Modelling and Simulation of Large Scale Distributed Parameter Systems

by

Issam Sullivan Strub-Brahimi

Maîtrise (Ecole Normale Supérieure, Cachan) 2002
DEA (Ecole Normale Supérieure, Cachan) 2003

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering Science - Civil and Environmental Engineering

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor A. M. Bayen, Chair
Professor L. C. Evans
Professor M. T. Stacey

Spring 2009

The dissertation of Issam Sullivan Strub-Brahimi is approved:

Chair Date

Date

Date

University of California, Berkeley

Spring 2009

Modelling and Simulation of Large Scale Distributed Parameter Systems

by

Issam Sullivan Strub-Brahimi

Abstract

Modelling and Simulation of Large Scale Distributed Parameter Systems

by

Issam Sullivan Strub-Brahimi

Doctor of Philosophy in Engineering Science - Civil and Environmental Engineering

University of California, Berkeley

Professor A. M. Bayen, Chair

The research presented in this dissertation is motivated by the need for modelling, simulation and optimisation of large scale distributed parameter systems, namely rivers and highways.

The problem of state estimation for two-dimensional river flows is investigated using a novel algorithm for two-dimensional Lagrangian data assimilation of shallow water flows and floating sensors. This algorithm is based on a quadratic programming formulation with the linearised two-dimensional shallow water equations used as constraints. It is compared in computer-based twin experiments with an ensemble Kalman filtering algorithm, and the performance of the two algorithms is evaluated in a number of settings. The sensitivity of the two data assimilation algorithms to the number of drifters, low or high discharge and time sampling frequency is analysed and the respective computational costs of each method compared. One of the conclusion is that the quadratic programming based algorithm introduced by the author presents a good balance of accuracy and low computational cost. The quadratic programming based algorithm is also applied to experimental drifter data collected during field experiments.

Another problem is the estimation of open boundary conditions in situations in which tidal forcing is dominant. A quadratic programming based variational data assimilation algorithm is applied to the estimation of open boundary conditions for tidal flows using one-dimensional shallow water equations and floating sensors. This algorithm is evaluated in computer generated experiments both with and without tidal reversion. The experiments show that during the period of time that observations are available, the inverse model is able to effectively estimate the harmonic constants and reproduce the flows.

The final part of the dissertation deals highway traffic modelling. We prove the existence and uniqueness of a weak solution to a scalar conservation law on a bounded domain through the introduction of a weak formulation of the boundary conditions. This result is applied to the Lighthill-Whitham-Richards model and discretised using a Godunov scheme. This numerical scheme is validated through a comparison with experimentally measured data. Finally, the existence of a minimiser of travel time is obtained, with corresponding optimal boundary control.

Professor A. M. Bayen
Dissertation Committee Chair

Contents

Acknowledgements

I would like to thank Prof. Lawrence Craig Evans for his incomparable help and support ever since I first came to the University of California during Summer 2001. Prof. Alexandre M. Bayen's thoughtful advising was greatly appreciated. I am indebted to Prof. Mark T. Stacey for his advice on environmental fluid mechanics which was very helpful for the research presented in the first two chapters. Prof. Laurent El Ghaoui provided crucial advice on convex optimisation techniques and other topics. I thank Prof. Claire Tomlin and Prof. Raja Sengupta for agreeing to be part of my qualifying examination committee and Prof. Carlos Daganzo for being part of my preliminary examination committee.

I also thank staff members from the Mathematics department, Thomas Brown, Haruko Bruce, Marsha Snow. Barbara Peavy and Catherine Pauling were very dedicated and efficient in helping the graduate student instructors and their students. I'm also grateful to Institute of Transportation Studies staff members Jillene Bohr, Meriel Ennik, and Civil and Environmental Engineering department staff members Joan Chamberlain and Shelley Okimoto.

Julie Percelay and Olli Pekka Tossavainen developed the Ensemble Kalman Filtering algorithm and Andrew Tinka provided the experimental drifter data. A number of students from the Civil and Environmental Engineering department, Saurabh Amin, Juan Carlos Herrera, Dengfeng Sun, Qingfang Wu should also be thanked.

Chapter 1

Introduction

1.1 Motivation

Distributed parameter systems can be found in many aspects of life such as biology [5], economics, environmental sciences [147], transportation [18, 146, 149], etc. In order to analyse, estimate, forecast and in some cases control the evolution of such systems, models typically based on ordinary or partial differential equations have been developed and used in the past. A model can be defined as a representation of a system, intended to facilitate understanding of the interactions of the elements of the system and its evolution as a whole. The extent to which a model helps in the development of our understanding of a system is the basis for deciding how good the model is. Most models describe a simplified version of the actual system in mathematical terms, hence there is a trade-off between the level of detail included in the model and its complexity. If too little detail is included, important aspects of the system may be missed whereas an excess of detail would render the model too complicated and causing it to become an obstacle to the understanding of the system or even its computational tractability. A simulation intends to describe the evolution of a system over time and space by algorithmic manipulations of the model usually done through a computer. Once a simulation is run (usually on a simplified test case), the resulting predicted evolution of the system can be compared with the observed evolution of the actual system in order to evaluate the accuracy of the model. If the agreement between the simulated and observed evolution is not deemed satisfactory, the model is modified accordingly and a new simulation is run; this cycle can be repeated until the model gives a representation of the system with the desired accuracy. Note that as part of this process, model calibration i.e. tuning of parameters plays a essential role. In this thesis, the development and improvement of models and simulations are studied in the context of environmental (rivers) and transportation (highways) systems. In some cases, methods have been developed with the objective of improving the model directly from the observed data; these methods fall into the field of inverse modelling and make up a large part of this dissertation.

1.2 Contributions of the thesis

The theoretical contributions of this thesis include:

- Development of a novel quadratic programming based algorithm for Lagrangian river state estimation. The application of quadratic programming to this problem was previously unknown, and enables the incorporation of unknown initial conditions as part of the search space.

- Development of a novel quadratic programming based algorithm for Lagrangian tidal flow estimation. This algorithm estimates tidally driven upstream boundary conditions continuously over an extended period of time (50 hours or more).

- Development of three numerical schemes for two dimensional linearised shallow water equations. These numerical schemes are a key part of the aforementioned algorithm for river state estimation.

- Development of a numerical scheme for non-orthogonal curvilinear coordinates. This allows the implementation of the state estimation algorithm to any river, irrespective of its geometry.

- Proof of existence and uniqueness of weak solutions for scalar conservations laws on a bounded domain. This defines a weak formulation of boundary conditions that makes the Lighthill-Whitham-Richards traffic flow model well posed.

- Development of a numerical scheme for the simulation of highway traffic flow, proof of convergence of the scheme.
- Development of a boundary control method for scalar conservation laws and application to highway congestion mitigation. This yields a strategy for congestion reduction based on ramp metering.

The implementations and numerical work presented in this thesis include:

- Implementation of these schemes for the river state assimilation algorithm.
- Implementation of the Preissmann scheme for tidal flow estimation.
- Data processing: Bathymetry and boundary conditions collection for the Sacramento River (eight years of hourly data).
- Twin experiments and comparisons between the quadratic programming and the ensemble Kalman filtering based river state assimilation algorithms in different settings.
- Implementation of the quadratic programming based river state assimilation algorithm on data collected from field experiments.
- Twin experiments for the tidal flow estimation algorithm.
- Boundary conditions collection, implementation of a numerical scheme, simulation and validation for highway traffic flow. These results show that the Lighthill-Whitham-Richards model can reproduce highway traffic flow accurately.

1.3 Organisation of this work

The thesis is organised as follows. Chapter 2 presents the quadratic programming based state assimilation algorithm as well as other existing data assimilation methods and compares the new algorithm with the ensemble Kalman filtering (EnKF) method on both computer simulated twin experiments and experimental data collected on the Sacramento River. In chapter 3, another quadratic programming based algorithm is introduced for the estimation of tidally induced boundary conditions and its efficiency analysed through twin experiments. In chapter 4, we turn to the field of highway traffic, beginning with a proof of existence and uniqueness result for scalar conservation laws and then developing a numerical scheme for the simulation of highway traffic flow which is evaluated by comparison with experimentally measured data. The concluding chapter 5 summarises the work presented in this thesis. The research on two-dimensional river state estimation presented in chapter 2 was published in journal article [148] and the part dealing with experimental data is part of journal article [155]. The research on inverse modelling of tidal flows from chapter 3 was published in journal article [147]. The work on highway traffic modelling from chapter 4 was published in journal article [146].

Chapter 2

State assimilation for river flows

2.1 Introduction

The modelling and monitoring of river hydraulics are increasingly important as they provide drinkable water for populations as well as irrigation for a variety of crops. These flows are usually modelled using the shallow water equations, whether in a one or two dimensional formulation [32]. In this approximation, the flow is assumed to be nearly horizontal and the water column well-mixed. Since these models are an approximation of the flow, they are bound to gradually drift away from the physical phenomenon and a realistic modelling requires the use of experimental measurements to keep the model accurate with respect to the actual flow. This constitutes the main objective of data assimilation, a technique which originated several decades ago in the fields of meteorology and oceanography [11, 97]. A first example of data assimilation can be found in the prediction by Gauss in [53] of the reappearance of Ceres, a planetoid which had disappeared behind the Sun in 1801, only to reappear a year later. The orbit of this planetoid was predicted using assumptions of a circular orbit as well as observations, all of which were combined with the help of the least squares method; the result was the Gauss was able not only to predict the timing of this event but also to direct astronomers to the exact location of its reappearance. Over the years, a number of different techniques have been introduced, some of which rely on variational methods [120, 123] while others use Kalman filtering and its extensions

[47, 94], optimal statistical interpolation [117], or more empirical techniques such as Newtonian relaxation [81, 127]. A few introductions to data assimilation can be found in [19, 20, 23, 77, 105, 116, 152]. In this chapter, this technique will be applied to river flows in order to estimate the river state at any time instant (starting from the initial state). This is the first time that this technique is applied to the specific setting of rivers, with the unknown being the initial state.

2.2 Shallow water equations

2.2.1 Derivation of the shallow water equations

We consider the flow of water with a free surface under gravity in a three-dimensional domain. The bed elevation is represented by a function $z = b(x, y)$. The water depth is $h(x, y, t)$ which yields the free surface $s(x, y, t) = b(x, y) + h(x, y, t)$. The velocity field is denoted by $\mathbf{V} = (u, v, w)$. The conservation of mass and momentum equations for a compressible fluid are given by:

$$\rho_t + \nabla \cdot (\rho \mathbf{V}) = 0 \tag{2.1}$$

$$\frac{\partial (\rho \mathbf{V})}{\partial t} + \nabla \cdot (\rho \mathbf{V} \otimes \mathbf{V} + p\mathbf{I} - \Pi) - \rho \mathbf{g} = 0 \tag{2.2}$$

where ρ and p denote the density and pressure, $\mathbf{g} = (g_1, g_2, g_3)$ is a body force vector and Π is the viscous stress tensor defined by:

$$\Pi = \begin{pmatrix} \tau^{xx} & \tau^{xy} & \tau^{xz} \\ \tau^{yx} & \tau^{yy} & \tau^{yz} \\ \tau^{zx} & \tau^{zy} & \tau^{zz} \end{pmatrix}$$

Expressing the preceding equations in terms of the primitive variables, we obtain the set of equations:

$$\rho_t + u\rho_x + v\rho_y + w\rho_z + \rho(u_x + v_y + w_z) = 0 \tag{2.3}$$

$$u_t + uu_x + vu_y + wu_z + \frac{1}{\rho}p_x - \frac{1}{\rho}(\tau^{xx}_x + \tau^{xy}_y + \tau^{xz}_z) = g_1 \tag{2.4}$$

$$v_t + uv_x + vv_y + wv_z + \frac{1}{\rho}p_y - \frac{1}{\rho}(\tau^{yx}_x + \tau^{yy}_y + \tau^{yz}_z) = g_2 \tag{2.5}$$

$$w_t + uw_x + vw_y + ww_z + \frac{1}{\rho}p_z - \frac{1}{\rho}(\tau^{zx}_x + \tau^{zy}_y + \tau^{zz}_z) = g_3 \tag{2.6}$$

In the case of open channel flow, the body force vector is $\mathbf{g} = (0, 0, -g)$ where g is the acceleration of gravity, and the fluid being incompressible, the density ρ is assumed to be uniform. The preceding equations thus simplify to:

$$u_x + v_y + w_z = 0 \tag{2.7}$$

$$u_t + uu_x + vu_y + wu_z + \frac{1}{\rho}p_x - \frac{1}{\rho}(\tau_x^{xx} + \tau_y^{xy} + \tau_z^{xz}) = 0 \tag{2.8}$$

$$v_t + uv_x + vv_y + wv_z + \frac{1}{\rho}p_y - \frac{1}{\rho}(\tau_x^{yx} + \tau_y^{yy} + \tau_z^{yz}) = 0 \tag{2.9}$$

$$w_t + uw_x + vw_y + ww_z + \frac{1}{\rho}p_z - \frac{1}{\rho}(\tau_x^{zx} + \tau_y^{zy} + \tau_z^{zz}) = -g \tag{2.10}$$

In the approximation of a nearly horizontal (or shallow) flow, the vertical component of the acceleration $\frac{dw}{dt} = w_t + uw_x + vw_y + ww_z$ is negligible and the z component of the conservation of momentum is reduced to the hydrostatics law: $p_z = -\rho g$. After assuming the atmospheric pressure is zero on the free surface, and integrating the continuity equation on the height of water, we obtain the two-dimensional shallow water equations [32, 156, 162]. In our case, we also neglect additional terms such as Coriolis forces and wind forces:

$$u_t + uu_x + vu_y + gh_x + gb_x - f_x = 0 \tag{2.11}$$

$$v_t + uv_x + vv_y + gh_y + gb_y - f_y = 0 \tag{2.12}$$

$$h_t + uh_x + vh_y + hu_x + hv_y = 0 \tag{2.13}$$

where (f_x, f_y) denote the viscous forces.

2.2.2 Initial and boundary conditions

For the two-dimensional shallow water equations, the correct number of boundary conditions can be found using the theory of characteristics. Depending on whether the flow at a given open boundary if sub- or supercritical and whether this is an inflow or outflow boundary (which is given by the sign of the normal velocity at the boundary), from zero to three boundary conditions have to be prescribed at each boundary [161, 162]. In this thesis, the flow considered are always subcritical ($\mathrm{Fr} = \frac{U}{\sqrt{gh}} < 1$) on the entire domain;

therefore two boundary conditions are required at the upstream boundary (inflow) and one at the downstream boundary (outflow). Upstream, we specify the water level and the normal and tangential components of velocity, the normal component being set equal to zero and the downstream boundary condition is given by the water level. For land boundaries, one boundary condition is needed. We set the normal velocity to zero and depending on the presence of a second order viscosity term in the shallow water model used, a viscous boundary layer may occur and a no slip condition is used.

If we call Γ_u, Γ_d and Γ_l the upstream, downstream and land boundaries, and $(u_{\mathbf{N}}, u_{\mathbf{T}})$ the normal and tangent velocity vectors, the boundary conditions can be written as:

$$\begin{cases} h(x,y,t) = h_u(x,y,t), u_{\mathbf{N}}(x,y,t) = u_u(x,y,t), u_{\mathbf{T}}(x,y,t) = 0, & (x,y) \in \Gamma_u \\ h(x,y,t) = h_d(x,y,t), & (x,y) \in \Gamma_d \\ u_{\mathbf{N}}(x,y,t) = 0, u_{\mathbf{T}}(x,y,t) = 0 & (x,y) \in \Gamma_l \end{cases} \tag{2.14}$$

where $h_u(x,y,t)$, $u_u(x,y,t)$ and $h_d(x,y,t)$ are prescribed. Additionally, the initial distribution of water level and velocity is specified for the entire domain.

2.3 Data assimilation methods

Data assimilation methods can be divided into two types (see [23] for a detailed presentation of most existing methods). Sequential methods rely on statistics to improve the model prediction as new observations are gathered. The most common sequential methods are optimal interpolation and Kalman filtering and its variations such as extended Kalman filtering and ensemble Kalman filtering. The object of optimal interpolation is to find an optimal state in the form of a linear combination of observations and model predicted states which minimises the variance of the estimation error. This solution is called *Best Linear Unbiased Estimator* (BLUE). More details about optimal interpolation can be found in [41, 42, 106, 110, 117, 129]. When this method is extended to a time varying setting, we obtain the Kalman filter [51, 52, 58, 60, 61, 64, 86, 88, 159]. A later evolution of this method, the *ensemble Kalman filter* (EnKF) [47] will be presented later in this section.

Unlike sequential methods, variational methods [97, 102, 108, 118, 123, 140, 151, 154] use all the observations available up to the end of the period considered and minimise a cost function containing both modelled and observed variables. Two algorithms of variational data assimilation will be presented, one relying on the adjoint method and the other on quadratic programming. Another possibility consists in expressing the model equations (in our case the shallow water equations) in Lagrangian coordinates which allows us to directly assimilate Lagrangian drifter positions [113, 114, 115].

2.3.1 Notations

We start this section by defining all the variables which will appear in the following. We employ the traditional notations of variational data assimilation, set forth in [78].

X_n : vector of state variables, namely the velocity components (u, v) and the water height h for each mesh point at a time instant t_n,

X_0 : initial state of the system, which respect to which the minimisation is done,

X^b : background term of the same size as X_0, which is introduced to ensure that the problem is well-posed and has a unique minimum,

Y_n : vector of observed variables, namely the velocity components (u, v) and (potentially) the water height h for some mesh points at a time instant t_n,

$\mathbf{B}$: covariance matrix of the background error (difference between the value of the state variables and background variables at each mesh point), taken equal to the identity in our case,

$\mathbf{R}_n$: covariance matrix of the observation error (difference between the value of the state variables and observed variables at each mesh point), also taken equal to the identity in this thesis,

H_n : $H_n = h_n^o \circ h^I$ is the observation operator; the operator h_n^o projects the space into the observation subspace. The operator h^I is the interpolation function. In general H_n is nonlinear although we manage to use a linear operator in our case by using the a posteriori knowledge of the position of the measurements, therefore encoded as a time varying observation matrix.

Variational data assimilation consists in obtaining the initial state that minimises a cost function representing the L^2 norm of the difference between the state and observed variables:

$$\mathcal{J}^o(X_0) = \sum_{n=0}^{N} (Y_n^o - H_n[X_n])^T \mathbf{R}_n^{-1} (Y_n^o - H_n[X_n]) \tag{2.15}$$

where the subscript n denotes the time step. The number of points in the computational domain is usually much greater than the number of measurements located along drifter trajectories, usually by a factor 10^2 or more. This can make the minimisation problem ill-posed; in particular, non uniqueness of the solution may result from the ill-posedness. Thus, a term representing the L^2 norm of the difference between an estimate of the initial state X^b and the observed variables, called background term $\mathcal{J}^b$, is introduced leading to a unique minimum of the cost function and providing a first guess which accelerates the convergence of the minimisation algorithm. Such an estimate of the initial state of the system may be available, for example, through historical data, simulation from a previous assimilated initial state, forecast, or interpolation from a limited number of fixed Eulerian sensors located at the boundaries of the system. This leads to a new cost function, by adding the background term $\mathcal{J}^b$ to the observation term $\mathcal{J}^o$:

$$\mathcal{J}(X_0) = (X_0 - X^b)^T \mathbf{B}^{-1} (X_0 - X^b) + \sum_{n=0}^{N} (Y_n^o - H_n[X_n])^T \mathbf{R}_n^{-1} (Y_n^o - H_n[X_n]) \tag{2.16}$$

$\mathbf{B}$ and $\mathbf{R}_n$ are the covariance matrices of the background error and observation error respectively and will taken equal to the identity in a simplifying assumption. Note that $\mathcal{J}^o(X_0)$ depends on X_0 implicitly through the state X_n, obtained from X_0 through the model equations, as will be detailed later.

2.3.2 Adjoint method

The constraints are given by the model equations, in our case the 2-dimensional nonlinear shallow water equations described in the previous section, discretised by a numerical scheme. To solve such a nonlinear optimisation problem with a large number of variables, the adjoint method can be used (a primer on the adjoint method can be found in [104, 122]). Adjoint methods were first introduced in the late 1980s as a tool for shape optimisation, in particular aircraft design [83] and were later applied to data assimilation [97, 120]. The direct approach which consists in calculating the gradient of the cost functional using finite differences is only possible when the number of control variables is small. In most real life problems, in particular data assimilation, this number is too large, making this approach untractable. A more efficient way of calculating gradients is to use the adjoint equations and boundary conditions, which can be solved using numerical schemes to yield the gradient of the cost function. Several variations of the adjoint methods exist depending on the stage at which the discretisation is done: the continuous adjoint method, the discrete adjoint method and automatic differentiation. In the continuous case, the adjoint equations and boundary conditions are computed from the model and eventually discretised. For the discrete adjoint method, the model equations are discretised first using a numerical scheme, then the adjoint equations of these discrete equations are computed. Finally, one can use automatic differentiation software to compute the adjoint of the numerical code that is used to simulate the model equations. This results in a adjoint code being generated which can be implemented directly into a nonlinear optimisation routine. An application of automatic differentiation to data assimilation of river flows appeared in [31]; adjoint methods for river control are developed in [136].

We start by illustrating this technique on a simple optimisation problem:

$$\min_x F(x, u(x)) \text{ subject to the constraint } C(x, u(x)) = 0$$

where $u = (u_1, \ldots, u_n)$, $C = (C_1, \ldots, C_n)$, $x = (x_1, \ldots, x_m)$. For a small variation of the control variable $x \rightarrow x + \varepsilon\bar{x}$, the variation in u will be:

$$u(x) \rightarrow u(x + \varepsilon\bar{x}) = u(x) + \varepsilon\bar{u} + O(\varepsilon^2)$$

A linearisation of the constraint equation gives a relation between $\bar{x}$ and $\bar{u}$:

$$\frac{\partial C}{\partial x}\bar{x} + \frac{\partial C}{\partial u}\bar{u} = 0$$

The variation of F is:

$$F(x+\varepsilon\bar{x}, u(x+\varepsilon\bar{x})) - F(x, u(x)) = \delta F = \varepsilon\left(\bar{x}^T\frac{\partial F}{\partial x} + \bar{u}^T\frac{\partial F}{\partial u}\right) + O(\varepsilon^2)$$

which would suggest using $\bar{x}^T\frac{\partial F}{\partial x} + \bar{u}^T\frac{\partial F}{\partial u}$ as a descent direction. However this quantity depends on $\bar{u}$, the first variation of u which can only be computed after setting a direction of change $\bar{x}$ for the control variable x. To avoid these complications, we try to eliminate $\bar{u}$ from the variation of the functional F. Noticing that $\bar{u}^T\frac{\partial C}{\partial u}^T + \bar{x}^T\frac{\partial C}{\partial x}^T = 0$, we multiply this quantity by an arbitrary vector $\lambda = (\lambda_1, \ldots, \lambda_n)$ and we add this term (of value zero) to the variation of the functional. Then choosing a specific value for λ, we will eliminate the dependence in $\bar{u}$:

$$\delta F = \varepsilon\left(\bar{x}^T\frac{\partial F}{\partial x} + \bar{u}^T\frac{\partial F}{\partial u}\right) + \varepsilon\left(\bar{u}^T\frac{\partial C}{\partial u}^T + \bar{x}^T\frac{\partial C}{\partial x}^T\right)\lambda + O(\varepsilon^2)$$

which can be written as:

$$\delta F = \varepsilon\bar{x}^T\left(\frac{\partial F}{\partial x} + \frac{\partial C}{\partial x}^T\lambda\right) + \varepsilon\bar{u}^T\left(\frac{\partial F}{\partial u}^T + \frac{\partial C}{\partial u}^T\lambda\right) + O(\varepsilon^2)$$

If we choose λ such that $\frac{\partial F}{\partial u}^T + \frac{\partial C}{\partial u}^T\lambda$=0, the variation of the functional becomes:

$$\delta F = \varepsilon\bar{x}^T\left(\frac{\partial F}{\partial x} + \frac{\partial C}{\partial x}^T\lambda\right) + O(\varepsilon^2)$$

with the additional constraint:

$$\frac{\partial F}{\partial u}^T + \frac{\partial C}{\partial u}^T\lambda = 0$$

which is called the adjoint equation. The variation of the functional no longer depends explicitly on $\bar{u}$, and a descent direction is:

$$\bar{x} = -\left(\frac{\partial F}{\partial x} + \frac{\partial C}{\partial x}^T\lambda\right)$$

At each iteration, we solve the original and adjoint equations using a numerical scheme and modify the descent direction accordingly into a nonlinear optimisation method.

We will now present the adjoint method on a general evolution equation:

$$\begin{cases} \frac{dX(t)}{dt} = M(X(t), t), \quad t \in (0, T) \\ X(0) = X_0 \end{cases} \tag{2.17}$$

where M represents the continuous model, and X_0 the initial state which is the control variable in this case. We consider the following general cost function which we try to minimise:

$$J(X_0) = \int_0^T \left(||H(t, X(t)) - Y^o||^2 + r||X_0 - X^b||^2 \right) dt \tag{2.18}$$

where H is the observation operator and Y^o the observation vector. The minimiser X^* solves the following system of equations :

$$\begin{cases} \frac{dX}{dt} = M(X(t), t), \quad t \in (0, T) \\ X(0) = X^* \end{cases} \tag{2.19}$$

and the adjoint equations [104, 122]:

$$\begin{cases} -\frac{dP}{dt} = \left[\frac{\partial M}{\partial X}\right]^* P(t) + \left[\frac{\partial H}{\partial X}\right]^* (H(t, X(t)) - Y^o) \\ P(T) = 0, \ - P(0) + r(X^* - X^b) = 0 \end{cases} \tag{2.20}$$

where P denotes the adjoint variable, and the linear operators $\left[\frac{\partial M}{\partial X}\right]^*$, $\left[\frac{\partial H}{\partial X}\right]^*$ correspond to the adjoint of the model M and the observation operator H. At each iteration the model and adjoint equations are solved yielding the gradient of the cost functional:

$$\nabla J(X) = -P(0) + r(X - X^b) \tag{2.21}$$

This gradient is then used in a nonlinear optimisation method, such as the limited memory version of the *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) method described in [29, 121, 170]. The adjoint equations for our problem can be obtained from the conservative formulation of the shallow water equations [145]:

$$X_t + F(X)_x + G(X)_y = S \tag{2.22}$$

where

$$X = \begin{pmatrix} h \\ uh \\ vh \end{pmatrix} \tag{2.23}$$

and

$$F = \begin{pmatrix} uh \\ u^2h + g\frac{h^2}{2} \\ uvh \end{pmatrix} \tag{2.24}$$

$$G = \begin{pmatrix} vh \\ uvh \\ v^2h + g\frac{h^2}{2} \end{pmatrix} \tag{2.25}$$

$$S = \begin{pmatrix} 0 \\ -ghb_x \\ -ghb_y \end{pmatrix} \tag{2.26}$$

from which we obtain the adjoint equations [136, 97]:

$$P_t + AP_x + BP_y + CP + R = 0 \tag{2.27}$$

where $q_1 = uh$, $q_2 = vh$ and A, B, C are given by:

$$A = \begin{pmatrix} 0 & \frac{q_1^2}{h^2} - gh & \frac{q_1 q_2}{h^2} \\ -1 & -2\frac{q_1}{h} & -\frac{q_2}{h} \\ 0 & 0 & -\frac{q_1}{h} \end{pmatrix} \tag{2.28}$$

$$B = \begin{pmatrix} 0 & \frac{q_1 q_2}{h^2} & \frac{q_2^2}{h^2} - gh \\ 0 & -\frac{q_2}{h} & 0 \\ -1 & -\frac{q_1}{h} & -2\frac{q_2}{h} \end{pmatrix} \tag{2.29}$$

$$C = \begin{pmatrix} 0 & -gb_x & -gb_y \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{2.30}$$

where the vector R is related to the derivatives of the cost function.

Note that the adjoint equations are not in conservative form which prevents the implementation of the numerical schemes used for the model equations. This problem can be circumvented using automatic differentiation [21] to generate the adjoint code directly from the forward code. The main drawback of the adjoint method is its computational cost as 50 to 100 iterations of the method are usually required for the method to converge which corresponds to 100 to 200 numerical resolutions of the direct and adjoint equations. Additionally, the nonlinearity of the problem can result in the fact that if the starting point (in our case the background term) is too far from the global minimum, the nonlinear optimisation method might converge to a local minimum instead.

2.3.3 Quadratic programming based data assimilation algorithm

While the adjoint method is helpful to solve nonlinear optimisation problems, in the case of a quadratic cost functions with linear constraints, a quadratic programming solver will be far more efficient and yield an optimal solution. Therefore, we will formulate the variational data assimilation problem 2.16 detailed earlier as a quadratic program. The first step is to obtain linear constraints, which in our case will be achieved by linearising the shallow water equations. Indeed, for numerous environmental flows, such as the ones we will consider in our experiments, the Froude number is smaller than 1, usually by several orders of magnitude, (Fr $\leqslant 0.05$ in our case), and the nonlinear effects can be neglected. For this setting, we can rewrite the equations (2.11), (2.12) and (2.13) under the form:

$$u_t + uu_x + vu_y + gh_x = -gb_x + F_x \tag{2.31}$$

$$v_t + uv_x + vv_y + gh_y = -gb_y + F_y \tag{2.32}$$

$$h_t + (hu)_x + (hv)_y = 0 \tag{2.33}$$

having neglected turbulence effects. The friction forces are given by the following Manning law:

$$F_x = -\frac{1}{\cos\alpha}\frac{gm^2}{h^{4/3}}u\sqrt{u^2+v^2} \tag{2.34}$$

$$F_y = -\frac{1}{\cos\alpha}\frac{gm^2}{h^{4/3}}v\sqrt{u^2+v^2} \tag{2.35}$$

where m is the Manning coefficient (usually denoted by n in the literature). We linearise these equations around a flow $(U_0(x,y,t), V_0(x,y,t), H_0(x,y,t))$, satisfying (2.31), (2.32) and (2.33) (see [161]). In general, $U_0(x,y,t)$ and $V_0(x,y,t)$ are not uniform, due to the geometry of the river. Based on the flow conditions, $(U_0(x,y,t),\ V_0(x,y,t)$ and $H_0(x,y,t))$ can be considered to be static or not. The linearised equations become:

$$u_t + U_0(x,y,t)u_x + V_0(x,y,t)u_y + gh_x = -gb_x - C_{x,y}u \tag{2.36}$$

$$v_t + U_0(x,y,t)v_x + V_0(x,y,t)v_y + gh_y = -gb_y - C_{x,y}v \tag{2.37}$$

$$h_t + U_0(x,y,t)h_x + V_0(x,y,t)h_y + H_0(x,y,t)(u_x + v_y) = 0 \tag{2.38}$$

where $C_{x,y} = \frac{1}{\cos\alpha(x,y)} \frac{gm^2}{H_0^{4/3}} \sqrt{U_0^2 + V_0^2}$ is a linearised friction coefficient.

Numerical schemes

The flows considered in the article are characterised by a Froude number such that Fr $\ll 1$. We are in the presence of a subcritical flow and since the Froude number represents the ratio between inertial and gravity forces, the inertial terms in the previous equations (uu_x, vu_y, uv_x, vv_y) can be neglected with respect to gravity terms (gh_x, gh_y) leading to the following set of equations [2, 161]:

$$u_t + gh_x = -gb_x - C_{x,y}u \tag{2.39}$$

$$v_t + gh_y = -gb_y - C_{x,y}v \tag{2.40}$$

$$h_t + H_0(x,y,t)(u_x + v_y) = 0 \tag{2.41}$$

In order to use quadratic programming techniques for data assimilation, the linearised shallow water equations should be discretised using a linear numerical scheme so that the resulting constraints equations are linear. Explicit numerical schemes require a *Courant-Friedrichs-Lewy* (CFL) number less than one to ensure stability. This in turn means that small time steps have to be used which would considerably increase the size of the constraint matrix. Therefore, implicit schemes, which are unconditionally stable, are more suited in the present case. We start by developing an implicit finite difference numerical scheme for the linearised shallow water equations (2.39), (2.40), (2.41) inspired by the Abbott scheme [1, 2]. For space steps Δx and

Δy and a time step Δt which are denoted by indices i, j and n respectively, we obtain:

$$\frac{u^{n+1}_{i,j}-u^{n}_{i,j}}{\Delta t}+g\frac{h^{n+\frac{1}{2}}_{i+1,j}-h^{n+\frac{1}{2}}_{i-1,j}}{2\Delta x}=-g\frac{b_{i+1,j}-b_{i-1,j}}{2\Delta x}-C_{i,j}u^{n}_{i,j} \tag{2.42}$$

$$\frac{h^{n+\frac{1}{2}}_{i,j}-h^{n}_{i,j}}{\frac{1}{2}\Delta t}+H_0^{i,j,n}\left(\frac{1}{2}\frac{u^{n+1}_{i+1,j}-u^{n+1}_{i-1,j}}{2\Delta x}+\frac{1}{2}\frac{u^{n}_{i+1,j}-u^{n}_{i-1,j}}{2\Delta x}+\frac{v^{n}_{i,j+1}-v^{n}_{i,j-1}}{2\Delta y}\right)=0 \tag{2.43}$$

$$\frac{v^{n+1}_{i,j}-v^{n}_{i,j}}{\Delta t}+g\left(\frac{1}{2}\frac{h^{n+1}_{i,j+1}-h^{n+1}_{i,j-1}}{2\Delta y}+\frac{1}{2}\frac{h^{n}_{i,j+1}-h^{n}_{i,j-1}}{2\Delta y}\right)=-g\frac{b_{i,j+1}-b_{i,j-1}}{2\Delta y}-C_{i,j}v^{n}_{i,j} \tag{2.44}$$

$$\frac{h^{n+1}_{i,j}-h^{n+\frac{1}{2}}_{i,j}}{\frac{1}{2}\Delta t}+H_0^{i,j,n}\left(\frac{1}{2}\frac{u^{n+1}_{i+1,j}-u^{n+1}_{i-1,j}}{2\Delta x}+\frac{1}{2}\frac{u^{n}_{i+1,j}-u^{n}_{i-1,j}}{2\Delta x}+\frac{v^{n+1}_{i,j+1}-v^{n+1}_{i,j-1}}{2\Delta y}\right)=0 \tag{2.45}$$

We can see that the first two equations (2.42), (2.43) are algorithmically connected in a first stage in such a way as to advance u directly from $n\Delta t$ to $(n+1)\Delta t$ while advancing h only from $n\Delta t$ to $(n+\frac{1}{2})\Delta t$. The second stage connects equations (2.44) and (2.45) algorithmically to advance v directly from $n\Delta t$ to $(n+1)\Delta t$ while advancing h from $(n+\frac{1}{2})\Delta t$ to $(n+1)\Delta t$. In this scheme, information is accumulated during the cycle on the h values only.

In order to implement this scheme as constraints in a quadratic program, it is more convenient to reduce the scheme to a two-level form. By subtracting equation (2.45) from equation (2.43) we obtain an expression for $h^{n+\frac{1}{2}}_{i,j}$:

$$h^{n+\frac{1}{2}}_{i,j}=\frac{h^{n}_{i,j}+h^{n+1}_{i,j}}{2}-H_0^{i,j,n}\frac{\Delta t}{8\Delta y}(v^{n}_{i,j+1}-v^{n}_{i,j-1}-v^{n+1}_{i,j+1}+v^{n+1}_{i,j-1}) \tag{2.46}$$

that can be used to eliminate $h^{n+\frac{1}{2}}_{i,j}$ from equation (2.42) to provide equation (2.47); then we add equation (2.43) to equation (2.45) to obtain equation (2.49):

$$\begin{aligned}&\frac{u^{n+1}_{i,j}-u^{n}_{i,j}}{\Delta t}+\frac{g}{4\Delta x}[(h^{n+1}_{i+1,j}-h^{n+1}_{i-1,j})+(h^{n}_{i+1,j}-h^{n}_{i-1,j})]\\&+\frac{gH_0^{i,j,n}\Delta t}{16\Delta x\Delta y}[(v^{n+1}_{i+1,j+1}-v^{n+1}_{i+1,j-1})-(v^{n}_{i+1,j+1}-v^{n}_{i+1,j-1})\\&-(v^{n+1}_{i-1,j+1}-v^{n+1}_{i-1,j-1})+(v^{n}_{i-1,j+1}-v^{n}_{i-1,j-1})]=-g\frac{(b_{i+1,j}-b_{i-1,j})}{2\Delta x}-C_{i,j}u^{n}_{i,j}\end{aligned} \tag{2.47}$$

$$\frac{v^{n+1}_{i,j}-v^{n}_{i,j}}{\Delta t}+\frac{g}{4\Delta y}[(h^{n+1}_{i,j+1}-h^{n+1}_{i,j-1})+(h^{n}_{i,j+1}-h^{n}_{i,j-1})]=-g\frac{(b_{i,j+1}-b_{i,j-1})}{2\Delta y}-C_{i,j}v^{n}_{i,j} \tag{2.48}$$

$$\begin{aligned}&\frac{h^{n+1}_{i,j}-h^{n}_{i,j}}{\Delta t}+\frac{H_0^{i,j,n}}{4}(\frac{u^{n+1}_{i+1,j}-u^{n+1}_{i-1,j}}{\Delta x}+\frac{u^{n}_{i+1,j}-u^{n}_{i-1,j}}{\Delta x}\\&+\frac{v^{n+1}_{i,j+1}-v^{n+1}_{i,j-1}}{\Delta y}+\frac{v^{n}_{i,j+1}-v^{n}_{i,j-1}}{\Delta y})=0\end{aligned} \tag{2.49}$$

One of the advantages of this scheme is that since the inertial terms were neglected, its matrix formulation is sparse thereby reducing the computational time required by the quadratic program solver to reach the solution. However, for the inverse modelling applications described in this chapter, the computational time is minimal (less 5 seconds) and the gain is hardly noticeable; since this comes at the expense of accuracy, two other schemes were formulated taking into account all the terms; the first one is a centered finite volume scheme. Note that the computational efficiency linked with the use of this particular scheme might become of use for larger problems, where runtime would be a true issue.

If we designate by $a^x_{i,j}$ and $a^y_{i,j}$ the dimensions of the rectangular cell (i,j) along the x and y axes, we obtain:

$$\begin{aligned}&\frac{u^n_{i,j} - u^{n-1}_{i,j}}{\Delta t} + \frac{1}{2a^x_i}(U_0^{i,j,n}(u^n_{i+1,j} - u^n_{i-1,j}) + g(h^n_{i+1,j} - h^n_{i-1,j}))\\ &+ \frac{V_0^{i,j,n}}{2a^y_j}(u^n_{i,j+1} - u^n_{i,j-1}) + C_{i,j}u^n_{i,j} = -\frac{g}{2a^x_i}(b_{i+1,j} - b_{i-1,j})\end{aligned} \tag{2.50}$$

$$\begin{aligned}&\frac{v^n_{i,j} - v^{n-1}_{i,j}}{\Delta t} + \frac{U_0^{i,j,n}}{2a^x_i}(v^n_{i+1,j} - v^n_{i-1,j}) + \frac{1}{2a^y_j}(V_0^{i,j,n}(v^n_{i,j+1} - v^n_{i,j-1})\\ &+ g(h^n_{i,j+1} - h^n_{i,j-1})) + C_{i,j}v^n_{i,j} = -\frac{g}{2a^y_j}(b_{i,j+1} - b_{i,j-1})\end{aligned} \tag{2.51}$$

$$\begin{aligned}&\frac{h^n_{i,j} - h^{n-1}_{i,j}}{\Delta t} + \frac{U_0^{i,j,n}}{2a^x_i}(h^n_{i+1,j} - h^n_{i-1,j}) + \frac{V_0^{i,j,n}}{2a^y_j}(h^n_{i,j+1} - h^n_{i,j-1})\\ &+ H_0^{i,j,n}(\frac{1}{2a^x_i}(u^n_{i+1,j} - u^n_{i-1,j}) + \frac{1}{2a^y_j}(v^n_{i,j+1} - v^n_{i,j-1})) = 0\end{aligned} \tag{2.52}$$

This scheme is convergent and stable (see for example [3]). Note that no assumption is made on the direction of the flow which means that this scheme can be readily implemented in situations where tidal forcing creates a flow reversal at regular intervals.

Finally, the third scheme has the added property of being more stable with respect to spatial variations of the state variables. With the same notations as above, the spatial derivative of a given variable u can be approximated by the following finite difference scheme:

$$u_x = \frac{u^{n+1}_{i+1,j} - u^{n+1}_{i-1,j} + u^n_{i+1,j} - u^n_{i-1,j}}{4a^x_{i,j}} \tag{2.53}$$

where i, j denote the spatial indices and n the time step. The implicit Euler scheme is used for the time discretisation: $\frac{\partial u_{i,j}}{\partial t}$ is approximated by $\frac{u_{i,j}^{n+1}-u_{i,j}^{n}}{\Delta t}$. The previous discretisation steps lead to the following implicit numerical scheme:

$$\begin{aligned}&\frac{u_{i+1,j+1}^{n+1}-u_{i+1,j+1}^{n}}{\Delta t}+\frac{U_0^{i,j,n}}{4a_{i,j}^{x}}(u_{i+1,j}^{n+1}-u_{i-1,j}^{n+1}+u_{i+1,j}^{n}-u_{i-1,j}^{n})\\&+\frac{V_0^{i,j,n}}{4a_{i,j}^{y}}(u_{i,j+1}^{n+1}-u_{i,j-1}^{n+1}+u_{i,j+1}^{n}-u_{i,j-1}^{n})\\&+\frac{g}{4a_{i,j}^{x}}(h_{i+1,j}^{n+1}-h_{i-1,j}^{n+1}+h_{i+1,j}^{n}-h_{i-1,j}^{n}+2(b_{i+1,j}-b_{i-1,j}))+C_{i,j}u_{i,j}^{n}=0\end{aligned} \tag{2.54}$$

$$\begin{aligned}&\frac{v_{i+1,j+1}^{n+1}-v_{i+1,j+1}^{n}}{\Delta t}+\frac{U_0^{i,j,n}}{4a_{i,j}^{x}}(v_{i+1,j}^{n+1}-v_{i-1,j}^{n+1}+v_{i+1,j}^{n}-v_{i-1,j}^{n})\\&+\frac{V_0^{i,j,n}}{4a_{i,j}^{y}}(v_{i,j+1}^{n+1}-v_{i,j-1}^{n+1}+v_{i,j+1}^{n}-v_{i,j-1}^{n})\\&+\frac{g}{4a_{i,j}^{y}}(h_{i,j+1}^{n+1}-h_{i,j-1}^{n+1}+h_{i,j+1}^{n}-h_{i,j-1}^{n}+2(b_{i,j+1}-b_{i,j-1}))+C_{i,j}v_{i,j}^{n}=0\end{aligned} \tag{2.55}$$

$$\begin{aligned}&\frac{h_{i+1,j+1}^{n+1}-h_{i+1,j+1}^{n}}{\Delta t}+\frac{U_0^{i,j,n}}{4a_{i,j}^{x}}(h_{i+1,j}^{n+1}-h_{i-1,j}^{n+1}+h_{i+1,j}^{n}-h_{i-1,j}^{n})\\&+\frac{V_0^{i,j,n}}{4a_{i,j}^{y}}(h_{i,j+1}^{n+1}-h_{i,j-1}^{n+1}+h_{i,j+1}^{n}-h_{i,j-1}^{n})\\&+\frac{H_0^{i,j,n}}{4a_{i,j}^{x}}(u_{i+1,j}^{n+1}-u_{i-1,j}^{n+1}+u_{i+1,j}^{n}-u_{i-1,j}^{n})\\&+\frac{H_0^{i,j,n}}{4a_{i,j}^{y}}(v_{i,j+1}^{n+1}-v_{i,j-1}^{n+1}+v_{i,j+1}^{n}-v_{i,j-1}^{n})=0\end{aligned} \tag{2.56}$$

This scheme will be used as constraints for the quadratic programs solved in this chapter.

Incorporation of Lagrangian measurements

One issue that comes up when attempting to employ quadratic programming techniques to solve this problem is that the interpolation operator H_n in equation (2.16) is usually nonlinear. However, as mentioned before, in the present case we are able to use a linear operator. Indeed, we assume that the Lagrangian sensors (drifters) have a sufficiently small time sampling period compared to the Lagrangian timescale of the body of water considered (which is realistic for the scenarios presented in this thesis); therefore, the Eulerian velocity can be approximated by the Lagrangian velocity obtained by finite differences [81, 117]:

$$u_D(t_n) = \frac{x_D(t_n) - x_D(t_{n-1})}{\Delta t} \tag{2.57}$$

$$v_D(t_n) = \frac{y_D(t_n) - y_D(t_{n-1})}{\Delta t} \tag{2.58}$$

where $x_D(t_n)$ and $y_D(t_n)$ are the positions of drifter D at time t_n and $u_D(t_n)$, $v_D(t_n)$ represent the velocity of the drifter at time t_n at location $(x_D(t_n), y_D(t_n))$. The observed variables are equal to the state variables and $h_n^o = \mathbb{I}$ and $H_n = h^I$. For the interpolation operator, which maps measurement points to grid points, we choose a bilinear interpolation. Note that if a finite volume scheme is used, the state variables are constant for each cell and therefore the observed variables can simply be matched to the state variables by looking in which cell the drifter is at a given time instant. We also assume the error covariance matrices $\mathbf{B} = \mathbb{I}$ and $\mathbf{R}_{\text{n}} = \mathbf{R}_{\text{n}} = \frac{1}{r}\mathbb{I}$ where r is an empirically determined scalar. The factor r is a weighting parameter to adjust the respective influences of the background and observation terms. For higher values of r, the analysed state will be closer to the background term than the observations, whereas for lower values the regularising effect of the background term is reduced, which in effect slows down the convergence of the algorithm. A careful choice of the parameter r is therefore essential to a good performance of the data assimilation algorithm.

Optimisation Program Formulation

We can now pose the data assimilation problem as a quadratic optimisation problem. The background error covariance matrix $\mathbf{B}$ can be chosen so that the cost function (2.16) is quadratic positive definite. The discretised dynamics of the flow are now encoded in the form of linear constraints, which is one of the benefits of the quadratic method used. While the numerical scheme is implicit, (i.e. has to be written in the form $EX_{n+1} = AX_n + Bu_n$ where E is not necessarily invertible), the quadratic program can incorporate these constraints at no further cost or complication. We concatenate the vectors $(u_{i,j}^n, v_{i,j}^n, h_{i,j}^n)$ (i.e. the vector X_n) for all (i, j, n) into a single vector called X, and abbreviate the dynamics constraints by $AX = b$, where this equation encodes the discretised flow equations. The search space X_0 from which all other quantities depend is allowed to evolve in a set of feasible initial conditions dictated by flow physics.

Because the $(u^n_{i,j}, v^n_{i,j}, h^n_{i,j})$ cannot take arbitrary values, the space in which X evolves can also be restricted to increase the speed of convergence of the method. These two constraints are encoded in the form of an inequality, $GX \leqslant h$. Finally, using the previous variable definitions, and because of the background term, the optimisation program including equation (2.16) as cost functional, can be written as

$$\begin{aligned} \text{minimise} \quad & J(X_0) = \tfrac{1}{2}X^T P X + q^T X + r \\ \text{subject to} \quad & GX \leqslant h \\ & AX = b \end{aligned} \tag{2.59}$$

2.3.4 Newtonian relaxation

Newtonian relaxation (also called nudging method), was introduced by Anthes in 1974 in [11], and is a data assimilation technique which, unlike the other methods presented in this chapter, does not involve any variational or sequential problem. It simply consists in adding to the model equations a source term which is proportional to the difference between the observed variables and the state variables. The effect of this source term is to correct the model evolution and nudge it closer to the observations. In other words, with this method the system evolves according to the model when no observations are available and is corrected towards the observations when they exist. If we assume that observations are collected over a period $[0, T]$ at time instants t_n for $0 \leqslant n \leqslant N$ and considering a given model of the type:

$$\frac{dX}{dt} = M(X) \tag{2.60}$$

the nudging method would consist in adding a source term to the model equations of the form [11, 13, 81, 127]:

$$\frac{dX}{dt} = M(X) - \sum_{n=0}^{N} \lambda(Y^o_n - H_n[X_n]) \tag{2.61}$$

where Y^o_n is the observation vector. The function λ is called the nudging factor and will determine how much weight is given to the observations with respect to the model, as well as the space and time scale over which observation data are discarded. This term is therefore crucial to the performance of the nudging algorithm and different types of nudging factors have been developed [74, 111, 127]. If we choose a nudging factor similar to the one used in [74, 81, 111], we obtain for $t \leqslant t_{\text{init}}$:

$$\lambda(x,y,t) = \begin{cases} \frac{1}{T_a}\exp(-\frac{x^2+y^2}{R^2_{\text{nudge}}})\exp(-\frac{t-t_{\text{init}}}{T_d}), & \sqrt{x^2+y^2} \leqslant 4R_{\text{nudge}}; \\ 0, & \sqrt{x^2+y^2} > 4R_{\text{nudge}}. \end{cases} \tag{2.62}$$

T_a is the timescale and will determine the strength of the nudging term. For larger values of T_a, the equation will be almost identical to the original model equations and little benefit will be gained from the observations; similarly, for lower values of T_a, the solution obtained will not closely follow the model equations. T_d is the dumping timescale which determines the amount of time after which observations are considered obsolete and discarded. R_{nudge} is the nudging radius and represents the spatial domain of influence of observations collected at a given point. Observation data will be discarded for points that are too far away from observations. In the case of the 2-dimensional nonlinear shallow water equations (2.11), (2.12), (2.13), the nudging method is given by the equations (see [145]):

$$u_t + uu_x + vu_y + gh_x = -gb_x + f_x - \sum_{n=0}^{N} \lambda_1(u^o_n - H_n[u_n]) \tag{2.63}$$

$$v_t + uv_x + vv_y + gh_y = -gb_y + f_y - \sum_{n=0}^{N} \lambda_2(v^o_n - H_n[v_n]) \tag{2.64}$$

$$h_t + (hu)_x + (hv)_y = -\sum_{n=0}^{N} \lambda_3(h^o_n - H_n[h_n]) \tag{2.65}$$

where λ_1, λ_2, λ_3 are the respective nudging factors for each equation, u_n, v_n, h_n the modelled variables at time t_n and u^o_n, v^o_n, h^o_n the observed velocities and height at time t_n. The nudging factors may not be the same for each equation as observations for different state variables may be relevant for different time and space scales. The main advantage of the nudging method when compared to other data assimilation algorithms is its great simplicity, as it does not involve minimising a cost function but only adding a source term to the numerical code used to solve the model equations. Therefore, this method does not require any knowledge of optimisation techniques and can be implemented easily from an existing simulation code. However, due to its nature, the nudging method is very sensitive to the number of drifters, geographical coverage and measurement accuracy.

2.3.5 Ensemble Kalman filtering

Among the family of sequential methods is the *Ensemble Kalman Filter* (EnKF) [45, 46, 47], first developed by Evensen in oceanography. The EnKF is a Monte-Carlo implementation of the Bayesian update problem: given a probability distribution of the modelled system (the prior, called forecast in geosciences) and data likelihood, the Bayes theorem is used to to obtain the probability distribution with the data likelihood taken into account (the posterior or analysis). The Bayesian update is combined with advancing the model in time, with the data being incorporated from time to time. The original Kalman Filter [88] relies on the assumption that the probability distributions are Gaussian, and provides algebraic formulae for the change of the mean and covariance by the Bayesian update, and a formula for advancing the covariance matrix in time provided the system is linear. However, this might not be computationally tractable for high dimensional systems, which the main motivation for the development of the EnKF [45, 76]. The EnKF represents the distribution of the system state using a random sample, called an ensemble, and replace the covariance matrix by the sample covariance of the ensemble. One advantage of the EnKF is that advancing the probability distribution in time is achieved by simply advancing each member of the ensemble. Note that the EnKF still relies on the Gaussian assumption, though it is used in practice for nonlinear problems, where the Gaussian assumption is not satisfied. In our case, the EnKF was applied to state estimation of river flows by J. Percelay and O. P. Tossavainen [148, 157].

Notations

The following notations are used for the EnKF work presented later:

θ_n : Vector of state variables, namely the velocity components (u, v), the water height h for each mesh point and the positions of the drifters x_D, y_D at a time instant t_n.

F_n : Time dependent discretised forward 2D shallow water equation model.

w_n : State noise representing the error between the forward model and the true state.

Q_n : Covariance of the state noise w_n.

y_n : Vector of observed variables, namely the positions of the drifters (x_D, y_D) at a time instant t_n.

ϵ_n : Measurement noise, representing the error in the GPS measurements of the drifter positions.

R_n : Covariance matrix of the measurement noise ϵ_n.

C_n : Observation matrix.

$\xi^p_{n|n}$: p^{th} ensemble member at time t_n.

K_n : Kalman gain at time t_n.

We employ a state-augmentation approach that has been earlier introduced in oceanography (see [79, 94, 135] for more details). We denote by θ_F the state variables corresponding to (u, v, h) on the computational domain. In this approach, Lagrangian drifter positions are embedded in the state vector in discretised form and the new state vector θ is written as

$$\theta = \begin{pmatrix} \theta_F \\ \theta_D \end{pmatrix}.$$

This approach enables us to take the Lagrangian nature of observations into account without deriving Eulerian quantities from Lagrangian measurements as was done previously in equations (2.57), (2.58).

By inspection, we note that there is a one-way coupled system from flow variables θ_F to drifter positions θ_D, that is, the drifter positions can be solved after the flow variables have been solved. We express this system by F_n, where n refers to the time step. Furthermore, due to the uncertainties in the modelling, we add a stochastic term w_n to represent modelling errors. The noise process w_n, with covariance Q_n, is a Gaussian noise process that is used to model inaccuracies in the evolution model (discretisation error, poorly known inputs etc., see for example [141]). Thus, we can write a nonlinear discrete stochastic state space model of the form:

$$\theta_{n+1} = F_n(\theta_n) + w_n. \tag{2.66}$$

Here, θ_n is the system state at time step n and $F_n(\theta_n)$ is one time step in the (non-linear) discretised shallow water and drifter model, which is time dependent. Note that if the equations were linear, X_n used in the previous section would be the portion of θ_n which does not contain any drifter position (labeled as θ_D). For the observations, we use an additive noise model

$$y_n = C_n \theta_n + \epsilon_n. \tag{2.67}$$

The observation vector is y_n and the observation model that relates the state variables to the measurements is C_n. The noise process ϵ_n is the measurement noise with covariance R_n. Without loss of generality, noise processes are assumed to have zero mean. Furthermore, with these choices, the observation model C_n simply becomes $C_n = (\mathbf{0}\,\mathbb{I})$ where **0** is a zero matrix of size three times $M \times N$ for the three variables (u, v and h) and $\mathbb{I}$ is the identity operator of size the number of drifters times two. The numerical solution of the 2D shallow-water equations and drifter positions in (2.66) is computed using the commercial hydrodynamic software TELEMAC 2D [72]. TELEMAC uses a streamline upwind Petrov-Galerkin based finite element solver for hydrodynamic equations. Therefore, in practice, the F_n step of equation (2.66) is performed numerically by the software.

In the filtering problem, the aim is to compute conditional expectations

$$\theta_{n|n} = E(\theta_n | y_n, \ldots, y_1).$$

In the case of linear observation and evolution equations and for Gaussian noise processes, the recursive Kalman filter algorithm can be used to determine the estimates of conditional expectation $\theta_{n|n}$ and covariance $\Gamma_{n|n}$. If the evolution and/or observation equations are nonlinear and differentiable, the extended Kalman filter can be applied [8]. However, to avoid the cumbersome linearisation of a nonlinear finite difference scheme and to preserve the higher order statistics, which may be lost in the linearisation, we employ the EnKF.

For the state space model (2.66)–(2.67), the EnKF algorithm can be summarised as in [47, 69]:

1. Initialisation: An ensemble of N_{states} states $\xi_0^{(p)}$ indexed by p are generated to represent the uncertainty in θ_0.

2. Time update:

$$\xi^{(p)}_{n|n-1} = F_n(\xi^{(p)}_{n-1|n-1}) + w^{(p)}_{n-1} \tag{2.68}$$

$$\theta_{n|n-1} = \frac{1}{N_{\text{states}}} \sum_{p=1}^{N_{\text{states}}} \xi^{(p)}_{n|n-1} \tag{2.69}$$

$$E_{n|n-1} = [\xi^{(1)}_{n|n-1} - \theta_{n|n-1}, \ldots, \xi^{(N_{\text{states}})}_{n|n-1} - \theta_{n|n-1}] \tag{2.70}$$

3. Measurement update:

$$\Gamma_{n|n-1} = \frac{1}{N_{\text{states}} - 1} E_{n|n-1} E^{\mathrm{T}}_{n|n-1} \tag{2.71}$$

$$K_n = \Gamma_{n|n-1} C^{\mathrm{T}}_n [C_n \Gamma_{n|n-1} C^{\mathrm{T}}_n + R_n]^{-1} \tag{2.72}$$

$$\xi^{(p)}_{n|n} = \xi^{(i)}_{n|n-1} + K_n[y_n - C_n \xi^{(p)}_{n|n-1} + \epsilon^{(p)}_n] \tag{2.73}$$

where the ensemble of state vectors are generated with the realisations $w^{(p)}_n$ and $\epsilon^{(p)}_n$ of the noise processes w_n and ϵ_n, respectively. In the previous equations, an important step is that at measurement times, each measurement is represented by an ensemble. This ensemble has the actual measurement y_n as mean and the variance of the ensemble is used to represent measurement errors. This is done by adding perturbations $\epsilon^{(p)}_n$ to measurements drawn from a distribution with zero mean and covariance equal to measurement error covariance matrix R_n. This ensures that the variance of the updated ensemble is not too low [28].

EnKF implementation

The initial ensemble for the EnKF is generated from historical knowledge on the boundary conditions in the river. We generate an ensemble of size $N_{\text{states}} = 100$ as follows. The idea is to produce N_{states} different initial velocity fields representing the different states of the river. In order to do this, we use data from the DSM2 model [9], which is available for a period of eight years. From this data, we extract statistical characteristics of the boundary conditions: a mean value of discharge upstream, a mean value of surface elevation downstream and a covariance matrix of those two quantities. From this, we generate a set of discharges and free surface elevations ($q^{(p)}$,$\eta^{(p)}$), $p = 1, \ldots, N_{\text{states}}$ that correspond to the statistical characteristics deduced from the historical data. For each discharge $q^{(p)}$, we deduce a velocity profile

$$(u^{(p)}, v^{(p)})\big|_{\partial\Omega_{\mathrm{up}}},$$

that is normal to the upstream boundary and proportional to the square root of the water depth at the upstream boundary. The downstream boundary condition is the surface elevation obtained directly from the historical data statistics:

$$\eta^{(p)}\big|_{\partial\Omega_{\mathrm{down}}} = \eta^{(p)}.$$

In order to obtain realistic initial states for the EnKF algorithm, a 2.5 hour stabilisation run is performed as follows. The forward solver TELEMAC 2D [72] is started from zero velocity and it runs for 2.5 hours for each ensemble member using the set $(u^{(p)}, v^{(p)})\big|_{\partial\Omega_{\mathrm{up}}}, \eta^{(p)}$ as boundary conditions. By doing this, numerically stable and physically meaningful initial states for flow variables $(u_0^{(p)}, v_0^{(p)}, h_0^{(p)})$ are obtained. Furthermore, the zero velocity initial condition that we use to instantiate TELEMAC 2D is not affecting the flow variables when the state estimation starts. When stable states are reached, state estimation can start. In the state estimation procedure we do not update the velocity at the nodes that correspond to the land boundary. This is to make sure that we do not violate the no-slip condition. The initial boundary conditions for each ensemble member at time step zero are $(u^{(p)}, v^{(p)})\big|_{\partial\Omega_{\mathrm{up}}}, \eta^{(p)}$; these are obtained from the last time step of the stabilising run. We set the standard deviation of the measurement noise ϵ_n to be 0.5 m in both x and y directions. This indicates our confidence in the measurement data, and can be seen as a realistic choice for the accuracy of GPS receivers. The state noise models the discrepancy between the model and reality. In this work it is assumed that the state noise for the flow variables can deduced from the properties of large number of ensemble members. The covariance matrix Q_n for the flow variables is computed as the covariance of the initial ensemble members around the mean flow. Thus, the state noise reflects the variations in the velocity and depth field due to the uncertain and changing boundary conditions. We further assume that the mesh nodes corresponding to the land nodes are not affected by the state noise in order not to violate the no-slip condition on the land boundaries. In addition, we assume that there is a small uncertainty in the model which predicts the drifter positions. This uncertainty is modelled by a white noise, that is the covariance matrix for the drifters is a diagonal matrix. This uncertainty is indicated by setting the standard deviation of the state noise in the covariance matrix Q_n corresponding to all drifter coordinates to be 0.5 m in both x and y directions.

2.4 Twin experiments

2.4.1 The Sacramento-San Joaquin Delta

The algorithms described earlier can obviously be applied to any river. Access to data such as an accurate bathymetry and boundary conditions is required to perform numerical simulations (parameters are needed in the constitutive equations). The present work focuses on the implementation of these algorithms on a body of water located in the Sacramento-San Joaquin Delta (Figure 2.1); the geographical proximity of this area was also helpful when it came to conducting field experiments such as releasing drifters or installing sensors to measure boundary conditions. The Delta is a web of channels and reclaimed islands formed by the confluence of the Sacramento River, itself fed by the northern Sierra Nevada runoff, and the San Joaquin River, which mingle with smaller tributaries to create a 700 mile labyrinth of sloughs and waterways surrounding 57 reclaimed islands [107, 112]. The Delta is of great importance to the state of California for a number of reasons. It is a source of drinking water for more than 20 million Californians and most of California's farmland relies on water tributary to the Delta for irrigation.

Additionally, the Delta is a hub for California's infrastructure (Figure 2.2). The state's major north-south highway (I-5), several rail lines and deep water shipping channels leading to the Sacramento and Stockton ports all pass through the Delta as do electricity and gas transmission lines, as well as aqueducts and canals conveying water to a number of local water utilities. The Delta also provides habitat for numerous California fish and wildlife species to live in or migrate through as well as land for agriculture and more recently for urban development. In recent years, there has been a general agreement that challenges currently affecting the Delta could progress into a crisis if no action is taken. Perhaps the most striking area is that of levees; for the past two centuries, more than 1,000 miles of levees have been built to protect reclaimed Delta islands (Figure

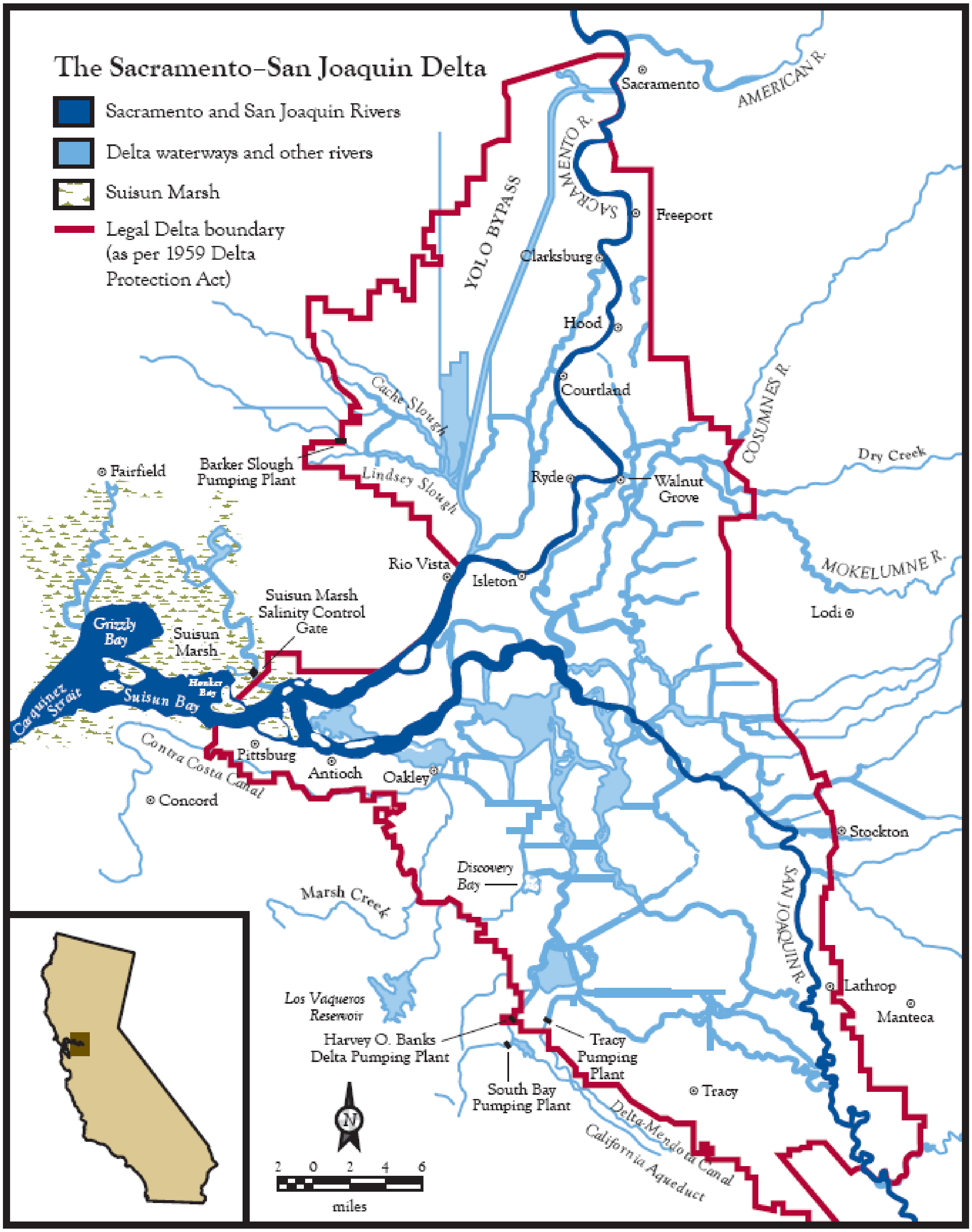

Figure 2.1: The Sacramento-San Joaquin Delta [107].

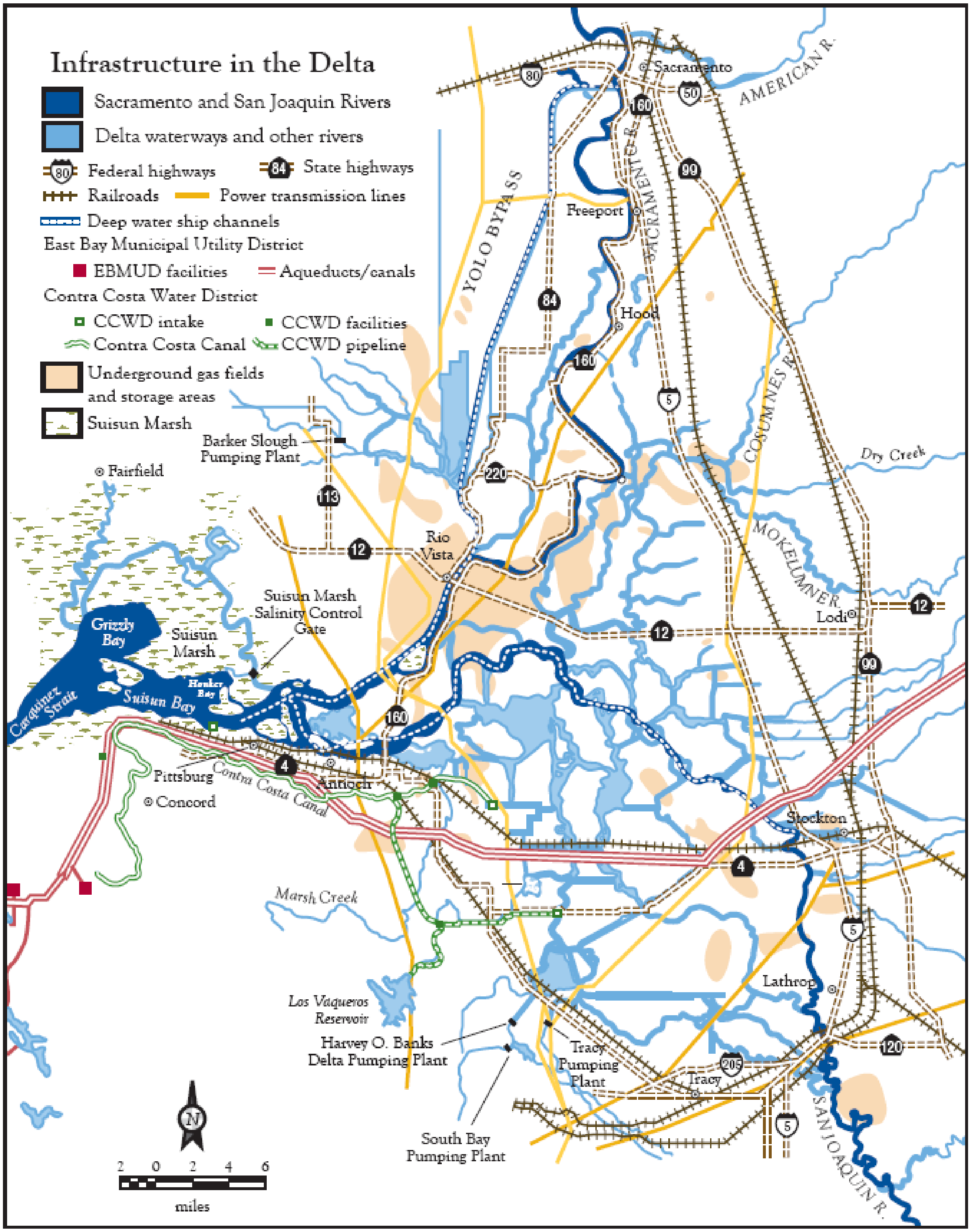

Figure 2.2: Infrastructure in the Delta [107].

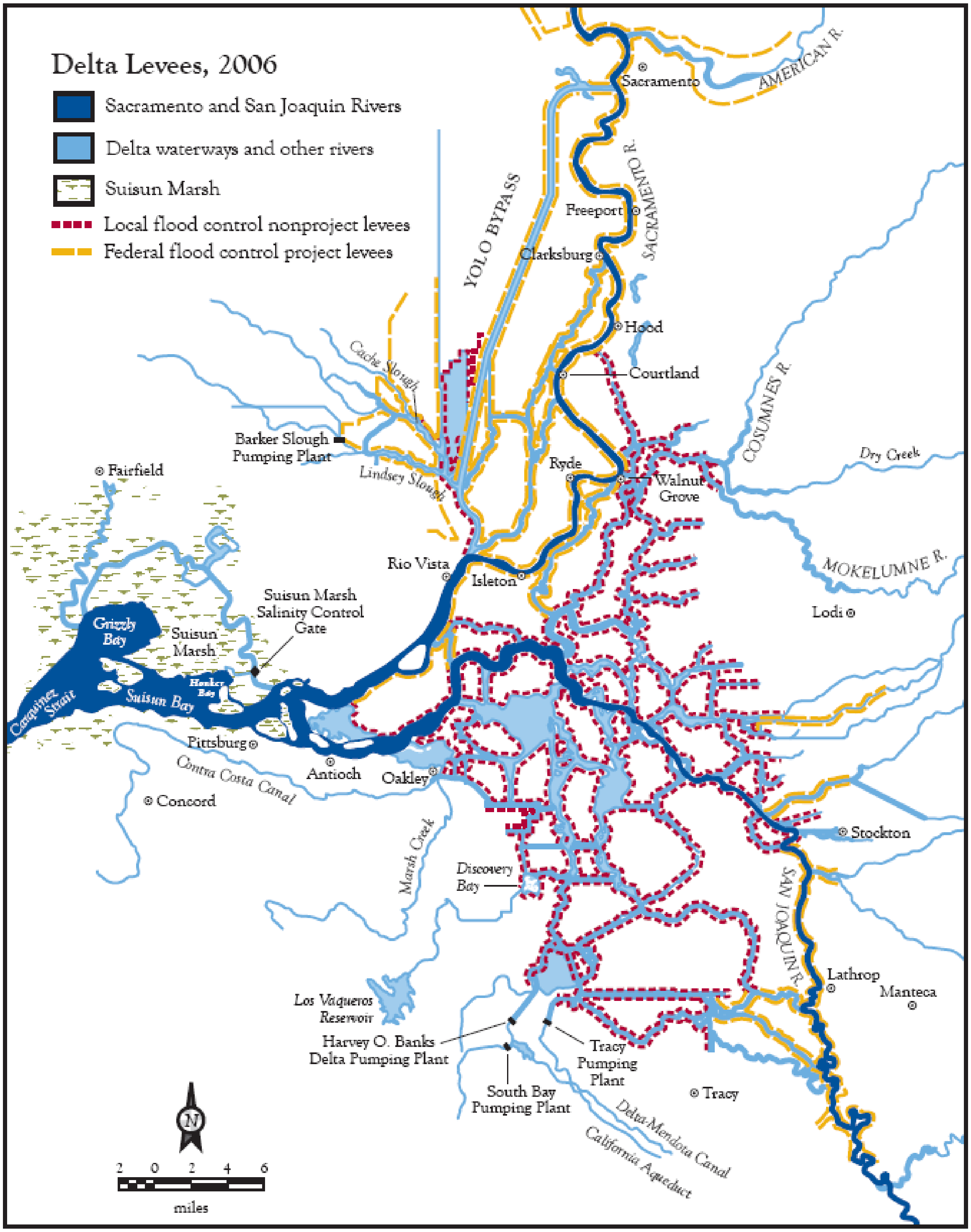

Figure 2.3: Delta levees [107].

2.3), many of which are 25 feet or more below sea level, by preventing flooding and thus allowing cultivation of the soil. A sound and well-maintained levee systems is vital to protect the farms and towns on Delta islands as well as the freshwater supply moving through the Delta waterways. Indeed, when levees fail, water which tends to be salty since it is drawn upstream from the San Francisco Bay, rushes into the islands located below sea level. Since 1980, 18 Delta islands have been partially or completely flooded and numerous studies have found that Delta levees are deteriorating, their maintenance costing millions of dollars. The numerous earthquake fault lines running through or near the Delta also pose a threat to levee stability. Water officials fear that a major earthquake could cause the levees to fail and flood numerous Delta islands, forcing Delta water users throughout California to rely on stored supplies and disrupting water delivery to central and southern California. The Delta is a major source of drinking water for California's fast growing population (predicted to reach 48 million by 2020), which will increase the amount of freshwater pumped from the Delta. Water salinity is another major concern, as it affects directly the potability of drinking water supplies, the productivity of farmland and the viability of organisms in the aquatic ecosystems. Salty ocean water typically creeps up the Delta channels during dry periods when freshwater flow is reduced. The intrusion of saltwater into the Delta forces the shutdown of pumps which provide drinking water to the state residents, thus requiring the purchase of water from other sources at a much higher cost. Finally, another issue is the impact of growing population and water consumption of the Delta's ecosystems. In the fall of 2004, routine fish surveys registered sharp declines in several pelagic species such as striped bass, green and white sturgeon, perch, Chinook salmon, threadfin shad and delta smelt [112]. This trend was confirmed in subsequent surveys and the delta smelt (*Hypomesus transpacificus*), often seen as an indicator of ecosystem health in the Delta, may be facing extinction in the near future unless this course is reversed. Invasive species have also come to pose extensive challenges to many of the native species present in the Delta. These mounting challenges resulted in the creation in 1995 as part of the Delta Accord, of a state-federal program named CALFED (*California Water Policy Council and Federal Ecosystem Directorate*) to find new solutions to the Deltas ecosystem and water supply issues. A part of this effort consists in developing modelling and simulation tools and the work presented in this and the next chapter was partially funded by a CALFED grant.

2.4.2 Comparison of the data assimilation algorithms

A standard benchmark for the validation of data assimilation algorithms is the twin experiment which consists in comparing the true state of the system with the assimilated state. For this, a simulation is run from time t_0 to time T using a two dimensional nonlinear shallow water model, yielding the so-called true state at every time instant between t_0 and T. At a chosen time $t_0 \leqslant t_1 \leqslant T$, drifters are released in simulation from the upstream end, and their trajectories are simulated using a Runge-Kutta method and the vector field provided by the nonlinear shallow water simulation; then, a data assimilation process is started and the assimilated state is generated by the QP algorithm or the EnKF between t_1 and T and compared with the true state at the same instant.

To compare the assimilated state and the true state, the relative *root mean square* (RMS) error is a commonly used metric in the field of data assimilation. The relative RMS error for the assimilated state is defined by:

$$E^a(t) = \left(\frac{\int_x \int_y |X^t(x,y,t) - X^a(x,y,t)|^2 dxdy}{\int_x \int_y |X^t(x,y,t)|^2 dxdy} \right)^{\frac{1}{2}} \tag{2.74}$$

where X^t is the true state and X^a the assimilated state.

Several experiments are performed in order to test the influence of the number of drifters and their initial position in the river on the accuracy of the assimilation. The domain is a stretch of the Sacramento River, located between the Delta Cross Channel and the Georgiana Slough in the Sacramento-San Joaquin Delta. The bathymetry is provided by the *United States Geological Survey* (USGS) as shown in Figure 2.4. A typical velocity field generated by a two dimensional nonlinear shallow water simulation is represented in Figure 2.5.

The boundary conditions used for the simulation are computed using the *Delta Simulation Model II* (DSM2) [9], a one dimensional shallow water model of the San Francisco Bay and Sacramento-San Joaquin Delta which provides hourly discharge and surface elevation at various locations. The bottom friction is modelled using Manning's law with a Manning coefficient chosen as constant in time and space and equal to 0.02,corresponding to a straight gravel bottom [36]. The simulation runs for two and a half hours before

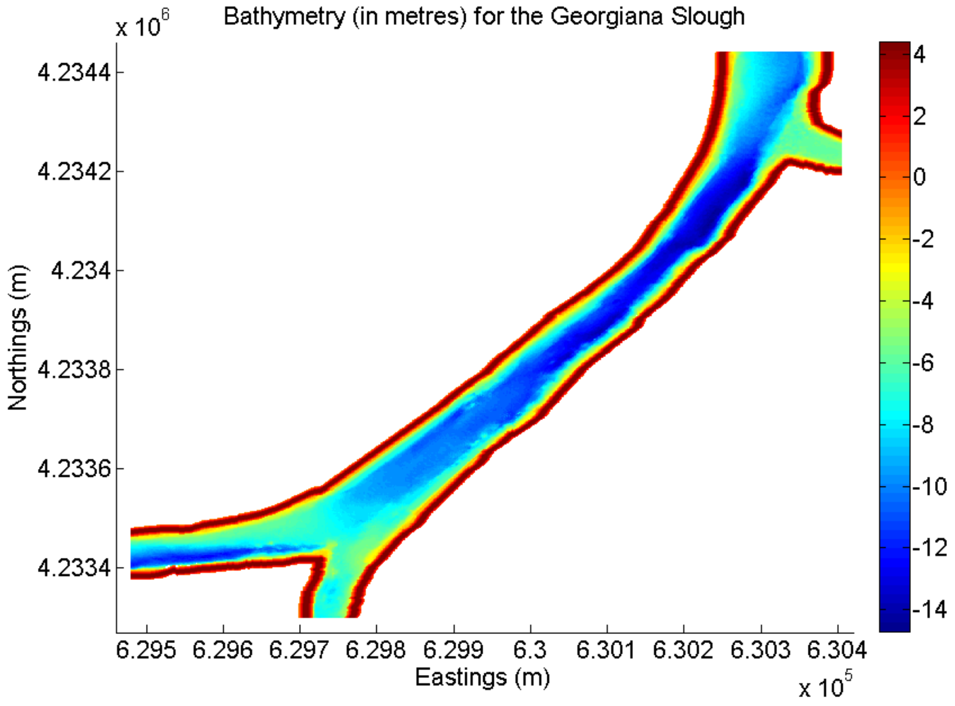

Figure 2.4: Bathymetry in the Sacramento River in metres.

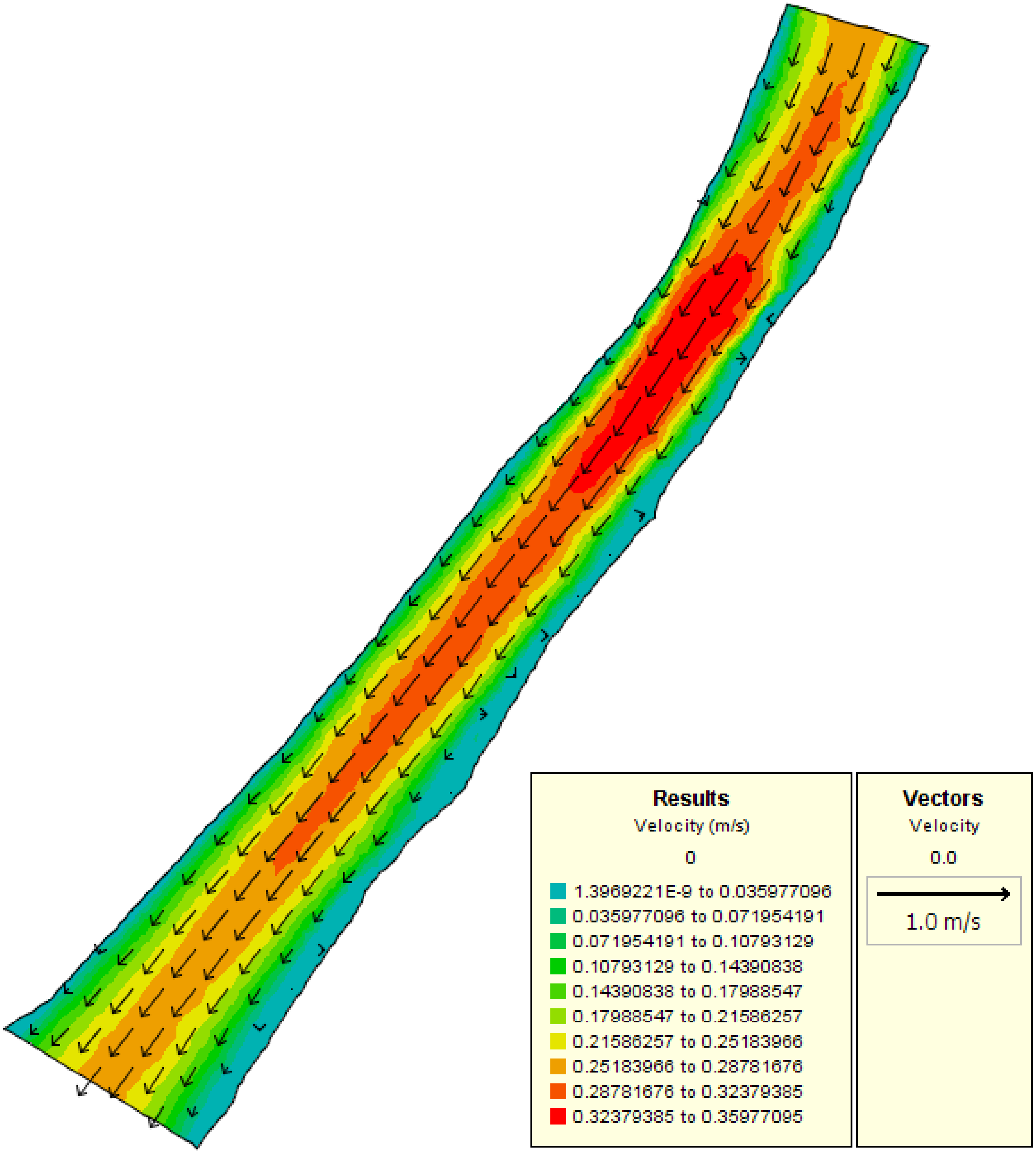

Figure 2.5: Velocity field in the Sacramento River in metres per second obtained from the forward simulation. The background colour represents the magnitude of the velocity. This velocity field represents the true state in the twin experiments.

the drifters are released in order to obtain a stable state. Three different test cases are considered. First is a situation with constant boundary conditions. The second and third cases are situations with varying boundary conditions extracted from DSM2 results, corresponding to high and low flow situations in March and June 2006 respectively. Different studies can be conducted using these simulations. First, we try to determine the influence of the number of drifters by using four, eight or twelve drifters in the assimilation algorithms. Then, we study the influence of measurements frequency by using only a measurement every minute instead of every 30 seconds.

The comparison between the two methods is made a posteriori: the quadratic program runs on a period of one hour and recovers the state of the system since the beginning of this period, when the drifters were released. Additionally, the quadratic programming based algorithm generates the state of the system at every time step over the period of assimilation, as part of the state X reconstructed. The evolution of the relative RMS error is computed for both the algorithms (quadratic programming based and EnKF) and the respective errors are compared for each twin experiment. The quadratic program is solved using a large scale quadratic programming software (in our case CPLEX [80]) with the algorithm itself being coded through the AMPL [49] modelling language. For a one hour period, the problem has about 100,000 variables and some 20 iterations of a barrier method are necessary for the solver to converge to a solution taking at most five seconds on a desktop computer. A background term, necessary for the quadratic programming algorithm, is generated from the available information (in our case, the upstream boundary conditions). The EnKF algorithm generates an estimate of the state vector at each time step. For the EnKF, the assimilation takes between six to eight hours.

Results and comparison

We begin by using constant boundary conditions chosen from DSM2 values. The discharge upstream is $Q_{\text{up}} = 235.3$ m^3/s and the surface elevation downstream is $\eta_{\text{down}} = 1.49$ m. This corresponds to normal conditions in the river. For low flow boundary conditions, we chose a one hour period on June 29th 2006 from 1 a.m. to 2 a.m. The discharge upstream varies from 300.2 m^3/s at 1 a.m. to 223.1 m^3/s at 2

Time		t_0		t_0+30min		t_0+60min		Mean	
N_d	M.S.	QP	EnKF	QP	EnKF	QP	EnKF	QP	EnKF
Constant Flow									
4	30s	19.8%	7.4%	20.0%	20.1%	20.0%	15.7%	20.0%	18.0%
	60s	19.8%	16.4%	20.0%	28.0%	20.0%	13.4%	20.0%	20.0%
8	30s	8.0%	7.0%	9.6%	9.0%	9.6%	9.8%	9.4%	8.7%
	60s	8.0%	18.1%	9.6%	15.3%	9.6%	14.7%	9.4%	12.1%
Low Flow									
4	30s	19.8%	6.6%	20.6%	13.4%	20.8%	20.5%	20.5%	15.3%
	60s	20.1%	N/A	20.2%	N/A	20.2%	N/A	20.2%	N/A
8	30s	4.8%	7.2%	4.9%	8.2%	5.8%	10.9%	5.1%	9.2%
	60s	5.0%	8.6%	5.1%	11.3%	5.8%	12.2%	5.2%	12.1%
High Flow									
4	30s	20.6%	7.9%	20.4%	12.8%	20.5%	10.5%	20.5%	11.2%
	60s	25.9%	8.1%	25.7%	12.9%	25.9%	12.2%	25.8%	11.6%
8	30s	7.5%	8.1%	6.8%	8.4%	7.0%	12.2%	7.0%	9.1%
	60s	19.8%	11.1%	19.5%	8.5%	19.7%	14.6%	19.6%	9.6%

Table 2.1: Relative RMS error as computed in equation (2.74) for the different test cases. N_d is the number of drifters and M.S. the measurement sampling time. For each case, the errors are reported for the initial time (t_0), after half of the total assimilation time (t_0 + 30 minutes) and at the end of the experiment (t_0 + 60 minutes). The mean error over the time period is also included.

a.m. The surface elevation downstream varies from 1.52 m at 1 a.m. to 1.58 m at 2 a.m. The discharge here is lower as the snow melt period is over, the variation of the discharge resulting from tidal forcing. The high flow boundary conditions correspond to a one hour period on March 17th 2006 from 4 p.m. to 5 p.m. The discharge upstream varies from 722.3 m^3/s at 4 p.m. to 686.1 m^3/s at 5 p.m. The surface elevation downstream varies from 2.91 m at 4 p.m. to 2.95 m at 5 p.m. These high flow conditions are typical of the snow melt season in the Delta. For each of the scenarios described previously, either four or eight drifters are released. The results for the different test cases are summarised in table 2.1.

Table 2.1 shows that the evolution of the error in time is different for both methods. The error is relatively stable in the case of the quadratic programming algorithm while it varies significantly for the EnKF. The reason of this variability in the EnKF is that the model used for the state noise may not be optimal. Finding an optimal model for state noise is currently an open problem. The error is usually highest near the land boundary for both methods due to the absence of drifters in this area.

For the EnKF, there is a small decrease in relative RMS error when the number of drifters is increased from four to eight drifters, except in the case of constant boundary conditions; this is explained by

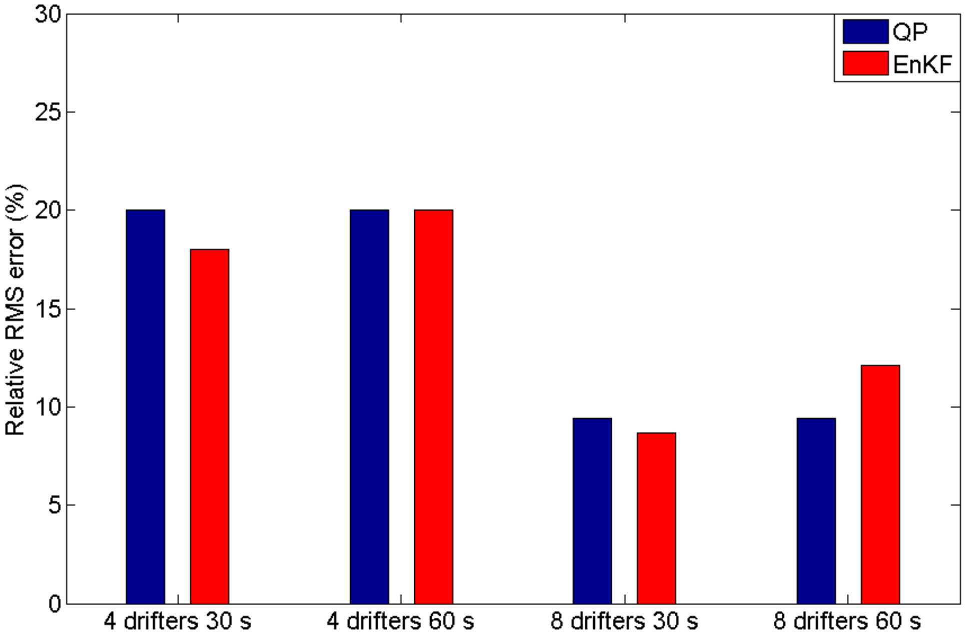

Figure 2.6: Relative RMS error (%) for both methods in the case of constant boundary conditions for four and eight drifters and 30s and 60s time sampling.

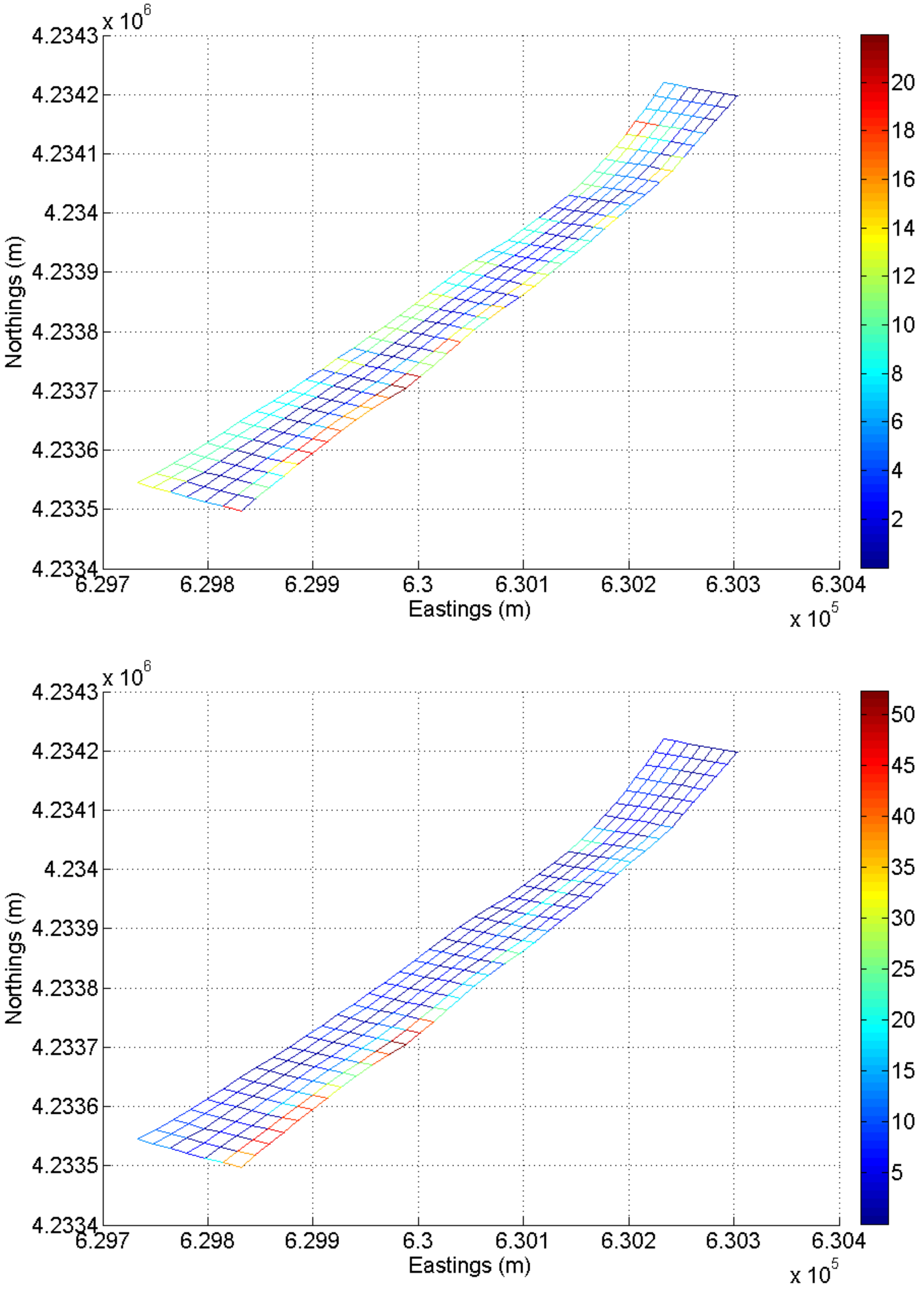

Figure 2.7: Relative RMS error (%) for the QP algorithm in the case of constant boundary conditions with eight drifters (top) and four drifters (bottom).

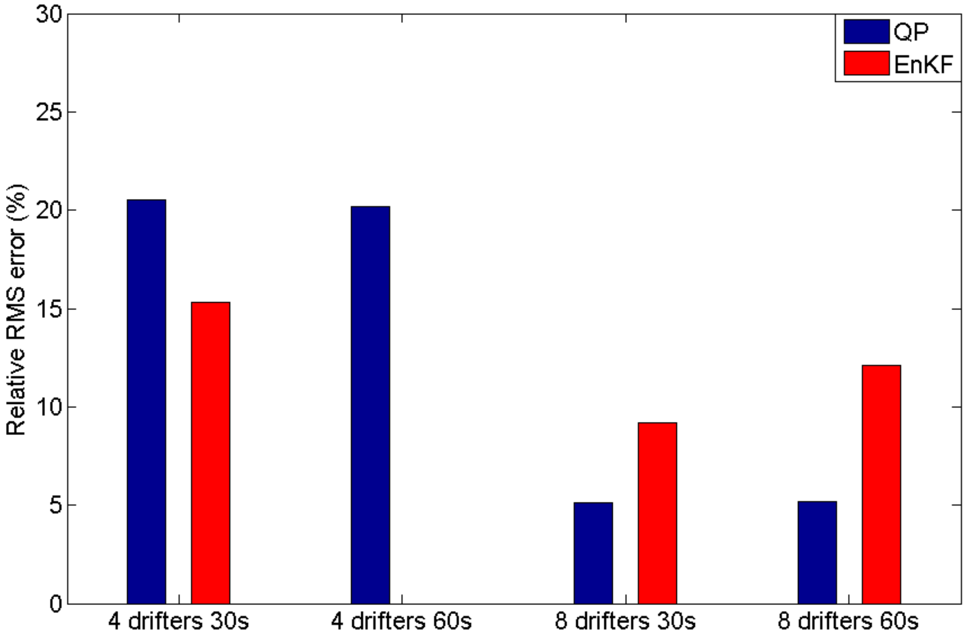

Figure 2.8: Relative RMS error (%) for both methods in the case of variable boundary conditions and low flow for four and eight drifters and 30s and 60s time sampling.

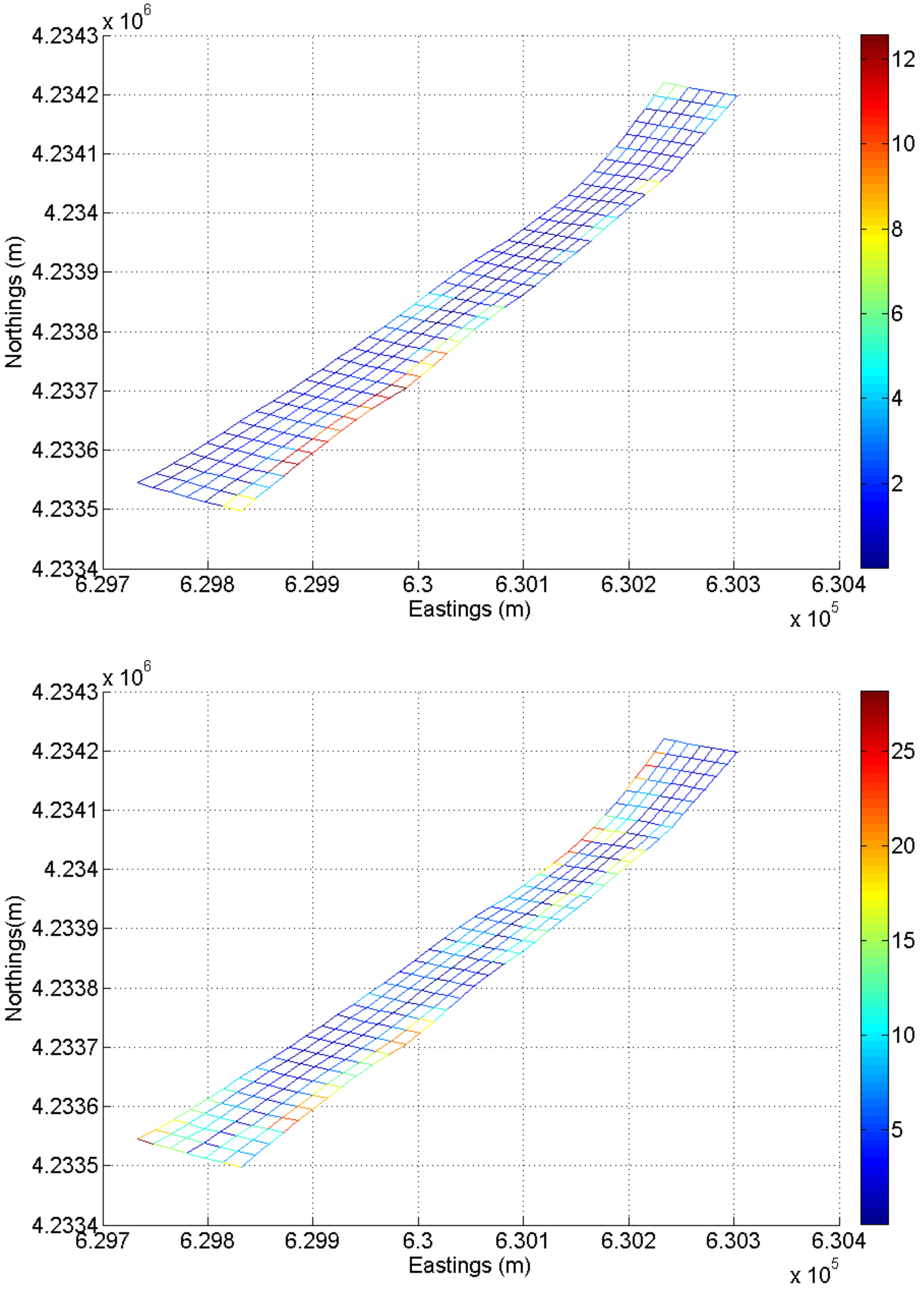

Figure 2.9: Relative RMS error (%) for the QP algorithm in the case of low boundary flow with eight drifters (top) and four drifters (bottom).

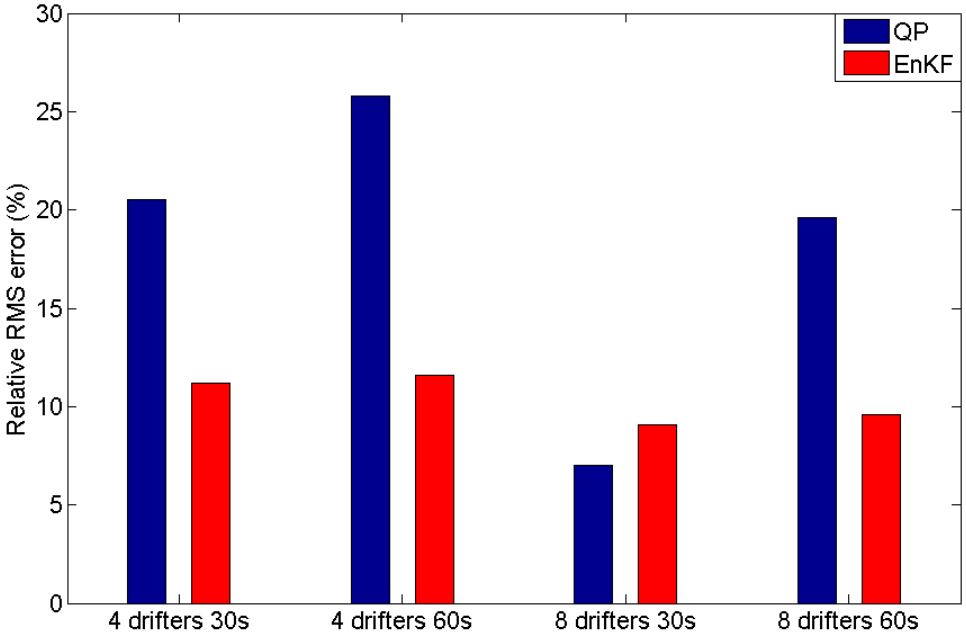

Figure 2.10: Relative RMS error (%) for both methods in the case of variable boundary conditions and high flow for four and eight drifters and 30s and 60s time sampling.

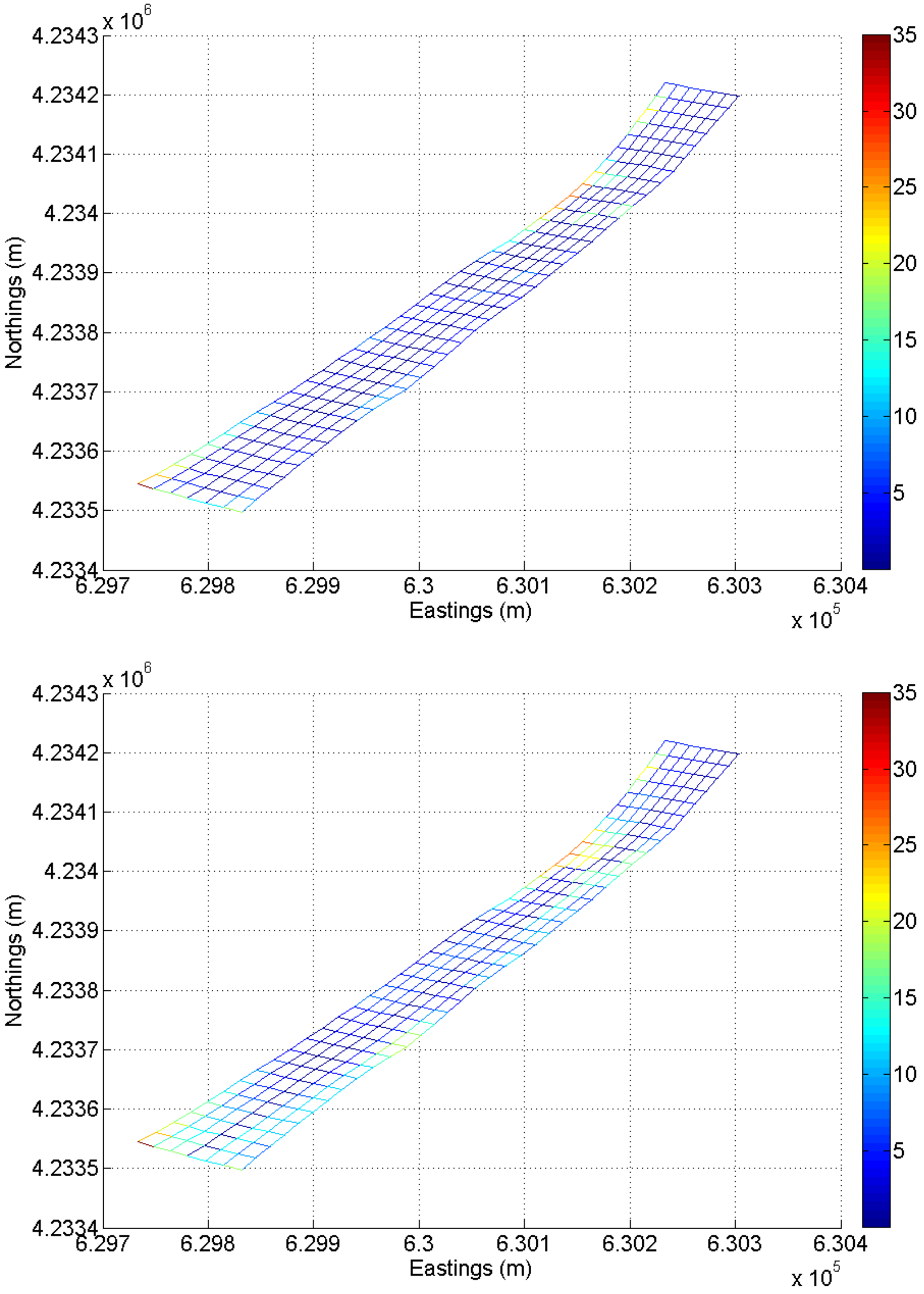

Figure 2.11: Relative RMS error (%) for the QP algorithm in the case of high boundary flow with eight drifters (top) and four drifters (bottom).

the fact that the EnKF is based on estimating the variation of the flow. On the other hand, the QP algorithm shows a significant improvement when the number of drifters increases with the relative RMS error being divided by two for constant (Figures 2.6 and 2.7) and high flows (Figures 2.10 and 2.11) and by three for low flows (Figures 2.8 and 2.9). This is understandable, as when fewer observations are available, the background term (with a relative RMS error close to 40%) becomes more dominant, thus increasing the error.

The time sampling has a greater influence on the quadratic programming estimate than on the EnKF estimate. Indeed, the difference in time sampling is not relevant for the EnKF for the cases with varying boundary conditions. For the QP algorithm, the observations time sampling has no influence on the relative RMS error for the constant boundary conditions (Figure 2.6) and varying boundary conditions with low flow (Figure 2.8); the error is constant for constant boundary conditions with four or eight drifters and increases by 0.3% for low flow with four drifters. However it makes a significant difference (5.3% increase in RMS error with four drifters and 12.6% increase with eight drifters) in the case of high flow (Figures 2.10 and 2.12). The difference in the estimation error is explained by the fact that at velocities encountered in this experiment (0.8 m/s) a drifter will have floated through two or three grid cells of the QP grid in between two measurements resulting in the absence of observations for about half of the grid cells. This is compensated in the variational data assimilation problem by giving a greater influence to the background term which automatically increases the relative RMS error. This phenomenon does not occur for the EnKF. When the velocity is lower (0.4 m/s), the drifters at most move from one cell to the next in between two measurements, and provide at least one observation for every cell on their trajectory. Thus the error for the QP is not affected by the time sampling for low flows.

2.4.3 Description of GPS drifters

In addition to the twin experiments described in the previous section, a fleet of GPS equipped drifters (Figure 2.18) was designed and manufactured in the Lagrangian Sensor Systems Laboratory of the Department of Civil and Environmental Engineering and used in the same part of the Sacramento River (Figure 2.19) for a number of field experiments [155], led by Andrew Tinka. Each drifter consists of a watertight

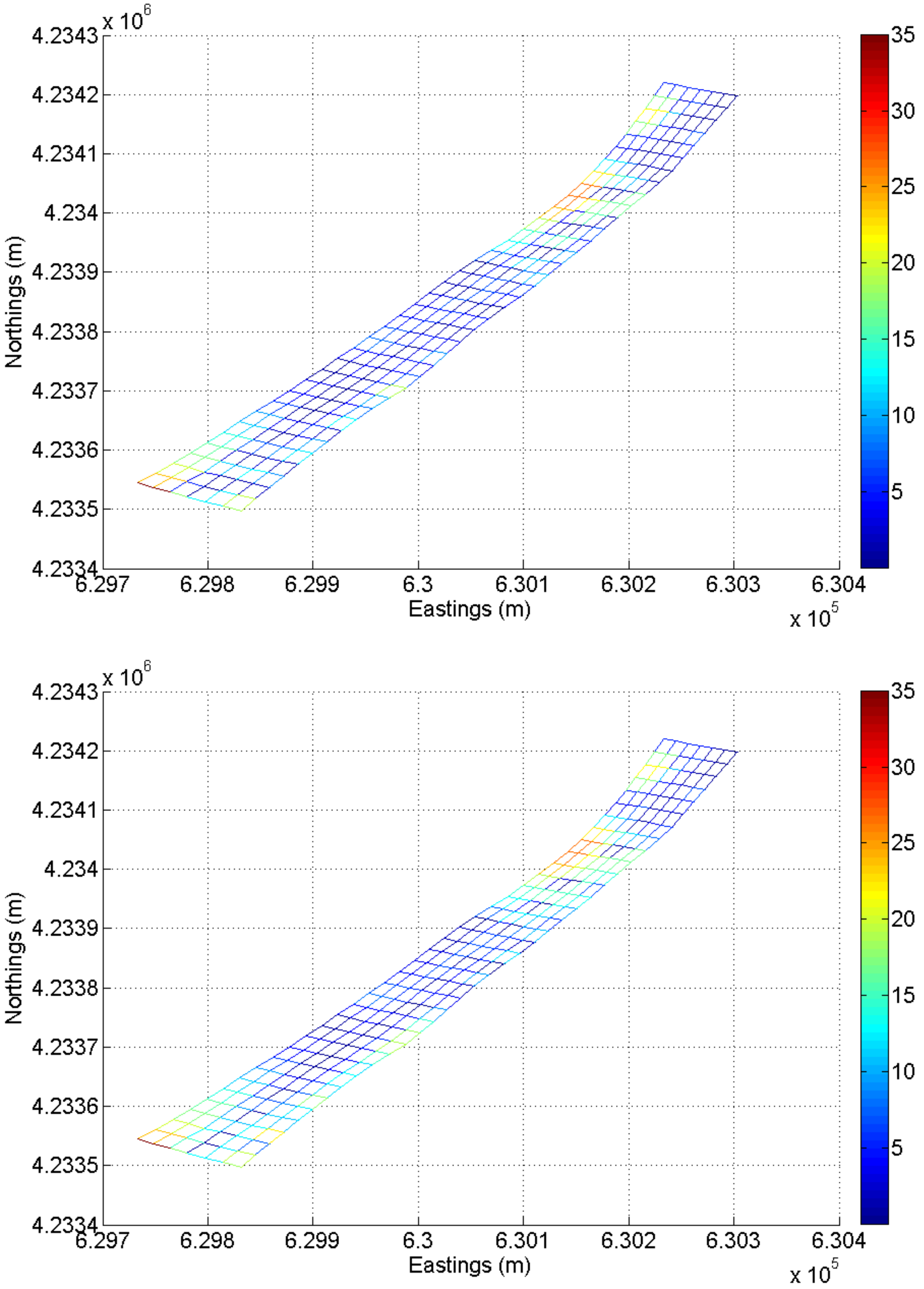

Figure 2.12: Relative RMS error (%) for the QP algorithm in the case of high boundary flow with eight drifters (top) and four drifters (bottom) for an observations time sampling of 60 seconds.

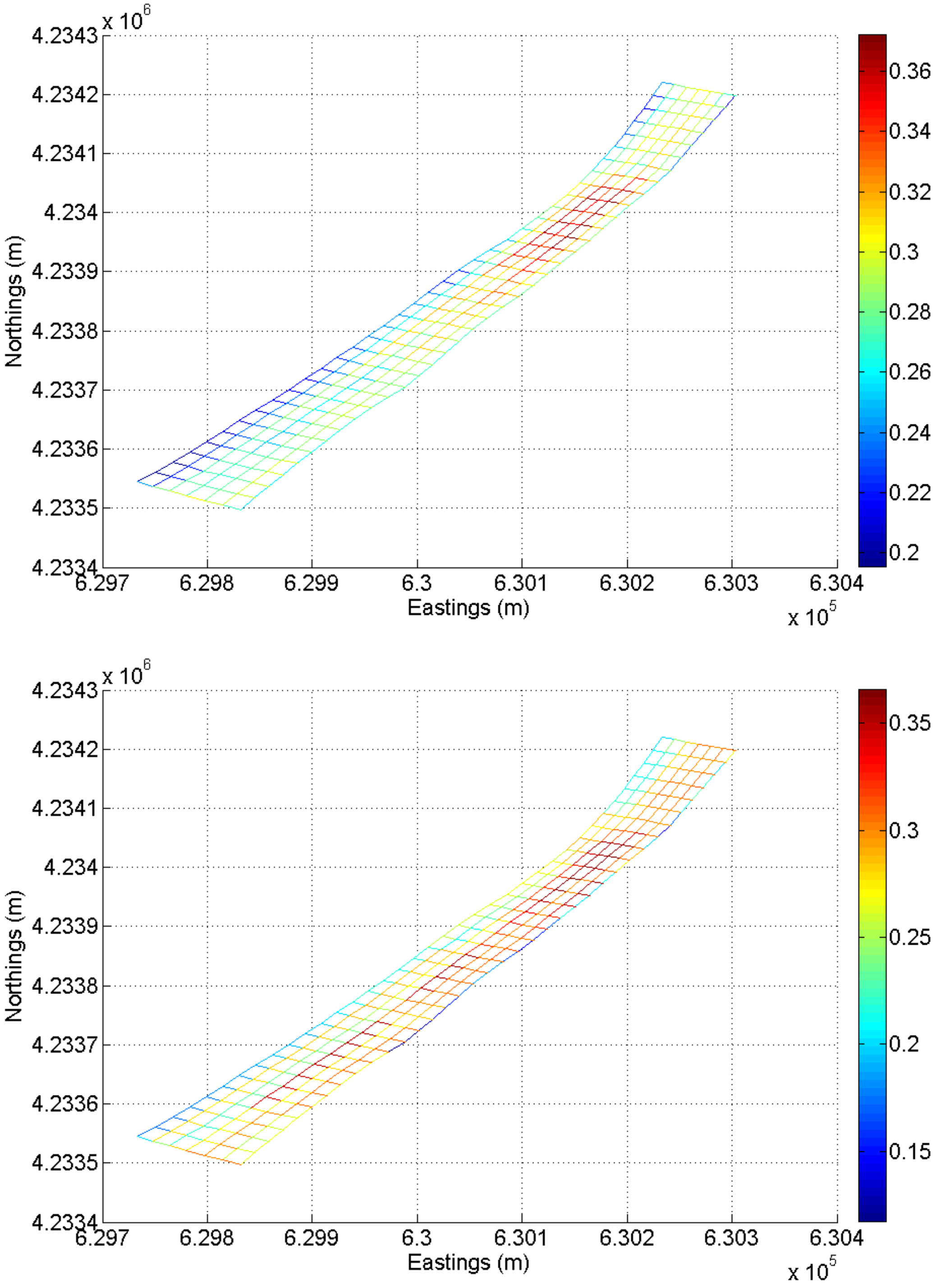

Figure 2.13: Magnitude of the assimilated velocity fields (m/s) generated by the QP algorithm for constant boundary conditions: 8 drifters (top), 4 drifters (bottom).

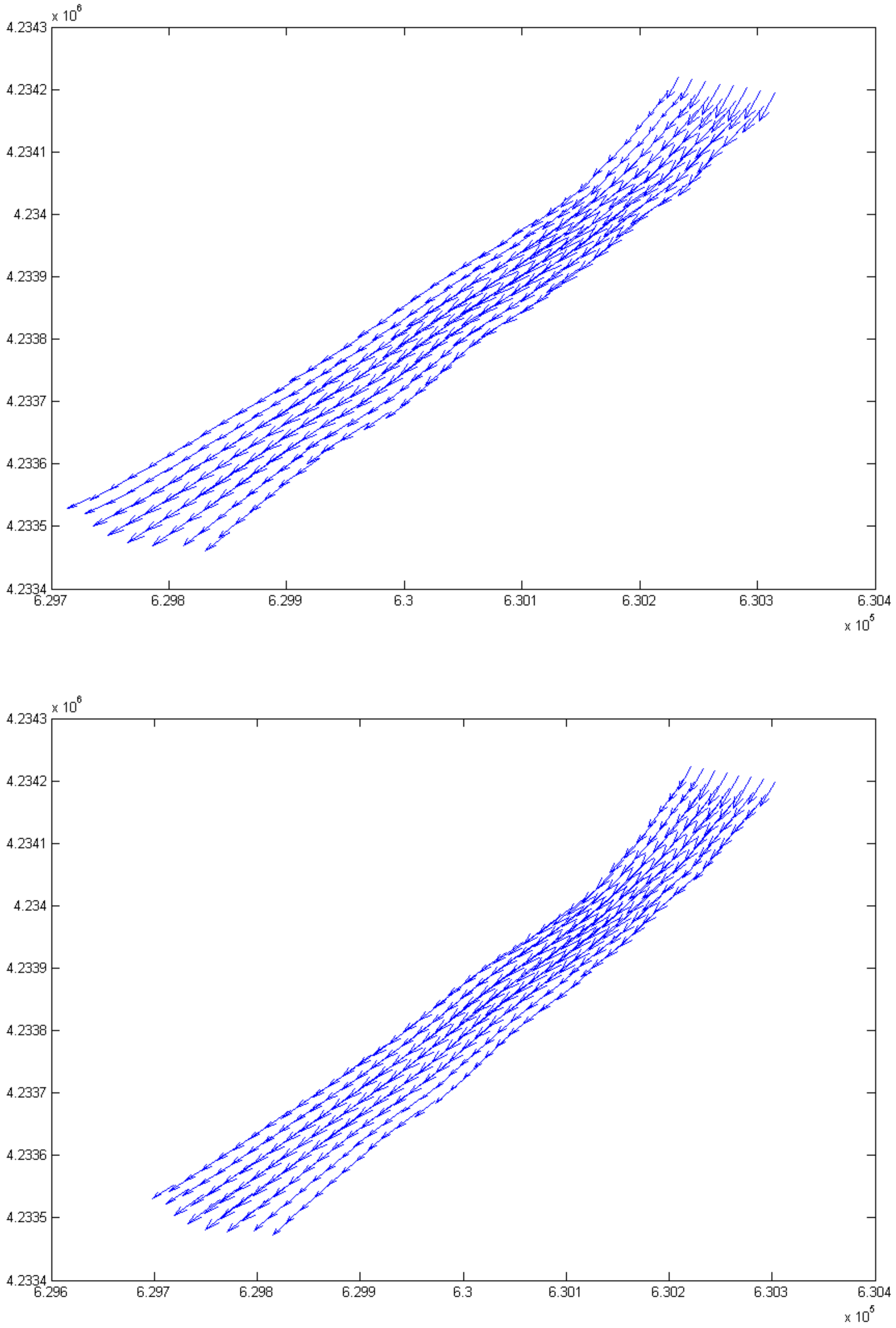

Figure 2.14: Quiver plots of the assimilated velocity field obtained from the QP algorithm in the case of constant boundary conditions with eight drifters (top) and four drifters (bottom).

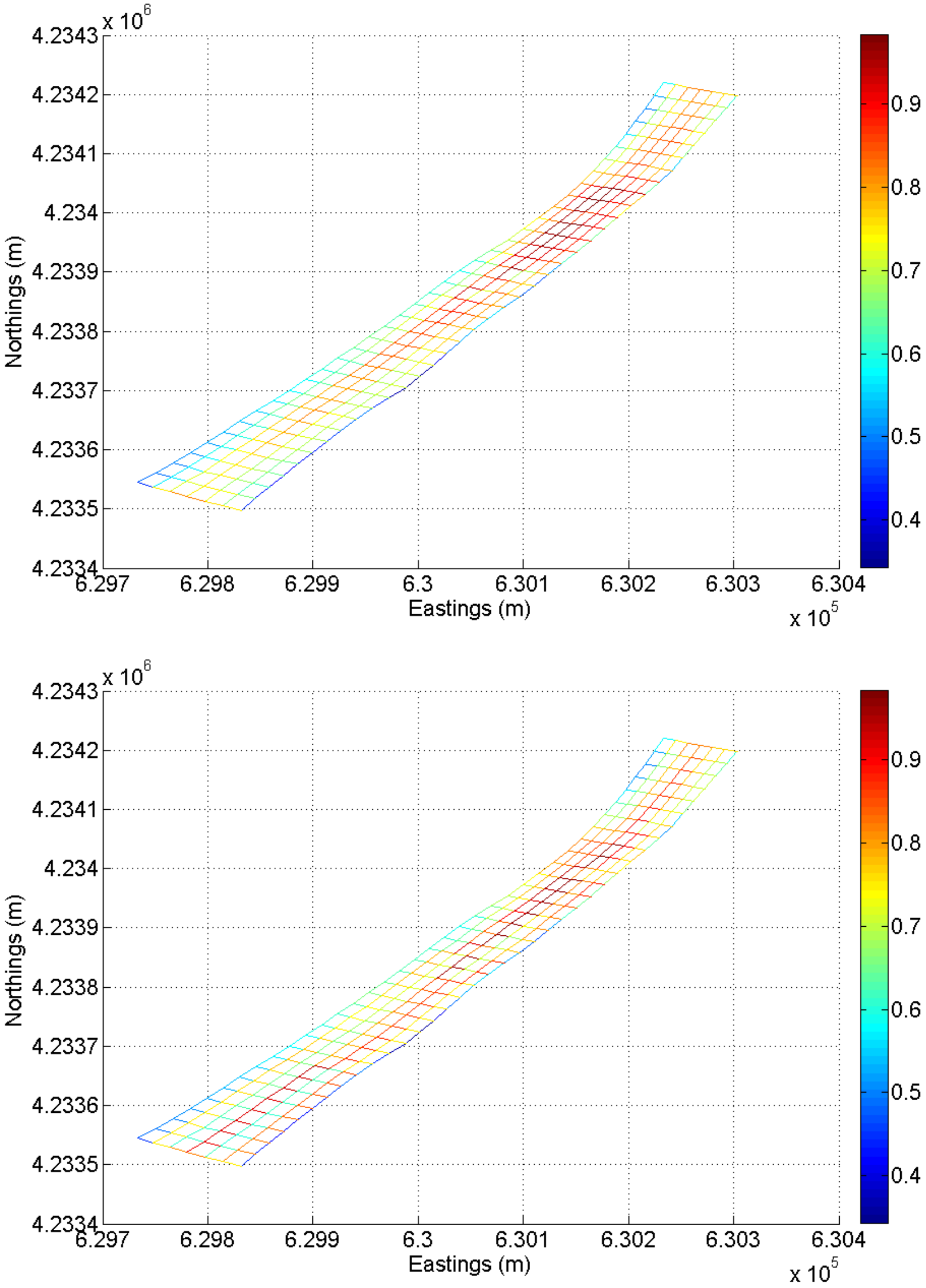

Figure 2.15: Magnitude of the assimilated velocity fields (m/s) generated by the QP algorithm for high flow boundary conditions: 8 drifters (top), 4 drifters (bottom).

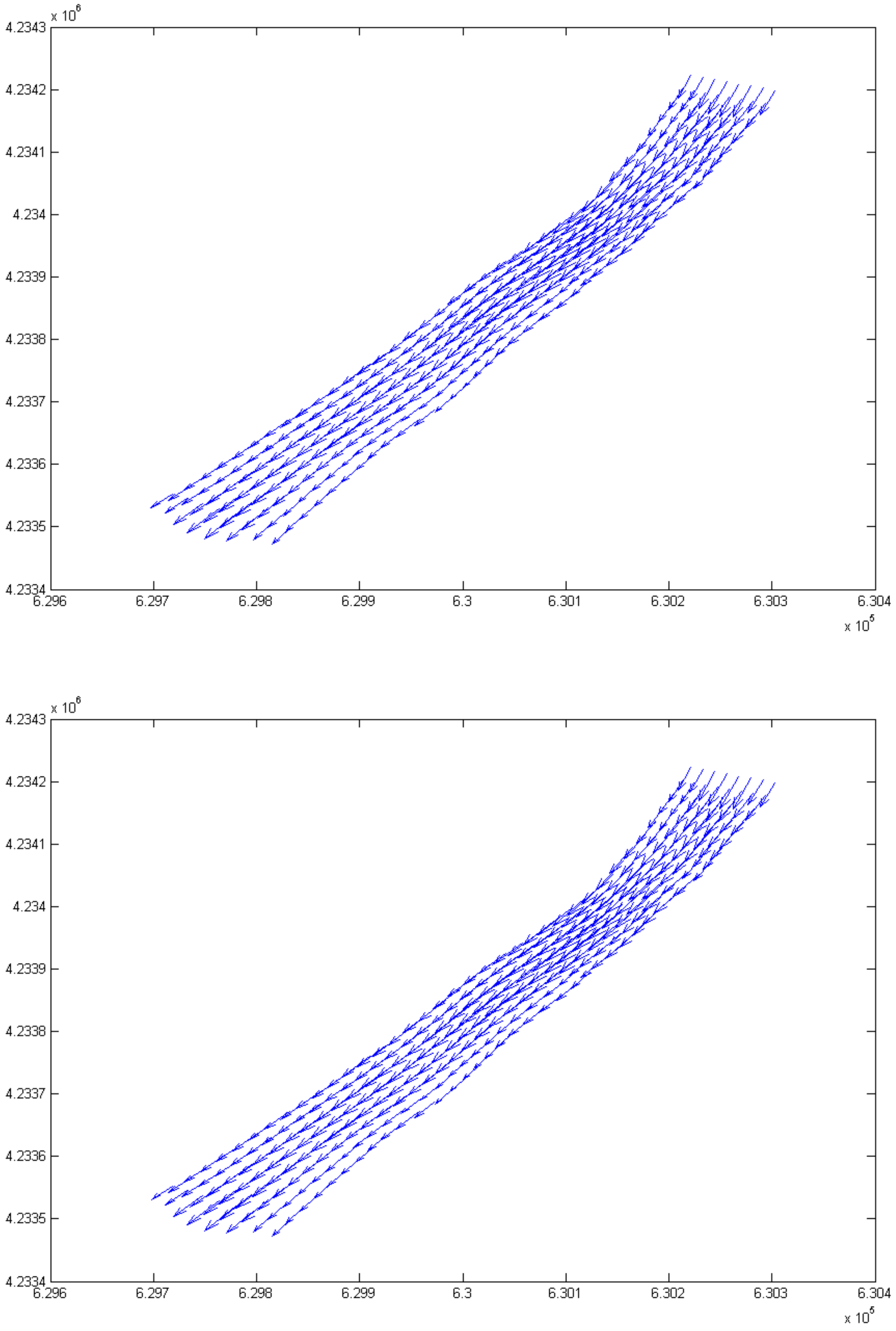

Figure 2.16: Quiver plots of the assimilated velocity field obtained from the QP algorithm in the case of high boundary flow with eight drifters (top) and four drifters (bottom).

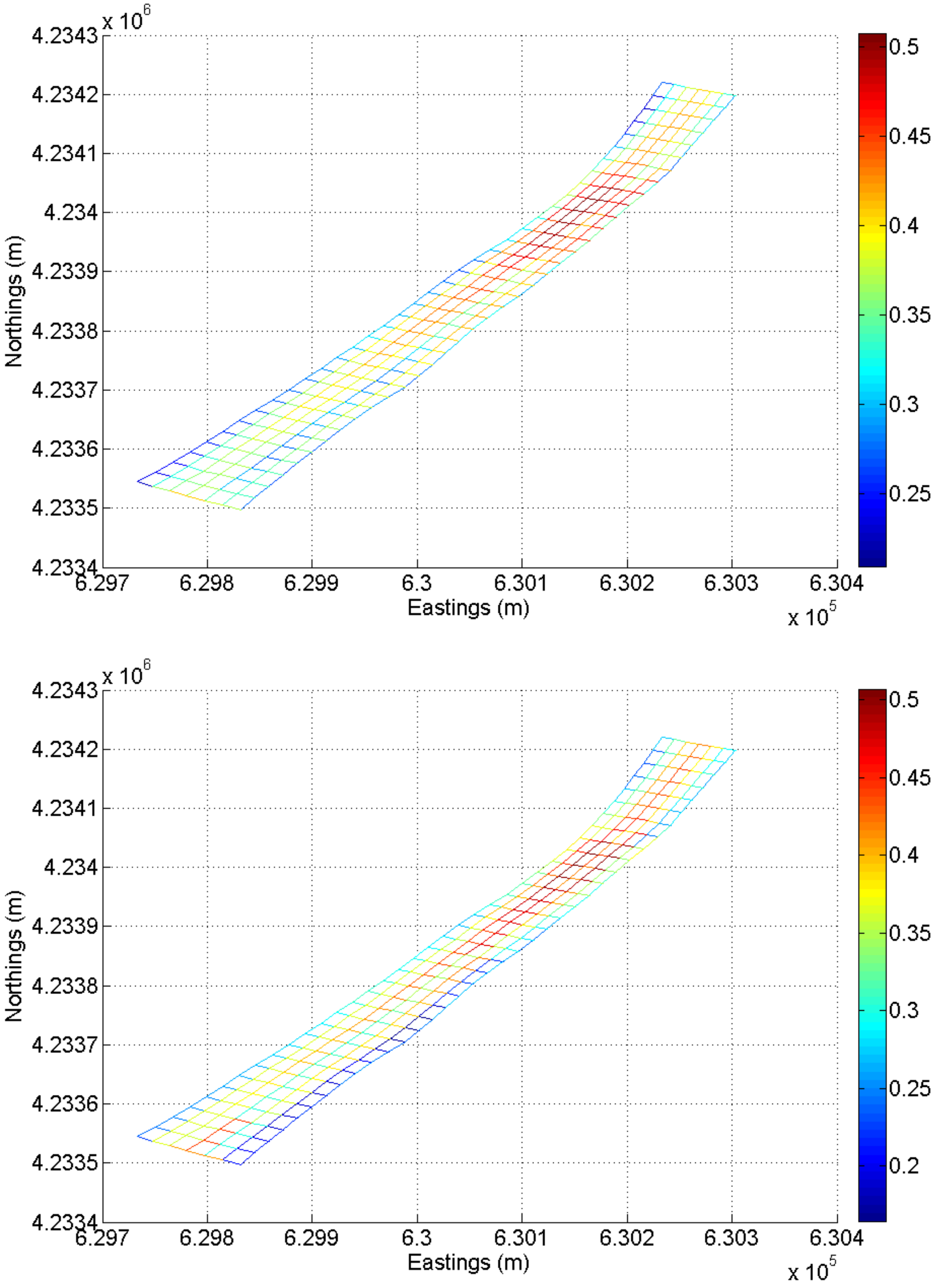

Figure 2.17: Magnitude of the assimilated velocity fields (m/s) generated by the QP algorithm for low flow boundary conditions: 8 drifters (top), 4 drifters (bottom).

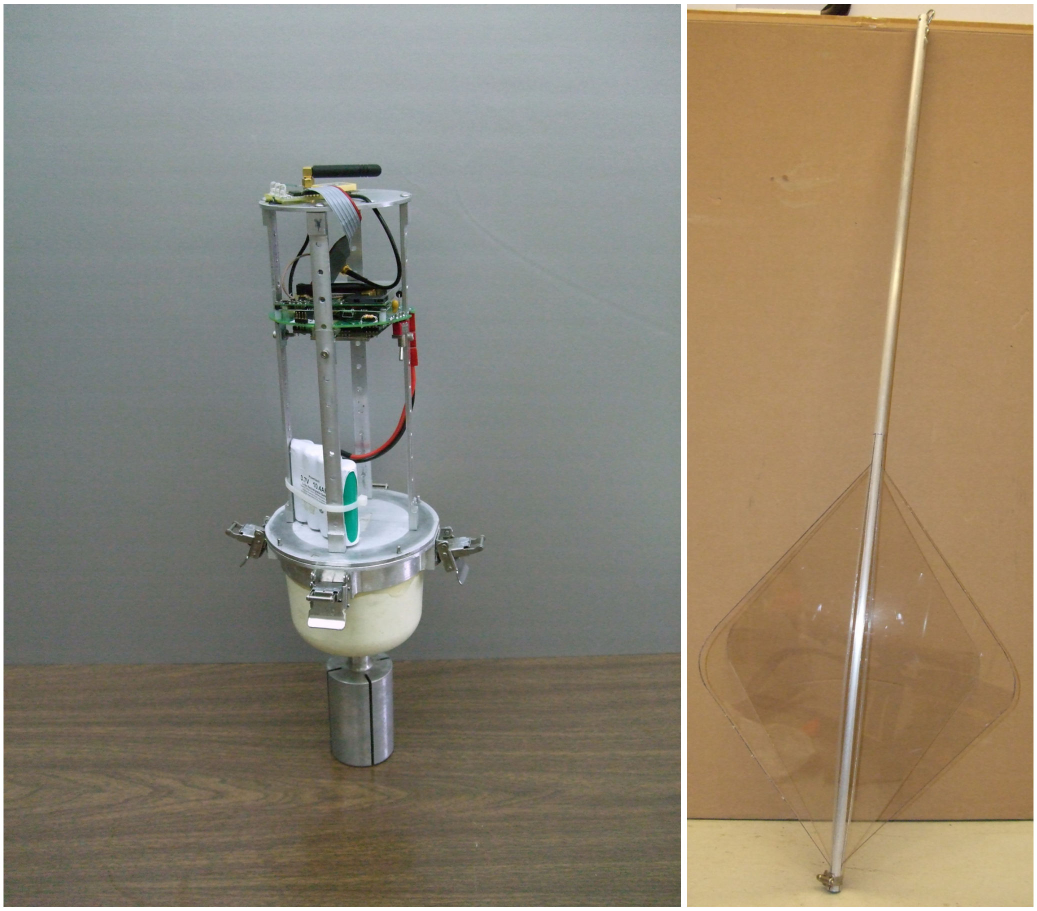

Figure 2.18: Left: each drifter contains a number of electronic devices including a GPS receiver, a GUMSTIX computing module, a GSM transmission module, a GSM antenna and a battery module. Right: a 1.3 metre aluminum tube is attached to the bottom of the drifter at the end of which are two polycarbonate plates, the objective being that the drifter be driven primarily by the current below the surface as opposed to the wind-mixed layer that may be present at the surface. Photos courtesy of Andrew Tinka.

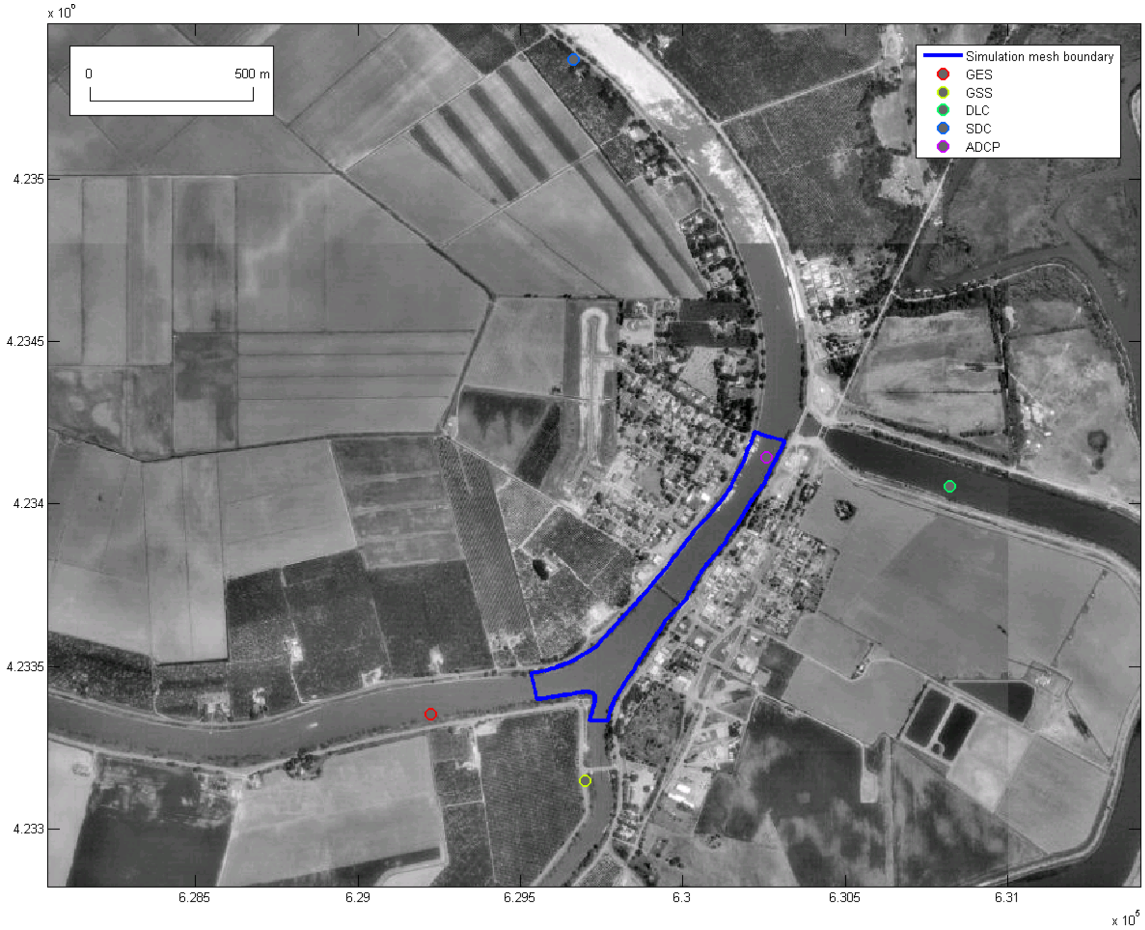

Figure 2.19: Satellite image of the portion of the Sacramento River studied, located near the town of Walnut Grove. The location of the ADCP sensor is represented by a blue dot in the top right part of the domain. Figure courtesy of Andrew Tinka.

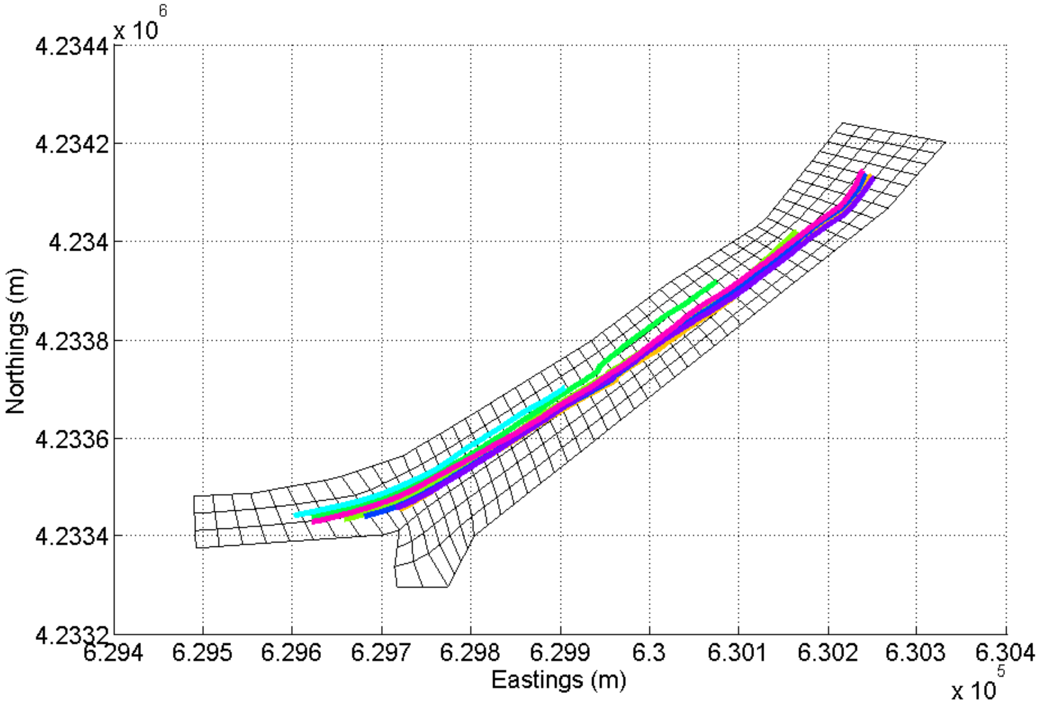

Figure 2.20: Trajectories of the drifters along the Sacramento River during one of the November 12 field operational tests.

fiberglass hull containing a GPS receiver, a low-power embedded computer running Linux (GUMSTIX), a GSM module for communication and an underwater drogue attached to the bottom of the hull (Figure 2.18). While the hull of the drifter floats at surface level, a 1.3 m aluminium tube is attached to a lug in the lower hull with a cotter pin. At the opposite end of the tube, two polycarbonate plates, each having an area of 1600 square centimetres, are mounted diagonally. This puts a large drag component about one metre below the drifter hull. The intent is to make sure that the drifter is driven primarily by the current below the surface as opposed to the wind-mixed layer that may be present at the surface. With a unit price of $700, such a drifter is quite cheap when compared to fixed measurement stations. On November 12, 2007 and November 16, 2007, four drifter deployments were performed on the portion of the Sacramento River studied earlier in the chapter. For each experiment, seven drifters were placed in the water by UC Berkeley students in a small boat. The initial positions were roughly in a straight line across the river, with approximately even spacing, but in the centre of the river to avoid obstacles such as marinas and shallow areas on the sides. The drifters logged their GPS positions as they travelled along the river. Figure 2.20 depicts an example of drifter trajectories reconstructed from the GPS position data downloaded from the drifters after the experiments. From this data, the velocity of the drifter can be computed using finite differences.

Other fixed sensors were also deployed during the period of the experiments, among which is an *Acoustic Doppler Current Profiler* (ADCP) installed near the upstream boundary (Figure 2.19). The ADCP [119] is placed on the river bed at a given location, and measures velocity magnitude and direction using the Doppler shift of acoustic energy reflected by material suspended in the water column above the sensor. The ADCP transmits pairs of short acoustic pulses along a narrow beam from each of its four transducers. As the pulses travel through the water column, they strike suspended sediment and organic particles (known as scatterers) that reflect some of the acoustic energy back to the ADCP. The ADCP receives and records the reflected pulses. The reflected pulses are separated by time differences into successive, uniformly spaced volumes called depth cells. The frequency shift (known as Doppler effect) and the time-lag change between successive reflected pulses are proportional to the velocity of the scatterers relative to the ADCP. The ADCP computes a velocity component along each beam; because the beams are positioned orthogonally to one

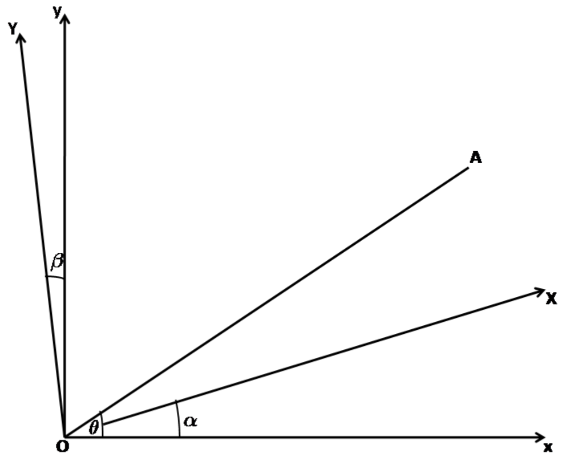

Figure 2.21: Non-orthogonal curvilinear coordinate system.

another and at a known angle from the vertical (usually 20 or 30 degrees), trigonometric relations are used to compute three-dimensional water velocity vectors for each depth cell. Thus, the ADCP produces vertical velocity profiles composed of water speeds and directions at a regularly spaced interval, in our case 0.25 m. These velocity readings are then averaged and used as a validation tool later in this section as the assimilated velocity at the location of the ADCP obtained after using the drifter data in the quadratic programming data assimilation algorithm will be compared to the average velocity measured by the ADCP.

2.4.4 Implementation of the method on a non-orthogonal curvilinear grid

In order to fit the channel boundary, we use a non-orthogonal curvilinear mesh. This implies a change in coordinates as the shallow water equations and corresponding numerical scheme written earlier on in this chapter are in cartesian coordinates. Several methods can be used such as using the distances measured in the two directions along the lines of the mesh as new coordinates (see [22, 50] for applications of this method to the shallow water equations and finite volume discretisation, [144] for the unstructured mesh case and [168] for the resolution of 3D shallow water equations in non-orthogonal curvilinear coordinates). In this work, we choose to follow the novel technique recently exposed in [59] in which the equations are developed in terms of the local deviations from Cartesian axes.

If Ox and Oy denote Cartesian axes while $OX$and OY are non-orthogonal curvilinear axes (see Figure 2.21), the curvilinear components of the velocity (U_{CL}, V_{CL}) can be expressed in terms of its Cartesian components (U_C, V_C):

$$U_{CL} = (U_C \cos\beta + V_C \sin\beta)\sec(\alpha - \beta)$$

$$V_{CL} = (V_C \cos\alpha - U_C \sin\alpha)\sec(\alpha - \beta) \tag{2.75}$$

Similarly, the contravariant components of the velocity (U_{CV}, V_{CV}), needed for the continuity equation are related to the Cartesian components by:

$$U_{CV} = (U_C \cos\alpha + V_C \sin\alpha)\sec(\alpha - \beta)$$

$$V_{CV} = (V_C \cos\beta - U_C \sin\beta)\sec(\alpha - \beta) \tag{2.76}$$

Conversely, the cartesian components of the velocity are given in terms of the curvilinear components (U_{CL}, V_{CL}) by:

$$U_C = U_{CL} \cos\alpha - V_{CL} \sin\beta$$

$$V_C = U_{CL} \sin\alpha + V_{CL} \cos\beta \tag{2.77}$$

Using these equalities we obtain the shallow water equations in non-orthogonal curvilinear coordinates:

$$\begin{aligned}&\frac{\partial U_{CL}}{\partial t} + U_{CL}\frac{\partial U_{CL}}{\partial X} + V_{CL}\frac{\partial U_{CL}}{\partial Y} + \left(U_{CL}\frac{\partial U_{CL}}{\partial Y} + V_{CL}\frac{\partial U_{CL}}{\partial X}\right)\sin(\alpha - \beta)\\ &+ g\frac{\partial h}{\partial X} + C_{x,y}U_{CL} = -g\frac{\partial b}{\partial x}\end{aligned} \tag{2.78}$$

$$\begin{aligned}&\frac{\partial V_{CL}}{\partial t} + U_{CL}\frac{\partial V_{CL}}{\partial X} + V_{CL}\frac{\partial V_{CL}}{\partial Y} + \left(U_{CL}\frac{\partial V_{CL}}{\partial Y} + V_{CL}\frac{\partial V_{CL}}{\partial X}\right)\sin(\alpha - \beta)\\ &+ g\frac{\partial h}{\partial Y} + C_{x,y}V_{CL} = -g\frac{\partial b}{\partial y}\end{aligned} \tag{2.79}$$

$$\frac{\partial h}{\partial t} + \left(\frac{\partial hU_{CV}}{\partial X} + \frac{\partial hV_{CV}}{\partial Y}\right)\sec(\alpha - \beta) = 0 \tag{2.80}$$

The corresponding linearised equations can obtained as was done in Cartesian coordinates:

$$\begin{aligned}&\frac{\partial U_{CL}}{\partial t} + \left(U^0_{CL}(x,y,t)\frac{\partial U_{CL}}{\partial Y} + V^0_{CL}(x,y,t)\frac{\partial U_{CL}}{\partial X}\right)\sin(\alpha - \beta)\\ &+ U^0_{CL}(x,y,t)\frac{\partial U_{CL}}{\partial X} + V^0_{CL}(x,y,t)\frac{\partial U_{CL}}{\partial Y} + g\frac{\partial h}{\partial X} + C_{x,y}U_{CL} = -g\frac{\partial b}{\partial x}\end{aligned} \tag{2.81}$$

$$\begin{aligned}&\frac{\partial V_{CL}}{\partial t}+\left(U^0_{CL}(x,y,t)\frac{\partial V_{CL}}{\partial Y}+V^0_{CL}(x,y,t)\frac{\partial V_{CL}}{\partial X}\right)\sin(\alpha-\beta)\\&+U^0_{CL}(x,y,t)\frac{\partial V_{CL}}{\partial X}+V^0_{CL}(x,y,t)\frac{\partial V_{CL}}{\partial Y}+g\frac{\partial h}{\partial Y}+C_{x,y}V_{CL}=-g\frac{\partial b}{\partial y}\end{aligned}\tag{2.82}$$

$$\begin{aligned}&\frac{\partial h}{\partial t}+\Bigg(U^0_{CV}(x,y,t)\frac{\partial h}{\partial X}+V^0_{CV}(x,y,t)\frac{\partial h}{\partial Y}+\frac{H_0(x,y,t)\partial U_{CV}}{\partial X}\\&+\frac{H_0(x,y,t)\partial V_{CV}}{\partial Y}\Bigg)\sec(\alpha-\beta)=0\end{aligned}\tag{2.83}$$

We then use the scheme described in Equations (2.50), (2.51), (2.52) to discretise these equations, denoting by $\alpha_{i,j}$ and $\beta_{i,j}$ the angles between the sides of the cell (i,j) and the Cartesian axes:

$$\begin{aligned}&\frac{u^n_{i,j}-u^{n-1}_{i,j}}{\Delta t}+\frac{1}{2a^x_i}(U^{0,i,j,n}_{CL}(u^n_{i+1,j}-u^n_{i-1,j})+g(h^n_{i+1,j}-h^n_{i-1,j}))\\&+\left(\frac{U^{0,i,j,n}_{CL}}{2a^y_j}(u^n_{i,j+1}-u^n_{i,j-1})+\frac{V^{0,i,j,n}_{CL}}{2a^x_i}(u^n_{i+1,j}-u^n_{i-1,j})\right)\sin(\alpha_{i,j}-\beta_{i,j})\\&+\frac{V^{0,i,j,n}_{CL}}{2a^y_j}(u^n_{i,j+1}-u^n_{i,j-1})+C_{i,j}u^n_{i,j}=-\frac{g}{2a^x_i}(b_{i+1,j}-b_{i-1,j})\end{aligned}\tag{2.84}$$

$$\begin{aligned}&\frac{v^n_{i,j}-v^{n-1}_{i,j}}{\Delta t}+\frac{1}{2a^y_j}(V^{0,i,j,n}_{CL}(v^n_{i,j+1}-v^n_{i,j-1})+g(h^n_{i,j+1}-h^n_{i,j-1}))\\&+\left(\frac{U^{0,i,j,n}_{CL}}{2a^y_j}(v^n_{i,j+1}-v^n_{i,j-1})+\frac{V^{0,i,j,n}_{CL}}{2a^x_i}(v^n_{i+1,j}-v^n_{i-1,j})\right)\sin(\alpha_{i,j}-\beta_{i,j})\\&+\frac{U^{0,i,j,n}_{CL}}{2a^x_i}(v^n_{i+1,j}-v^n_{i-1,j})+C_{i,j}v^n_{i,j}=-\frac{g}{2a^y_j}(b_{i,j+1}-b_{i,j-1})\end{aligned}\tag{2.85}$$

$$\begin{aligned}&\frac{h^n_{i,j}-h^{n-1}_{i,j}}{\Delta t}\\&+\Bigg(\frac{U^{0,i,j,n}_{CL}\sec(\alpha_{i,j}-\beta_{i,j})+V^{0,i,j,n}_{CL}\tan(\alpha_{i,j}-\beta_{i,j})}{2a^x_i}(h^n_{i+1,j}-h^n_{i-1,j})\\&+\frac{V^{0,i,j,n}_{CL}\sec(\alpha_{i,j}-\beta_{i,j})+U^{0,i,j,n}_{CL}\tan(\alpha_{i,j}-\beta_{i,j})}{2a^y_j}(h^n_{i,j+1}-h^n_{i,j-1})\Bigg)\\&+H^{i,j,n}_0\Bigg(\frac{1}{2a^x_i}\big((u^n_{i+1,j}-u^n_{i-1,j})\sec(\alpha_{i,j}-\beta_{i,j})+(v^n_{i+1,j}-v^n_{i-1,j})\tan(\alpha_{i,j}-\beta_{i,j})\big)\\&+\frac{1}{2a^y_j}\big((v^n_{i,j+1}-v^n_{i,j-1})\sec(\alpha_{i,j}-\beta_{i,j})\\&+(u^n_{i,j+1}-u^n_{i,j-1})\tan(\alpha_{i,j}-\beta_{i,j})\big)\Bigg)\sec(\alpha_{i,j}-\beta_{i,j})=0\end{aligned}\tag{2.86}$$

where the relation between the contravariant components of the velocity and the curvilinear components:

$$U_{CV} = U_{CL}\sec(\alpha - \beta) + V_{CL}\tan(\alpha - \beta) \tag{2.87}$$

$$V_{CV} = V_{CL}\sec(\alpha - \beta) + U_{CL}\tan(\alpha - \beta) \tag{2.88}$$

has been used in the discretisation of the continuity equation.

2.4.5 Results

For each of the four drifter releases, the data was input in the QP data assimilation algorithm and generated the assimilated velocity fields at the initial time (Figures 2.22 and 2.24). While for twin experiments, a uniform coverage of the width of the river was obtained, in the present case, numerous obstacles such as marinas or partially submerged trees prevent us from releasing drifters near the banks of the river, resulting in most of the drifters being concentrated in the centre of the river, as can be seen from Figure 2.20. This, in turn, results in a lack of observed data away from the middle of the river, which is compensated in our algorithm by the background term. Therefore, we chose to use a two-dimensional shallow water simulation provided by the software package TELEMAC as background term for each of these experiments; the approximate discharge upstream and stage downstream at the time of each experiment were obtained from USGS sensors placed near the domain (see Figure 2.19). This allowed us to improve the accuracy of the assimilation algorithm compared to the situation of the twin experiments in which a uniform background term was used. Note however that for the twin experiments, the true state was given by a TELEMAC simulation and it was therefore impossible to use such a simulation as background term since it would render the comparison between assimilated state and true state meaningless.

As can be seen on the velocity fields, the velocity on the sides of the river tends to be more uniform than in the centre. This is due to the fact that since drifters are located in the centre, the background term has a higher influence on the sides. The assimilated velocity at the location of the ADCP (in our case cell (3,5)) for each of the four experiments is compared to the velocity measured by the ADCP at that time in Figure 2.26. The respective relative errors are in chronological order: 5.4 %, 10.3 %, 11.1 % and 8.8 %; note that

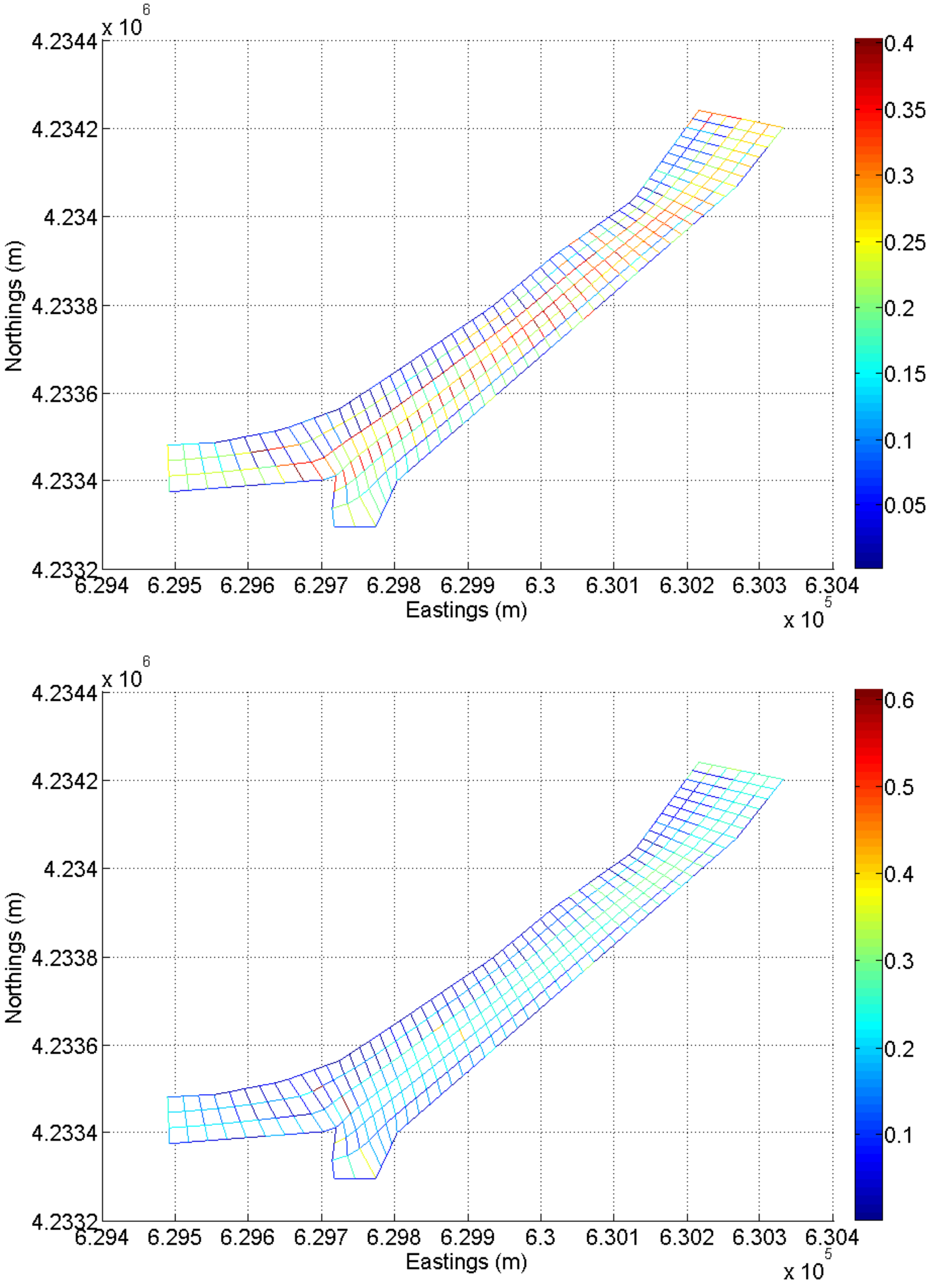

Figure 2.22: Magnitude of the assimilated velocity fields (m/s) generated by the QP algorithm for the two November 12, 2007 experiments.

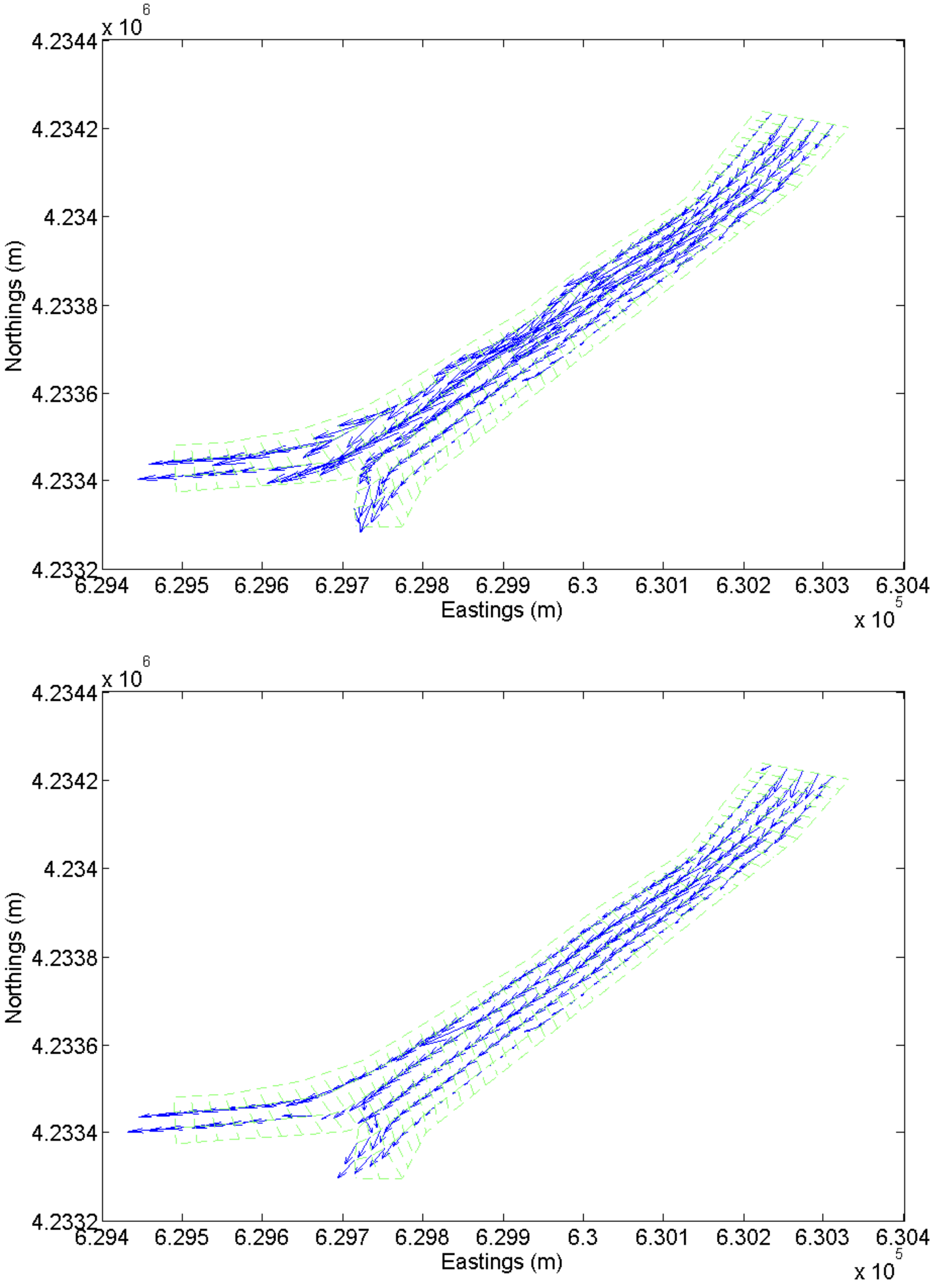

Figure 2.23: Quiver plots of the assimilated velocity fields generated by the QP algorithm for the two November 12, 2007 experiments.

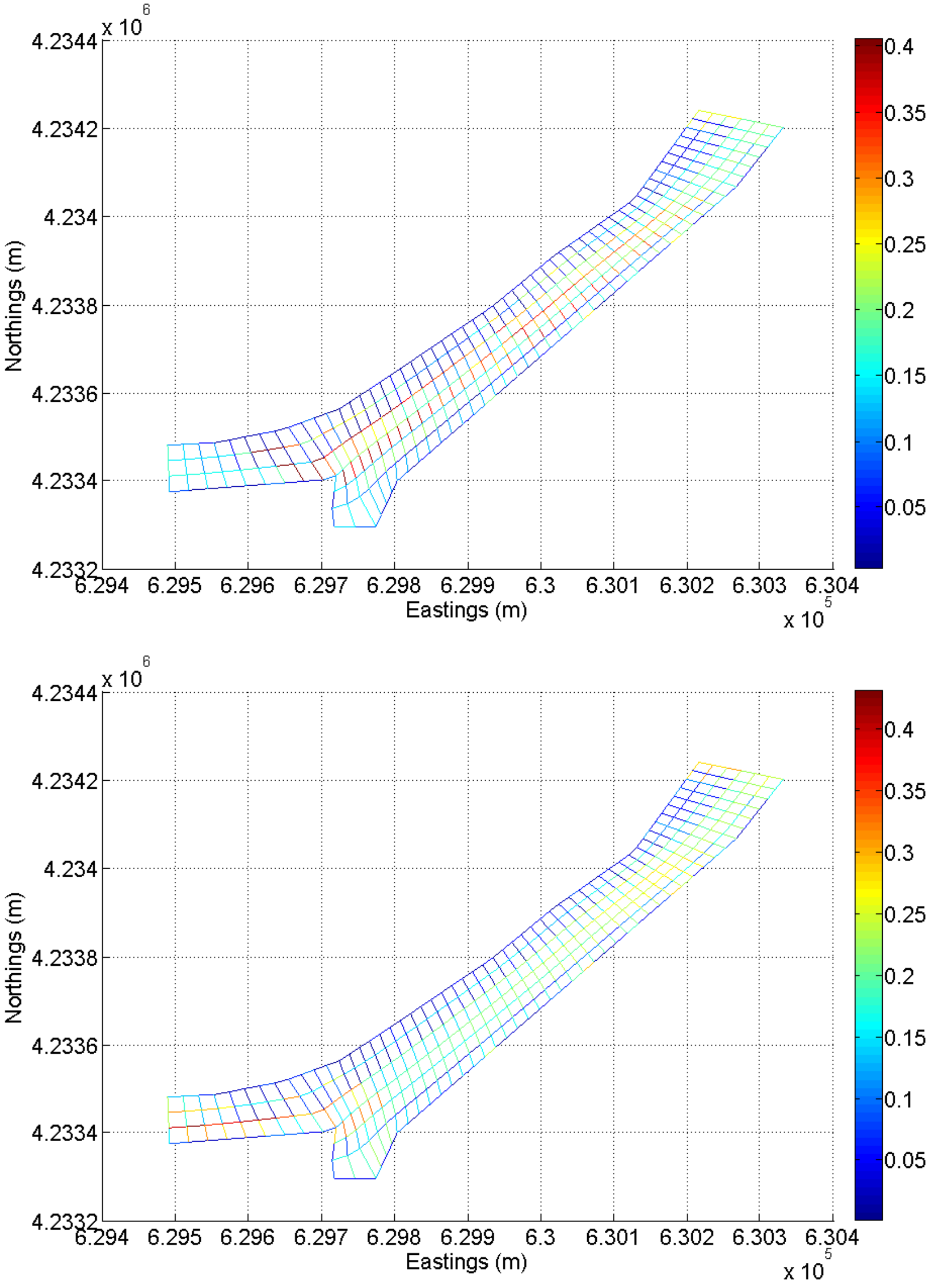

Figure 2.24: Magnitude of the assimilated velocity fields (m/s) generated by the QP algorithm for the two November 16, 2007 experiments.

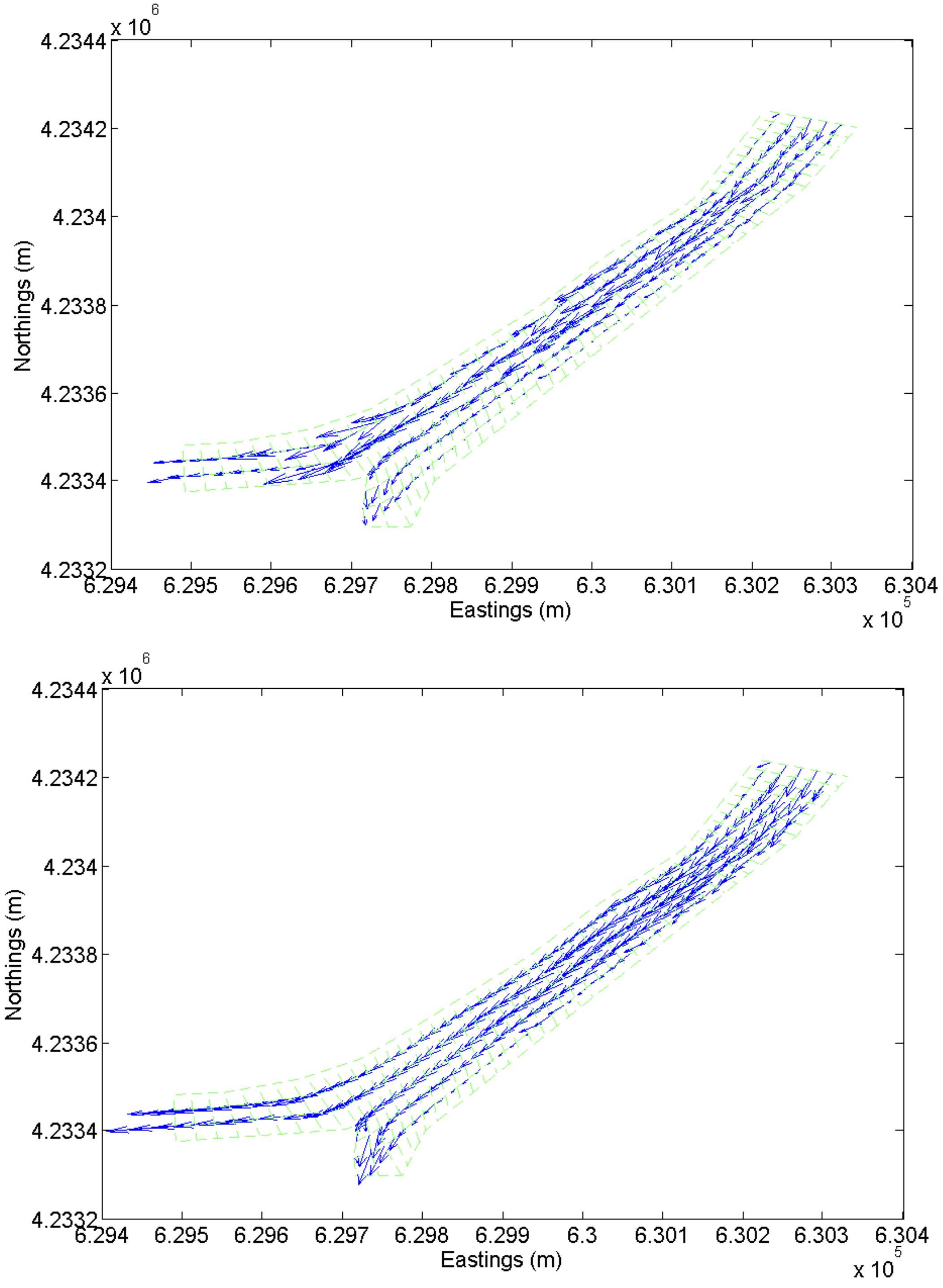

Figure 2.25: Quiver plots of the assimilated velocity fields generated by the QP algorithm for the two November 16, 2007 experiments.

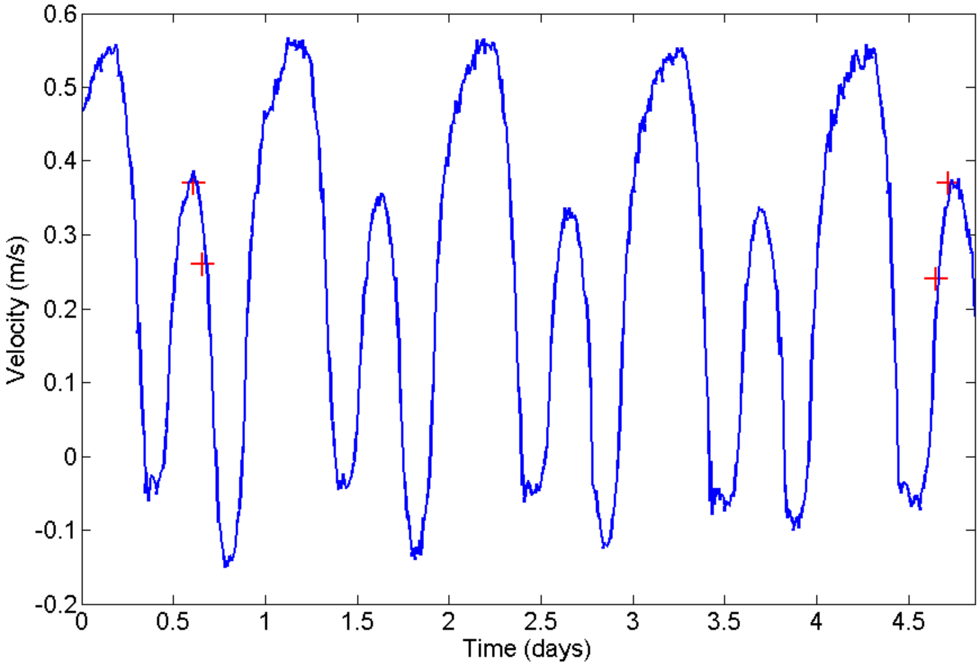

Figure 2.26: Comparison between the ADCP measured velocity and the assimilated velocity at the ADCP location. Each of the four velocities corresponding to the four experiments is represented by a cross.

these values are in the same range (5 – 10 %) as the earlier results obtained with eight drifters in the case of twin experiments. Overall, we can see that the assimilation of experimental data collected by GPS equipped drifters is done quickly (less than 5 seconds) and offers a very good match with the velocities measured by an ADCP at the time of the experiments.

2.4.6 Conclusion

In this chapter, a number of data assimilation methods were presented. Two of them were applied to a stretch of the Sacramento River located in the Sacramento-San Joaquin Delta, and their respective performances were compared. The comparison was made using twin experiments, in which drifter trajectories are simulated, implemented with experimentally measured bottom topography and boundary conditions. The sensitivity of the different data assimilation techniques to the number of drifters, low or high discharge and time sampling frequency was analysed and the respective computational costs of each method compared. One of the conclusion of this study is that the quadratic programming based algorithm introduced in this work presents a good balance of accuracy and low computational cost. While the ensemble Kalman filter algorithm which uses the nonlinear two dimensional shallow water equations is slightly more accurate in some cases, this is at the expense of a much higher computational time (up to 10 hours compared to the few seconds needed for the quadratic programming algorithm) thereby precluding its use for any real time data assimilation application. The quadratic programming method returns higher relative RMS errors when the number of drifters is reduced from eight to four or when the time sampling frequency is changed from 30 to 60 seconds while the ensemble Kalman filter appears less sensitive to time sampling and number of drifters. A first attempt at estimating the optimal deployment strategy was made, showing that the quadratic programming requires one row of drifters released at the beginning of the assimilation period while the ensemble Kalman filter gives better estimate when several rows of drifters are released over the assimilation period. The quadratic programming algorithm was also applied to data collected by drifters used in the same portion of the Sacramento River during four experiments. The comparisons between the velocity field generated by the data assimilation algorithm and velocities measured by an Acoustic Doppler Current Profile located near the upstream boundary of the domain showed a good agreement between the two data sets with a low relative error. This demonstrates the ability of this algorithm to provide the user with an accurate two dimensional velocity field almost instantaneously; this is of great interest when it comes to dealing with unexpected events such as flooding due to levee breach, release of a contaminant, etc. Future research directions include adding a salinity or sediment model to the shallow water equations and applying the data assimilation algorithms in this new setting, as well as running the quadratic programming algorithm in real time during longer experiments covering a larger area. Another issue is the optimal placement of the drifters and the optimal release points in order to maximise the accuracy of the method.

Chapter 3

Inverse estimation and prediction of open boundary conditions in tidal channels

3.1 Introduction

The study of tidal forcing in bays and estuaries is crucial to the monitoring of water quality issues in these areas [48, 167]. Depending on the application of interest and on the required accuracy, flows can be simulated using a full three dimensional approach, or simplified models such as the one or two dimensional shallow water equations [32]. In any case, the numerical models used need to be thoroughly calibrated with parameters depending on geometry and flow features. In addition, a knowledge of open boundary conditions is required for the model to perform adequately. Such boundary conditions can be inferred from measurements made with fixed sensors placed at the boundaries of the domain of interest when they are available or, as is developed in this chapter, using drifters that are circulating in the system under consideration. In this chapter, a novel quadratic programming based variational data assimilation algorithm will be developed and applied to the estimation of open boundary conditions for tidal channel flows [147]. The topic of inverse estimation of boundary conditions in meteorology and oceanography has been studied over the past few decades starting with the pioneering work [137] and more recently [143, 166, 167, 169].

The need for boundary condition estimation is driven in our case by operational requirements: we are interested in deploying drifter fleets in specific areas of the Sacramento-San Joaquin Delta for which current sensing infrastructure and modelling capabilities are insufficient. The Sacramento-San Joaquin Delta is at risk of extensive levee failures, which could be caused by earthquakes, flood or human activities. In such an event, the water quality of the Delta will be negatively affected, due to sediment suspension, salinity intrusion, and potentially agricultural contaminants. It is critical that transport models be available for use following such an event, even in the case where the entire geometry of the system has been altered. Unfortunately, existing models of the Sacramento-San Joaquin Delta rely heavily on historical data sets for calibration. These models would be of limited utility if the system were radically altered, as would occur in the case of extensive levee failures. The use of rapidly-deployed Lagrangian drifters and the estimation of boundary conditions is motivated by this need: we aim to develop a sensing-modelling system that is capable of predicting regional flows and transport in the Delta in real-time without dependence on historical data. The timescale of interest for this analysis is on the timescale of days: we would like to be able to project transport patterns forwards in time by a few days based on only a few days of data. The immediate objective is to estimate open boundary conditions for a model of flows in tidal channels of the Sacramento-San Joaquin Delta. To this end, we develop variational data assimilation techniques in which a cost function measuring the norm of the difference between observations gathered by drifters and model predictions is minimised, subject to constraints given by the discretised model equations; the control variables are the coefficients of the dominant tidal modes present in the upstream boundary condition. A similar variational approach for Lagrangian data assimilation in rivers applied to bottom topography estimation was presented in [75]. The estimation of boundary conditions in the Sacramento-San Joaquin Delta using fixed sensors and data reconciliation techniques was presented in [165] which presents some similarities with our work, but specifically tackles the problem of Eulerian measurements. While a two-dimensional model could be chosen for the identification of the boundary conditions, its computational cost is high and can be avoided by using simpler models such as a one dimensional model. The choice we make here is motivated by the desire to have a rapid, robust estimate of boundary conditions that can be used in real time flow simulations. A somewhat similar approach was used in [57] as a one dimensional shallow water model was combined with a local two dimensional model through optimal control methods. In [55, 56], a method for the estimation of open boundary conditions for the Navier-Stokes equations with a free surface from fixed depth and velocity measurements is developed. All the articles mentioned above use the adjoint method combined with a quasi Newton solver to solve a variational problem. The adjoint method has the main drawback of a high computational cost as 50 to 100 iterations are usually required which corresponds to 100 to 200 numerical resolutions of the direct and adjoint equations. Additionally, the nonlinearity of the problem means that convergence to a global minimum is not guaranteed. In this chapter we use a quadratic programming approach which eliminates both issues. Indeed, since the cost function is quadratic, this problem can be solved as a quadratic program provided the constraints are linear. Another novelty in this work is the use of drifters for the estimation of boundary conditions while the other articles previously mentioned rely on fixed measurement stations. Note also that unlike adjoint based optimisation, the quadratic programming technique used in the present chapter does not require the definition and resolution of an adjoint (backward) problem.

This chapter is organised as follows: we start by presenting the basics of tidal channels and we state the problem solved; we then use standard data assimilation terminology to develop the estimation algorithm; twin experiments settings are later described before the results for estimation and prediction of tidal flow are presented and analysed.

3.2 Tidal channels

Tidal channels are bodies of water in which periodic changes in the water level and velocity field occur under the influence of tides, along the dominant tidal frequencies such as the K1 tide generated by the Sun, the M2 tide generated by the Moon or the MK3 shallow water tide created by the nonlinear interaction

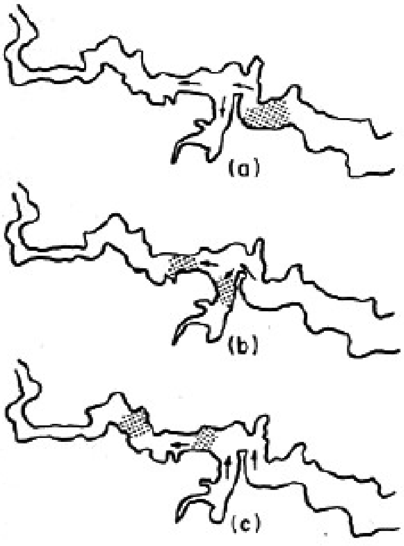

Figure 3.1: Phase effect in a branching channel [48]. (a) A cloud of tracer being carried upstream on a flooding tide. (b) At high water some of the particles are trapped in the branch. (c) During the early stages of the receding tide the flow in the main channel is still upstream. The particles trapped in the branch reenter the main channel, but are separated from their previous neighbours.

between the K1 and M2 tides resulting from the bottom friction. In channel networks, tidal trapping [48] describes the effects of side embayments and small branching channels which are enhanced by tidal action, as first noted in [138]. An example of tidal trapping can be seen in Figure 3.1 which represents a main channel with a number of side branches. Typically, the tidal elevations and velocities in the main channel are not in phase, with high water occurring before high slack tide and low water before low slack tide, due to a higher flow momentum in the main channel with respect to the side channel. In the former, the current will continue to flow against an opposing pressure gradient while in the latter the current direction changes as the water level begins to drop. Figure 3.1(a) shows a cloud of tracer particles being carried upstream by a flooding tide. Some of the particles go into the side channel while some continue upstream in the main channel as can be seen in Figure 3.1(b). As the water level begins to decrease, the particles in the side channel return into the main channel but separated from the rest of the cloud (Figure 3.1(c)). To summarise, tidal trapping occurs due to phase effects, with the flow directions changing at different times in the deeper and shallower branches. The implications for transport and dispersion are profound and tidal trapping is likely the dominant dispersion mechanism in such systems.

The flow in such a channel can be modelled using two dimensional shallow water equations such as [161] and as presented earlier in equations (2.11), (2.12), (2.13):

$$u_t + uu_x + vu_y + gh_x + gb_x - F_x = 0 \tag{3.1}$$

$$v_t + uv_x + vv_y + gh_y + gb_y - F_y = 0 \tag{3.2}$$

$$h_t + uh_x + vh_y + hu_x + hv_y = 0 \tag{3.3}$$

where (u, v, h) represent the velocity components and water depth, b the bottom elevation, g the acceleration of gravity and (F_x, F_y) the viscous forces term.

3.2.1 Problem description

An accurate simulation of tidal trapping relies on a prediction of the phase differences which is usually estimated according to historical data sets. In this article, our goal is to estimate open boundary conditions, more precisely the amplitudes and phases corresponding to the main four to eight tidal modes using velocity and position measurements provided by a number of drifters which are released on a portion of the Sacramento river located in the Sacramento-San Joaquin Delta in California. While the model used for the data assimilation problem is two dimensional, the estimation of the boundary conditions is performed on a one dimensional model which yields substantial benefits with respect to the size of the problem solved and therefore the computational time. The use of a one-dimensional estimation of the open boundary conditions is justified by the rapid lateral adjustment time of the free surface (less than 1 min); furthermore, the velocity profile can be reconstructed using interpolation from the average velocity in the cross-section. For example, in the case of the software package TELEMAC 2D [72] used in this work, the velocity profile is reconstructed with the assumption that the velocity is proportional to the square root of the depth along the cross section.

The algorithm proposed by this article consists in minimising the difference between measured velocity at the location of the drifter, and velocity estimated by the model. The constraints are the constitutive equations for the water, discretised in a linear manner. The decision variables, i.e. the variables with respect to which the optimisation is run are the upstream boundary conditions in the form of the coefficients of the dominant tidal modes. The problem thus consists in identifying the proper forcing at the upstream boundary of the domain, using Lagrangian sensing only.

In the next section we present the cost function to be minimised using the standard formulation of variational data assimilation, the one dimensional shallow water equations and linearised numerical scheme used as constraints.

3.3 Estimation algorithm

3.3.1 Notations

We start this section by defining the variables which will appear in the rest of this chapter. As in the previous chapter, we employ the traditional notations of variational data assimilation, set forth in [78].

X_n : vector of discretised state variables, namely the velocity component u and the water height h for each mesh point at a time instant t_n,

Y_n : vector of observed variables, namely the velocity components u and (potentially) the water height h for some mesh points at a time instant t_n,

$\mathbf{R}_n$: covariance matrix of the observation error (difference between the value of the state variables and observed variables at each mesh point),

H_n : $H_n = h_n^o \circ h^I$ is the observation operator; the operator h_n^o projects the space into the observation subspace. The operator h^I is the interpolation function. In general H_n is nonlinear although we manage to use a linear operator in our case by using the a posteriori knowledge of the position of the measurements, therefore encoded as a time dependent observation matrix.

We minimise the following cost function:

$$\mathcal{J}^o(U_{\text{up}}) = \sum_{n=0}^{n}(Y_n^o - H_n[X_n])^T\mathbf{R}_n^{-1}(Y_n^o - H_n[X_n]) \tag{3.4}$$

with respect to the boundary conditions U_{up} upstream, using as constraints the one dimensional linear discretised shallow water equations presented next.

3.3.2 One-dimensional numerical scheme

We choose to use a one dimensional linearised shallow water model to estimate the upstream boundary condition, and then use this as input in a nonlinear two dimensional solver which will generate a simulation of the flow taking into account the full geometry of the river. Compared to implementing a two dimensional shallow water model, this method presents the added advantage of greatly reducing the size of the quadratic program (by a factor at least 10 in our case) therefore allowing us to run the algorithm over extended time horizons (50 hours in the present case) for an acceptable computational cost (a few seconds on a desktop computer). Indeed for the example that will be considered in the following, the quadratic program has about 25,000 variables and is solved almost instantaneously by barrier methods. However, if we were to use a two-dimensional approach, the size of the problem would increase to at least 250,000 variables which exceeds the capacities of standard quadratic programming solvers.

The nonlinear one-dimensional shallow water equations can be written as:

$$\frac{\partial H}{\partial t} + \frac{\partial HU}{\partial x} = 0 \tag{3.5}$$

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} + g\frac{\partial H}{\partial x} + g\frac{\partial b}{\partial x} + g\frac{U|U|}{C^2 H} = 0 \tag{3.6}$$

where H is the water depth, b the one dimensionally averaged bottom elevation, U the velocity, and C the Chézy friction coefficient related to the Manning coefficient m (presented later for the two dimensional model) by: $C = \frac{H^{\frac{2}{3}}}{m}$.

We replace the friction term from equation (3.6) by an empirical drag coefficient $r = g\frac{|U_0|}{C^2 H_0}$ and linearise equations (3.5) and (3.6) in the case of small perturbations around a nominal flow $(U_0(x,t), H_0(x,t))$, with $U = u + U_0$, $H = h + H_0$. We then neglect the derivatives of the nominal variables (see for example

[163]) to obtain:

$$\frac{\partial h}{\partial t} + H_0(x,t)\frac{\partial u}{\partial x} + U_0(x,t)\frac{\partial h}{\partial x} = 0 \tag{3.7}$$

$$\frac{\partial u}{\partial t} + U_0(x,t)\frac{\partial u}{\partial x} + g\frac{\partial h}{\partial x} + g\frac{\partial b}{\partial x} + ru = 0 \tag{3.8}$$

The Preissmann scheme (see [130, 158]) is applied to these equations yielding:

$$\begin{aligned} h_{i+1}^{n+1} - h_{j+1}^{n} + h_{i}^{n+1} - h_{i}^{n} &= -H_0\frac{\Delta t}{\Delta x}(u_{i+1}^{n+1} - u_{i}^{n+1} + u_{i+1}^{n} - u_{i}^{n}) \\ &- U_0\frac{\Delta t}{\Delta x}(h_{i+1}^{n+1} - h_{i}^{n+1} + h_{i+1}^{n} - h_{i}^{n}) \end{aligned} \tag{3.9}$$

$$\begin{aligned} u_{i+1}^{n+1} - u_{i+1}^{n} + u_{i}^{n+1} - u_{i}^{n} &= -U_0\frac{\Delta t}{\Delta x}(u_{i+1}^{n+1} - u_{i}^{n+1} + u_{i+1}^{n} - u_{i}^{n}) \\ &- g\frac{\Delta t}{\Delta x}(h_{i+1}^{n+1} - h_{i}^{n+1} + h_{i+1}^{n} - h_{i}^{n}) - ru_{i}^{n} - g\frac{2\Delta t}{\Delta x}(b_{i+1} - b_{i}) \end{aligned} \tag{3.10}$$

where i denotes the space step and n the time step. This is an implicit scheme which allows us to choose fewer time steps than in the explicit case. This scheme was used as constraints in a quadratic program minimising the L^2 norm of the difference between the observed velocity and water height and the variables obtained through the Preissmann scheme. This algorithm will be used in the following to estimate boundary conditions for tidal flows, running a simulation for 50 hours and attempting to estimate up to eight modes.

3.3.3 Modelling assumptions

We can now pose the estimation problem as a quadratic optimisation problem. Using simulated drifter trajectories, we estimate the amplitude and phase associated with four to eight dominant frequencies and the signal thus obtained is compared to the original flow. The coefficients representing the amplitudes of the main modes are estimated using a data assimilation algorithm which consists in minimising the cost function (3.4) with respect to the constraints given by the Preissmann scheme detailed earlier. The boundary condition is specified as:

$$u_{\text{up}}(t) = a_0 + \sum_{k=1}^{N_{\text{modes}}} a_k\cos(\omega_k t) + b_k\sin(\omega_k t) \tag{3.11}$$

where $\omega_k = \frac{2\pi}{T_k}$ is the frequency associated to one of the periods identified in the spectral analysis of the data. The coefficients (a_k, b_k) fully characterise the boundary conditions and the data assimilation problem is posed in a way which makes them decision variables.

3.3.4 Optimisation program

The control variables are the coefficients (a_k, b_k). The discretised dynamics of the flow are encoded in the form of linear constraints through the Preissmann scheme developed above. As in the previous chapter, while the numerical scheme is implicit, i.e. it can be written in the form $EZ_{n+1} = AZ_n + Bu_n$ (where Bu_n comes from the bottom elevation term in the one dimensional shallow water equations), the quadratic program can incorporate the constraints at no further cost or complication. Note that this is a major advantage of the quadratic programming approach over techniques such as Kalman filtering, which would require an explicit inversion of the Preissmann scheme, so that it could be expressed as a linear time invariant system $Z_{n+1} = AZ_n + Bu_n$. We concatenate the vectors Z_n for all n into a single vector called X, and abbreviate the dynamics constraints by $CX = b$, where this equation encodes the discretised flow equations. The search space $\{(a_k, b_k), 1 \leqslant k \leqslant N_{\text{modes}}\}$ from which all other quantities depend is allowed to evolve in a set of feasible initial conditions dictated by flow physics. Because (u_i^n, h_i^n) cannot take arbitrary values, the space in which X evolves can also be restricted to increase the speed of convergence of the method. These two constraints are encoded in the form of an inequality, $GX \leqslant h$. Finally, using the previous variable definitions, equation (3.4) can be written as:

$$\begin{array}{ll} \text{minimise} & J(X_{\text{up}}) = \frac{1}{2}X^T P X + q^T X + r \\ \text{subject to} & \qquad GX \leqslant h \\ & \qquad AX = b \end{array} \tag{3.12}$$

The constraints are thus as follows:

- The components (u_i^n, h_i^n) of X must verify the Preissmann scheme at all space and time steps.
- State variables have to satisfy boundary and initial conditions (X_{up} represents the boundary conditions

upstream and is related to the state of the system X through the linearised Preissmann scheme).

- Inequality constraints need to keep (u_i^n, h_i^n) in a realistic set (in our case positivity of the water height and bound on the absolute value of the velocity $|u| < 10$ m/s).

3.4 Numerical experiment setting

A standard benchmark for the validation of data assimilation algorithms is the twin experiment which consists in comparing the true state of the system with the assimilated state. For this, a *forward* simulation is run from time t_0 to time T using a two dimensional nonlinear shallow water model, yielding the so-called *true state* at every time instant between t_0 and T. At a chosen time t_1, drifters are released from the upstream end and their trajectories are simulated using a Runge-Kutta method and the vector field provided by the nonlinear shallow water forward simulation; then, a data assimilation process is started and the estimated boundary conditions are computed and compared with the true boundary conditions. In the present case, the true state of the river and the position of the drifters are generated using the software TELEMAC 2D and we realise several experiments with and without tidal flow reversal in order to test the influence of the number of modes estimated on the accuracy of the algorithm. The boundary conditions for the two dimensional forward simulation are computed using the Delta Simulation Model II (DSM2) [9] (Figure 3.2), a one dimensional model of the Sacramento-San Joaquin Delta that provides discharge and surface elevation at various locations every hour.

3.4.1 Description of the experimental protocol

As in the previous chapter, the domain considered is a portion of the Sacramento River comprised between the Delta Cross Channel and the Georgiana Slough in the Sacramento-San Joaquin Delta (Figure 2.4).

The boundary conditions are computed using the Delta Simulation Model II (DSM2) (Figure 3.2), a model of the San Francisco Bay and Sacramento-San Joaquin Delta that provides discharge and surface

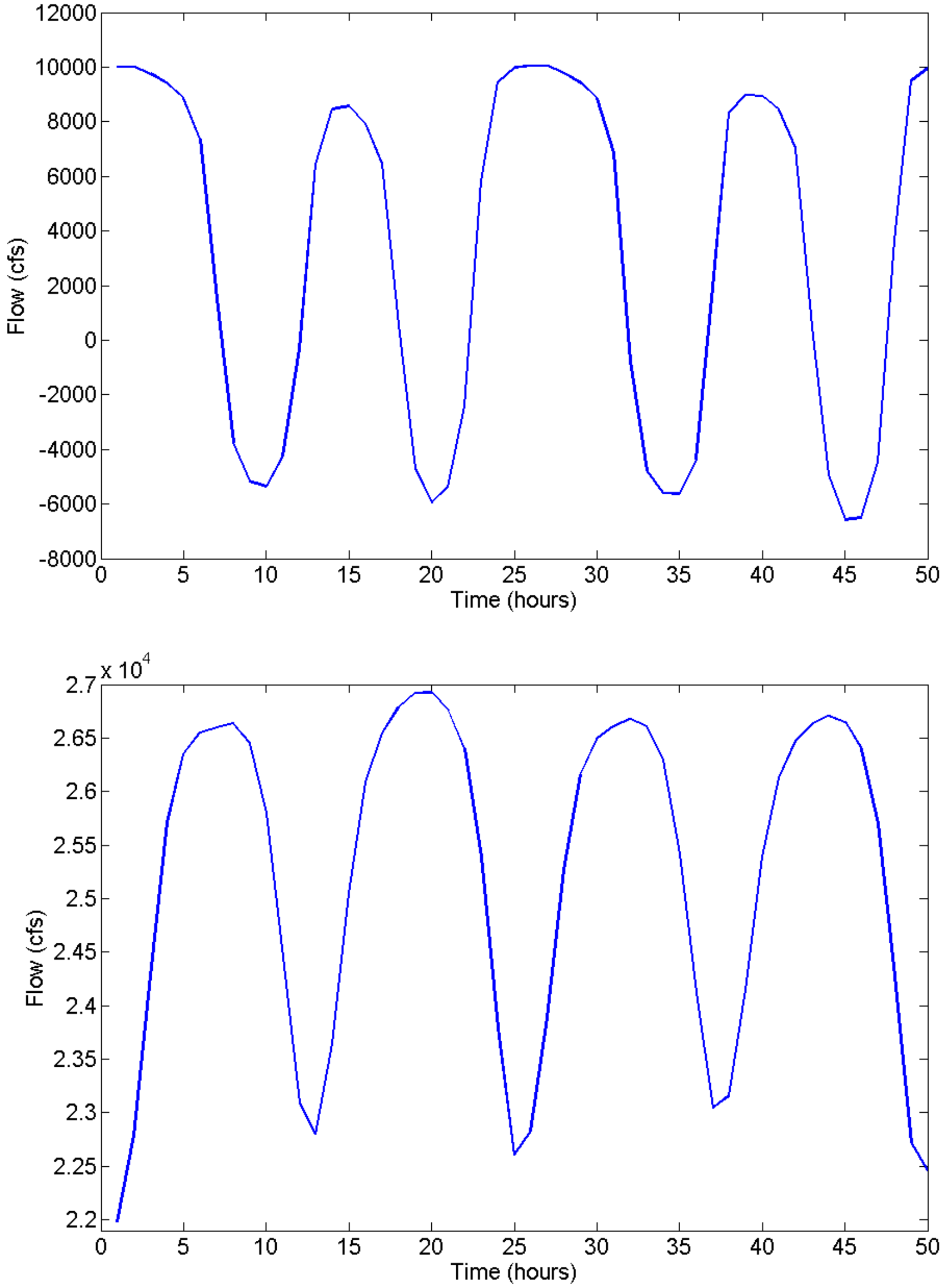

Figure 3.2: Hourly evolution of the flow over a 50 hour period in October 2006 (top) and March 2006 (bottom) used as boundary condition in the data assimilation algorithm.

elevation at various locations every hour.

The governing hydrodynamic equations used in the simulation run with TELEMAC 2D are a version of the two dimensional shallow water equations incorporating friction forces and a specific turbulence model:

$$u_t + uu_x + vu_y + gh_x = -gb_x + F_x + \frac{1}{h}\nabla \cdot (h\nu_t \nabla u) \tag{3.13}$$

$$v_t + uv_x + vv_y + gh_y = -gb_y + F_y + \frac{1}{h}\nabla \cdot (h\nu_t \nabla v) \tag{3.14}$$

$$h_t + (hu)_x + (hv)_y = 0 \tag{3.15}$$

The friction forces are given by the following Manning law:

$$F_x = -\frac{1}{\cos\alpha}\frac{gm^2}{h^{4/3}} u\sqrt{u^2+v^2} \tag{3.16}$$

$$F_y = -\frac{1}{\cos\alpha}\frac{gm^2}{h^{4/3}} v\sqrt{u^2+v^2} \tag{3.17}$$

where h is the total depth of water, (u, v) is the velocity field, g is the acceleration of gravity, b is the bottom elevation, ν_t is the coefficient of turbulence diffusion obeying the so called k-epsilon model (see [133] for more details), $\alpha = \alpha(x, y)$ is the slope of the bottom, and m is the Manning coefficient (usually denoted by n). In the present case, the Manning coefficient is chosen to be constant in time and space and equal to 0.02, corresponding to a highly frictional bottom.

3.4.2 Initial and boundary conditions

The boundary and initial conditions of equations (3.13), (3.14), (3.15) are given by

$$u(x,y,t)\big|_{\partial\Omega_{\text{land}}} = 0, \quad v(x,y,t)\big|_{\partial\Omega_{\text{land}}} = 0, \tag{3.18}$$

$$(u(x,y,t), v(x,y,t))\big|_{\partial\Omega_{\text{upstream}}} = f(x,y,t), \tag{3.19}$$

$$\eta(x,y,t)\big|_{\partial\Omega_{\text{downstream}}} = g(x,y,t), \tag{3.20}$$

$$u(x,y,0) = u_0, \quad v(x,y,0) = v_0, \quad h(x,y,0) = h_0, \tag{3.21}$$

where $\partial\Omega$ represents the boundaries of our computational domain and f, g are known functions (in the present case obtained from experimental measurements and the DSM2 numerical tool).

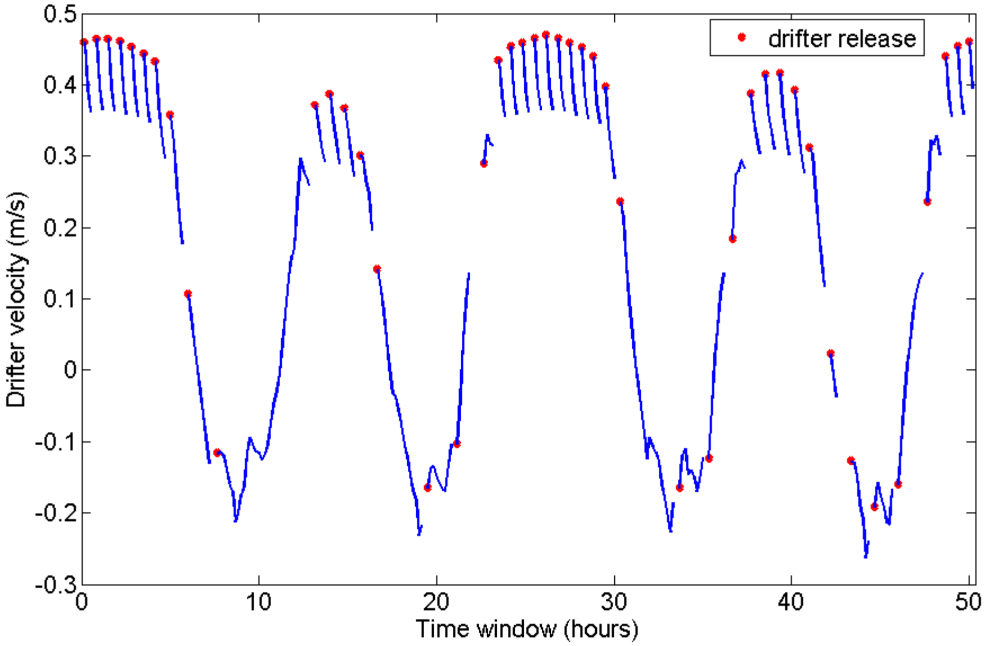

Figure 3.3: Velocity of the drifter during the simulation in the case of tidal inversion. The dots indicate the release of a new drifter.

3.4.3 Lagrangian drifters

In addition to the equations (3.13), (3.14) and (3.15) describing the flow dynamics, we model the deployed drifters as passive Lagrangian tracers. Hence, the drifters move with the local fluid velocity, obeying the following equations:

$$\frac{dx_D(t)}{dt} = u[x_D(t), y_D(t), t], \tag{3.22}$$

$$\frac{dy_D(t)}{dt} = v[x_D(t), y_D(t), t], \tag{3.23}$$

with the initial conditions

$$x_D(0) = x_{D,0}, \quad y_D(0) = y_{D,0}. \tag{3.24}$$

Equations (3.22) and (3.23) have to be understood for each x_D and y_D corresponding to a specific drifter.

The simulation runs for two and a half hours before the drifters are released so that a stable state is reached. A simulation is run using TELEMAC 2D during a 50 hour period in October 2006, during which tidal flow reversal occurs and during another 50 hour period in March 2006 in which flow is unidirectional (due to freshwater flow). The release of drifters is simulated during this period (see Figure 3.3) with a drifter being released from one end or the other depending on the direction of the flow since a tidal reversal occurs eight times during the simulation (drifter releases are indicated by red dots on the graph). The nearly vertical lines at the top of each sinusoid correspond to the velocity of the drifter along the river, the velocity decreasing as the drifter floats down the channel. The positions and velocities of the drifters are recorded every 10 minutes and used in the data assimilation algorithm. Note that for these twin experiments, the only observations from the channel flow are those given by the drifters. A drifter is released from the upstream end whenever the previous drifter exits the domain so that there is one drifter in the model domain at all time during the simulation.

3.5 Results and analysis

3.5.1 Tidal flow in the Sacramento river

Eight years of hourly data coming from a DSM2 simulation of the Delta were available for the study. The spectral analysis of the DSM2 data for the month of October 2006 shows eight dominant frequencies (see Figure 3.4). A similar work is done with DSM2 data from March 2006, a situation in which there is no tidal reversal (see Figure 3.2). The spectral analysis of the flow during the month of March 2006 shows a highly dominant nominal flow with minimal amplitudes for the other modes. These results and a description of each tidal mode are given in tables 3.1 and 3.2.

Symbol	Species	Period (hr)
O1	Lunar diurnal	25.81933871
K1	Lunar diurnal	23.93447213
N2	Larger lunar elliptic semidiurnal	12.65834751
M2	Principal lunar semidiurnal	12.4206012
MK3	Shallow water terdiurnal	8.177140247
M4	Shallow water overtides of principal lunar	6.210300601
M6	Shallow water overtides of principal lunar	4.140200401
M8	Shallow water eighth diurnal	3.105150301

Table 3.1: Description and period of the main tidal modes encountered in the Sacramento-San Joaquin Delta.

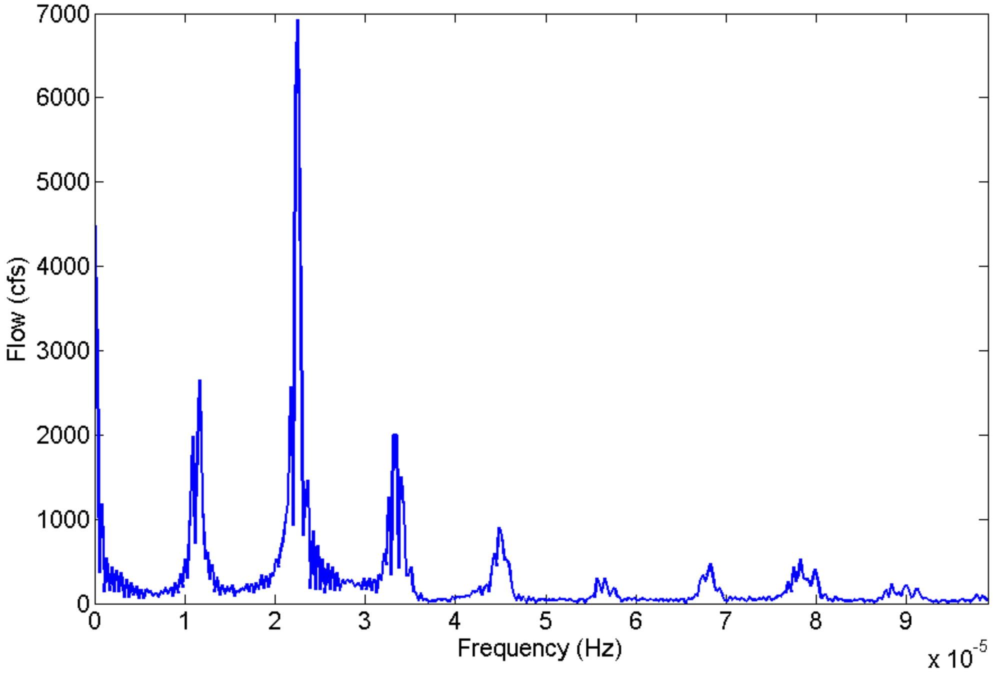

Figure 3.4: Spectral decomposition of the flow in the Sacramento River using DSM2 data for the month of October 2006.

Tide	Amplitude (October 2006)	Amplitude (March 2006)
Mean flow	2400 cfs	25215 cfs
O1	1983 cfs	819 cfs
K1	2646 cfs	6036 cfs
N2	2574 cfs	747 cfs
M2	6926 cfs	3110 cfs
MK3	2002 cfs	2476 cfs
M4	908 cfs	1508 cfs
M6	468 cfs	115 cfs
M8	527 cfs	87 cfs

Table 3.2: Amplitudes of the main tidal modes in October 2006 (tidal inversion) and March 2006 (no inversion).

3.5.2 Estimation of flow

Either four or eight modes are estimated using the data assimilation algorithm from which the flow is reconstructed and compared to the original DSM2 data. The relative *Root Mean Square error* (RMS) given by:

$$E = \left(\frac{\sum_{n=1}^{50} |u_{\text{up}}^{t}(n) - u_{\text{up}}^{a}(n)|^2}{\sum_{n=1}^{50} |u_{\text{up}}^{t}(n)|^2} \right)^{\frac{1}{2}} \tag{3.25}$$

where $u_{\text{up}}^{t}(n)$ and $u_{\text{up}}^{a}(n)$ represent respectively the true one dimensional boundary conditions (DSM2 data) and the assimilated one dimensional boundary conditions at time step n. The quadratic program is solved with CPLEX, a large scale quadratic programming solver, using a barrier method. Depending on whether four or eight modes are used, the relative RMS error of the flow obtained is respectively 15.6% and 7.2%. The comparative plots of the reconstructed flow and the true flow are given in Figure 3.5.

A twin experiment similar to the one presented above realised in March 2006 yields a relative RMS error of 7.2% and 4.6% for four and eight modes respectively (see Figure 3.6). The errors are smaller than in the October 2006 case which may be explained by the fact that the linear model used in the data assimilation algorithm is a better fit when there is no flow reversal. Indeed, during the flow reversal around slack water, nonlinear effects are likely to be more pronounced given that the overall flow is very low.

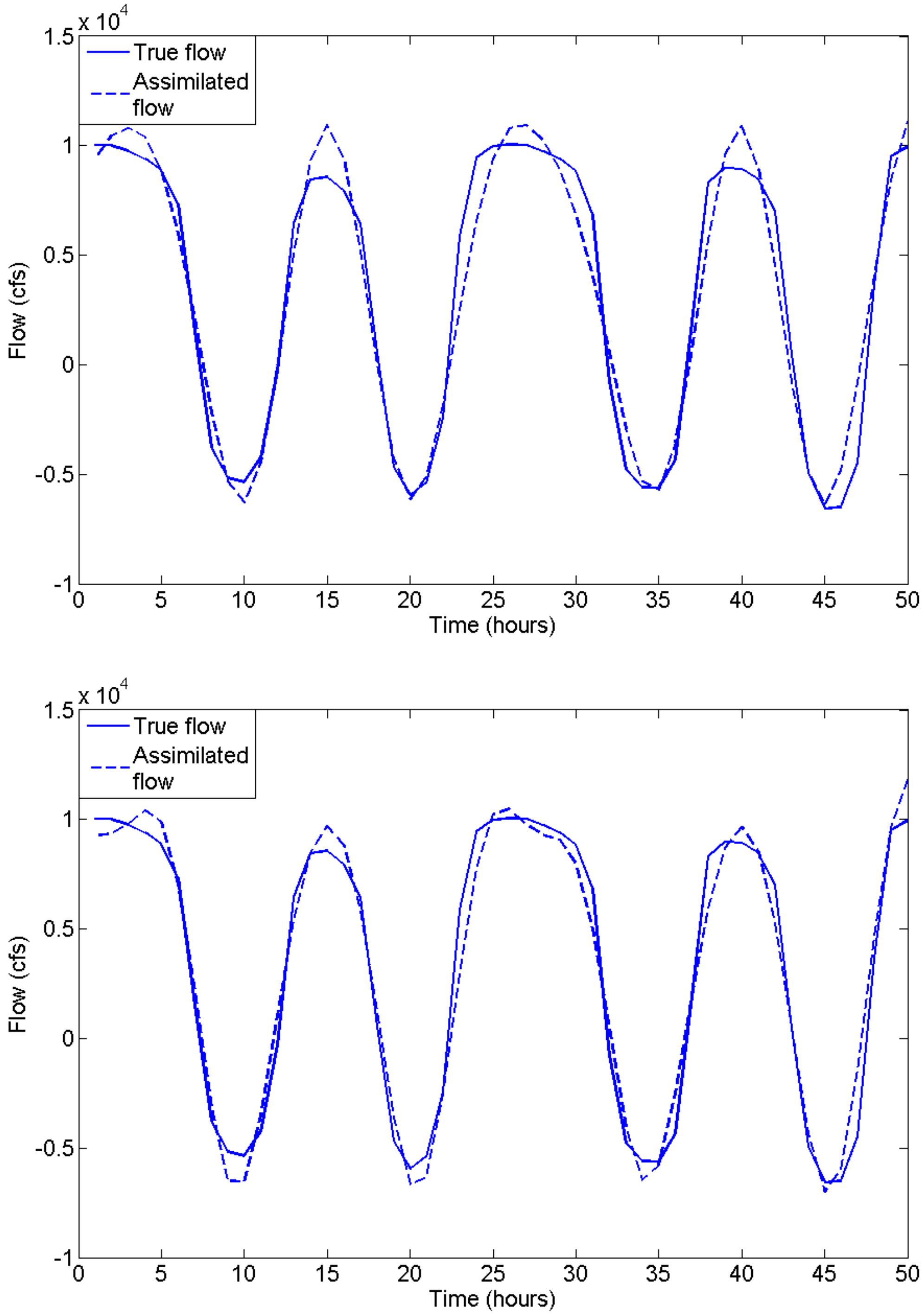

Figure 3.5: Hourly evolution of the true and assimilated flow using four modes (top) or eight modes (bottom) over a 50 hour period with tidal inversion (October 2006).

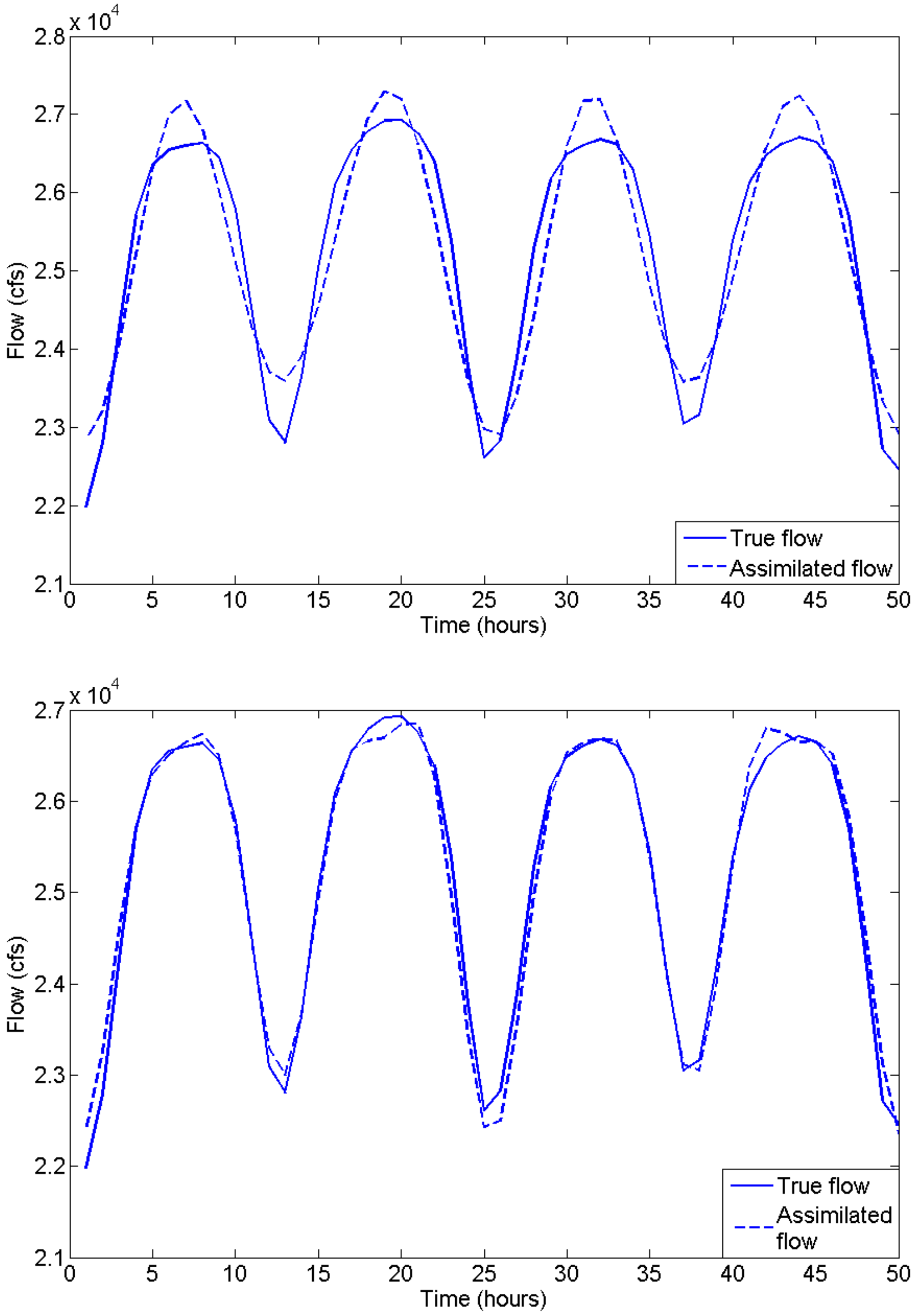

Figure 3.6: Hourly evolution of the true and assimilated flow using four modes (top) or eight modes (bottom) over a 50 hour period with no tidal inversion (March 2006).

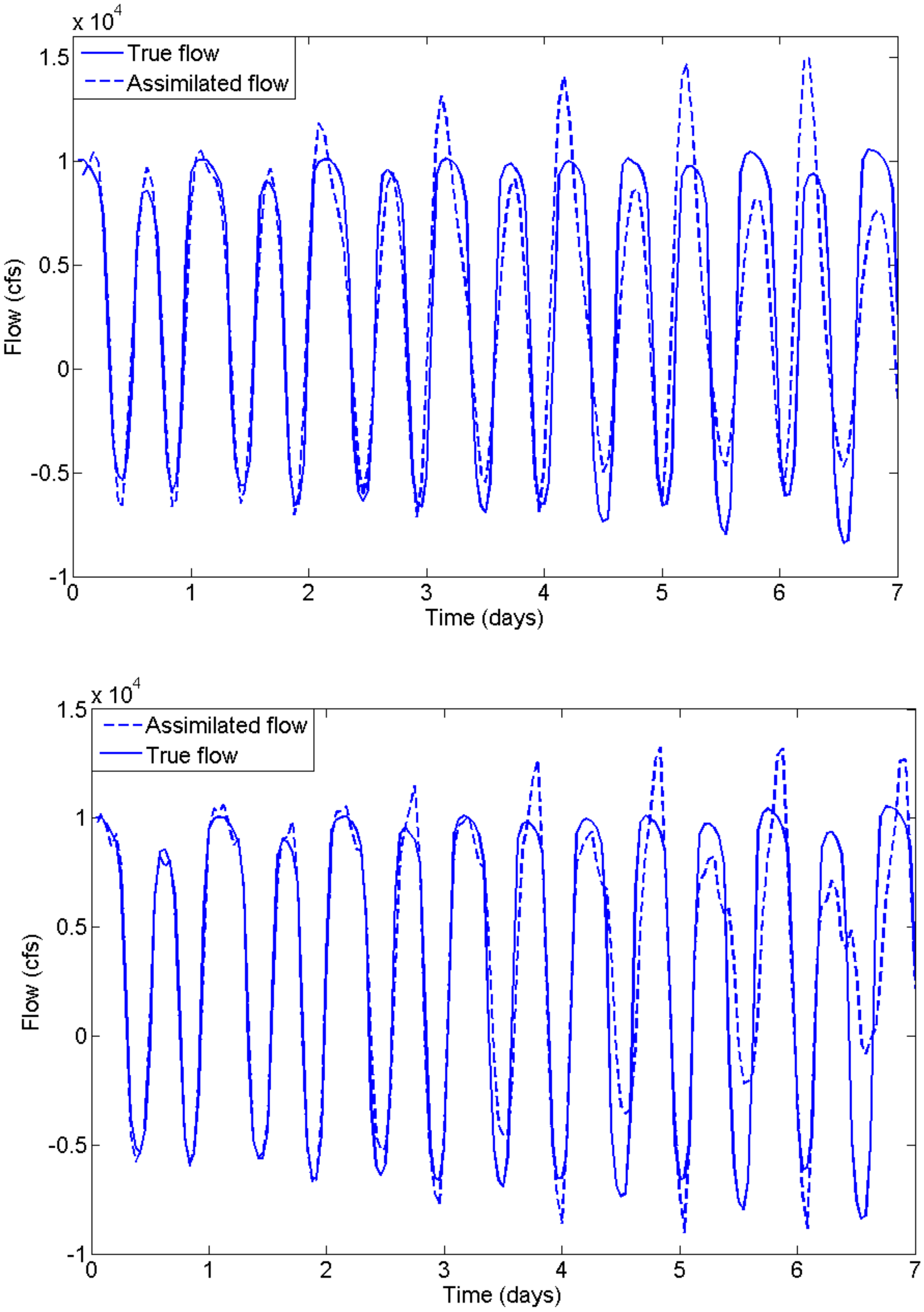

Figure 3.7: Hourly evolution of the true and assimilated flow using four modes (top) or eight modes (bottom) over a one week period with tidal inversion (October 2006).

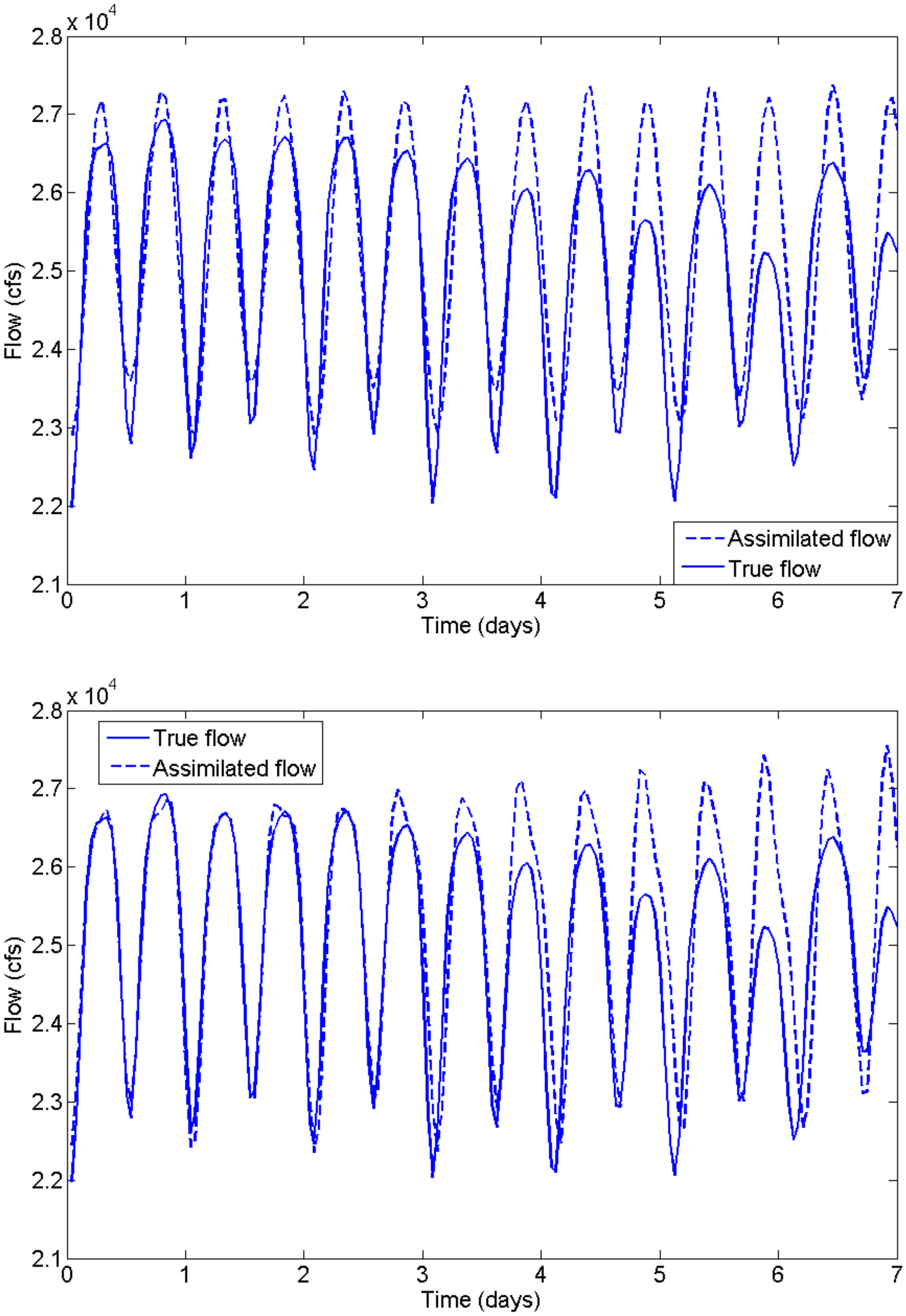

Figure 3.8: Hourly evolution of the true and assimilated flow using four modes (top) or eight modes (bottom) over a one week hour period with no tidal inversion (March 2006).

Modes	Error (October 2006)	Error (March 2006)
Mean, K1, M2, MK3, M4	15.6%	7.8%
Mean, K1, M2, MK3, M4, M6, M8, O1, N2	7.2%	4.6%

Table 3.3: Relative RMS error of the estimated flow for four and eight modes in October 2006 (tidal inversion) and March 2006 (no inversion).

3.5.3 Prediction of flow evolution

The next experiment aims to evaluate the predictive capabilities of this data assimilation algorithm. Namely, once the amplitudes corresponding to the dominant frequencies have been estimated, we try to reconstruct the signal for a one week period and compare it with the DSM2 data. For the first data set from October 2006 which presents a flow reversal, the relative RMS error is 39.7% and 37.3% for four and eight modes respectively (see Figure 3.7). The increase of the flow in the last few days of the week considered is not predicted by either reconstructed flow hence the similar relative RMS error.

In the case of the data set from March 2006, the relative RMS error is 30.6% and 28.4% for four and eight modes respectively (see Figure 3.8). Once again, the decrease of the flow in the second half of the period considered is not predicted by either of the reconstructed flows which explains the almost identical error for four and eight modes signals.

When eight modes are considered, the results presented in Figure 3.9 show that adding the larger lunar elliptic semidiurnal tide N2 does however reduce the error at the beginning of the prediction period (Figure 3.9) but the prediction error remains above 30% over a one week period. It would appear that events other than tidal forcing (such as reservoir release or local precipitation) influence the flow in the Sacramento Delta; this is confirmed by the comparison between the spectra of the flow three weeks before and three weeks after the experiment period (Figure 3.10) which shows significant differences in the amplitude of the mean flow (4200 cfs before and 2830 cfs after the experiment) and the M2 tide (3105 cfs before and 3800 cfs after) thereby limiting the predictive capability of an algorithm based on identifying tidal components over a short period of time. In the case without tidal reversal, the amplitudes of the O1, M6, M8 and N2 tides are too small compared to the mean flow and other dominant tides for their inclusion to have a significant effect

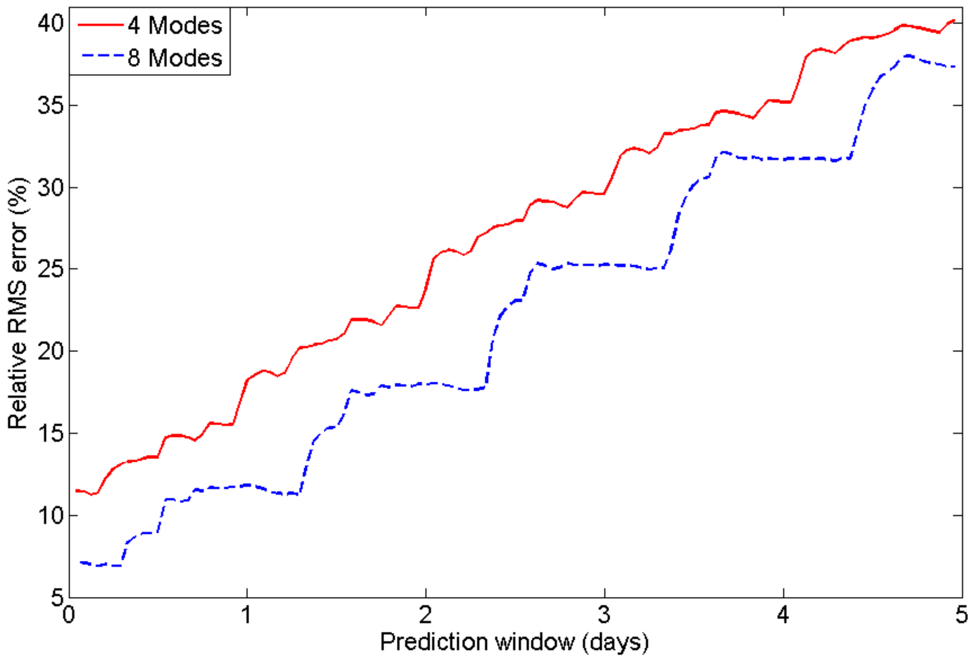

Figure 3.9: Evolution of the relative RMS error after of the observation period with four and eight modes considered.

on the estimation error. The preceding results are summarised in table 3.4.

Modes	50 hours		4 days		1 week	
	(10/06)	(03/06)	(10/06)	(03/06)	(10/06)	(03/06)
4 modes	15.6%	7.8%	22.6%	15.1%	39.7%	30.6%
8 modes	7.2%	4.6%	17.9%	11.2%	37.3%	28.4%

Table 3.4: Relative RMS error of the estimated flow for four and eight modes in October 2006 (tidal inversion) and March 2006 (no inversion) after 50 hours, four days and one week.

3.6 Conclusion

In this chapter, a novel data assimilation algorithm was developed applied to estimate open boundary conditions for a tidal channel in the Sacramento San-Joaquin Delta. Its performance was assessed using twin experiments. Given that tidal forcing is dominant, the focus was placed on identifying the main frequencies and then estimating their corresponding amplitudes. Using 50 hours of DSM2 data, the flow in the

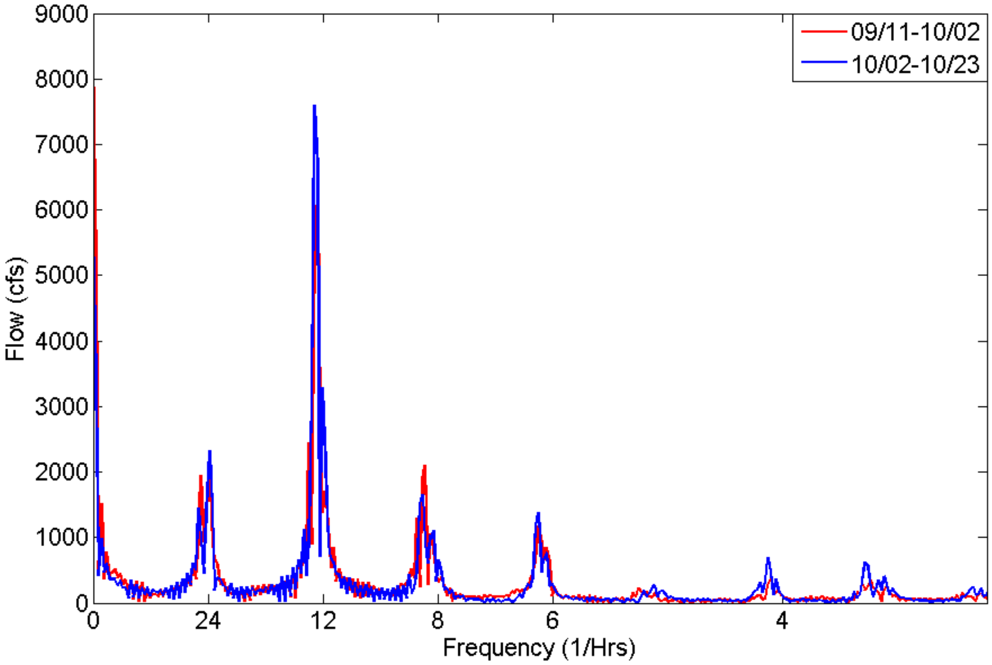

Figure 3.10: Comparison of the spectra of the flow in the Sacramento River three weeks before the October 2006 experiment and three weeks after. Note the significant difference of amplitude in the mean flow and M2 tide.

Sacramento River was simulated with TELEMAC 2D and drifter trajectories were generated both in a reversing tidal flow and when the flow does not reverse. The velocities of the drifters along the trajectories were used to estimate the amplitude of the main four to eight modes and enabled us to generate a reconstructed flow which was compared to the original DSM2 data. The relative root mean square error varies from 15.6% to 4.6% depending on whether a flow reversal occurs and on the number of modes estimated. This shows that during the period of time that observations are available, the inverse model is able to effectively estimate the harmonic constants and reproduce the flows. It is important to note that we have not constrained the tidal amplitudes or phases in any way: the only tidal information added are the frequencies of the dominant modes. For them, the case we consider here is a stringent one on which there is only one drifter available; most Lagrangian measurements include a number of drifters which would improve the ability to estimate boundary conditions.

Considering projections in time beyond the observation period, the ability of the harmonics to track the true values is somewhat limited. An attempt was made to evaluate the predictive capabilities of the algorithm by comparing the reconstructed flow and DSM2 data over a period of one week. The prediction error was found to be between 22.6% and 11.2% after four days and 39.7% and 28.4% after one week. The relatively high value of the RMS error after one week appears to be due to the fact that tidal harmonics are limited in their ability to predict flows in this reach of the Sacramento Delta, where unsteady freshwater flows influence the evolution. Indeed, a comparison of the tidal spectra three weeks before and after the experiment reveals significant differences in the amplitudes of the main tidal modes which limit the predictive capability of an algorithm based on identifying tidal components over a short period of time (50 hours).

Chapter 4

Highway traffic flow modelling, simulation and control

4.1 Introduction

This work is motivated by recent research efforts which investigate the problem of simulating, estimating, and controlling highway traffic flow with metering strategies that can be applied at the on-ramps of the highway (see in particular [150] and references therein). Several models have been developed over the years [90], some of which were based on a microscopic approach analogous to the Boltzmann equation for gases [70, 131, 132], while others were macroscopic conservation laws similar to fluid dynamics equations [14, 103, 134]. One of the macroscopic approaches, called *Lighthill-Whitham-Richards* (LWR) theory, will be used in this chapter; it relies on a scalar hyperbolic conservation law, with a concave flux function. So far, very few articles have tackled the problem of boundary control of scalar conservation laws in bounded domains in an explicit manner directly applicable for engineering purposes. Unlike the viscous Burgers equation, which has been the focus of numerous ongoing studies, very few results exist for the inviscid Burgers equation, which is traditionally used as a model problem for hyperbolic conservation laws, although differential flatness [128] and Lyapunov theory [92] have been explored and appear as promising directions of future investigations. The proper notion of weak solution for the LWR partial differential equation (PDE), called *entropy solution*, was first defined by Oleinik [124] in 1957. Even though this work was known to the traffic community, it does not (as far as we know) appear explicitly in the transportation literature before 1990 with the work of Ansorge [10]. The entropy solution has since been acknowledged as the proper weak solution to the LWR PDE [40] for traffic models. Unfortunately the work of Oleinik in its initial form [124] does not hold for bounded domains, i.e. it would only work for infinitely long highways with no on-ramps or off-ramps. Bounded domains, i.e. highways of finite length (required to model on and off-ramps) imply the use of boundary conditions, for which the existence and uniqueness of a weak solution is not straightforward. The first result of existence and uniqueness of a weak solution of the LWR PDE in the presence of boundary conditions follows from the work of Bardos, Le Roux and Nédélec [17], in the more general context of a first order quasi-linear PDE on a bounded open set of $\mathbb{R}^n$. In particular, they introduce a weak formulation of the boundary conditions for which the initial-boundary value problem is well-posed.

We begin by explaining that in general, one cannot expect the boundary conditions to be fulfilled point-wise a.e. (almost everywhere) and we provide several examples to illustrate this fact. We then turn to the specific case of highway traffic flow, for which we are able to state a simplified weak formulation of the boundary conditions, and prove the existence and uniqueness of a weak solution to the LWR PDE, the former resulting from the convergence of the associated Godunov scheme to the entropy solution of the PDE. This represents a major improvement from the existing traffic engineering literature, where boundary conditions are expected to be fulfilled point-wise and therefore existence of a solution and convergence of the numerical schemes to this solution are not guaranteed. We illustrate the applicability of the method and the numerical scheme developed in this work with a highway scenario, using data from the Interstate-80 highway, obtained from the Berkeley Highway Laboratory [95]. In particular, we show that the model is able to reproduce flow variations on the highway with a very good accuracy over 24 hours in both free flow and congestion modes. The last part is devoted to the boundary control of the LWR PDE and its application to an optimisation problem, in which boundary control is used to minimise travel time on a given stretch of highway.

4.1.1 Need for weak boundary conditions

This section shows three examples of the sort of trouble one runs into when prescribing the boundary conditions in the strong sense. Unfortunately, numerous articles, in particular in the transportation engineering community, solve a discrete version of this type of problems. Regardless of the numerical schemes used (Godunov [62], Jameson-Schmidt-Turkel [84, 85], Daganzo [39, 40]), if boundary conditions are not imposed in the weak sense, the numerical solution provided by these schemes is meaningless as far as the PDE problem is concerned. Indeed, while the numerical schemes listed above might still yield a numerical output, this numerical data would correspond to an initial boundary-value problem that does not have a solution in the first place (because the corresponding continuous problem is usually ill-posed). To sum up, boundary conditions may only be prescribed on the part of the boundary where the characteristics are incoming, that is entering the domain.

Example 1: advection equation. We start by considering the simple example where the propagation speed is a constant c,

$$\frac{\partial \rho}{\partial t} + c\frac{\partial \rho}{\partial x} = 0 \text{ for } (x,t) \in (a,b) \times (0,T). \tag{4.1}$$

In that case, one can clearly see that the boundary condition is either prescribed on the left $(x = a)$ if the speed c is positive or the right $(x = b)$ if the speed is negative. While finding the sign of the speed is quite simple in the linear case, this becomes more subtle when dealing with a nonlinear conservation law such as the LWR PDE as this sign is no longer constant.

Example 2: LWR PDE, shock wave back-propagation due to a bottleneck. For this example, we consider the LWR PDE with a Greenshields flux function [66]:

$$\frac{\partial \rho}{\partial t} + v\left(1 - \frac{2\rho}{\rho^*}\right)\frac{\partial \rho}{\partial x} = 0 \tag{4.2}$$

where $\rho = \rho(x,t)$ is the vehicle density on the highway, ρ^* is the *jam density* and v is the *free flow density* (see [39, 40] for more explanations on the interpretation of these parameters). We consider a road of length $L = 30$, $\rho^* = 4$ and $v = 1$ (dummy values), and an initial density profile given by $\rho_0(x) \triangleq \rho(x,0) = 2$ if $x \in [0,10]$, $\rho_0(x) \triangleq \rho(x,0) = 4$ if $x \in (10,20]$, $\rho_0(x) \triangleq \rho(x,0) = 1$ if $x > 20$. The highway might be bounded or unbounded on the right at $x = L = 30$ (it does not matter for our problem). We assume free

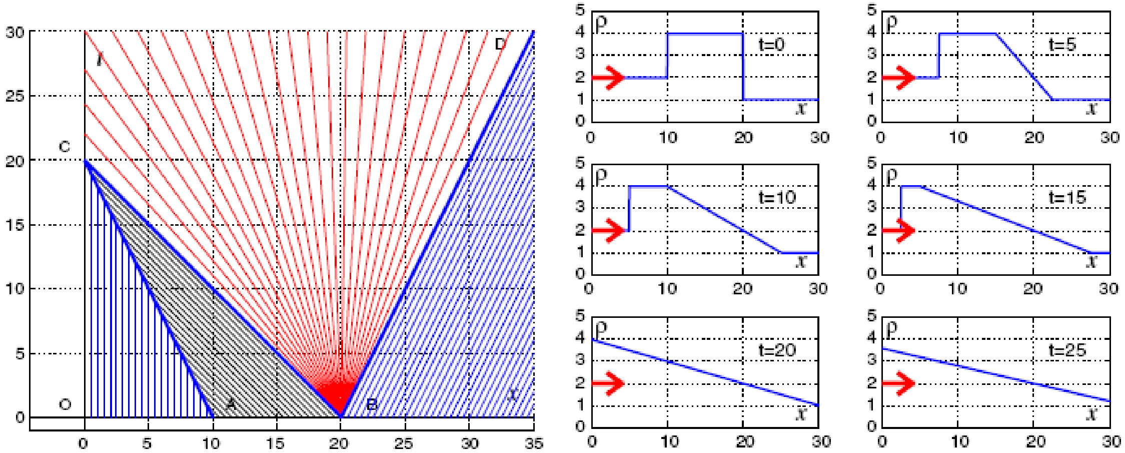

Figure 4.1: **Left:** Characteristics for the solution of the LWR PDE for Example 2. **Right:** corresponding value of the solution at successive times. The arrow represents the value of the input at $x = 0$, which becomes irrelevant for $t \geqslant 20$.

flow conditions at $x = L$, that we can control the inflow at $x = 0$, and we try to prescribe it point-wise, i.e. $\rho(0,t) = 2$ for all t (this corresponds to sending the maximum flow onto the highway). The solution to this problem can easily be computed by hand (for example by the method of characteristics, see Figure 4.1, left). The solution to this problem reads

$$\begin{cases} \rho(x,t) = 2 & \text{if} \quad t \leqslant 2(10-x) & \text{AC: shock} \\ \rho(x,t) = 4 & \text{if} \quad 2(10-x) \leqslant t \leqslant 20-x & \text{BC: left edge of expansion wave} \\ \rho(x,t) = 2(1-(x-20)/t) & \text{if} \quad t \geqslant \max\{20-x, 2(x-20)\} & \text{CBD: expansion wave} \\ \rho(x,t) = 1 & \text{if} \quad t \leqslant 2(x-20) & \text{BD:right edge of expansion wave} \end{cases} \tag{4.3}$$

As can be seen, $\lim_{x \to 0^+} \rho(x,t) = 2$ for $t \leqslant 20$ and $\lim_{x \to 0^+} \rho(x,t) = 2(1 + 20/t)$ for $t > 20$. Thus, the boundary condition $\rho(0,t) = 2$ is no longer verified as soon as $t \geqslant 20$. This phenomenon is crucial in traffic flow models: it represents the back-propagation of congestion (i.e. upstream). If the location $x = 0$ was the end of a link merging into the highway (that we could potentially control), the case when $\rho(0^+,t) > \rho^*$ is congested would correspond to a situation in which the upstream flow ($x = 0^-$) is imposed by the downstream flow ($x = 0^+$), i.e. the boundary condition on the left becomes irrelevant. When $\rho(0^+,t) < \rho^*$ is not congested, the boundary condition is relevant and can be imposed point-wise.

Example 3: Burgers equation. We now consider the inviscid Burgers equation on $(0,1) \times (0,T)$. If we try to prescribe strong boundary conditions at both ends, the problem becomes ill-posed. The Burgers equation reads:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0 \tag{4.4}$$

We set the initial value $u(x,0) = 1$, and the boundary conditions $u(0,t) = u(1,t) = 0$ on $[0,1]$. The solution of (4.4) with these boundary conditions is for $t < 1$:

$$\begin{cases} u(x,t) = \frac{x}{t} \quad \text{if} \quad x < t \quad \text{self similar expansion wave} \\ u(x,t) = 1 \quad \text{if} \quad x > t \quad \text{convection to the right with speed 1} \end{cases} \tag{4.5}$$

We notice that the boundary condition is not satisfied at $x = 1$. Since the data propagates at speed u, the characteristics are leaving $[0,1]$ at $x = 1$ while they stay in $[0,1]$ as a rarefaction wave at $x = 0$.

4.2 Scalar conservation laws

4.2.1 Proof of uniqueness

We consider a mixed initial-boundary value problem for a scalar conservation law on $(a,b)\times(0,T)$.

$$\frac{\partial \rho}{\partial t} + \frac{\partial q(\rho)}{\partial x} = 0 \tag{4.6}$$

with the boundary conditions

$$\rho(a,t) = \rho_a(t) \text{ and } \rho(b,t) = \rho_b(t) \text{ on } (0,T) \tag{4.7}$$

and the initial condition

$$\rho(x,0) = \rho_0(x) \text{ on } (a,b). \tag{4.8}$$

As usual with nonlinear conservation laws, in general there are no smooth solutions to this equation and we have to consider weak solutions (see for example [24], [43], [142]). In this chapter we use the space BV of functions of bounded variation which appears very often when dealing with conservation laws. A function of bounded variation is a function in L^1 whose gradient is a signed measure, the positive and negative parts of which have finite mass. We refer the intrigued readers to [4] for many more properties and applications of BV functions. Other valuable references on BV functions include [44] and [160].

In our problem, we make the assumption that the flux q is continuous and that the initial and boundary conditions ρ_0, ρ_a, ρ_b are functions of bounded variation. When the flux q models the flux of cars in terms of the car density ρ we obtain the LWR PDE. As explained earlier on, boundary conditions may not be fulfilled point-wise almost everywhere, thus following [17], we shall require that an entropy solution of 4.6 satisfy a weak formulation of the boundary conditions:

$$L(\rho(a,t),\rho_a(t)) = 0 \text{ and } R(\rho(b,t),\rho_b(t)) = 0 \tag{4.9}$$

where

$$L(x,y) = \sup_{k\in I(x,y)} (sgn(x-y)(q(x)-q(k))) \text{ and} \tag{4.10}$$

$$R(x,y) = \inf_{k\in I(x,y)} (sgn(x-y)(q(x)-q(k))) \text{ for } x,y \in \mathbb{R} \tag{4.11}$$

and $I(x,y) = [\inf(x,y), \sup(x,y)]$, where sgn denotes the sign function. In the case of a strictly concave flux (such as the Greenshields [66] and Greenberg [65] models used in traffic flow modelling), a simplified formulation of the boundary conditions can be developed (Le Floch gives analogous conditions in the case of a strictly convex flux in [98]):

$$\begin{cases} \rho(a,t) = \rho_a(t) \text{ and } q'(\rho_a(t)) \geqslant 0 \text{ or} \\ q'(\rho(a,t)) \leqslant 0 \text{ and } q'(\rho_a(t)) \leqslant 0 \text{ or} \\ q'(\rho(a,t)) \leqslant 0 \text{ and } q'(\rho_a(t)) \geqslant 0 \text{ and } q(\rho(a,t)) \leqslant q(\rho_a(t)) \end{cases} \tag{4.12}$$

Similarly, the boundary condition at b is:

$$\begin{cases} \rho(b,t) = \rho_b(t) \text{ and } q'(\rho_b(t)) \leqslant 0 \text{ or} \\ q'(\rho(b,t)) \geqslant 0 \text{ and } q'(\rho_b(t)) \geqslant 0 \text{ or} \\ q'(\rho(b,t)) \geqslant 0 \text{ and } q'(\rho_b(t)) \leqslant 0 \text{ and } q(\rho(b,t)) \leqslant q(\rho_b(t)) \end{cases} \tag{4.13}$$

As noticed in [98], we can always assume the boundary data are entering the domain at both ends. Indeed, if for example $q'(\rho_a(t)) < 0$ on a subset I of $\mathbb{R}_+$ of positive measure, the boundary data:

$$\tilde{\rho}_a(t) = \begin{cases} q'^{-1}(0) \text{ if } t \in I \\ \rho_a(t) \text{ otherwise} \end{cases} \tag{4.14}$$

will yield the same solution. With this assumption the boundary conditions can be written as:

$$\begin{cases} \rho(a,t) = \rho_a(t) \text{ or} \\ \\ q'(\rho(a,t)) \leqslant 0 \text{ and } q(\rho(a,t)) \leqslant q(\rho_a(t)) \end{cases} \tag{4.15}$$

and

$$\begin{cases} \rho(b,t) = \rho_b(t) \text{ or} \\ \\ q'(\rho(b,t)) \geqslant 0 \text{ and } q(\rho(b,t)) \leqslant q(\rho_b(t)) \end{cases} \tag{4.16}$$

We can now define an notion of entropy solution for a scalar conservation law 4.6 with initial and boundary conditions.

Definition: A solution of the mixed initial-boundary value problem for the PDE 4.6 is a function $\rho \in L^\infty((a,b)\times(0,T))$ such that for every $k \in \mathbb{R}$, $\varphi \in C_c^1((0,T))$, the space of C^1 functions with compact support, and $g \in C_c^1((a,b)\times(0,T))$ with φ and g nonnegative:

$$\int_a^b \int_0^T (|\rho - k|\frac{\partial g}{\partial t} + sgn(\rho - k)(q(\rho) - q(k))\frac{\partial g}{\partial x})dxdt \geqslant 0 \tag{4.17}$$

and there exist E_0, E_L, E_R three sets of measure zero such that :

$$\lim_{t\to 0, t\notin E_0} \int_a^b |\rho(x,t) - \rho_0(x)|dx = 0 \tag{4.18}$$

$$\lim_{x\to a, x\notin E_L} \int_0^T L(\rho(x,t), \rho_a(t))\varphi(t)dt = 0 \tag{4.19}$$

$$\lim_{x\to b, x\notin E_R} \int_0^T R(\rho(x,t), \rho_b(t))\varphi(t)dt = 0 \tag{4.20}$$

With this definition, we now establish the uniqueness by proving an L^1-semigroup property following the method introduced in [93] (see also [89], [126] and [139]). Let ρ, σ be two solutions of 4.6, φ and ψ two test functions in $C_c^1((0,T))$ and $C_c^1((a,b))$ respectively and nonnegative; the aforementioned definition yields:

$$\int_a^b \int_0^T (|\rho(x,t) - \sigma(x,t)|\psi(x)\varphi'(t)$$

$$+ sgn(\rho(x,t) - \sigma(x,t))(q(\rho(x,t)) - q(\sigma(x,t)))\varphi(t)\psi'(x))dxdt \geqslant 0 \tag{4.21}$$

This proof is similar to the proof of Theorem 1 in [93] except for the presence of boundary conditions. For ψ approximating $\chi\mid_{[a,b]}$, the characteristic function of the interval $[a,b]$, we have:

$$\int_a^b \int_0^T |\rho(x,t)-\sigma(x,t)|\varphi'(t)dt \geqslant \liminf_{x\to b} \int_0^T sgn(\rho(x,t)-\sigma(x,t))(q(\rho(x,t)) \\ -q(\sigma(x,t)))\varphi(t)dt - \limsup_{x\to a} \int_0^T sgn(\rho(x,t)-\sigma(x,t))(q(\rho(x,t))-q(\sigma(x,t)))\varphi(t)dt. \tag{4.22}$$

For a fixed $x \notin E_L$ and $t \in (0,T)$, we can always define

$$k(x,t) \in I(\sigma(x,t),\rho_a(t)) \cap I(\rho(x,t),\rho_a(t))$$

such that:

$$sgn(\rho(x,t)-\sigma(x,t))(q(\rho(x,t))-q(\sigma(x,t))) = sgn(\rho(x,t)-\rho_a(t))(q(\rho(x,t)) \\ -q(k(x,t))) + sgn(\sigma(x,t)-\rho_a(t))(q(\sigma(x,t))-q(k(x,t))) \leqslant L(\rho(x,t),\rho_a(x,t)) \\ +L(\sigma(x,t),\rho_a(x,t)) \tag{4.23}$$

Indeed, we can assume $\rho(x,t) > \sigma(x,t)$; if not we exchange $\rho(x,t)$ and $\sigma(x,t)$. If $\rho(x,t) > \rho_a(t)$ and $\sigma(x,t) < \rho_a(t)$, then the result is trivial as the $q(k(x,t))$ terms cancel out. If $\rho(x,t) > \rho_a(t)$ and $\sigma(x,t) > \rho_a(t)$, we have:

$$sgn(\rho(x,t)-\rho_a(t))(q(\rho(x,t))-q(k(x,t))) + sgn(\sigma(x,t)-\rho_a(t))(q(\sigma(x,t))-q(k(x,t))) = \\ q(\rho(x,t))+q(\sigma(x,t))-2q(k(x,t))$$

and choosing $k(x,t)=\sigma(x,t)$ yields the desired equality. Similarly if $\rho(x,t) < \rho_a(t)$ and $\sigma(x,t) < \rho_a(t)$, we choose $k(x,t)=\rho(x,t)$. Thus the equality holds in all cases.

The situation is similar in a neighbourhood of b which eventually yields:

$$\int_a^b \int_0^T |\rho(x,t)-\sigma(x,t)|\varphi'(t)dtdx \geqslant 0. \tag{4.24}$$

Therefore, for $0 < t_0 < t_1 < T$,

$$\int_a^b |\rho(x,t_1)-\sigma(x,t_1)|dx \leqslant \int_a^b |\rho(x,t_0)-\sigma(x,t_0)|dx \tag{4.25}$$

which proves the L^1-semigroup property from which the uniqueness follows.

4.3 Numerical methods

In this section, we prove the existence of a solution to equation 4.6 through the convergence of the Godunov scheme. Let $h = \frac{b-a}{M}$ and $I_i = [a + h(i - \frac{1}{2}), a + h(i + \frac{1}{2}))$ for $i \in \{0, ..., M\}$. For $r > 0$, let $J_n = [(n - \frac{1}{2})rh, (n + \frac{1}{2})rh)$ with $n \in \{0, 1, ..., N = E(1 + \frac{T}{rh})\}$. We approximate the solution ρ by ρ_i^n on each cell $I_i \times J_n$, with ρ_h the resulting function on $[a, b] \times [0, T]$. The initial and boundary conditions can be written as:

$$\begin{cases} \rho_i^0 = \frac{1}{h}\int_{I_i} \rho_0(x)dx \ , \ 0 \leqslant i \leqslant M \\ \\ \rho_0^n = \frac{1}{rh}\int_{J_n} \rho_a(t)dt \text{ and } \rho_M^n = \frac{1}{rh}\int_{J_n} \rho_b(t)dt \ , \ 0 \leqslant n \leqslant N \end{cases} \tag{4.26}$$

According to the Godunov scheme [62], ρ_i^{n+1} is computed from ρ_i^n by the following algorithm:

$$\begin{cases} \rho_{i+\frac{1}{2}}^n \text{ is an element } k \text{ of } I(\rho_i^n, \rho_{i+1}^n) \text{ such that } sgn(\rho_{i+1}^n - \rho_i^n)q(k) \text{ is minimal} \\ \\ \rho_i^{n+1} = \rho_i^n - r(q(\rho_{i+\frac{1}{2}}^n) - q(\rho_{i-\frac{1}{2}}^n)) \end{cases} \tag{4.27}$$

4.3.1 Proof of convergence of the numerical scheme and existence of solutions

Let $M_0 = \max(\|\rho_0\|_\infty, \|\rho_a\|_\infty, \|\rho_b\|_\infty)$; if the CFL (Courant-Friedrichs-Lewy) condition [101]

$$r \sup_{|k|<M_0} |q'(k)| \leqslant 1 \tag{4.28}$$

is verified, ρ_h converges in $L^1((a, b) \times (0, T))$ to a solution $\rho \in BV((a, b) \times (0, T))$. The CFL condition yields the following estimates:

$$|\rho_i^{n+1}| \leqslant (1 + C_0 h) \sup(|\rho_{i-\frac{1}{2}}^n|, |\rho_i^n|, |\rho_{i+\frac{1}{2}}^n|) + C_1 h \text{ for every } i \in \mathbb{Z} \tag{4.29}$$

$$\sum_{1 \leqslant i \leqslant M} |\rho_{i+1}^{n+1} - \rho_i^{n+1}| \leqslant (1 + C_2 h) \sum_{|i| \leqslant M+1} |\rho_{i+1}^n - \rho_i^n| + C_3 M h^2 \text{ for every } M \in \mathbb{N} \tag{4.30}$$

$$\sum_{|i| \leqslant M} |\rho_i^{n+1} - \rho_i^n| \leqslant \sum_{|i| \leqslant M+1} |\rho_{i+1}^n - \rho_i^n| + C_4 M h(1 + \sup_{i \in \mathbb{Z}} |\rho_i^n|) \text{ for every } M \in \mathbb{N} \tag{4.31}$$

from which we can deduce that a subsequence ρ_{h_n} converges strongly to a function $\rho \in L^\infty((a, b) \times (0, T))$ of bounded variation and verifying the initial condition. We also have for k of $I(\rho_i^n, \rho_{i+1}^n)$

$$|\rho_i^{n+1} - k| \leqslant |\rho_i^n - k| - r(sgn(\rho_{i+\frac{1}{2}}^n - k)(q(\rho_{i+\frac{1}{2}}^n) - q(k)) - sgn(\rho_{i-\frac{1}{2}}^n - k)(q(\rho_{i-\frac{1}{2}}^n) - q(k))) \tag{4.32}$$

which shows that ρ is a weak solution of 4.6. If $\varphi^n = \frac{1}{rh}\int_{I_n}\varphi(t)dt$ for $\varphi \in C_c^1((0,T))$, non negative, we have:

$$\sum_{0\leqslant n\leqslant N} sgn(\rho^n_{i+\frac{1}{2}} - k)(q(\rho^n_{i+\frac{1}{2}}) - q(k))\varphi^n rh \leqslant \sum_{0\leqslant n\leqslant N} sgn(\rho^n_{\frac{1}{2}} - k)(q(\rho^n_{\frac{1}{2}}) - q(k))\varphi^n rh$$

$$+ih\|\varphi'\|_\infty T(M_0 + |k|). \tag{4.33}$$

Let $\lambda(t)$ be the weak * limit in $L^\infty((0,T))$ of a subsequence of $q(\rho^i_{\frac{1}{2}})$; the following inequality holds:

$$\int_0^T sgn(\rho(x,t) - k)(q(\rho(x,t)) - q(k))\varphi(t)dt \leqslant \int_0^T sgn(\rho_a(t) - k)(\lambda(t) - q(k))\varphi(t)dt$$

$$+|x-a|\|\varphi'\|_\infty T(M_0 + |k|) \tag{4.34}$$

using that $sgn(\rho^n_{\frac{1}{2}} - k)(q(\rho_{\frac{1}{2}}) - q(k)) \leqslant sgn(\rho^n_0 - k)(q(\rho_{\frac{1}{2}}) - q(k))$. $\rho(x,.)$ is of bounded variation, therefore it converges strongly in L^1 sense to a limit $\alpha \in L^\infty((0,T))$ and it verifies:

$$sgn(\alpha(t) - k)(q(\alpha(t)) - q(k)) \leqslant sgn(\rho_a(t) - k)(\lambda(t) - q(k)) \tag{4.35}$$

for every $k \in \mathbb{R}$ and almost every $t \in (0,T)$. This inequality shows that $\lambda = q(\alpha)$ almost everywhere. and $L(\alpha(t), \rho_a(t)) \leqslant 0$ and ρ verifies the weak boundary condition at $x = a$. Similarly, ρ verifies the corresponding condition at $x = b$ and the existence is proved.

4.4 Simulations and comparison with experimental data

4.4.1 Practical formulation and implementation

We now turn to the practical implementation of the Godunov scheme for the LWR PDE. The scheme is written as follows:

$$\rho_i^{n+1} = \rho_i^n - r(q_G(\rho_i^n, \rho_{i+1}^n) - q_G(\rho_{i-1}^n, \rho_i^n)), \text{ for } 0 \leqslant n \leqslant N-1, 0 \leqslant i \leqslant M \tag{4.36}$$

If the flux q is concave, which is often the case in traffic flow modelling, it reaches its only maximum

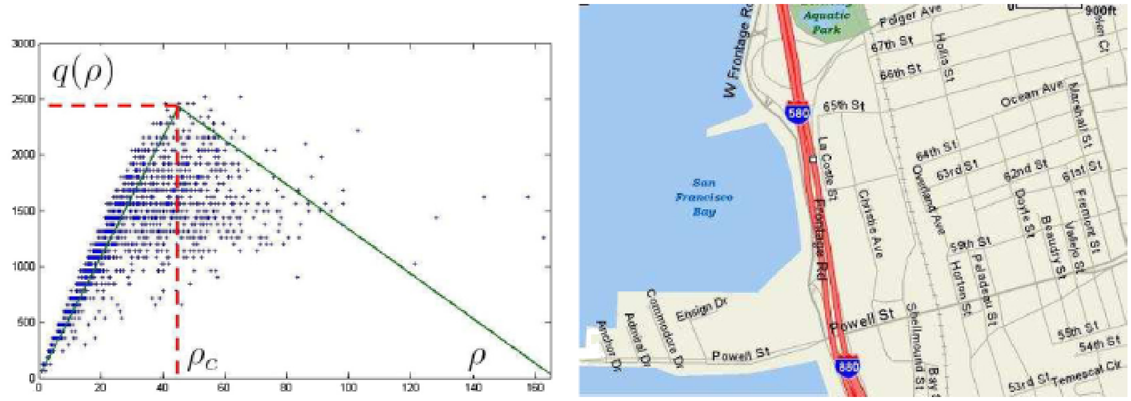

Figure 4.2: **Left:** Illustration of the empirical data obtained from the Berkeley Highway Laboratory. The horizontal axis represents the normalised density ρ (i.e. occupancy, see [63, 95] for more details). The vertical axis represents the flux $q(\cdot)$. Each track corresponds to a loop detector measurement. This data can easily be modelled with a triangular flux function, for which we display the critical density ρ_c and the jam density, ρ^*. **Right:** Location of the loop detectors used for measurement and validation purposes.

at the critical density ρ_c (see Figure 4.2) and the numerical flux q_G can be written explicitly as:

$$q_G(\rho_1, \rho_2) = \begin{cases} q(\rho_2) & \text{if } \rho_c < \rho_2 < \rho_1, \\ q(\rho_c) & \text{if } \rho_2 < \rho_c < \rho_1, \\ q(\rho_1) & \text{if } \rho_2 < \rho_1 < \rho_c, \\ \min(q(\rho_1), q(\rho_2)) & \text{if } \rho_1 \leqslant \rho_2. \end{cases} \tag{4.37}$$

The boundary conditions are treated via the insertion of a ghost cell on the left and on the right of the domain, that is:

$$\rho_0^{n+1} = \rho_0^n - r(q_G(\rho_0^n, \rho_1^n) - q_G(\rho_{-1}^n, \rho_0^n)) \tag{4.38}$$

with $\rho_{-1}^n = \frac{1}{rh} \int_{J_n} \rho_a(t) dt$, $0 \leqslant n \leqslant N-1$ for the left boundary condition and

$$\rho_M^{n+1} = \rho_M^n - r(q_G(\rho_M^n, \rho_{M+1}^n) - q_G(\rho_{M-1}^n, \rho_M^n)) \tag{4.39}$$

with $\rho_{M+1}^n = \frac{1}{rh} \int_{J_n} \rho_b(t) dt$, $0 \leqslant n \leqslant N-1$ on the right of the domain.

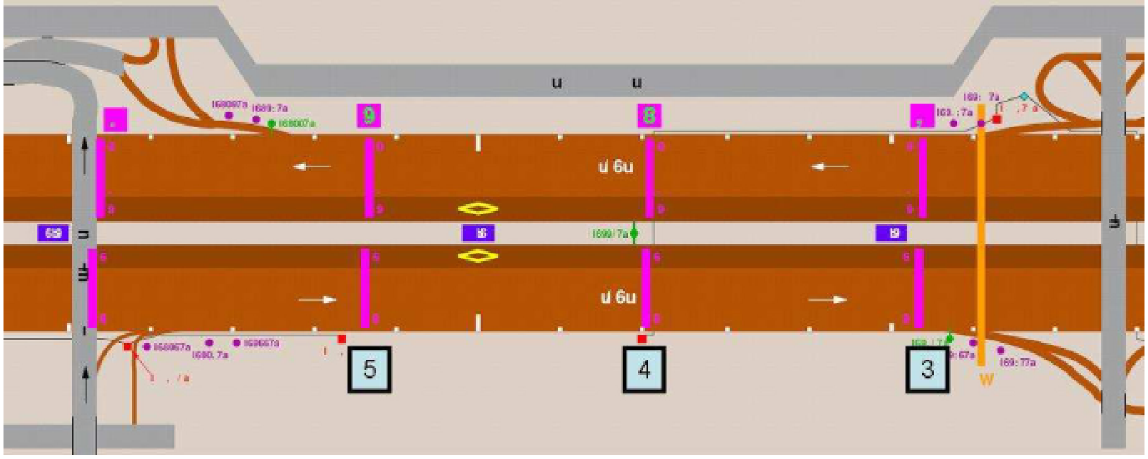

Figure 4.3: Set up of sensors 3, 4 and 5 used for this study in the Berkeley Highway Lab.

4.4.2 Comparison with experimental data

We apply this Godunov scheme to the simulation of highway traffic. A comparison of the density obtained numerically with the corresponding experimental density measured by the loop detectors is performed. We consider Interstate-80 Eastbound in West Berkeley and Emeryville and focus on a section going from loop detector 5 to 3 during 24 hours; we observe both free flow and congestion modes. The data measured by the inductive loop detectors is accessible through the Berkeley Highway Laboratory web site [95] (see Figure 4.2).

We measure the flow and density at the loop detector 3 to 5 (right subfigure in Figure 4.2 and Figure 4.3). The need for signal processing is quite visible from Figure 4.4; for this example, it was done using Fast Fourier Transform methods. The densities at detectors 5 and 3 are used as boundary values in the numerical scheme and the simulated and measured densities at detector 4 are compared for a 24 hours period. The numerical scheme was implemented using a grid of 100 points in space and 278,000 points in time and ran in less than 5 minutes on a desktop computer. The results shown in this figure illustrate the fact that the method is able to reproduce traffic flow patterns both in free flow and congestion modes. In particular, the accuracy is quite remarkable in the congestion periods when the density is above the critical density.

The graphs of the density in function of the distance at given times provide a practical signification

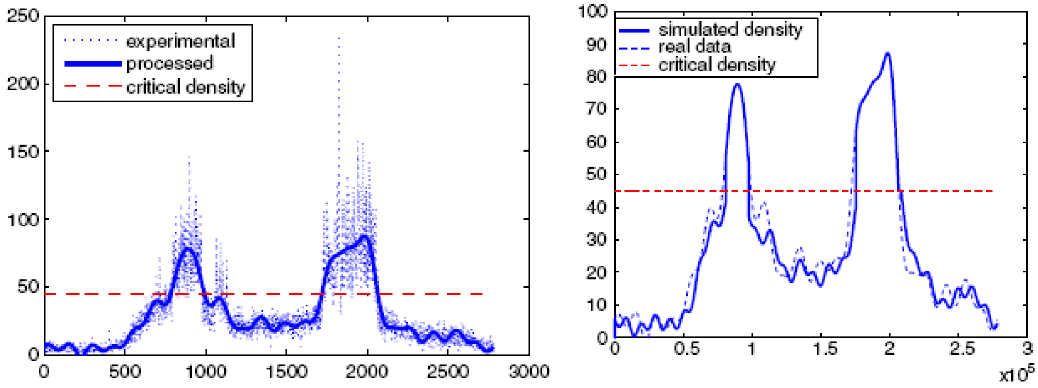

Figure 4.4: **Left:** BHL data used for the simulation, measured at loop detector 4. The horizontal axis represents time, the vertical axis represents the density at detector 4. **Right:** Comparison between loop detector 4 measurements and density simulations predicted by the model at the same location. The horizontal axis is time; the vertical axis is the vehicle density. Experimental data obtained from [95]

of the weak formulation of boundary conditions. For the sake of clarity, it was decided to represent the density on the entire section of highway considered as well as the ghost cell values. Whether a boundary condition at one end is applied point-wise or not can be seen immediately by checking on the graph for the presence of a discontinuity at that end. Indeed, when boundary conditions apply, the ghost cell value is equal to the value of the solution at this end ($\rho_a = \rho(a,t)$ or $\rho_b(t) = \rho(b,t)$). In free flow (density smaller than the critical density $\rho_c = 45$ vehicles/mile), characteristics enter the domain on the left and exit on the right, therefore the boundary conditions only applies on the left. Indeed this is easily verified on the corresponding graph (Figure 4.5, left bottom subfigure) as the value of the density on the left is equal to that of the ghost cell whereas these two values are different on the right creating a discontinuity. The next picture (Figure 4.5, right bottom subfigure) presents a fully developed congestion (density greater than the critical density $\rho_c = 45$) for which the characteristics leave the domain on the left and enter it on the right. Similarly, a discontinuity is seen on the left but not on the right. Next is a graph of a situation where the density is below the critical density on the left and above on the right; characteristics enter the domain at both ends, hence the boundary conditions apply point-wise at the two extremities (Figure 4.5, top left subfigure); no discontinuity can be seen at the ends. The last picture presents the opposite case with characteristics leaving the domain at both ends and no boundary conditions applying at all (Figure 4.5, top right subfigure); note the discontinuities at

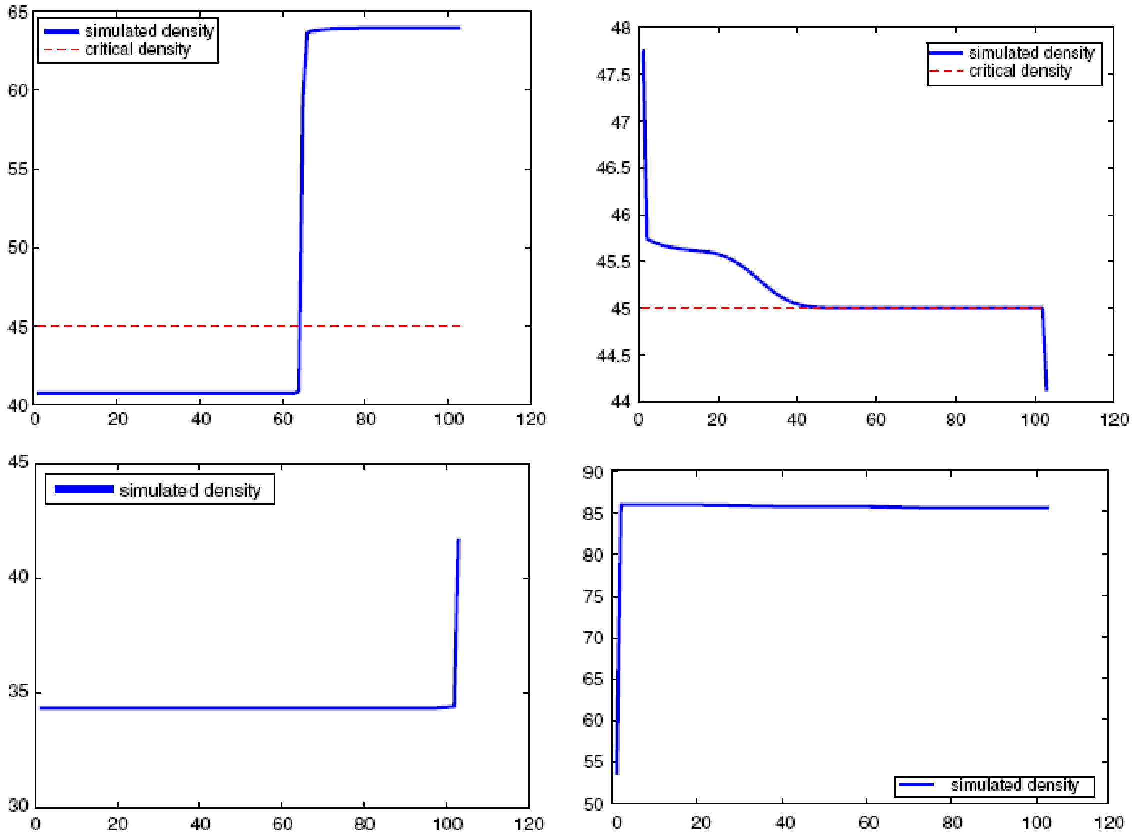

Figure 4.5: **Top Left:** The horizontal axis represents the distance and the vertical axis the density. Boundary conditions apply at both ends. Note the shock wave moving upstream.**Top Right:** Boundary conditions do not apply anywhere. **Bottom Left:** Free flow. Boundary conditions apply only on the left. **Bottom Right:** Fully developed congestion. Boundary conditions apply only on the right.

each end. This is the first time that these phenomena are described in a traffic engineering work.

These graphs also present a confirmation of the correct modelling of weak boundary conditions by the numerical scheme developed in the chapter. Indeed, while one may be under the impression that the conditions are always imposed point-wise at both ends through the ghost cells, the preceding pictures show that when the characteristics leave the domain at one end, the ghost cell values are ignored and the value of the density at this extremity is unrelated to that of the ghost cell.

4.5 Optimisation of traffic flow

Our next endeavour is directed towards the minimisation of the mean time spent by cars travelling through a stretch of highway between $x = a$ and $x = x_1$ via the adjustment of the density of cars entering the highway. The results from Ancona and Marson [6], [7] enable us to solve this problem. The first step consists in studying the attainable set at a fixed point in space x_1: $A(x_1, \mathcal{C}) = \{\rho(x_1, .)\}$, ρ being a solution of the LWR PDE with $\rho_0 = 0$ and $\rho_a \in \mathcal{C}$ for a given set of admissible controls $\mathcal{C} \subset L^1_{loc}$. Using the method of generalised characteristics introduced by Dafermos ([37], [38]), the attainable set is shown to be compact, the key argument being that the set of fluxes $\{q(\rho_a), \rho_a \in \mathcal{C}\}$ is weakly compact in L^1 (see [96] for functional analysis in L^p spaces). The compactness of the attainable set in turn yields the existence of a solution to the optimal control problem

$$\min_{\rho_a \in \mathcal{C}} F(S_{(.)}\rho_a(x_1)) \tag{4.40}$$

for $F : L^1([0, T]) \to \mathbb{R}$ a lower semicontinuous functional and $\mathcal{C}$ a set of admissible controls. We use the semigroup notation $S_t\rho_a$ to designate the unique solution of the LWR PDE at time t (we refer to the textbook [43] for more on semigroup theory). In the case of traffic modelling on a highway, we wish to minimise the difference between the average incoming time of cars at $x = x_1$ and at $x = x_0$ which can be written as:

$$\min_{\rho_a \in \mathcal{C}} \left(\int_0^T tq(S_t\rho_a(x_1))dt - \int_0^T tg(t)dt \right) \left(\int_0^T g(t)dt \right)^{-1} \tag{4.41}$$

where $g(t)$ represents the number of cars entering the stretch of highway per unit of time. This amounts to solving the equivalent problem:

$$\min_{\rho_a \in \mathcal{C}} \int_0^T tq(S_t\rho_a(x_1))dt \tag{4.42}$$

For this particular problem, we make the following additional assumptions:

- the net flux of cars entering the highway is equal to the total number of cars arriving at the entry:

$$\int_0^T q(\rho_a(s))ds = \int_0^T g(s)ds$$

- for every time $t > 0$ the total number of cars which have entered the highway is smaller than or equal to the total number of cars that have arrived at the entry from time 0 to t:

$$\int_0^t q(\rho_a(s))ds \leqslant \int_0^t g(s)ds$$

- the number of cars entering the highway is at most equal to the maximum density of cars on the highway:

$$\rho_a(t) \in [0, \rho_m]$$

- after a given time T no cars enter the highway:

$$\rho_a(t) = 0 \text{ for } t > T.$$

The map $F : \rho \to \int_0^T tq(\rho(t))dt$ is obviously a continuous functional on $L^1_{loc}([0,T])$, hence the existence of a solution of an optimal control ρ_a. Furthermore, a comparison principle for solutions of scalar nonlinear conservation laws with boundary conditions established by Terracina in [153] will allow us to find an explicit expression of the optimal control. Indeed if $\rho(x,t)$ is a weak solution of the LWR PDE, $u(x,t) = -\int_x^{+\infty} \rho(y,t)dy$ is the viscosity solution [35] of the Hamilton-Jacobi equation

$$\frac{\partial u}{\partial t} + q(\frac{\partial u}{\partial x}) = q(0). \tag{4.43}$$

Since viscosity solutions verify a comparison property [34], so will the solution of the LWR PDE. Using the assumption made earlier that the net flux of cars entering the highway is equal to the total number of cars arriving at the entry, and since

$$\int_0^T tq(S_t\rho_a(x_1))dt = T\int_0^T q(S_t\rho_a(x_1))dt - \int_0^T \int_0^t q(S_s\rho_a(x_1))dsdt \tag{4.44}$$

the boundary control problem can be rewritten as:

$$\max_{\rho_a \in \mathcal{C}} \int_0^T \int_0^t q(S_s\rho_a(x_1))dsdt. \tag{4.45}$$

As we can assume that the boundary data is always incoming, the comparison principle shows that the optimal control $\tilde{\rho}$ should verify:

$$\int_0^t q(\tilde{\rho}(s))ds \geqslant \int_0^t q(\rho_a(s))ds, \text{ for every } t > 0 \text{ and } \rho_a \in \mathcal{C}. \tag{4.46}$$

Eventually we obtain the following expression of the optimal control $\tilde{\rho}$:

$$\tilde{\rho}(t) = \begin{cases} q^{-1}(\rho_m) & \text{if } g(t) \leqslant q(\rho_m) \text{ and } \int_0^t q(\tilde{\rho}(s))ds < \int_0^t g(s)ds \text{ or } g(t) > q(\rho_m) \\ q^{-1}(g(t)) & \text{if } g(t) \leqslant q(\rho_m) \text{ and } \int_0^t q(\tilde{\rho}(s))ds = \int_0^t g(s)ds \end{cases} \tag{4.47}$$

4.6 Conclusion

We have proved the existence and uniqueness of a weak solution to a scalar conservation law on a bounded domain. The proof relies on a weak formulation of the boundary conditions which is necessary for the problem to be well posed. For strictly concave flux functions, the simplified expression of the weak formulation of the boundary conditions was written explicitly. The corresponding Godunov scheme was developed and applied on a highway traffic flow application, using Berkeley Highway Laboratory data for Highway Interstate-80. The numerical scheme and the parameters identified for this highway were validated experimentally against measured data. Finally, the existence of a minimiser of travel time was obtained, with corresponding optimal boundary control.

This work should be viewed as a first step towards building sound metering control strategies for highway networks: it defines the mathematical solution, and appropriate boundary conditions to apply in order to pose and solve the optimal control problem properly. Not using the framework developed here while computing numerical solutions of the LWR PDE would lead to ill-posed problems and therefore the data obtained through a numerical simulation would be meaningless.

Our result is crucial for highway performance optimisation, since by nature, in most highways, traffic flow control is achieved by on-ramp metering, i.e. boundary control. However, results are still lacking in order to generalise our approach to a real highway network. For such a network, the existence and uniqueness of a solution to the LWR PDE on each link no longer holds. The existence was proved in [73] using front-tracking methods by adding a condition that a functional based on the fluxes of the incoming links had to be maximised at the junction; uniqueness was proved using a similar condition in [33] and traffic flow simulations based on these models were presented in [25, 26, 27]. In order to solve optimisation problems arising in transportation networks, several approaches rely on the computation of the gradient of the optimisation functional, which for example could be achieved using adjoint-based techniques [67, 68].

Chapter 5

Summary and future work

In this brief final chapter, we will summarise the work presented in this dissertation. The topic of modelling and simulation of large scale distributed parameter systems was developed through the description and implementation of two inverse modelling algorithms applied to river flows as well the weak formulation of boundary conditions and implementation of a numerical scheme for highway traffic flow modelling.

5.1 Contributions on river flow estimation and future work

Two algorithms for inverse modelling of river flows were developed in this thesis. They both use on quadratic programming techniques which presents the advantage of a small computational cost and certainty of reaching the optimal solution. The first algorithm combines a model driven by the two-dimensional shallow water equations and observations collected by drifting sensors equipped with GPS units for real time river state estimation [148]. In order to apply quadratic programming methods to this variational data assimilation problem, three linear numerical schemes based on the linearised shallow water equations were developed and implemented. These schemes were chosen to be of implicit nature so that they would be unconditionally stable thereby limiting the size of the problem. This algorithm was then evaluated on the Sacramento River through computer based twin experiments in which the QP based method was compared to the EnKF algorithm as well as in field experiments in which data obtained from drifters released in the Sacramento River was used to reconstruct velocity fields. For the latter, the numerical schemes were extended to non-orthogonal curvilinear coordinates which means that they can be applied to rivers of any geometry. The second algorithm deals with estimating tidally driven open boundary conditions over time, using a one-dimensional shallow water model and observations from drifters [147]. The Preissmann scheme was used to discretise the one-dimensional shallow water equations and numerical experiments were performed to validate the algorithm. Future work on river flow estimation includes adding salinity or sediment modelling, optimal placement of drifters and extending the boundary condition estimation algorithm to networks of channels. Both algorithms on inverse modelling of river flows are expected to be implemented for real time data assimilation on the next generation drifters built by Andrew Tinka and his team.

5.2 Contributions on highway traffic flow and future work

The proper weak formulation of boundary conditions for scalar conservation laws such as the ones used to model highway traffic was introduced and a numerical scheme developed for highway traffic simulation was proved to converge to the unique solution of the corresponding continuous partial differential equation (the Lighthill-Whitham-Richards equation). An optimal control strategy was developed for congestion mitigation through ramp metering. Another avenue of transportation research in recent years has been the development of algorithms for highway state estimation using the vehicles as probes, an approach similar to the one presented for river flows with GPS equipped drifters being replaced by GPS equipped cars. Indeed, knowledge of the density and velocity fields is crucial to determine bottleneck locations, physical extent of queues, and travel time. Using data collected from stationary detectors, such as inductive loop detector stations as in this chapter, these fields can be reconstructed to a good accuracy. However, the deployment and maintenance of this type of infrastructure is expensive as it requires physically implanting inductive loops in the road and its reliability varies; this is particularly problematic in developing countries which are often faced with increasing road use and congestion coupled with limited funding for their transportation infrastructure but where mobile phones are widespread due to poor public telecommunication networks. This situation prevents accurate modelling of traffic as experimental data is limited. A possible solution is the use of data assimilation techniques based on Lagrangian (moving along with the flow) sensors on board vehicles. Indeed, new technology developments such as the Global Positioning System (GPS) allow mobile phones and navigation systems to broadcast their position and velocity in real time. If the mobile phone is located in a moving vehicle, the information collected from the phone can be used as a replacement or in supplement of the information provided by static sensors for traffic state estimation. For public transportation agencies, there is no cost of deploying this new sensing technology, which is market driven. In order to incorporate these Lagrangian measurements into an Eulerian model, data assimilation techniques are required and some of methods presented in the second chapter were applied using the LWR PDE discretised by the Godunov scheme developed earlier. In [71], the nudging method is used and compared with the Kalman filtering, while in [164] the ensemble Kalman filter is chosen. Separately, the adjoint method was applied to a second order model [14] in [82] to estimate the initial state of a stretch of highway. Several field experiments [71, 164] have been conducted with hundreds of cars equipped with GPS capable cellular phones being monitored as the complete loops on a stretch of highway. More experiments are planned and this approach to highway state estimation is likely to become a common fixture of the coming years in the field of transportation research.

Bibliography

[1] M. B. ABBOTT and F. IONESCU. On the numerical computation of nearly-horizontal flows. *J. Hydraulic Res., IAHR*, 5(2):97–117, 1967.

[2] M. B. ABBOTT and A. W. MINNS. *Computational Hydraulics*. Ashgate, 2nd edition, 1998.

[3] K. ADAMY and D. PHAM. A finite volume implicit Euler scheme for the linearized shallow water equations: stability and convergence. *Numerical Functional Analysis and Optimization*, 27(7–8):757–783, 2006.

[4] L. AMBROSIO, N. FUSCO, and D. PALLARA. *Functions of bounded variation and free discontinuity problems*. Oxford Mathematical Monographs. The Clarendon Press, Oxford University press, New York, 2000.

[5] K. AMONLIRDVIMAN, N. A. KHARE, D. R. P. TREE, W.-S. CHEN, J. D. AXELROD, and C. J. TOMLIN. Mathematical modeling of planar cell polarity to understand domineering nonautonomy. *Science*, 307(5708):423–426, 2005.

[6] F. ANCONA and A. MARSON. On the attainable set for scalar nonlinear conservations laws with boundary control. *SIAM J. Control Optim.*, 36(1):290–312, 1998.

[7] F. ANCONA and A. MARSON. Scalar non-linear conservations laws with integrable boundary data. *Nonlinear Analysis*, 35:687–710, 1999.

[8] B. D. O. ANDERSON and J. B. MOORE. *Optimal filtering*. Prentice-Hall, 1979.

[9] J. ANDERSON and M. MIERZWA. DSM2 tutorial, an introduction to the Delta Simulation Model II (DSM2). Technical report, State of California, Department of Water Resources, 2002. Available online at `http://wwwdwr.water.ca.gov/`.

[10] R. ANSORGE. What does the entropy condition mean in traffic flow theory? *Transportation Research*, 24B(2):133–143, 1990.

[11] R. A. ANTHES. Data assimilation and initialization of hurricane prediction models. *J. Atmos. Sci.*, 31:702–719, 1974.

[12] J. A. ATWELL, J. T. BORGGAARD, and B. B. KING. Reduced order controllers for Burgers' equation with a nonlinear observer. *Applied Mathematics and Computational Science*, 11(6):1311–1330, 2001.

[13] D. AUROUX and J. BLUM. Back and forth nudging algorithm for data assimilation problems. *C. R. Math. Acad. Sci. Paris*, 340(12):873–878, 2005.

[14] A. AW, A. KLAR, T. MATERNE, and M. RASCLE. Derivation of continuum traffic flow models from microscopic follow-the-leader models. *SIAM J. Appl. Math.*, 63(1):259–278, 2000.

[15] J. BAKER, A. ARMAOU, and P. D. CHRISTOFIDES. Nonlinear control of incompressible fluid flow: Application to Burgers' equation and 2d channel flow. *Journal of Mathematical Analysis and Applications*, 252:230–255, 2000.

[16] A. BALOGH and M. KRSTIC. Burgers' equation with nonlinear boundary feedback: H_1 stability, well posedness, and simulation. *Mathematical Problems in Engineering*, 6:189–200, 2000.

[17] C. BARDOS, A. Y. LE ROUX, and J. C. NÉDÉLEC. First order quasilinear equations with boundary conditions. *Communications in partial differential equations*, 4(9):1017–1034, 1979.

[18] A. M. BAYEN, R. RAFFARD, and C. TOMLIN. Adjoint-based control of a new eulerian network model of air traffic flow. *IEEE Transactions on Control Systems Technology*, 14(5):804–818, 2006.

[19] A. F. BENNETT. *Inverse Methods in Physical Oceanography*. Cambridge University Press, 1992.

[20] A. F. BENNETT. *Inverse Modeling of the Ocean and Atmosphere*. Cambridge University Press, 2002.

[21] C. BISCHOF, H. M. BÜCKER, and A. RASCH. Sensitivity analysis of turbulence models using automatic differentiation. *SIAM J. Sci. Comp.*, 26(2):510–522, 2005.

[22] A. G. L. BORTHWICK and G. A. AKPONASA. Reservoir flow prediction by contravariant shallow water equations. *J. Hydraul. Eng.*, 123:432–439, 1997.

[23] F. BOUTTIER and P. COURTIER. Data assimilation concepts and methods. Lecture Notes, ECMWF, 1999.

[24] A. BRESSAN. *Hyperbolic systems of conservation laws: the one dimensional Cauchy problem*. Oxford Lecture Series in Mathematics and its Applications, 20, Oxford University Press, 2000.

[25] G. BRETTI, R. NATALINI, and B. PICCOLI. Fast algorithms for the approximation of a traffic flow model on networks. *Discrete Contin. Dyn. Syst. Ser. B*, 6(3):427–448, 2006.

[26] G. BRETTI, R. NATALINI, and B. PICCOLI. Numerical approximations of a traffic flow model on networks. *Netw. Heterog. Media*, 1(1):57–84, 2006.

[27] G. BRETTI, R. NATALINI, and B. PICCOLI. A fluid-dynamic traffic model on road networks. *Arch. Comput. Methods Eng.*, 14(2):139–172, 2007.

[28] G. BURGERS, P. VAN LEEUWEN, and G. EVENSEN. Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.*, 126:1719–1724, 1998.

[29] R. H. BYRD, P. LU, and J. NOCEDAL. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Stat. Computing*, 16(5):1190–1208, 1995.

[30] C. I. BYRNES, D. S. GILLIAM, and V. I. SHUBOV. Semiglobal stabilization of a boundary controlled viscous Burgers' equation. In *Proceedings of the* 38^{th} *IEEE Conference on Decision and Control*, pages 680–681, Phoenix, AZ, Dec. 1999.

[31] W. Castaings, D. Dartus, M. Honnorat, F. X. Le Dimet, Y. Youkili, and J. Monnier. Automatic differentiation: a tool for variational data assimilation and adjoint sensitivity analysis for flood modeling. In *Proceedings of the 4th International Conference on Automatic Differentiation*, Chicago, July 2004.

[32] A. Chadwick, J. Morfett, and M. Borthwick. *Hydraulics in Civil and Environmental Engineering*. Spon Press, London, 4th edition, 2004.

[33] G. M. Coclite, M. Garavello, and B. Piccoli. Traffic flow on a road network. *SIAM J. Math. Anal.*, 36(6):1862–1886, 2005.

[34] M. G. Crandall. Viscosity solutions: a primer. In *Viscosity solutions and applications, Lecture Notes in Mathematics 1660*, pages 1–43, Springer, Berlin, 1997.

[35] M. G. Crandall, L. C. Evans, and P.-L. Lions. Some properties of viscosity solutions of Hamilton-Jacobi equations. *Trans. Amer. Math. Soc.*, 283(2):487–502, 1984.

[36] C.T. Crowe, D.F. Elger, and J.A. Roberson. *Engineering Fluid Mechanics*. John Wiley & Sons, Inc, US, 2001.

[37] C. Dafermos. Generalized characteristics and the structure of solutions of hyperbolic conservation laws. *Indiana Math. J.*, 26:1097–1119, 1977.

[38] C. Dafermos. *Hyberbolic conservation laws in continuum physics*. Grundlehren der Mathematischen Wissenchaften, 325, Springer-Verlag, 2000.

[39] C. Daganzo. The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research*, 28B(4):269–287, 1994.

[40] C. Daganzo. The cell transmission model, part II: network traffic. *Transportation Research*, 29B(2):79–93, 1995.

[41] P. De Mey and Y. Ménard. Synoptic analysis and dynamical adjustment of GEOS 3 and SEASAT altimeter eddy fields in the northwest Atlantic. *J. Geophys. Res.*, 94:6221–6230, 1989.

[42] P. De Mey and A. R. Robinson. Assimilation of altimeter eddy fields in a limited area quasi-geostrophic model. *J. Phys. Oceanogr.*, 17:365–384, 1991.

[43] L. C. Evans. *Partial Differential Equations*. American Mathematical Society, 1998.

[44] L. C. Evans and R. F. Gariepy. *Measure theory and fine properties of functions*. CRC Press, 1991.

[45] G. Evensen. Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, 99(C5):143–162, 1994.

[46] G. Evensen. The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dynamics*, 53:343–367, 2003.

[47] G. Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer-Verlag, Berlin, 2007.

[48] H. B. Fischer, E. J. List, R. C. Y. Koch, J. Imberger, and N. H. Brooks. *Mixing in inland and coastal waters*. Academic Press, New York, 1979.

[49] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Thomson Brooks/Cole, 2nd edition, 2003.

[50] M. Fujihara and A. G. L. Borthwick. Godunov-type solution of curvilinear shallow-water equations. *J. Hydraul. Eng.*, 126:827–836, 2000.

[51] I. Fukumori. Assimilation of Topex sea level measurements with a reduced gravity, shallow water model of the tropical Pacific Ocean. *J. Geophys. Res.*, 100(C12):25027–25039, 1995.

[52] I. Fukumori, J. Benveniste, C. Wunsch, and D. B. Haidvogel. Assimilation of sea surface topography into an ocean circulation model using a steady state smoother. *J. Phys. Oceanogr.*, 23:1831–1855, 1993.

[53] C. F. GAUSS. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium.* F. Perthes and I.H. Besser, Hamburg, 1809.

[54] D. C. GAZIS. *Traffic theory*. International series in Operations Research and Management Science, Kluwer Academic, Boston, 2002.

[55] I. Y. GEJADZE and G. J. M. COPELAND. Open boundary control problem for Navier-Stokes equations including a free surface: adjoint sensitivity analysis. *Comput. Math. Appl.*, 52(8–9):1243–1268, 2006.

[56] I. Y. GEJADZE, G. J. M. COPELAND, and I. M. NAVON. Open boundary control problem for Navier-Stokes equations including a free surface: data assimilation. *Comput. Math. Appl.*, 52(8–9):1269–1288, 2006.

[57] I. Y. GEJADZE and J. MONNIER. On a 2D 'zoom' for the 1D shallow water model: Coupling and data assimilation. *Comput. Methods Appl. Mech. Engrg.*, 196(45–48):4628–4643, 2007.

[58] A. GELB. *Applied Optimal Estimation.* MIT Press, Cambridge, MA, 1974.

[59] K. GEORGE. A depth-averaged tidal numerical model using non-orthogonal curvilinear co-ordinates. *Ocean Dynamics*, 57:363–374, 2007.

[60] M. GHIL. Meteorological data assimilation for oceanographers. Part I: description and theoretical framework. *Dyn. Atmos. Oceans*, 13:171–218, 1989.

[61] M. GHIL and P. MALANOTTE-RIZZOLI. Data assimilation in meteorology and oceanography. *Adv. Geophys.*, 23:141–265, 1991.

[62] S. K. GODUNOV. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Math. Sbornik*, 47:271–306, 1959.

[63] G. C. GOMES. *Optimization and microsimulation of on-ramp metering for congested freeways*. Ph.D. thesis, University of California, Berkeley, 2004.

[64] L. GOURDEAU, S. ARNAULT, Y. MÉNARD, and J. MERLE. Geosat sea-level assimilation in a tropical atlantic model using Kalman filter. *Ocean. Acta*, 15:567–574, 1992.

[65] H. GREENBERG. An analysis of traffic flow. *Operation Res.*, 7(1):79–85, 1959.

[66] B. D. GREENSHIELDS. A study of trac capacity. *Proc. Highway Res. Board*, 14:448–477, 1935.

[67] M. GUGAT, M. HERTY, A. KLAR, and G. LEUGERING. Optimal control for traffic flow networks. *J. Optim. Theor. Appl.*, 126(3):589–616, 2005.

[68] M. GUGAT, M. HERTY, A. KLAR, and G. LEUGERING. Conservation law constrained optimization based upon front tracking. *Math. Model. Num. Anal.*, 40(5):939–960, 2006.

[69] A. W. HEEMINK, M. VERLAAN, and A. J. SEGERS. Variance reduced ensemble Kalman filtering. *Mon. Wea. Rev.*, 129:1718–1728, 2001.

[70] R. HERMAN, T. LAM, and I. PRIGOGINE. Kinetic theory of vehicular traffic : comparison with data. General Motors Research Laboratories, 1971.

[71] J. C. HERRERA and A. M. BAYEN. Traffic flow reconstruction using mobile sensors and loop detector data. In *87th Transportation Research Board Annual Meeting*, Washington D.C., 2008.

[72] J. M. HERVOUET and L. VAN HAREN. *TELEMAC 2D. Version 5.2 – Principle note*, 2002.

[73] H. HOLDEN and N. H. RISEBRO. A mathematical model of traffic flow on a network of unidirectional roads. *SIAM J. Math. Anal.*, 26(4):999–1017, 1995.

[74] W. R. HOLLAND and P. MALANOTTE-RIZZOLI. Alongtrack assimilation of altimeter data into ocean circulation model: space versus time resolution studies. *J. Phys. Oceanogr.*, 19:1507–1534, 1989.

[75] M. HONNORAT, J. MONNIER, and F. X. LE DIMET. Lagrangian data assimilation for river hydraulics simulations. *Comput. Visual. Sci.*, DOI 10.1007/s00791-008-0089-x, 2009.

[76] P. HOUTEKAMER and H. L. MITCHELL. Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.*, 126:796–811, 1998.

[77] X. Y. HUANG and X. YANG. Variational data assimilation with the lorenz model. Hirlam Technical Report Number 26, 1996.

[78] K. IDE, P. COURTIER, M. GHIL, and A. LORENC. Unified notation for data assimilation: Operational, sequential and variational. *J. Met. Soc. Japan*, 75(1B):181–189, 1997.

[79] K. IDE, L. KUZNETSOV, and C. K. R. T. JONES. Lagrangian data assimilation for point vortex systems. *Journal of Turbulence*, 3(053), 2002.

[80] ILOG. *ILOG AMPL CPLEX System Version 11.0 User Guide*, 2008.

[81] Y. ISHIKAWA, T. AWAJI, K. AKITOMO, and B. QIU. Successive correction of the mean sea surface height by the simultaneous assimilation of drifting buoy and altimetric data. *J. Phys. Oceanogr.*, 26:2381–2397, 1996.

[82] P. JAISSON and F. DE VUYST. Data assimilation and inverse problem for fluid traffic flow models and algorithms. *Int. J. Numer. Meth. Eng.*, 76:837–861, 2008.

[83] A. JAMESON. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3:233–260, 1988.

[84] A. JAMESON. Analysis and design of numerical schemes for gas dynamics 1: Artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence. *International Journal of Computational Fluid Dynamics*, 4:171–218, 1995.

[85] A. JAMESON. Analysis and design of numerical schemes for gas dynamics 2: Artificial diffusion and discrete shock structure. *International Journal of Computational Fluid Dynamics*, 4:1–38, 1995.

[86] A. H. JAZWINSKI. *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

[87] K. T. JOSEPH and G. D. VEERAPPA GOWDA. Explicit formula for the solution of convex conservation laws with boundary condition. *Duke Math. J.*, 62(2):401–416, 1991.

[88] R. KALMAN. A new approach to linear filtering and prediction problems. *J. Phys. Oceanogr.*, 23:2541–2566, 1960.

[89] B. L. KEYFITZ. Solutions with shocks. *Comm. Pure and Appl. Math.*, 24:125–132, 1971.

[90] A. KLAR, R. D. KÜHNE, and R. WEGENER. Mathematical models for vehicular traffic. *Surv. Math. Ind.*, 6:215–239, 1996.

[91] T. KOBAYASHI. Adaptive regulator design of a viscous Burgers' system by boundary control. *IMA Journal of Mathematical Control and Information*, 18(3):427–437, 2001.

[92] M. KRSTIĆ. On global stabilization of burgers' equation by boundary control. *Systems and Control Letters*, 37:123–142, 1999.

[93] S. N. KRUŽKOV. First order quasilinear equations in several independent variables. *Math. USSR Sbornik*, 10:217–243, 1970.

[94] L. KUZNETSOV, K. IDE, and C. K. R. T. JONES. A method for assimilation of Lagrangian data. *Mon. Wea. Rev.*, 131(10):2247–2260, 2003.

[95] Berkeley Highway Laboratory. *http://bhl.its.berkeley.edu:9006/bhl/index.html*.

[96] P. D. LAX. *Functional analysis*. Wiley-Interscience, 2002.

[97] F. X. LE DIMET and O. TALAGRAND. Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus*, 38A:97–110, 1986.

[98] P. LE FLOCH. Explicit formula for scalar non-linear conservation laws with boudary condition. *Math. Meth. Appl. Sci.*, 10:265–287, 1988.

[99] A. Y. LE ROUX. On the convergence of the Godunov's scheme for first order quasi linear equations. *Proc. Japan Acad.*, 52:488–491, 1976.

[100] A. Y. LE ROUX. Vanishing viscosity method for a quasilinear first order equation with boundary conditions. In *Conference on the Numerical Analysis of Singular Perturbation Problems*, Nijnegen, 1978.

[101] R. J. LEVEQUE. *Finite Volume Methods for Hyperbolic Problems,*. Cambridge University Press, Cambridge, U.K., 2002.

[102] J. LEWIS, S. LAKSHMIVARAHAN, and S. K. DHALL. *Dynamic Data Assimilation: A Least Squares Approach*. Encyclopedia of Mathematics and its Applications, 104. Cambridge University Press, Cambridge, UK, 2006.

[103] M. J. LIGHTHILL and G. B. WHITHAM. On kinematic waves. I: Flow movement in long rivers. II: A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London*, A229(1178):281–345, 1955.

[104] J. L. LIONS. *Contrôle optimal de systèmes gouvernés par des équations aux dérivées partielles*. Dunod, Paris, France, 1968.

[105] A. C. LORENC. Analysis methods for numerical weather prediction. *Mon. Wea. Rev.*, 112:1177–1194, 1986.

[106] A. C. LORENC, R. S. BELL, and B. MACPHERSON. The meteorological office analysis correction data assimilation scheme. *Q. J. R. Meteorol. Soc.*, 117:59–89, 1991.

[107] J. LUND, E. HANAK, W. FLEENOR, R. HOWITT, J. MOUNT, and P. MOYLE. *Envisioning futures for the Sacramento-San Joaquin Delta*. Public Policy Insitute of California, San Francisco, CA, 2007.

[108] B. LUONG, J. BLUM, and J. VERRON. A variational method for the resolution of a data assimilation problem in oceanography. *Inverse Problems*, 14:979–997, 1998.

[109] H. LY, K. D. MEASE, and E. S. TITI. Distributed and boundary control of the viscous Burgers' equation. *Numerical Functional Analysis and Optimization*, 18(1-2):143–188, 1997.

[110] C. MAES, M. BENKIRAN, and P. DE MEY. Sea level comparison between Topex/Poseidon altimetric data and a global ocean circulation model from an assimilation perspective. *J. Geophys. Res.*, 104:15575–15585, 1999.

[111] P. MALANOTTE-RIZZOLI and R. E. YOUNG. How useful are localized clusters of traditional oceanographic measurement for data assimilation ? *Dyn. Atmos. Oceans.*, 17:23–61, 1992.

[112] S. MCCLURG. *Layperson's guide to the Delta*. Water Education Foundation, Sacramento, CA, 2000.

[113] J. L. MEAD. The Shallow Water equations in Lagrangian coordinates. *J. Comp. Phys*, 200(2):654–669, 2004.

[114] J. L. MEAD. Assimilation of simulated float data in Lagrangian coordinates. *Ocean Modelling*, 8(4):369–394, 2005.

[115] J. L. MEAD and A. F. BENNETT. Towards regional assimilation of Lagrangian data: The Lagrangian form of the Shallow Water reduced gravity model and its inverse. *J. Mar. Syst.*, 29:365–384, 2001.

[116] W. MENKE. *Geophysical Data Analysis: Discrete Inverse Theory*. Academic Press, 1984.

[117] A. MOLCARD, L. I. PITERBARG, A. GRIFFA, T. ÖZGÖKMEN, and A. MARIANO. Assimilation of drifter observations for the reconstruction of the Eulerian circulation field. *J. Geophys. Res.*, 108(C3):3056, 2003.

[118] A. M. MOORE. Data assimilation in a quasigeostrophic open-ocean model of the Gulf Stream region using the adjoint model. *J. Phys. Oceanogr.*, 21:398–427, 1991.

[119] S. E. MORLOCK. Evaluation of Acoustic Doppler Current Profiler measurements of river discharge. United States Geological Survey Water-Resources Investigations Report 95-4218, 1996.

[120] I. M. NAVON. Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. *Dyn. Atmos. Oceans*, 27:55–79, 1997.

[121] J. NOCEDAL and S. J. WRIGHT. *Numerical Optimization*. Springer, New York, 2nd edition, 2006.

[122] M. NODET. *Modélisation mathématique et assimilation de données lagrangiennes pour l'océanographie*. PhD thesis, Université de Nice Sophia Antipolis, 2005.

[123] M. NODET. Variational assimilation of Lagrangian data in oceanography. *Inverse Problems*, 22:245–263, 2006.

[124] O. A. OLEINIK. On discontinuous solutions of nonlinear differential equations. *Uspekhi Mat. Nauk.*, 12:3–73, 1957. English translation: *American Mathematical Society*, Ser. 2 No. 26 pp. 95–172, 1963.

[125] M. ÖNDER and E. ÖZBAY. Low dimensional modelling and dirichlet boundary controller design for Burgers equation. *International Journal of Control*, 77(10):895–906, 2004.

[126] F. OTTO. Initial-boundary value problem for a scalar conservation law. *C. R. Acad. Sci. Paris Ser. I Math.*, 322(8):792–734, 1996.

[127] C. PANICONI, M. MARROCU, M. PUTTI, and M. VERBUNT. Newtonian nudging for a Richards equation-based distributed hydrological model. *Adv. Water. Resour.*, 26(2):161–178, 2003.

[128] N. PETIT. *Systèmes à retard. Platitude en génie des procédés et contrôle de certaines équations des ondes*. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, 2000.

[129] N. A. PHILLIPS. On the completeness of a multi-variate optimum interpolation for large-scale meteorological analysis. *Mon. Wea. Rev.*, 110:1329–1334, 1982.

[130] A. PREISSMANN. Propagation des intumescences dans les canaux et rivières. In *1st Congress of the French Association for Computation*, Grenoble, 1961.

[131] I. PRIGOGINE and F. C. ANDREWS. A Boltzmann-like approach for traffic flow. *Operations Research*, 8(6):789–797, 1988.

[132] I. PRIGOGINE and R. HERMAN. *Kinetic theory of vehicular traffic*. American Elsevier, New York, 1971.

[133] A. K. RASTOGI and W. RODI. Predictions of heat and mass transfer in open channels. *ASCE J. Hydr. Div.*, 104(3):397–419, 1978.

[134] P. I. RICHARDS. Shock waves on the highway. *Operations Research*, 4(1):42–51, 1956.

[135] H. SALMAN, L. KUZNETSOV, and C. K. R. T. JONES. A method for assimilating Lagrangian data into a Shallow-water equation ocean model. *Mon. Wea. Rev.*, 134(4):1081–1101, 2006.

[136] B. F. SANDERS and N. D. KATOPODES. Adjoint sensitivity analysis for shallow-water wave control. *ASCE Journal of Engineering Mechanics*, 126(9):909–919, 2000.

[137] Y. SASAKI, P. GU, and L. YAN. A fundamental study of the numerical prediction based on the variational principle. *J. Meteor. Soc. Japan*, 33:262–275, 1955.

[138] J. B. SCHIJF and J. C. SCHÖNFELD. Theoretical considerations on the motion of salt and fresh water. In *Proc. Minn. Intern. Hydr. Conv.*, pages 321–333, 1953.

[139] M. E. SCHONBEK. Existence of solutions to singular conservation laws. *J. Math. Anal.*, 15(6):1125–1139, 1984.

[140] J. SCHROTER, U. SEILER, and M. WENZEL. Variational assimilation of Geosat data into an eddy-resolving model of the gulf stream area. *J. Phys. Oceanogr.*, 23:925–953, 1993.

[141] A. SEPPÄNEN, M. VAUHKONEN, E. SOMERSALO, and J. KAIPIO. State space models in process tomography – approximation of state noise covariance. *Inverse Problems in Engineering*, 9:561–585, 2001.

[142] D. SERRE. *Systems of conservation laws. Volume 1: Hyperbolicity, entropies, shock waves*. Cambridge University Press, Cambridge, 1999.

[143] I. SHULMAN, J. K. LEWIS, A. F. BLUMBERG, and B. N. KIM. Optimized boundary conditions and data assimilation with application to the M2 tide in the Yellow Sea. *J. Atmos. Ocean. Tech.*, 15:1066–1071, 1998.

[144] P. A. SLEIGH, P. H. GASKELL, M. BERZINS, and N. G. WRIGHT. An unstructured finite-volume algorithm for predicting flow in rivers and estuaries. *Comput. Fluids*, 27:479–508, 1998.

[145] I. S. STRUB and A. M. BAYEN. Lagrangian data assimilation of shallow water flows. Working paper, 2008.

[146] I. S. STRUB and A. M. BAYEN. Weak formulation of boundary conditions for scalar conservation laws: An application to highway traffic modelling. *Int. J. Robust Nonlinear Control*, 16:733–748, 2006.

[147] I. S. STRUB, J. PERCELAY, M. T. STACEY, and A. M. BAYEN. Inverse estimation of open boundary conditions in tidal channels. *Ocean Modelling*, 29(1):85–93, 2009.

[148] I. S. STRUB, J. PERCELAY, O. P. TOSSAVAINEN, and A. M. BAYEN. Comparison of two inverse modelling algorithms for river flows. To appear in *Networks and Heterogeneous Media*, 2009.

[149] D. SUN, I. S. STRUB, and A. M. BAYEN. Comparison of the performance of four eulerian network flow models for strategic air traffic management. *Networks and Heterogeneous Media*, 2(4):569–595, 2007.

[150] X. SUN and R. HOROWITZ. Localized switching rampmetering controller with queue length regulator for congested freeways. In *Proceedings of the American Control Conference*, pages 2141–2146, Portland, June 2005.

[151] O. TALAGRAND and P. COURTIER. Variational assimilation of meteorological observations with the adjoint vorticity equation. Part I : theory. *Q. J. R. Meteorol. Soc.*, 113:1311–1328, 1987.

[152] A. TARANTOLA. *Inverse Problem Theory. Methods for data fitting and model parameter estimation*. Elsevier, 1987.

[153] A. TERRACINA. Comparison properties for scalar conservation laws with boundary conditions. *Nonlinear Anal. Theory Meth. Appl.*, 28:633–653, 1997.

[154] W. C. THACKER and R. B. LONG. Fitting dynamics to data. *J. Geophys. Res.*, 93:1227–1240, 1988.

[155] A. TINKA, I. S. STRUB, O. P. TOSSAVAINEN, J. PERCELAY, Q. WU, and A. M. BAYEN. Inverse modelling in estuaries using passive drifting sensors. In preparation, 2009.

[156] E. F. TORO. *Shock-Capturing Methods for Free-Surface Shallow Flows*. J. Wiley and Sons, 2001.

[157] O. P. TOSSAVAINEN, J. PERCELAY, A. TINKA, Q. WU, and A. M. BAYEN. Ensemble Kalman filter based state estimation in 2D shallow water equations using Lagrangian sensing and state augmentation. In *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008.

[158] M. VENUTELLI. Stability and accuracy of weighted four-point implicit finite difference schemes for open channel flow. *J. Hydraul. Eng.*, 128(3):281–288, 2002.

[159] J. VERRON, L. GOURDEAU, D. T. PHAM, R. MURTUGUDDE, and A. J. BUSALACCHI. An extended Kalman filter to assimilate satellite altimeter data into a non-linear numerical model of the tropical Pacific Ocean: method and validation. *J. Geophys. Res.*, 104:5441–5458, 1999.

[160] A. I. VOL'PERT. The spaces BV and quasilinear equations. *Math. USSR Sbornik*, 2:225–267, 1967.

[161] C. B. VREUGDENHIL. *Numerical Methods for Shallow Water Flow*. Kluwer Academic Publishers, Dordrecht, 1994.

[162] T. WEIYAN. *Shallow Water Hydrodynamics: Mathematical Theory and Numerical Solution for a Two-dimensional System of Shallow Water Equations*. Elsevier Oceanography Series, 55. Elsevier, Amsterdam, NL, 1992.

[163] P. WESSELING. *Principles of Computational Fluid Dynamics*. Springer, Berlin, New York, 2001.

[164] D. WORK, O. P. TOSSAVAINEN, S. BLANDIN, A. M. BAYEN, T. IWUCHUKWU, and K. TRACTON. An ensemble Kalman filtering approach to highway traffic estimation using GPS enabled mobile devices. To appear, IEEE Conference on Decision and Control, 2008.

[165] Q. WU, X. LITRICO, and A. BAYEN. Data reconciliation of an open channel flow network using modal decomposition. *Advances in Water Resources*, 32:193–204, 2009.

[166] Z. YANG and J. M. HAMRICK. Optimal control of salinity boundary condition in a tidal model using a variational inverse method. *Estuarine, Coastal and Shelf Science*, 62:13–24, 2005.

[167] J. YI, P. GU, and L. YAN. Physical and mathematical models of pollution dispersion in the Yangtze estuary. *China Ocean Engineering*, 3(1):95–106, 1989.

[168] E. F. YOUNG, J. BROWN, and J. N. ALDRIDGE. Application of a large area curvilinear model to the study of the wind-forced dynamics of flows through the North Channel of the Irish Sea. *Cont. Shelf Res.*, 21:1403–1434, 2001.

[169] A. ZHANG, E. WEI, and B. B. PARKER. Optimal estimation of tidal open boundary conditions using predicted tides and adjoint data assimilation technique. *Continental Shelf Research*, 23(11):1055–1070, 2003.

[170] C. ZHU, R. H. BYRD, and J. NOCEDAL. L-BFGS-B: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization. *ACM Trans. Math. Soft.*, 23(4):550–560, 1997.