the module:

```
import numpy as np
```

Transformation.py contains the following functions:

- *Log_forward*: the parameters are log transformed. The function requires the vector of parameters in their physical space as input and returns it in the transformed space.

```
def Log_forward(xx):
    X_t=np.log(xx)
    return X_t
```

- *Log_backward*: the parameters are back transformed from the log space. The function requires the vector of parameters in the transformed space as input and returns it in their physical space.

```
def Log_backward(xx):
    X_bt=np.exp(xx)
    return X_bt
```

- *LogLim_forward*: the parameters are log transformed in a bounded space following the modified log-transformation (Eq. 1.10). The function requires the vector of parameters in their physical space and the bounded space interval as input and returns the parameters in the transformed space.

```
def LogLim_forward(xx,Xmin,Xmax):
    X_t=np.log((xx-Xmin)/(Xmax-xx))
    return X_t
```

- *LogLim_backward*: the parameters are back transformed from the modified log space (Eq. 1.11). The function requires the vector of parameters in the transformed space and the bounded space intervals as input and returns it in their physical space.

```
def LogLim_backward(xx,Xmin,Xmax):
    X_bt=(np.exp(xx)*Xmax+Xmin)/(1+np.exp(xx))
    return X_bt
```

- *SquareRoot_forward*: the parameters are transformed using the square root transformation. The function requires the vector of parameters in their physical space as input and returns it in the transformed space.

```
def SquareRoot_forward(xx):
    X_t=xx**(1/2)
    return X_t
```

- *SquareRoot_backward*: the parameters are back transformed from square root space. The function requires the vector of parameters in the transformed space as input and returns it in their physical space.

```
def SquareRoot_backward(xx):
    X_bt=xx**(2)
    return X_bt
```

- *SquareRootLim_forward*: the parameters are transformed using the square root transformation in a bounded space (Eq. 1.12). The function requires the vector of parameters in their physical space and the bounded space interval as input and returns the parameters in the transformed space.

```
def SquareRootLim_forward(xx,Xmin,Xmax):
    X_t=((xx-Xmin)/(Xmax-xx))**(1/2)
    return X_t
```

- *SquareRootLim_backward*: the parameters are back transformed from the modified square root space (Eq. 1.13). The function requires the vector of parameters in the transformed space and the bounded space intervals as input and returns it in their physical space.

```
def SquareRootLim_backward(xx,Xmin,Xmax):
    X_t=(Xmax-Xmin)*xx**2/(1+xx**2)+Xmin
    return X_t
```

Metrics.py

This module contains the functions that are used to compute the metrics for the evaluation methodology performance. The numpy and math libraries are used, which are called at the beginning of the codes as:

```
import math
import numpy as np
```

The metrics.py module contains five metrics:

- *RMSE*: it is the function that calculates the root mean square error between the actual and predicted values, which can be parameters or observations.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(\overline{S}_i - O_i\right)^2}{n}}, \tag{2.4}$$

where n is the sample size, $\mathbf{O}_i$ is the i-th actual value and $\overline{\mathbf{S}}_i$ is the ensemble mean of the i-th estimated value. The function requires the vector of the actual values and the ensemble mean of the predicted ones as input.

```
def RMSE(actual, predicted):
    N=actual.shape[0]
    rmse=math.sqrt(np.sum((predicted-actual)**2)/N)
    return rmse
```

- *AES*: it is the function that calculates the average ensemble spread; it is defined as:

$$AES = \sqrt{\frac{\sum_{i=1}^{n} \sigma_i^2}{n}}, \tag{2.5}$$

where n is the sample size of parameters or observations and σ_i^2 is the ensemble variance of the i-th value. The function requires the ensemble of parameters or predicted values of observations as input.

```
def AES(ensemble):
    N=ensemble.shape[0]
    aes=math.sqrt(np.sum(np.var(ensemble,axis=1))/N)
    return aes
```

- *NSE*: it is the function that calculates Nash-Sutcliffe efficiency criterion defined as

$$NSE = \left(1 - \frac{\sum_{i=1}^{n} \left(\overline{S}_i - O_i\right)^2}{\sum_{i=1}^{n} \left(O_i - \overline{O}\right)^2}\right) \cdot 100, \tag{2.6}$$

where n is the sample size, $\overline{S}_i$ is the ensemble mean of the i-th estimated

value, O_i is the i-th actual value and $\overline{O}$ is the average of the actual values. The function requires the vector of the actual data and the ensemble mean of predicted values as input.

```
def NSE(actual, predicted):
    nse=(1-(np.sum((predicted-actual)**2))/\
          (np.sum((actual-np.mean(actual))**2)))*100
    return nse
```

- *RSS*: it is the function that calculates the residual sum of squares between actual and estimated data.

$$RSS = \sum_{i=1}^{n} \left(\overline{S}_i - O_i\right)^2, \tag{2.7}$$

where n is the sample size, $\overline{S}_i$ is the ensemble mean of the i-th estimated value and O_i is the i-th actual value. The function requires the vector of the actual values and the ensemble mean of the predicted ones as input.

```
def RSE(actual, predicted):
    rse=sum((predicted-actual)**2)
    return rse
```

- *spatial_distance*: it is the function that calculates the spatial distance between actual and estimated locations. It can be used when spatial coordinates are parameters to be estimated.

$$L = \sqrt{\left(\overline{x}_s - x_0\right)^2 + \left(\overline{y}_s - y_0\right)^2} \tag{2.8}$$

where $\overline{x}_s$ and $\overline{y}_s$ are the ensemble means of the estimated spatial coordinates of the parameter and (x_0, y_0) is the actual location. The function requires the vector of the actual locations and the ensemble mean of the predicted ones as input.

```
def spatial_distance(actual, prediction):
    dd=math.sqrt(np.sum(prediction-actual)**2)
    return dd
```

3

Reverse flow routing

3.1. Introduction

The knowledge of discharge hydrographs at specific river sections is essential for flood-risk assessment, planning and management of water resource systems, or optimization of existing hydraulic infrastructures and design of new ones, among others. However, only few river sections are equipped to record data; therefore, an indirect determination of discharge hydrographs is often required. When a flood wave propagates along a river reach or passes through a reservoir, it usually experiences a delay and an attenuation. Although the forward flow routing (estimation of downstream discharge hydrographs based on information available upstream) is common and widely used by practitioners, the estimation of discharge hydrographs at ungauged sections that do not have reliable data upstream is still challenging. Discarding the use of rainfall-runoff models, due to their high uncertainty, a tech-

nique that could overcome this problem is the reverse flow routing process that couples the information recorded downstream (discharges or water levels) and the channel or reservoir characteristics to estimate the upstream inflow. The two main approaches to solve this problem in open channels are the application of hydrological routing models (see e.g. Das 2009, Koussis & Mazi 2016) in a reverse form, and the backward solution in time of the de Saint Venant equations (see e.g. Eli et al. 1974, Szymkiewicz 1993, Bruen & Dooge 2007). A more recent approach makes use of optimization procedures to determine the hydrograph that, once propagated downstream, reproduces the available observations. Saghafian et al. (2014) and Zucco et al. (2015) coupled a Genetic algorithm with a one-dimensional forward hydraulic model and with a simplified routing model, respectively. D'Oria & Tanda (2012), D'Oria et al. (2014) and Ferrari et al. (2018) applied a Bayesian Geostatistical Approach (BGA) to perform the reverse flow routing in combination with hydraulic models that solve the one-dimensional or two-dimensional shallow water equations. Zoppou (1999) faced the problem of reverse routing of flood hydrographs in reservoirs inverting a simple storage equation under a level pool approximation. Spurious oscillations arise in some circumstances; D'Oria et al. (2012) and Leonhardt et al. (2014) solved this problem applying a stochastic approach based on BGA.

Here, reverse flow routing problem is solved by means of the ensemble smoother with multiple data assimilation (ES-MDA). The objective is the estimation of an unknown inflow hydrograph discretized in time by coupling ES-MDA with a given forward routing model that relates inflow hydrograph and downstream observations. Two realistic synthetic examples are presented to show the capabilities of the methodology. The first case is an application of the reverse flow routing problem to a linear reservoir, where the outflow hydrograph and the reservoir characteristics are known; the second one focuses on the estimation of the inflow hydrograph to an open channel from water level information recorded downstream. For the second

problem, which is nonlinear, the impact of the ensemble size and covariance localization and inflation techniques are also tested. Then, the ES-MDA was applied for the solution of a real case study; an inflow hydrograph in the Parma–Baganza river confluence at the city of Parma (Italy), was estimated on the basis of water levels information collected downstream on the main reach.

This Chapter is derived in part from Todaro et al. (2019).

3.2. Synthetic examples

For the synthetic examples, the inflow hydrograph I, to be estimated is a multi-peak wave modeled as the summation of M gamma functions, that is:

$$I(t) = A + \sum_{r=1}^{M} B_r \cdot f_r \left(t \mid n_r, k_r\right), \tag{3.1}$$

where t is the time, A [L^3T^{-1}] represents the base flow, B [L^3] the flood volume of each gamma wave r, and f [T] is a gamma distribution function of coefficients n (shape) and k (scale):

$$f(t \mid n, k) = \frac{1}{k^n \Gamma\left(n\right)} t^{n-1} e^{-t/k}, \tag{3.2}$$

where $\Gamma\left(n\right)$ is the gamma function.

The synthetic test cases allow the comparison between the results of the inverse algorithm and the reference solution. The performance of the methodology is evaluated using three different metrics: the root mean square error ($RMSE$), the Nash-Sutcliffe efficiency criterion (NSE; Nash & Sutcliffe 1970) and the relative

error in the peak discharge (E_p). $RMSE$ is computed as:

$$RMSE = \sqrt{\frac{\sum_{d=1}^{N_p} \left(\mathbf{I}_d - \overline{\mathbf{X}}_d\right)^2}{N_p}}, \tag{3.3}$$

where N_p is the number of parameters, $\mathbf{I}_d$ is the d-th true inflow discharge and $\overline{\mathbf{X}}_d$ is the ensemble mean of the d-th estimated inflow discharge.

NSE is defined as:

$$NSE = \left(1 - \frac{\sum_{d=1}^{N_p} \left(\mathbf{I}_d - \overline{\mathbf{X}}_d\right)^2}{\sum_{d=1}^{N_p} \left(\mathbf{I}_d - \bar{\mathbf{I}}_d\right)^2}\right) \cdot 100, \tag{3.4}$$

where $\mathbf{I}_d$ is the mean of the true inflow hydrograph. NSE=100% indicates a perfect match between estimated and actual discharges.

E_p is evaluated as:

$$E_p = \left(\frac{\mathbf{I}_p}{\overline{\mathbf{X}}_p} - 1\right) \cdot 100 \tag{3.5}$$

where $\mathbf{I}_p$ and $\overline{\mathbf{X}}_p$ represent the true and estimated (ensemble mean) peaks of the inflow hydrographs, respectively.

The results of the second synthetic example are also compared with those obtained applying the Bayesian Geostatistical Approach (BGA) proposed by D'Oria & Tanda (2012). BGA needs multiple iterations to reach an optimal solution due to the nonlinearity of the forward problem and the need to estimate the hyperparameters of the prior covariance model, which control the structure of the unknown hydrograph, in addition to the discharge values (parameters). At each inner linearization iteration, the Jacobian matrix (sensitivity of observations to unknown parameters) must be calculated and it requires, in a finite difference approximation, as many forward model runs as the number of parameters, N_p, plus

1. Therefore, the total number of forward model runs, N_t, required by BGA is:

$$N_t = (N_p - 1)\, N_o N_i + 1, \tag{3.6}$$

where N_o and N_i are the numbers of BGA iterations needed for hyperparameters (outer loop) and parameters estimation (inner loop), respectively.

3.2.1. Reverse flow routing for a linear reservoir

The test aims to estimate the inflow hydrograph to a reservoir based on the knowledge of the outflow hydrograph and the reservoir characteristics.

Under the level pool routing approximation (reservoir dynamics are negligible and water surface inside the reservoir is horizontal), the inflow $I(t)$ and the outflow $Q(t)$ in a reservoir are related by a simple continuity equation:

$$I(t) - Q(t) = \frac{dS}{dt}, \tag{3.7}$$

where S is the instantaneous volume stored in the reservoir and t is the time. The outflow discharge is related to the storage; for a linear reservoir it can be expressed as:

$$S(t) = KQ(t), \tag{3.8}$$

where the constant proportionality factor K[T] is known as the storage coefficient.

The solution of the continuity equation for the linear reservoir (starting from a steady state condition) on a continuous time scale is represented by the following convolution integral (Chow 1988):

$$Q(t) = \int_0^t \frac{1}{K} e^{-(t-\tau)/K} I(\tau) d\tau. \tag{3.9}$$

A solution at discrete intervals of time can be obtained by means of a discrete

convolution equation.

The synthetic test considers a reservoir with storage coefficient K=3 h and an inflow hydrograph with two peaks as defined by Eq. 3.1 (M=2) and the coefficients reported in Table 1. The resulting hydrograph has a first peak of about 500 m^3/s at 3.5 h and a second peak with a discharge of about 240 m^3/s at 11.4 h.

Table 3.1. Case 1: coefficients of the two gamma functions used for the description of the inflow hydrograph.

A [m^3/s]		**B [m^3]**	**n [-]**	**k [h]**
50	f_1	$5.5 \cdot 10^6$	8	0.5
	f_2	$4.5 \cdot 10^6$	20	0.6

The total simulation time is 30 h. The inflow hydrograph is discretized in equal interval of 9 min resulting in a number of parameters to be estimated N_p=201.

Preliminarily, the actual inflow hydrograph is forward routed through Eq. 3.9 to obtain the true outflow hydrograph; this last one was observed every 6 min for a total of 301 observations (m=301) to be used in the inverse procedure. In applying the ES-MDA procedure, it is considered an observation error ε equal to 5% of the true discharge values.

The initial ensemble (Fig. 3.1) is composed of 200 realizations of the inflow hydrograph. They are all individual gamma functions generated using Eq. 3.1 with M=1 and the other coefficients selected randomly over a wide range of values. In particular, the range is [10, 150] m^3/s for A, [$1.5 \cdot 10^5$, $5.0 \cdot 10^7$] m^3 for B, [3, 10] for n and [0.7, 4.5] h for k; the extremes of the ranges, selected on the basis of expert knowledge, guarantee that all the realizations are consistent with the considered problem.

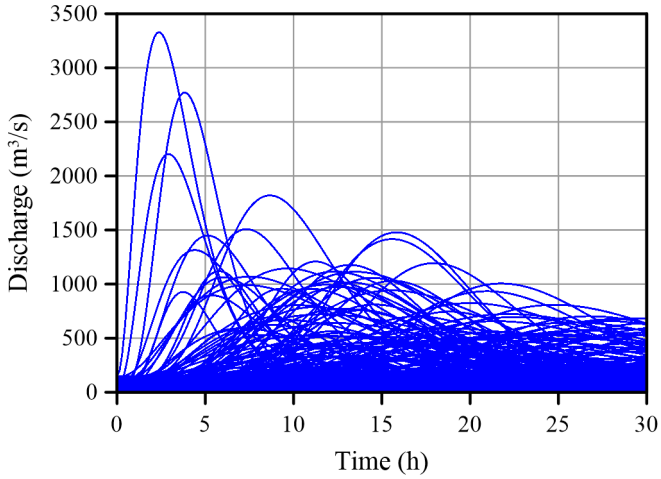

Figure 3.1. Case 1: initial ensemble of inflow hydrograph (200 realizations).

For the ES-MDA, 5 iterations with a constant α equal to 5 (α_{geo}=1, Eqs. 1.3 and 1.4) are performed. In this case, no localization or inflation are applied, and a large ensemble is considered with the aim to show the capability of the method.

Fig. 3.2 presents the results of the inversion at the end of the iterative process: it shows the ensemble mean of the estimated inflow and outflow hydrographs with their 95% confidence interval computed from the ensemble; the actual inflow and outflow hydrographs are reported for comparison.

The ES-MDA method accurately reproduces the inflow hydrograph (NSE= 99.94%) with a very narrow confidence interval, as well as the simulated outflow hydrograph. The $RMSE$ at each iteration, shown in Fig. 3.3, slightly decreases during the procedure reaching the lowest value of 2.9 m^3/s at the end of the simulation. The two inflow peaks and their timing are properly reproduced with a slight underestimation ($E_{p,1}$=-1.1%; $E_{p,2}$=-0.4%, where the subscript 1 stands for the first peak and 2 for the second one).

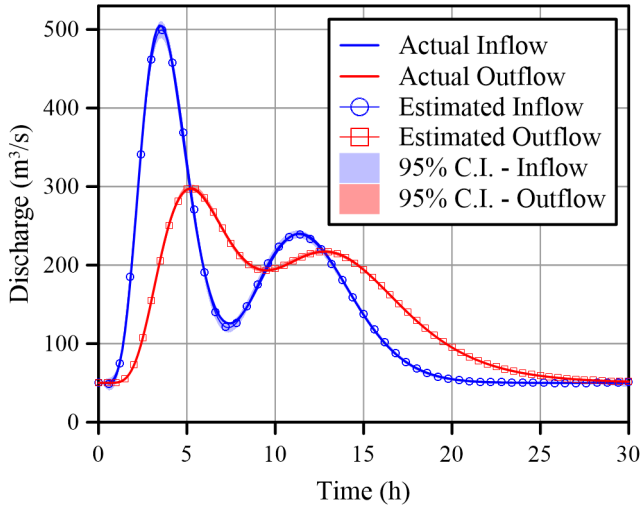

Figure 3.2. Case 1: actual and estimated inflow and outflow hydrographs with 95% credibility intervals.

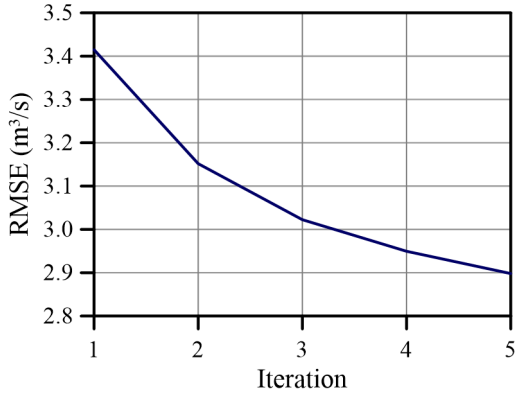

Figure 3.3. Case 1: root-mean-square error (RMSE) of the estimated inflow hydrograph at each iteration.

3.2.2. Reverse flow routing in an open channel

The second test focuses on the estimation of the inflow hydrograph to an open channel based on water level information collected in a downstream section using a given numerical model for the forward routing. In this work, the Hydrologic Engineering Center's River Analysis System (HEC-RAS), developed by the US Army Corps of Engineers (Brunner 2010), is used; it simulates one-dimensional unsteady flow by solving the Saint-Venant equations.

It is considered a prismatic channel, 20 km long, with a longitudinal slope of 0.0005 and compound cross sections spaced by 250 m consisting of a trapezoidal main channel and two symmetric floodplains (Fig. 3.4). The main channel has a bottom width of 50 m, a side slope of 2 and a depth of 6 m; each floodplain has a width of 50 m, horizontal bottom and vertical banks. Manning coefficients of 0.05 $m^{-1/3}/s$ and 0.1 $m^{-1/3}/s$ are adopted for the main channel and the floodplain, respectively.

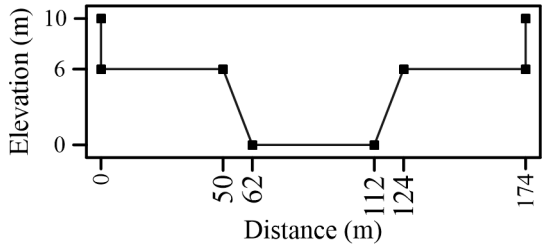

Figure 3.4. Case2: compound cross section of the prismatic channel.

The true upstream hydrograph is defined by Eq. 3.1 (M=2) with the coefficients reported in Table 3.2. The hydrograph has a first peak of about 1000 m^3/s at 5 h, a second peak with a discharge of about 500 m^3/s at 14 h and a base flow of 50 m^3/s. The total simulation time is 30 h and the upstream hydrograph is discretized in equal intervals of 30 min (N_p=61).

Table 3.2. Case 1: coefficients of the two gamma functions used for the description of the inflow hydrograph.

A [m^3/s]		**B [m^3]**	**n [-]**	**k [h]**
50	f_1	$1.6{\cdot}10^7$	8	0.7
	f_2	$1.4{\cdot}10^7$	18	0.8

The initial condition is obtained from a steady-state simulation according to the first inflow discharge value, assuming a steady-state condition before the flood event. The upstream and downstream boundary conditions are represented by the inflow hydrograph and the normal depth based on the Manning's equation, respectively.

The actual inflow hydrograph has been forward routed by means of HEC-RAS to obtain the water levels used as observations, which are recorded in the section in the middle of the channel, located 10 km downstream from the upstream section, every 30 min (m=61). It is consider a random observation error ε with normal distribution, zero mean and variance $2.8{\cdot}10^{-4}$m^2, that results in the 99.7% of the cases in errors in the range 0.05 m.

In this work, different settings of the inverse algorithm have been tested in the estimation of the upstream hydrograph; the impact of the ensemble size, the choice of the coefficient α during the iteration process, the covariance localization and the covariance inflation techniques are analyzed .

Three ensemble sizes have been analyzed, they are equal to: half the number of parameters (N_e= 31), the number of parameters (N_e=61) and three times the number of parameters (N_e=83). All the realizations of the initial ensembles are individual gamma functions generated using Eq. 3.1 with M=1 and coefficients randomly selected over the same wide range of values ([2, 180] m^3/s for A, [$8{\cdot}10^4$, $8{\cdot}10^7$] m^3 for B, [3, 18] for n; [0.6, 4.8] h for k).

For each ensemble size, four tests are carried out: the first test (T1) is per-

formed with a constant coefficient α used for all iterations and without other modifications on the inverse algorithm; the second one (T2) attempts to evaluate the effect of decreasing coefficient α as iterations progress; the third one (T3) studies the effect of covariance localization and covariance inflation keeping α constant; and the last test (T4), combines covariance modification (localization and inflation) with a decreasing α.

For each test, 6 iterations were performed with a constant α equal to 6 (α_{geo}=1, Eqs. 1.3 and 1.4), for test T1 and T3 and a decreasing α=[364; 121.33; 40.44; 13.48; 4.49; 1.50], obtained with α_{geo}=3 (Eqs. 1.3 and 1.4), for T2 and T4 (recall that the sum of the inverses of α values should add up to 1 (Eq. 1.2). Covariance localization and covariance inflation are applied using the coefficient b equal to 6 h (Eq. 1.18) and the inflation factor equal to 1.01 (Eq. 1.19), respectively. In this case, the update step is performed in logarithmic space to avoid the appearance of negative values.

The results of all tests are compared in term of the root mean squared error ($RMSE$) between the estimated hydrograph and the reference solution (Fig. 3.5). In all cases, the RMSE significantly decreases at each iteration, reaching low values at the end of the inversion. For the smaller ensemble size (Fig 3.5a) the method performs better when a decreasing α (T2) is used and when covariance inflation and localization techniques are used (T3), with the best results obtained when both options are used simultaneously (T4). For the larger ensemble size (Fig. 3.5b; Fig. 3.5c), the final RMSE is always smaller than for the smaller ensemble, and in all four experiments the final hydrograph is very close to the real one. Yet, the best performance, at the last iteration, is obtained for the experiment T4. Tables 3.3, 3.4 and 3.5 report the $RMSE$s at the end of each test, together with the Nash-Sutcliffe efficiency criterion and the relative errors in the peak discharge.

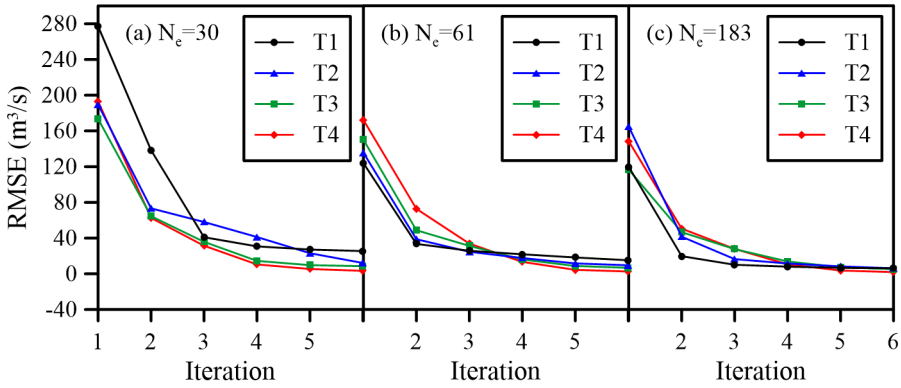

Figure 3.5. Case 2: RMSE of the estimated inflow hydrograph for ensemble size N_e=30 (a), N_e=61 (b) and N_e=138 (c).

Table 3.3. Case 2: root mean square error (RMSE), Nash-Sutcliffe efficiency criterion (NSE) and relative error in the peak discharge (E_p) between estimated and true inflow hydrographs for the four different tests (T1-T4) and for ensemble size N_e=30 at the end of the iterative process.

N_e=30	T1	T2	T3	T4
$RMSE$ [m^3/s]	25.47	12.12	8.57	3.32
NSE [%]	99.06	99.78	99.89	99.98
$E_{p,1}$ [%]	8.15	-0.18	1.79	-0.20
$E_{p,2}$ [%]	-0.11	1.61	2.09	-0.24

Table 3.4. Case 2: root mean square error (RMSE), Nash-Sutcliffe efficiency criterion (NSE) and relative error in the peak discharge (E_p) between estimated and true inflow hydrographs for the four different tests (T1-T4) and for ensemble size N_e=61 at the end of the iterative process.

N_e=61	T1	T2	T3	T4
$RMSE$ [m^3/s]	15.17	9.53	6.71	2.56
NSE [%]	99.67	99.87	99.93	99.99
$E_{p,1}$ [%]	0.73	2.13	0.27	-0.28
$E_{p,2}$ [%]	0.62	0.46	1.47	0.36

Table 3.5. Case 2: root mean square error ($RMSE$), Nash-Sutcliffe efficiency criterion (NSE) and relative error in the peak discharge (E_p) between estimated and true inflow hydrographs for the four different tests (T1-T4) and for ensemble size N_e=183 at the end of the iterative process.

N_e=183	T1	T2	T3	T4
$RMSE$ [m^3/s]	6.11	6.17	5.17	1.89
NSE [%]	99.95	99.94	99.96	99.99
$E_{p,1}$ [%]	1.28	1.53	-0.01	-0.66
$E_{p,2}$ [%]	0.65	0.73	0.87	-0.10

All the NSE values are above 99% indicating an accurate reproduction of the shape of the upstream hydrograph; the peaks are properly reproduced with E_p values lower than 2.15%, with only an exception (T1, N_e=30). Like $RMSE$, the metrics NSE and E_p confirm that decreasing α during the iterative process and adopting covariance modification techniques improve the performance of the ES-MDA especially when a small ensemble size is used.

For the sake of brevity, it is shown only the hydrographs resulting from the inversion obtained when the ensemble size is small and for two of the experiments, the one with no modifications of the ES-MDA algorithm (T1) and the one using a decreasing α and covariance localization and inflation techniques (T4). In Fig. 3.6 the true values and the ensemble means of the estimated inflow hydrographs with their 95% confidence intervals are depicted. Fig. 3.7 shows the true observations and the ensemble means of the estimated water levels with their 95% confidence interval. In both figures the residuals between actual and estimated values are also shown.

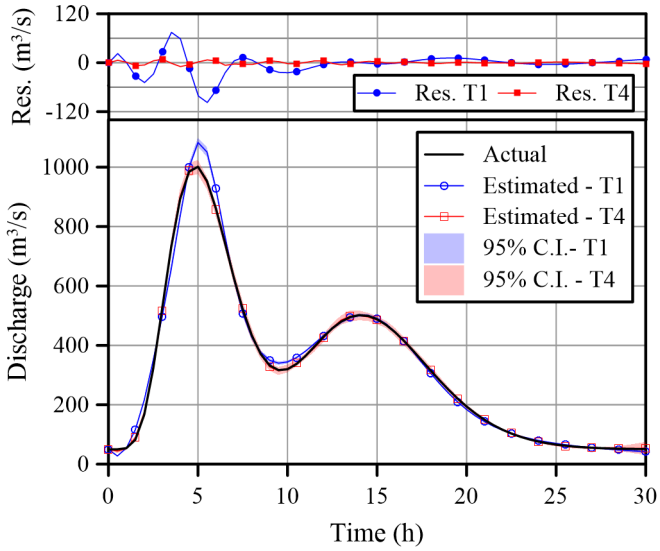

Figure 3.6. Case 2: actual and estimated upstream hydrographs with 95% confidence intervals (bottom) and residuals between actual and estimated values (top) resulting from tests T1 and T4 with N_e=30.

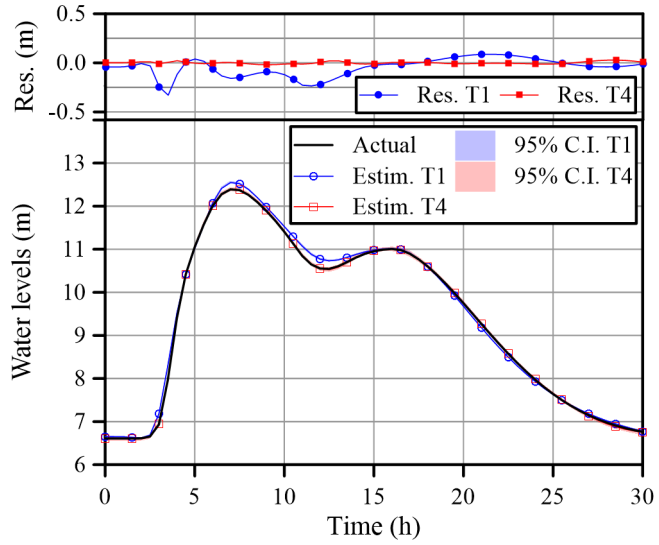

Figure 3.7. Case 2: actual and estimated water levels with 95% confidence intervals (bottom) and residuals between actual and estimated values (top) resulting from tests T1 and T4 with N_e=30.

Test T1 reproduces the shape of the inflow hydrograph quite well (NSE= 99.06%), but with a larger error on the first peak ($E_{p,1}$=8.15); the observations are not perfectly reproduced everywhere and the residuals are high in some points. Meanwhile, test T4 leads to a good match between the true and estimated inflow hydrograph (NSE=99.98%) and the true and estimated water levels with very small residuals. The inflow peaks and their timing are properly reproduced with negligible errors ($E_{p,1}$=-0.2%; $E_{p,2}$=-0.4%).

Comparison between ES-MDA and BGA

The results of test T4, obtained with the smaller ensemble size, are compared with those of the Bayesian Geostatistical Approach. The test is performed coupling BGA with the same forward model used for the solution of Case 2, considering the same simulation time (30 h) and discretization of the unknown hydrograph (N_p=61). The true observations were perturbed with random errors with zero mean and variance $2.8{\cdot}10^{-4}$ m^2. The number of iterations for the linearization process (inner loop) are equal to N_i=5 and equal to N_o=4 for the outer loop required to estimate the hyperparameters.

The results of the comparison are reported in Fig. 8. The BGA method accurately estimates the inflow hydrograph ($RMSE$=4.2 m^{3}3/s, NSE=99.97%) with small residuals and small errors in the estimation of the peaks ($E_{p,1}$=$-$1.0%; $E_{p,2}$=$-$0.3%). The two approaches show fully comparable results, which are confirmed by a very similar residual range and the almost equal values of the performance metrics. However, ES-MDA outperform BGA in terms of total number of forward model runs required and hence computational time: 1241 (Eq 3.6) runs and 182 for ES-MDA, given by the product of the number of ensemble realizations and number of iterations ($N_e \cdot N_i$).

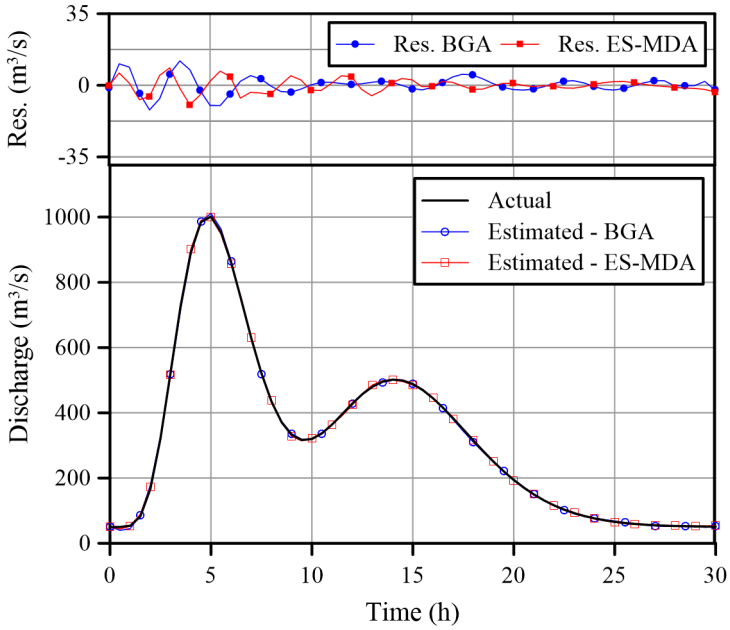

Figure 3.8. Case 2: actual and estimated upstream hydrographs (bottom) and residuals between actual and estimated values (top) resulting from BGA and ESMDA (T4, N_e=30) approaches.

3.3. Real test case

In this section, the proposed inverse procedure is validated investigating a real flood event occurred between the 10th and the 13rd of November, 2012, with a total duration of 63 h. The studied domain is a portion of the Parma-Baganza system, located in Northern Italy. The test case aims to estimate the inflow hydrograph on the tributary Baganza River based on the knowledge of the inflow hydrograph on the Parma River (main reach), water level data collected downstream the confluence and a reliable calibrated HEC-RAS hydraulic model. The same flood event was simulated by D'Oria et al. (2014) by means of the BGA

approach allowing a comparison between the two methods.

The considered river system is sketched in Figure 3.9. The simulated part of the Parma River has a length of about 10.7 km; the upstream boundary condition (Section 1) is represented by the outflow hydrograph from a flood control dam equipped with movable gates; the downstream boundary condition (Section 5) is the normal depth as evaluated from the Manning's equation. The hydrometric site is located at Section 4 on the downstream side of the Parma River. The selected part of the Baganza river has a total length of about 1.1 km and its upstream boundary condition (Section 2) in terms of inflow hydrograph represents the unknown of the problem. A level gauge is also available on the Baganza River upstream the confluence (Section 3).

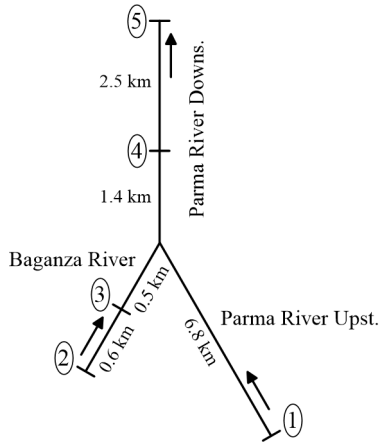

Figure 3.9. Case 3: Sketch of the Parma-Baganza reach system.

The dam gates were moved during the flood to control the released discharge resulting in a unnatural shape of the inflow hydrograph on the Parma River depicted in Figure 3.13; the hydrograph was recorded with a temporal discretization equal to 30 min.

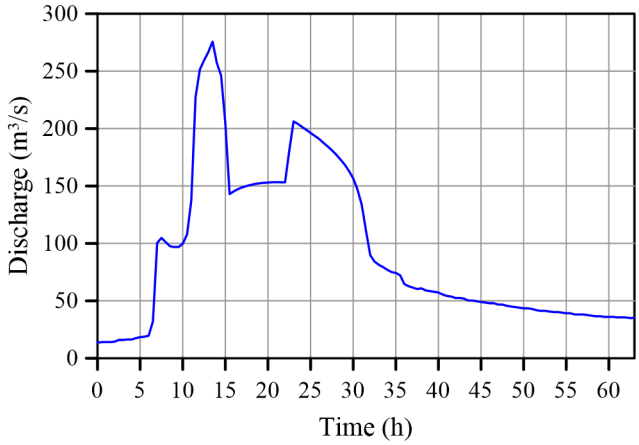

Figure 3.10. Case 3: Observed Parma River inflow hydrograph at the dam location (Section 1).

The inverse procedure is performed assuming the tributary river completely ungauged and only using the water level data collected at Section 4 as observations. The stage hydrographs were recorded with a temporal discretization equal to 30 min (m=30). The unknown discharge hydrograph on the Baganza River is discretized in equal interval of 1 h, therefore the total number of parameters to be estimated is N_p=63. The stage hydrograph recorded at Section 3 is used in post-processing to assess the reliability of the proposed method.

The ES-MDA setting, used for this case, derived from the results of the second synthetic example presented above. A small ensemble size (N_e=30), a number of iterations equal to 6 and the configuration of test T4 are adopted: decreasing α (α_{geo}=3, Eqs 1.3 and 1.4), covariance localization (b=6h, Eq 1.18) and covariance inflation (r=1.01, Eq. 1.19). The observation error is considered, also in this case, normally distributed with zero mean and variance $2.8{\cdot}10^{-4}$ m^2. The initial ensemble is composed of individual gamma functions (M=1, Eq. 3.1) generated using the coefficients selected randomly over the ranges: [1, 50] m^3/s for A, [$8{\cdot}10^4$, $8{\cdot}10^7$] m^3 for B, [3, 18] for n; [0.6, 4.8] h for k.

Following the results of the ES-MDA and the comparison with the BGA procedure are presented. Both methods are performed using the same forward model and discretization times (for more detail, see D'Oria et al. 2014). Figure 3.11 shows the estimated inflow hydrographs on the Baganza River with their 95% confidence intervals by means of ES-MDA and BGA; the two approach leads to a very similar result. In figure 3.12 the observed and estimated water depth recorded at Section 4, which were not been used in the inverse procedures, are depicted with the residuals between actual and estimated values. Both the method accurately reproduced the observations, the Nash–Sutcliffe efficiency coefficient, calculated using the water depths instead of the discharge data (Eq. 3.4) is 96.66% for ES-MDA and 96.78% for BGA; the peak value and its timing is well reproduced for both the methods. It should be noted that only ES-MDA allow to assess the uncertainty of the predicted water levels, which results in a 95 %confidence interval that contains almost all the actual values.

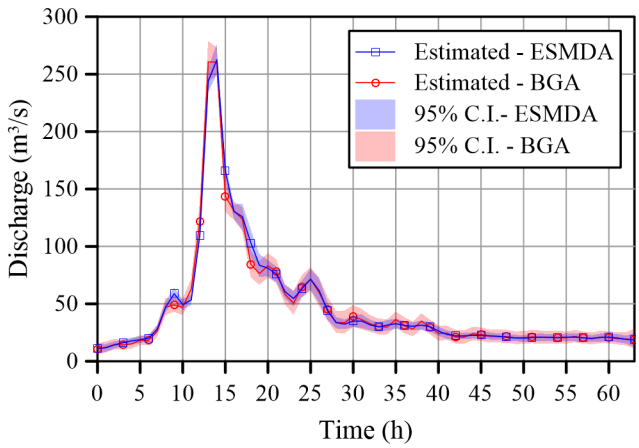

Figure 3.11. Case 3: Estimated Baganza River inflow hydrographs (with 95% credibility interval) resulting from ES-MDA and BGA.

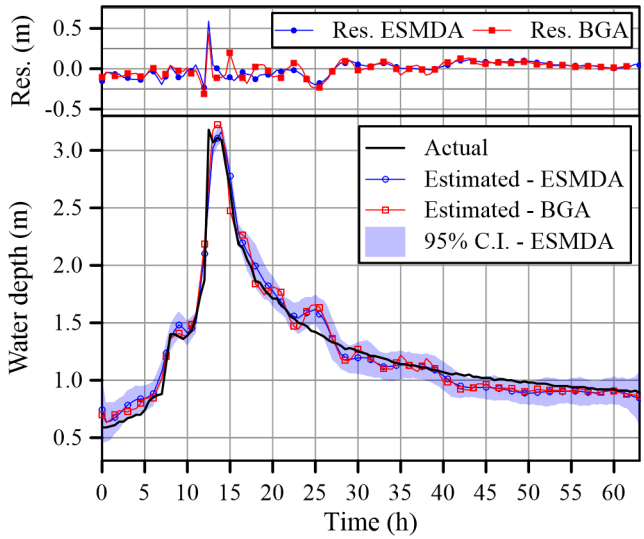

Figure 3.12. Case 3: actual and estimated water levels, collected upstream the confluence on the tributary Baganza River, with 95% confidence intervals (bottom) and residuals between actual and estimated values (top) resulting from ES-MDA and BGA.

Figure 3.13 shows the observed and estimated water depth, used in the inverse procedures, collected downstream the confluence at Section 4 (Figure 3.9). The ES-MDA and BGA leads to almost the equal predicted values, which are very close to the true ones. The residuals between actual and estimated water levels are similar and small; the 95% credibility interval, estimated through ES-MDA, is very narrow.

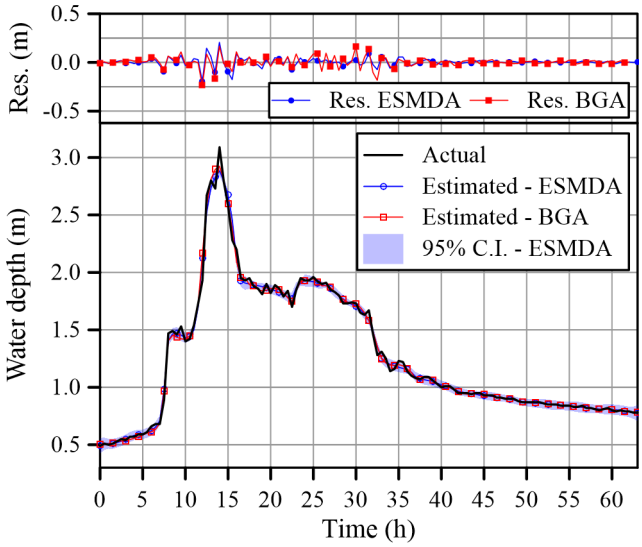

Figure 3.13. Case 3: actual and estimated water levels, collected downstream the confluence, with 95% confidence intervals (bottom) and residuals between actual and estimated values (top) resulting from ES-MDA and BGA.

Although, the two approaches lead to comparable results, ES-MDA requires fewer runs of the forward model to reach the solution, leading to a significant reduction of the computational cost. The total number of forward model runs are 1576 for BGA (N_i=5 and N_o=5, Eq 3.6) and 180 ($N_e \cdot N_i$) for ES-MDA; which means a reduction of the computational effort by a factor of about 9.

3.4. Concluding remarks

In this chapter, a new approach for the solution of the reverse flow routing problem has been proposed. The Ensemble Smoother with Multiple Data Assimilation (ES-MDA) has been applied for the estimation of the inflow to a hydraulic system based information recorded downstream. Two synthetic examples were considered to test the methodology, which was then applied for the solution of a real case

study.

The first case shows the capability of the inverse procedure in estimating the inflow hydrograph to a linear reservoir, where the outflow hydrograph and the reservoir characteristics are known. It is noteworthy that for linear problems the ensemble smoother methods should lead to the exact solution in a single update step, provided that the observations are free of errors and the initial ensemble is statistically representative of the variability of the unknowns. In this case, due to the presence of corrupted observations, the ES-MDA updates the vector of parameters in multiple iterations. At the end of the process, the true inflow hydrograph is accurately reproduced; the Nash-Sutcliffe efficiency criterion (NSE) is 99.94%, the errors in the peak discharges are less than 1.1% and the RMSE reaches the small value of 2.9 m^3/s.

The second case study validates the method for non-linear problems by estimating the inflow hydrograph to an open channel based on water level information collected in a downstream section and for given forward routing model. The effects of different settings of the inverse algorithm were investigated: the ensemble size, the decreasing α during the iterative process and the temporal localization and inflation of the covariances. In all tests, the NSE exceeds 99% and, as expected, the ES-MDA reaches a better solution increasing the ensemble size. However, as the ensemble becomes larger, the computational time increases, since, at each iteration, the method requires a number of forward runs equal to the number of realizations. The results of our tests show that a significant improvement in the inverse solution is obtained if a decreasing α and the covariance modifications are applied, the ensemble size being equal. This is particularly clear working with small ensemble sizes, since covariance localization and inflation overcome the problem of undersampling that occurs when a low number of realizations is used. The test performed with the smaller ensemble size using a decreasing α and the covariance modifications reproduces very well the inflow hydrograph with negligi-

ble errors. The NSE is 99.98% and the relative error in the peak discharges are less than 0.3%; these values are fully comparable with those obtained with the larger ensemble size. The RMSE is 3.32 m^3/s, which corresponds to a reduction of about 87% compared to test with constant α and the basic algorithm for the same ensemble size.

The third case study analyzes a flood event occurred on the Parma-Baganza reach system at the city of Parma, in Northern Italy. The setting of the inverse procedure for the solution of the real case study make use of the information derived from the several configurations analyzed for the second synthetic example. Therefore, ES-MDA is performed, for this case study, using a small ensemble size and applying the covariance localization and inflation. The results show the capabilities of ES-MDA to reach a good solution also for complex river systems with a small computational cost.

In summary, the modified ES-MDA method allows to solve the reverse flow routing problems using also small ensemble sizes (with a total number of realizations less than the number of parameters) leading to a significant reduction of the computational burden. The modified algorithm provides results comparable with those of the other optimization methods presented in the recent literature, although ES-MDA achieves the solution with a lower number of forward runs. In addition, the forward runs related to the ensemble realizations can be easily parallelized allowing an additional reduction of the computational time. Moreover, another important advantage of the method is the capability to assess the uncertainty in the estimations from the realizations of the ensemble. It allows to quantify the uncertainty associated with both the unknown parameters and the reproduction of the observations, which is a novelty in the solution of the reverse flow routing problem.

It is noteworthy to point out that one can handle non-Gaussian distributed parameters, and it is well known that the ensemble Kalman filter methods are op-

timal for multiGaussian distributed variables. The results, for the analyzed case studies, show that ES-MDA was able to reach a good solution in all cases. However, for those cases in which the method may fail due to the non-Gaussianity of the parameters, different approaches are presented in the literature to overcome the problem; for instance, ES-MDA can be couple with the Normal-Score transformation, which it is shown to work properly with ensemble Kalman filter methods (Zhou et al. 2011, Li et al. 2018).

Finally, another aspect that should be taken into account is the uncertainty in the forward model. Since the inverse methodology requires a numerical model able to accurately describe the forward processes, the errors in the model structure could add to the measurement noise. Therefore, in real applications, a proper and calibrated forward model is crucial to obtain a reliable inverse solution and a careful check of the most uncertain model parameters is advisable.

4

Calibration of a numerical hydrological-hydraulic model

4.1. Introduction

Model calibration is a crucial step to obtain mathematical models able to well reproduce the behaviour of natural systems. The calibration process is a type of inverse problem in which unknown model parameters are to be inferred from available data representing the calibration target. In this chapter, the ensemble smoother with multiple data assimilation is applied to calibrate the Parflood Rain model developed by Prost (2019) and Aureli et al. (2020) for the simulation of rainfall-runoff processes.

The objective is the estimation of the Manning and infiltration coefficients, which are some of the input data required by Parflood Rain, on the basis of a known

discharge hydrograph in the outlet river section. In the literature, the roughness and infiltration coefficients have been usually defined as physically interpretable parameters identifiable on the basis of the system characteristics, such as soil type and use. Nevertheless, an automatic calibration of these parameters, for each specific case, can lead to more reliable numerical models.

The capability of the proposed methodology was firstly tested by means of two synthetic cases and then applied to a real one. The first example is a V-shaped rainfall-runoff test case, widely used in the literature; the second one is a synthetic case study for which the real domain and rainfall event were used. Finally, the ES-MDA is applied for the Parflood Rain calibration related to the real flood event occurred on October the 13th, 2014 on the Baganza reach system located at the city of Parma, in Northern Italy.

4.2. Forward model: Parflood Rain

Parflood Rain is a GPU-parallelized numerical scheme that solves the complete 2D shallow-water equations (SWEs) allowing to fast simulate the flood propagation process on a watershed. ES-MDA and Parflood Rain have been coupled to take advantage of the parallel computing; the available high-performance computing technology allow to simultaneously run several forward simulations, related to the ensemble realizations, leading to a reduction of the computational burden.

For all the test cases, the calibration of the Parflood Rain model is performed through the one factor method. The unknowns parameters to be estimated are scale factors that apply to initial maps of roughness and infiltration obtained from information on soil types and land use; this limits the parameter number and, as a consequence, the computational time required to perform the inverse procedure. Therefore, the spatial distributions of the input maps do not change, but they are

only uniformly scaled in order to maintain the information derived from the soil type and use.

The first calibration input is the map of the Manning roughness coefficients. The parameter to be estimated through the ES-MDA procedure is the scale factor (c_k) that applies to the roughness values expressed according to the Strickler formulation, which are subsequently transformed into Manning coefficients, according to the relation:

$$n = \frac{1}{c_k \cdot k}, \tag{4.1}$$

where n and k are the Manning and Strickler coefficients, respectively.

The second parameter to calibrate is the infiltration. The Soil Conservation Service Curve Number method (SCS-CN) is used by Parflood Rain to evaluate the infiltration for a given rainfall event; the map of the curve number values is used as input to the model. The ES-MDA aims at estimating the scale factor that apply to the potential maximum retention (S_∞), which is related to the curve number by the expression:

$$S_\infty = 25.4 \cdot \left(\frac{1000}{CN} - 10 \right). \tag{4.2}$$

First, the values of S_∞ corresponding to the map of the curve numbers are computed; then, at each iteration, they are modified applying the scale factor c_s and, finally, the updated values of CN are obtained by inverting Eq. 4.2 and rounding to the nearest integer:

$$CN = 1000 \cdot \frac{25.4}{c_s \cdot S_\infty + 254} \tag{4.3}$$

4.3. Test cases

In this section, the test cases used to investigate the ES-MDA capability to calibrate the Parflood Rain numerical model are presented. The methodology is first applied to two synthetic cases and then to a real case study; for the second synthetic case and the real one, the simulated domain is the Baganza river basin located in Northern Italy.

4.3.1. V-shaped rainfall-runoff test case

The first synthetic example is an application of ES-MDA for the calibration of Parflood Rain applied to solve a two-dimensional V-shaped rainfall-runoff case. The V-shaped catchment, depicted in Fig 4.1, is characterized by two symmetric hillsides that flow into a rectangular channel, 1000 m long and 20 m wide, with a 0.02 slope in the y-direction. Each side has a width of 800 m and a 0.05 slope in the x-direction. The domain is subject to a uniform rainfall with intensity 100 mm/h for a duration of 1.5 h.

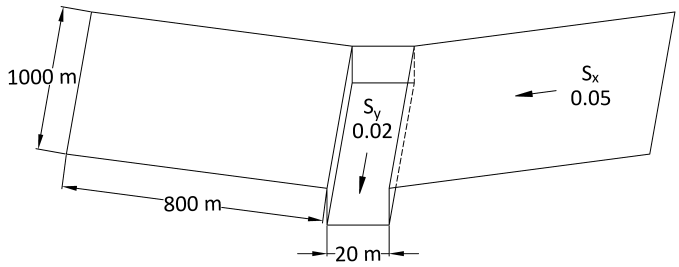

Figure 4.1. V-Shaped rainfall-runoff test case: domain schematization.

The adopted infiltration distribution map is shown in Fig. 4.2; each index corresponds to a value of CN as reported in Table 4.1.

Table 4.1. V-Shaped rainfall-runoff test case: infiltration indices and related curve number.

Index	1	2	3
CN	99	80	60

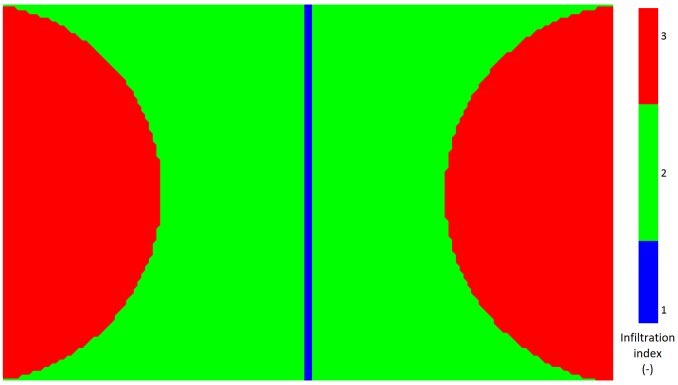

Figure 4.2. V-Shaped rainfall-runoff test case: infiltration index map.

The adopted Manning coefficients are equal to 0.15 $\mathrm{sm}^{-1/3}$ and 0.1 $\mathrm{sm}^{-1/3}$ for the main channel and the hillsides, respectively. The actual scale factors c_k and c_s, which are the investigated parameters (N_p=2), are assumed of unit value. They are used as input to perform a forward run in order to obtain the observations of the problem, which is a discharge hydrograph extracted in the downstream section of the main channel. The total simulation time is 3 h and the observed hydrograph is discretized in equal intervals of 3 min resulting in a total number of observations m=61.

The inverse procedure is performed considering random observation errors ε normally distributed with zero mean and variance 2.8$\cdot$ 10^{-4} $(\mathrm{m}^3/\mathrm{s})^2$. The initial ensemble consists of 18 realizations of the parameters c_k and c_s (N_e=18), which are coefficients randomly selected from a uniform distribution over the same range of values [0.5, 1.5]. The ES-MDA is run with 5 iterations and a decreasing α

obtained with α_{geo}=3 (Eqs. 1.3 and 1.4).

Table 4.2 compares the reference and estimated scale factors c_k and c_s; the ensemble means with their 95% uncertainty intervals are reported. The proposed method accurately reproduces the actual factors as well as the observed discharge hydrograph (Fig. 4.3). The estimated coefficients present percentage errors of 0.1% and 0.5% for c_s and c_k, respectively.

Table 4.2. V-Shaped rainfall-runoff test case: actual and estimated parameters with 95% uncertainty interval.

Parameters	**Actual**	**Estimated**
c_k	1.000	0.999±0.002
c_s	1.000	0.995±0.005

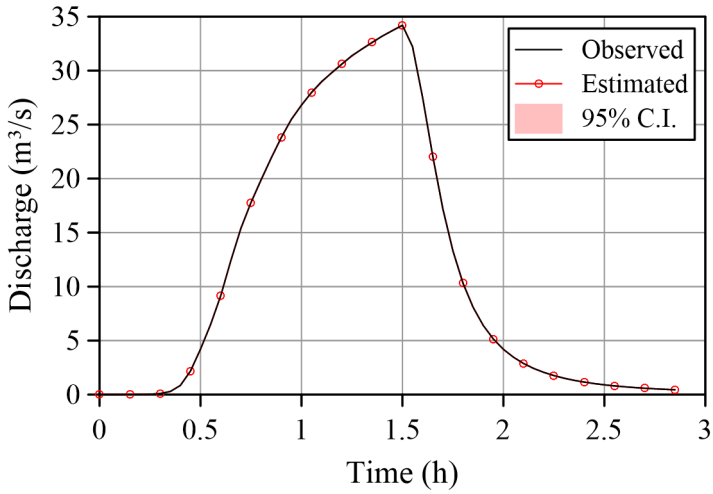

Figure 4.3. V-Shaped rainfall-runoff test case: observed and estimated discharge hydrograph with its 95% uncertainty interval.

4.3.2. Baganza basin cases

The ES-MDA method is applied for the calibration of Parflood Rain model for the simulation of a flood event occurred in a real basin field. The studied domain is the Baganza river basin, located in Northern Italy. The investigated event occurred between the 13rd and the 15th of October 2014, with a total duration of 40 h. The first 30 hours of the flood event are simulated, corresponding to a computational time of 1.25 h. First, a synthetic case is performed assuming the coefficients known and then a real calibration test is carried out.

The unknown parameters are represented by the scale factor of the infiltration map c_s (Table 4.3 and Fig. 4.4), and two multiplicative factors that apply to the roughness map (Fig. 4.5), which differ for the area of the reach (c_{k1}) and the rest part of the basin (c_{k2}); the total number of parameters to be estimated is N_p=3.

Table 4.3. Baganza basin: infiltration indices and related curve number.

Index	1	2	3	4	5	6
CN	57	61	70	72	75	76
Index	7	8	9	10	11	12
CN	84	86	90	96	98	99

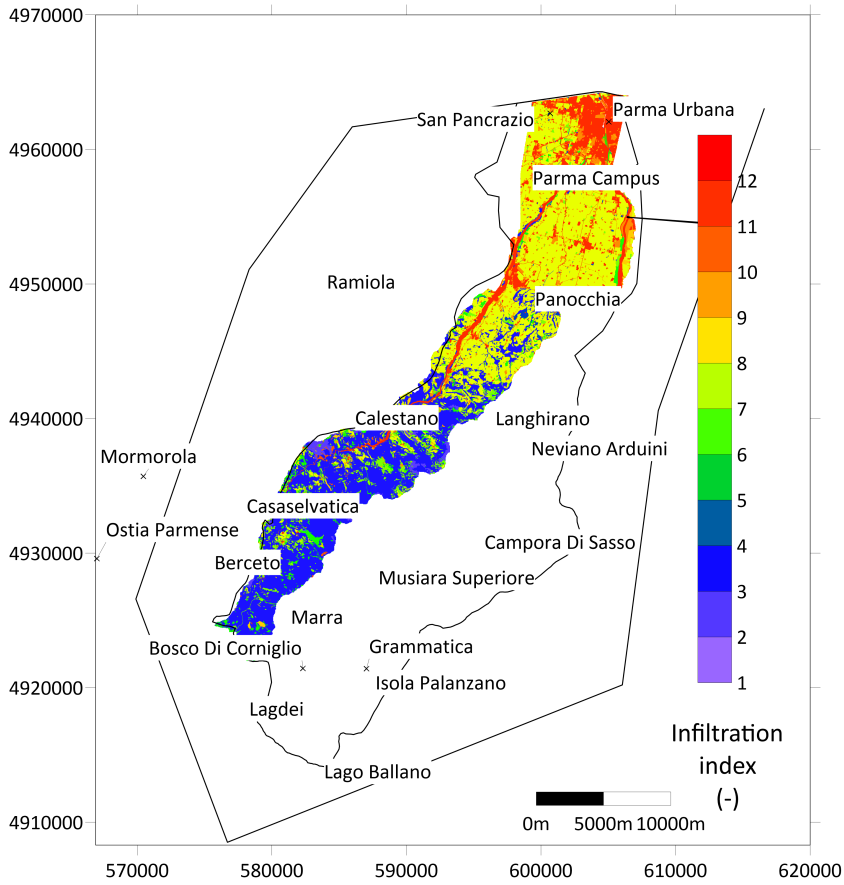

Figure 4.4. Baganza basin: infiltration index map.

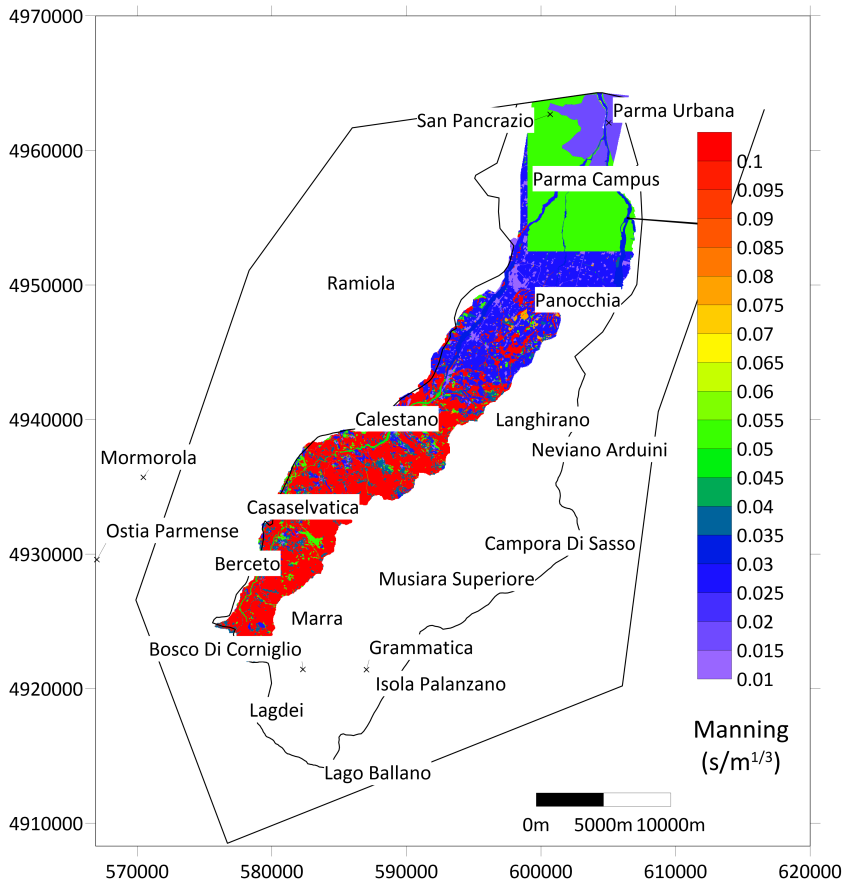

Figure 4.5. Baganza basin: Manning's roughness map.

Synthetic case

The synthetic case on the Baganza basin is performed considering the scale factors equal to 0.87 for c_s, 1.01 for c_{k1} and 1.51 for c_{k2}; which represent the reference solution and are used to obtain the observations through a forward run of the numerical model. The observations consist of a discharge hydrograph collected in a section of the Baganza river located upstream the flooded area; which is recorded for the last 16.5 h of the simulation with a time step of 30 min, resulting in a total

number of observations m=33.

The ES-MDA is performed considering an observation error equal to 5% of the true discharge values: the errors are normally distributed with zero mean and variances defined so that the 99.7% of the errors lies within the 5% of the corresponding discharge value. The ensemble size is equal to 20 and the initial parameter realizations are random values selected from a uniform distribution in the following ranges: [0.8, 1.6] for c_{k1} and c_{k2}, [0.7, 1.0] for c_s. 5 iterations are performed with a decreasing *alpha* coefficient obtained with $\alpha_{geo} = 3$.

Table 4.4 summarizes the results of the inverse procedure, the actual coefficients and the ensemble mean with its 95% uncertainty interval are reported. The estimated scale factors are in good agreement with the actual ones; the percentage estimation errors are 1.6% for c_{k1}, 6.2% for c_{k2} and 0.0% for c_s. Figure 4.6 shows the observed and estimated discharge hydrograph; the ES-MDA leads to a good reconstruction of the observed values with a narrow confidence band.

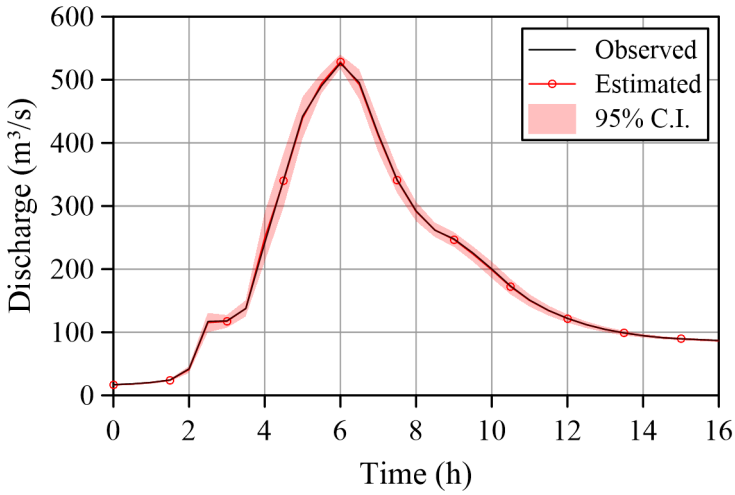

Figure 4.6. Baganza basin synthetic case: observed and estimated discharge hydrograph with its 95% uncertainty interval.

Table 4.4. Baganza basin synthetic case: actual and estimated parameters with 95% uncertainty interval.

Parameters	**Actual**	**Estimated**
c_s	0.868	0.868 ±0.012
c_{k1}	1.056	1.073±0.079
c_{k2}	1.508	1.420±0.146

Real case

Finally, the ES-MDA is applied for the calibration of Parflood Rain model aimed at simulate the Baganza flood event of October 2014. The objective is the identification of the scale factors c_s, c_{k1} and c_{k2} ($N_p = 3$) on the basis of a known discharge hydrograph at the basin outlet. Since no gauging stations are available on the study domain, the discharge hydrograph was estimated by other means and it is treated hereafter as observation. To apply the inverse procedure, the discharge values for the last 16 h of the simulation with a discretization of 15 min are considered; the volume under the flood hydrograph is assumed as additional observation, leading in a total number of observations m=66.

It was reported that the reconstructed flow hydrograph, used as calibration target, presents more reliable discharge data around the peak then those on the rising and falling limbs; therefore, different measurement error ranges are considered in applying the inverse procedure. The observation errors are normally distributed with zero mean and variance equal to 0.2 $(m^3/s)^2$ for the discharge values greater than 300 m^3/s and 6 $(m^3/s)^2$ for the other ones; a variance equal to $5 \cdot 10^7$ m^3 is considered for the volume. The initial ensemble is composed of 20 realizations of random scale factors selected from a uniform distribution over the following range of values: [0.5, 1.8] for c_{k1} and c_{k2} and [1.0, 2.0] for c_s. The ES-MDA is performed with 5 iterations and a decreasing α obtained with α_{geo}=4.

The estimated parameters with their 95% uncertainty interval are reported

in Table 4.5. Fig. 4.7 shows the comparison between the actual and estimated discharge hydrograph with its 95% uncertainty interval. The ES-MDA is able to well reconstruct the shape of the flow hydrograph; the Nash–Sutcliffe efficiency coefficient (Eq. 3.4) is equal to 93.58%. The peak timing is well reproduced, but the error on the peak discharge is quite large (E_p=-14.36%, Eq. 3.5). This can be due to errors in the reconstruction of the discharge hydrograph, used here as observation, or in the rainfall input; it is known that a rain station was affected by a technical problem during the event and, even if some corrections were applied, errors in the spatial and temporal distribution of the rainfall can be present. Further analysis will be conducted to investigate these problems.

Table 4.5. Baganza basin real case: estimated parameters with 95% uncertainty interval.

Parameters	Actual	Estimated
c_s	-	1.899±0.009
c_{k1}	-	1.867±0.007
c_{k2}	-	0.651±0.004

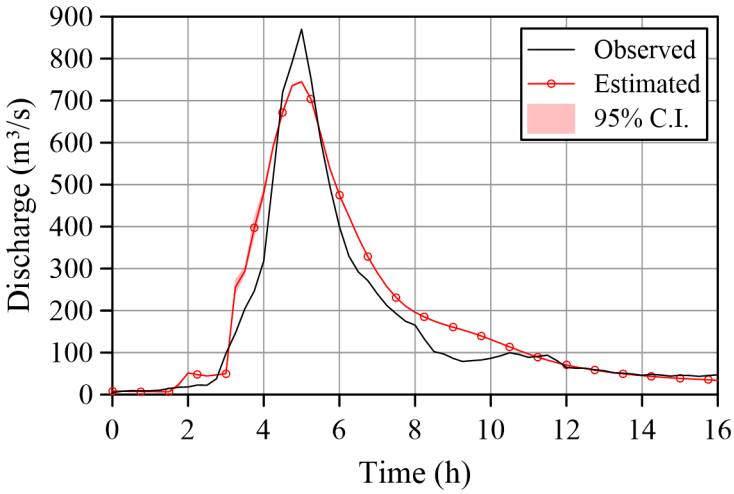

Figure 4.7. Baganza basin real case: Observed and estimated discharge hydrograph with its 95% uncertainty interval.

4.4. Concluding remarks

In this chapter, the Ensemble Smoother with Multiple Data Assimilation (ES-MDA) has been applied for the solution of the inverse problem that aimed at calibrating a numerical hydrological-hydraulic model. The objective is to estimate the input maps of roughness and infiltration parameters to the Parflood Rain numerical model on the basis of a known observed discharge hydrograph in a downstream section of the considered basin. Two synthetic examples, which allow to compare the estimation with a reference solution, and a real case were performed to test the methodology. The results of the synthetic tests prove the capability of ES-MDA to deal with this type of inverse problem leading to an accurate reconstruction of the investigated parameters. The application for the calibration of the hydrological-hydraulic numerical model related to a real flood event reaches satisfactory solutions; the shape of the discharge hydrograph, used as calibration target, and the peak time are well reproduced. However, the estimated peak value present a quite large error, which will be object of future investigation in order to improve the solution.

The proposed methodology is a promising approach to perform an automatic calibration of hydrological numerical models for each specific cases. The direct estimation of roughness and infiltration coefficients may also allow to compensate for different source of errors in the model setup leading to better results. In the present literature, these parameters are often physically determined a-priori since the calibration of complex numerical models is very time consuming. The parallelization of the inverse procedure, which is one of the main advantage of the ES-MDA method, leads to a reduction of the computational burden and allow to perform the model calibration in an acceptable time.

Future development of the proposed methodology that couple ES-MDA with Parflood Rain model will focus on the possibility to simultaneously perform the

reverse routing for the estimation of an inflow hydrograph to the analyzed system and the calibration of some model parameters, such as the Manning's roughness coefficients.

5

Simultaneous identification of the release history and the source location of a pollutant in groundwater

5.1. Introduction

Monitoring, protection and restoration of the groundwater quality have received much attention in the past decades, thanks to the growing interest in environmental issues and the importance of groundwater for water supply. The first steps in any remediation strategy are the identification of the source location and the release history of the contaminant, since they allow to identify the cause of the contamination, to implement an effective remediation strategy and to share the

costs among the responsible parties.

When groundwater contamination is first detected, the source location and the release history are usually unknown. Recovering these variables from sparse data of the spatial distribution of the pollutant concentration in the aquifer is a type of inverse problems. Inverse problems are inherently ill-posed, which means that the solution is generally non-unique and could be not stable to small perturbations in the data. Several deterministic and stochastic methods have been proposed to solve this problem. The first category includes the Tikhonov regularization (Skaggs & Kabala 1994), nonlinear optimization with embedding (Mahar & Datta 1997), non-regularized nonlinear least squares (Alapati & Kabala 2000), progressive genetic algorithms (Aral et al. 2001), a constrained robust least squares (Sun et al. 2006) and heuristic harmony search algorithms (Ayvaz 2010). The second category adopts probability-based methods: statistical pattern recognition (Datta et al. 1989); minimum relative entropy (Woodbury & Ulrych 1996, Woodbury et al. 1998, Cupola et al. 2015); geostatistical approaches (Snodgrass & Kitanidis 1997, Michalak & Kitanidis 2004*a*,*b*, Neupauer et al. 2000, Butera & Tanda 2003, Butera et al. 2006, 2012, Gzyl et al. 2014, Cupola et al. 2015); empirical Bayesian methods combined with Akaike's Bayesian Information Criterion (Zanini & Woodbury 2016); Bayesian global optimization (Pirot et al. 2019) and ensemble Kalman filter methods (Xu & Gómez-Hernández 2016, 2018, Chen et al. 2018, Xu et al. 2020).

However, only a few of the presented studies allow to simultaneously identify the source location and the release history of a groundwater contaminant. The method proposed by Aral et al. (2001) used a progressive genetic algorithm to solve an iterative nonlinear optimization problem, in which the source location and release history were explicitly defined as continuous unknown variables and contaminant concentrations were used as observations. Sun et al. (2006) combined a constrained robust least squares estimator with a global optimization solver

for iteratively identifying release histories and source locations on the basis of concentration measurements. Ayvaz (2010) used an optimization method based on the heuristic harmony search algorithm to identify locations and release histories for pollution sources, minimizing residuals between the simulated and measured contaminant concentrations. All these methods are deterministic and do not allow to quantify the uncertainty of the results.

Butera et al. (2012) applied a Bayesian geostatistical approach for the simultaneous identification of the release function and the source location based on concentration data. The methodology has then been tested by Cupola et al. (2015) on real data collected in a laboratory sandbox. The method requires a preliminary delineation of possible sources and some hypotheses about the structure of the unknown release function. The approach aims to recover the contaminant release history considering all the possible sources simultaneously and selecting the location where the highest amount of pollutant is estimated. The method adopts a transfer function approach for the solution of the forward problem (Butera et al. 2006).

Here, the Ensemble Smoother with Multiple Data Assimilation (ES-MDA) is proposed as a new approach for the joint identification of the source location and the release history of a pollutant in groundwater. Compared with the Bayesian geostatistical approach (Butera et al. 2012), the ES-MDA does not require the explicit time-consuming calculation of sensitivity matrices to solve the inverse problem, since they are embedded in the covariance matrices of the ensemble. Moreover, it allows the simulation of groundwater flow and mass transport even in complex cases.

In this study, the parameters are represented by the spatial coordinates of the source and the temporal discretization of the release history; the observations are sparse concentration data measured at different monitoring locations and time. Notice that in other practical applications, piezometric head data may be available,

which could also be assimilated and used in the solution of the inverse problem; it is not the case in the laboratory experiment described next, for which no piezometric head data were available.

Two applications of ES-MDA are presented. First, it is used for the solution of a a synthetic case from the literature with the aim to show its capabilities and to obtain guidelines for its application to real cases. Second, the ES-MDA is used to validate the methodology on experimental data collected in a laboratory sandbox that mimics an unconfined aquifer; it was also preliminary used for the calibration of the numerical model required to perform the inverse procedure.

The synthetic case study allows to investigate in detail different settings of the inverse procedure with a limited computational effort. In particular, it is evaluated the impact of the observations sampling scheme and different algorithm settings in the context of ill-posedness of inverse problems. The ill-conditioning increases as uncertainties about the model increase and as the quantity and quality of the observed data decrease. Therefore, it is important to design a monitoring network that makes a good compromise between valuable information about the concentration evolution and the costs of monitoring actions, which would limit the number of monitoring points.

Localization and inflation techniques are applied to overcome the well-known problem of undersampling in ensemble-based methods. In this study, parameters and observations are both space and time dependent, furthermore the distance between them is not fixed since the source position is unknown, what complicates the use of standard localization techniques. The new localization approach, which takes into account both spatial and temporal distance and iteratively update the distance between the unknown parameters and observations, is used.

The manuscript is organized as follows: first, the forward problem is described; then, the synthetic and the laboratory case study are presented and discussed.

5.2. Forward problem: groundwater flow and transport

The flow equation of an incompressible fluid in saturated porous media can be written as

$$\nabla \cdot (\mathbf{K}(\mathbf{x})\nabla h(\mathbf{x},t)) - S_s(\mathbf{x})\frac{\partial h(\mathbf{x},t)}{\partial t} + Q(\mathbf{x},t) = 0, \tag{5.1}$$

where $h(\mathbf{x},t)\,[\mathrm{L}]$ is the piezometric head at location $\mathbf{x}$ and time t, $\mathbf{K}(\mathbf{x})\left[\mathrm{LT}^{-1}\right]$ is the hydraulic conductivity tensor, $S_s(\mathbf{x})\left[\mathrm{L}^{-1}\right]$ is the specific storage coefficient, and $Q(\mathbf{x},t)\left[\mathrm{T}^{-1}\right]$ is the injection flow rate per unit volume.

In this study, the transport of a non-reactive contaminant injected in the aquifer at a point source is considered, the advection-dispersion equation is:

$$\begin{aligned}\frac{\partial\left(\phi(\mathbf{x})C\left(\mathbf{x},t\right)\right)}{\partial t} =& \nabla\cdot\left[\phi(\mathbf{x})\mathbf{D}\left(\mathbf{x}\right)\nabla C\left(\mathbf{x},t\right)\right] - \nabla\left[\phi(\mathbf{x})\mathbf{v}\left(\mathbf{x},t\right)C\left(\mathbf{x},t\right)\right] \\ &+ s\left(\mathbf{x_0},t\right)\delta\left(\mathbf{x}-\mathbf{x_0}\right),\end{aligned} \tag{5.2}$$

where $\phi(\mathbf{x})\,[-]$ is the effective porosity, $C\left(\mathbf{x},t\right)\left[\mathrm{ML}^{-3}\right]$ is the solute concentration, $\mathbf{D}\left(\mathbf{x}\right)\left[\mathrm{L}^2\mathrm{T}^{-1}\right]$ is the hydrodynamic dispersion coefficient tensor, $\mathbf{v}\left(\mathbf{x},t\right)\left[\mathrm{LT}^{-1}\right]$ is the effective flow velocity, obtained from the solution of the flow model, and $s\left(\mathbf{x_0},t\right)\left[\mathrm{MT}^{-1}\right]$ is the the contaminant flux injected into the aquifer through the source located at $\mathbf{x_0}$ given by

$$s\left(\mathbf{x_0},t\right) = C_0\left(t\right)\cdot q_0\left(\mathbf{x_0},t\right), \tag{5.3}$$

where $C_0\left(t\right)\left[\mathrm{ML}^{-3}\right]$ is the concentration of the released pollutant at time t and $q_0\left(\mathbf{x_0},t\right)\left[\mathrm{L}^3\mathrm{T}^{-1}\right]$ is the injection flow rate. Assuming a uniform porosity, $\phi(\mathbf{x}) = \phi$, initial condition $C\left(\mathbf{x},0\right) = 0$, and boundary condition, $C(\infty,t) = 0$, Eq. (5.2) can be solved by the convolution integral

$$C(\mathbf{x},t) = \int_0^t s(\mathbf{x_0},\tau)\, g(\mathbf{x},t-\tau)\; d\tau \tag{5.4}$$

where $g(\mathbf{x},t-\tau)$ is a Kernel function that represents the response at location $\mathbf{x}$ and time t to a pulse injection at the source location $\mathbf{x_0}$ and time τ.

In two-dimensional cases, with uniform flow, $v_y = 0$ and constant dispersion coefficients, the Kernel function can be determined analyticall, and the solution of Eq. (5.4) is

$$\begin{aligned} C(x,y,t) = \int_0^t & s(x_0,y_0,\tau)\, \frac{1}{4\pi\sqrt{D_x D_y\,(t-\tau)}} \\ & \cdot \exp\left[-\frac{((x-x_0)-v\,(t-\tau))^2}{4D_x\,(t-\tau)} - \frac{(y-y_0)^2}{4D_y\,(t-\tau)}\right] d\tau \end{aligned} \tag{5.5}$$

For complex cases in which the flow field is not uniform (for instance, non-isotropic and heterogeneous aquifers), the advection-dispersion equation can not be solved analytically and it is necessary to employ numerical methods. Here, for the second case study for which the analytical solution cannot be used, the flow equation (5.1) is solved using the numerical model MODFLOW (Harbaugh 2005), and the transport equation (5.2) with MT3DMS (Zheng & Wang 1999).

5.3. Analytical case

ES-MDA is applied to an analytical case study with the aim to show the capabilities of the method to simultaneously identify a contaminant source location and its release history in an aquifer. This case requires a small computational time and the results can be compared with a reference solution. This also allows to investigate different configurations of the inverse algorithm, in order to determine the optimal setting to be used for real cases.

The analytical case simulates a pollution event in an infinite homogeneous two-dimensional aquifer, with uniform flow, as result of the injection of a nonreactive contaminant at a point (Butera & Tanda 2003). It is assumed that the water discharge $q_0(\mathbf{x_0}, t)$ is of unit value and small enough such that it does not affect the uniform groundwater flow. Therefore, the release history $s(\mathbf{x_0}, t)$, defined in Eq. 5.3, is equivalent to the concentration history $C_0(t)$. All quantities are considered with unspecified but consistent units. The uniform velocity and the dispersion coefficients are assumed known: $v = 1$, $D_x = 1$ and $D_y = 0.1$. It is considered the same expression for the release function $s_r(\mathbf{x_0}, t)$ used elsewhere (Skaggs & Kabala 1994, Woodbury & Ulrych 1996, Snodgrass & Kitanidis 1997, Butera & Tanda 2003, Butera et al. 2012, Zanini & Woodbury 2016) to define the reference solution

$$\begin{aligned} s_r(\mathbf{x_0}, t) = &\exp\left(-\frac{(t-130)^2}{50}\right) + 0.3\exp\left(-\frac{(t-150)^2}{200}\right) \\ &+ 0.5\exp\left(-\frac{(t-190)^2}{98}\right) \end{aligned} \tag{5.6}$$

The actual source location $\mathbf{x_0}$ is $x_0 = 50$ and $y_0 = 20$. The concentration history has a total duration of 300; it is discretized into 101 intervals with a time step of $\Delta t = 3$ resulting in a total number of parameters to be estimated $N_p = 103$ (the two spatial coordinates plus the 101 temporal solute fluxes). The reference solution, depicted in Fig. 5.1, is used to obtain the reference observations, which are computed by evaluating Eq. (5.5) using numerical integration.

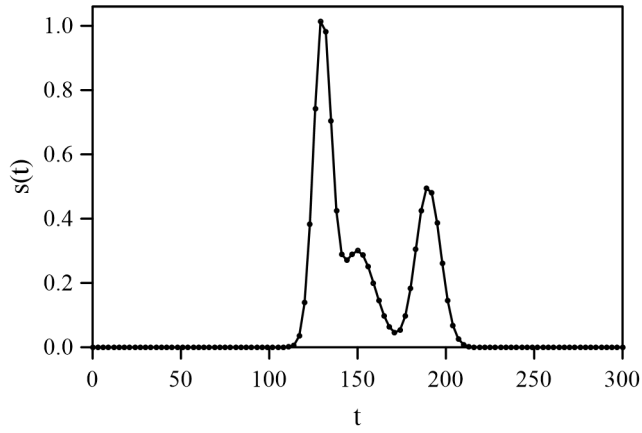

Figure 5.1. Analytical case: reference release history.

Different test cases are carried out to investigate the impact of the observation sampling scheme, ensemble size, covariance localization and inflation techniques. The test cases will be evaluated in terms of equifinality, that is, when different source functions are identified that are consistent with the observations, and in terms of sensitivity to the initial ensemble values. For this purposes, for each test case, 100 experiments were performed to identify the source history changing only the random component of the initial ensemble and the observation measurement errors. At the end of each experiment, the performance of the method is evaluated using the following metrics:

- The Nash-Sutcliffe efficiency criterion (NSE) to evaluate the agreement between the actual and estimated release history:

$$NSE = \left(1 - \frac{\sum_{i=1}^{N_p-2}\left(\overline{X}_i - s_{r,i}\right)^2}{\sum_{i=1}^{N_p-2}\left(s_{r,i} - \overline{s}_r\right)^2}\right) \cdot 100 \tag{5.7}$$

  where $N_p - 2$ is equal to 101, the number of intervals used to discretize $s(t)$; $s_{r,i}$ represents the discretized source function and is the $i-$th actual

amount of released contaminant, $\overline{s}_{r,i}$ is the time average of the reference release history ($\frac{1}{N_p-2}\sum_{i=1}^{N_p-2} s_{r,d}$) and $\overline{X}_i$ is the ensemble mean of the i-th estimated amount of released contaminant ($\frac{1}{N_e}\sum_{j=1}^{N_e} X_i^j$, with X_i^j the final estimate of parameter X_i in realization j). The closer to 100, the better.

- The root mean square error ($RMSE$) between observations and model predictions:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m}\left(D_i - \overline{Y}_i\right)^2}{m}} \tag{5.8}$$

where D_i is the i-th observed concentration and $\overline{Y}_i$ is the ensemble mean of the i-th predicted concentration ($\frac{1}{N_e}\sum_{j=1}^{N_e} Y_i^j$, with Y_i^j the prediction of Y_i in realization j). The closer to zero, the better.

- The spatial distance between the true and estimated source location (L):

$$L = \sqrt{\left(\overline{x}_s - x_0\right)^2 + \left(\overline{y}_s - y_0\right)^2} \tag{5.9}$$

where $\overline{x}_s$ and $\overline{y}_s$ are the ensemble means of the estimated spatial coordinates of the source and (x_0, y_0) is the true source location. The closer to zero, the better.

These metrics are compared with reference threshold values to evaluate the performance of the method. Three cases are considered: i) good performance when the reproduction of the observed concentrations is good, the identification of the source location is good and the identification of the release function is good; ii) equifinality performance, when reproduction of the observed concentrations is good, but neither the source location nor the release function are well identified; iii) poor performance, otherwise:

i) Good performance when

$RMSE < RMSE_{thr}$ and $NSE > NSE_{thr1}$ and $L < L_{thr}$

ii) Equifinality performance when
$RMSE < RMSE_{thr}$ and ($NSE < NSE_{thr2}$ or $L > L_{thr}$)

iii) Otherwise, fail.

The selected threshold values ($RMSE_{thr}$, NSE_{thr1}, NSE_{thr2}, L_{thr}) are reported in Table 5.1, where σ is the standard deviation of the observation errors. Two NSE thresholds are defined to avoid identifying a solution that is close to the good performance threshold as a multiple solution; equifinality is considered only when the NSE is less than 60%. With these criteria, it is possible to define the percentage of successful tests, tests with multiple solutions and failed tests for each case, on the basis of the 100 experiments.

Table 5.1. Threshold values used to define test criteria.

$RMSE_{thr1}[-]$	4σ
$NSE_{thr1}[\%]$	70
$NSE_{thr2}[-]$	60
$L_{thr}[-]$	5

5.3.1. Impact of the concentration sampling scheme

The effect of the spatial distribution of the observation points. For this case, a large ensemble was used to avoid the need of using localization or inflation techniques in the implementation of ES-MDA. The observation network geometry, displayed in Fig. 5.2, are:

A. Concentrations collected at 2 monitoring points, located on the same line as the source ($y = 20$) at (150, 20) and (200, 20), and 31 sampling times from $T = 0$ up to $T = 450$ with a time step $\Delta t = 15$. The total number of observations is $m = 2 \cdot 31 = 62$.

B. Concentrations collected at 21 monitoring points distributed on the same

line of the source ($y = 20$) at uniform intervals between $x = 90$ and $x = 290$; only one observation from each location at time $T = 300$. The total number of observations is $m = 22 \cdot 1 = 22$.

C. Concentrations collected at 4 monitoring points distributed on the same line of the source ($y = 20$) at x-coordinates (80, 115, 150, 185) and the same 31 sampling times of set A. The total number of observations is $m = 4{\cdot}31 = 124$.

D. Concentrations collected at 4 monitoring points vertically distributed on the line $x = 150$ and at y-coordinates (11, 16, 21, 26); the sampling times are the same as for sets A and C. The total number of observations is $m = 4 \cdot 31 = 124$.

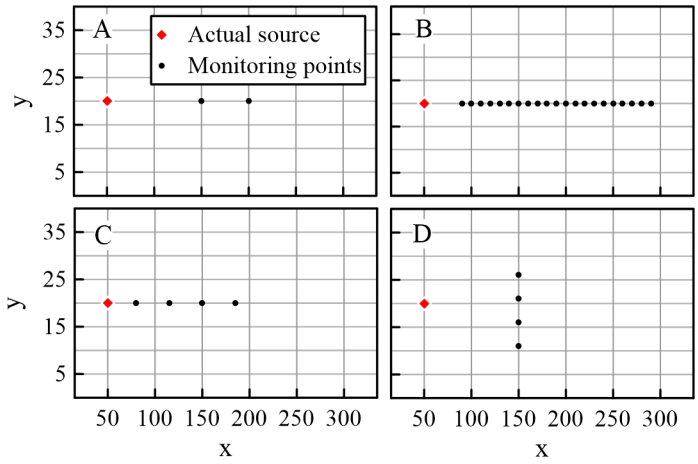

Figure 5.2. Analytical case: location of the measurement points for sets A, B, C and D; the red diamond is the actual source location.

The observation error ε is random and normally distributed with zero mean and variance $5{\cdot}10^{-8}$ for all the performed tests. The initial ensemble of parameters is composed of 1000 realizations. The realizations of the source coordinates are random values selected in the range [5, 80] for x and [10, 30] for y. The realizations

of the release history are normal functions described by the following expression:

$$f(t) = \Delta + \Gamma \cdot \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)}, \tag{5.10}$$

where t is the time, Δ is a base amount of released concentration, Γ is the volume under the Gaussian function of mean μ and variance σ^2. These coefficients are selected randomly from uniform distributions, $\Delta \in \mathcal{U}[1{\cdot}10^{-10}, 1{\cdot}10^{-3}]$, $\Gamma \in \mathcal{U}[10, 40]$, $\mu \in \mathcal{U}[89, 210]$ and $\sigma \in \mathcal{U}[6, 59]$. The ES-MDA is run with 10 iterations and a decreasing series of α values following the sequence [113.33; 75.55; 50.37; 33.58; 22.39; 14,92; 9.95; 6.63; 4.42; 2.95].

Table 5.2 summarizes the results of the four test cases, T denotes the percentage of successful tests over the 100 synthetic experiments and E indicates the percentage of synthetic experiments in which equifinality is detected.

Table 5.2. ES-MDA performance for observations sets A, B, C and D and ensemble size N_e=1000. T indicates the percentage of successful tests and E the percentage of tests that present equifinality.

A	B	C	D
T:10%	T:19%	T:21%	T:98%
E:53%	E:34%	E:12%	E:0%

The observation network geometry greatly impacts the final results. The synthetic experiments that give reliable solutions (NSE>70 and L<5) are less than 21% for observation sets A, B and C. Furthermore, equifinality occurs in large proportions for cases A and B, and to a lesser extent for case C. Only in case D, the ES-MDA is able to identify successfully the source location and the release function without equifinality.

5.3.2. Impact of the ensemble size and application of localization and inflation techniques

The test cases designed to investigate the impact of the ensemble size, covariance localization and inflation techniques make use of the observation set D. Five ensemble sizes are tested N_e of 1000, 500, 250, 100 and 50 with and without covariance corrections. The number of iterations, α values, and distributions used to generate the initial ensembles are the same ones used in the previous section. Covariance localization is applied using the coefficients b_s equal to 210 and b_t equal to 300. The factor r used for the covariance inflation is equal to 1.01. The results obtained from each set of 100 synthetic experiments are reported in Table 5.3. The ES-MDA performs better for increasing ensemble sizes and when covariance inflation and localization techniques are applied. The percentage of successful tests is high for large ensembles, with even better numbers when covariance corrections are applied. The presence of equifinality is detected when the ensemble size reduces, but the corrections on the algorithm help to reduce it. The effects of covariance and inflation techniques are more evident for small ensemble sizes; considering N_e equal to 100, the percentage of successful tests is 46% for the experiments without corrections and 64% for those with corrections; multiple solutions are detected for 43% of the experiments without corrections and for 14% of those with corrections. The tests computed with the smaller ensemble size (N_e=50) lead to unsatisfactory results with a percentage of successful tests lower than 45% and a high probability of equifinality.

Table 5.3. ES-MDA performance for observation set D and ensemble sizes of 1000, 500, 250, 100 and 50, with and without corrections on the covariance calculation. T indicates the percentage of succesful tests and E the percentage of tests that present equifinality.

N_e	without corrections	with corrections
1000	T:98% E:0%	T:100% E:0%
500	T:85% E:8%	T:96% E:0%
250	T:71% E:19%	T:87% E:4%
100	T:46% E:43%	T:64% E:14%
50	T:20% E:60%	T:45% E:29%

The results of a test performed with a small ensemble size of 100 realizations and with corrections in the computation of the covariance. Among the 100 synthetic experiments, it is selected as the best estimate of the release function the median of the successful tests, and the set of successful tests to build uncertainty intervals about the median is used. In Fig. 5.3 the reference solution and the ensemble mean with its 95% uncertainty interval are depicted. Fig. 5.4 shows the comparison between observed and predicted concentrations at observation locations. The ES-MDA reproduces quite well the release history and the source location estimate is very close to the true one (x_0=50, y_0=20). The NSE is 80.46% and the ensemble means of x and y coordinates are, respectively, equal to 52.66 ($\pm$1.78, 95% uncertainty interval) and 20.00 ($\pm$0.06, 95% uncertainty interval). The test leads to a good match between observations and predictions with an RMSE at the last iteration equal to $3.3{\cdot}10^{-4}$ and a narrow 95% uncertainty interval.

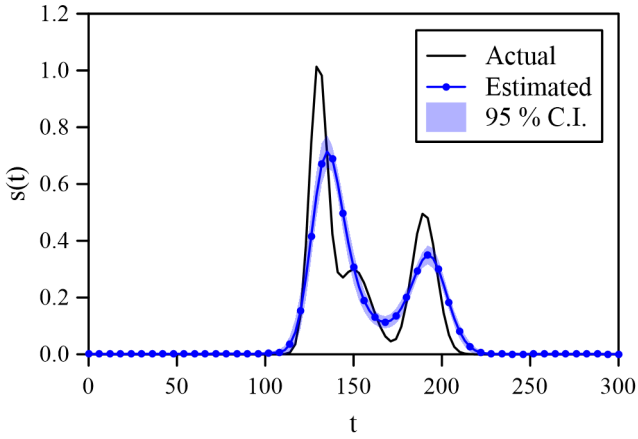

Figure 5.3. Analytical case: actual and estimated release history with 95% uncertainty interval resulting from a test performed with $N_e = 100$ and observation set D.

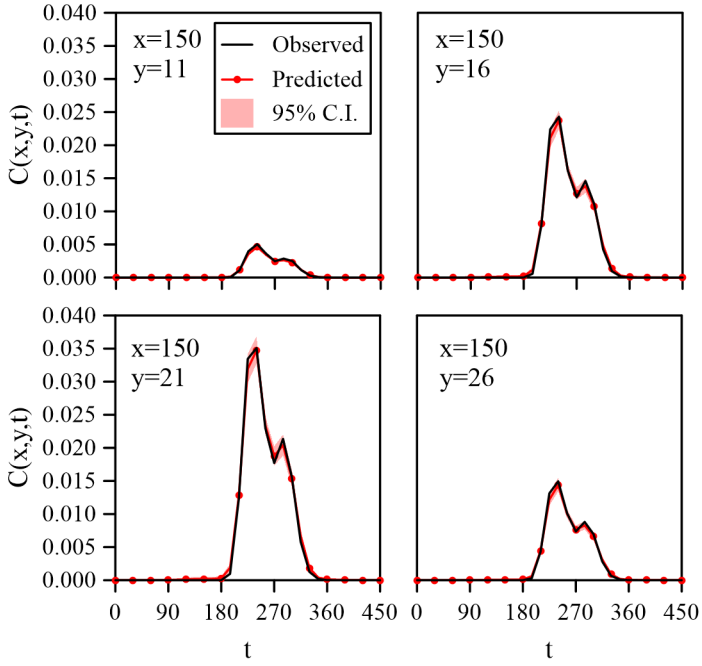

Figure 5.4. Analytical case: observed and predicted concentrations with 95% uncertainty interval.

5.4. Experimental case

The second case study uses a laboratory experimental dataset following the work by Cupola et al. (2014). The experimental device, depicted in Fig. 5.5, is a sandbox that reproduces an unconfined aquifer characterized by two-dimensional flow in a vertical plane. The sandbox has external dimensions of 120 cm × 14 cm × 73 cm and it is made of three parts along the longitudinal direction: upstream and downstream tanks and an internal chamber of 95 cm × 10 cm × 70 cm, which contains the porous media consisting of glass beads with diameter in the range between 0.75 mm and 1 mm.

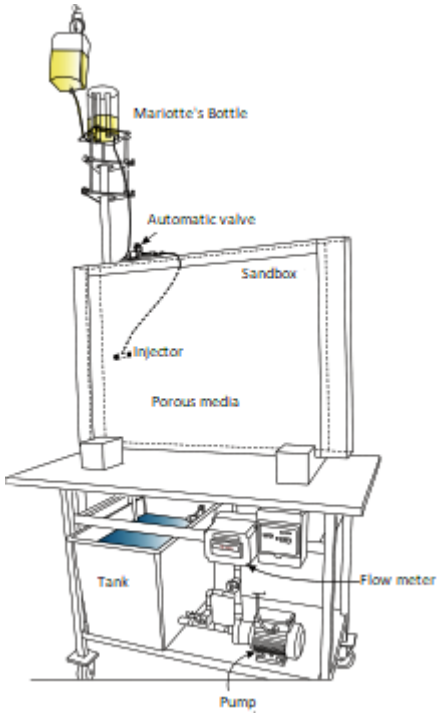

Figure 5.5. Sketch of the experimental device. (Image from: Cupola, F. (2016), Theory and application of inverse problems in groundwater: numerical, laboratory and field studies., Doctoral thesis thesis, Università degli Studi di Parma.)

The flow is governed by constant upstream and downstream water levels equal to 59.9 cm and 53.6 cm above the horizontal bottom of the tank, respectively. Fluorescein sodium salt was used as tracer solution and it was injected at a variable mass rate through an injector located in the upstream part of the sandbox at coordinates $x = 14.25$ cm and $y = 32.75$ cm, that extends through the entire thickness of the sandbox. The test had a duration of 2200 s; the injection started at time 310 s and ended at 1800 s; the concentration of the fluorescein sodium salt is constant and equal to 20 mg·l^{-1}, while the flow rate changes over time. The resulting mass rate ranges from 0 to about 55 μg·l^{-1} and presents three peaks of different magnitude (Fig. 5.6).

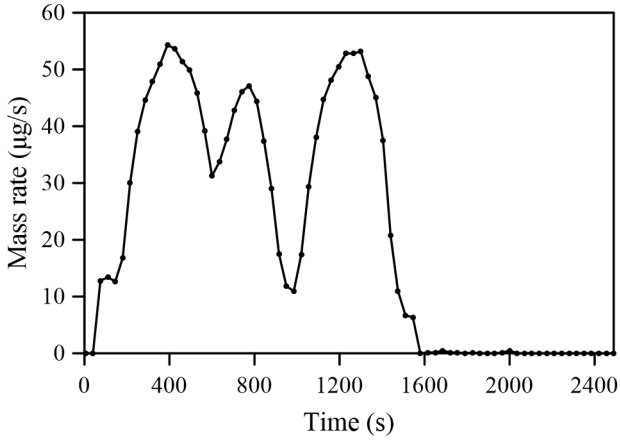

Figure 5.6. Experimental case: reference release history.

The observed concentrations are recorded over the entire sandbox by taking pictures with a digital camera and then converting luminosity into concentration through image processing techniques (for more details see Citarella et al. (2015)). Modeling is performed in two dimensions, since no lateral movement orthogonal to the sandbox plane is expected. A comparison between the results obtained with a two-dimensional model and a three-dimensional one is reported by Uribe-Asarta

(2019), showing no differences between the two models.

The inverse methodology requires a calibrated numerical model able to describe as accurately as possible the forward process. Groundwater flow was modeled with MODFLOW 2005 (Harbaugh 2005) and mass transport with MT3DMS (Zheng & Wang 1999). The effect of the injection on the background flow is not negligible; therefore, a transient flow model is considered. The numerical model was preliminary calibrated using the ES-MDA procedure.

5.4.1. Calibration of the numerical model

The calibration of the numerical model has been performed through the ES-MDA procedure considering the release history and source location known; the hydraulic and transport parameters are the unknown of the inverse procedure and concentrations collected at several monitoring points and times are used as observations.

The experimental device aimed to reproduce a homogeneous isotropic field; however, during the laboratory experiment, some disturbances may lead to heterogeneity and anisotropy. One of the main problem is the presence of trapping air; in fact, it can modify the flow and generate errors in the imaging acquisition system. The observed concentrations are obtained through image processing techniques converting luminosity into concentration: air bubbles reduce the light intensity causing corruptions in the analysis of the images. With the aim to minimize the appearance of bubbles air inside the pores, the central tank of the sandbox was packed under saturated conditions layer by layer; but the problem cannot be completely eliminated. Another source of noise can be the presence of the injector that locally disturbs the field. Moreover, the experimental configuration may change over time due to compaction or the development of preferential lines, since the flow always has the same direction.

For these reasons, the calibration process has been performed considering different field configurations. The unknown transport parameters are the longitudinal

dispersivity and the transverse dispersivity, the hydraulic parameters to be estimated depends on the analyzed case. Three hydraulic conductivity fields have been considered:

- Homogeneous and isotropic field: the hydraulic parameter to be estimated is the constant hydraulic conductivity.
- Homogeneous and anisotropic field: the hydraulic parameters to be estimated are the hydraulic conductivity and the anisotropy ratio.
- Heterogeneous and anisotropic field: the investigated hydraulic parameters are the hydraulic conductivity in some points of the aquifer and the hydraulic conductivity anisotropy ratio.

The effective porosity of the glass beads is fixed at 0.37 and the specific storage coefficient at $10^{-}4\text{cm}^{-3}$. Many observations have been considered to perform the calibration through ES-MDA in order to well characterize the evolution of the contaminant plume; 55 monitoring points are equally distributed on 5 lines perpendicular to the direction of the plume progress and 24 monitoring points are distributed on the the same flow line of the source. Concentrations are recorded, for each monitoring points, at 45 time steps resulting in a total number of observations m=3554. For all cases, ES-MDA was performed with a random observation error ε normally distributed with zero mean and variance $7{\cdot}10^{-3}$ $(\text{mg/l})^2$, 10 iterations and a decreasing α obtained with α_{geo}=1.5 (Eqs. 1.3 and 1.4). The update step is performed in the transformed space; the longitudinal dispersivity, the transverse dispersivity and the anisotropy ratio of conductivity are transformed using the root square transformation. The modified log-transformation is applied to transform the hydraulic conductivity parameters; this allows to constrain the hydraulic conductivity to acceptable values; the selected interval is [0.5, 0.9] cm/s (Eqs. 1.10 and 1.11).

Homogeneous and isotropic field

The calibration of the numerical model is initially performed considering the field homogeneous and isotropic. The total number of parameters to be estimated are three ($N_p = 3$): constant hydraulic conductivity, longitudinal dispersivity and transverse dispersivity. The ensemble size is N_e=40 and the initial realizations are random values selecting from a uniform distribution in the following ranges: [0.6, 0.8] cm/s for the hydraulic conductivity; [0.05, 0.2] cm for the longitudinal dispersivity and [0.01, 0.5] cm for the transverse dispersivity. Covariance inflation (r=1.01, Eq. 1.19) and linear relaxation (w=0.2, Eq. 1.9) have been applied.

Table 5.4 summarizes the estimated parameters of the flow and transport models at last ES-MDA iteration. Figure 5.8 shows the comparison between observed and predicted concentrations at 55 monitoring points; it can be seen that the misfit between them is very large. The RMSE at the last iteration is 3.91 mg/l, which denotes that the calibrated numerical model does not accurately describe the forward processes.

In Figure 5.7, the experimental and modeled plume at time T=1500 s after the beginning of the injection are compared. The experimental concentration field results from the analysis of the raw images, the estimated one is the output of the numerical model, developed with MODFLOW and MT3D and performed using the estimated parameter reported in Table 5.4, at the same time. It is clear that the numerical model is not able to accurately reproduce the flow and transport in the sandbox. In particular, the model overestimates the transverse extension of the plume and does not well reproduce the plume propagation direction. The direction of the experimental plume is mainly along the injector axis, while the simulated plume propagates along a different direction governed by the hydraulic gradient. It should be noted that the hydraulic and transport parameters to be estimated do not affect the plume propagation direction, therefore it is not possible

to well reproduce the experimental data considering the field homogeneous and isotropic.

Table 5.4. Estimated transport and hydraulic parameters, assuming the field homogeneous and isotropic; the ensemble mean and 95% confidence interval are reported

	Ensemble mean	95% C.I.
Hydraulic conductivity (cm/s)	0.671	$3{\cdot}10^{-4}$
Longitudinal dispersivity (cm)	0.158	0.004
Transverse dispersivity (cm)	0.173	0.005

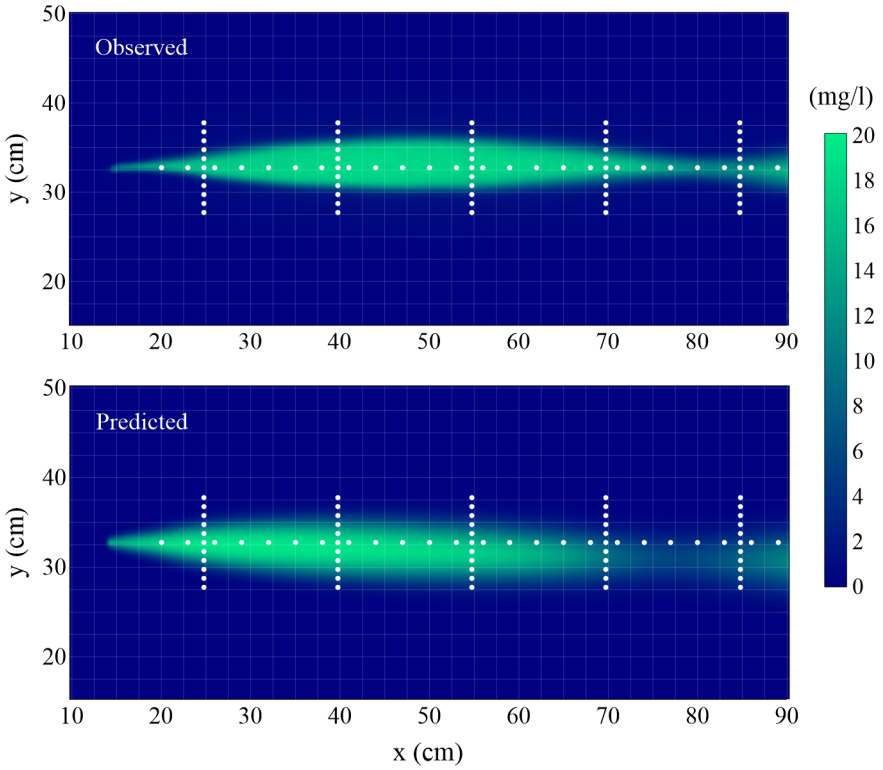

Figure 5.7. Concentration field observed and predicted at time 1500 s after the start of the injection. The hydraulic conductivity field is considered homogeneous and isotropic. The white dots denote the monitoring points used to perform ES-MDA.

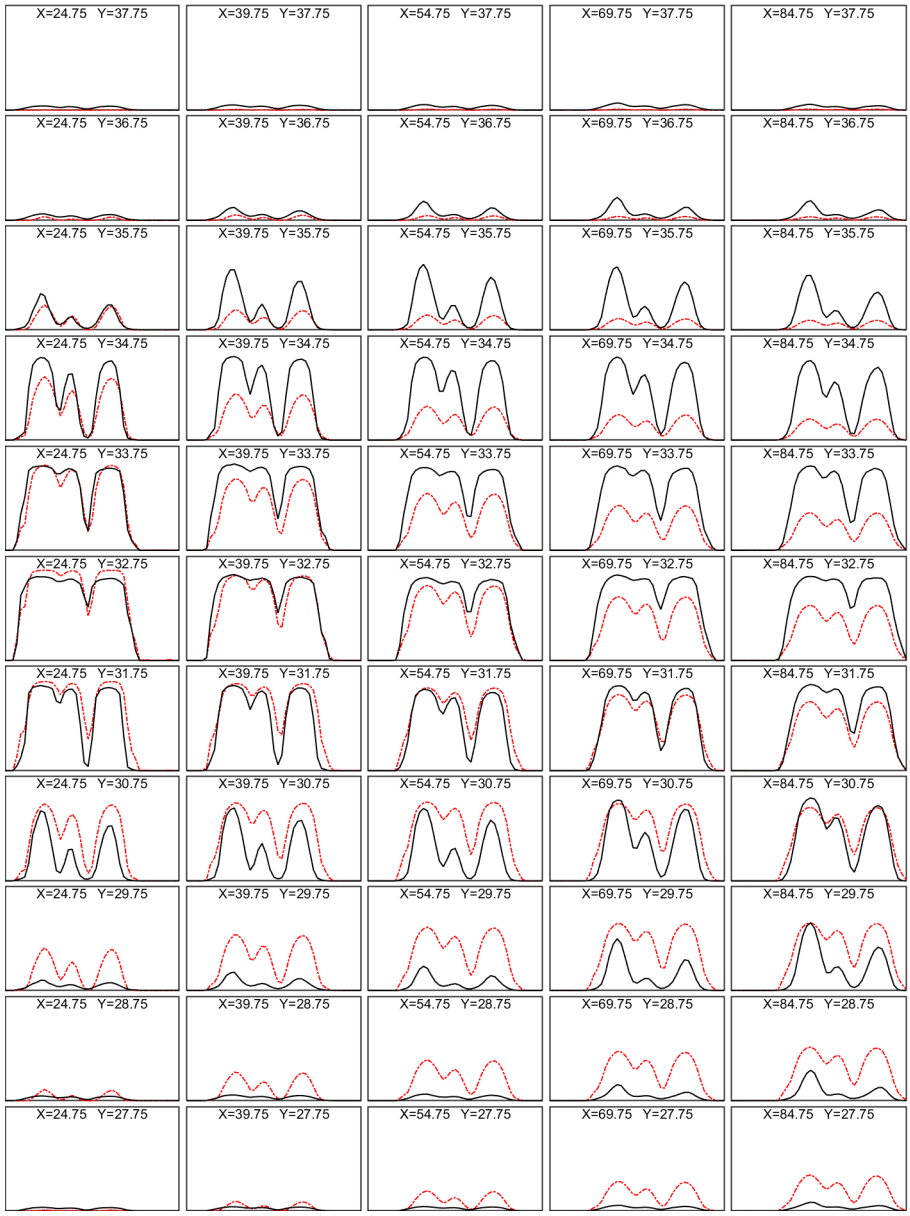

Figure 5.8. Observed (black line) and predicted (red dashed line) concentrations, assuming the field homogeneous and isotropic. X-axis is time from 0 to 2200 s, where time 0 s represents the time at which injection starts. Y-axis is concentration from 0 to 23 mg/l.

Homogeneous and anisotropic field

For the test performed assuming the field homogeneous and anisotropic, the total number of unknown parameters are four ($N_p = 4$): constant hydraulic conductivity, anisotropy ratio of conductivity, longitudinal dispersivity and transverse dispersivity. The ensemble size is the same as the previous test (N_e=40) as are the ranges used to generate the initial ensemble of hydraulic conductivity, longitudinal dispersivity and transverse dispersivity. The initial realizations of the anisotropy ratio of conductivity (kh/Kv) are random values selected from a uniform distribution in the range [0.9, 6]. Covariance inflation (r=1.01, Eq. 1.19) and linear relaxation (w=0.2, Eq. 1.9) have been applied.

The estimated parameters of the flow and transport models are reported in Table 5.5. Figure 5.10 shows the comparison between observed and predicted concentrations at the 55 monitoring points distributed on the five lines perpendicular to the direction of the plume progress. In Figure 5.9, the experimental plume at time T=1500 s, after the beginning of the injection, is compared with the plume obtained at the same time from the numerical model designed with the parameters reported in Table 5.5. The experimental plume is quite well reproduced and the misfit between observations and predictions is not large (RMSE is equal to 1.77 mg/l). However, the estimated anisotropy ratio of conductivity, which allows to reach this good agreement, is very large (Kh/Kv=18.30) and considered not acceptable to describe the experimental field.

Table 5.5. Estimated transport and hydraulic parameters, assuming the field homogeneous and anisotropic; the ensemble mean and 95% confidence interval are reported

	Ensemble mean	95% C.I.
Hydraulic conductivity (cm/s)	0.677	$3{\cdot}10^{-4}$
Vertical anisotropy of conductivity (Kh/Kv)	18.30	0.20
Longitudinal dispersivity (cm)	0.098	0.004
Transverse dispersivity (cm)	0.154	0.006

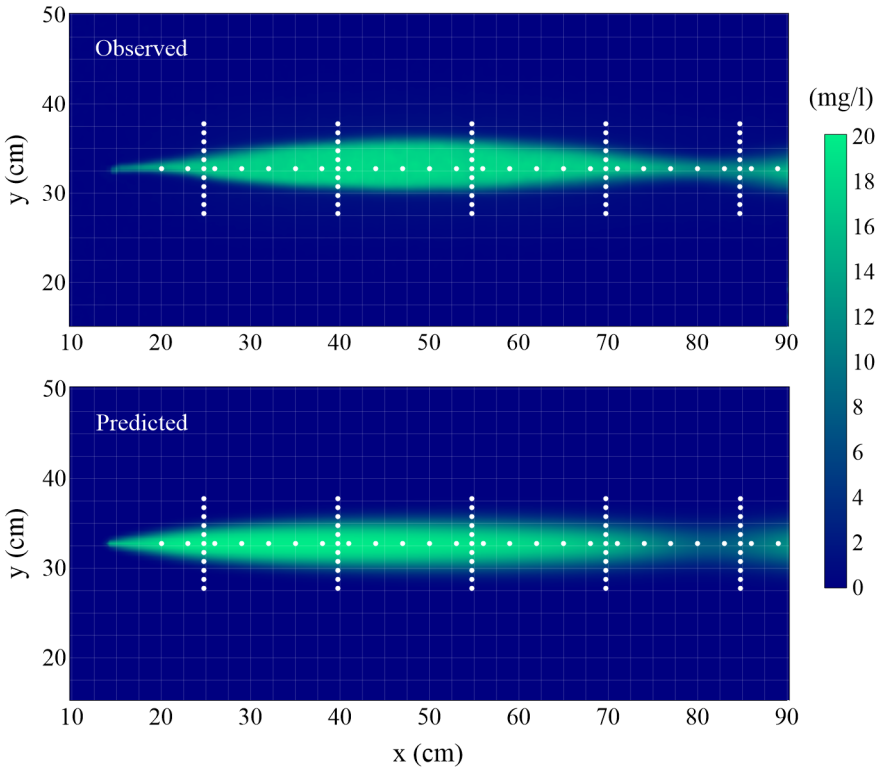

Figure 5.9. Concentration field observed and predicted at time 1500 s after the start of the injection. The hydraulic conductivity field is considered homogeneous and anisotropic. The white dots denote the monitoring points used to perform ES-MDA.

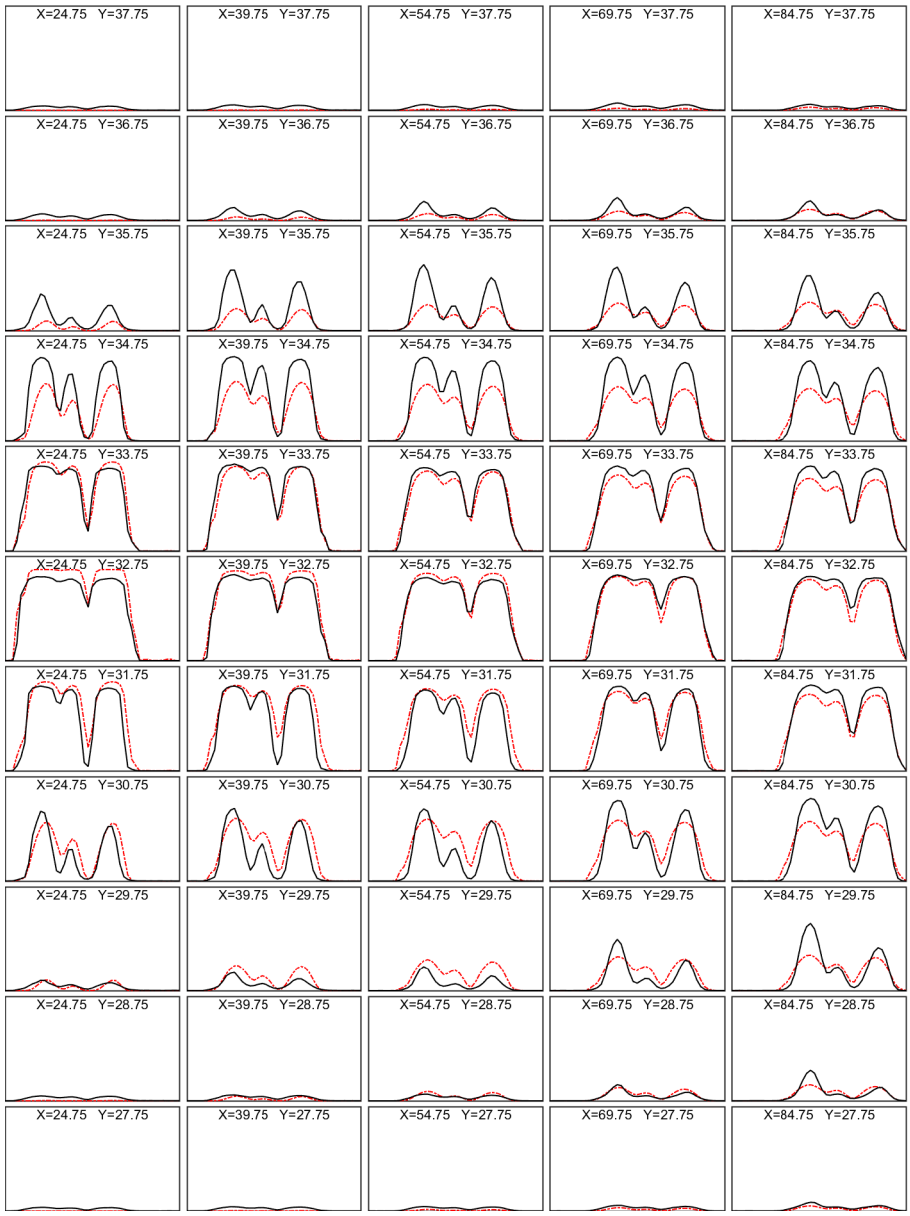

Figure 5.10. Observed (black line) and predicted (red dashed line) concentrations, assuming the field homogeneous and anisotropic. X-axis is time from 0 to 2200 s, where time 0 s represents the time at which injection starts. Y-axis is concentration from 0 to 23 mg/l.

Heterogeneous and anisotropic field

The third configuration considers the field slightly heterogeneous and anisotropic; the unknowns are the log-conductivity field, the hydraulic conductivity anisotropy ratio, the longitudinal dispersivity and the transverse dispersivity. The log-conductivity field is estimated using the pilot points method; it consists of estimating the values of the hydraulic conductivity in a finite number of points and then interpolating them to obtain the solution over the whole model domain. This reduces the number of parameters to be estimated and consequently the computational burden compared to a full parameterization approach. Ordinary kriging has been applied as method of interpolation and a linear variogram has been used to perform the kriging, since no variogram information are available and the principle of parsimony has been chosen. During the calibration, 72 pilot points concentrated in the area of influence of the plume have been considered (Figure 5.12), resulting in a total number of parameters to be estimated $N_p = 75$. The flow chart in Figure 5.11 summarizes the procedure to couple ES-MDA and the pilot points method.

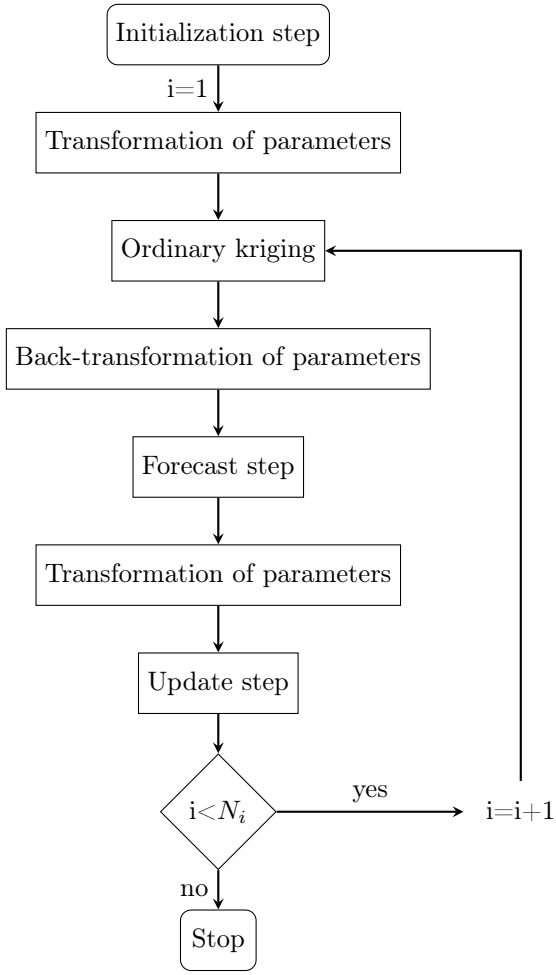

Figure 5.11. Flow chart of ES-MDA for the model calibration using the pilot points method and ordinary kriging as interpolation technique.

For this case, the ensemble size is N_e=80; the initial realizations of the longitudinal dispersivity, the transverse dispersivity and the anisotropy ratio of conductivity (kh/Kv) are random values selected from uniform distributions in the same range of the previous cases. The initial ensemble of the hydraulic conductivity fields is composed of 80 (ensemble size) different homogeneous fields; for each

realization, the value of the hydraulic conductivity is constant for all the pilot points. The hydraulic conductivity of each initial homogeneous field is a random value selected from a uniform distribution over the range [0.6, 0.8] cm/s. Covariance inflation (r=1.01, Eq. 1.19), covariance localization (b_s=200 and b_t=2500, Eq. 1.18) and linear relaxation (w=0.2, Eq. 1.9) have been applied. Covariance localization is mandatory, for this case, to deviate from the initial homogeneous fields and obtain heterogeneous field.

The estimated parameters of the flow and transport models are reported in Table 5.6. The estimated hydraulic conductivity field and its variance in log scale are reported in Figures 5.12 and 5.13, respectively. The calibration process results in a field that reproduces the sandbox with lower values of hydraulic conductivity in the lower right part of the experimental field and higher values in the upper part. This can be ascribed to nonuniform compaction related to the constant direction of propagation. It should also be noted that the uncertainty in the estimated hydraulic conductivity is very high for the zone outside the area of influence of the plume due to the design of the pilot points located where they provide most information.

Figure 5.15 shows the comparison between observed and predicted concentrations at the 55 monitoring points distributed on the lines perpendicular to the direction of the plume progress. In Figure 5.14, the experimental plume at time T=1500 s, after the beginning of the injection, is compared with the output of the numerical model calibrated which reproduces a heterogeneous and anisotropic field. The observed concentrations are well reproduced (RMSE= 1.30 mg/l) and there is a good match between the experimental and numerical plume at the fixed time.

In conclusion, the third configuration, which assumes that the hydraulic conductivity field is heterogeneous and anisotropic, is considered the best one to describe the flow and transport processes.

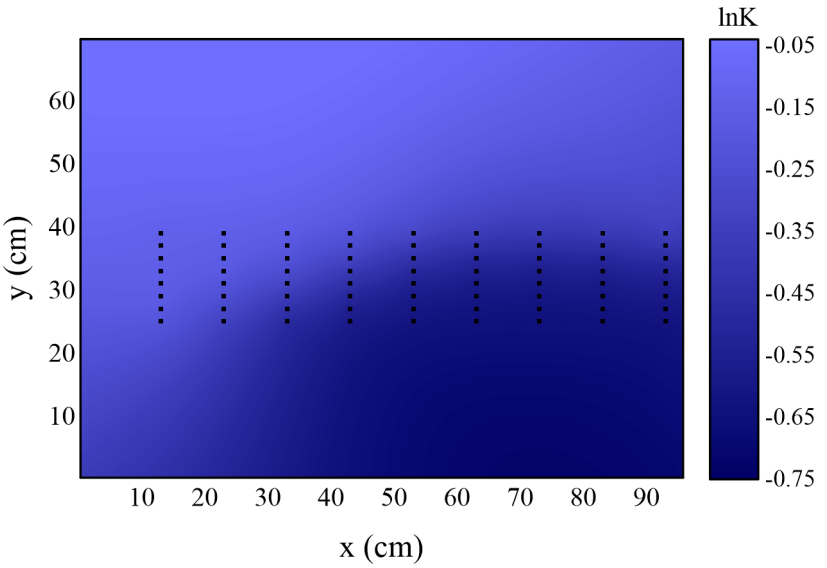

Figure 5.12. Hydraulic conductivity field in log scale. The black squares denote the pilot points used to perform the Kriging.

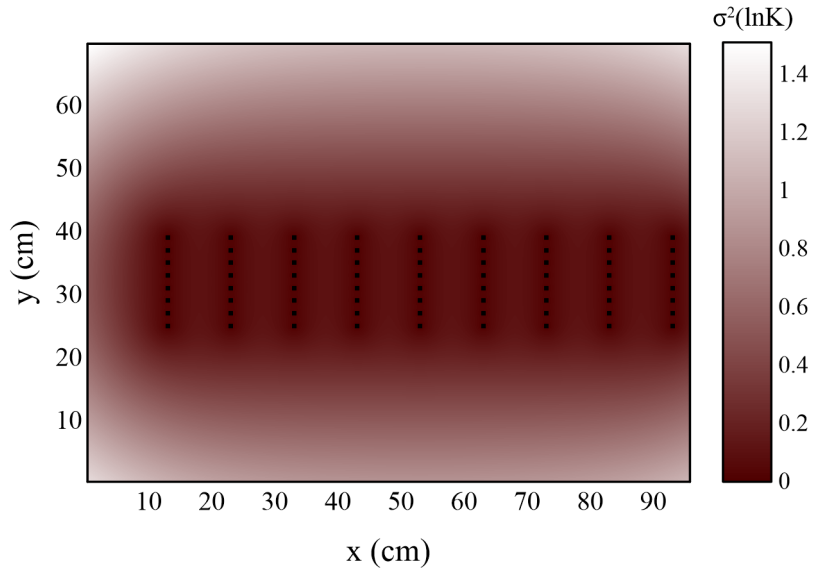

Figure 5.13. Variance of the hydraulic conductivity field in log scale. The black squares denote the pilot points used to perform the Kriging.

Table 5.6. Estimated transport and hydraulic parameters, assuming the field heterogeneous and anisotropic; the ensemble mean and 95% confidence interval are reported

	Ensemble mean	95% C.I.
Vertical anisotropy of conductivity (Kh/Kv)	3.27	0.03
Longitudinal dispersivity (cm)	0.178	0.003
Transverse dispersivity (cm)	0.065	0.001

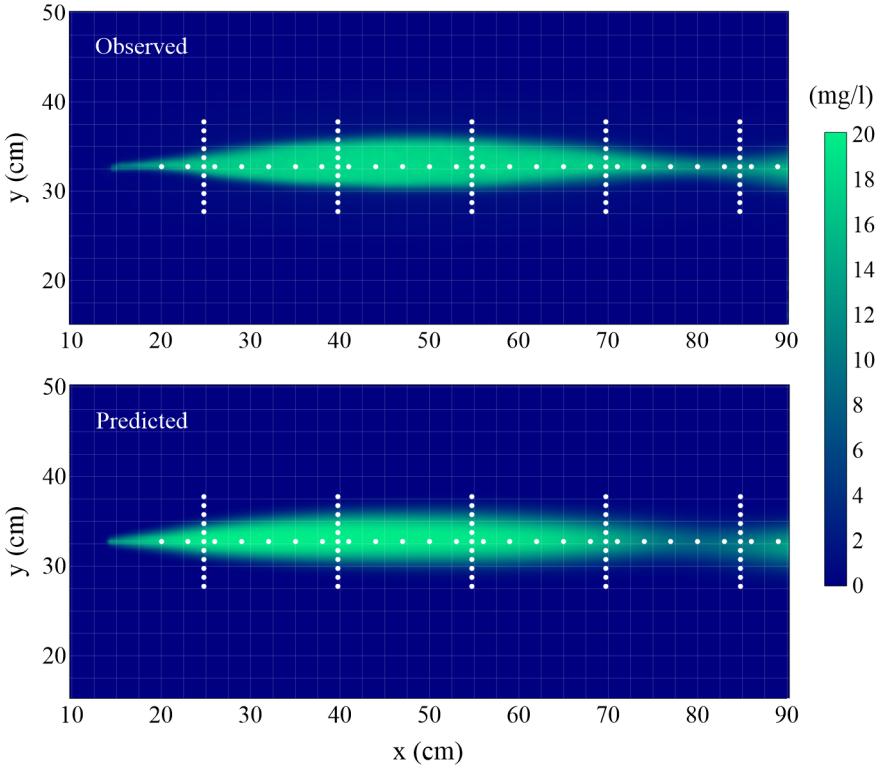

Figure 5.14. Concentration field observed and predicted at time 1500 s after the start of the injection. The hydraulic conductivity field is considered heterogeneous and anisotropic. The white dots denote the monitoring points used to perform ES-MDA.

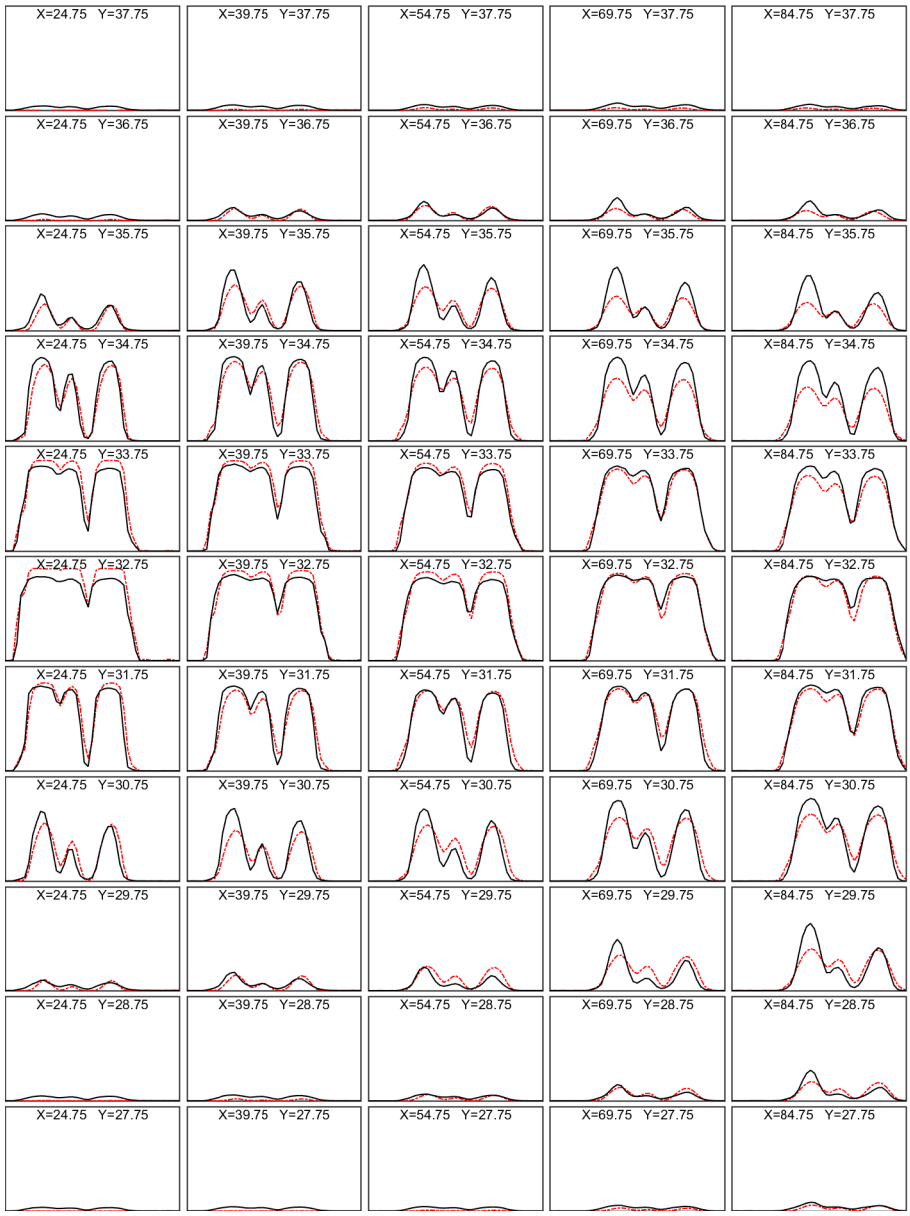

Figure 5.15. Observed (black line) and predicted (red dashed line) concentrations, assuming the field heterogeneous and anisotropic. X-axis is time from 0 to 2200 s, where time 0 s represents the time at which injection starts. Y-axis is concentration from 0 to 23 mg/l.

5.4.2. Identification of the release history and the source location

Since the concentration of the contaminant is known, the estimation of the release history is limited to identifying the injected flow rate. The release duration is discretized into 72 intervals with a time step of $\Delta t = 3$ s resulting in a total number of parameters $N_p = 74$, of which two are the spatial coordinates of the source. The initial ensemble of parameters is made up of 81 realizations ($N_e = 81$); the spatial coordinates of the source are random values selected from uniform distributions $x \in U[5, 30]$ cm, and $y \in U[30, 34]$ cm. The initial realizations of the injected flow rate history follow expression Eq. (5.10), with parameters selected randomly from the following uniform distributions, $\Delta \in \mathcal{U}[1 \cdot 10^{-10}, 1{\cdot}10^{-1}]$, $\Gamma \in U[800, 1000]$, $\mu \in U[490, 1400]$ and $\sigma \in U[60, 365]$. The four monitoring points are vertically distributed on the line $x = 54.75$ cm and at y-coordinates 29.00, 32.75, 34.75 and 36.75 cm. For each monitoring point, the observed concentrations are recorded at 45 sampling times from $T = 0$ s to $T = 2200$ s (total number of monitoring data is $m = 180$). The random measurement error ε is assumed normally distributed with zero mean and variance $1{\cdot}10^{-2}$ ($\mathrm{mg}^2{\cdot}\mathrm{l}^{-2}$). The ES-MDA with 6 iterations and decreasing α=[63.0; 31.5; 15.8; 7.88 3.9; 2.0] is used for the inversion. Covariance localization and covariance inflation are applied using the coefficients b_s=200, b_t=2500 and r=1.01, and linear relaxation with the coefficient w=0.1.

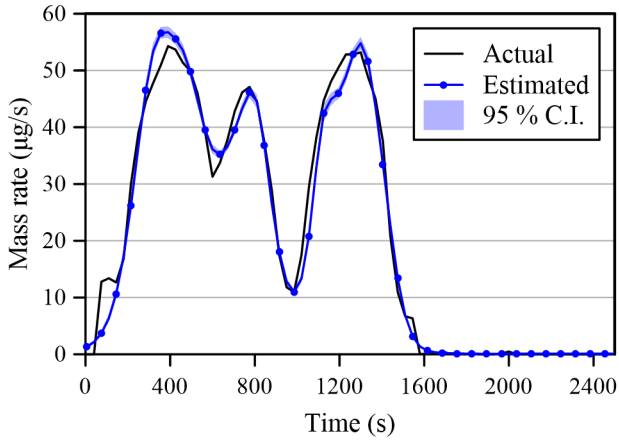

Figure 5.16. Experimental case: actual and estimated release history with 95% confidence interval. Time 0 s represents the time at which injection starts.

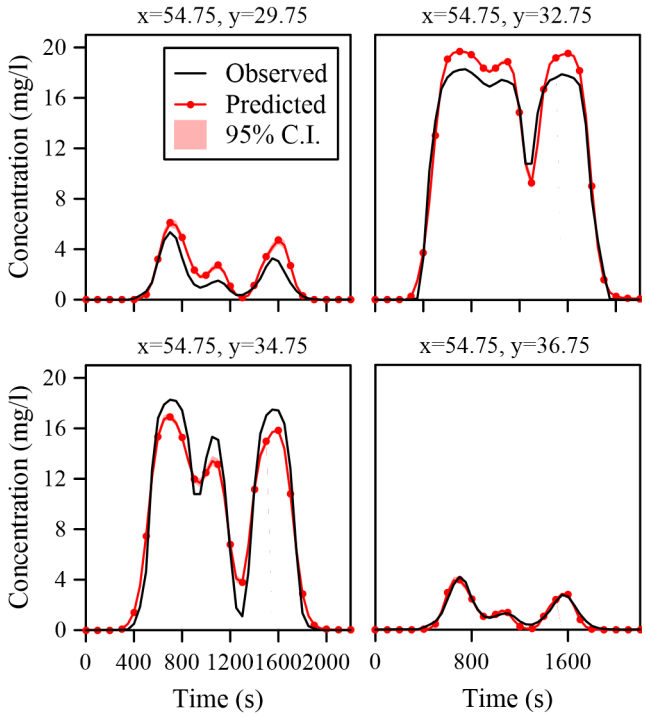

Figure 5.17. Experimental case: observed and predicted concentrations with 95% confidence interval. Time 0 s represents the time at which injection starts.

Fig. 5.16 shows the results of the experimental case; the ensemble mean of the release history with its 95% confidence interval and the true solution are depicted. ES-MDA leads to a good agreement between the two curves, the NSE is 98.34% and with a satisfactory representation of peak magnitudes and times. The ensemble means of the x and y coordinates of the source are, respectively, equal to 14.71 cm ($\pm$0.45, 95% uncertainty interval) and 32.91 ($\pm$0.14, 95% confidence interval); the distance between the true and estimated source location is less then 0.5 cm. In Fig. 5.17 the experimental and predicted observations are compared. The retrieved source parameters reproduce quite well the observed concentrations with a narrow 95% uncertainty interval; the RMSE at the last iteration is equal to 0.96 mg/l, which is comparable with the experimental observation errors.

5.5. Concluding remarks

In this chapter, the Ensemble Smoother with Multiple Data Assimilation (ES-MDA) is proposed for the simultaneous identification of the source location and the release history of a groundwater contamination event from observed sparse concentration data collected downstream from the spill. The procedure is tested by means of an analytical case study and an experimental one.

The analytical case serves to demonstrate the capability of ES-MDA to solve this type of inverse problem and to analyze the impact of the different settings on the final identification. The impact of the observation network geometry and density, ensemble size, covariance and inflation techniques and also the effect of different sets of initial realizations are investigated. The aim is to find out a configuration that leads to a reliable solution and mitigates the ill-conditioness of the inverse procedure. Equifinality is analyzed in the analytical case, finding that there are some network geometries that may lead to acceptable results (in terms

of reproduction of the observed concentrations) but with very different release functions.

The effect of the observation network geometry and density is evaluated considering four sets of observed concentrations, a large ensemble size (N_e=1000) and the other factors being the same. The results show that location, time and number of observations significantly impact the final solution of ES-MDA; for the sets in which the observations are located in a line parallel to the main flow direction, the percentage of successful tests is low and equifinality is detected. Instead, for the set with the observations in a line orthogonal to the main flow direction, the number of successful tests is 98% and the algorithm simultaneously estimates the release history and the source location.The observation points located in a line orthogonal to the main flow directions are more informative than those located along the same line. In the latter case, it is easy to think of multiple solutions that should lead to the same observations, for instance, by estimating the source location in the direction orthogonal to flow symmetrically with respect to the line of observations. This indicates the importance of a good design of the observation network, since if observations provide poor information, the ill-posed inverse problem is difficult to solve and the impact of random factors increases; it is also noteworthy that, in real cases, only a limited number of concentration measurements are available given the field sampling costs; for this reason, an optimal design of new monitoring points has a great relevance.

The observation set orthogonal to the flow direction is used to check the effect of the ensemble size and the application of covariance localization and covariance inflation techniques in the performance of the ES-MDA. The results show that the ES-MDA works better when large ensembles and the correction on the algorithm are used, demonstrating the capability of the proposed spatio-temporal iterative localization to improve the ES-MDA performance. The percentage of successful tests increases with the ensemble size and the covariance corrections and, at the

same time, the chances that equifinality happens decrease. Covariance inflation and, in particular, covariance localization, overcome the undersampling problems noticed in the ensemble-based methods; and for this reason, their effects are more evident for small ensemble sizes. The tests performed with an ensemble size of 50 realizations lead to unreasonable results with a low percentage of passed tests and a high percentage of tests with multiple solutions. It is suggested to use, for this type of problems, ensemble sizes greater than the number of unknown parameters to identify.

The experimental case study uses real data collected through a laboratory test. The experimental device is a sandbox that reproduces an unconfined aquifer under controlled conditions; it allows to validate the ES-MDA methodology in a real test case.

ES-MDA was preliminary used to calibrate the numerical model required for the inverse procedure. The hydraulic and transport parameters have been estimated on the basis of many observed concentrations and assuming the release history known. Three different configurations of the hydraulic conductivity field have been investigated: homogeneous and isotropic, homogeneous and anisotropic and heterogeneous and anisotropic. The results show that the best hydraulic conductivity field to reproduce the flow and transport processes inside the sandbox is the heterogeneous and anisotropic one.

Once the numerical model has been calibrated, the study proceeds with the estimation of the source location and release history of the contaminant. The algorithm parameters, such as the monitoring network and the ensemble size, were chosen after the results of the analytical study. For this case, the initial ensemble of source coordinates has been generated considering a limited suspect area, which guarantees that all the realizations of the ensemble are representative. This decision was taken based on preliminary tests performed with large suspect areas. Even if it is not mandatory that the initial ensemble contains the solution,

a well designed ensemble helps to reach better results.

The results prove the capability of ES-MDA to solve this type of inverse problem in a real cases, when the available observations are usually noisy. The method reproduces very well both the contaminant release history and the spatial coordinates of the source; the NSE is about 98% and the distance between the true and estimated source location is less than 0.5 cm.

In summary, the proposed procedure is a novelty method able to simultaneously recover the release history and the source location of a groundwater pollutant on the basis of sparse observed concentration data. A well-designed monitoring network and the application of covariance localization and covariance inflation techniques lead to satisfactory results and reduce the inherent equifinality encountered in parameter estimation problems.

6

Effect of climate change on the groundwater levels: evaluation of local changes as a function of antecedent precipitation indices

6.1. Introduction

Groundwater represents a precious resource, especially in the critical period of the years when the surface flows are very low and of poor quality and the water demand increases. Climate change may affect groundwater sustainability due to variations in average climate conditions or seasonal distribution; this can impact on the groundwater recharge and cause severe and long droughts, leading to temporary or permanent damage. Therefore, the study of droughts characteristics and the

monitoring of their evolution is of crucial importance. Droughts can be classified into four types: meteorological, agricultural, hydrological and socioeconomic. The meteorological drought occurs in concert with precipitation deficiency and possible increase in potential evapotranspiration. The hydrological drought is associated with depletion of surface and subsurface water causing very low groundwater levels and stream flow. The agricultural drought is related to a soil moisture deficit, which affects the crop productivity. The socioeconomic drought is a consequence of the above-mentioned drought associated with anthropogenic activities; it occurs when the water resources systems are not able to meet the water demand.

The assessment of groundwater availability depends on several factors, such as groundwater storage, recharge, anthropogenic withdrawals, irrigation volumes, aquifer type and areal extents. Due to the complexity of quantifying these elements, there are difficulties in set up a complete subsurface model and it is challenging to evaluate the effects of climate change on the groundwater resource. Thus, this topic has not been sufficiently explored in the present literature and a few studies were presented. van Engelenburg et al. (2017) used a calibrated hydrological model to study the projected impact of climate change on groundwater in the Veluwe area in the Netherlands. Kahsay et al. (2018) investigated the effects of climate changes on groundwater recharge and base flow in Tekeze sub-catchment in Ethiopia usinf a spatially distributed hydrologic model (WetSpa).

In this chapter, a simple statistical approach to analyze the variation of groundwater levels as a function of meteorological indices is presented. The drought indices analyzed are the Standardized Precipitation Index (SPI; McKee et al. 1993), based on precipitation data, and the Standardized Precipitation Evapotranspiration Index (SPEI; Vicente-Serrano et al. 2010), that incorporates also temperature information.

Different studies have been proposed to study the correlation between groundwater levels and drought indices. Kahsay et al. (2018) used the SPI to track

drought and assess the impact of rainfall on water tables in some irrigation areas of the Murray-Darling Basin in Australia. Bloomfield & Marchant (2013) studied the relationship between normalized groundwater levels and SPI using observations collected at 14 sites across the UK. Kumar et al. (2016) assessed the ability of SPI to characterize the behavior of groundwater droughts using observations at more then 2000 wells distributed in different areas of Germany and Netherlands; SPI at different accumulation period have been correlated against standardized anomalies in the groundwater levels. Leelaruban et al. (2017) analyzed the relationship between different drought indices and groundwater level using data from U.S. Geological Survey Ground-Water Climate Response Network wells.

However, the presented approaches only evaluated the effect of climate variables on groundwater level in historical periods. The novelty of this work is to employ drought indices for the evaluation of the impact of climate change on groundwater levels in future periods up to 2100 using the projections of 13 EURO-CORDEX climate models (Jacob et al. 2013).

6.2. Method

6.2.1. Study area and data

The study area involves the basins of the Parma, Taro and Enza rivers, in Northern Italy (Figure 6.1). Groundwater level data available from the Emilia Romagna Regional Environmental Agency (ARPAE) were used; 41 wells were selected based on the data abundance in the monitoring years 1977-2017. The monthly groundwater levels in the spring season, which presents minimal anthropogenic disturbances due to pumping and irrigation, were chosen for the analysis. The historical precipitation and temperature data (available from ARPAE) were collected for 15 rain gauges from 1917 to 2017 and 4 temperature stations in the period 1976-2017. For

the analysis, the precipitation data in the period 1976-2017 were used.

Climate variables have been processed with the Thiessen polygon techniques to obtain average areal values.

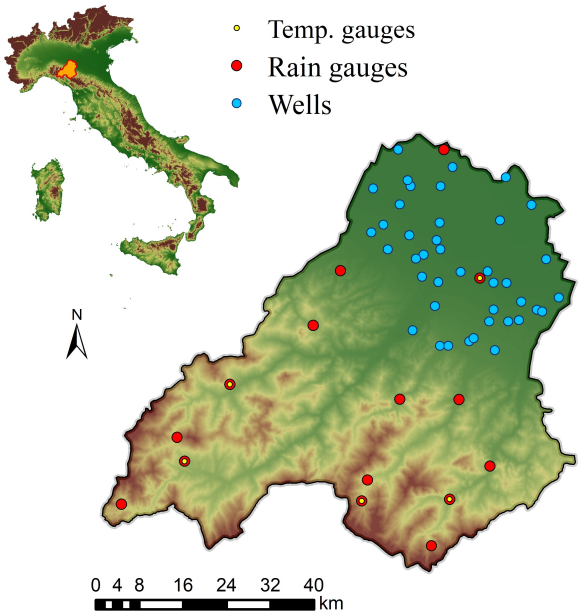

Figure 6.1. Study area, monitoring wells and temperature and rain gauging station locations. Overlapping symbols identify temperature and rain gauges located in the same position.

The future precipitation and temperature data were extracted from 13 climate models, combination of different Regional Climate Models (RCMs) and General Climate Models (GCMs) of the EURO-CORDEX ensemble (Table 6.1). The data have a grid resolution of 0.11° (grid EUR-11, 12.5 km) and are analyzed under two emission scenarios adopted by the Intergovernmental Panel on Climate Change (IPCC) in the Fifth Assessment Report (AR5; Pachauri et al. 2014): the Representative Concentration Pathways (RCPs), RCP4.5 and RCP8.5. The RCPs describe possible climate futures depending on anthropogenic greenhouse gas emission. The

RCP4.5 is the intermediate scenario; instead, the RCP8.5 is the pessimistic one without strong climate mitigation policies.

The future projections were analyzed in three time periods: 2016-2035 (short term, S.T.), 2046-2060 (medium term, M.T.), 2081-2100 (long term, L.T.); the 1986-2005 data were considered as a reference period (R.P.). The raw climate model data were bias corrected using the quantile mapping method (D'Oria et al. 2017).

Table 6.1. EURO-CORDEX ensemble (www.euro-cordex.net), combination of different RCMs and GCMs, used to extract temperature and precipitation data.

		GCM				
		CNRM-CM5	EC-EARTH	HadGEM2-ES	MPI-ESM-LR	IPSL-CM5A-MR
RCM	CCLM4-8-17	X	X	X	X	
	HIRHAM5		X			
	WRF331F				X	
	RACMO22E		X	X		
	RCA4	X	X	X	X	X

In the following, as an example, it is reported the temporal evolution of precipitation and temperature in the period 1917-2100 at a meteorological station located in the city of Parma. Figure 6.2 shows the 10-year moving average of the annual precipitation amount observed in the historical periods (1917-2017) and evaluated by the RCMs in the period 1976-2100 under the RCP4.5 and RCP8.5 scenarios. The precipitation time-series do not show significant trends for both emission scenarios; the fluctuations are probably due to the natural variability of the hydrological cycle. On the contrary, the temperature time-series (Figure 6.3) indicate a gradual warming over the century in the study area; especially under the RCP8.5 emission scenario.

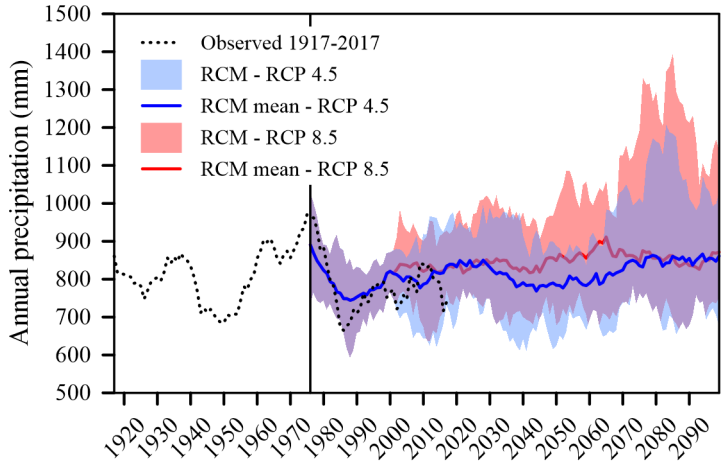

Figure 6.2. Annual precipitation for the Parma Università station (10-year moving average): observed data and projections of the 13 RCMs up to 2100 according to RCP 4.5 and 8.5 scenarios.

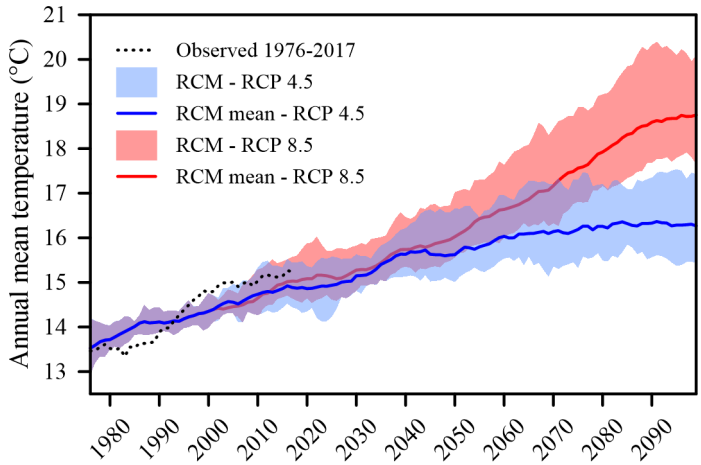

Figure 6.3. Annual mean temperature for the Parma Università station (10-year moving average): observed data and projections of the 13 RCMs up to 2100 according to RCP 4.5 and 8.5 scenarios.

6.2.2. Drought Indices

Among all the drought indices, SPI and SPEI have been selected for their simple computation, since they depend on climate variables only. The SPI (McKee et al. 1993) is computed using monthly precipitation (P) as input data. The SPEI (Vicente-Serrano et al. 2010) uses the monthly differences (D) between precipitation and potential evapotranspiration (PET):

$$D_i = P_i - PET_i, \tag{6.1}$$

where i denotes the i-th month. The potential evapotranspiration was calculated according to the Thornthwaite (1948) equation, which depends on temperature data only:

$$PET_i = 16K \left(\frac{10T_i}{I} \right)^m, \tag{6.2}$$

where T_i is the average daily temperature (°C) of the month i, I is the heat index of the average year:

$$I = \sum_{n=1}^{12} \left(\frac{\overline{T}_n}{5} \right)^{1.514}; \tag{6.3}$$

here, $\overline{T}_n$ is the mean of each monthly temperature over the investigated period (historical period or one of the three future periods) and m is a coefficient depending on I:

$$m = 6.75 \cdot 10^{-7} I^3 - 7.71 \cdot 10^{-5} I^2 + 1.79 \cdot 10^{-2} I + 0.492; \tag{6.4}$$

and K is a correction coefficient dependent on the latitude and month:

$$K = \left(\frac{N}{12} \right) \left(\frac{NDM}{30} \right). \tag{6.5}$$

Where NDM is the number of days of the month and N is the maximum number of sun hours, computed as:

$$N = \left(\frac{24}{\pi}\right) \overline{w}_s; \tag{6.6}$$

$\overline{w}_s$ is the hourly angle of sun rising, which is computed using:

$$\overline{w}_s = \arccos\left(-\tan\phi \tan\delta\right), \tag{6.7}$$

where ϕ (rad) is the latitude and δ (rad) is the solar declination, computed as:

$$\delta = 0.4093 \sin\left(\frac{2\pi J}{365} - 1.405\right); \tag{6.8}$$

here, J is the average Julian day of the month.

SPI and SPEI are normalized indices representing the probability of occurrence of P and D compared with the ones over the long climatology reference period; negative values represent a deficit, whereas positive indices indicate a surplus. The indices can be evaluated at different time scales; in this work the periods of 3, 6, 9, 12, 18, 24 and 36 months were chosen. For instance, a 3-month SPI at the end of January 2000 compares the precipitation total of November 1999, December 1999 and January 2000 with the November-December-January precipitation totals of the reference period. The scheme for the SPI and SPEI computation is summarized in Figure 6.4. First, the cumulated monthly P and D at the different time scales are computed; then, the best cumulative distribution function (cdf) that describes observed P and D is fitted. Finally, the cdf is transformed to a standard normal distribution; the normal value is the searched index.

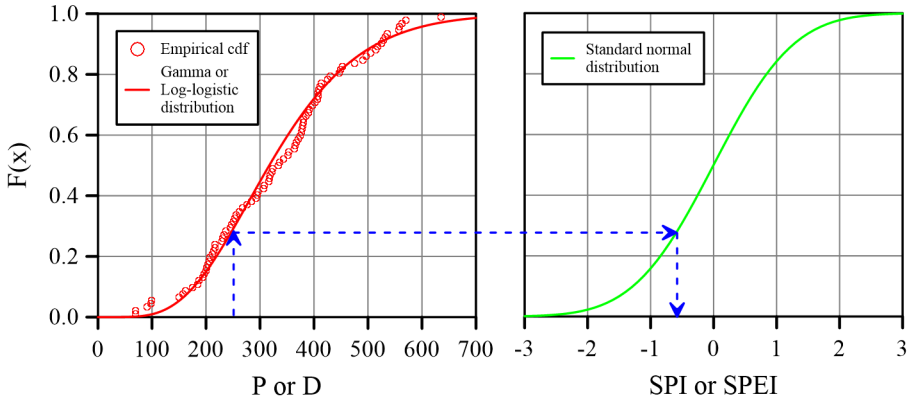

Figure 6.4. Scheme for the computation of SPI and SPEI

The SPI is computed using the well known gamma probability distribution function, which is a good fitting model for precipitation frequencies. For the computation of the SPEI, the three-parameter log-logistic distribution was considered suitable to model the D series; it is expressed as:

$$f(x) = \frac{\beta}{\alpha}\left(\frac{x-\gamma}{\alpha}\right)^{\beta-1}\left(1+\left(\frac{x-\gamma}{\alpha}\right)^{\beta}\right)^{-2}, \tag{6.9}$$

where α, β and γ are the scale, shape and origin parameters, respectively. Following this distribution, D can take values in the range $[\gamma, \infty]$ and, therefore, can assume also negative values, which are common for this type of data. The L-moment procedure (Ahmad et al. 1988) was used, in this study, for the estimation of the log-logistic parameters:

$$\beta = \frac{2w_1 - w_0}{6w_1 - w_0 - 6w_2}, \tag{6.10}$$

$$\alpha = \frac{(w_0 - 2w_1)\,\beta}{\Gamma\left(1+\frac{1}{\beta}\right)\Gamma\left(1-\frac{1}{\beta}\right)}, \tag{6.11}$$

$$\gamma = w_0 - \alpha\Gamma\left(1+\frac{1}{\beta}\right)\left(1-\frac{1}{\beta}\right), \tag{6.12}$$

where $\Gamma(\beta)$ is the gamma function of β and w_s are the probability weighted moments (PWMs) of order s. An unbiased estimator (Hosking 1986) was used for the estimation of PWMs. The unbiased PWMs are given by

$$w_s = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(\frac{N-i}{s}\right)D_i}{\left(\frac{N-1}{s}\right)} \tag{6.13}$$

The cumulative distribution function of the D series according to the log-logistic distribution is:

$$F(x) = \left[1+\left(\frac{\alpha}{x-\gamma}\right)^{\beta}\right]^{-1} \tag{6.14}$$

Then, the SPEI is given by the standardized values of $F(x)$. In this study, the approximation of Abramowitz & Stegun (1965) was used:

$$SPEI = W - \frac{C_0 + C_1 W + C_2 W^2}{1 + d_1 W + d_2 W^2 + d_3 W^3}, \tag{6.15}$$

where W depends on the probability $\mathcal{P}$ of exceeding a determined D value, $\mathcal{P} = 1 - F(x)$:

$$W = \begin{cases} -2\ln(P) & \text{for } \mathcal{P} \leq 0.5 \\ -2\ln(1-P) & \text{for } \mathcal{P} > 0.5 \end{cases} \tag{6.16}$$

The constants are: C0=2.515517, C1=0.802853, C2=0.010328, d1=1.432788, d2=0.189269, d3=0.001308.

6.2.3. Implemented procedure

After the estimation of the drought indices in the period 1976-2017, the first step of this study is to verify if a good correlation exists between the observed groundwater levels and the SPI and SPEI computed at the different time scales. The correlations

are computed for each well and time scale according to the Pearson correlation coefficient:

$$r_{AB} = \frac{\text{cov}\,(A, B)}{\sigma_A \sigma_B}, \tag{6.17}$$

where $\text{cov}\,(A, B)$ is the covariance between the variables A and B, σ_A is the standard deviation of A and σ_B is the standard deviation of B.

The wells that present data with at least one Pearson correlation coefficient greater than 0.7, among the analyzed time scales, were used for the subsequent analysis. For each of the selected wells, a linear relationship was fitted:

$$GL = b_0 + b_1(DI), \tag{6.18}$$

where GL and DI denote the groundwater level and the drought index, respectively; and b_0 and b_1 are the coefficients, which were tested for statistical significance at the 5% level.

Then, the resulting regression coefficients were used to compute the future groundwater levels according to the drought indices computed considering the precipitation and temperature data of the climate models in the three future periods. The future analyses were carried out for each well, using only the SPI and SPEI at the time scale with the higher correlation coefficient in the historical period.

6.3. Results and discussion

6.3.1. Estimated SPI and SPEI in the historical periods

Figures 6.5 and 6.6 show the areal SPI and SPEI computed in the same period 1976-2017 at the time scales of 3, 6, 9, 12, 18, 24 and 36 months, respectively. The two indices behave in agreement detecting the same dry and wet periods; however,

the negative values of the SPEI index are lower than the SPI ones, especially for the last decades. For instance, in the period 2002-2010, an extremely dry period for the study area, it can be noticed that the drought duration is the same for SPI and SPEI, but the SPEI values denote a more severe drought. This is due to the gradual warming of the considered area and only the SPEI drought index takes into account the temperature data.

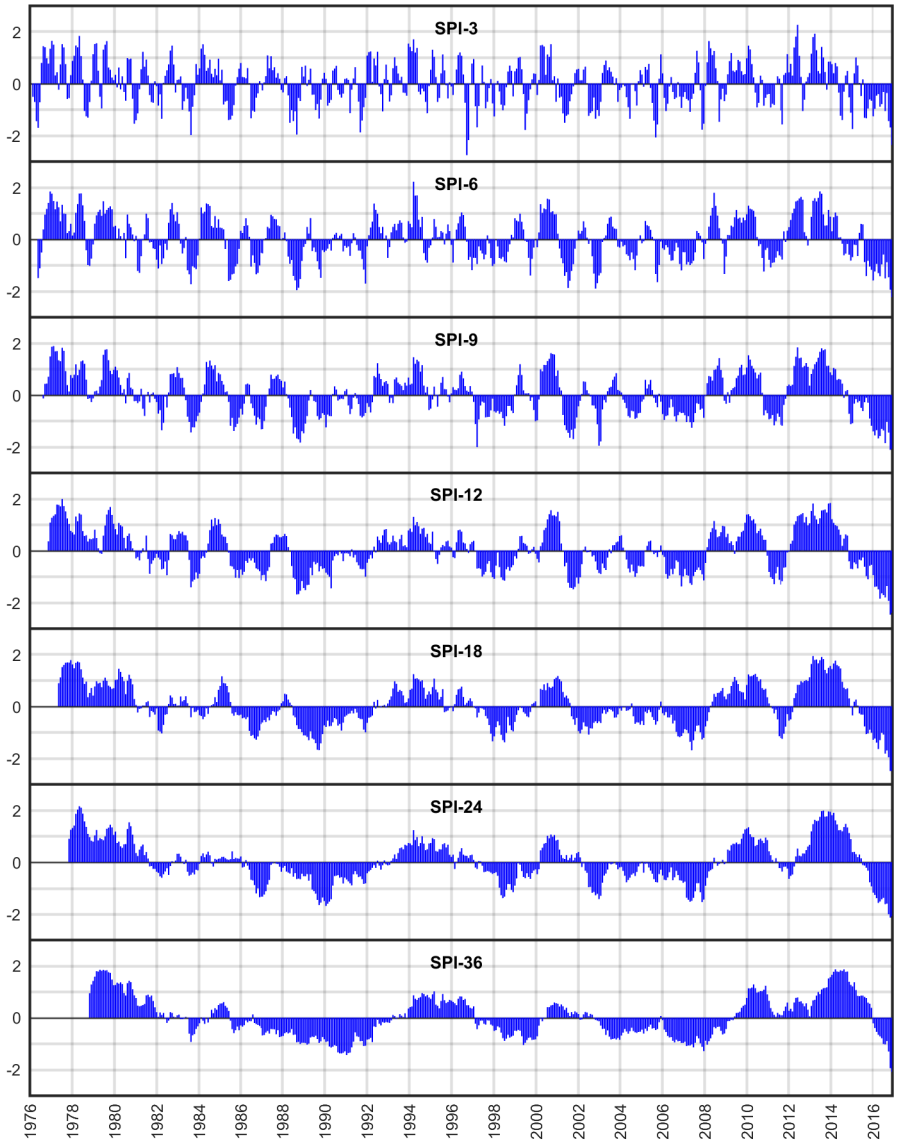

Figure 6.5. The areal SPI computed for the period 1976-2010 at the time scale of 3, 6, 9, 12, 18 and 36 months

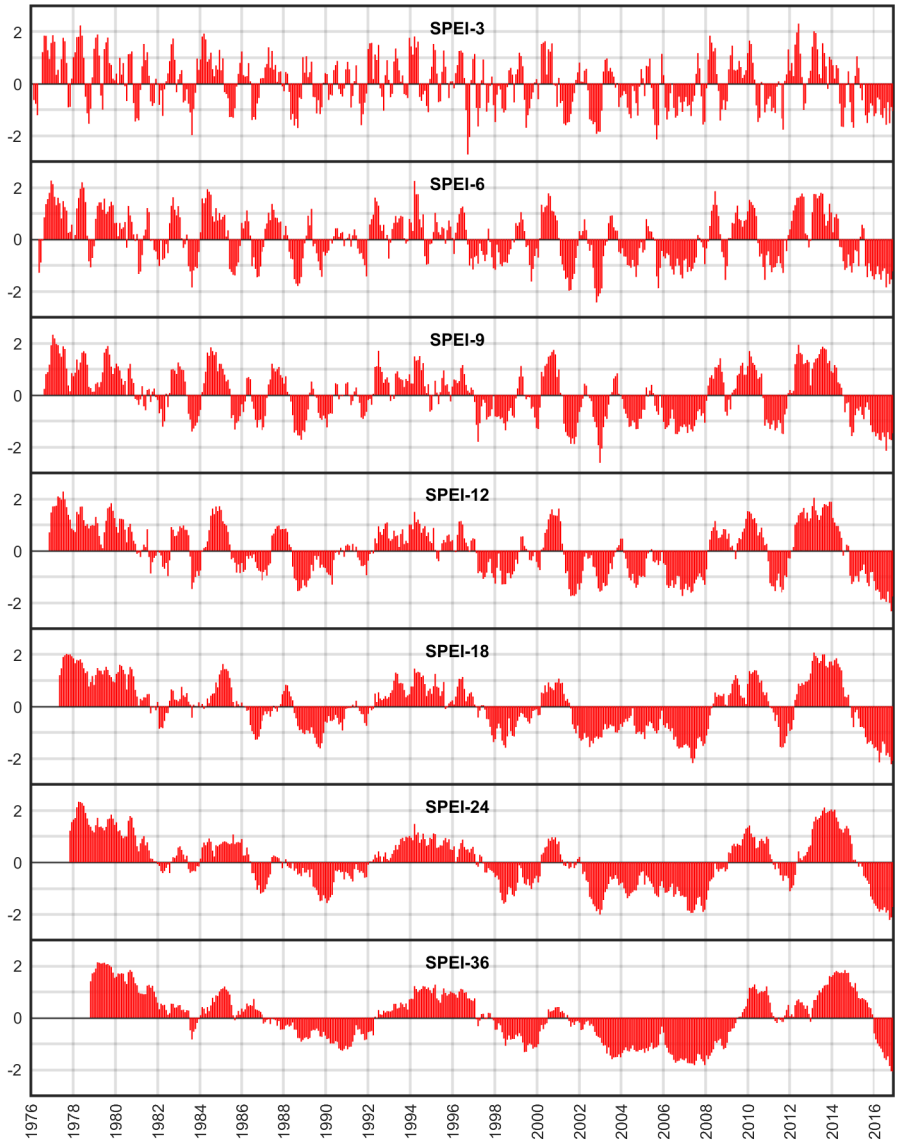

Figure 6.6. The areal SPEI computed for the period 1976-2010 at the time scale of 3, 6, 9, 12, 18 and 36 months

6.3.2. Correlation between groundwater levels and drought indices

The correlation between the observed groundwater levels at each well and the SPI and SPEI indices computed at the different time scales are depicted in Figure 6.7 by means of a color scale. The Pearson correlation coefficients are similar for the two indices, but different for each well and time scale, indicating a different sensitivity of the well data to the climate variables. This stems from the fact that the wells are located in different aquifer types; the distinct characteristics of the wells and the complexity of the aquifer in this area lead to a different response of groundwater levels to climate variability. For instance, the well PR99-00, which is the deepest well sampled in the study area (depth from land surface is 175 m), shows the highest correlation with the SPI and SPEI at the time scale of 36 months; which means that the groundwater level responds to the climate variables with a considerable time lag. On the contrary, the shallow wells, such as PRA1-00 (depth from land surface is 30 m), present high correlation with the indices at medium time scale. For the majority of the wells, the higher coefficients are observed at the time scales of 9, 12 and 18 months.

The groundwater level series that present at least one Pearson correlation coefficient with the drought indices, computed at the different time scales, greater than 0.7 were used for the subsequent analysis. The selected wells are 24 for the analysis performed with the SPI index, indicated with crosses in Fig. 6.7, and 28 for the ones performed with the SPEI, indicated with dots in Fig. 6.7.

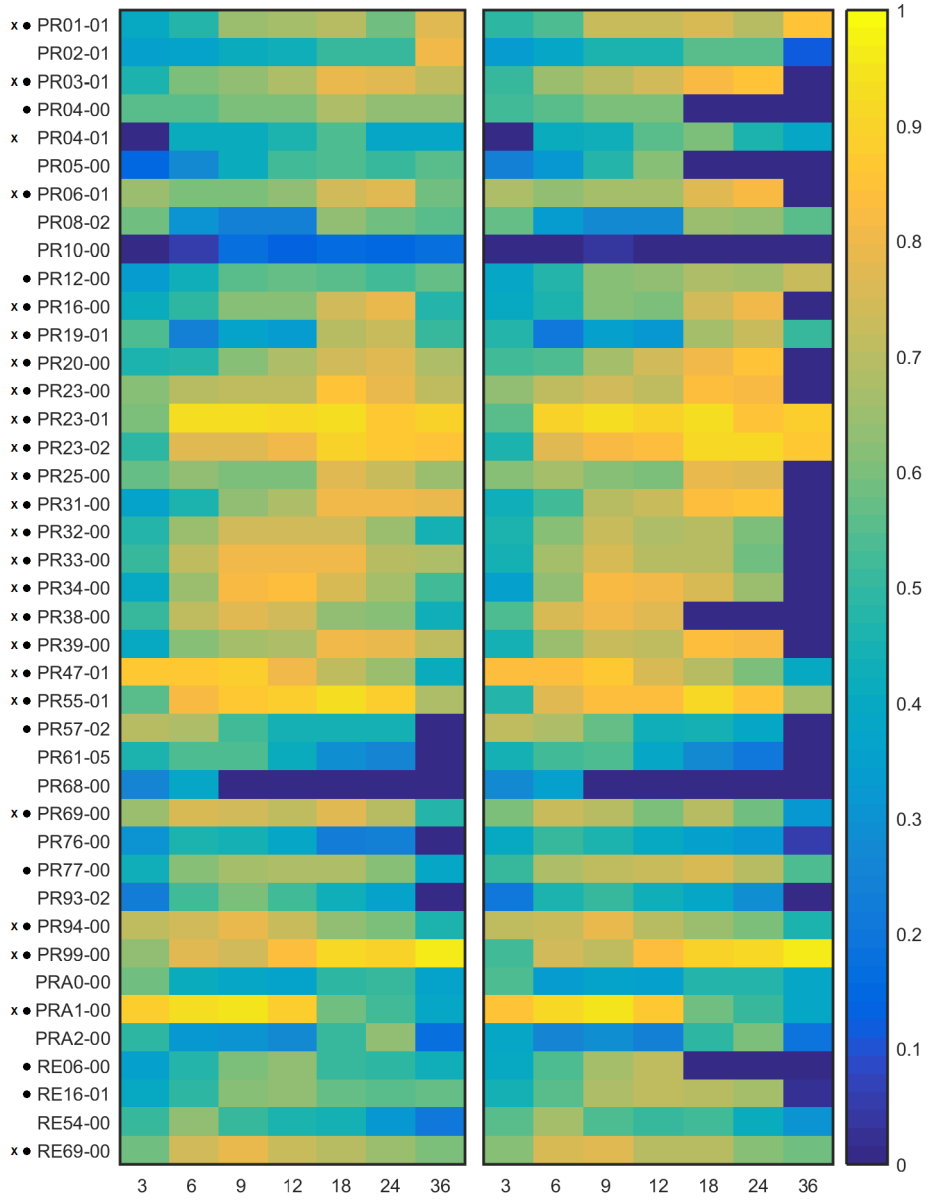

Figure 6.7. The Pearson correlation coefficients between groundwater levels observed at the 41 wells (y-axis) and the SPI(left) and SPEI (right) indices at the time scale of 3, 6, 9, 12, 18, 24 and 36 months (x-axes). The cross and the dot next to the well name denote that the well presents at least one Pearson correlation coefficient greater than 0.7 with SPI and SPEI, respectively.

6.3.3. The relationship between groundwater levels and drought indices

The relationship between groundwater level and drought indices was investigated for each of the selected wells in the previous step and at the time scales of 3, 6, 9, 12, 18 and 36 months. For the sake of brevity, only the results for one well, named PR55-01, are shown. Figure 6.8 shows the linear regression model on the basis of SPI indices; the results of the regression analysis are reported in Table 6.2. The t-tests were performed to assess the statistical significance at the 5% level of the estimated coefficients. The observed significant values (p-values) are less then 0.05 for all the estimated coefficients, denoting that the relationship between the groundwater level and SPI indices is significant. The correlation is very high for all the time scales, the Pearson correlation coefficients are greater than 0.8. The best correlation occurs at the time scale of 18 months (r=0.93), resulting in a linear relationship given by the equation:

$$GL = 45.57 + 1.14SPI18, \tag{6.19}$$

where the groundwater levels and the regression coefficients are expressed in meters above sea level.

In Figure 6.9 and Table 6.3 the results of the linear regression analysis performed with the SPEI indices are reported. The estimated coefficients are similar to those obtained with the SPI and they are all statistically significant. Also in this case, the best correlation is observed at the time scale of 18 months (R=0.91), the equation of the linear relationship is:

$$GL = 45.74 + 0.88SPEI18 \tag{6.20}$$

Therefore, with reference to well PR55-01, the analysis to compute the future groundwater levels were computed according to the projected drought indices at the time window of 18 months.

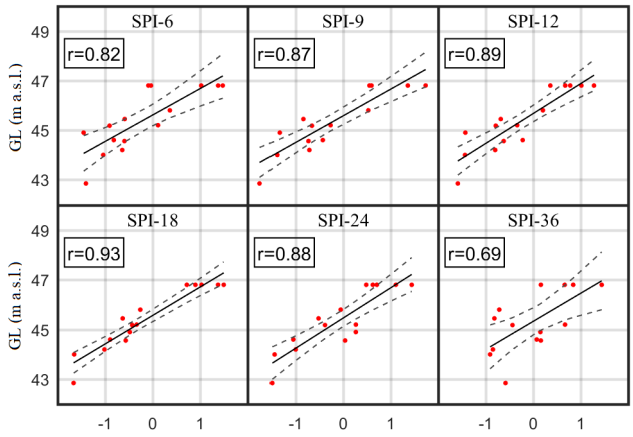

Figure 6.8. Linear regression model for well PR55-01 with SPI indices. The x-axis shows the SPI values and the y-axis the groundwater level in m a.s.l. The points represent the observed groundwater levels, the solid line is the regression line and the dashed lines are the confidence intervals (95%); the correlation coefficients are reported in the boxes.

Table 6.2. Results of the regression analysis for the well PR55-01 and the SPI index at time scales 6, 9, 12, 18, 24 and 36 months. The estimated coefficients of the regression models, standard error (SE) of coefficients, t-test statistic values, and p-values are reported.

		Coef. [m a.s.l.]	SE	t-value	p-value
SPI-6	b_0	45.61	0.20	226.46	$9.16 \cdot 10^{-25}$
	b_1	1.07	0.22	4.94	$2.72 \cdot 10^{-4}$
SPI-9	b_0	45.58	0.15	299.62	$2.41 \cdot 10^{-26}$
	b_1	1.08	0.15	7.15	$7.44 \cdot 10^{-6}$
SPI-12	b_0	45.67	0.15	302.12	$2.16 \cdot 10^{-26}$
	b_1	1.21	0.16	7.37	$5.39 \cdot 10^{-6}$
SPI-18	b_0	45.57	0.11	426.52	$2.44 \cdot 10^{-28}$
	b_1	1.14	0.11	10.78	$7.15 \cdot 10^{-8}$
SPI-24	b_0	45.46	0.16	288.74	$3.89 \cdot 10^{-26}$
	b_1	1.21	0.18	6.73	$1.41 \cdot 10^{-5}$
SPI-36	b_0	45.33	0.25	184.40	$1.32 \cdot 10^{-23}$
	b_1	1.13	0.34	3.30	$5.79 \cdot 10^{-3}$

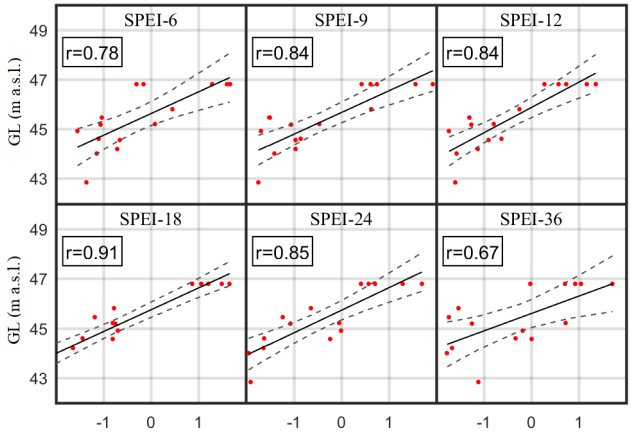

Figure 6.9. Linear regression model for the well PR55-01 with SPEI indices. The x-axis shows the SPEI values and the y-axis the groundwater level in m a.s.l. The points are the observed groundwater levels, the solid line is the regression line and the dashed lines are the confidence intervals (95%); the correlation coefficients are reported in the boxes.

Table 6.3. Results of the regression analysis for the well PR55-01 and the SPEI index at time scales 6, 9, 12, 18, 24 and 36 months. The estimated coefficients of the regression models, standard error (SE) of coefficients, t-test statistic values, and p-values are reported.

		Coef. [m a.s.l.]	SE	t-value	p-value
SPEI-6	b_0	45.61	0.22	203.67	$3.63 \cdot 10^{-24}$
	b_1	0.88	0.21	4.18	$1.07 \cdot 10^{-3}$
SPEI-9	b_0	45.66	0.18	250.39	$2.48 \cdot 10^{-25}$
	b_1	0.88	0.15	5.78	$6.39 \cdot 10^{-5}$
SPEI-12	b_0	45.86	0.19	246.35	$3.07 \cdot 10^{-25}$
	b_1	1.02	0.17	6.11	$3.71 \cdot 10^{-5}$
SPEI-18	b_0	45.74	0.13	365.10	$1.84 \cdot 10^{-27}$
	b_1	0.88	0.09	9.38	$3.77 \cdot 10^{-7}$
SPEI-24	b_0	45.72	0.18	252.24	$2.26 \cdot 10^{-25}$
	b_1	0.91	0.15	5.96	$4.73 \cdot 10^{-5}$
SPEI-36	b_0	45.59	0.26	174.88	$2.63 \cdot 10^{-23}$
	b_1	0.70	0.23	3.11	$8.34 \cdot 10^{-3}$

6.3.4. Estimated SPI and SPEI in the future periods

The future drought indices are computed using the precipitation and temperature data extracted from the 13 RCMs under the two scenarios RCP4.5 and 8.5. Figure 6.10 depicts the frequency distributions of the SPI at the time scale of 18-months projected in the three future periods in the study area. The points represent the mean frequency in the reference period and the box-whiskers plot describe the variability between the 13 RCMs, the blue and red box-plots show the results under the RCP4.5 and RCP 8.5 emission scenario, respectively. The SPI indices do not show significantly changes in the three future period, the class frequency in the reference period is always contained in the RCMs variability.

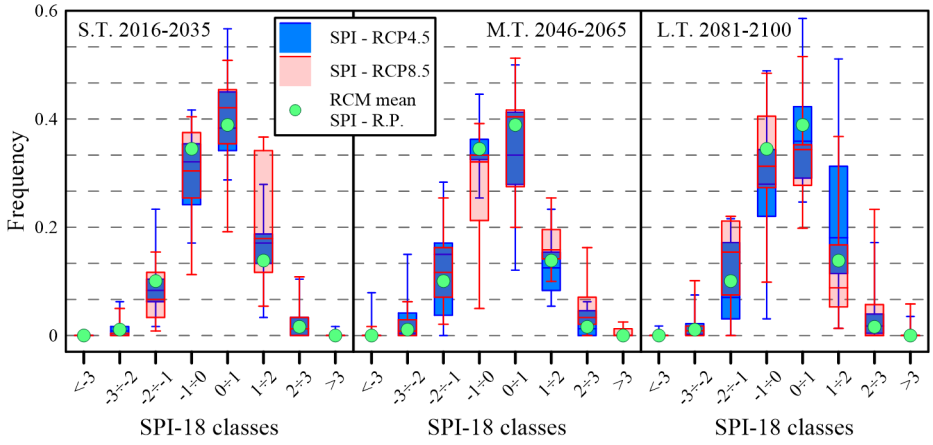

Figure 6.10. Frequency distributions of the SPI at the time scale of 18-months projected in the three future periods. The points represent the mean frequency in the reference period and the box-whiskers plot describe the variability between the 13 RCMs, the blue and red box-plots show the results under the RCP4.5 and RCP 8.5 emission scenario, respectively.

Figure 6.11 shows the frequency distributions of the projected SPEI at the same time scale. The frequencies of the SPEI lower than 1, which denote extremely drought, are expected to increase in the three future periods, especially at medium

and large term and under the RCP8.5 scenario. For example, at the long term (2081-2100), the mean frequency of the SPEI-18 in the range [-2,-1] is expected to increase, with respect to the reference period, of 2.18% under the RCP4.5 emission scenario and 16.81% under the RCP8.5.

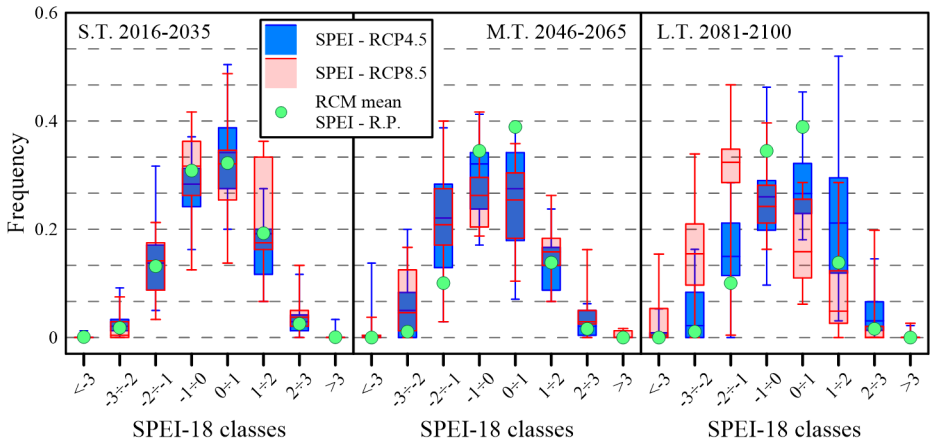

Figure 6.11. Frequency distributions of the SPEI at the time scale of 18-months projected in the three future periods. The points represent the mean frequency in the reference period and the box-whiskers plot describe the variability between the 13 RCMs, the blue and red box-plots show the results under the RCP4.5 and RCP 8.5 emission scenario, respectively.

6.3.5. Future groundwater levels

For each well, the future analyses were carried out using the drought indices at the time scale that presents the higher correlation coefficient. In the following, the results for the well PR55-01 are presented, considering May as the reference month. Figure 6.12 shows the empirical cumulative distribution function of the groundwater level in May as a function of the SPI-18 under the RCP4.5 and RCP8.5 scenarios. All the results of the 13 models have been considered as a single realization with equal reliability. The blue line represents the groundwater level cumulative distribution frequency in the reference period, the red, green and pink lines are the cumulative distribution frequencies predicted at the short, medium

and long term. The analysis with the SPI does not detect significant changes in the three future periods and for both the emission scenarios.

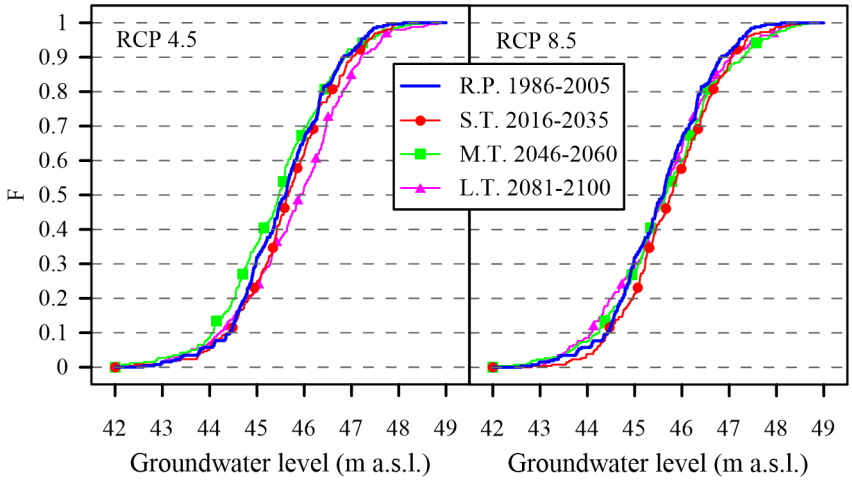

Figure 6.12. Cumulative distribution function of groundwater level in May projected in the three future periods according to the analysis performed with the SPI-18 under the RCP4.5 (left) and RCP8.5 (right) emission scenarios.

The analysis conducted with the SPEI predicts a decrease in groundwater levels, especially at medium and long term and for the RCP8.5 scenario (Figure 6.13). The frequency of the lower groundwater levels is expected to increase. For example, the frequency of the groundwater level corresponding to the 10th percentile of the groundwater level in the reference period (44.8 m a.s.l.), increases of 4% at short term, 15% at medium term and 10% at long term, under the RCP4.5 emission scenario; the increase is of 3% at short term, 14% at medium term and 26% at long term, under the RCP8.5 scenarios.

The results presented for the specific well PR55-01 are reproduced by almost half of all those analyzed; some wells, instead, show no significant alteration in future periods. The different groundwater levels responses depend on the various characteristic of the wells and the type of aquifer in which they are located. Future

development of the work will involve the analysis on areas with more data availability with the aim of being able to compute normalized indices of groundwater levels, similarly to the SPI and SPEI, which allow to compare the results across different locations.

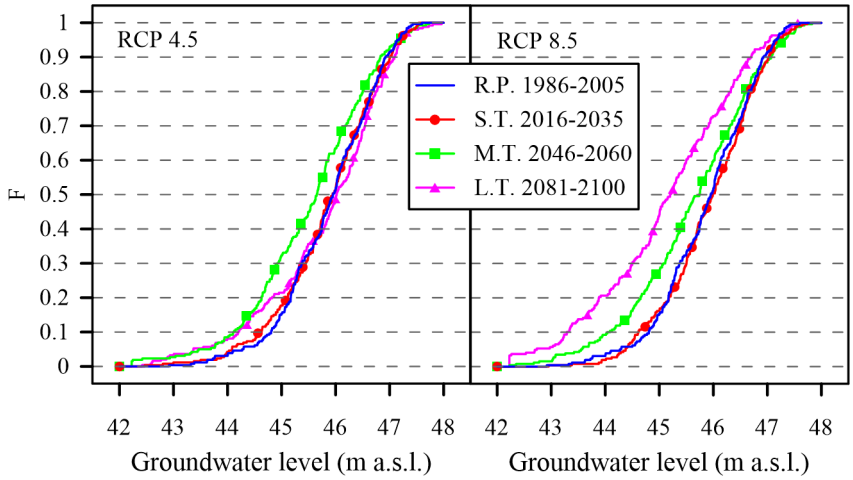

Figure 6.13. Cumulative distribution function of groundwater level in May projected in the three future periods according to the analysis performed with the SPEI-18 under the RCP4.5 (left) and RCP8.5 (right) emission scenarios.

6.4. Concluding remarks

In this chapter, a simple method to predict the impact of climate change on groundwater levels was presented. Two multiscale drought indices, the Standard Precipitation Index (SPI) and the Standard Precipitation Evapotranspiration Index (SPEI), were used to relate meteorological variables and groundwater levels and to evaluate the future levels projections using the climate data extracted from 13 Regional Climate Models (RCMs).

In the historical period, the SPI and SPEI behavior is similar and both indices

show high correlations with observed levels, denoting the ability of the selected indices to assess the groundwater level fluctuations.

The future climate projections, instead, are different for the two indices. The analysis performed with the SPI predicts a non-critical alteration of the groundwater levels. On the contrary, the projected SPEI denotes an increase of the frequency of the low groundwater levels, which means that the groundwater droughts are expected to increase in the future periods leading to quantity and quality water problems.

The difference in the results for the analysis carried out with the SPI and the SPEI is due to the climate projections. In this study area, in fact, the RCMs indicate a very small decrease of the precipitation and a remarkable temperature increase. Therefore, an index that only consider precipitation cannot be able to detect the effect of climate change on the groundwater levels; it is suggested to prefer the use of the index SPEI, since it allow to jointly assess the effect of the precipitation-temperature variability.

The proposed method is an advantageous simple statistical method that allows to fast evaluate the variation in groundwater levels based on precipitation and temperature data only. The assessment of the impact of climate change on the groundwater resource is challenging since it needs the knowledge of several factor that are difficult to quantify or that require very expensive procedures for their evaluation. The groundwater level, even if it does not fully characterize the aquifer, is a good indicator and it provides precious information about the aquifer conditions.

Conclusions

The objective of this thesis is to investigate advanced techniques dealing with inverse problems and climate change analysis in the context of surface and subsurface hydrology.

The solution of inverse problems focuses on the ensemble Kalman filter approaches, which are selected for their computational efficiency, the flexibility to be coupled with almost any forward model, the capability to jointly estimate multiple parameters and the possibility to quantify the uncertainty of both parameters and state variables. Among the available variants, the Ensemble Smoother with Multiple Data Assimilation (ES-MDA) was extensively analyzed and improved for different applications. Part of the work aimed at developing a Python software package for the solution of inverse problems based on the proposed methodology. The software package is easy to use and it has a flexible workflow so that it can be applied for different case studies and adapted to other ensemble Kalman methods. The codes include various functionalities that allow to implement different configurations of the algorithm suiting different type of problem. In particular, the package presents useful tools for the solution of inverse problems aimed at identifying time series function, which is a novelty aspect for the ES-MDA method.

The ensemble Kalman filter techniques have been used in the present literature to estimate time-independent parameters; in this work, these methods have been adapted for the estimation of discretized time functions allowing to extend the

applications of Ensemble Kalman-based method to many types of problems.

The ES-MDA was applied as a new approach for the solution of different inverse problems in the hydrology and hydrogeology fields. The first application aimed to solve the reverse flow routing problem; the objective is the estimation of an inflow hydrograph, which is a function of time, to a hydraulic system on the basis of observations collected downstream that can be water level information or discharge hydrographs and a known forward routing model that relates parameters and observations. Two synthetic examples were presented to show the capabilities of the methodology also looking at different ES-MDA settings such as ensemble sizes and the use of covariance localization and inflation techniques. A new procedure to perform covariance localization considering temporal lapses rather then spatial distances was introduced. The procedure was then applied for the solution of a real case study. The results show the capability of ES-MDA to solve this type of problem even for complex river systems with a small computational cost. The method reach better results using large ensemble size, however, the application of covariance and inflation techniques has led to significant improvements in the solution and computational burden. The ES-MDA competes in accuracy with other optimization methods presented in the literature, but outperform them in terms of computational efficiency.

The second application dealt with the calibration of a hydraulic numerical model that simulates rainfall-runoff processes. The inverse procedure is applied for the estimation of roughness and infiltration input maps to the Parflood Rain numerical model, which represents the forward model, on the basis of an observed discharge hydrograph at the basin outlet and assuming the other characteristic of the system known. Also in this case, the proposed method was initially tested by means of two synthetic examples, which have demonstrated that ES-MDA is able to accurately reproduced both the investigated parameters and calibration target. Then, the methodology was applied for the calibration of Parflood Rain

related to a real flood event leading to satisfactory results. The capability of ES-MDA to be amenable by parallel computing allowed to perform the calibration of a complex hydraulic model with acceptable computational time; this can lead to a more accurate set up of numerical models, which usually used roughness and infiltration coefficients determined a-priori on the basis of system characteristic that may not suit the specific problem.

The application related to the groundwater field concerns the simultaneous identification of the source location and the release history of a contaminant spill in an aquifer, based on the knowledge of concentration data collected at a few points and a given forward model. First, an analytic case study was used to investigate different alternatives in the ES-MDA settings: observation sampling scheme, ensemble size and the application of covariance localization and covariance inflation. Here, a new spatiotemporal iterative localization was introduced, which allows to take into account both spatial and temporal distances and to update them during the iterative process. The results validate the ES-MDA method for the estimation of a time series function, which is represented by the discretized release history of the pollutant, and for the simultaneously identification of different type of parameters. A well-designed monitoring network and the application of covariance localization improves the performance of the proposed inverse procedure and help to minimize ill-posedness and equifinality problems. Finally, an experimental case that uses data collected in a laboratory sandbox that reproduces an unconfined aquifer validates the capability of the proposed inverse procedure to simultaneously reconstruct the source location and release history of a groundwater pollutant also in real case. This is the first time that a stochastic method is applied to solve this type of problem allowing to assess the estimation uncertainty and directly identify the location of the contaminant source jointly with its release history.

The last part of the thesis deals with the evaluation of the impact of climate

change on the groundwater availability. A simple statistical approach has been proposed to project the future groundwater levels up to 2100 in a study area involving the basins of the Parma, Taro ed Enza rivers, in northern Italy. The first step of the study focused on the analysis of the correlations between groundwater levels collected at several wells over the study area in an historical period and two drought indices that depend on precipitation and temperature data only, the Standard Precipitation Index (SPI) and the Standard Precipitation Evapotranspiration Index (SPEI). The high correlations detected in the historical periods were used to define a linear relationship to be applied in the future projections of groundwater levels on the basis of the SPI and SPEI computed using climate data extracted from Regional Climate Model. The results indicate a progressive increase in the frequency of the low groundwater levels in most of the investigated wells, when both precipitation and temperature data are involved in the analysis. Therefore, groundwater droughts are expected to increase over the century in this area, as a consequence of climate changes, resulting in a deterioration of the quantity and quality of the available fresh water. The proposed approach is a surrogate model that allow to assess the impact of climate variability on groundwater systems in a simple and fast way. It represents a valid alternative for the solution of this challenging problem and may help to fill the gap in the present literature, which provides very few works about this topic due to the complexity to set up a complete model describing the subsurface processes.

Suggestions for future research

This section of the thesis provides an overview of future potential improvements of the presented works and new lines of research.

The Python software package can be integrated with additional useful function-

alities. For instance, the covariance inflation methods have not been extensively analyzed in this thesis; tools for the application of different covariance inflation approaches can be implemented. Also, parallelization tools, that convert the sequential procedure in a parallel one, can be developed.

Future works will focus on new applications of the Ensemble Kalman filter methods. Possible case studies can involve the solution of inverse problems in urban water networks, such as the detection of the infiltration and inflow (I/I) of unwanted water in sewers, the identification of the pollutant source in water distribution networks or the location of aqueduct leaks, on the basis of known measurements of pollutant concentration or water flowrate. Another potential application could be the identification of the source location of an air pollutant emissions based on quality data collected at some monitoring stations. The application of ES-MDA for the calibration of hydrological-hydraulic models will be tested for the direct estimation of the investigated parameters moving from the one factor method. Furthermore, the possibility to implement this procedure for a self-calibration of the numerical model in real-time will be considered.

Future works related to the investigation of the impact of climate change on groundwater levels will focus on applying the presented methodology to a different study area with continuous groundwater level monitoring. The abundance of data permits to define the standard groundwater indexes (SGI) in order to better characterize the groundwater droughts and their relationships with other climatic indexes.

In the context of climate change, the Ensemble Kalman filter methods will be investigated for the use in the calibration of stochastic rainfall models to be developed at a basin scale for the appropriate investigation of the impacts on the frequency and severity of the flood events. Nowadays, the climate models provide, mainly, daily projections, not useful for the investigations on flood events in small and medium size Italian basins. It is then necessary to develop synthetic rainfall

data at small time scale that must satisfy daily statistic constrains and spatial correlations on the different basin zones. It is known that such stochastic models are very hard to calibrate and the application of an ensemble type method can be decisive.

Appendix

In this Appendix, the Python codes written for a specific application are reported. The InputSettings.py and Mod.py modules developed for the solution of the inverse problem introduced in Chapter 4, which aims to simultaneously identify the source location and the release history of a pollutant in groundwater, are presented. These are the only two modules of the software package that are specific for the analyzed study case and require to be edited by the user.

The Flopy Python package is used to run and post-process MODFLOW and MT3DMS models, which simulate the groundwater flow and the contaminant transport process, respectively.

The different features of the two modules are described in Chapter 2.

InputSettings.py

```
import numpy as np

def Func_ens(par, ens):
    #function for the generation of the initial ensemble for the
    #simultaneous identification of the release history and source
    #location of a pollutant in groundwater
    # X[1,2]---> coordinates of the source location
    from Tools import EnsembleGenerator
    time_all=par[:,2]
```

```
    time=time_all[2:]
    N_par=time_all.size
    Ensemble=np.zeros((N_par,ens))

    (Xmin,Xmax)=(5,30)
    (Ymin,Ymax)=(30,34)
    Ensemble[0,:]=EnsembleGenerator.Random(Xmin,Xmax,1,ens)
    Ensemble[1,:]=EnsembleGenerator.Random(Ymin,Ymax,1,ens)

    (aMin,aMax)=(1e-8,0.05)
    (muMin,muMax)=(np.quantile(time,0.2),np.quantile(time,0.55))
    (sigmaMin,sigmaMax)=(np.quantile(time,0.03),np.quantile(time,0.15))
    (bMin,bMax)=(800,1000)
    Ensemble[2:N_par,:]=EnsembleGenerator.PdfNormal(aMin,aMax,
                                                    muMin,muMax,
                                                    sigmaMin,sigmaMax,
                                                    bMin,bMax,
                                                    time,
                                                    N_par-2,ens)
    return Ensemble

def Func_err(N_obs,ens):
    from Tools import ErrorGenerator
    var_y=1e-2
    eps,R=ErrorGenerator.NormalError(var_y, N_obs, ens)
    return (eps,R)

def forward_transf(xx):
    from Tools import Transformation as T
    # (Xmin1,Xmax1)=(6,30)
    # (Xmin2,Xmax2)=(30,35)
    # xx[0,:]=T.LogLim_forward(xx[0,:],Xmin1,Xmax1)
    # xx[1,:]=T.LogLim_forward(xx[1,:],Xmin2,Xmax2)
    # xx[2:-1,:]=T.Log_forward(xx[2:-1,:])
    xx=T.Log_forward(xx)
    return xx

def backward_transf(xx):
    from Tools import Transformation as T
    # (Xmin1,Xmax1)=(6,30)
    # (Xmin2,Xmax2)=(30,34)
```

```
    # xx[0,:]=T.LogLim_backward(xx[0,:],Xmin1,Xmax1)
    # xx[1,:]=T.LogLim_backward(xx[1,:],Xmin2,Xmax2)
    # xx[2:-1,:]=T.Log_backward(xx[2:-1,:])
    xx=T.Log_backward(xx)
    return xx

def localization(ens_par,par,obs,iter_loc):
    time_par=par[:,2]
    pos_obs=obs[:,0:2]
    time_obs=obs[:,2]
    if iter_loc=='n':
        pos_par=par[:,0:2]
    else:
        ens_m=ens_par.mean(1)
        pos_par=np.tile(ens_m[0:2],(time_par.shape[0],1))
    from Tools import Localization as Loc
    a_space=150
    a_time=2500
    [rho_yy_sp,rho_xy_sp,rho_xx_sp]=Loc.SpaceLocal(a_space,
                                                   pos_par,
                                                   pos_obs)
    [rho_yy_tm,rho_xy_tm,rho_xx_tm]=Loc.TimeLocal(a_time,
                                                  time_par,
                                                  time_obs)
    rho_yy=rho_yy_sp*rho_yy_tm
    rho_xy=rho_xy_sp*rho_xy_tm
    rho_xx=rho_xx_sp*rho_xx_tm
    rho_yy[np.isnan(rho_yy)]=1
    rho_xy[np.isnan(rho_xy)]=1
    rho_xx[np.isnan(rho_xx)]=1
    return (rho_yy,rho_xy,rho_xx)

def Metrics_obs(Xprev,pred,True_par,obs):
    from Tools import Metrics as m
    par_rel=Xprev[2:,:]
    RMSE_obs=m.RMSE(obs, pred.mean(1))
    AES=m.AES(par_rel)
    metrics_dict ={}
    for variable in ['RMSE_obs','AES']:
        metrics_dict[variable]=eval(variable)
    return metrics_dict
```

```
    metrics_dict['rmse_obs']=eval('rmse_obs')
    metrics_dict['rmse_par']=eval('rmse_par')
    return metrics_dict

def Metrics_obs_par(Xprev,pred,True_par,obs):
    from Tools import Metrics as m
    par_pos=Xprev[0:2,:]
    true_pos=True_par[0:2]
    par_rel=Xprev[2:,:]
    true_rel=True_par[2:]
    RMSE_obs=[m.RMSE(obs.flatten(), pred.mean(1))]
    RMSE_par=[m.RMSE(true_rel, par_rel.mean(1))]
    dist_par=[m.spatial_distance(true_pos,par_pos.mean(1))]
    NSE_par=[m.NSE(true_rel, par_rel.mean(1))]
    AES=[m.AES(par_rel)]
    metrics_dict ={}
    for variable in ['RMSE_obs', 'RMSE_par', 'dist_par','NSE_par','AES']:
        metrics_dict[variable]=eval(variable)
    return metrics_dict
```

Mod.py

```
import numpy as np
import math, os
import flopy
import flopy.modflow as mf
import flopy.mt3d as mt
import flopy.utils as fu

os.chdir('Model')
True_par_file=np.loadtxt('True_input.txt', dtype=float)
True_par_file_model=np.loadtxt('True_input_model.txt', dtype=float)
# par=np.arange(0.,252.,1)
# par[0]=14.25
# par[1]=32.75

modelname='FlowModel'
namemt3d='TransMod'
workspace='ModelFiles'

mfMod=mf.Modflow.load(f'{modelname}.nam', model_ws=workspace,
                      exe_name='mf2005dbl.exe')
mtMod=mt.Mt3dms.load(f'{namemt3d}.nam', model_ws=workspace,
                     modflowmodel=mfMod,
                     exe_name='mt3d-usgs_1.1.0_64.exe')

dis=mfMod.get_package('dis')
wel=mfMod.get_package('wel')
# AA=mfMod.wel.stress_period_data.to_array(kper=0)
delr=dis.delr.array[0]
delc=dis.delc.array[0]
top=dis.top.array[0,0]
botm=dis.botm.array[-1,-1,-1]
nlay=dis.nlay
delv=(top-botm)/nlay

t=True_par_file[2:,2]
t_model=True_par_file_model[:,2]
if t[0]>t_model[0]:
    t=np.concatenate(([t_model[0]],t))
    mod_start=True
```

```
else:
    mod_start=False
if t[-1]<t_model[-1]:
    t=np.concatenate((t,[t_model[-1]]))
    mod_end=True
else:
    mod_end=False

os.chdir('..')

def write_input(par):
    x_pos=par[0]
    y_pos=5
    z_pos=par[1]
    source_col=math.ceil(x_pos/delr)
    source_row=math.ceil(y_pos/delc)
    source_lay=nlay-math.floor(z_pos/delv)
    rel=par[2:]
    if mod_start:
        rel=np.concatenate(([rel[0]],rel))
    if mod_end:
        rel=np.concatenate((rel,[rel[-1]]))
    rel_model=np.interp(t_model,t,rel)
    sp_data={}
    sp_data[0]=[source_lay-1,source_row-1,source_col-1,0]
    for i in range(1,t_model.shape[0]+1):
        sp_data[i]=[source_lay-1,source_row-1,source_col-1,rel_model[i-1]]
    mfMod.remove_package("wel")
    wel_par=flopy.modflow.ModflowWel(mfMod,stress_period_data=sp_data)
    wel_par.write_file()  #write file

    #write mt3dms input (SSM file)
    # ssm=mtMod.get_package('ssm')
    itype = 2
    C=20
    ssm_data = {}
    ssm_data[0] = [source_lay-1,source_row-1,source_col-1, C, itype]
    mtMod.remove_package("ssm")
    ssm_par = flopy.mt3d.Mt3dSsm(mtMod, stress_period_data=ssm_data)
    ssm_par.write_file()
```

```
def run():
    try:
        os.remove(os.path.join(workspace, 'MT3D001.UCN'))
    except:
        pass
    mfMod.run_model()
    mtMod.run_model()

def read_output():
    Obs_file=np.loadtxt('True_obs.txt', dtype=float)
    N_obs=Obs_file.shape[0]
    x_obs=Obs_file[:,0]
    z_obs=Obs_file[:,1]
    # y_obs=5.0
    time_obs=Obs_file[:,2]
    obs_col=np.zeros((N_obs),int)
    # obs_row=math.ceil(y_obs/delc)
    obs_lay=np.zeros((N_obs),int)
    for ll in range(0,N_obs):
        obs_col[ll]=math.ceil(x_obs[ll]/delr)
        obs_lay[ll]=nlay-math.floor(z_obs[ll]/delv)
    concobj=fu.UcnFile(os.path.join(workspace,'MT3D001.UCN'))
    times_mod = np.array(concobj.get_times())
    # conc = concobj.get_data(totim=times[150])
    conc = concobj.get_alldata()
    concobj.close()
    # t=conc.shape[0]
    C=np.zeros((N_obs))
    tt=[]
    for i in range(0,N_obs):
        tt+=np.where(times_mod==time_obs[i])
        C[i]=conc[tt[i],obs_lay[i]-1,0,obs_col[i]-1]
    return C
```

Bibliography

Abramowitz, M. & Stegun, I. A. (1965), Handbook of mathematical functions with formulas, graphs, and mathematical table, *in* 'US Department of Commerce', National Bureau of Standards Applied Mathematics series 55.

Ahmad, M. I., Sinclair, C. D. & Werritty, A. (1988), 'Log-logistic flood frequency analysis', *Journal of Hydrology* **98**(3-4), 205–224.

Alapati, S. & Kabala, Z. J. (2000), 'Recovering the release history of a groundwater contaminant using a non-linear least-squares method', *Hydrological Processes* **14**(6), 1003–1016.

Anderson, J. L. (2007), 'Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter', *Physica D: Nonlinear Phenomena* **230**(1-2), 99–111.

Anderson, J. L. & Anderson, S. L. (1999), 'A monte carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts', *Monthly Weather Review* **127**(12), 2741–2758.

Aral, M. M., Guan, J. & Maslia, M. L. (2001), 'Identification of contaminant source location and release history in aquifers', *Journal of Hydrologic Engineering* **6**(3), 225–234.

Aureli, F., Prost, F., Vacondio, R., Dazzi, S. & Ferrari, A. (2020), 'A GPU-accelerated shallow-water scheme for surface runoff simulations', *Water* **12**(3), 637.

Ayvaz, M. T. (2010), 'A linked simulation–optimization model for solving the unknown groundwater pollution source identification problems', *Journal of Contaminant Hydrology* **117**(1-4), 46–59.

Bloomfield, J. P. & Marchant, B. P. (2013), 'Analysis of groundwater drought building on the standardised precipitation index approach', *Hydrology and Earth System Sciences* **17**(12), 4769–4787.
URL: *https://hess.copernicus.org/articles/17/4769/2013/*

Bruen, M. & Dooge, J. C. I. (2007), 'Harmonic analysis of the stability of reverse routing in channels', *Hydrology and Earth System Sciences* **11**(1), 559–568.

Brunner, G. W. (2010), *HEC-RAS, River Analysis System Hydraulic Reference Manual Version 4.1.*, US Army Corps of Engineers, Institute for Water resource, Hydrologic Engineering Center. Davis, California.

Butera, I. & Tanda, M. G. (2003), 'A geostatistical approach to recover the release history of groundwater pollutants', *Water Resources Research* **39**(12).

Butera, I., Tanda, M. G. & Zanini, A. (2006), 'Use of numerical modelling to identify the transfer function and application to the geostatistical procedure in the solution of inverse problems in groundwater', *Journal of Inverse and Ill-posed Problems* **14**(6), 547–572.

Butera, I., Tanda, M. G. & Zanini, A. (2012), 'Simultaneous identification of the pollutant release history and the source location in groundwater by means of a geostatistical approach', *Stochastic Environmental Research and Risk Assessment* **27**(5), 1269–1280.

Chen, Y. & Oliver, D. S. (2009), 'Cross-covariances and localization for EnKF in multiphase flow data assimilation', *Computational Geosciences* **14**(4), 579–601.

Chen, Z., Gómez-Hernández, J. J., Xu, T. & Zanini, A. (2018), 'Joint identification of contaminant source and aquifer geometry in a sandbox experiment with the restart ensemble kalman filter', *Journal of Hydrology* **564**, 1074–1084.

Chow, V. (1988), *Applied hydrology*, McGraw-Hill, New York.

Citarella, D., Cupola, F., Tanda, M. G. & Zanini, A. (2015), 'Evaluation of dispersivity coefficients by means of a laboratory image analysis', *Journal of Contaminant Hydrology* **172**, 10–23.

Crestani, E., Camporese, M., Baú, D. & Salandin, P. (2013), 'Ensemble kalman filter versus ensemble smoother for assessing hydraulic conductivity via tracer test data assimilation', *Hydrology and Earth System Sciences* **17**(4), 1517–1531.

Cupola, F., Tanda, M. G. & Zanini, A. (2014), 'Laboratory sandbox validation of pollutant source location methods', *Stochastic Environmental Research and Risk Assessment* **29**(1), 169–182.

Cupola, F., Tanda, M. G. & Zanini, A. (2015), 'Contaminant release history identification in 2-d heterogeneous aquifers through a minimum relative entropy approach', *SpringerPlus* **4**(1).

Das, A. (2009), 'Reverse stream flow routing by using muskingum models', *Sadhana* **34**(3), 483–499.

Datta, B., Beegle, J. E., Kavvas, M. L. & Orlob, G. T. (1989), *Development of an expert-system embedding pattern-recognition techniques for pollution-source identification. Report for 30 September 1987-29 November 1989.*

D'Oria, M., Ferraresi, M. & Tanda, M. G. (2017), 'Historical trends and high-resolution future climate projections in northern tuscany (italy)', *Journal of Hydrology* **555**, 708–723.

D'Oria, M., Mignosa, P. & Tanda, M. G. (2012), 'Reverse level pool routing: Comparison between a deterministic and a stochastic approach', *Journal of Hydrology* **470-471**, 28–35.

D'Oria, M., Mignosa, P. & Tanda, M. G. (2014), 'Bayesian estimation of inflow hydrographs in ungauged sites of multiple reach systems', *Advances in Water Resources* **63**, 143–151.

D'Oria, M. & Tanda, M. G. (2012), 'Reverse flow routing in open channels: A bayesian geostatistical approach', *Journal of Hydrology* **460-461**, 130–135.

Eli, R. N., Wiggert, J. M. & Contractor, D. N. (1974), 'Reverse flow routing by the implicit method', *Water Resources Research* **10**(3), 597–600.

Emerick, A. A. & Reynolds, A. C. (2012), 'History matching time-lapse seismic data using the ensemble kalman filter with multiple data assimilations', *Computational Geosciences* **16**(3), 639–659.

Emerick, A. A. & Reynolds, A. C. (2013), 'Ensemble smoother with multiple data assimilation', *Computers & Geosciences* **55**, 3–15.

Evensen, G. (1994), 'Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics', *Journal of Geophysical Research* **99**(C5), 10143.

Evensen, G. (2018), 'Analysis of iterative ensemble smoothers for solving inverse problems', *Computational Geosciences* **22**(3), 885–908.

Evensen, G. & van Leeuwen, P. J. (2000), 'An ensemble kalman smoother for nonlinear dynamics', *Monthly Weather Review* **128**(6), 1852–1867.

Ferrari, A., D'Oria, M., Vacondio, R., Palù, A. D., Mignosa, P. & Tanda, M. G. (2018), 'Discharge hydrograph estimation at upstream-ungauged sections by coupling a bayesian methodology and a 2-d GPU shallow water model', *Hydrology and Earth System Sciences* **22**(10), 5299–5316.

Gaspari, G. & Cohn, S. E. (1999), 'Construction of correlation functions in two and three dimensions', *Quarterly Journal of the Royal Meteorological Society* **125**(554), 723–757.

Gzyl, G., Zanini, A., Frączek, R. & Kura, K. (2014), 'Contaminant source and release history identification in groundwater: A multi-step approach', *Journal of Contaminant Hydrology* **157**, 59–72.

Hamill, T. M., Whitaker, J. S. & Snyder, C. (2001), 'Distance-dependent filtering of background error covariance estimates in an ensemble kalman filter', *Monthly Weather Review* **129**(11), 2776–2790.

Harbaugh, A. W. (2005), 'MODFLOW-2005: the u.s. geological survey modular ground-water model–the ground-water flow process'.

Hosking, J. R. (1986), *The theory of probability weighted moments*, IBM Research Division, TJ Watson Research Center.

Houtekamer, P. L. & Mitchell, H. L. (1998), 'Data assimilation using an ensemble kalman filter technique', *Monthly Weather Review* **126**(3), 796–811.

Jacob, D., Petersen, J., Eggert, B., Alias, A., Christensen, O. B., Bouwer, L. M., Braun, A., Colette, A., Déqué, M., Georgievski, G., Georgopoulou, E., Gobiet, A., Menut, L., Nikulin, G., Haensler, A., Hempelmann, N., Jones, C., Keuler,

K., Kovats, S., Kröner, N., Kotlarski, S., Kriegsmann, A., Martin, E., van Meijgaard, E., Moseley, C., Pfeifer, S., Preuschmann, S., Radermacher, C., Radtke, K., Rechid, D., Rounsevell, M., Samuelsson, P., Somot, S., Soussana, J.-F., Teichmann, C., Valentini, R., Vautard, R., Weber, B. & Yiou, P. (2013), 'EURO-CORDEX: new high-resolution climate change projections for european impact research', *Regional Environmental Change* **14**(2), 563–578.

Kahsay, K. D., Pingale, S. M. & Hatiye, S. D. (2018), 'Impact of climate change on groundwater recharge and base flow in the sub-catchment of tekeze basin, ethiopia', *Groundwater for Sustainable Development* **6**, 121–133.

Kalman, R. E. (1960), 'A new approach to linear filtering and prediction problems', *Journal of Basic Engineering* **82**(1), 35–45.

Koussis, A. D. & Mazi, K. (2016), 'Reverse flood and pollution routing with the lag-and-route model', *Hydrological Sciences Journal* pp. 1–15.

Kumar, R., Musuuza, J. L., Van Loon, A. F., Teuling, A. J., Barthel, R., Ten Broek, J., Mai, J., Samaniego, L. & Attinger, S. (2016), 'Multiscale evaluation of the standardized precipitation index as a groundwater drought indicator', *Hydrology and Earth System Sciences* **20**(3), 1117–1131.
URL: *https://hess.copernicus.org/articles/20/1117/2016/*

Leelaruban, N., Padmanabhan, G. & Oduor, P. (2017), 'Examining the relationship between drought indices and groundwater levels', *Water* **9**(2), 82.

Leonhardt, G., D'Oria, M., Kleidorfer, M. & Rauch, W. (2014), 'Estimating inflow to a combined sewer overflow structure with storage tank in real time: evaluation of different approaches', *Water Science and Technology* **70**(7), 1143–1151.

Li, H., Kalnay, E. & Miyoshi, T. (2009), 'Simultaneous estimation of covariance

inflation and observation errors within an ensemble kalman filter', *Quarterly Journal of the Royal Meteorological Society* **135**(639), 523–533.

Li, L., Stetler, L., Cao, Z. & Davis, A. (2018), 'An iterative normal-score ensemble smoother for dealing with non-gaussianity in data assimilation', *Journal of Hydrology* **567**, 759–766.

Liang, X., Zheng, X., Zhang, S., Wu, G., Dai, Y. & Li, Y. (2011), 'Maximum likelihood estimation of inflation factors on error covariance matrices for ensemble kalman filter assimilation', *Quarterly Journal of the Royal Meteorological Society* **138**(662), 263–273.

Mahar, P. S. & Datta, B. (1997), 'Optimal monitoring network and groundwater–pollution source identification', *Journal of Water Resources Planning and Management* **123**(4), 199–207.

McKee, T. B., Doesken, N. J. & Kleist, J. (1993), The relationship of drought frequency and duration to time scales, *in* 'Proceedings of the 8th Conference on Applied Climatology, Anaheim, CA, USA, 17–22 January 1993', pp. 179–183.

Michalak, A. M. & Kitanidis, P. K. (2004*a*), 'Application of geostatistical inverse modeling to contaminant source identification at dover AFB, delaware', *Journal of Hydraulic Research* **42**(sup1), 9–18.

Michalak, A. M. & Kitanidis, P. K. (2004*b*), 'Estimation of historical groundwater contaminant distribution using the adjoint state method applied to geostatistical inverse modeling', *Water Resources Research* **40**(8).

Nash, J. E. & Sutcliffe, J. V. (1970), 'River flow forecasting through conceptual models part i — a discussion of principles', *Journal of Hydrology* **10**(3), 282–290.

Neupauer, R. M., Borchers, B. & Wilson, J. L. (2000), 'Comparison of inverse

methods for reconstructing the release history of a groundwater contamination source', *Water Resources Research* **36**(9), 2469–2475.

Pachauri, R. K., Allen, M. R., Barros, V. R., Broome, J., Cramer, W., Christ, R., Church, J. A., Clarke, L., Dahe, Q., Dasgupta, P. et al. (2014), *Climate change 2014: synthesis report. Contribution of Working Groups I, II and III to the fifth assessment report of the Intergovernmental Panel on Climate Change*, Ipcc.

Pirot, G., Krityakierne, T., Ginsbourger, D. & Renard, P. (2019), 'Contaminant source localization via bayesian global optimization', *Hydrology and Earth System Sciences* **23**(1), 351–369.

Prost, F. (2019), Un solutore 2D alle acque basse parallelizzato su GPU per la modellistica idrodinamica a scala di bacino., PhD thesis, Università degli Studi di Parma.

Saghafian, B., Jannaty, M. H. & Ezami, N. (2014), 'Inverse hydrograph routing optimization model based on the kinematic wave approach', *Engineering Optimization* **47**(8), 1031–1042.

Skaggs, T. H. & Kabala, Z. J. (1994), 'Recovering the release history of a groundwater contaminant', *Water Resources Research* **30**(1), 71–79.

Snodgrass, M. F. & Kitanidis, P. K. (1997), 'A geostatistical approach to contaminant source identification', *Water Resources Research* **33**(4), 537–546.

Sun, A. Y., Painter, S. L. & Wittmeyer, G. W. (2006), 'A robust approach for iterative contaminant source location and release history recovery', *Journal of Contaminant Hydrology* **88**(3-4), 181–196.

Szymkiewicz, R. (1993), 'Solution of the inverse problem for the saint venant equations', *Journal of Hydrology* **147**(1-4), 105–120.

Thornthwaite, C. W. (1948), 'An approach toward a rational classification of climate', *Geographical Review* **38**(1), 55.

Todaro, V., D'Oria, M., Tanda, M. G. & Gómez-Hernández, J. J. (2019), 'Ensemble smoother with multiple data assimilation for reverse flow routing', *Computers & Geosciences* **131**, 32–40.

Uribe-Asarta, J. (2019), Modelación numérica de un experimento de transporte de masa en un tanque de arena de laboratorio, Master's thesis, Universitat Politècnica de València.

van Engelenburg, J., Hueting, R., Rijpkema, S., Teuling, A. J., Uijlenhoet, R. & Ludwig, F. (2017), 'Impact of changes in groundwater extractions and climate change on groundwater-dependent ecosystems in a complex hydrogeological setting', *Water Resources Management* **32**(1), 259–272.

van Leeuwen, P. J. & Evensen, G. (1996), 'Data assimilation and inverse methods in terms of a probabilistic formulation', *Monthly Weather Review* **124**(12), 2898–2913.

Vicente-Serrano, S. M., Beguería, S. & López-Moreno, J. I. (2010), 'A multiscalar drought index sensitive to global warming: The standardized precipitation evapotranspiration index', *Journal of Climate* **23**(7), 1696–1718.

Wang, X. & Bishop, C. H. (2003), 'A comparison of breeding and ensemble transform kalman filter ensemble forecast schemes', *Journal of the Atmospheric Sciences* **60**(9), 1140–1158.

Woodbury, A. D. & Ulrych, T. J. (1996), 'Minimum relative entropy inversion: Theory and application to recovering the release history of a groundwater contaminant', *Water Resources Research* **32**(9), 2671–2681.

Woodbury, A., Sudicky, E., Ulrych, T. J. & Ludwig, R. (1998), 'Three-dimensional plume source reconstruction using minimum relative entropy inversion', *Journal of Contaminant Hydrology* **32**(1-2), 131–158.

Xu, T. & Gómez-Hernández, J. J. (2016), 'Joint identification of contaminant source location, initial release time, and initial solute concentration in an aquifer via ensemble kalman filtering', *Water Resources Research* **52**(8), 6587–6595.

Xu, T. & Gómez-Hernández, J. J. (2018), 'Simultaneous identification of a contaminant source and hydraulic conductivity via the restart normal-score ensemble kalman filter', *Advances in Water Resources* **112**, 106–123.

Xu, T., Gómez-Hernández, J. J., Chen, Z. & Lu, C. (2020), 'A comparison between ES-MDA and restart EnKF for the purpose of the simultaneous identification of a contaminant source and hydraulic conductivity', *Journal of Hydrology* p. 125681.

Zanini, A. & Woodbury, A. D. (2016), 'Contaminant source reconstruction by empirical bayes and akaike's bayesian information criterion', *Journal of Contaminant Hydrology* **185-186**, 74–86.

Zheng, C. & Wang, P. P. (1999), 'MT3DMS : a modular three-dimensional multispecies transport model for simulation of advection, dispersion, and chemical reactions of contaminants in groundwater systems; documentation and user's guide'.

Zheng, X. (2009), 'An adaptive estimation of forecast error covariance parameters for kalman filtering data assimilation', *Advances in Atmospheric Sciences* **26**(1), 154–160.

Zhou, H., Gómez-Hernández, J. J., Franssen, H.-J. H. & Li, L. (2011), 'An ap-

proach to handling non-gaussianity of parameters and state variables in ensemble kalman filtering', *Advances in Water Resources* **34**(7), 844–864.

Zoppou, C. (1999), 'Reverse routing of flood hydrographs using level pool routing', *Journal of Hydrologic Engineering* **4**(2), 184–188.

Zucco, G., Tayfur, G. & Moramarco, T. (2015), 'Reverse flood routing in natural channels using genetic algorithm', *Water Resources Management* **29**(12), 4241–4267.