Enabling technologies for fast, nonlinear data assimilation in a coastal margin observatory

Sergey Frolov
B.S. Radioecology, International Sakharov Environmental University,
(Minsk, Belarus), 1999.
M.Sc.: *Environmental Science and Policy,* Central European University,
(Budapest, Hungary), 2000.

A dissertation presented to the faculty of the
OGI School of Science & Engineering
at Oregon Health & Science University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
in
Environmental Science and Engineering

November 2007

The dissertation “Enabling technologies for fast, nonlinear data assimilation in a coastal margin observatory” by Sergey Frolov has been examined and approved by the following Examination Committee:

_____

Dr. António M. Baptista, Professor
Dissertation Research Advisor
Oregon Health & Science University

_____

Dr. Todd K. Leen, Professor
Department of Computer Science and Electrical Engineering
Oregon Health & Science University

_____

Dr. Robert Miller, Professor
Oregon State University

_____

Dr. Mike Foreman
Head, Integration and Prediction
Institute of Ocean Sciences, Fisheries and Oceans Canada

_____

Dr. John C. Kindle
Head, Dynamics of Coupled Processes Section
Oceanography Division, Naval Research Laboratory

DEDICATIONS

To those who believed in me and
gave me a gift of their time.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Enabling technologies for fast, nonlinear data assimilation in a coastal margin observatory

Sergey Folov

Ph.D., OGI School of Science & Engineering at Oregon Health & Science University.
November 2007

Research Advisor: Dr. António M. Baptista

There is a building consensus among scientists, educators, managers, and politicians that integrated ocean observatories are a critical backbone to future scientific exploration, workforce training, and science-based management of coastal resources in the U.S. An integral part of the future infrastructure for these observatories is data assimilation (DA)—a mathematical technique that uses sparse observations of the ocean to constrain and improve the dynamics of a numerical model. The uses of DA in a coastal margin observatory are multiple and include an optimal estimate of the ocean state, an estimate of uncertainty for this state estimate, a suggestion for improving the design of the observational array, and a suggestion for improving the formulation of the numerical model. However, the wide application of DA in coastal margin observatories has been hampered, among other things, by the computational cost of existing algorithms, by the logistical difficulties in developing adjoint codes for rapidly evolving coastal models, by

the strong nonlinearity of coastal circulation processes, and by our ignorance about the statistics of model and forcing errors.

In this dissertation, we overcome many of the algorithmic and logistical challenges that impede wide application of advanced DA algorithms in coastal margin observatories. We demonstrate an application of the developed methods in an observatory for the Columbia River (CR) estuary and plume—an excellent test-bed for developing DA methods, with well documented, yet challenging dynamics. Once implemented, we used DA system in the CR to:

1) Assimilate in situ measurements of water levels, salinity and temperature into a multi-annual hindcast of the estuary.
2) Study the impact of DA on the dynamics of ecologically significant circulation features in the estuary and plume, such as the orientation and size of the plume, and the length of the salinity intrusion in the estuary.
3) Guide optimal placement of observational arrays in the estuary and plume.
4) Develop a real-time, assimilative forecast system for the estuary.

Our successful application of the enabling algorithms in the CR suggests that the developed technologies for fast, model independent DA and for optimization of observational arrays can be applied in many other coastal margin and coastal ocean observatories, enabling the implementation of these observatories at low computational and personnel costs.

Chapter 1

Introduction

1.1 Integrated ocean observatories

The Pew Oceans Commission report *America's Living Oceans: Charting a Course for Sea Change* (Pew Oceans Commission 2003) calls attention to the declining health of coastal ecosystems in the U.S. that threatens the coast-based economy, the livelihood, and the legacy of this nation. Examples of these threats include: rapid growth of coastal communities; land, water, and air pollution; unsustainable exploitation of fishery resources; and, frequently, ineffective management of coastal resources. The price tag for this threatened ocean- and coast-based economy is estimated at $1 trillion (U.S. Commission on Ocean Policy 2004). To prevent these threats to coastal economies and ecosystems, the U.S. Commission on Ocean Policy suggested replacing the current, fragmented management of coastal resources with a new ecosystem-based management (U.S. Commission on Ocean Policy 2004). Coastal economies and ecosystems can uniquely benefit from such a management strategy, which is based on an investment in scientific exploration and science-based management. The blueprint for this investment in scientific and management infrastructure was outlined by the U.S. Commission on Ocean Policy (U.S. Commission on Ocean Policy 2004), with a follow-up report from the Joint Ocean Commission Initiative (Joint Ocean Commission Initiative 2006) identifying the top 10 priorities for Congress to act on. One of these top ten priorities is to enact legislation that will authorize and fund the Integrated Ocean Observing System (IOOS).

Once fully enacted, the annual funding for the IOOS initiative is estimated to cost $500 million (Joint Ocean Commission Initiative 2006).

The proposed IOOS observatory is one of many existing and emerging ocean observatories, some examples of which include the existing Tropical Atmosphere and Ocean Array in the tropical Pacific (McPhaden 1995), the emerging ORION observatory from the National Science Foundation (National Research Council 2003), and numerous regional coastal observatories (Edson, et al. 2001, Ruef, et al. 2003, Proctor, et al. 2005, Baptista 2006). A common goal of such observatories is to provide a sustained, integrated view of ocean conditions for the stakeholders of the observatory, which often include scientists, educators, and natural resource managers. To achieve this objective, a typical integrated ocean observatory consists of three components: (1) observation, (2) data management, and (3) data analysis and modeling. The focus of this dissertation is on the modeling component of an integrated ocean observatory, specifically on the data assimilation (DA), which is expected to provide scientists and managers with an integrated view of ocean conditions and to guide deployment of ocean observing assets.

1.2 Data assimilation in a coastal-margin observatory

Data assimilation is an integral part of the modeling infrastructure for integrated ocean observatories. DA is a modeling technique that uses sparse observations of the ocean to constrain the dynamics of a numerical model. The uses of DA in an ocean observatory are many and include optimal estimates of the ocean state, estimates of uncertainty for the model prediction, suggestions for improving the design of the observational array, and suggestions for improving the formulation of the numerical model.

The theory of DA goes back to the seminal works of Gandin, Le Dimet , Bennett, Miller, and Cohn (Gandin 1963, Cohn 1982, Le Dimet, et al. 1986, Miller 1986, Bennett 1992), which proposed to use methods developed in the fields of optimal estimation and optimal control to problems of ocean and atmospheric DA. One group of these methods is based on the Kalman filter (KF) recursion (Kalman 1960)—a two-step recursion that first forecasts a model state and error statistics and then estimates the best fit of the model

state to measurements, using the minimum least square criterion. The direct implementation of the KF recursion is out of computational reach for high-dimensional, nonlinear ocean models, principally due to the prohibitive computational cost of forecasting error statistics. As a result, a number of suboptimal KF algorithms have emerged (Gandin 1963, Cane, et al. 1996, Lermusiaux, et al. 1999, Pham 2001, Oke, et al. 2002, Evensen 2003) One such suboptimal algorithm is the ensemble KF (Evensen 2003), which approximates the computationally expensive forecast of error-statistics using a limited (~100) ensemble of model states. Another such suboptimal algorithm is the optimal interpolation (Gandin 1963), which approximates error statistics using statistical models that are stationary in time and/or space.

Many of the KF-based algorithms were previously applied to assimilate measurements into models of a coastal ocean (Martin 2000, Oke, et al. 2002, Kindle 2005, Kurapov, et al. 2005, Wilkin, et al. 2005, Lermusiaux 2007). However, the applications of DA in the coastal margin, including estuaries, lagoons, and river plumes, are still rare (Xu, et al. 2002, Bertino, et al. 2002, Tores, et al. 2006). Moreover, many of these past applications in coastal ocean and coastal margin are based on variants of optimal interpolation—a simplified DA algorithm. Some of the challenges that have been precluding application of more advanced DA algorithms, like the ensemble KF, to problems in coastal ocean and coastal margin include:

- The dynamics of coastal-margin circulation are often highly nonlinear. These nonlinearities can lead to stability constraints in many DA methods that are based on the linearization of model dynamics and observations, such as variational DA (Le Dimet, et al. 1986, Bennett 1992) and the extended KF (Evensen 1992). Although these stability problems can be overcome by iterative methods (Gelb 1974, Bennett 2002), such methods are computational expensive.
- Circulation and ecosystem models for coastal margin are still rapidly evolving and usually do not come with a hard-to-develop adjoint models, required by variational DA (Le Dimet, et al. 1986, Bennett 1992), hence restricting the choice of algorithms to those based on approximations to the KF recursion (Gandin

1963, Evensen 1994, Cane, et al. 1996, Lermusiaux, et al. 1999, Pham 2001, Oke, et al. 2002, Evensen 2003).

- The computational cost of coastal ocean models is very high, since coastal circulation involves a multitude of temporal (from hours to decades) and spatial (from meters to hundreds of kilometers) scales. The high computational cost of these forward models determines the extremely high cost of DA algorithms since most existing DA algorithms require computational efforts equivalent to tens to hundreds of forward model integrations (Brusdal, et al. 2003, Ngodock, et al. 2006).
- Simulation of coastal ocean circulation usually requires specification of poorly known ocean boundary conditions and atmospheric forcings. In a forward model, these forcings are usually prescribed based on the predictions of larger-scale oceanic and atmospheric models. However, in an assimilative model, an error statistic for these forcings is also required, for which even less information is usually available.

1.3 Objective of the research

The objective of the research reported in this dissertation was to develop and to verify algorithms that will enable DA in complex, nonlinear environments that are common in coastal-margin observatories. The specific aims of our research were three fold:

1) To develop DA algorithms, which are computationally efficient, nonlinear, free of the adjoint model, and come with freely available software.
2) To develop algorithms for optimal placement of sensors in a coastal margin observatory.
3) To verify the developed DA and optimal sensor-placement algorithms for a challenging application in a coastal margin observatory.

The algorithmic approach that addresses the specific aims of our research is outlined in section 1.4. For verification of the developed algorithms we used the coastal-

margin observatory in the Columbia River (CR) estuary and plume, which was chosen for the following reasons:

1) The three-dimensional baroclinic dynamics of the estuary and plume are highly nonlinear and statistically non-stationary, which is likely to challenge the assumptions at the core of our DA algorithms. One such challenging, nonlinear dynamic is that of salinity intrusion, the length of which is determined by the balance of tidal mixing and the density difference between ocean and river waters. The statistical non-stationarity of the CR estuary circulation is partially due to non-stationarity in the river discharge, which modulates the length of salinity intrusion in the CR.
2) The dynamics of the CR estuary and plume are well documented (Hamilton 1990, Jay, et al. 1990, Jay, et al. 1997, Hickey, et al. 1998, Garcia-Berdeal, et al. 2002, Baptista, et al. 2005, Hickey, et al. 2005), providing the theoretical framework for analysis of DA estimates.
3) Observations and numerical models for the CR estuary and plume are readily available (Hickey 2004, Baptista, et al. 2005, Baptista 2006), hence removing the need for independent development of such observational and modeling programs.
4) Improved understanding of the CR estuary and plume dynamics, which is expected from the DA estimates, is of high practical and scientific value. In the past, model characterizations of the dynamics in the CR estuary and plume were used to support fisheries research (Bottom, et al. 2005, Burla, et al. 2007, Burla, et al. submitted) and management of the CR ecosystem (USACE 2001).

1.4 Overview of our algorithmic solutions

To overcome the algorithmic challenges for advanced DA in a coastal margin observatory, we[1] created and verified the following enabling technologies and algorithmic solutions in the CR estuary and plume:

[1] Much of the initial work on algorithms for training of model surrogates and for the formalism of the reduced-dimension Kalman filter was conducted in close collaboration with scientists in the Adaptive Systems Lab at Oregon Health & Science University and is reported in two articles (Lu, et al. 2007, van der Merwe, et al. 2007) that are not included in this dissertation.

- To expedite the computation of the KF, we developed the technology for neural-network model surrogates. Model surrogates use fast neural networks to approximate the dynamics of slow circulation models. Because it is not computationally feasible to train the model surrogate in the high-dimensional space of the forward model, the surrogate is trained in the subspace spanned by the Empirical Orthogonal Functions (EOF) of the forward model.
- To address strong nonlinearity in the dynamics of the CR estuary, we made a series of deliberate algorithmic choices. For example, instead of using more traditional linear autoregressive models (Hoteit, et al. 2003), we used nonlinear model surrogates, based on neural networks. To avoid linearization of observational and dynamical operators, we used a sigma point KF (van der Merwe, et al. 2001, van der Merwe 2004)—a nonlinear extension to the classical KF (Kalman 1960).
- To address the lack of an adjoint model for SELFE (Zhang, et al. submitted), the circulation code of choice in the CR estuary, we developed our DA method using a sigma point KF, which, like all KFs, does not require development of an adjoint model.
- To estimate the reduced-dimension dynamics of the model surrogate in the EOF subspace, we developed formalism of the reduced-dimension KF that extends a similar linear filter of (Cane, et al. 1996) to nonlinear systems and properly accounts for the EOF representation errors.
- To estimate the unknown parameters of errors in forcings, we used cross-validation (Bishop 1995)—a machine learning technique that effectively reuses available observational data for training and validation of a DA algorithm.

To find a strategy for optimal placement of fixed sensors in a coastal-margin observatory, we used the following algorithmic solutions that were validated in the CR estuary and plume:

- To quantify the utility of a candidate array, we used the statistical machinery of the best linear unbiased estimator (BLUE) (Ripley 1987), which predicts how

accurately the ocean state can be reconstructed from sparse measurements of a candidate array.

- To characterize the covariance structure for the field of interest, required by the BLUE estimator, we used a realistic, stationary, low-rank approximation of the covariance matrix, computed from the output of non-assimilative (Baptista, et al. 2005, Baptista 2006) and assimilative (Chapter 2 of this thesis) models of the CR estuary and plume.
- To search for optimal placements of sensors, we used several modified versions of the exchange-type optimization algorithm (Fedorov 1994).
- To quantify the statistical redundancy of an existing observational array, we analyzed the eigen-spectrum of the observation covariance matrix, using analysis similar to the array modes of Bennett (Bennett 1985, 1992).

1.5 Structure of this dissertation

The research presented in this dissertation consists of three components.

In the first component of our research, a research group that included the author developed algorithms for fast, model-independent, nonlinear DA. This research is documented in two manuscripts that are not included in this dissertation. The first of these complementary manuscripts (van der Merwe, et al. 2007) is now published in *Neural Networks* and documents training algorithms for model surrogates and shows their applicability to fast simulation of large ocean systems. The second complimentary article (Lu, et al. 2007) is submitted to *Journal of Inverse Problems* and documents the framework for reduced-dimension KF. A compact, self-sufficient overview of the model surrogate and the reduced-dimension KF is presented in the Chapter 2 of this thesis.

In the second component of our research, the developed algorithms were verified in the CR estuary and plume. The results of these verification experiments are presented in Chapters 2, 3, and 4, which are formatted for publication as independent papers in peer-reviewed journals. The compact overview of the chapters follows:

- In Chapter 2, we verified the applicability of the developed DA algorithms to estimation of salinity, temperature, and water level variability in the CR estuary.
- In Chapter 3, we verified the applicability of the developed DA algorithms to estimation of ecologically significant circulation features of the CR estuary and plume, such as direction and size of the CR plume, and the length of salinity intrusion in the CR estuary.
- In Chapter 4, we verified the applicability of the developed DA algorithms to a real-time assimilative nowcast of the CR estuary.

In the third component of our research, the algorithms for evaluation and optimization of fixed observational network in a costal-margin observatory were developed and verified in the CR estuary and plume. The network optimization algorithms and the verification experiments are presented in Chapter 5.

References

Baptista, A. M. (2006), "CORIE: The First Decade of a Coastal Margin Collaborative Observatory," in *Oceans'06 MTS/IEEE*, Boston.

Baptista, A. M., Zhang, Y.-L., Chawla, A., Zulauf, M. A., Seaton, C., Myers, E. P., Kindle, J., Wilkin, M., Burla, M., and Turner, P. J. (2005), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: Ii. Application to the Columbia River," *Continental Shelf Research*, 25, 935-972.

Bennett, A. F. (1985), "Array Design by Inverse Methods," *Progress in oceanography*, 15, 129-156.

Bennett, A. F. (1992), *Inverse Methods in Physical Oceanography*, Cambridge University Press.

Bennett, A. F. (2002), *Inverse Modeling of the Ocean and Atmosphere*, Cambridge, UK: Cambridge University Press.

Bertino, L., Evensen, G., and Wackernagel, H. (2002), "Combining Geostatistics and Kalman Filtering for Data Assimilation in an Estuarine System," *Inverse Problems*, 18, 1-23.

Bishop, C. (1995), *Neural Networks for Pattern Recognition*, Oxford University Press.

Bottom, D. L., Simenstad, C. A., Burke, J., Baptista, A. M., Jay, D. A., Jones, K. K., Casillas, E., and Schiewe, M. H. (2005), "Salmon at River's End. The Role of the Estuary in the Decline and Recovery of Columbia River Salmon," NOAA Technical Memorandum NMFS-NWFSC-68, U.S. Dept. Commer.

Brusdal, K., Brankart, J. M., Halberstadt, G., Evensen, G., Brasseur, P., van Leeuwen, P. J., Dombrowsky, E., and Verron, J. (2003), "A Demonstration of Ensemble Based Assimilation Methods with a Layered Ogcm from the Perspective of Operational Ocean Forecasting Systems," *Journal of Marine Systems*, 40-41, 253-289.

Burla, M., Baptista, A. M., Casillas, E., and Williams, J. G. (submitted), "The Influence of the Columbia River Plume on the Survival of Steelhead (Oncorhynchus Mykiss) and Chinook Salmon (O. Tshawytscha): A Numerical Exploration.," *Can. J. Fish. Aquat. Sci.*

Burla, M., Baptista, A. M., Zhang, Y.-L., Casillas, E., Bottom, D. L., and Simenstad, S. A. (2007), "Salmon Habitat Opportunity in the Columbia River Estuary: Modeling the Physical Environment to Inform Management Decisions," in *Coastal Zone '07*, Portland, OR.

Cane, M. A., Kaplan, A., Miller, R. N., Tang, B., Hackert, E., and Busalacchi, A. J. (1996), "Mapping Tropical Pacific Sea Level: Data Assimilation Via a Reduced State Space Kalman Filter," *Journal of Geophysical Research*, 101 (C10), 22599-22617.

Cohn, S. E. (1982), "*Methods of Sequential Estimation for Determining Initial Data in Numerical Weather Prediction,*" Ph.D. thesis, NYU.

Edson, J., Austin, T., McGillis, W., Purcell, M., Petitt, R., Ware, J., McElroy, M., Hurst, S., and Grant, C. (2001), "The Martha's Vineyard Coastal Observatory," in *OHP/ION*

Joint Symposium on Long-term Observations in the Oceans, Yamanashi Prefecture, Japan.

Evensen, G. (1992), "Using the Extended Kalman Filter with a Multilayer Quasi-Geostrophic Ocean Model," *Journal of Geophysical Research*, 97 (C11), 17905-17924.

Evensen, G. (1994), "Sequential Data Assimilation with a Non-Linear Quasi-Geostrophic Model Using Monte Carlo Methods to Forecast Error Statistics," *Journal of Geophysical Research*, 99 (C5), 10143–10162.

Evensen, G. (2003), "The Ensemble Kalman Filter: Theoretical Formulation and Practical Implementation," *Ocean Dynamics*, 53 (4), 343 - 367.

Fedorov, V. V. (1994), "Optimal Experimental Design: Spatial Sampling," *Calcutta Statistical Association Bulletin*, 44 (173-174).

Gandin, L. (1963), *Objective Analysis of Meteorological Fields*, Leningrad: Hydrometeoizdat.

Garcia-Berdeal, I., Hickey, B., and Kawase, M. (2002), "Influence of Wind Stress and Ambient Flow on a High Discharge River Plume," *Journal of Geophysical Research*, 107 (C9), 3130.

Gelb, A. (ed.) (1974), *Applied Optimal Estimation.*, MIT Press.

Hamilton, P. (1990), "Modeling Salinity and Circulation for the Columbia River Estuary," *Progress in Oceanography*, 25, 113-156.

Hickey, B. (2004), "River Influences on Shelf Ecosystems: Initial Impressions," in *AGU Fall meeting*, San Francisco.

Hickey, B., Geier, S., Kachel, N., and MacFadyen, A. (2005), "A Bi-Directional River Plume: The Columbia in Summer," *Continental Shelf Research*, 25 (14), 1631-1656.

Hickey, B. M., Pietrafesa, L. J., Jay, D. A., and Boicourt, W. C. (1998), "The Columbia River Plume Study: Subtidal Variability in the Velocity and Salinity Fields," *Journal of Geophysical Research*, 103 (C5), 10,339-10,369.

Hoteit, I., and Pham, D. T. (2003), "Evolution of the Reduced State Space and Data Assimilation Schemes Based on the Kalman Filter," *Journal of Meteorological Society Japan*, 81, 21-39.

Jay, D., and Smith, J. D. (1990), "Circulation, Density Distribution and Neap-Spring Transitions in the Columbia River Estuary," *Progress in Oceanography*, 25 (1-4), 81-112.

Jay, D. A., and Flinchem, E. P. (1997), "Interaction of Fluctuating River Flow with a Barotropic Tide: A Demonstration of Wavelet Tidal Analysis Methods," *Journal of Geophysical Research*, 102 (C3), 5705-5720.

Joint Ocean Commission Initiative. (2006), "From Sea to Shining Sea: Priorities for Ocean Policy Reform."

Kalman, R. E. (1960), "A New Approach to Linear Filtering and Prediction Problems," *ASME Journal of Basic Engineering*, 82, 35-45.

Kindle, J. (2005), "Near Real-Time Depiction of the California Current System," in *AMS Conference on Coastal Atmospheric and Oceanic Prediction and Processes*, San Diego, CA.

Kurapov, A. L., Allen, J. S., Egbert, G. D., Miller, R. N., Kosro, P. M., Levine, M., Boyd, T., Barth, J. A., and Moum, J. (2005), "Assimilation of Moored Velocity Data in a Model of Coastal Wind-Driven Circulation Off Oregon: Multivariate Capabilities," *Journal of Geophysical Research*, 110 C10S08.

Le Dimet, F.-X., and Talagrand, O. (1986), "Variational Algorithms for Analysis and Assimilation of Meteorological Observations: Theoretical Aspects," *Tellus A*, 38, 97-110.

Lermusiaux, P. F. J. (2007), "Adaptive Modeling, Adaptive Data Assimilation and Adaptive Sampling," *Physica D*, 230 (1-2), 172-196.

Lermusiaux, P. F. J., and Robinson, A. R. (1999), "Data Assimilation Via Error Subspace Statistical Estimation, Part I: Theory and Schemes," *Monthly Weather Review*, 127 (8), 1385-1407.

Lu, Z., Leen, T. K., van der Merwe, R., Frolov, S., and Baptista, A. M. (2007), "Sequential Data Assimilation with Sigma-Point Kalman Filter on Low-Dimensional Manifold," *submited to Journal of Inverse Problems*, also available at http://www.stccmop.org/files/CMOP-TR-07-001.pdf [Viewed on September 14, 2007].

Martin, P. J. (2000), *Description of the Navy Coastal Ocean Model Version 1.0.*, Naval Research Laboratory, Technical Report.

McPhaden, M. J. (1995), "The Tropical Atmosphere-Ocean Array Is Completed," *Bulletin of the American Meteriological Society*, 76, 739-742.

Miller, R. N. (1986), "Toward the Application of the Kalman Filter to Regional Open Ocean Modeling," *Journal of Physical Oceanography*, 16, 72-86.

National Research Council. (2003), "Enabling Ocean Research in the 21st Century: Implementation of a Network of Ocean Observatories."

Ngodock, H., E., Jacobs, G., A., and Chen, M. (2006), "The Representer Method, the Ensemble Kalman Filter and the Ensemble Kalman Smoother: A Comparison Study Using a Nonlinear Reduced Gravity Ocean Model," *Ocean Modeling*, 12 (3-4), 378-400.

Oke, P. R., Allen, J. S., Miller, R. N., Egbert, G. D., and Kosro, P. M. (2002), "Assimilation of Surface Velocity Data into a Primitive Equation Coastal Ocean Model," *Journal of Geophysical Research*, 107 (C9), 3122.

Pew Oceans Commission. (2003), "America's Living Oceans: Charting a Course for Sea Change. A Report to the Nation."

Pham, D. T. (2001), "Stochastic Methods for Sequential Data Assimilation in Strongly Nonlinear Systems," *Monthly Weather Review*, 129 (5), 1194-1207.

Proctor, R., Howarth, J., Holt, J., Wolf, J., Knight, P., Allen, I., and Holt, M. (2005), "The Synthesis of Modeling and Measurements in a Coastal Observatory," in *Asian and Pacific Coasts*, Jeju, Korea.

Ripley, B. D. (1987), *Spatial Statistics*, New York: John Wiley.

Ruef, W., Devol, A., Emerson, S., Dunne, J., Newton, J., Reynolds, R., and Lynton, J. (2003), "In Situ and Remote Monitoring of Water Quality in South Puget Sound: The Orca Time-Series," in *Georgia Basin/Puget Sound Research Conference*, Vancouver, British Columbia.

Tores, R., Allen, J. I., and Figueiras, F. G. (2006), "Sequential Data Assimilation in an Upwelling Influenced Estuary," *Journal of Marine Systems*, 60, 317–329.

U.S. Commission on Ocean Policy. (2004), "An Ocean Blueprint for the 21st Century. Final Report."

USACE. (2001), "Columbia River Channel Improvement Project: 2001 Biological Assessment," U. S. Army Corps of Engineers.

van der Merwe, R. (2004), "*Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*," Ph.D., OHSU.

van der Merwe, R., Leen, T. K., Lu, Z., Frolov, S., and Baptista, A. M. (2007), "Fast Neural Network Surrogates for Very High Dimensional Physics-Based Models in Computational Oceanography," *Neural Networks*.

van der Merwe, R., and Wan, E. (2001), "Efficient Derivative-Free Kalman Filters for Online Learning," in *Proceedings of the European Symposium on Artificial Neural Networks (ESANN)*, (Bruges, Belgium).

Wilkin, J. L., Arango, H. G., Haidvogel, D. B., Lichtenwalner, C. S., Glenn, S. M., and Hedström, K. S. (2005), "A Regional Ocean Modeling System for the Long-Term Ecosystem Observatory," *Journal of Geophysical Research*, 110 (C06S91).

Xu, J., Shenn-Yu, C., Raleigh, H. R., Wang, H. V., and Boicourt, W. C. (2002), "Assimilating High-Resolution Salinity Data into a Model of a Partially Mixed Estuary," *Journal of Geophysical Research*, 107 (C7), 11-11.

Zhang, Y.-L., and Baptista, A. M. (submitted), "A Semi-Implicit Eulerian-Lagrangian Finite-Element Model for Cross-Scale Ocean Circulation," *Ocean Modeling*.

Chapter 2

Fast Data Assimilation Using a Nonlinear Kalman Filter and a Model Surrogate: an Application to the Columbia River Estuary

List of authors: Frolov, S., Baptista, A. M., Leen T., Lu, Z., van der Merwe, R.

Abstract:

A fast, adjoint-free, nonlinear data assimilation (DA) system was developed for a 3D baroclinic simulation of the Columbia River (CR) estuary. The DA system took advantage of two recently developed technologies: (1) a nonlinear model surrogate (van der Merwe, et al. 2007) that executes forward simulation three orders of magnitude faster than a traditional numerical circulation code and (2) a nonlinear extension to the reduced-dimension Kalman filter (Lu, et al. 2007) that estimates the state of the model surrogate. The noise sources in the Kalman filter were calibrated using empirical cross-validation and included errors in model forcings and model error.

The developed DA system was applied to assimilate *in situ* measurements of water levels, salinities, and temperatures into the model surrogate of the CR estuary. To validate the DA results, we used a combination of cross-validation studies, process-oriented studies, and tests of statistical and dynamical consistency. The validation studies showed that DA improved upon the non-assimilative model in representing several important processes in the CR estuary, such as nonlinear tides, salinity intrusion, seasonal cycle of heating and cooling, and responses of the estuary to coastal winds.

Notation:

k – index of the discrete time;

$x(k)$ – state vector in the full space;

$u(k)$ – vector of forcings;

$y(k)$ – vector of observations;

$x_s(k)$ – state vector in the reduced space;

$\tilde{x}(k)$ – reconstruction of the vector $x_s(k)$ in the full space $\mathcal{M}$;

$\hat{x}_s^-(k)$ – optimal prediction of the state at time k conditioned on the observations up to time k-1 (state forecast);

$\hat{x}_s(k)$ – estimate of the reduced state at time k conditioned on the observations up to time k (state analysis);

$\hat{y}(k)$ – optimal prediction of the observation at time k;

X_k – time-lagged input to the model surrogate at time k (vector of past states);

U_k – time-lagged input to the model surrogate at time k (vector of past and future forcings);

Y_k – time-lagged output of the model surrogate at time k (vector of future states);

$f(.)$ – nonlinear dynamical model in the full space;

$h(.)$ – nonlinear observation function in the full space;

$f^s(.)$ – nonlinear model of the dynamics in the reduced space;

$h^s(.)$ – nonlinear observation function for the reduced state x_s;

$f^{ms}(.)$ – nonlinear model surrogate;

v, v_{ui}, v_x,, v_{ms} and v_{xms} – process, forcing, model, model surrogate, and combined model and model surrogate errors, and Q_{ui}, Q_x, Q_{ms}, and Q_{xms} are the respective error covariance matrixes;

n_m – measurement (sensor) errors;

n_o – vector of observation errors for the reduced space observation function $h^s(.)$;

$\mathcal{M}$ – full space $\mathbb{R}^l$ of state vectors $x(k)$;

$\mathcal{M}^{nl}$ – low-dimensional, possibly nonlinear, manifold of model trajectories;

$\mathcal{M}^s$ – reduced space;

Π – EOF dimension-reduction operator;

ε – reconstruction error;

μ – mean of the state vector;

σ_{vart} – normalization constant for the components of the state vector;

$E[.]$ – expectation operator;

$\mathcal{W}$ – concatenated vector of unknown weights and bias terms in the neural network;

λ – regularization parameter of the neural network;

T_i – internal PCA operator of the neural network;

$\mathbf{P}_{xx}^-$ – error covariance matrix for the state $\hat{x}_s^-$ (forecast error covariance);

$\mathbf{P}_{xx}$ – error covariance matrix for the state estimate $\hat{x}_s$ (analysis error covariance);

$\mathbf{P}_{yy}^-$ – covariance of innovation ($y(k) - \hat{y}^-(k)$);

K_k – Kalman gain at time k;

α_u and α_x – inflation factors for the forcing and model error covariance Q_u and Q_{xms};

$\hat{\sigma}_{inov}^2$ and σ_{inov}^2 – predicted and measured variance of the forecast error.

2.1 Introduction

Data assimilation (DA) plays a central role in emerging costal observatories. However, a wide application of many existing DA methods has been hampered, among other things, by the limited computational resources available to coastal oceanographers, by the logistical difficulties of developing adjoint codes for rapidly evolving coastal models, and by the strong nonlinearity of coastal flows. To address the need for fast, adjoint-free, nonlinear data assimilation methods, we recently proposed a nonlinear extension to the reduced-dimension Kalman filter (KF) (Lu, et al. 2007). The computational efficiency of the new method comes, in part, from the use of neural network model surrogates (van der Merwe, et al. 2007) that execute forward simulations three orders of magnitude faster than the traditional numerical circulation codes. This article reports on the application of the enabling technologies from (Lu, et al. 2007, van der Merwe, et al. 2007) to a realistic simulation of tides, salinity intrusion, and temperature in the Columbia River (CR) estuary.

The CR estuary is an excellent test-bed for developing DA methods for coastal observatories. The nonlinear dynamics of the CR estuary are well-documented (Hamilton 1990, Jay, et al. 1990, Jay, et al. 1997) and are likely to challenge the assumptions at the core of a DA algorithm. One such challenging, nonlinear dynamic is that of salinity intrusion, the length of which is determined by the balance of tidal mixing and the density difference between ocean and river waters. In the CR estuary, salinity intrusion length is also modulated by variations in astronomic tides, non-stationary river discharge, and coastal winds. A realistic representation of the salinity intrusion in the CR estuary requires a nonlinear model, capable of representing the flow of a stratified fluid, and requires precise knowledge of the forces that modulate the length of the salinity intrusion, which is hard to obtain.

CORIE—a coastal observatory for the CR estuary and plume—studies the dynamics of the CR estuary using a real-time observation network (Baptista 2006) and a modeling system for 3D baroclinic circulation of the CR estuary and the adjacent ocean (Baptista, et al. 2005). CORIE data and modeling products (Baptista, et al. 2005, Baptista

2006) are used to support the research and development of novel modeling techniques (Zhang, et al. 2004, Zhang, et al. submitted), fisheries research (Bottom, et al. 2005, Burla, et al. 2007, Burla, et al. submitted), and the CR ecosystem management (USACE 2001). The modeling products include multi-annual hindcasts and real-time forecasts of the baroclinic circulation, which are produced by SELFE (Zhang, et al. submitted), a finite-element, Eulerian-Lagragian forward model. DA is expected to improve the quality of these modeling products by reducing uncertainties in the model forcings (e.g., in salinity and temperature of the open ocean) and by correcting some persistent modeling errors (e.g., in seasonal cycle of heating and cooling).

A plethora of algorithms is available in oceanography for DA, with most of them falling into two broad categories: sequential and variational methods. Most sequential algorithms are based on KF (Kalman 1960)—a two-step recursion that first forecasts a model state and error statistics and then estimates the best fit of the model state to measurements, using the least squares criterion. The direct implementation of the KF recursion is out of computational reach for high-dimensional, nonlinear models like those used at CORIE, principally due to the prohibitive computational cost of forecasting error statistics. As a result, a number of sub-optimal KF-solutions have emerged (Gandin 1963, Cane, et al. 1996, Lermusiaux, et al. 1999, Pham 2001, Oke, et al. 2002, Evensen 2003). One such sub-optimal solution is the ensemble KF (EnKF) (Evensen 2003), which approximates the computationally expensive forecast of error-statistics using a limited (~100) ensemble of model states. Another such sub-optimal solution is the optimal interpolation (Gandin 1963), which approximates error statistics using statistical models that are stationary in time and/or space. In contrast to the sequential methods, the variational DA methods are based on solutions to Euler-Lagrange equations, where the numerical model is used as a dynamic constraint. Four leading variational algorithms are the 3DVAR (Le Dimet, et al. 1986), the 4DVAR (Le Dimet, et al. 1986), the representer method (Bennett 1992), and the adjoint method for parameter estimation (Carrera, et al. 1986).

In the past, DA algorithms have been successfully applied to assimilate measurements into models of the coastal ocean (Martin 2000, Heemink, et al. 2002, Oke, et al. 2002, Kurapov, et al. 2005, Wilkin, et al. 2005) and estuaries (Xu, et al. 2002, Bertino, et al. 2002). However, the existing DA algorithms are not likely to satisfy the needs of the CORIE observatory. First, the advanced DA algorithms are too computationally expensive to implement in the context of multi-annual hindcasts and real-time forecasts of the CR estuary. For example, implementing DA algorithms that allow for time-varying error statistics (like the EnKF, the 4DVAR, and the representer methods) might require a 100-fold increase in computational resources at CORIE. Second, the logistical costs of implementing variational methods (like the 4DVAR and the representer methods) are very high, since these methods require the development of a tangent-linear model and an adjoint model that are usually not available for emerging models like SELFE. Furthermore, while the computational and the logistical costs of implementing optimal interpolation are relatively low, the stationary approximation of the error covariance, used by the optimal interpolation, is likely to be inadequate for modeling the statistics of a system as dynamic as the CR estuary.

To overcome the limitations of the existing DA algorithms, we have recently developed a number of enabling technologies (Lu, et al. 2007, van der Merwe, et al. 2007) that rely on a model surrogate to expedite the KF computations. A model surrogate (van der Merwe, et al. 2007) is a very fast, nonlinear neural network that is trained to approximate a slow forward model. It is estimated that the model surrogate is 1,000-10,000 faster than the forward model (van der Merwe, et al. 2007). Because it is not computationally feasible to train the model surrogate in the high-dimensional space of the forward model, the surrogate is trained in the subspace spanned by the Empirical Orthogonal Functions (EOF) of the forward model. In our previous work (van der Merwe, et al. 2007), we developed algorithms for training of the model surrogates and demonstrated a successful application of the model surrogates to a month-long simulation of the CR estuary and plume.

To implement the KF recursion, we used the reduced-dimension KF (Lu, et al. 2007). Unlike similar filters (Cane, et al. 1996, Hoteit, et al. 2003) that used simplified models to propagate the error statistics and the original forward model to propagate the central forecast, the reduced-dimension KF assimilates the observations directly into the model surrogate. As a result, the need for a slow forward model in the reduced-dimension KF is obviated. Because the state of the model surrogate is usually small (<100 degrees of freedom), it is possible to implement the reduced-dimension KF using a full-rank KF. In our experiments, we used a Sigma Point KF (SPKF) (van der Merwe, et al. 2003, van der Merwe 2004)—a nonlinear extension to the classical KF that uses an ensemble of deterministically sampled model states (sigma points) to sample the covariance of model, forcing, and observational errors. In our previous work (Lu, et al. 2007), we demonstrated the application of the reduced-dimension KF to a synthetic estuary, successfully estimating both the state of the synthetic estuary and the time-varying errors in forcings.

Our reduced-dimension KF is similar to the existing reduced-space KFs, like the ensemble KF (Evensen 2003), the singular evolutive interpolated KF (Pham, et al. 1998), and the reduced-order extended KF (Cane, et al. 1996). One such similarity is the assumption that a model trajectory lies on a manifold of a lower dimension than the state-space of the numerical model. Most existing methods used this assumption to justify the rank reduction for the error covariance, while preserving the complexity of the forward model, e.g., (Pham, et al. 1998, Evensen 2003). In the reduced-dimension KF framework, we approximate the forward model with a model surrogate that already operates in the EOF space of a reduced dimension, which restricts the span of the error covariance to the same EOF subspace. To highlight this difference and to distinguish our KF framework from the existing reduced-space KFs, we use the term reduced-dimension KF. Since the performance of our reduced-dimension KF depends critically on the quality of the model surrogate, we speculate that our DA framework may work well for applications where the forward model has high skill in representing the statistics of the modeled system, hence providing good training samples for the model surrogate and the EOF basis. The reduced-dimension KF framework can be especially relevant in coastal applications, were errors are likely to come from uncertainties in the boundary conditions.

In this paper, we demonstrate an application of the model surrogate (van der Merwe, et al. 2007) and the reduced-dimension KF (Lu, et al. 2007) to a realistic, 1-year-long hindcast simulation of the 3D baroclinic circulation in the CR estuary. We assimilated *in situ* measurements of water levels, salinities, and temperatures into the model surrogate of the CR estuary, which was trained using 6 years of the high-resolution hindcast simulation data for the CR estuary. Our experiments showed that, for an application like the CR estuary, both an EOF and a model surrogate can be trained to perform well outside of the period they were originally trained on, thus allowing their application on unseen data in a real-time assimilative nowcast. Our validation studies showed that DA improves upon the non-assimilative model in representing several important processes in the CR estuary, such as nonlinear tides, responses of the estuary to coastal winds, salinity intrusion, and seasonal cycle of heating and cooling.

2.2 Data assimilation in the Columbia River estuary

One of the world's classic river-dominated estuaries, the CR estuary, is a highly energetic and dynamic system that responds quickly to changes in ocean tides, regulated river discharge, and coastal winds. The tides, with amplitudes of up to 3.6 m, are driven by nonlinear interactions of astronomic tides with complicated bathymetry, non-stationary river discharge, and coastal wind (Jay, et al. 1997). Compressed and often stratified, the estuarine circulation in the CR is subject to extreme variations in salinity intrusion and stratification regimes (Hamilton 1990, Jay, et al. 1990, Baptista, et al. 2005). In this paper, we report on the first data assimilation study of the 3D baroclinic circulation in the CR estuary.

2.2.1 Observational data for the Columbia River estuary

The observational network for CR estuary is presented in Figure 1. Each station in the network was instrumented with an array of sensors that measured a variable combination of parameters, including water level, salinity, temperature, velocity profile, air temperature, and wind velocity. In our DA experiments, we used measurements of water level, salinity, and temperature from 34 distinct sensors at 13 stations that are shown in Figure 1. Sensors at most of the stations were deployed at a single depth,

usually within a meter above the bottom. The exceptions were *red26* and *am169*, where three salinity and temperature sensors were placed throughout the water column to observe the vertical stratification of the estuary. In addition, three locations in the estuary (*red26*, *tansy*, and *am169*) were instrumented with acoustic Doppler profilers, which were used to validate our DA results. Most of the stations in Figure 1 were operated by CORIE[2], with the exception of tidal stations *tpoint* and *wauna,* which were operated by National Atmospheric and Oceanic Administration[3].

During the period of our study, there was a significant data loss in the observation network, due to a combination of bio-fouling, telemetry drop-outs, and gaps in the deployment schedule of sensors. To illustrate the extent of the problem, Figure 2 shows how the number of active sensors changed during the DA experiments. The black line in Figure 2 corresponds to all active sensors and the gray line, to salinity sensors influenced by the estuarine salt wedge. Of 34 total sensors, an average of only 23 sensors (70%) was available at any given time. From all sensors, the salinity sensors located in the estuarine salt wedge were most prone to bio-fouling, which led to inconsistent observations of the salinity intrusion. For example, it was common in July-August 2004 that only one or two, out of 10, salinity sensors remained active.

2.2.2 Hindcast simulation of the Columbia River estuary

One type of simulation generated by the CORIE modeling group is a multi-annual, high-resolution hindcast simulation of the CR estuary and plume. At the time of our experiments, the most accurate hindcast simulation of the CR estuary was DB16. DB16 had a realistic representation of the circulation, notwithstanding uncertainties in the boundary conditions and some persistent model errors, such as errors in atmospheric heat fluxes. We used the results of DB16 to compute the EOFs and to train the model surrogates described in sections 2.4 and 2.5. Here we present only a short summary of modeling choices and analysis of model results for DB16.

[2] Available at http://www.stccmop.org/corie/observation_network
[3] Available at http://www.co-ops.nos.noaa.gov/.

The simulation domain of DB16 (Figure 1) started at the mouth of the CR estuary and extended 88 kilometers upstream. DB16 was generated using SELFE (Zhang, et al. submitted) – a finite element, Eulerian-Lagrangian model that solves a set of nonlinear, baroclinic, shallow-water equations. The domain of DB16 was discretized using an unstructured horizontal grid (~16.000 elements) and a sigma-coordinate vertical grid (26 levels). The prognostic variables for DB16 included horizontal velocities, salinities, temperatures, and water levels.

DB16 was generated using the following boundary conditions:

Ocean boundary: The time series of water level, salinity, and temperature at the mouth of the estuary were extracted from DB14—a river-to-ocean model;

River boundary: The temperature and the flow of the CR estuary were based on measurements from the USGS station Beaver Army (*beaver*)[4];

Atmospheric forces: The 2D fields of wind stress, atmospheric pressure, and surface heat flux were generated using a combination of forecast and re-analysis products, which were provided by the National Center for Environmental Prediction and Oregon State University.

The modeled circulation in DB16 exhibited strong nonlinearity through the baroclinicity of the estuarine salt wedge, through sharp gradients of velocities, and through shallow-water tides. The spatial scales, resolved by DB16, were limited by the resolution of the numerical grid and ranged from ~40 m in the main channel of the estuary to ~600 m in tidal flats. The resolved periodic timescales included nonlinear tides (4-6 hours), astronomical tides (12-24 hours), spring-neap tidal cycle (~15 days), and seasonal changes in surface heat-fluxes and river temperature (~1 year). Nonperiodic timescales included weather changes (~2-10 days) and inter-annual variation in ocean conditions (>1 year).

[4] USGS stands for U.S. Geological Survey. Data for station Beaver Army is available at: http://waterdata.usgs.gov/usa/nwis/uv?site_no=14246900.

2.2.3 Aims for data assimilation in the Columbia River estuary

In the DA study presented in this article, we aimed to assimilate a long-term record of observations to ensure that the assimilated hindcast was consistent with the observed data. Specifically, we were interested in how the assimilation of a 1-year-long record of water level, salinity, and temperature measurements could improve the simulated long-term variability of tides, salinity intrusion, and temperature in the CR estuary. Several alternative aims for DA in the CR estuary are not addressed in this article. One such aim is assimilating high-resolution data from research cruises to improve the forecasts of salinity intrusion. Another is assimilating velocity data to improve the simulated transport of baroclinic tracers (e.g., salinity and temperature) and passive tracers (e.g., nutrients and pollutants) in the CR estuary.

2.3 DA method

2.3.1 Overview

The DA algorithm employed in our DA experiments was based on the framework for the reduced-dimension KF (Lu, et al. 2007), where observations are assimilated into the model surrogate, instead of into the original forward model. This formulation led to the following four-step algorithm:

1) A long, statistically representative hindcast simulation of the system, using the forward model SELFE, was used to provide training samples for a model surrogate of the CR estuary.
2) The training algorithms developed in (van der Merwe, et al. 2007) were used to train the model surrogate. Because it was not computationally feasible to train the model surrogate in the high-dimensional space of the forward model, the surrogate was trained in the EOF subspace.
3) The state of the model surrogate was estimated using a KF of choice. Because the estimated state of the model surrogate was small (50 degrees of freedom), we choose to use the SPKF– a state-of-the-art, full-rank KF (van der Merwe, et al. 2003, van der Merwe 2004).
4) The output of the KF was analyzed by reconstructing the estimated state and the estimated model errors from the EOF subspace, where the model surrogate and

the KF operated, to the full space of the model, where many analysis and visualization tools operated.

The four steps are grouped into two stages in Figure 3: the off-line stage, which involves one-time, off-line training of the model surrogate (steps 1 and 2 above; left box in Figure 3), and the on-line stage, which involves multiple DA experiments (steps 3 and 4 above; right box in Figure 3). Since all DA experiments (on-line stage, step 3 above) were performed in the reduced EOF subspace, the CPU, memory, and storage requirements for these DA experiments were very low. In the case of DB16, it took SELFE ~40 weeks to generate 6 years of the hindcast simulation using a single AMD Athlon© CPU. It took us ~4 weeks to compute the EOF basis and to train the model surrogate. And it took us only ~0.5 week to assimilate 6 years of observational data into the model surrogate. In practice, all these computations were distributed over multiple CPU processors, leading to shorter "wall-clock" times.

We present the description of our DA method for the CR estuary in two parts. In the first part (section 2.3), we present a concise overview of the concepts that comprise the reduced-dimension KF (Lu, et al. 2007), which include the EOF-based dimension reduction (Jolliffe 1986), the model surrogate (van der Merwe, et al. 2007), the classical KF (Kalman 1960), and the SPKF (van der Merwe, et al. 2003, van der Merwe 2004). Each of these concepts is well established in the literature, and readers are referred to the original sources for further algorithmic and theoretical details. In the second part (section 2.4), we document our implementation of the reduced-dimension KF for the CR estuary.

2.3.2 State model in the full space

The forward model and the observational system, described in sections 2.2.2 and 2.2.1, can be formalized as nonlinear dynamical and observation equations defined on the full space of the forward model:

$$x(k) = f(x(k-1), u(k), v(k)), \quad (1)$$

$$y(k) = h(x(k), n_m(k)), \quad (2)$$

where $x(k) \in \mathbb{R}^l$ is a state of the system (e.g., water level, salinity, temperature, and velocity) at discrete time *k*, $u(k) \in \mathbb{R}^p$ is a vector of forcings (e.g., wind, tides, ets), $y(k) \in \mathbb{R}^m$ is a measurement vector, $f(.) : \mathbb{R}^l \mapsto \mathbb{R}^l$ is a nonlinear dynamical model (e.g., SELFE), $h(.) : \mathbb{R}^l \mapsto \mathbb{R}^m$ is a nonlinear measurement model (e.g as described in section 2.2.1), and $v \in \mathbb{R}^v$ and $n_m \in \mathbb{R}^n$ are the process- and measurement-noise sources. We further denote the space $\mathbb{R}^l$ of state vectors $x(k)$, the full space $\mathcal{M}$.

2.3.3 Dimension reduction

To enable training of the model surrogates, computationally impossible in the full space $\mathcal{M}$, an EOF-based dimension reduction (also known as principle component analysis or proper orthogonal decomposition) is employed. To justify the use of EOF, it is assumed that the trajectories of the system occupy a low-dimensional (possibly nonlinear) manifold $\mathcal{M}^{nl}$ embedded in the full state-space $\mathcal{M}$. This nonlinear manifold is approximated by a linear, low-dimensional subspace $\mathcal{M}^s$, $\dim(\mathcal{M}^s) \equiv s << \dim(\mathcal{M})$. The EOF subspace is a good candidate for the linear subspace $\mathcal{M}^s$, since, of all linear subspaces of dimension *s*, the EOF subspace captures the highest percentage of the state variance.

The EOF subspace is characterized by an orthonormal projection operator **Π** that maps the vectors of the full space $x \in \mathcal{M}$ to the vectors of the reduced space $x_s \in \mathcal{M}^s$:

$$x_s = \mathbf{\Pi}(x - \mu), \qquad (3)$$

and reconstructs (embeds) the reduced space vectors $x_s \in \mathcal{M}^s$ in to the full space as $\tilde{x} \in \mathcal{M}$:

$$\begin{aligned} \tilde{x} &= \mathbf{\Pi}^T x_s + \mu \\ x &= \tilde{x} + \varepsilon = [\mathbf{\Pi}^T \mathbf{\Pi}(x - \mu) + \mu] + \varepsilon \end{aligned}, \qquad (4)$$

where μ is the mean of the dataset and ε is the reconstruction error.

The EOF projection operator **Π** is characterized by the *r* leading eigen vectors (φ_i) of the covariance matrix **C** for state vectors *x(k)*:

$$\mathbf{\Pi} = [\varphi_1, ..., \varphi_r]. \quad (5)$$

In practice, a Singular Vector Decomposition (SVD) (Golub, et al. 1996) is used to compute the operator **Π** efficiently, without computing the covariance matrix **C** explicitly (Bai, et al. 2000).

2.3.4 Nonlinear model surrogate

The projection and embedding equations (3 and 4) enable an easy interchange between the full space $\mathcal{M}$ and the reduced space $\mathcal{M}^s$ representation of the system state. However, for the purpose of simulation, a representation of the dynamics *f(.)* (Eq. 1) in the EOF basis (Eq. 5) is also needed. In the case of linear dynamics, the forward model *f(.)* can be expressed as a matrix operator and projected to the space $\mathcal{M}^s$ using the projection operator **Π** (Eq. 3) (Cane, et al. 1996). In a nonlinear case, the Galerkin method can be used to project the nonlinear equations of the dynamics to the EOF basis (Holmes, et al. 1996). However, such nonlinear projection is non-trivial to implement. To simplify our task, we used a model surrogate to approximate the dynamics of the forward model (Eq. 1) in the reduced basis (Eq. 5).

In the past, linear model surrogates were used to expedite DA algorithms. For example, Hoteit et al. (Hoteit, et al. 2003) used a linear autoregressive model to expedite the DA of tropical Pacific. In our preliminary experiments, we also trained a linear model surrogate using the autoregressive model with exogenous inputs (ARX) and the robust least-squares optimization method (Nabney 2004). However, we found that the trained surrogate was often unstable and led to the exponential growth of the ARX predictor response over time. To address the problem of the unstable linear ARX predictor, we trained a nonlinear model surrogate using a neural network. Output of the neural network is bound by design and cannot yield the exponential response that the linear ARX model yields.

The detailed discussion of the network architecture and the network's training and validation algorithms is presented in (van der Merwe, et al. 2007). The implementation details and the analysis of the accuracy are presented in section 2.4.4 of the present paper.

Below, we present a brief review of the concepts and terminology that underlie model surrogate training, many of which are common to the neural network literature (Bishop 1995).

Model surrogate: A model surrogate is a time-lagged, nonlinear neural network designed for iterative prediction:

$$Y_k = f^{ms}(X_{k-1}, U_k) + v_{ms} =$$

$$\begin{bmatrix} x_s(k+p') \\ . \\ x_s(k) \end{bmatrix} = f^{ms}\left(\begin{bmatrix} x_s(k-1) \\ . \\ x_s(k-n) \end{bmatrix}, \begin{bmatrix} u(k+p') \\ . \\ u(k) \\ . \\ u(k-p) \end{bmatrix} \right) + v_{ms}, \qquad (6)$$

where: *Y, X,* and *U* are the vectors of the time-lagged outputs, inputs, and forcings; v_{ms} is the error of the surrogate prediction; $f^{ms}(.)$ is a neural network implementation of the model surrogate; *k* is the index of the current time step; and *n*, *p* and *p'* are time-embedding indexes.

Intuitively, the reduced-space representation x_s captures the spatial frequencies of the modeled system, while time-lag in the input and the forcing vectors *X* and *U* captures the temporal frequencies and improves the accuracy of the network prediction. Time-lag in the output vector Y_k helps to train models for more accurate iterative predictions.

Although the dependence of the predicted state $x_s(k)$ on the future forcings *[u(k+1):…: u(k+p)]* in Eq. 6 may seem counterintuitive, this dependency should be viewed as an artifact of the training procedure. As training methods for model surrogates mature, the time-lag of prediction targets and future forcings may become obsolete, for example as in the algorithm SUR2 in (van der Merwe, et al. 2007). At this time, after training is complete, we discard the predictions of the future steps and only use the prediction of the first time step $x_s(k)$ in the iterative prediction loop, which leads to the following simplification in the notation for the model surrogate (Eq. 6):

$$x_s(k) = f^{ms}(X_{k-1}, U_k) + v_{ms}. \qquad (7)$$

Neural network: We implemented the model surrogate (Eq. 6) with a standard feed-forward multi-layer perceptron network. The network had a single hidden layer (with hyperbolic tangent activation functions), a linear output layer, and input pre-processing with an internal PCA (principle component analysis). Such networks are very well-equipped for modeling nonlinear relations among high-dimensional inputs and outputs where large datasets are available for model fitting (training). Where there is significant nonlinearity, their performance far exceeds that of traditional linear models such as ARX, ARMA, ARMAX and GLMs (Bishop 1995).

A mathematical expression for our network architecture is the following:

$$
\begin{aligned}
&Y_k = f^{ms}(X_{k-1}, U_k) + v_{ms} = \\
&= \sum_i^m w_{2i} \tanh\left(w_{1i}^T T_I \begin{bmatrix} X_{k-1} \\ U_k \end{bmatrix} + b_1 \right) + b_2 + v_{ms} \quad , \qquad (8) \\
&\mathcal{W} = [w_1(:); b_1(:); w_2(:); b_2(:)]; \\
&T_I = N_{I2} \Pi_I N_{I1}
\end{aligned}
$$

where *tanh(.)* is a hyperbolic tangent function; m is the size of the hidden layer; w_1, b_1 and w_2, b_2 are the weight and bias vectors for the first and second layer of the network; and $\mathcal{W}$ is a concatenated vector of all unknown parameters: w_1, b_1, w_2, and b_2.

Inputs to the network are pre-processed using matrix T_I – a dimension reduction operator that consists of the orthonormal projection operator Π_I (computed using PCA of the network inputs) and two diagonal normalization matrixes N_{I1} and N_{I2}. The matrixes N_{I1} and N_{I2} normalize the inputs and the outputs of the internal PCA operator T_I, by dividing each input and output with their standard deviations. After normalization, the inputs and the outputs of the operator T_I have unit variances. The pre-processing with matrix T_I reduces the number of unknown parameters in the network, improves the scaling of the network's cost function, and, as a result, leads to the faster convergence of the optimization algorithm.

The vector of unknown parameters $\mathcal{W}$ in Eq. 8 was optimized using a scaled conjugate-gradient algorithm (Møller 1996) that minimizes a regularized cost function:

$$J(\mathcal{W}) = \sum_{i=1}^{N} \left\| t_i - f^{ms}(X_i, U_i) \right\|_2^2 + \lambda \left\| \mathcal{W} \right\|_2^2 , \qquad (9)$$

where X_i and U_i are the input and forcing vectors that correspond to the training target t_i, N is the number of available targets (data points), $\| \ \|_2^2$ is the square of the 2-norm, and λ is the weight decay parameter determined through cross-validation. The cost function (Eq. 9) is a balance between the first term that penalizes the high prediction error v_{ms} of the neural network and a weight-decay regularizer (Bishop 1995) that penalizes large weights in the neural network and reduces model variance. The regularization of the network is critical to prevent over-fitting to data, which is a consequence of having too few data and too many free parameters. We determined the magnitude of the weight-decay parameter λ through cross-validation. For details on the optimization and the cross-validation algorithms, see (van der Merwe, et al. 2007) and section 2.4.3 of this paper.

Errors in the model surrogate: We assume that errors of a well-trained model surrogate are time-independent, zero-mean, Gaussian variables. We estimate the covariance of errors as:

$$Q_{ms} = E[v_{ms} v_{ms}^T] = E[(t - f^{ms}(X,U)(t - f^{ms}(X,U))^T] . \qquad (10)$$

2.3.5 Kalman filter in the reduced space

State equations in the subspace: To estimate the state of the dynamical system in the reduced space (Eq. 5), we define a reduced-dimension KF that assimilates data into the model surrogate (Eq. 7) (instead of the original dynamic model (Eq. 1)). We start by defining a dynamical equation in the reduced space:

$$x_s(k) = f^s(X_{k-1}, U_k, v(k)) \ , \qquad (11)$$

where $f^s(.)$ is a reduced space model implemented with the model surrogate (Eq. 7); X_i and U_i are the time-lagged input and forcing vectors; and *v(k)* is a process-noise vector that concatenates all error sources in the dynamical model $f^s(.)$, including the model surrogate error v_{ms}. We expand the definition of the noise vector *v(k)* for our application in the implementation section 2.4.6.

To redefine the observation function *h(.)* for the reduced state x_s, we employ the embedding equation (Eq. 4):

$$\begin{aligned} &y(k) = h(x(k), n_m(k)) = \\ &= h(\Pi^T x^s(k) + \mu + \varepsilon, n_m(k)) = \\ &= h^s(x^s(k), n_o(k)) \\ &n_o(k) = [\varepsilon; n_m(k)] \end{aligned}, \qquad (12)$$

where $h^s(.)$ is the observation function for reduced state x_s, ε is the reconstruction error from Eq. 4, and $n_o(k)$ is the concatenation of the measurement n_m and reconstruction error ε. We expand on implementation of the noise source $n_o(k)$ in section 2.4.6.

Kalman Filter recursion: The following recursive minimum-mean-square-error estimate[5] for the state $\hat{x}_s(k)$ in Eq. 11, conditioned on the observational data *y(k)* in Eq. 12, can be derived (Kalman 1960):

$$\begin{aligned} &\hat{x}_s(k) = \hat{x}_s^-(k) + K_k(y(k) - \hat{y}^-(k)) \\ &\mathbf{P}_{xx} = \mathbf{P}_{xx}^- - K_k \mathbf{P}_{yy} K_k^T \end{aligned}, \qquad (13)$$

where $\hat{x}_s^-(k)$ is the optimal prediction of the state at time *k* conditioned on the observations up to time *k-1* (state forecast), $\hat{y}^-(k)$ is the optimal prediction of the observation at time *k*, $\mathbf{P}_{xx}^-$ is the error covariance matrix for the state $\hat{x}_s^-(k)$ (the forecast error covariance), $\mathbf{P}_{yy}^-$ is the covariance of innovation $(y(k) - \hat{y}^-(k))$, and K_k is the Kalman gain at time *k*. The optimal terms in this recursion are given by:

$$\begin{aligned} &\hat{x}_s^-(k) = E[f^s(X_{k-1}, U_k, v(k))] \\ &\hat{y}^-(k) = E[h^s(\hat{x}_s^-(k), n_o(k))] \\ &\mathbf{P}_{xx}^- = E[(x_s(k) - \hat{x}_s^-(k))(x_s(k) - \hat{x}_s^-(k))^T] \\ &\mathbf{P}_{yy} = E[(y(k) - \hat{y}^-(k))(y(k) - \hat{y}^-(k))^T] \\ &\mathbf{P}_{xy} = E[(x_s(k) - \hat{x}_s^-(k))(y(k) - \hat{y}^-(k))^T] \\ &K_k = \mathbf{P}_{xy} \mathbf{P}_{yy}^{-1} \end{aligned}. \qquad (14)$$

[5] For a Gaussian prior and the linear state equations (11 and 12), the posterior distribution on the state $\hat{x}_s(k)$ is Gaussian. In this case, the mean of the posterior distribution is an optimal estimate under the minimum mean-square error, the maximum likelihood, and the maximum a posteriori criteria. In the case of a non-Gaussian noise source or non-linear state equations, the KF is only optimal under the minimum mean-square error criterion.

The recursion (Eq. 13) is linear in *K*; however, optimal terms (Eq. 14) do not assume linearity of the model $f^s(.)$ or measurement operator $h^s(.)$.

Implementing the KF recursion: For linear state equations (11 and 12), the solution for the optimal terms in the recursion (Eq. 14) is the well-known linear KF (Kalman 1960). To compute the optimal terms (Eq. 14) for nonlinear state equations (11 and 12), we used the SPKF (van der Merwe, et al. 2003, van der Merwe 2004). However, other versions of the KF (e.g., the ensemble KF) can be used together with a model surrogate to compute the optimal terms (Eq. 14) in the KF recursion, provided that the model surrogate and projection errors are properly accounted for. An example of such implementation using the ensemble KF can be found in (Lu, et al. 2007).

2.3.6 Sigma point Kalman filter

In the last two decades, several innovative algorithms were developed that implement the KF recursion for a nonlinear and/or a large system. In oceanographic literature, these algorithms include the ensemble KF (Evensen 2003), the singular evolutive interpolated KF (SEIK) (Pham 2001), and the singular evolutive extended KF (SEEK) (Pham, et al. 1998). In machine learning and control literature, the SPKF family of filters (van der Merwe, et al. 2001, 2003, van der Merwe 2004) is one of such novel algorithms. The major difference between the two groups of filters is that the oceanographic filters support very large estimation problems (~10^7 degrees of freedom), which is achieved by allowing the time- and measurement-updates in the KF recursion using a reduced-rank error covariance. For example, in the ensemble KF these updates involve only a limited (~100) number of ensemble states that span the subspace of the reduced-rank error covariance. In our experiments, we estimated a much smaller state of the model surrogate (50 degrees of freedom), which allowed us to use a full-rank KF, such as the SPKF. The SPKF is a state-of-the-art nonlinear extension to the classical KF; it is well-analyzed (van der Merwe, et al. 2003, van der Merwe 2004); and it comes with theoretically proven statistical properties (van der Merwe 2004), a freely available Matlab© toolkit (van der Merwe 2002-2006), and a large user-base.

We start introducing the SPKF algorithm by comparing it to the ensemble filters commonly used in oceanography, such as the ensemble KF (Evensen 2003) and the SEIK (Pham 2001). Like other ensemble methods (e.g., the ensemble KF and the SEIK), the SPKF represents the model statistics through a set of ensemble members (sigma points). As a result, the SPKF avoids the need for a tangent-linear approximation used by the variants of the extended KF, such as the SEEK (Pham, et al. 1998). Similarly to the SEIK, the SPKF algorithm uses the deterministic sampling of the prior covariance, unlike the ensemble KF that uses the Monte Carlo sampling, which is a more computationally expensive sampling method. Another advantage of the SPKF is that it directly supports nonlinear observation function and nonlinear observation- and process-noise models, which are only partially supported by the ensemble KF and the SEIK filter.

From a technical viewpoint, the SPKF algorithm addresses the problem of computing the optimal terms $\hat{x}_s^-(k)$, $\hat{y}^-(k)$, and K_k (Eq. 14) in the KF recursion (Eq. 13). In the SPKF, the state distribution is modeled as a Gaussian and is represented by a set of carefully chosen, weighted sample points (the sigma point transformation of the random variable). Figure 4 illustrates the sigma point transformation for the mean and the covariance of a highly nonlinear function $y=g(x)$ and a Gaussian random variable x. First, the prior distribution is sampled using 2n+1 sigma points drawn according to one of the deterministic methods (the central-difference method here). The mean and the covariance of the sigma points is exactly that of the prior. Second, the sigma points are propagated through a nonlinear function (a random neural network here). Lastly, the posterior mean and covariance are computed as a weighted sample mean and covariance of the updated sigma points, which capture the posterior mean and the covariance to the second order of accuracy (van der Merwe 2004).

A generic algorithm for the SPKF is listed in Appendix A1. Various implementations of the SPKF algorithm, including the hybridization with the particle filter, are discussed in (van der Merwe 2004). The details of the algorithm used in our experiment are specified in the implementation section 2.4.5.

2.4 Implementation

2.4.1 State vector and model forcings

Two representations of the state vector were used in our experiments:

1) The full-space state vector *x(k)* in Eq. 1, which described the state of DB16 and provided training samples for estimating the EOF dimension-reduction operator (Eq. 3); and
2) The reduced space state vector $x_s(k)$ in Eq. 11, which was used for training of the model surrogate (Eq. 6) and for the state estimation in the KF (Eq. 13).

The full-space state vector consisted of concatenated model variables defined at the nodes of the computational grid, namely: water level, salinity, temperature, and *u* and *v* components of velocity. The total length of the state vector was ~10^6.

Prior to concatenation, each model variable was normalized with the inverse of its standard deviation:

$$\sigma_{\text{vart}} = \sqrt{\frac{1}{l}\sum_{j=1}^{l}\left[\frac{1}{k}\sum_{i=1}^{k}(x_{\text{vart}}(j,i)-\mu_{\text{vart}}(j))^2\right]}, \qquad (15)$$

where *i* is the time index, *j* is the index over grid nodes, the subscript $_{\text{vart}}$ indicates the variable type (e.g., salinity or temperature), x_{vart} is the state vector for this variable type, μ_{vart} is the mean vector, *k* is the number of samples x_{vart}, and *l* is the number of nodes in the state vector x_{vart}. Only one normalization constant was used for each variable type.

To represent the extensive wetting and drying of tidal flats in the CR estuary, for which the model variables were set to missing values by the forward model, we used the following procedure to handle the missing values in the EOF algorithm. For water levels, we substituted the missing values with the values of the bathymetry at these locations. For the rest of the variables, we substituted the missing values with the mean values at these locations. During the reconstruction of the full state from the reduced state, we first reconstructed the water levels. Then, similarly to the masking of dry areas in SELFE, we

masked the nodes as dry at the locations where the water column was shallower than 0.1 m.

To describe the forcing vector of the model surrogate, we concatenated the original forcings of the forward model (DB16), including the time series of the ocean, river, and atmospheric boundary conditions. The boundary conditions were pre-processed using an EOF dimension reduction, resulting in the forcing vector with 29 degrees of freedom.

2.4.2 EOF dimension-reduction

To compute the EOF dimension-reduction operator for a large hindcast database of the CR estuary, we used two SVD-based algorithms, described in Appendix A2. Both algorithms address the computational limitations of the existing SVD algorithms, such as the QR algorithm in LAPACK (Anderson, et al. 1999), the Lancoz algorithm in ARPACK (Sorensen 1992), and the incremental SVD algorithm of (Brand 2002). None of these algorithms were able to directly compute the SVD decomposition of a 4Tb dataset that represented the 6.5 years of the simulated data in DB16.

Both EOF algorithms (Appendix A2) address the computational limitations of the existing SVD algorithms by computing an approximation to the EOF basis, based on a small random sub-sample of the original dataset. The second of the two algorithms—the divide-and-conquer EOF algorithm (DC-EOF)—has less stringent computational requirements, since it computes the EOF approximation in two steps: (i) the DC-EOF divides the large SVD problem into smaller sub-problems (e.g., by variable type or by region) and (ii) it combines the solutions of sub-problems into a global solution using the secondary EOF.

The EOF basis that was used in our experiments was originally computed using the DC-EOF algorithm, due to a stringent 2 Gb limitation on the size of the dataset, imposed by a 32-bit computer that was available at the time of experiments. At a later time, we compared the results of the two EOF algorithms discussed in Appendix A2,

using a 64-bit computer. We used 1000 random samples in the first step of both EOF algorithms. After retaining 50 EOF modes, we found that the reconstruction error was equally small for either algorithm. For the DC-EOF, the error was 2.0%; for the random sub-sampling EOF algorithm, the error was 1.9%. Since such a small difference in the reconstruction error was not likely to adversely affect the performance of the DA system, we chose to retain the DA results computed using the DC-EOF algorithm.

2.4.3 Model surrogate

To train and validate the model surrogates, we used the training and the cross-validation algorithms described in (van der Merwe, et al. 2007). The software implementation of these algorithms was developed by (van der Merwe, et al. 2007) and was based on Netlab (Nabney 2004)—a free network optimization library for MATLAB©. The neural network cost function (Eq. 9) was optimized using the scaled conjugate gradient algorithm (Møller 1996), with the network inputs normalized using the internal PCA (Eq. 8). A two-fold cross-validation, using a simple random permutation[6], estimated the value of the weight-decay parameter λ=5000.

To improve the accuracy of a model surrogate in the recursive prediction, we introduced training algorithm SUR3, which modifies algorithm SUR1 originally described in (van der Merwe, et al. 2007). The original SUR1 algorithm trained a model surrogate to predict only a single, next state of the system ($p'=0$ in Eq. 6), which was inconsistent with our intended use of the model surrogate for the recursive prediction. To resolve this inconsistency, van der Merwe (van der Merwe, et al. 2007) proposed the algorithm SUR2 that recursively trained the network to predict several steps in the future. Unfortunately, the algorithm SUR2 proved to be too computationally expensive for the intended application in the CR estuary. Instead, we introduced algorithm SUR3, which predicts a short time history of the future states ($p'>0$ in Eq. 6). The SUR3 algorithm removes the computationally expensive recursion from the training procedure, yet it informs the network of the long-term dynamics in the system.

[6] The term "random shuffle" cross-validation from (van der Merwe, et al. 2007)

Using simple trail-and-error calibration, we found the following values for the time-embedding parameters in Eq. 6: the time step for states and forcings Δt=0.5 hour, the past history of states n=12 hours, the past history of forcings p=24 hours, and the future history of state targets and forcings p'=6 hours. In the internal PCA pre-processing (with operator T_i in Eq. 8), we retained 99% of the varaince, which reduced the input layer of the network from 2911 inputs to 548 inputs.

2.4.4 Accuracy of EOF subspace and model surrogate

The accuracy of the model surrogate and of the EOF basis is critical to the performance of the reduced-dimension KF. In our analysis, we distinguished between two aspects of accuracy: the accuracy in reproducing data that was seen in training (the training error) and the accuracy in reproducing new, unseen data (the generalization error). In a hindcast DA, it can be sufficient to have a small training error, since it is always possible to overlap the time interval of a hindcast DA experiment with the time interval that was used in training of the model surrogate. However, a low training error is not sufficient in a real-time assimilative nowcast, which also requires a low generalization error on unseen data. In case of a high generalization error, a cumbersome retraining of the EOF basis and the model surrogate may become necessary (however, this was not necessary in our experiments; see section 2.5.3).

To determine whether the EOF subspace and the model surrogate, which were trained using DB16, generalized well beyond their training interval, we trained the model surrogate MS_B and the basis EOF_B. Both were trained using the first 5 years of DB16 and were tested using the remaining 1.5 years. The model surrogate MS_B and the basis EOF_B can generalize well if they have similar errors on training and test sets. To eliminate the possibility that the difference between the training and the test error was due to the temporal variability of the error, we compared the test errors for the EOF_B and the MS_B with the errors for the globally trained EOF_A and MS_A. The EOF_A and MS_A were trained on the entire 6.5 years of the DB16 data[7].

[7] The 6.5 years of DB16 simulation covered the period of data from April 1999 to September of 2005. During this period, no El Niño events were observed, which precluded us from testing the generalization

The analysis of the EOF-reconstruction error and the model surrogate prediction error[8], presented in Figure 5, showed that both the EOF_B and the model surrogate MS_B had low training errors (2.1% for EOF_B and 1.9% for MS_B). However, we found that only the EOF basis EOF_B was able to generalize well (errors of 2.1% on the training set and 2.0% on the test set). The model surrogate MS_B had a small discrepancy between errors on the training and the test set (1.5% vs. 2.3%), which was confirmed to be a generalization error, after we compared the errors of the model surrogate MS_A and the MS_B.

It is likely that the generalization error for the MS_B can be reduced by using more sophisticated cross-validation strategies for calibrating the weight-decay parameter λ in Eq. 9; see (van der Merwe, et al. 2007) for such strategies. Section 2.5.1.1 presents further analysis of how this generalization error affected the overall performance of the DA system and, hence, the prospect of deploying our DA system as a real-time nowcast.

To illustrate how the generalization error for the model surrogate MS_B affected the predicted values of salinity and temperature in the CR estuary, we compared the daily-averaged salinity (Figure 6.I) and the daily-averaged temperature (Figure 6.II) for the model surrogate MS_A and the MS_B. The comparisons were for the bottom sensor at station *am169*. Figure 6 shows that both the model surrogates, MS_A (sub-panels a) and MS_B (sub-panels b), tracked the solution of DB16 well. For example, the differences between DB16 and the model surrogates MS_A and MS_B, as measured by the root mean square (RMS) error, were very small: 2.2 psu vs. 2.8 psu for salinity and 0.5 C° vs. 0.6 C° for temperature.

---

skills of the EOF and model surrogate in reproducing the response of the estuary to climatic forcing from La Niña and El Niño.

[8] The model surrogate error was measured for a continuous 6.5 years feedback integration of the model surrogate, using the initial condition and the model forcings from DB16. The mean square errors were normalized by the variance of the state vector in the full space.

2.4.5 Sigma point Kalman Filter

To estimate the state of the system, we used a square root central difference formulation of the SPKF (van der Merwe, et al. 2003, van der Merwe 2004) with the scaling parameter $h = \sqrt{3}$. The square root version of the SPKF propagates the covariance matrix $\mathbf{P}_{xx}$ in numerically more stable and computationally more efficient square root form; the central difference refers to a sampling algorithm for the sigma points; and the scaling parameter determines the spread of the sigma points. Our implementation was based on ReBel (van der Merwe 2002-2006)—a well-tested, freely available Matlab© package for the recursive Bayesian estimation.

To ensure that the measurements of different types (like water level and salinity) were weighted similarly in the update equations for the KF (13), we normalized the measurements the same way we normalized the state vector of the system (15). The observations were assimilated by the SPKF every half-hour.

2.4.6 Noise models

A successful state estimation requires a well-designed process-noise model and a well-designed measurement-noise model. In section 2.3, we kept our derivation of the KF as general as possible by accumulating all sources of uncertainty in a generic noise vector *v(k)* for the process noise and $n_o(k)$ for the observational noise. In this section, we describe how we implemented these noise sources for the CR experiments.

Process noise: To characterize the errors in the model forcings, the errors of the model surrogate prediction, and the errors of the original forward model, we designed the following process-noise model:

$$
\begin{aligned}
& x_s(k) = f^s(X_{k-1}, U_k, v(k)) = \\
& = \sum_i^m w_{2i} \tanh\left(w_{1i}^T T_I \begin{bmatrix} X_{k-1} \\ U_k \end{bmatrix} + b_1 + v_{ui} \right) + b_2 + v_x + v_{ms} , \qquad (16) \\
& v_{xms} = v_x + v_{ms} \\
& v(k) = [v_{ui}; v_{xms}]
\end{aligned}
$$

where *tanh(.)* is a hyperbolic tangent function, *m* is the size of the hidden layer , w_1, b_1 and w_2, b_2 is the trained weight and bias vectors for the model surrogate (Eq. 8) (truncated to provide only one-step-ahead prediction), and T_I is the internal PCA operator of the trained model surrogate.

The noise sources in the Eq. 16 are:

$$\begin{aligned} v_{ui} &\sim N(\mathbf{0}, \mathbf{Q}_{ui}) \\ v_{xms} &= v_x + v_{ms} \sim N(\mathbf{0}, \mathbf{Q}_{xms}) = N(\mathbf{0}, \alpha_x \mathbf{Q}_{ms}) \end{aligned}, \qquad (17)$$

where v_{ui}, v_x, and v_{ms} is the forcing, the forward model, and the model surrogate errors, with the error covariance $\mathbf{Q}_{ui}$, $\mathbf{Q}_{\mathrm{x}}$, and $\mathbf{Q}_{\mathrm{ms}}$ respectively. The prediction errors for the model surrogate v_{ms} and for the forward model v_x are combined in to a single noise source v_{xms}, with the combined covariance $\mathbf{Q}_{xms}$.

To characterize the combined prediction error v_{xms}, we approximated the error covariance $\mathbf{Q}_{xms}$ with a scaled variance of the model surrogate error v_{ms}:

$$\mathbf{Q}_{xms} = \alpha_x \mathbf{Q}_{ms} = \alpha_x * diag(\mathrm{var}[v_{ms}]), \qquad (18)$$

where α_x is the tunable scaling parameter.

To characterize the forcings error covariance $\mathbf{Q}_{ui}$, we used a scaled version of the forcing variance that was projected on the subspace of the network weights using the internal PCA operator T_I :

$$\begin{aligned} \mathbf{Q}_{ui} &= T_i \begin{bmatrix} \mathbf{Z} & 0 \\ 0 & \mathbf{Q}_u \end{bmatrix} T_i^T \\ \mathbf{Q}_u &= \alpha_u * diag(\mathrm{var}[U]) \end{aligned}, \qquad (19)$$

where *Z* is a square zero matrix with the size equal to the length of past states *X*, and α_u is a scaling parameter that requires tuning. The projection (Eq. 19) accomplished two objectives: (i) it expedited the SPKF computation by reducing the dimension of the process-noise vector, and (ii) it introduced the temporal correlations in the forcing noise, which were encoded in the operator T_I .

The KF in this study estimated only the current time step $\hat{x}_s(k)$, with the uncertainty in the initial conditions entering a single past-time-step $x_s(k-1)$. The remaining past states in the model surrogate $[x_s(k-2);...;x_s(k-n)]$ were treated as a bookkeeping variable. The KF in our study is unlike the original reduced-dimension KF (Lu, et al. 2007) that estimated the entire time history of the present and the past states $[\hat{x}_s(k);...;\hat{x}_s(k-n)]$, which led to a fix-lag formulation of the Kalman smoother.

To summarize, uncertainty (noise) entered our modeling system (Eq. 11) in several ways. Uncertainty in the initial conditions entered into the past state $x_s(k-1)$ through the covariance matrix $\mathbf{P_{xx}}$. Uncertainty in forcings entered the entire time-embedded history of forcings U_k as a white Gaussian noise, projected to the subspace of the network weights. Finally, the uncertainty in the accuracy of the modeling system entered as a scaled version of the model surrogate error v_{ms}.

The process-noise model (Eqs. 16-19) was easy to implement and was based on simple, tunable proxies that did not require us to specify the hard-to-obtain realistic models of the forcing and model noises. In cases where the simple process-noise model (Eqs. 16-19) is not sufficient, a more sophisticated model can be developed using problem-specific information on the statistics of the process and forcing noise. For example, in our previous work (Lu, et al. 2007), we incorporated the knowledge about the autocorrelation in the forcing noise into the KF that was used to simultaneously estimate the state of the synthetic estuary and the time series of the forcing noise.

Observation noise: In section 2.3, we used a generic nonlinear observation function (Eq. 12) in the derivation of the reduced-dimension KF. In our experiments, the nonlinearity in observations was due to vertical interpolation on the sigma grid. Assuming that the reconstruction error ε was small, in comparison to the reconstructed state $\tilde{x}$, we used the following approximation to the Eq. 12:

$$\begin{aligned} &y(k) = h^s(x^s(k), n_o(k)) = \\ &= h(\tilde{x} + \varepsilon) + n_m \approx h(\tilde{x}) + h_\mu(\varepsilon) + n_m \end{aligned} \qquad (20)$$

where $\varepsilon \sim N(0, R_\varepsilon)$ and $n_m \sim N(0, R_m)$ are the reconstruction and the measurement errors, and $h_\mu(\varepsilon)$ is the observation of the reconstruction error for the mean state μ. The covariance of the reconstruction error $\mathbf{R}_\varepsilon$ was estimated as:

$$\mathbf{R}_\varepsilon = diag[\mathrm{var}(\varepsilon)]. \qquad (21)$$

We made the following best guesses for the values of the standard deviation for the measurement error n_m for: water level, 0.01 m; salinity, 0.5 psu; and temperature, 0.1 °C.

2.4.7 Calibration of process-noise models through cross-validation

To calibrate the meta-parameters α_x and α_u of the noise model (Eqs. 16-19), we used three different optimization criteria:

1) The *n*-fold cross-validation (Bishop 1995);
2) The maximum likelihood criterion of (Dee 1995); and
3) The consistency criterion described in section 2.5.5.1 (Eq. 24).

The *n*-fold cross-validation is an empirical cross-validation technique that can reuse the observational data effectively to calibrate and validate the DA algorithm. In the *n*-fold cross-validation, the dataset is divided into *n* parts. A new model is trained using *n-1* randomly selected parts (training dataset), while the remaining part is used to validate the trained model (validation dataset). The training/validation procedure is repeated *n* times for all possible combinations of the training and validation datasets. The resulting error is an average across all trained models.

Unlike the cross-validation criterion, which optimizes DA to accurately estimate the system-state at a rotating set of validation stations, the maximum likelihood and the consistency criteria optimize DA to accurately predict the statistics of the model error. That is, they adjust the parameters of the noise model until the predicted forecast error covariance matches the observed forecast error covariance, e.g., as measured at the locations of the observations.

The results of the cross-validation experiment are presented in Figure 7, with the estimated values of α_x=20 and α_u=1. To fit the noise model parameters α_x and α_u we used a four-fold (*n=4*) cross-validation technique. The errors from different sensors were normalized similarly to section 2.4.5. We first searched for the optimal value of α_x and then for the value of α_u. To optimize for data availability and to save computational resources, we ran cross-validation experiments using three weeks of observational data (05/07/2004-05/28/2004).

The calibration results based on the maximum likelihood and the consistency criteria showed good agreement with the result obtained using the cross-validation criterion, suggesting that the noise model (Eqs. 16-19) was of high-quality. Under the maximum likelihood criterion, the calibrated value of α_x was 15; and for the consistency criterion (Eq. 24), α_x was 30. As shown by the cross-validation curve for α_x (Figure 7 panel a), there was no significant difference between the values of two estimates (α_x=15 and α_x=30), since the quality of the state estimate did not change much between the two values of α_x. In the DA-experiments, presented in section 2.5, we used the parameters optimized under the cross-validation criterion, hence trading a sub-optimal forecast of the model uncertainty for a better quality of the state estimates.

2.5 Validation experiments

2.5.1 Validation strategy

To validate our DA algorithm described in sections 2.3 and 2.4, we examined how DA can improve the simulated variability of the CR estuary. Specifically, we evaluated the improvements at locations other than the locations of assimilated measurements and the improvements to the long-term hindcast simulation of the CR estuary. The assimilated data included water level, salinity, and temperature measurements, from the fixed network of sensors described in section 2.2.1. In the absence of extensive observational data from sensors independent of the fixed network, such as sensors deployed on mobile or remote sensing platforms, we used a cross-validation procedure.

The cross-validation procedure is discussed in detail in section 2.5.1.1, and is similar to the procedure that was used to calibrate the process-noise model in section 2.4.7.

We focused on answering four specific questions regarding the skill of the DA system that we developed for the CR estuary:

1) **Does our DA method improve the simulated variability of the CR estuary?** To evaluate the improvement, we used a leave-one-out cross-validation procedure to estimate the reduction of model errors across the CR estuary. The details of the procedure are documented in section 2.5.1.1.
2) **Is it possible to apply our DA method to a real-time nowcast, without retraining the model surrogate and the EOF basis?** To study this possibility, we used the leave-one-out cross-validation procedure to compare the error reduction for two DA experiments that used different model surrogates. The first model surrogate, MSA, was used in a prototype of the hindcast DA experiment, since MSA was partially trained during the period of the cross-validation study. The model surrogate MSB was used in a prototype of the nowcast DA experiment, since it was trained outside the cross-validation period.
3) **Does our DA method improve the accuracy of a long, hindcast simulation?** To evaluate such improvements, we conducted a year-long experiment, where the accuracy of the DA experiment was validated at key stations in the estuary. Specifically, we evaluated whether DA could improve the representation of tidal and sub-tidal variability of water levels, the response of salinity intrusion to changes in river and ocean conditions, and the representation of the seasonal cycle of heating and cooling in the CR.
4) **Are the DA-estimates of the CR estuary statistically and dynamically consistent?** To study the statistical consistency of the DA estimates, we checked whether the error-statistics predicted by the DA were consistent with the observed error-statistics. To study the dynamical consistency of the DA estimates, we checked whether the estimates obeyed mass- and volume-conservation laws.

To distinguish between different DA experiments, we introduced the following nomenclature. The baseline DA experiment, which used all available data, was denoted *DA-all*. For validation experiments, we used a three-part abbreviation *DA-(abbreviated variable name)-(station name)* to indicate the nature of data that were withheld for validation. For example, the name *DA-wl-tpoint* denoted the validation experiment where the water level (*wl*) data from the station *tpoint* were withheld for validation.

2.5.1.1 Using cross-validation to validate DA experiments

A cross-validation procedure is a machine-learning technique (Bishop 1995) that is well suited for estimating the accuracy of a DA solution across model variables and throughout a computational domain. Obtaining such estimates can be difficult due to the shortage of independent validation data. For example, the observations that are used to constrain the DA experiment cannot be used for the independent validation of the same DA experiment. It is common to resolve this difficulty by dividing all available data into assimilation and validation datasets. However, this division results in using fewer data in the assimilation dataset, which can degrade the quality of the assimilated solution, especially in cases where the observations come from a sparse observation network.

A leave-one-out cross-validation is a validation procedure that can effectively reuse observational data for both assimilation and validation. This reuse is achieved by averaging the validation errors over multiple DA experiments. In each experiment, data from only a single sensor are withheld for the validation, hence leading to a minimal impact on the accuracy of the assimilated solution. Such multiple DA experiments are too computationally expensive in the context of traditional DA algorithms that use expensive-to-compute forward and adjoint models. However, our DA method is driven by a fast model surrogate, which leads to low computational cost of the cross-validation study.

A concern with using the cross-validation criterion to measure the DA accuracy is that errors at the validation stations may be strongly correlated with the errors at assimilation stations, hence providing poor candidates for independent validation of the

DA accuracy. However, from the design-of-experiment literature (Fedorov 1972, Bretherton, et al. 1976), we know that the strength of the error correlations is an indication of the density of the observation network. For example, in the CR estuary some stations are likely to have stronger error correlations than others, indicating better-sampled areas in the estuary. The cross-validation addresses the problem of estimating DA-accuracy across the well-sampled and the under-sampled areas by averaging the error over multiple validation experiments. Each of these experiments validates the DA accuracy at a different station, located in different areas of the estuary.

To illustrate how the error-correlation strength varied between different validation stations, we used the R^2 statistic[9]. This statistic measures the percentage of the error-variance at a validation station, explained by the cross correlation with errors at assimilation stations. Figure 8 presents the variability in R^2 values as an empirical cumulative distribution function.

Results in Figure 8 show that our observational array had the strongest error correlation for water level measurements (average R^2=0.93) and had the weakest error correlation for salinity measurements (average R^2=0.37). The R^2 values varied strongly among different stations in the CR estuary. In the case of salinity measurements, the R^2 value varied from 0.16 for the surface sensor at *am169* to 0.62 for the mid-water sensor at *am169*. The strong error correlation for water levels, as compared to the weak error correlation for salinity, indicated an under-sampled salinity-error field, as compared to the well-sampled water-level-error field.

2.5.2 Overall improvement in simulated variability of the CR estuary

To evaluate the overall accuracy of our DA system, we applied the leave-one-out cross-validation procedure to a three-week-long time interval (05/07/2004–05/21/2004), which was chosen to simplify the analysis of the DA accuracy by minimizing the impact of gaps in observational data. The results of the cross-validation study are presented in two tables (1 and 2). Table 1 presents the results in terms of RMS errors, while Table 2

[9] The R^2 statistic was computed using the best linear unbiased estimator as described in (Fedorov 1994). The empirical error covariance matrix was computed from the time series of the model surrogate errors.

presents the errors as a percentage of the data variance. The cross-validation results are presented for the locations of assimilation stations (columns D and F in both tables) and for the locations of validation stations (columns E and G). The error at validation stations was averaged over all validation experiments. In this section, we present the results of the hindcast DA experiment (columns D and E), which was based on the model surrogate MS_A, described in section 2.4.4. For a comparison between the hindcast and the nowcast experiment, see section 2.5.3.

The cross-validation study showed that DA effectively reduced errors in water levels, salinities, and temperatures at both the assimilation stations (column D) and the validation stations (column E). As expected, our results showed little improvement in velocity-magnitude errors, since observations of water velocity were not used in the DA experiments. At the assimilation stations (column D), the average error (over all stations and time) was reduced from 0.14 m to 0.06 m for water levels, from 4.3 psu to 1.2 psu for salinity, and from 1.2 °C to 0.5 °C for temperature. When the errors were expressed as the percentage of the data variance at stations, the errors for water levels were reduced from 2.5% to 0.5%, for salinity from 13% to 1.1%, and for temperature from 0.7% to 0.1%. As expected, the errors after DA at the assimilation stations were at or below the combined observation error n_o, which varied for water levels from 0.03 m to 0.06 m, for salinity from 0.5 psu to 2.0 psu, and for temperature from 0.12 °C to 0.34 °C. At the validation stations (column E), the reduction in errors was smaller than at the assimilation stations (column D), but was still substantial: for water levels from 0.15 m to 0.10 m (from 2.5% to 1.2%), for salinity from 4.3 psu to 3.0 psu (from 13.0% to 6.1%), and for temperature from 1.2 °C to 0.7 °C (from 0.7% to 0.3%).

2.5.3 Comparison of error reduction in the nowcast and hindcast experiments

To evaluate the ability of our DA algorithm to assimilate data in a real-time nowcast, we compared the performance of DA experiments based on two different model surrogates, described in section 2.4.4. The model surrogate MS_A was used in a prototype of the hindcast DA experiment, since MS_A was partially trained during the period of the

cross-validation study. The model surrogate MS_B was used in a prototype of the nowcast DA experiment, since it was trained outside of the cross-validation period.

The comparison of cross-validation errors in Tables 1 and 2 suggests that the nowcast DA experiment (columns F and G) was as accurate as the hindcast DA experiment (columns D and E). The result of this comparison is of significance for the operational applications of our DA method, since this result shows that the reduced-dimension KF can be successfully applied in a real-time nowcast without the need to retrain the model surrogate and the EOF on a new set of data.

2.5.4 Improvements in the simulated variability of a long-term hindcast experiment

To determine whether our DA improved the simulated variability of a long-term hindcast simulation, we analyzed the accuracy of the simulated tides, salinity intrusion, and temperature in a one-year-long hindcast simulation of the CR estuary.

2.5.4.1 Tidal variability

Tidal variability in the CR estuary is nonlinear and non-stationary (Jay, et al. 1997); it is governed by the nonlinear interactions between several external forcings, such as ocean tides, ocean set-up due to atmospheric influences, bottom-friction forces, and river discharge. The uncertainties in these forcings can lead to errors in the simulated tides. To study whether DA can reduce the tidal errors, we performed the DA experiment *DA-wl-tpoint*, where the water level data were assimilated at all stations, but *tpoint*, which was used for an independent validation.

The results from the validation experiment *DA-wl-tpoint* showed that the DA improved the representation of the tides in the CR estuary, as seen in Figure 9, which compares the maximum daily errors at the validation station *tpoint* for DB16 and the DA experiment *DA-wl-tpoint*. The water level errors were reduced from 0.14 m to 0.07 m for the all-pass band, from 0.10 m to 0.04 m for the tidal band (T<30 h), and from 0.10 m to 0.06 m for the sub-tidal band (T>30 h). While the error reduction in the tidal band was consistent throughout the experiment, the reduction in the sub-tidal band was most

prominent during winter storms (January-March and October-December). The exceptions to this effective error-reduction were two short periods, marked with arrows in Figure 9, when most of the water-level sensors failed, and when the errors in the DA experiment were similar to the errors of DB16.

2.5.4.2 Salinity intrusion

To determine whether our DA method improved upon the simulated salinity in DB16, we studied the errors in simulated salinity at four distinct regions of the CR estuary: the mouth of the estuary, the mid-estuary, the upper estuary, and in a small lateral channel of the estuary. For each region, the results of two DA experiments were analyzed. In the first experiment, all available data were assimilated (*DA-all*). In the second experiment, the data at one of the stations in the region were withheld for validation. To present the long-term variability of salinity at the validation stations, we used the daily salinity-means as a proxy statistics in Figure 10. The daily-averages were computed over a tidal day of 30 hours.

The results of the validation experiments showed that DA improved the simulated salinity at three regions of the CR estuary: in the mouth of the estuary, in the mid-estuary, and in the small lateral channel. When all data were used in assimilation, the simulated variability improved substantially, including the reduction in the model bias and the better representation of episodic events. When the validation data were withheld from the assimilation, the improvements at these validation stations were modest. The strongest improvement was at the mid-estuary station *am169*, which suggests that the observation network was more informative in the middle of the estuary than in the mouth or in the lateral channel of the estuary.

In the fourth region—the upper estuary—the DA method failed to improve the simulated salinity, possibly due to the limitation of the EOF subspace, where our KF was defined. In fact, the EOF space, trained on the forward model, encoded the upper estuary as permanently fresh, since the forward model (DB16) failed to propagate any salinity to the upper estuary. As a result, the KF, defined in this subspace, also failed to estimate

nonzero salinity in the upper estuary. However, the region of the upper estuary, where the KF failed, was relatively small, in comparison to the rest of the estuary, where the KF succeeded in improving the simulated salinity. The detailed analysis of the DA results follows.

Mouth of the CR estuary: To evaluate the simulated salinity in the mouth of the CR estuary, we used the salinity data from the bottom sensor at station *jetta*. In addition to the strong tidal signal (not shown in Figure 10), the changes in ocean upwelling, spring-neap tidal cycle and river discharge all influenced the salinity signal at *jetta*. Examples of these influences can be seen in the daily-mean salinity signal displayed in Figure 10, panel I.

DB16 (Figure 10.I.a) represented the tidal variation in the salinity signal at *jetta* relatively well, with an RMS error of 4.5 psu and a bias of -2.3 psu, which indicated that the model was on average too saline. After the assimilation, in the DA experiment *DA-all*, the simulated salinity at *jetta* was greatly improved, with an RMS error of 1.8 psu and a bias of -0.6 psu. Specifically, the assimilated solution improved the tracking of the episodic variations in the daily-mean salinity. An example of such improvement was a series of events during May-June of 2004 (Figure 10.I.b). In the validation experiment *DA-salt-jetta* (Figure 10.I.c), where the data from *jetta* were withheld from the assimilation and used as the validation data, the simulated salinity at *jetta* improved modestly, with an RMS error of 4.3 psu and a bias of -1.7 psu.

Mid-estuary: To evaluate the simulated salinity in the middle of the CR estuary, we used the salinity data from the bottom sensor at station *am169*. The salinity variability at *am169* had a strong non-stationary component, which was defined by the interaction of tides with river discharge and ocean conditions. For example, the daily-average salinity in Figure 10.II.a had a strong spring/neap tidal cycle, which was modulated by non-stationary changes in the river discharge. An example of this modulation was the period of higher minimum salinity (~10 psu vs. 0 psu) that coincided with the period of the low

river discharge in the months of July through November, e.g., as seen in the visual comparison of panels V and II in Figure 10.

RMS errors for DB16 at station *am169* (RMSE=6.9 psu) were dominated by a model bias of -5.1 psu, which suggests that the simulated salinity in DB16 was too saline at *am169* (Figure 10.II.a). The model bias and the RMS error were greatly reduced in both of the DA experiments: the *DA-all* (bias=0.2 psu, RMS=1.0 psu) and the *DA-salt-am169* (bias=0.2 psu, RMS=4.6 psu). In the validation experiment *DA-salt-am169*, all salinity measurements collected at station *am169* were excluded from the assimilation to avoid their influence on the error reduction at the bottom senor, which was used for the validation.

Lateral channel of the CR estuary: To evaluate the simulated salinity in the lateral channel of the CR estuary, we used the salinity data from the bottom sensor at station *cbnc3*. During our study, salinity reached *cbnc3* only episodically, usually during periods of the very low river-discharge, as seen in August-October in Figure 10.III.a.

DB16 represented the episodic nature of the salinity intrusion at *cbnc3* correctly. However, DB16 failed to simulate the correct magnitude and timing of the salinity-intrusion events. In the DA experiment *DA-all,* the salinity intrusion events were represented better than in DB16, as was measured by a reduction in an RMS error (from 1.5 psu to 0.6 psu) and a model bias (from 0.7 psu to 0.0 psu). The validation experiment *DA-salt-cbnc3* improved the salinity intrusion at *cbnc3* modestly; the examples of such improvements were two isolated events in March of 2004, Figure 10.III.c. These modest improvements are likely to indicate that the lateral channel, where *cbnc3* was located, was under-sampled, as compared to the main channel, where *am169* was located.

Upper CR estuary: To evaluate the simulated salinity in the upper CR estuary, we used the salinity data from the bottom sensor at station *eliot*. Similar to *cbnc3*, salinity reached *eliot* only episodically. However, since *eliot* was located further upstream, these events were even less frequent than at *cbnc3*.

During our study, DB16 failed to propagate any salinity to station *eliot*; see Figure 10.IV.a. Assimilating salinity measurements from station *eliot* did not improve the simulated salinity either; see Figure 10.IV.b.

To further illustrate the ability of the DA to control the length and the shape of the salt wedge, we compared the transects of model-salinity from DB16 and the DA experiment *DA-all*. The transects were taken along the southern channel of the estuary, during a period (September 20-27, 2004) of higher model error. In DB16 the simulated estuary was over stratified and too saline at station am169, as seen in Figure 11.1.a. After DA, the stratification at *am169* was corrected, as shown in Figure 11.2.a, resulting in a more mixed estuary, as shown in Figure 11.2.b.

2.5.4.3 Temperature variability

To determine whether DA improved upon the simulated temperature in DB16, we studied three regions of the CR estuary: the mouth of the estuary, the mid-estuary, and a small lateral channel of the estuary. The results of the validation studies showed that DA improved the temperature representation at all three regions of the estuary. In the mouth of the estuary and in the mid-estuary, DA improved the representation of cold water intrusion events. In the small lateral channel, DA further improved the representation of the annual temperature trend. The detailed analysis of the validation experiments for each of the regions follows.

Mouth of the CR estuary: To evaluate the simulated temperature in the mouth of the CR estuary, we used temperature data from the bottom sensor at station *jetta*. In Figure 12.I, the effect of coastal upwelling is highly visible in the time series of daily-mean temperature. For example, during the summer months of June through October the maximums of the events are shown with arrows. The timing of these events correlated with the relaxation of the upwelling-favorable winds, as seen in the time series of the North-South winds off-shore of the CR mouth (Figure 12.IV).

DB16 tracked the annual temperature trend well, but failed to track the mean temperature during the episodes of ocean upwelling; see Figure 12.I.a. In both DA experiments, the *DA-all* and the *DA-temp-jetta*, the tracking of the upwelling events have improved, for example as seen in the summer months of June through October of 2004 (Figure 12.I.b and 12.I.c). However, an RMS error improved only for the experiment *DA-all* (reduction from 1.2 °C to 0.4 °C).

Mid-estuary: To evaluate the simulated temperature in the middle of the CR estuary, we used the temperature data from the bottom sensor at station *am169*. The observational data indicated a strong influence of both the ocean and the river water at the location of *am169*. The influence of the ocean water at *am169* was seen in several episodes, when the cold ocean water lowered the daily-mean temperatures during the months of May trough October (blue circles in Figure 12.II.a). The influence of the river water was seen in the annual trend of the temperature, which varied at the river boundary from ~5 °C in winter to ~25 °C in summer.

Unlike the observational data (blue circles in Figure 12.II.a), the simulated temperature in DB16 (red line Figure 12.II.a) was dominated by the annual trend in river temperature, with no substantial influence of the cold ocean water in the months of May trough October. In both DA experiments, the *DA-all* and the *DA-temp-am169,* the tracking of the cold-temperature events improved. RMS errors were reduced from 2.4 °C, for DB16, to 1.4 °C, for the *DA-all* experiment*,* and to 1.9 °C, for the *DA-temp-am169* experiment.

Lateral channel of the CR estuary: To evaluate the simulated temperature in the lateral channel of the CR estuary, we used the temperature data from the bottom sensor at station *cbnc3*. The temperature signal at *cbnc3* was dominated by the temperature of the river water, including the influences of atmospheric heating and cooling as the river water reached the tidal flats, where *cbnc3* was located.

DB16 (red line in Figure 12.III.a) repressed the annual temperature-trend at *cbnc3* relatively well (RMS error of 1.4 °C). However, DB16 failed to accurately capture the observed daily-mean temperature (blue circles in Figure 12.III.a), possibly due to the misrepresentation of the heat exchange with the atmosphere. This misrepresentation was exemplified by the warmer model temperatures in the winter and the colder model temperatures in the summer, as can be seen in Figure 12.III.a. In both DA experiments, the *DA-all* and the *DA-temp-cbnc3,* the simulated daily-mean temperature at *cbnc3* improved substantially. The improvement was measured by reduced RMS errors, which reduced from 1.4 °C, for DB16, to 0.4 °C, for the *DA-all* experiment, and to 0.6 °C, for the *DA-temp-cbnc3* experiment.

2.5.5 Consistency of assimilated results

2.5.5.1 Statistics of the innovation errors

To determine whether the estimate of error statistics predicted by the DA was statistically consistent with the measured error statistics, we compared the predicted error variance for the state forecast $\hat{x}_s^-(k)$

$$\hat{\sigma}^2_{inov}(k) = trace[\mathbf{P}^-_{yy}(k)]/m \qquad (22)$$

with the measured variance of the innovation errors

$$\sigma^2_{inov}(k) = \mathrm{var}[y(k) - \hat{y}^-(k)], \qquad (23)$$

where *y(k)* are the observational data, $\hat{y}^-(k)$ are the observations of the state forecast (Eq. 14), $\mathbf{P}^-_{yy}$ is the covariance of the innovation (Eq. 14) predicted by the KF, and *m* is the number of measurements. A consistent KF should on average satisfy the following equality (Bar-Shalom, et al. 1988):

$$\hat{\sigma}^2_{inov}(k) = \sigma^2_{inov}(k). \qquad (24)$$

For the DA experiment *DA-all*, the time average of the measured error variance σ^2_{inov} =0.3 exceeded the predicted error variance $\hat{\sigma}^2_{inov}$ =0.2, indicating a slightly inconsistent KF. The calibration experiments in section 2.4.7 showed that it was possible to satisfy the criterion (Eq. 24) at all times, by optimizing parameters in the process-noise model (Eq. 16). However, we decided against such optimization, since further

optimization led to the slightly degraded accuracy of the state estimates. See section 2.4.7 for the results of the cross-validation study.

The gaps in the observational data, discussed in section 2.2.1, introduced an additional source of uncertainty to the analysis of data presented in Figure 13. For example, during the months of June-August, the predicted error variance was underestimated, as shown by the gray line in Figure 13. However, it was during the same time period that most of the salinity sensors in the CR estuary failed, as shown by the gray line in Figure 2. It is unclear if the decrease in the predicted error variance in Figure 13 is attributable to the changes in the error statistics of the KF, or it is attributable to the changes in the available observational data that were used to estimate the predicted error variance.

2.5.5.2 Conservation laws

To determine whether the DA estimates were dynamically consistent, we evaluated the conservation properties of the DA estimates. Obeying conservation laws, such as mass, volume, momentum, and transport conservation, is an important attribute of a numerical algorithm and is crucial for the conservative transport of scalar tracers like salinity, nutrients, and pollutants. The conservation of mass is enforced in a forward model, like SELFE, by the deliberate design of the numerical algorithms. However, at the time of experiments, neither the EOF projection, nor the model surrogate, nor the DA algorithm enforced the conservation laws.

To quantify the conservation errors in the DA estimates, we concentrated on volume conservation and transport conservation laws. A numerical algorithm is called volume-conservative when the flux through the boundary of a control volume is equivalent to the change in the volume of a control volume (via movement of the free surface). A numerical algorithm is called transport-conservative when the mass-flux of a tracer (e.g., salinity) trough the boundary of a control volume is equivalent to the change in the mass of a tracer occurring inside of a control volume. To evaluate the conservation

quantities in the year-long DA experiment *DA-all*, we used the entire domain of DB16 as the control volume.

Volume conservation was well preserved in SELFE with a low[10] conservation error of 0.7%. Unlike the states in the SELFE model, the states reconstructed from the EOF projection and from the DA estimate had a slightly larger error: 1.6% for the EOF and 6% for the DA experiment. The larger error for the DA experiment is a likely indicator of inconsistencies between the water levels that experienced strong corrections from the DA and the velocities that experienced little change from the DA.

Transport conservation was also well preserved in SELFE, leading to a small conservation error of 1.3%. The states reconstructed from the EOF projection had a similar error of 1.4%, possibly indicating a very low reconstruction error for the velocity and the salinity fields. Unlike DB16 and the EOF, the DA results had a larger conservation error of 15%.

2.6 Conclusions

In applying the reduced-dimension KF to a realistic simulation of the CR estuary, we determined that:

1) The novel framework of the reduced-dimension KF can be applied to estimate the state of a complex, nonlinear coastal circulation, such as the CR estuary;
2) In the CR estuary, our DA method improved upon the non-assimilative model in simulating the long-term variability of tides, salinity, and temperature;
3) The reduced-dimension KF was successful in a real-time nowcast of the CR estuary, without the need to retrain the model surrogates and the EOF basis;
4) Implementing our DA method using the reduced-dimension KF led to substantial computational and logistical economies. Computationally, our DA method was ~100 times faster than the forward model and ~100-10,000 times faster than most

[10] The conservation error was even lower when it was computed using model variables internal to the SELFE numerics (e.g., the velocities defined at side-centers and the water levels at element-centers). However, only the external representation of these variables was available for our experiments (e.g., defined on the nodes of the computational grid), which led to a higher conservation-error.

existing DA algorithms. Logistically, our DA method did not require a tedious development of the tangent linear and adjoint models.

Acknowledgments:

We thank the anonymous reviewers, whose comments helped us to improve the quality of this paper. We thank Michael Wilkin for his work on maintaining the CORIE observational network, Dr. Yinglong Zhang for his work on the SELFE model and DB16, and Dr. Sandra Oster for her help with the text of the paper.

The National Science Foundation (ACI-0121475, OCE-0424602) and National Oceanic and Atmospheric Administration (AB133F-04-CN-0033) provided financial support for this research. Any statements, opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views or policies of the federal sponsors, and no official endorsement should be inferred.

Appendix A1: Algorithm of the Sigma Point Kalman Filter

The SPKF algorithm[11] is a straightforward application of the sigma point transformation to the calculation of the optimal terms (Eq. 14) in the KF recursion (Eq. 13), where the state of the system is redefined as an augmented state $x^a(k) = [\hat{x}_s(k); v(k); n_o(k)]$. Below we present the algorithm for a generic SPKF filter:

- Initialization:

$$\hat{x}_s(0) = E[x_s(0)], \mathbf{P}_{xx}(0) = E[(x_s(0) - \hat{x}_s(0))(x_s(0) - \hat{x}_s(0))^T]$$
$$\hat{x}^a(0) = E[x^a(0)] = [\hat{x}_s(0); \overline{v}(0); \overline{n}(0)]$$
$$\mathbf{P}^a_{xx}(0) = E[(x^a(0) - \hat{x}^a(0))(x^a(0) - \hat{x}^a(0))^T] =$$
$$= \begin{bmatrix} \mathbf{P}_x(0) & 0 & 0 \\ 0 & \mathbf{Q} & 0 \\ 0 & 0 & \mathbf{R} \end{bmatrix}$$

- Kalman filter loop k=1..end

[11] The description of the SPKF algorithm is adopted from (van der Merwe, et al. 2003, van der Merwe 2004).

1) Calculate sigma points

$$\mathcal{X}^a(k-1) = [\hat{x}^a(k-1), \hat{x}^a(k-1) + \varsigma\sqrt{\mathbf{P}^a(k-1)}, \hat{x}^a(k-1) - \varsigma\sqrt{\mathbf{P}^a(k-1)}]$$

2) Time-update equations

$$\mathcal{X}^x(k) = f^s(\ddot{\ddot{\mathcal{X}}}^x(k-1), U_k, \mathcal{X}^v(k))$$

$$\hat{x}_s^-(k) = \sum_{i=0}^{2L} w_i^m \mathcal{X}_i^x(k)$$

$$\mathbf{P}_{xx}^-(k) = \sum_{i=0}^{2L}\sum_{j=0}^{2L} w_{ij}^c (\mathcal{X}_i^x(k))(\mathcal{X}_j^x(k))^T$$

3) Measurement-update equations

$$\mathcal{Y}^x(k) = h^s(\mathcal{X}^x(k), \mathcal{X}^n(k))$$

$$\hat{y}^-(k) = \sum_{i=0}^{2L} w_i^m \mathcal{Y}_i\ (k)$$

$$\mathbf{P}_{yy}^-(k) = \sum_{i=0}^{2L}\sum_{j=0}^{2L} w_{ij}^c (\mathcal{Y}_i\ (k))(\mathcal{Y}_j\ (k))^T$$

$$\mathbf{P}_{xy}^-(k) = \sum_{i=0}^{2L}\sum_{j=0}^{2L} w_{ij}^c (\mathcal{X}_i^x(k))(\mathcal{Y}_j\ (k))^T$$

$$\mathbf{K}_k = \mathbf{P}_{xy}(k)\mathbf{P}_{yy}^{-1}(k)$$

$$\hat{x}_s(k) = \hat{x}_s^-(k) + K_k(y(k) - \hat{y}^-(k))$$

$$\mathbf{P}_{xx} = \mathbf{P}_{xx}^- - K_k\mathbf{P}_{yy}K_k^T$$

- Parameters: $\mathcal{X}^a = [\mathcal{X}^x; \mathcal{X}^v; \mathcal{X}^n]$ is the partitioning of the sigma-point matrix, ζ is a scaling parameter that determines the spread of the sigma points, L is the dimension of the augmented state, and w_i^m and w_{ij}^c are the scalar weights. Parameters ζ, w_i^m, and w_{ij}^c depend on the chosen form of the SPKF.
- Notes: the state variable $\ddot{\ddot{\mathcal{X}}}^x(k-1) = [\mathcal{X}^x(k-1); \hat{x}_s(k-2); ...; \hat{x}_s(k-n)]$ is introduced to keep the record of past state estimates $[\hat{x}_s(k-2); ...; \hat{x}_s(k-n)]$, required by the model surrogate (Eq. 11). The described KF filter only estimates current state $\hat{x}_s(k)$ and does not re-estimate the past states $[\hat{x}_s(k-1); ...; \hat{x}_s(k-n)]$ in the bookkeeping variable $\ddot{\ddot{\mathcal{X}}}^x$.

Appendix A2: EOF algorithms for large datasets

Random sub-sampling: A simple strategy for approximating the EOF subspace of a large dataset is based on the following random sub-sampling technique:

1) Compute the approximation of the EOF basis $\hat{\mathbf{\Pi}}'$ and the mean state $\hat{\mu}'$ based on a small random sample of states (e.g., a 1,000 out of 100,000):

$$\hat{\mu}' = mean(x(j))$$
$$\hat{\mathbf{\Pi}}'\mathbf{S}\mathbf{V}^{\mathbf{T}} = svd([..., x(j) - \mu', ...]), \quad \text{(A.1)}$$

where j is the index over a random sample of states;

2) Project the entire database of states into the approximated subspace $\hat{\mathbf{\Pi}}'$:

$$x_s(i) = \hat{\mathbf{\Pi}}'\left(x(i) - \hat{\mu}'\right); \quad \text{(A.2)}$$

3) Compute the orthonormal rotation of the approximated subspace $\hat{\mathbf{\Pi}}'$ by decorrelating the time series of the projected states:

$$\mu_s = mean(x_s(i))$$
$$\mathbf{U}\mathbf{S}\mathbf{V}^{\mathbf{T}} = svd([..., x_s(i) - \mu_s, ...])$$
$$\hat{\mathbf{\Pi}} = \hat{\mathbf{\Pi}}'\mathbf{U}; x_s(i) = \mathbf{S}\mathbf{V}^{\mathbf{T}}; \mu = \hat{\mu}' - \hat{\mathbf{\Pi}}^T\mu_s. \quad \text{(A.3)}$$

The resulting EOF operator $\hat{\mathbf{\Pi}}$ is a true EOF in the subspace spanned by its columns. That is, the EOF modes of operator $\hat{\mathbf{\Pi}}$ are oriented along the major axis of variability in this subspace.

Although the SVD computation (Eq. A.1) of a small random sample is feasible, it may remain a computational bottleneck for the EOF algorithm (Eqs. A.1-3). For example, consider a case where, due to computational limitations, the random sample used in SVD computation (Eq. A.1) is too small and not statistically representative of the entire simulation database. In this case, the spanning subspace $\hat{\mathbf{\Pi}}'$ would be of poor quality, which in turn would lead to a poor approximation of the global EOF.

Divide-and-conquer EOF algorithm: To address the computational bottleneck discussed above, we devised a divide-and-conquer EOF (DC-EOF) algorithm similar to the algorithm of (Xue, et. al. 1997). The DC-EOF divides the large SVD problem (Eq. A.1) into smaller sub-problems (e.g., by variable type or by geographic region) and then

joins the smaller sub-problems using a secondary EOF. Since each of the sub-problems is smaller than the original EOF problem A.1, more samples can be used to compute the EOF of each sub-problem, leading to the improved representation of the spanning subspace. The drawback of the DC-EOF algorithm comes from ignoring the cross-correlations among reconstruction errors of each sub-problem, which can be tolerated when the reconstruction errors for each sub-problem remain small.

The DC-EOF algorithm consists of three parts. Where appropriate, we use the MATLAB© notation for matrix operations. The MATLAB© code for this algorithm is available on request.

1) Compute EOFs for *n* sub-problems: Assume that we have a large, zero-mean dataset **D** of column vectors, and we have *n* indexes idx_{bi} that determine the division of the large dataset into smaller sub-problems $\mathbf{D}_{bi}$. We also have temporal index idx_t that determines random sub-sampling of the large dataset **D**. Then the datasets for each sub-problem can be abbreviated as:

$$\mathbf{D}_{bi}^{t1} = \mathbf{D}(idx_{bi}, idx_t). \qquad \text{(A.4)}$$

The length of the temporal index idx_t is selected to allow for practical computation of the EOF analysis using the SVD algorithm of choice. The first part of the DC-EOF is completed by computing *n* dimension-reduction operators $\mathbf{\Pi}_{bi}$ using EOF decomposition:

$$\mathbf{\Pi}_{bi} \leftarrow \mathrm{EOF}(\mathbf{D}_{bi}^{t1}). \qquad \text{(A.5)}$$

2) Assemble intermediate results:

$$\mathbf{D}_{sb} = \mathbf{\Pi}_{sb}\mathbf{D} \Leftrightarrow$$

$$\Leftrightarrow \begin{bmatrix} \mathbf{\Pi}_{b1}\mathbf{D}_{b1} \\ \vdots \\ \mathbf{\Pi}_{bn}\mathbf{D}_{bn} \end{bmatrix}^T = \begin{bmatrix} \mathbf{\Pi}_{b1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{\Pi}_{bn} \end{bmatrix} \begin{bmatrix} \mathbf{D}_{b1} \\ \vdots \\ \mathbf{D}_{bn} \end{bmatrix}, \qquad \text{(A.6)}$$

where $\mathbf{\Pi}_{sb}$ is the intermediate EOF operator, and $\mathbf{D}_{sb}$ is the intermediate dataset.

The intermediate reconstruction error is defined as:

$$\varepsilon_{sb} = \left(E\left[(\mathbf{D} - \mathbf{\Pi}_{sb}{}^{T} \mathbf{D}_{sb})^2 \right] \right)^{1/2} . \quad \text{(A.7)}$$

Formulas A.5 and A.6 can be computed for each individual block $_{bi}$, without assembling the blocked matrix $\mathbf{\Pi}_{sb}$ explicitly. Similarly, the expectation operator E[.] can be computed incrementally.

3) Compute secondary EOF and assemble final result: The result (A.6) is already an admissible dimension reduction (e.g., $\mathbf{\Pi}_{sb}$ is orthonormal), and in our case it yields a considerable reduction in the degrees of freedom (from ~10^6 to 10^3). However, this dimension-reduction is not yet efficient, since there are correlations among individual sub-problems; that is, rows of $\mathbf{D}_{sb}$ are correlated.

To achieve an efficient dimension reduction, a secondary EOF is computed:

$$\mathbf{\Pi}_A \leftarrow \text{EOF}(\mathbf{D}_{sb})$$
$$\mathbf{D}_s = \mathbf{\Pi}_A \mathbf{D}_{sb} = \mathbf{\Pi}_A (\mathbf{\Pi}_{sb} \mathbf{D}) = \mathbf{\Pi} \mathbf{D} , \quad \text{(A.8)}$$

where $\mathbf{\Pi}_A$ is the secondary EOF operator and $\mathbf{D}_s$ is the final dataset in the reduced space defined by the global EOF operator $\mathbf{\Pi}$.

The new reconstruction error for the operator $\mathbf{\Pi}$ is defined as:

$$x = \mathbf{\Pi}_{sb}{}^{T} (\mathbf{\Pi}_A{}^{T} x_s + \varepsilon_A) + \varepsilon_{sb} =$$
$$= \mathbf{\Pi}^T x_s + \mathbf{\Pi}_{sb}{}^{T} \varepsilon_A + \varepsilon_{sb} = \mathbf{\Pi}^T x_s + \varepsilon , \quad \text{(A.9)}$$

where x and x_s are the state vectors that are elements of the datasets $\mathbf{D}$ and $\mathbf{D}_s$.

The output of the algorithm is the global EOF operator $\mathbf{\Pi}$ and the global reconstruction error ε:

$$\mathbf{\Pi} = \mathbf{\Pi}_A \mathbf{\Pi}_{sb}$$
$$\varepsilon = \mathbf{\Pi}_{sb}{}^{T} \varepsilon_A + \varepsilon_{sb} . \quad \text{(A.10)}$$

All products with the blocked matrix $\mathbf{\Pi}_{sb}$ can be arranged without assembling $\mathbf{\Pi}_{sb}$ explicitly.

Appendix A3: Statistics of the innovation signal

The statistics of the innovation signal:

$$\delta y^-(k) = y(k) - \hat{y}^-(k), \quad \text{(A.11)}$$

where $\hat{y}^-(k)$ is the observation of the DA prior, can provide insights to the optimality of the DA system {Bar-Shalom, 1988 #84; Bennett, 2002 #60}. In a statistically consistent DA system the innovation vectors δy^- should be zero-mean, temporally uncorrelated, and drawn from the normal distribution

$$\delta y^- \sim \mathrm{N}(0, \mathbf{P}_{yy}^-), \quad \text{(A.12)}$$

where $\mathbf{P}_{yy}^-$ is the covariance of the prior innovations estimated by the DA system (Eq. 14). In this appendix, we determine whether the innovations statistics δy^- in our DA experiments were zero-mean, temporally uncorrelated, and were drawn from the distribution in Eq. A.12.

To verify that the innovation sequences δy^- were zero-mean, we compared the mean innovation from the non-assimilated model $E\left[\delta y^{non-assim.}\right]$ (column B in Table 3) with the mean innovations of DA prior $E\left[\delta y^-\right]$ (column C in Table 3). Column B shows that that, in violation of DA assumptions, the non-assimilated model was biased, with non-zero mean innovation $E\left[\delta y^{non-assim.}\right]$. Since our DA was not designed to account for a bias in the non-assimilated model, it would be too optimistic to expect that the innovations for the DA prior be unbiased. Table 3 shows that, in fact, the mean innovations after DA (column C) were small but often non-zero; however, these mean innovations after DA were also consistently smaller than the mean innovations before DA (column B).

To verify that the innovation sequences δy^- were temporally uncorrelated, we compared in Table 3 the temporal decorrelation scales T_D of the innovations in non-assimilated model $T_D\left[\delta y^{non-assim.}\right]$ (column D) with the decorrelation scales of

innovations in the DA prior $T_D\left[\delta y^-\right]$ (column E). The decorrelation scales T_D were estimated from the lag-time of the first zero crossing in the vector autocorrelation function:

$$c(l)=\sum_{k=1}^{K}\left[\left(\delta y(k)-\overline{\delta y}\right)\left(\delta y(k-l)-\overline{\delta y}\right)^T\right]* \left[\sum_{k=1}^{K}\left[\left(\delta y(k)-\overline{\delta y}\right)\left(\delta y(k)-\overline{\delta y}\right)^T\right]\sum_{k=1}^{K}\left[\left(\delta y(k-l)-\overline{\delta y}\right)\left(\delta y(k-l)-\overline{\delta y}\right)^T\right]\right]^{-1/2}, \quad \text{(A.13)}$$

where *c(l)* is the autocorrelation function for the time-lag l and *K* is the number of time instances in the vector time series of innovations δy.

Table 3 shows that after DA the temporal correlations were often shorter, indicating that the information from observations were extracted by DA, leading to an increase in the skill of the model. Specifically, for water levels, decorrelation scales decreased from 31 hours, which is twice the 12.4 hour period of the dominant M^2 tide, to only 4 hours, which is shorter than the period of M^2 tide. For salinity and temperature, we computed decorrelation scales separately for ocean stations, which were strongly influenced by the ocean, and for river stations, which were strongly influenced by the river and only occasionally by the ocean. After DA, the decorrelation length scale for ocean salinity sensors was 5.5. hours, which is shorter than the period of M^2 tide. For river salinity sensors, decorrelation scale was 5.5 days, which is shorter than the spring-neap cycle. For ocean temperature sensors, decorrelation scale decreased from 12 weeks, which is comparable to the seasonal cycle, to 1 week, which is shorter than the spring-neap cycle. Finally, for river temperature sensors, decorrelation scale remained at 12 weeks, which is shorter than the seasonal cycle of solar heating.

To verify that the innovation sequences δy^- were drawn from a normal distribution consistent with the predicted innovation covariance $\mathbf{P}_{yy}^-$, we considered the random variable:

$$X^2=\delta y^-\left[\mathbf{P}_{yy}^-\right]^{-1}\left[\delta y^-\right]^T, \quad \text{(A.14)}$$

which should be distributed according to a χ^2 distribution with *m* degrees of freedom, where *m* is the number of observations {Bar-Shalom, 1988 #84}. The time series of the X^2 variable, normalized by the number of observations *m*, are shown in Figure 14.a. To facilitate visual comparisons, all the time series in Figure 14 were filtered, with fluctuations shorter than one week removed. On average, the normalized variable (X^2/m) should be equal to unity, with 98% of fluctuations captured between ~0.5 and ~1.8 normalized units. As Figure 14.a shows the normalized X^2/m was sometimes close to the expected value of unity, however also often as large as 29. The comparisons of the time series for the variable X^2/m with the time series of river discharge (panel b) shows that DA system was close to optimal during periods of strong to medium river discharges and was not optimal during periods of weak river discharge. The comparison with the time series of model surrogate and EOF errors (panel c) does not show any strong correlations with the temporal changes in the accuracy of the model surrogate or EOF, with a possible exception of slightly higher EOF errors during periods of elevated levels of X^2/m variable.

To summarize, the results of our optimality tests presented in Table 3 and Figure 14 show that our DA system was often close to optimal. However more work is needed to: (i) account for systematic biases in the non-assimilated model, (ii) better extract information from salinity and temperature measurements on the timescale of a single tidal period, and (iii) improve the optimality of the filter during periods of low river discharge.

Figures:

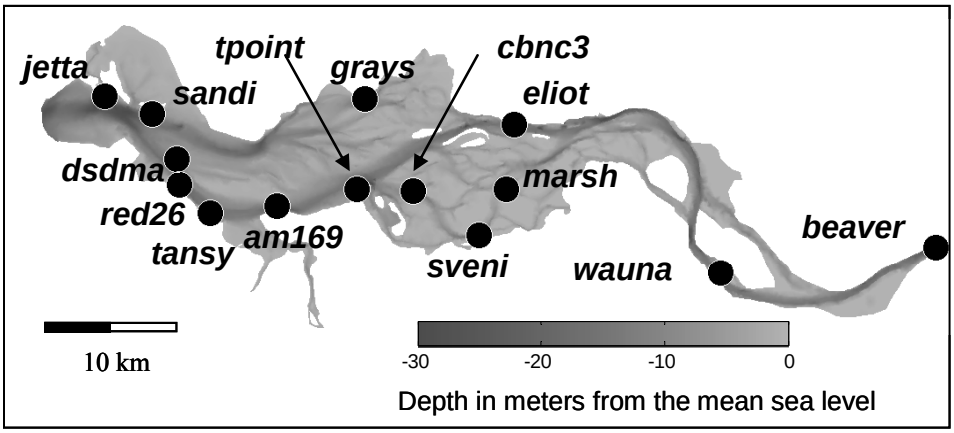


Figure 1: Map of the bathymetry in the computational domain. Overlaid are the names and locations of the observation stations.

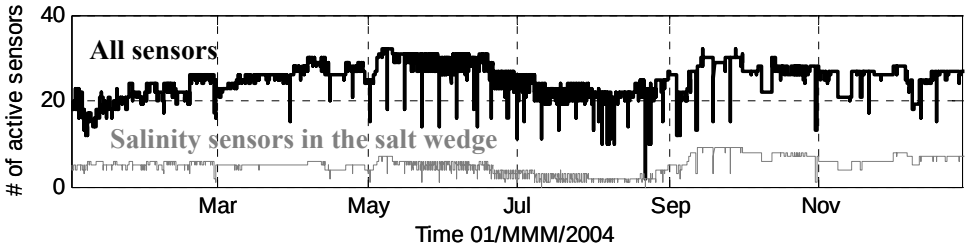


Figure 2: Number of active sensors available for assimilation in 2004.
Bold black line shows all available sensors. Light gray line shows sensors influenced by the estuarine salt wedge (from *jetta* to *am169*, excluding the top-sensor at *am169*).

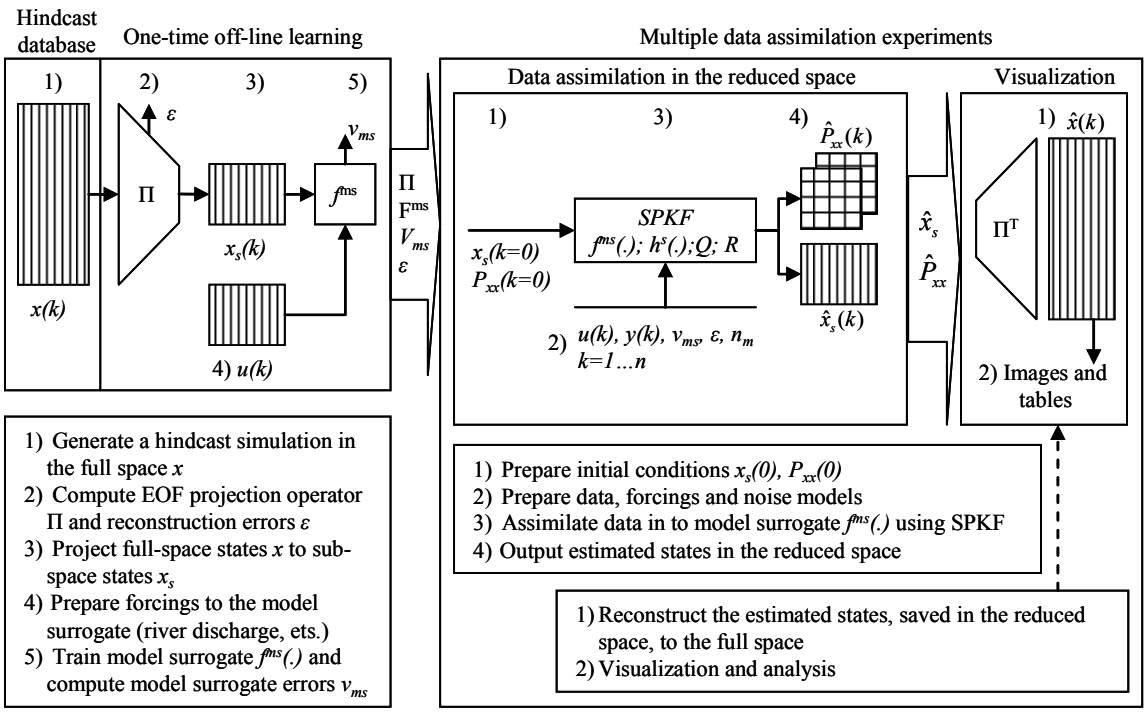


Figure 3: Information flow in the data assimilation method.

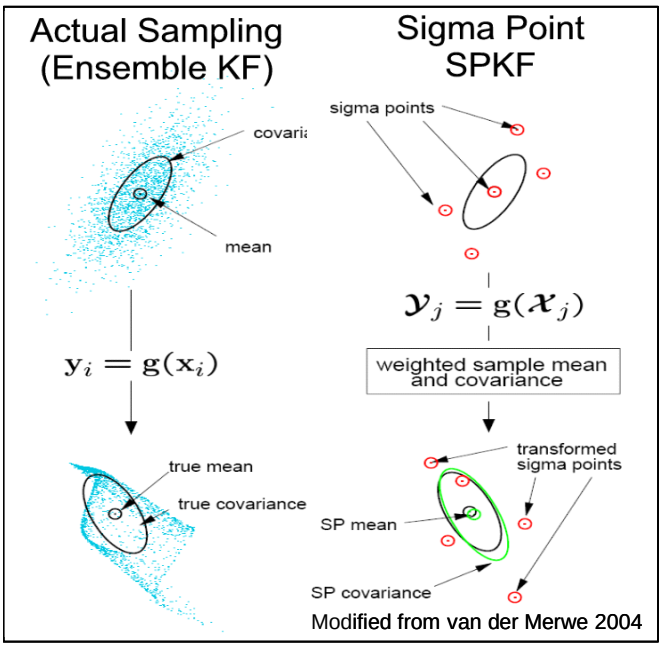


Figure 4: Illustration of the sigma point transformation for an arbitrary nonlinear function y=g(x) and a Gaussian random variable x.
(Image adopted from (van der Merwe 2004) with the permission of the author).

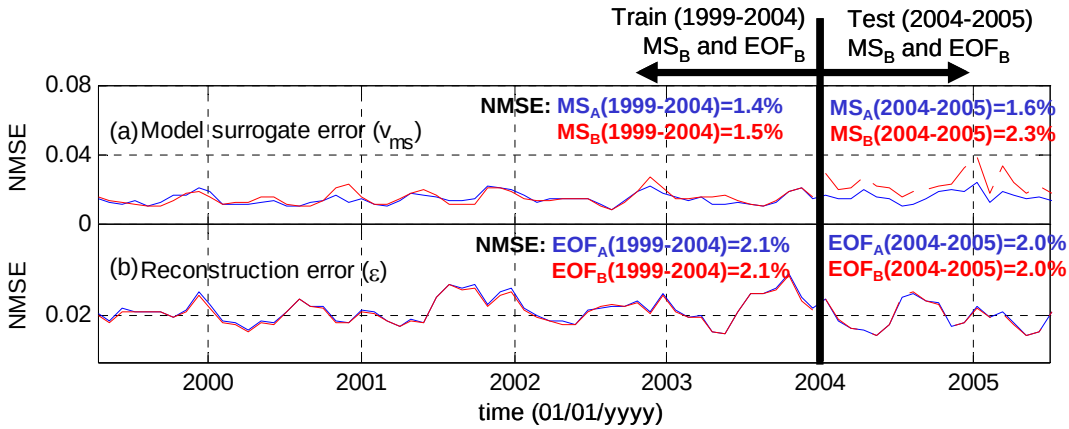


Figure 5: Time series of monthly averaged, normalized mean square error (NMSE) for model surrogate and EOF errors.
The NMSE was computed as a feedback prediction error for the model surrogates MS_A and MS_B and as a reconstruction error for the EOF_A and EOF_B. Training and testing periods for MS_B and EOF_B are delineated by the black vertical line.

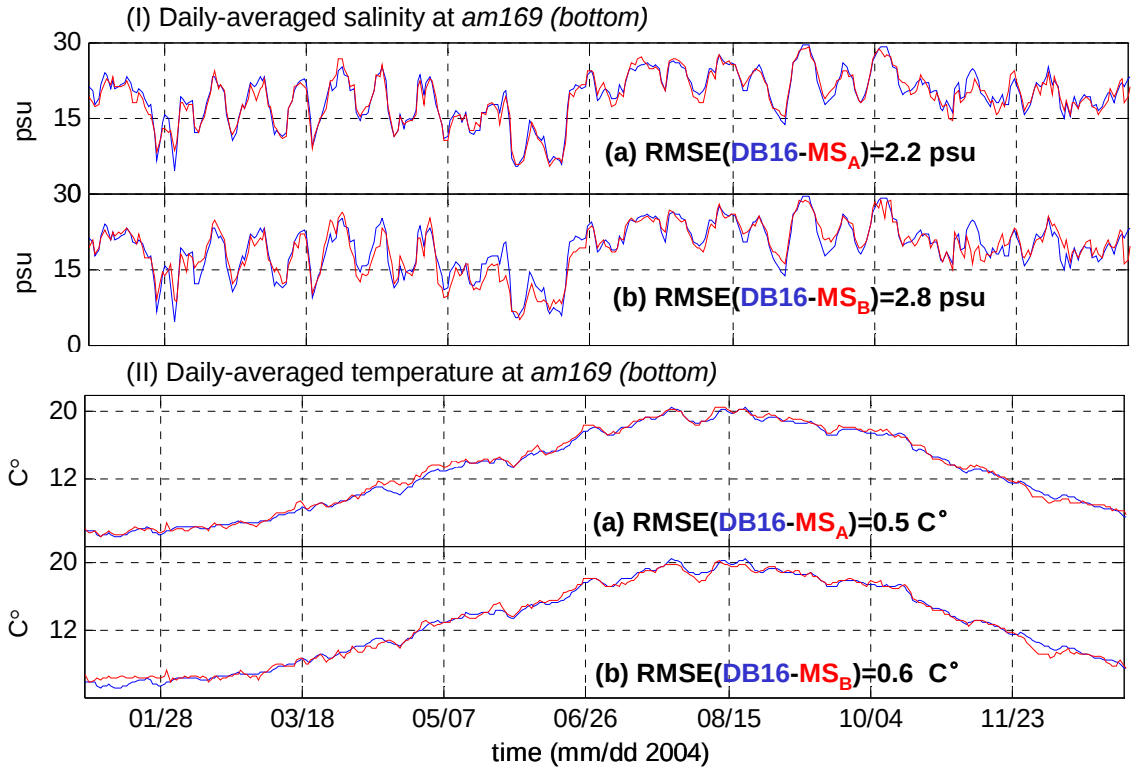


Figure 6: Daily averages for salinity (I) and temperature (II) at am169 (bottom sensor).

Panel (a) is for the model surrogate MS_A, and panel (b) is for the model surrogate MS_B. The root mean square errors (RMSE) are between DB16 (blue line) and the model surrogates (red line).

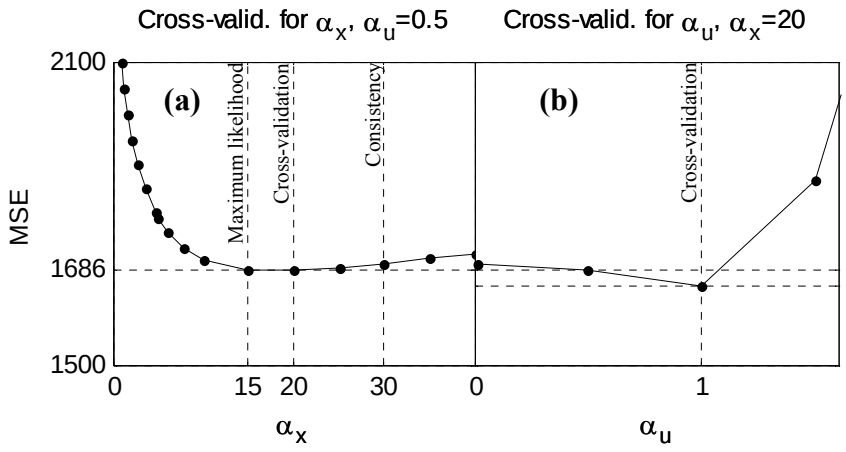
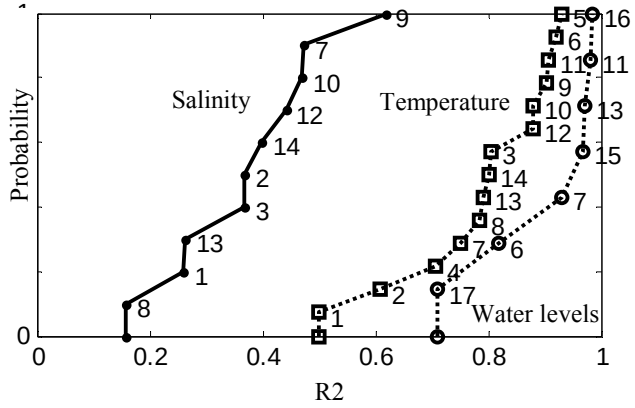


Figure 7: Cross-validation curves for αx (panel a) and αu (panel b).



1)	jetta		10)	am169 (bot)
2)	sandi		11)	marsh
3)	dsdma		12)	cbnc3
4)	red26 (top)		13)	grays
5)	red26 (mid)		14)	eliot
6)	red26 (bot)		15)	tpoin
7)	tansy		16)	sveni
8)	am169 (top)		17)	wauo3
9)	am169 (mid)			

Figure 8: Empirical cumulative distribution function (ECDF) for the percentage of the error variance at a validation stations (list on the right) explained by the assimilation stations.

X-axis: R^2 – the percentage of the error variance at the validation station explained by the assimilation stations.
Y-axis: probability that the explained error variance at a validation station is lower than *x*.

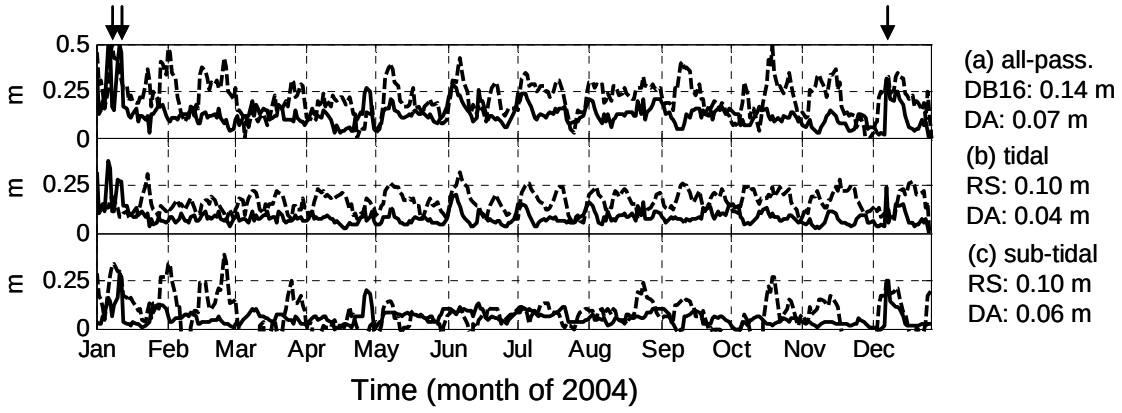


Figure 9: Maximum daily error in simulated water levels for DB16 (dashed) and DA experiment *DA-wl-tpoint* (solid).
All-pass errors (a), errors in tidal band (b), and errors in sub-tidal band (c). Separation into tidal and sub-tidal bands was with a butterworth filter (T=30h). Values of the root mean square errors are listed to the left of panels. Vertical arrows indicate periods when most water level sensors in the estuary failed.

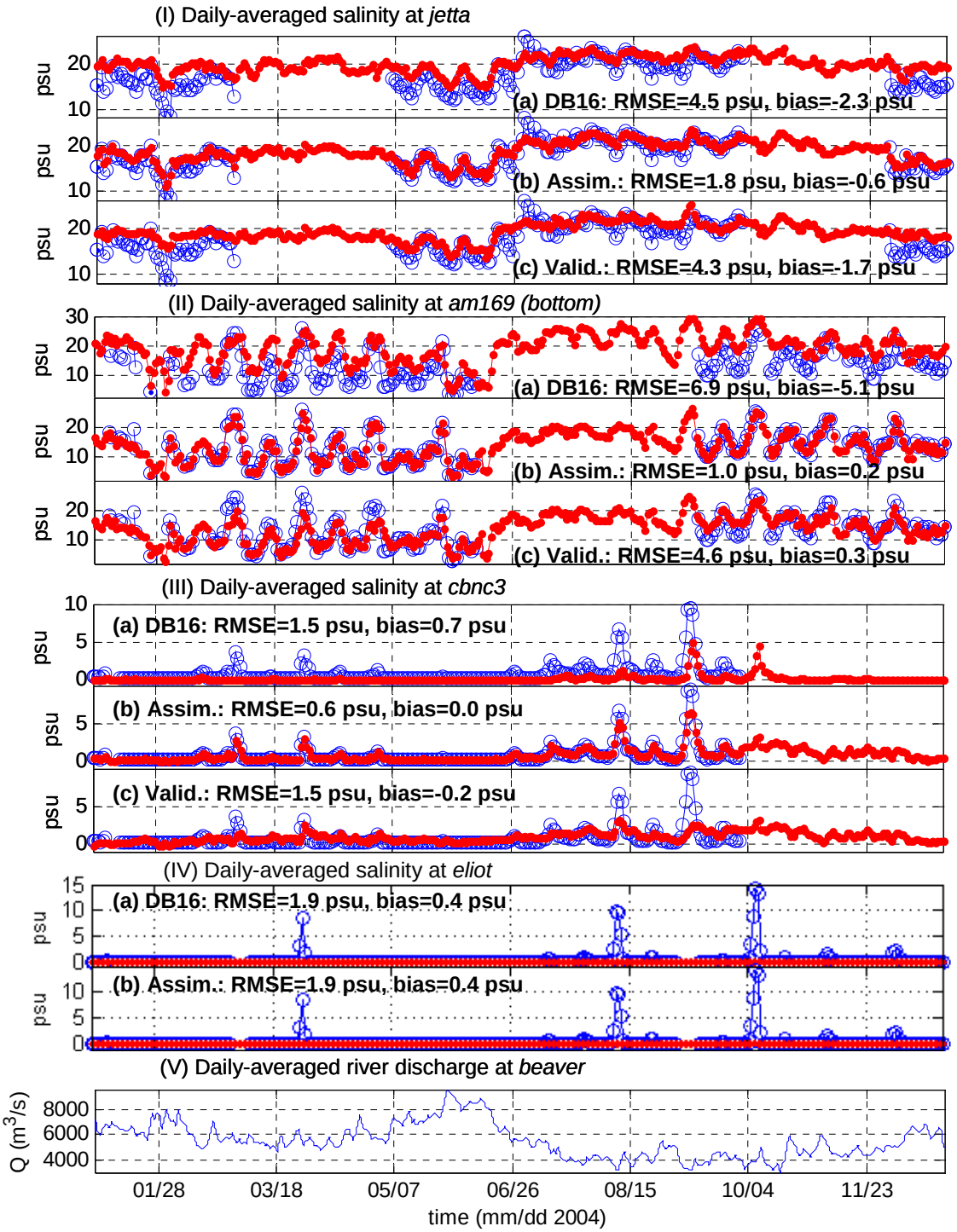


Figure 10: Daily-averaged salinities at stations jetta (I), am169 (II), cbnc3 (III), and eliot (IV). Observed, daily-averaged river discharge (V).
Observational data are in blue circles, and model data are in red dots (panels I, II, and III). For each station, except for *eliot*, three results are presented: (a) no data were assimilated (DB16), (b) data from this station were used for assimilation (*DA-all*), and (c) data from this station were withheld for validation. Values of a root mean square error (RMSE) and a model bias are listed for each panel.

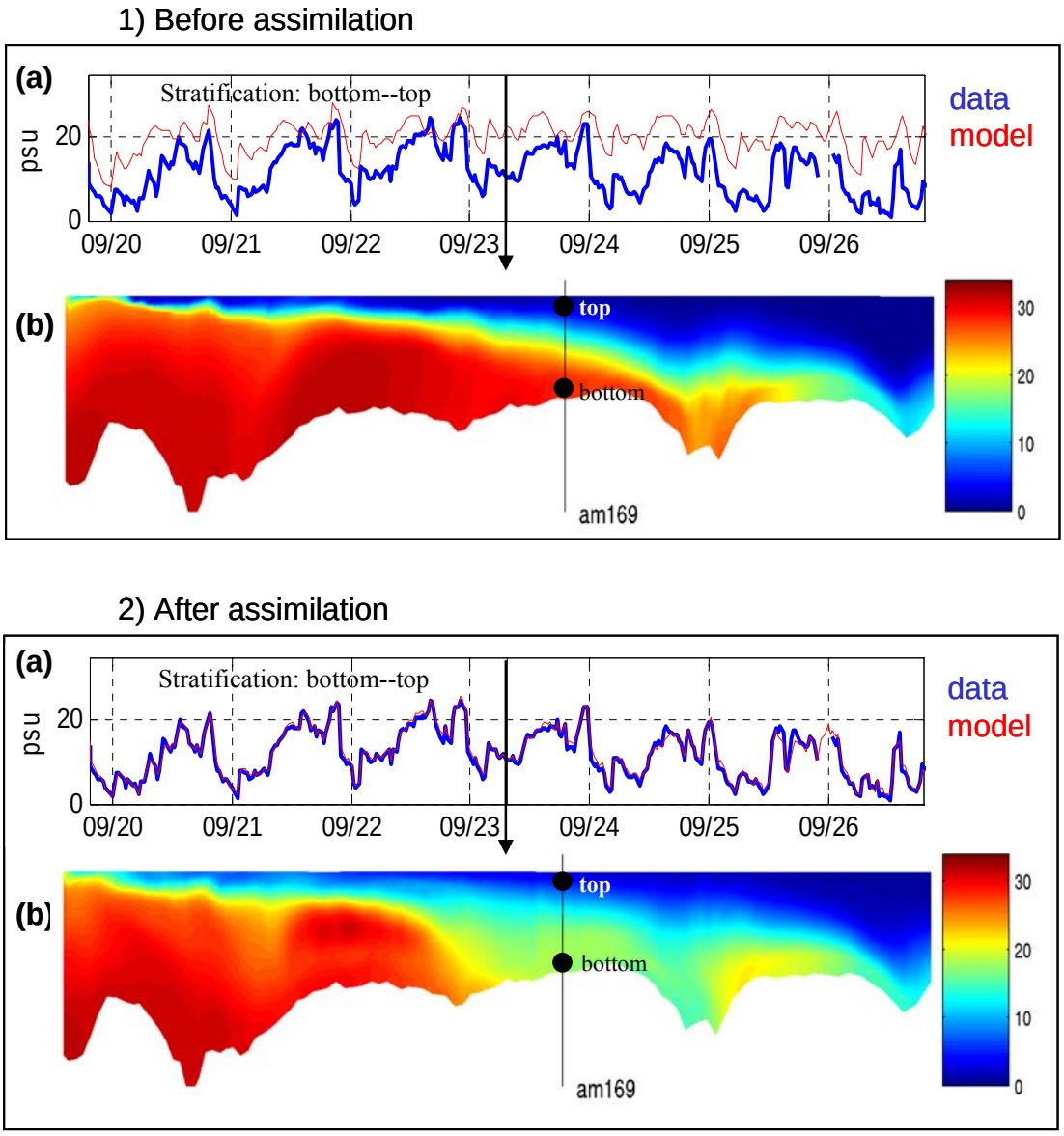


Figure 11: Salinity stratification at am169 (shown on sub-panels [a]), compared to the transect of salinity along the south channel (shown on sub-panels [b]).

In sub-panel [a] observations are in blue; and model data are in red. Stratification is computed as a difference of salinities between the bottom and top sensors. Transect is taken at September 23, 2004 at 07:15 (solid vertical line on panel [a]). Panel (1) is before data assimilation, and panel (2) is after data assimilation.

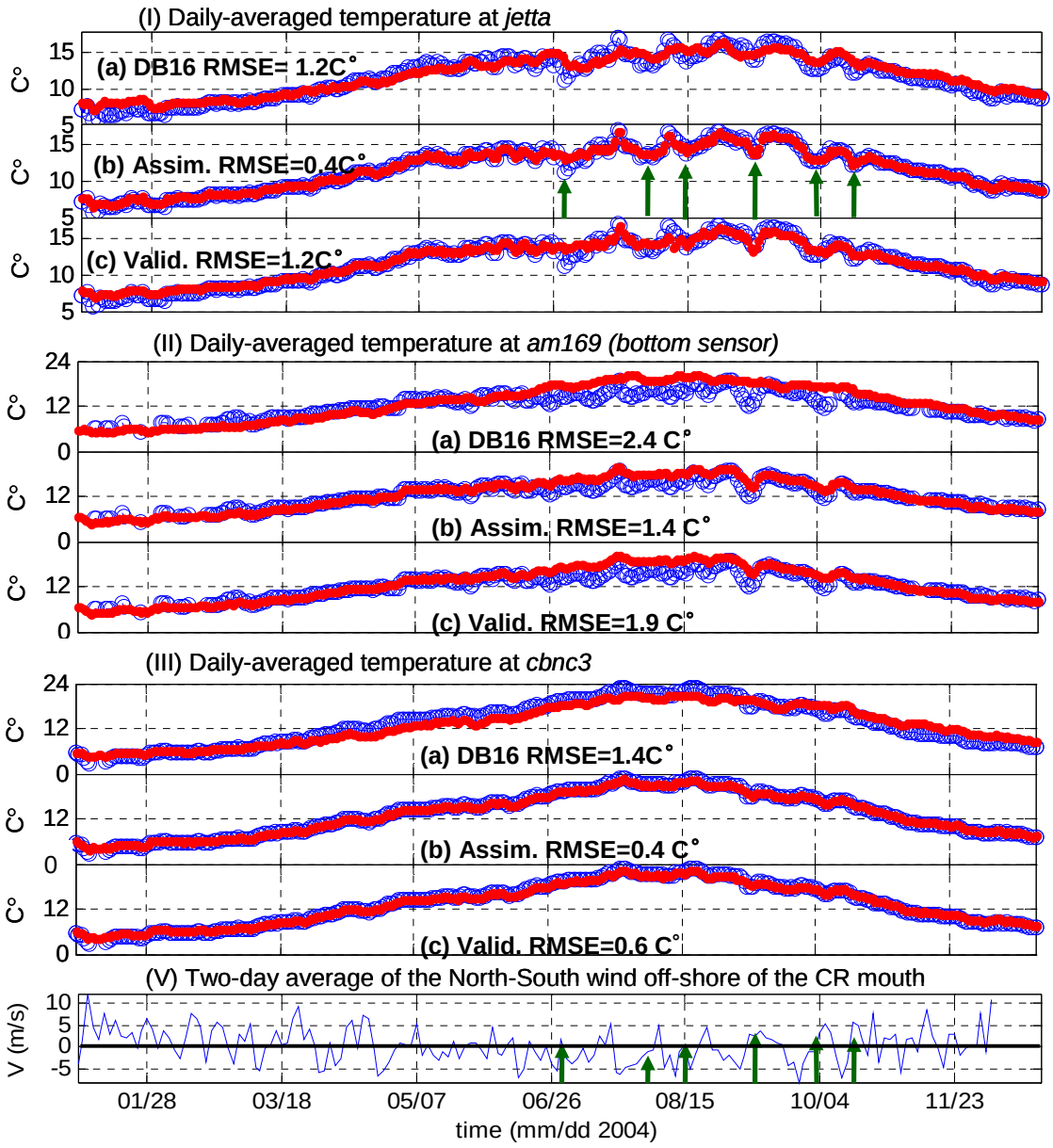


Figure 12: Daily-averaged temperatures at stations jetta (I), am169 (II), and cbnc3 (III). The two-day average of the North-South wind measured off-shore of the CR mouth (NOAA buoy 46029), panel IV.
Observational data are in blue circles, and model data are in red dots (panels I, II, and III). For each station three results are presented: (a) no data were assimilated (DB16), (b) data from this station were used for assimilation (DA-all), and (c) data from this station were withheld for validation. Values of a root mean square error (RMSE) and a model bias are listed for each panel. Green arrows highlight upwelling episodes.

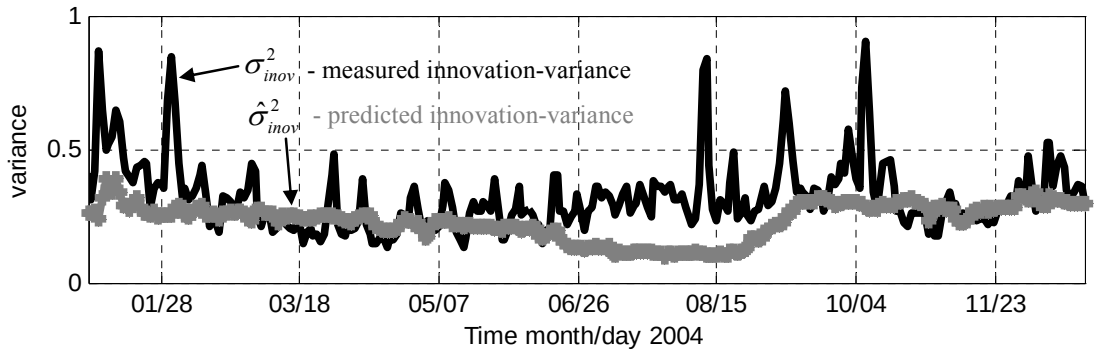


Figure 13: Measured (black line) and expected (gray line) variance of the innovation signal for DA experiment DA-all.
The low pass (T>30h) filter was used to improve the visual representation of the time series.

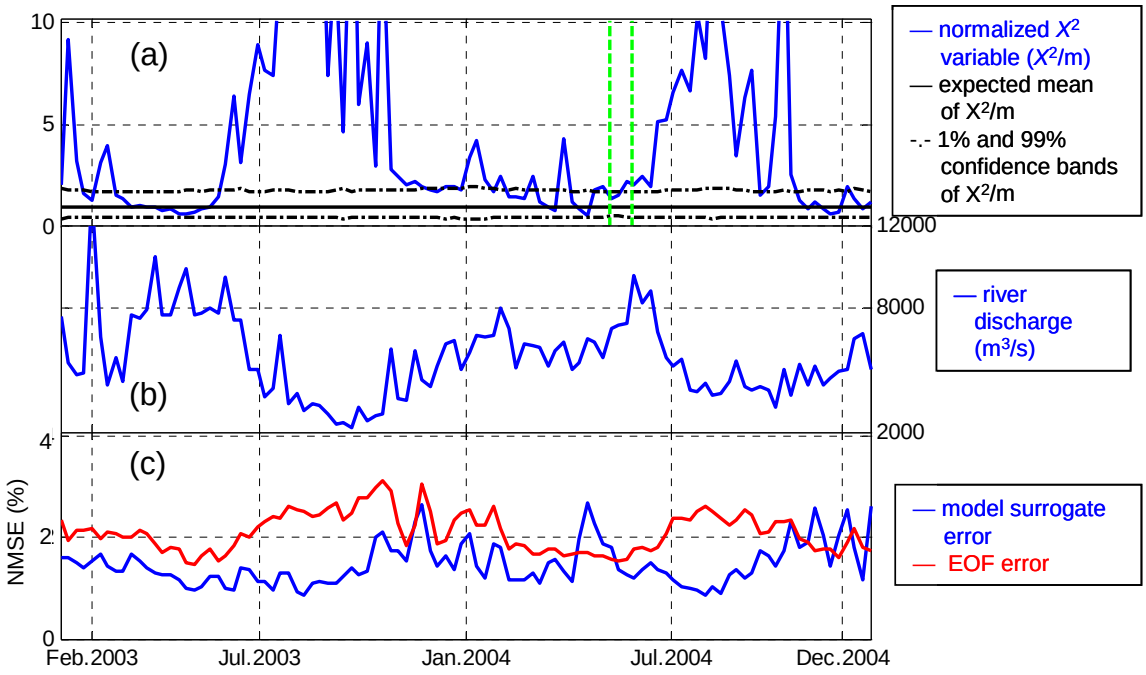


Figure 14: Statistics of the innovation signal.
(a) Normalized X^2 variable compared to its expected mean and variance. Normalization was with the number of observations m. (b) Time series of the river discharge. (c) Time series of the model surrogate (blue) and EOF (red) errors. Errors in (c) were normalized by the variance of the state. Vertical green lines in panel (a) indicate the time period during which the DA system was calibrated. To facilitate visual comparisons, all time series in were filtered, with fluctuations shorter than one week removed. Confidence bands estimated using Matlab's© chi2inv(.) function.

Tables:

Table 1: Root mean square errors (RMSE) for the cross-validation experiments.

	Observ. noise		RMSE				
Variable	n_m	n_o	DB16	Hindcast		Nowcast	
				Assim.	Valid.	Assim.	Valid.
	A	B	C	D	E	F	G
W. levels (m)	0.01	0.04	0.14	0.06	0.10	0.06	0.10
Salinity (psu)	0.5	1.5	4.3	1.2	3.0	1.2	3.0
Temp. (C)	0.1	0.3	1.2	0.5	0.7	0.5	0.7
Velocity Magnitude (m/s)	n/a	n/a	0.274	n/a	0.268	n/a	0.270

Table 2: Errors in cross-validation experiments as a percentage of the data variance at stations.

Variable	DB16	Hindcast		Nowcast	
		Assim.	Valid.	Assim.	Valid.
	C	D	E	F	G
W. levels (%)	2.5	0.5	1.2	0.5	1.2
Salinity (%)	13.0	1.1	6.1	1.0	6.2
Temp. (%)	0.7	0.1	0.3	0.1	0.3
Velocity Magnitude (%)	30.1	n/a	28.8	n/a	29.3

Table 3: Optimality measures for model innovations.

	Bias		Temporal autocorrelation	
Variable	$E\left[\delta y^{non-assim.}\right]$	$E\left[\delta y^{-}\right]$	$T_D\left[\delta y^{non-assim.}\right]$	$T_D\left[\delta y^{-}\right]$
A	B	C	D	E
Water levels	0.041 m	0.039 m	31 hours	4 hours
Salinity (ocean stations)	-3.6 psu	-0.4 psu	5.5 hours	5.5 hours
Salinity (river stations)	0.74 psu	0.2 psu	7 days	5.5 days
Temperature (ocean stations)	0.3 °C	0.1 °C	12 weeks	1 week
Temperature (river stations)	-0.4 °C	0.0 °C	12 weeks	12 weeks

Note: Ocean stations include *jetta, sandi, dsdma, red26, tansy*. River stations include *cbnc3, eliot, grays, marsh*.

References:

Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., and Sorensen, D. (1999), *Lapack Users' Guide, Third Edition*, Philadelphia: SIAM.

Bai, Z., Demmel, J., Dongarra, J., Ruhe, A., and van der Vorst, H. (eds.) (2000), *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, Philadelphia: SIAM.

Baptista, A. M. (2006), "CORIE: The First Decade of a Coastal Margin Collaborative Observatory," in *Oceans'06 MTS/IEEE*, Boston.

Baptista, A. M., Zhang, Y.-L., Chawla, A., Zulauf, M. A., Seaton, C., Myers, E. P., Kindle, J., Wilkin, M., Burla, M., and Turner, P. J. (2005), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: Ii. Application to the Columbia River," *Continental Shelf Research*, 25, 935-972.

Bar-Shalom, Y., and Fortmann, T. E. (1988), *Tracking and Data Association*, New York NY, USA: Academic Press.

Bennett, A. F. (1992), *Inverse Methods in Physical Oceanography*, Cambridge University Press.

Bertino, L., Evensen, G., and Wackernagel, H. (2002), "Combining Geostatistics and Kalman Filtering for Data Assimilation in an Estuarine System," *Inverse Problems*, 18, 1-23.

Bishop, C. (1995), *Neural Networks for Pattern Recognition*, Oxford University Press.

Bottom, D. L., Simenstad, C. A., Burke, J., Baptista, A. M., Jay, D. A., Jones, K. K., Casillas, E., and Schiewe, M. H. (2005), "Salmon at River's End. The Role of the Estuary in the Decline and Recovery of Columbia River Salmon," NOAA Technical Memorandum NMFS-NWFSC-68, U.S. Dept. Commer.

Brand, M. (2002), "Incremental Singular Value Decomposition of Uncertain Data with Missing Values," *COMPUTER VISON - ECCV 2002, PT 1*, 2350, 707-720.

Bretherton, F. P., Davis, R. E., and Fandry, C. B. (1976), "A Technique for Objective Analysis and Design of Oceanographic Experiments Applied to Mode-73," *Deep-Sea Research*, 23, 559-582.

Burla, M., Baptista, A. M., Casillas, E., and Williams, J. G. (submitted), "The Influence of the Columbia River Plume on the Survival of Steelhead (Oncorhynchus Mykiss) and Chinook Salmon (O. Tshawytscha): A Numerical Exploration.," *Can. J. Fish. Aquat. Sci.*

Burla, M., Baptista, A. M., Zhang, Y.-L., Casillas, E., Bottom, D. L., and Simenstad, S. A. (2007), "Salmon Habitat Opportunity in the Columbia River Estuary: Modeling the Physical Environment to Inform Management Decisions," in *Coastal Zone '07*, Portland, OR.

Cane, M. A., Kaplan, A., Miller, R. N., Tang, B., Hackert, E., and Busalacchi, A. J. (1996), "Mapping Tropical Pacific Sea Level: Data Assimilation Via a Reduced State Space Kalman Filter," *Journal of Geophysical Research*, 101 (C10), 22599-22617.

Carrera, J., and Neumann, S. P. (1986), "Estimation of Aquifer Parameters under Transient and Steady State Conditions: Maximum Likelihood Method Incorporating Prior Information.," *Water Resources Research*, 22 (2), 199-210.

Dee, D. P. (1995), "On-Line Estimation of Error Covariance Parameters for Atmospheric Data Assimilation," *Monthly Weather Review*, 123, 1128-1145.

Evensen, G. (2003), "The Ensemble Kalman Filter: Theoretical Formulation and Practical Implementation," *Ocean Dynamics*, 53 (4), 343 - 367.

Fedorov, V. V. (1972), *Theory of Optimal Experiments.*, Academic Press.

Fedorov, V. V. (1994), "Optimal Experimental Design: Spatial Sampling," *Calcutta Statistical Association Bulletin*, 44 (173-174).

Gandin, L. (1963), *Objective Analysis of Meteorological Fields*, Leningrad: Hydrometeoizdat.

Golub, G., and van Loan, C. (1996), *Matrix Computations*, Baltimore: The John Hopkins University Press.

Hamilton, P. (1990), "Modeling Salinity and Circulation for the Columbia River Estuary," *Progress in Oceanography*, 25, 113-156.

Heemink, A. W., Mouthaan, E. E. A., Roest, M. R. T., Vollebregt, A. E. H., Robaczewska, K. B., and Verlaan, M. (2002), "Inverse 3d Shallow Water Flow Modeling of the Continental Shelf," *Continental Shelf Research*, 22, 465-484.

Holmes, P., Lumley, J., and Berkooz, G. (1996), *Turbulence, Coherent Structures, Dynamical Systems and Symmetry (Cambridge Monographs on Mechanics)*, Cambridge: Cambridge University Press.

Hoteit, I., and Pham, D. T. (2003), "Evolution of the Reduced State Space and Data Assimilation Schemes Based on the Kalman Filter," *Journal of Meteorological Society Japan*, 81, 21-39.

Jay, D., and Smith, J. D. (1990), "Residual Circulation in Shallow Estuaries, Ii. Weakly Stratified and Partially Mixed, Narrow Estuaries," *Journal of Geophysical Research*, 95 (C1), 733-748.

Jay, D. A., and Flinchem, E. P. (1997), "Interaction of Fluctuating River Flow with a Barotropic Tide: A Demonstration of Wavelet Tidal Analysis Methods," *Journal of Geophysical Research*, 102 (C3), 5705-5720.

Jolliffe, I. T. (1986), *Principal Component Analysis*, Springer-Verlag.

Kalman, R. E. (1960), "A New Approach to Linear Filtering and Prediction Problems.," *ASME Journal of Basic Engineering*, 82, 35-45.

Kurapov, A. L., Allen, J. S., Egbert, G. D., Miller, R. N., Kosro, P. M., Levine, M., Boyd, T., Barth, J. A., and Moum, J. (2005), "Assimilation of Moored Velocity Data in a Model

of Coastal Wind-Driven Circulation off Oregon: Multivariate Capabilities," *Journal of Geophysical Research*, 110 C10S08.

Le Dimet, F.-X., and Talagrand, O. (1986), "Variational Algorithms for Analysis and Assimilation of Meteorological Observations: Theoretical Aspects," *Tellus A*, 38, 97-110.

Lermusiaux, P. F. J., and Robinson, A. R. (1999), "Data Assimilation Via Error Subspace Statistical Estimation, Part I: Theory and Schemes," *Monthly Weather Review*, 127 (8), 1385-1407.

Lu, Z., Leen, T. K., van der Merwe, R., Frolov, S., and Baptista, A. M. (2007), "Sequential Data Assimilation with Sigma-Point Kalman Filter on Low-Dimensional Manifold," *submited to Journal of Inverse Problems*, also available at http://www.stccmop.org/files/CMOP-TR-07-001.pdf [Viewed on September 14, 2007].

Martin, P. J. (2000), *Description of the Navy Coastal Ocean Model Version 1.0.*, Naval Research Laboratory, Technical Report.

Møller, M. (1996), "A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning.," *Neural Networks*, 6 (4), 525–533.

Nabney, I. T. (2004), *Netlab: Algorithms for Pattern Recognition*, Springer.

Oke, P. R., Allen, J. S., Miller, R. N., Egbert, G. D., and Kosro, P. M. (2002), "Assimilation of Surface Velocity Data into a Primitive Equation Coastal Ocean Model," *Journal of Geophysical Research*, 107 (C9), 3122.

Pham, D. T. (2001), "Stochastic Methods for Sequential Data Assimilation in Strongly Nonlinear Systems," *Monthly Weather Review*, 129 (5), 1194-1207.

Pham, D. T., Verron, J., and Roubaud, M. C. (1998), "A Singular Evolutive Extended Kalman Filter for Data Assimilation in Oceanography," *Journal of Marine Systems*, 16, 323-340.

Sorensen, D. (1992), "Implicit Application of Polynomial Filters in a K-Step Arnoldi Method," *Siam Journal on Matrix Analysis and Applications*, 13 (1), 357-385.

USACE. (2001), "Columbia River Channel Improvement Project: 2001 Biological Assessment," U. S. Army Corps of Engineers.

van der Merwe, R. (2002-2006), "Rebel: Recursive Bayesian Estimation Library," http://choosh.ece.ogi.edu/rebel/ [Viewed on September 14, 2007].

van der Merwe, R. (2004), "*Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*," Ph.D., OHSU.

van der Merwe, R., Leen, T. K., Lu, Z., Frolov, S., and Baptista, A. M. (2007), "Fast Neural Network Surrogates for Very High Dimensional Physics-Based Models in Computational Oceanography," *Neural Networks*, 20, 462-478.

van der Merwe, R., and Wan, E. (2001), "Efficient Derivative-Free Kalman Filters for Online Learning," in *Proceedings of the European Symposium on Artificial Neural Networks (ESANN)*, (Bruges, Belgium).

van der Merwe, R., and Wan, E. (2003), "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models," in *Workshop on Advances in Machine Learning*, Montreal, Canada.

Wilkin, J. L., Arango, H. G., Haidvogel, D. B., Lichtenwalner, C. S., Glenn, S. M., and Hedström, K. S. (2005), "A Regional Ocean Modeling System for the Long-Term Ecosystem Observatory," *Journal of Geophysical Research*, 110 (C06S91).

Xu, J., Shenn-Yu, C., Raleigh, H. R., Wang, H. V., and Boicourt, W. C. (2002), "Assimilating High-Resolution Salinity Data into a Model of a Partially Mixed Estuary," *Journal of Geophysical Research*, 107 (C7), 11-11.

Xue, Y., Cane, M. A., and Zebiak, S. E. (1997), "Predictability of a Coupled Model of Enso Using Singular Vector Analysis. Part I: Optimal Growth in Seasonal Background and Enso Cycles," *Monthly Weather Review*, 125, 2043-2056.

Zhang, Y.-L., and Baptista, A. M. (submitted), "A Semi-Implicit Eulerian-Lagrangian Finite-Element Model for Cross-Scale Ocean Circulation," *Ocean Modeling*.

Zhang, Y.-L., Baptista, A. M., and Myers, E. P. (2004), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: I. Formulation and Skill Assessment," *Continental Shelf Research*, 24, 2187-2214.

Chapter 3

Estimation of Ecologically Significant Circulation Features of the Columbia River Estuary and Plume Using a Reduced-Dimension Kalman Filter

Authors: Frolov, S., A. M. Baptista, Y. Zhang, C. Seaton

To be submitted to Continental Shelf Research

Abstract:

A data assimilation method was used to estimate the variability of three ecologically significant features of the Columbia River (CR) estuary and plume: the size of the CR plume, the orientation of the CR plume, and the length of the salinity intrusion in the CR estuary. Our data assimilation method was based on a reduced-dimension Kalman filter that enables fast data assimilation of nonlinear dynamics in the CR estuary and plume. Assimilated data included measurements of salinity, temperature, and water levels at 13 stations in the CR estuary and at 5 moorings in the CR plume.

The analysis of our experimental results showed that data assimilation played a significant role in controlling the magnitude and timing of dynamic events in the CR estuary and plume, such as events of extreme salinity intrusion and events of regime transitions in the CR plume. Data assimilation also changed the response of the salinity intrusion length to variations in the CR discharge, hence imposing a new dynamic on the CR estuary. The validation of the assimilated solution with independent data showed that these corrections were likely to be realistic, since the assimilated model was closer to the true ocean than the original non-assimilated model.

3.1 Introduction

The dynamics of the Columbia River (CR) estuary and plume exert significant control over the ecological productivity of the CR ecosystem, specifically over the survival of juvenile salmon (Bottom, et al. 2005, Burla, et al. submitted). Since many features of these dynamics, *e.g.*, the volume of the CR plume, are hard to measure directly in nature, numerical models are used (Bottom, et al. 2005, Burla, et al. submitted) to characterize the variability of these dynamical features. However, such numerical models are limited in their ability to realistically simulate nature due to uncertainties in model forcings (*e.g.*, offshore winds) and due to approximations in model physics (*e.g.*, parameterization of vertical mixing in the estuary). In Chapter 2 of this thesis, we showed that errors due to these uncertainties and approximations can be reduced through data assimilation—a modeling technique that uses observational data to constrain the solution of a numerical. In this article, we describe how data assimilation affects the simulated dynamics of the ecologically significant features in the CR estuary and plume, specifically the length of the salinity intrusion in the CR estuary, and the orientation and the size of the CR plume.

To assimilate observational data, we used recently developed reduced-dimension Kalman filter (Lu, et al. 2007)—a nonlinear extension to the classical Kalman filter that employs fast neural network model surrogates (van der Merwe, et al. 2007) to expedite computation of the Kalman filter equations. The assimilated data included the *in situ* measurements of salinity and temperature from 13 stations in the CR estuary and from five moorings in the CR plume. Our validation experiments, against independent data from fixed stations and research cruises, showed that data assimilation was more accurate than a non-assimilated model in simulating distribution of salinity in the CR estuary and plume. However, at the time of experiments, we did not have sufficient observational data to reliably quantify the improvements in simulation of such hard-to-measure quantities as salinity intrusion length and size of the CR plume, which limited our analysis to qualitative comparisons.

The analysis of our experimental results, presented in section 3.4, shows that data assimilation played a significant role in controlling the simulated magnitude and timing of dynamic events in the CR estuary and plume, such as events of extreme salinity intrusion and events of regime transitions in the CR plume. Data assimilation was also able, with limited success, to correct for some persistent biases in the numerical model, such as the underprediction of the salinity intrusion in the CR estuary. However, the ability of data assimilation to correct for the underprediction of salinity intrusion was limited by the observational data available for the assimilation and by the accuracy of the non-assimilated model.

3.2 Oceanographic setting

The freshwater output of the CR, the second largest in the U.S., sustains two prominent oceanographic and ecological features in the region: the CR estuary and the CR plume. Both the CR estuary and the CR plume are highly responsive to variations in river, ocean, and atmospheric forcings, and are tightly coupled by the transport of mass and momentum through the mouth of the CR estuary. In the following paragraphs, we summarize the dynamics of, the modeling successes and challenges in, and the data assimilation goals for the CR estuary and plume.

The CR plume: The large buoyant plume of the CR is controlled by a multitude of forces. The volume and the size of the CR plume are controlled by the river discharge and mixing, while the orientation of the CR plume is controlled by ambient ocean currents and local winds. Seasonal changes in the ambient ocean currents and the local winds tend to orient the CR plume towards north in fall and winter, and towards southwest in spring and summer (Hickey, et al. 1998). However, the instantaneous orientation of the CR plume can respond quickly (hours to days) to changes in local winds (Garcia-Berdeal, et al. 2002, Baptista, et al. 2005, Hickey, et al. 2005). In fact, it is not uncommon that the CR plume has a bi-directional structure during the periods of highly variable winds (Hickey, et al. 2005).

In this article, we study the effect of data assimilation on two ecologically significant features of the CR plume: the plume size (as measured by the plume's

volume, area, and thickness) and the plume orientation (as manifested by the salinity levels at a distributed array of moorings and by the position of the plume's centroid). Burla (Burla, et al. 2007, Burla, et al. submitted) showed that the size and location of the CR plume is a good predictor for the survival of juvenile salmon, as they migrate from the CR estuary to the ocean. Verification with cruise data, remote sensing data, and data from fixed moorings suggest that the non-assimilated model is mostly correct in predicting the orientation and the horizontal extent of the CR plume (Baptista, et al. 2005). However, as shown in section 3.4.1, it was not uncommon for the non-assimilated model to mischaracterize the transitions of the CR plume from one direction to another, possibly due to errors in wind forcings.

CR estuary: The CR estuary, a classical river-dominated estuary, responds quickly to changes in ocean tides, regulated river discharge, and coastal winds. The tidal dynamics in the CR is significant (tidal amplitudes of ~3.6 meters) and is driven by the non-linear interactions of astronomic tides with complicated bathymetry, non-stationary river discharge, and coastal winds (Jay, et al. 1997). Compressed and often stratified, the estuarine circulation in the CR is subject to extreme variations in salinity intrusion and stratification regimes (Hamilton 1990, Jay, et al. 1990, Baptista, et al. 2005).

An ecologically significant feature of the CR estuary circulation, studied in this article, is the salinity intrusion length in the CR estuary, which is controlled by the river discharge, tidal mixing, and the density of the ocean water supplied to the CR estuary from the continental shelf. The stronger river discharge tends to displace the position of the salt wedge further downstream; while the enhanced tidal mixing during spring tides counteracts the baroclinic density differences that force salinity upstream. Jay (Jay, et al. 1990) reported that the daily maximum of the salinity intrusion varies in the modern CR estuary from a low of ~28 km, during high river flow conditions, to a high of ~52 km, during low river flow conditions. The report by Bottom (Bottom, et al. 2005) identified salinity intrusion as a possible factor determining the habitat opportunity for juvenile salmon, which requires further investigation using modeling studies. At the time of our experiments, the non-assimilated model of the CR estuary underpredicted the length of

the salinity intrusion in the CR estuary. For more discussion on model-data comparisons in the CR estuary; see section 3.4.1.

3.3 Data assimilation methods

3.3.1 Overview of the DA method

To assimilate observational data into a model of the CR estuary and plume, we used the reduced-dimension Kalman filter (Lu, et al. 2007)—a recently developed data assimilation method, which is fast, model-independent, adjoint-free, and nonlinear. The computational efficiency of the new method comes, in part, from the use of a neural network model surrogate (van der Merwe, et al. 2007) that executes forward simulations three to four orders of magnitude faster than a traditional numerical circulation code. The new method was successfully applied for data assimilation in the CR estuary (Chapter 2 of this thesis) and, in this research, was extended to the coupled dynamics in the CR estuary and plume.

Our implementation of the data assimilation algorithm closely follows the description presented in Chapter 2 of this thesis, with the differences in the implementation described in section 3.3.4. Following is the short overview of the framework for reduced-dimension Kalman filter that was used in our experiments, which was described in detail in Chapter 2 of this thesis and in (Lu, et al. 2007):

1) A long, statistically representative hindcast simulation of the CR system was generated using the forward model SELFE (Zhang, et al. submitted). Two such hindcast databases were used: database 14 (DB14) for the CR estuary and plume, and database 16 (DB16) for the CR estuary.
2) Samples drawn from the hindcast database in step (1) were used to train a model surrogate, using training algorithms developed in (van der Merwe, et al. 2007). Because it was not computationally feasible to train the model surrogate in the high-dimensional space of the forward model, the model surrogate was trained in the subspace spanned by the empirical orthogonal functions (EOFs) of the forward model.

3) The state of the model surrogate was estimated using a Kalman filter of choice. Because the estimated state of the model surrogate was small (50-100 degrees of freedom), we choose to use the sigma point Kalman filter– a state-of-the-art, full-rank Kalman filter (van der Merwe, et al. 2003, van der Merwe 2004).
4) The output of the Kalman filter was analyzed by reconstructing the estimated state and the estimated model error from the EOF subspace, where the model surrogate and the Kalman filter operated, to the full space of the model, where many analysis and visualization tools operated.

Since all data assimilation experiments in step (3) above were performed in the reduced EOF subspace, CPU, memory, and storage requirements for these data assimilation experiments were very low. For the CR estuary domain, data assimilation was ~100 times faster than the forward model SELFE. For example, using a single AMD Athlon©, it took SELFE ~100 days to generate a two-year non-assimilated simulation of the CR estuary, while it took only ~2 days for the reduced-dimension KF to assimilated two years of observational data. In practice, all these computations were distributed over multiple CPU processors, leading to shorter "wall-clock" times.

3.3.2 Hindcast simulations

To train the model surrogates and the EOF subspaces (step 2 in section 3.3.1), we used model states drawn from two hindcast simulations of the CR system (Baptista, et al. 2005, Baptista 2006). The first model surrogate was trained using hindcast database DB14, which simulated the coupled dynamics of the CR estuary and plume. The second model surrogate was trained using hindcast database DB16, which simulated the dynamics of the CR estuary, using a high-resolution model nested inside of the DB14 domain. The hindcast databases were generated using SELFE (Zhang, et al. submitted)—a finite element, Eulerian-Lagrangian model that solves a set of nonlinear, baroclinic, shallow-water equations. The prognostic variables included horizontal velocities, salinities, temperatures, and water levels. The model domains were discretized using unstructured horizontal grids (40.000 and 17.000 elements in DB14 and DB16 respectively), and sigma-z vertical grids (54 and 26 levels respectively).

To generate DB14 the following boundary conditions were used:

- **Ocean conditions**: Tides were imposed using eight tidal constituents (M_2, S_2, N_2, K_2, K_1, O_1, P_1, Q_1) derived from the data-assimilated tidal model (Foreman 1977, Myers, et al. 2001). The sea surface slope (Z0) was imposed based on the daily averaged sea-surface height from the navy costal ocean modal (NCOM) (Martin 2000). Away from the CR plume, ocean temperature and salinity were nudged to the NCOM prediction.
- **River inputs**: The temperature and the flow of the CR estuary were based on the measurements from the USGS station Beaver Army (*beaver*)[12].
- **Atmospheric boundary conditions:** The 2D fields of wind stress, atmospheric pressure, and surface heat flux were generated using a combination of forecast and re-analysis products, which were provided by the National Center for Environmental Prediction and Oregon State University.

Boundary conditions for DB16 were similar, except for the ocean boundary in the CR mouth, where the time series of salinity, temperature, and elevations, which were extracted from DB14, were imposed as the ocean boundary condition in DB16.

The spatial scales, resolved by both hindcast models, were limited by the resolution of the numerical grid and ranged from ~30 m in the main channel of the estuary, to ~600 m in tidal flats, to ~1 km in the plume, and to 24 km in the coastal ocean. The resolved periodic timescales included nonlinear tides (4-6 hours); astronomical tides (12-24 hours); spring-neap tidal cycle (~15 days); and seasonal changes in surface heat-fluxes, river temperature, and river discharge (~1 year). Nonperiodic timescales included the response of the CR plume and estuary to weather changes (~2-10 days) and inter-annual variations in ocean conditions (>1 year).

[12] USGS stands for U.S. Geological Survey. Data for station Beaver Army is available at: http://waterdata.usgs.gov/usa/nwis/uv?site_no=14246900.

3.3.3 Observational system

The locations of the observation stations used in our data assimilation experiments are presented in Figure 15. At each station an array of sensors measured a variable combination of parameters, including water level, salinity, temperature, velocity profile, wind temperature, and wind velocity. The assimilated data in the CR estuary included water level, salinity, and temperature data from 15 CTD sensors and 6 tidal gages located at 13 stations in Figure 15.3. The assimilated data in the CR plume included salinity and temperature data from 13 CTD sensors located at five ocean moorings: *risec*, *rises*, *risen*[13], *ogi01*[14], and *ogi02* (Figure 15.2). Most of the stations in Figure 15 were operated by CORIE[15], with the exception of tidal stations *tpoint* and *wauna,* which were operated by National Atmospheric and Oceanic Administration[16], and stations *risen*, *risec*, and *rises*, which were deployed as a part of the RISE project[17].

In addition to observations from the fixed sensors, which were used for assimilation, we used data collected by research cruises in the CR estuary and plume to validate our data assimilation experiments. Remote sensing measurements of the surface temperature, color, salinity, and velocities were not available or had poor temporal coverage during the period of our data assimilation experiments and hence were not used for assimilation or validation.

3.3.4 Implementation of the DA method

The implementation of the data assimilation method for estuary-only simulation, DB16, was described in detail in Chapter 2 of this thesis. The implementation for the estuary-plume simulation, DB14, was similar, notwithstanding differences in

---

[13] In DB14 the width of the northward-going jet of the CR plume did not extend as far west as the location of the station *risen*, where observational data for the variability of the northward jet was collected. Since assimilating data from *risen* at it's actual location produced unphysical shape of the CR plume, we opted for assimilating *risen* data at the location of *risen**—the artificially placed station within the jet of the CR plume, located 13 km east of the actual location of *risen*. The eastern displacement of stations *risen** was selected to minimize model-data misfits with DB14.

[14] Data from the mooring *ogi01* was assimilated only for a short period, since this mooring was lost during the 2004 season.

[15] Available at http://www.stccmop.org/corie/observation_network

[16] Available at http://www.co-ops.nos.noaa.gov/.

[17] See http://www.ocean.washington.edu/rise/

implementation that emerged from the higher complexity of the coupled dynamics in the CR estuary and plume.

To compute the EOF modes of the coupled estuary-plume model, we used the DC-EOF algorithm (Chapter 2 of this thesis), where in the first step of the DC-EOF algorithm the EOFs for the plume variables where computed separately from the estuarine variables. After the DC-EOF algorithm merged the preliminary EOFs for plume and estuary, we used the first 100 merged EOF for training of the model surrogate, capturing 96.5% of the state variance in the CR plume and 98.3% of the state variance in the CR estuary. Unfortunately, due to the larger size and higher complexity of the coupled CR estuary-plume system, we failed to train a model surrogate that was accurate outside of the training interval. Instead, we trained a model surrogate that was accurate (within 5.7%) during year 2004, in the summer of which the observational data in the CR plume were available. As was discussed in Chapter 2 of this thesis, it is still possible to use this model surrogate, trained during year 2004, for hindcast data assimilation during the same time interval. The parameters of the process-noise model in DB14 were selected using maximum-likelihood criterion of Dee (Dee 1995).

3.4 Results

We present the results of data assimilation experiments in two subsections. In section 3.4.1, we describe how data assimilation changed the simulated size and the orientation of the CR plume, and, in section 3.4.2, we describe how data assimilation changed the simulated salinity intrusion length in the CR estuary. In each section, we first determine, using independent data, whether data assimilation improved upon the non-assimilated models in simulating salinity of the CR estuary and plume. Then, we analyze how data assimilation changed the representation of ecologically significant features of interest in the CR estuary and plume.

3.4.1 Size and orientation of the CR plume

3.4.1.1 Accuracy of the assimilated solution

To determine whether data assimilation improved the simulated salinity in the CR plume, we compared the results of our data assimilation experiments to observations of salinity at fixed stations and to vertical profiles of salinity. At stations, where observational data were assimilated, data assimilation reduced root mean square (RMS) errors in simulated salinity from 2.0 to 1.1 psu at the station *rises*, from 2.3 to 1.3 psu at the station *risec*, and from 2.2 to 1.2 psu at the station *risen*. These reductions in errors were consistent at all times, and were especially significant during periods when DB14 had large errors. See the low-pass (T>30h) time series in Figure 16, panels III.a-III.c. For example, large errors in DB14 at the station *risen* on August 5th were substantially reduced by data assimilation, as shown in Figure 16, panel III.c.

The validation with independent data from the vertical profiles of salinity showed consistent improvement; see Table 1 for the list of RMS errors before and after data assimilation. All, but one short cruise in Table 1, showed a reduction in the RMS error. An average RMS error over all cruises was reduced by 19%, from 2.5 psu to 2.0 psu. Data for one of the research cruises in Table 1 are presented in Figure 17. The research cruise in Figure 17 was conducted on July 27th and measured vertical distribution of salinity and temperature inside of the CR estuary and in the southern branch of the CR plume. For the map of the cruise path see panel III in Figure 17. Vertical profiles of salinity and temperature inside of the CR plume are presented in Figure 17 on panels I and II respectively. On average, the RMS error for this cruise was reduced from 4.7 psu to 3.6 psu for salinity, and for temperature the RMS error was reduced from 2.7 °C to 1.7 °C. The strongest corrections from data assimilation were applied to vertical profiles of temperature, as seen in Figure 17, panels II.a and II.b. For example for the vertical profile at 12:16 PST, the RMS error was reduced from 3.3 °C to 0.5 °C.

The validations of our data assimilative model against data from the fixed stations and against data from the research cruises showed that data assimilation was more

accurate in simulating salinity distribution in the CR plume. However, these validation data lacked the synoptic coverage necessary to quantitatively validate the predicted size of the CR plume. One source of such synoptic data is the remote sensing data, such as airborne salinity measurements and satellite measurements of sea surface temperature, ocean color, and chlorophyll. Unfortunately, the airborne measurements of salinity in the CR plume were not available during the deployment of the RISE moorings. Our attempts to use proxy measurements—such as sea surface temperature, ocean color, and chlorophyll—were unsuccessful, primarily due to the extensive cloud cover and high uncertainty of comparisons with proxy measurements.

3.4.1.2 Size and orientation of the CR plume in assimilated and non-assimilated model

To characterize changes that data assimilation introduced to the dynamics of the CR plume, we compiled Figure 16 that presents the low-pass (T>30h) time series for:

- **Characteristics of the CR plume:** Panels I.a-I.c present the volume, area, and thickness of the CR plume, as was defined by the 28 psu isohaline. Panels II.a and II.b show the east-west and the north-south displacements of the CR plume centroid from the seasonal mean.
- **Measurements of salinity in the CR plume**, Panels III.a-III.c compare the observations of salinity (blue) at stations *rises*, *risec*, and *risen* with the simulated salinity from DB14 (green) and from data assimilation (red). Salinity observed at *risen* is compared with modeled salinity extracted at the location of *risen**, a virtual station located inside of the coastal jet for the CR plume.
- **Plume forcings:** Panels IV.a-IV.b show the magnitude of wind velocity in the east-west and the north-south directions. The wind velocities were observed at the National Oceanic and Atmospheric Administration buoy 46029 and are plotted in blue; the velocities that were used as model forcings are plotted in green. Panel V shows the observed time series of the river discharge that were used to force the river boundary in all models.
- **Snapshots of surface salinity:** Panels VI.a-VI.d present four selected cases of the CR plume before and after data assimilation.

Size of the CR plume: To characterize changes in the size of the CR plume, we plotted in Figure 16 the volume, area, and thickness of the CR plume, as was defined by the 28 psu isohaline. The comparison of the plume's size in non-assimilated DB14 (green line in Figure 16, panels I.a-I.c) and assimilated model (red line) suggests that data assimilation episodically changed the volume and the area of the CR plume, while maintaining similar plume thickness at most times. It is likely that the size of the CR plume estimated by data assimilation was more representative of the true size of the CR plume than the size predicted by DB14, since the timing of the changes in the estimated size coincided with periods when data assimilation was correcting for large errors in DB14.

To illustrate how correction of large errors in non-assimilated DB14 changed the size of the CR plume, we considered two examples: one of a larger plume, on July 9th, and one of a smaller plume, on August 8th. On July 9th, the simulated salinity in DB14 was overpredicted (too saline) at stations *risec* (Figure 16.III.b) and *rises* (Figure 16.III.a), suggesting that the true plume of the CR was larger than predicted by DB14. After data assimilation corrected for these overpredictions, the size of the CR plume increased, as seen in a snapshot in Figure 16.IV.a and in the time series of the volume and area of the CR plume in Figure 16.I.a and 16.I.b. In the second case, on August 8th, data assimilation corrected underprediction of salinity at stations *rises* and *risec*, resulting in a smaller plume, as seen in a snapshot in Figure 16.VI.b and in the time series of the volume and area of the CR plume in Figure 16.I.a and 16.I.b.

Orientation of the CR plume: To characterize changes in the orientation of the CR plume, we plotted in Figure 16 the deviation of the CR plume from the location of the plume's seasonal centroid, which was located in summer of 2004 within 1.7 km from the station *risec*. As with the size of the CR plume, data assimilation changed the position of the centroid episodically, usually coinciding with periods when large errors in non-assimilated DB14 were corrected. To illustrate the result of these corrections, we selected two episodes when the position of the CR plume was changed dramatically by data

assimilation. In the first episode, on August 5th, the CR plume was redirected by data assimilation from south to north, as a result of correcting for a large overprediction of salinity at the station *risen* and for a smaller underprediction at the station *rises*, seen in Figure 16.III.c and 16.III.a. In the second episode, on August 25th, the CR plume was restructured from a uni-directional to bi-directional, in response to correcting for the overprediction of salinity at the station *rises*.

3.4.1.3 Summary

The analysis of the CR plume variability showed that data assimilation changed the size and the orientation of the CR plume, primarily in response to correcting for episodes of large errors in non-assimilated model (DB14). The comparison of the assimilated solution with independent data suggested that these corrections were likely to be realistic, since the assimilated model was closer to the true ocean than the non-assimilated DB14. Since the direct measurements of the area and volume of the CR plume were not available, it was impossible to quantify the accuracy of the predicted area and volume before and after data assimilation.

3.4.2 Salinity intrusion in the CR estuary

3.4.2.1 Accuracy of the assimilated model

To determine whether data assimilation improved upon the non-assimilated model (DB16) in representing salinity intrusion length in the CR estuary, we used four, progressively more complex, validation studies: (1) a cross-validation study that used data from the fixed observation stations, (2) a validation study that used independent data collected by research cruises, (3) a validation study against direct, but sparse, measurements of the salinity intrusion length in the CR estuary, and (4) a validation study against data at the station *eliot* that recorded several infrequent events of extreme salinity intrusion in the CR estuary.

Cross-validation study: To determine whether on average data assimilation improved the simulated salinity in the CR estuary, we used a cross-validation study described in Chapter 2 of this thesis. The cross-validation established that data

assimilation was effective in reducing salinity errors in the CR estuary. Specifically, before assimilation, an average RMS error for 15 salinity sensors in the CR estuary was 4.3 psu. After assimilation, the salinity RMS error was reduced to 1.2 psu for the assimilation stations, and to 3.0 psu for the cross-validation stations.

Validation against cast data: To determine whether data assimilation improved the simulated salinity in the navigational channel of the CR estuary, where the length of the salinity intrusion is measured, we validated the data assimilation results against data collected by research cruises in the navigation channel of the CR estuary. During the period of our study, data from two such cruises were available. In the first cruise, on September 20th, 2004, the salinity RMS error was reduced from 5.0 to 4.0 psu. In the second cruise, shown in Figure 18, the RMS error was reduced from 3.3 to 1.8 psu. In addition to the reduction of the RMS error, the visual comparison of vertical profiles for non-assimilated DB16 and for data assimilation in Figure 18 suggests that data assimilation results were more consistent with observational data. For example, both the observational data and the data assimilation results show non-zero bottom salinity of ~2 psu after 11:45 PST, unlike the non-assimilated DB16 that shows 0 psu salinity after 11:45 PST.

Validation against observations of the salinity intrusion length: To determine whether data assimilation predicted the salinity intrusion length more accurately than DB16, we compared the simulated salinity intrusion length in the CR estuary with a few direct observations of the salinity intrusion length. During the period of our model study (2003-2004) the only available observation of the salinity intrusion length came from the research cruise on May 09, 2003; see Figure 18 for the time series of observed vertical salinity profiles. To determine the length of the salinity intrusion based on the profiles of vertical salinity, we used the time series of the bottom salinity extracted from these vertical profiles and shown in Figure 19, panel I. When the observed bottom salinities of two consecutive profiles were on different sides of the 2 psu cutoff, we estimated the observed salinity intrusion length using linear interpolation. These observed intrusion lengths are compared to the predicted lengths in Figure 19, panel II. The model-data

comparisons in Figure 19 show that data assimilation better represented the bottom salinity in the CR estuary and, as a consequence, better represented the salinity intrusion length. For example, data assimilation reduced the RMS error for the bottom salinity from 6.0 psu to 3.0 psu, and, when the data were available, the RMS error for the salinity intrusion length was reduced from 9.7 to 1.6 km.

Validation against measurements of the strong salinity intrusion events: To determine whether data assimilation improved representation of infrequent events of strong salinity intrusion in the CR estuary, we compared simulated salinity with observations of salinity collected from the station *eliot*. As seen in Figure 20 (observational data plotted as a solid line) the sensor at *eliot* saw salt only occasionally, usually during periods of low river discharge and neap tides. Unfortunately, both the assimilated and the non-assimilated DB16 failed to propagate salinity as far as *eliot*. Hence, in Figure 20, we compared the observations of salinity at *eliot* with the simulated salinity at the station *encu5*—an artificial station in the navigational channel of the CR estuary, located 4.5 km downstream of *eliot*.

Comparisons of data in Figure 20 shows that, prior to data assimilation, DB16 (dashed line in panel I) was able to represent only few events of strong salinity intrusion, *e.g.*, on July 27th and on August 11th. After data assimilation (dashed line in panel II), the representation of events was greatly improved. For example, all of the salinity intrusion events that were registered at the station *eliot* were reproduced at the station *encu5* by data assimilation, shown as a dashed line in panel II. The assimilated solution was also more accurate in reproducing the timing and magnitude of the salinity intrusion events than the non-assimilated DB16, *e.g.*, compare the timing of the event on July 12th shown in Figure 20. Unfortunately, the assimilated solution was able to represent the salinity intrusion events only when observations of salinity upstream of *tpoint* were available, specifically from the stations *mottb* and *cbnc3*. As shown in Figure 20, panel III, when observational data from *mottb* and *cbnc3* were not assimilated, the assimilated solution lost the ability to represent the events of strong salinity intrusion.

3.4.2.2 Comparison of salinity intrusion length in assimilated and non-assimilated model

In section 3.4.2.1, we established that data assimilation was more accurate in simulating the length of salinity intrusion in the CR estuary than non-assimilated DB16. To understand the differences between the dynamics of salinity intrusion length in assimilated and non-assimilated model, we:

1) Compared the daily maximum salinity intrusion length in assimilated and non-assimilated models; and
2) Compared the response of the daily maximum salinity intrusion length to river discharge in assimilated and non-assimilated models.

Daily maximum salinity intrusion length: To characterize the differences in the representation of salinity intrusion length in the assimilated and non-assimilated models, we plotted in Figure 21 the time series of the daily maximum salinity intrusion length in the CR estuary before and after assimilation. Figure 21 shows that salinity intrusion length was longer in assimilated (black line) than in non-assimilated DB16 (gray line), by an average of 3.6 km, i.e., the average salinity intrusion length was 32.9 km in DB16 and, 36.5 km in data assimilation. Although data assimilation was successful in extending the average length of salinity intrusion, the absolute maximum of salinity intrusion, which usually coincides with period of low river flow and neap tides, remained the same after assimilation (~40 km) and was short of the historically observed 52 km (Jay, et al. 1990).

It is likely that inability of data assimilation to propagate the maximum length of salinity intrusion demonstrated the limitation of the reduced-dimension Kalman filter—the data assimilation algorithm used in our experiments. In fact, since the reduced-dimension Kalman filter estimates the dynamics of the CR estuary in the EOF subspace of the forward model, the estimated salinity intrusion can not exceed the maximum length in the non-assimilated model. To verify this hypothesis empirically, we compared data assimilation results that were computed from two different hindcast models: DB16 that had higher maximum intrusion length of 43 km and DB14 that had lower maximum intrusion length of 40 km. In Figure 22, we plotted the salinity intrusion length for these

two databases before and after data assimilation. Before assimilation (panel a), the salinity intrusion length differed between DB14 and DB16 during periods of high salinity intrusion and was the same during periods of low salinity intrusion. After assimilation (panel b), salinity intrusion length shows similar behavior in both assimilated databases; however, as our hypothesis suggested, the assimilated solution based on DB16 showed higher salinity intrusion length, by 2 km. This analysis, together with the results of validation studies in section 3.4.2.1, suggests that, during periods of low salinity intrusion, the estimate of salinity intrusion length is more accurate in the assimilated model than in the non-assimilated model. However, during periods of high salinity intrusion, the estimate in the assimilated model is at least as accurate as in the non-assimilated model.

Response to river discharge: To determine whether data assimilation not only changed the average salinity intrusion length in the CR estuary, but also changed the response of the salinity intrusion length to variation in the CR river discharge, we compared the regression relationships for the assimilated (solid line in Figure 23) and the non-assimilated (dashed line in Figure 23) model of the CR estuary. These relationships regressed the length of the daily maximum salinity intrusion length on the daily average river flow. As Figure 23 shows, these regression relationships differed, suggesting that the salinity dynamics in the assimilated and non-assimilated model differed. Specifically, the assimilated model showed a weaker response to changes in the river discharge (lower R^2), hence indicating that other factors may play a stronger role in determining the salinity intrusion length in the CR estuary. A further validation of regression lines in Figure 23 is warranted as more observations of salinity intrusion will become available.

3.4.2.3 Summary

Model-data comparisons showed that data assimilation had more realistic representation of the salinity intrusion length in the CR estuary than non-assimilated DB16. Specifically, data assimilation was able, with limited success, to correct for the underestimation of the salinity intrusion length in the non-assimilated model. Data assimilation also improved the representation of infrequent events of strong salinity

intrusion in the CR estuary, specifically the timing and the magnitude of these events. On average, the salinity intrusion length in data assimilation was longer, by 3.6 km, and had a weaker response to the variations in the CR flow than non-assimilated model had. In spite of multiple improvements, the maximum salinity intrusion length in the assimilated model was still too short and was limited by the maximum salinity intrusion length in the non-assimilated model. To further improve the prediction of the maximum salinity intrusion length, an improvement in the dynamics of the non-assimilated model will be needed.

3.5 Summary and conclusions

The reduced-dimension Kalman filter was applied to estimate the variability of three ecologically significant circulation features of the CR estuary and plume: the size of the CR plume, the orientation of the CR plume, and the length of the salinity intrusion in the CR estuary. From the analysis of our data assimilation estimates, we determined that:

1) Data assimilation was able to constrain the simulated dynamics of the CR estuary and plume, producing more accurate estimates of salinity than the non-assimilated model;
2) Data assimilation changed the simulated size and the orientation of the CR plume, usually in response to correcting for episodes of large errors in the non-assimilated model;
3) Data assimilation corrected, with a limited success, for the underestimation of the simulated salinity intrusion length in the CR estuary, and improved the representation of infrequent events of strong salinity intrusion;
4) Data assimilation estimates had weaker response of the salinity intrusion length to variations in the CR discharge.

Acknowledgements

We thank Michael Wilkin for his work on maintaining the CORIE observational network; Dr. Ed Dever for providing data from RISE moorings; Professor Todd Leen, Rudolph van der Merwe, and Zhengdong Lu for help with the developing the code of the original data assimilation system; and Michela Burla for providing advanced copies of her papers.

The National Science Foundation (ACI-0121475, OCE-0424602) and National Oceanic and Atmospheric Administration (AB133F-04-CN-0033) provided financial support for this research. Any statements, opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views or policies of the federal sponsors, and no official endorsement should be inferred.

Figures

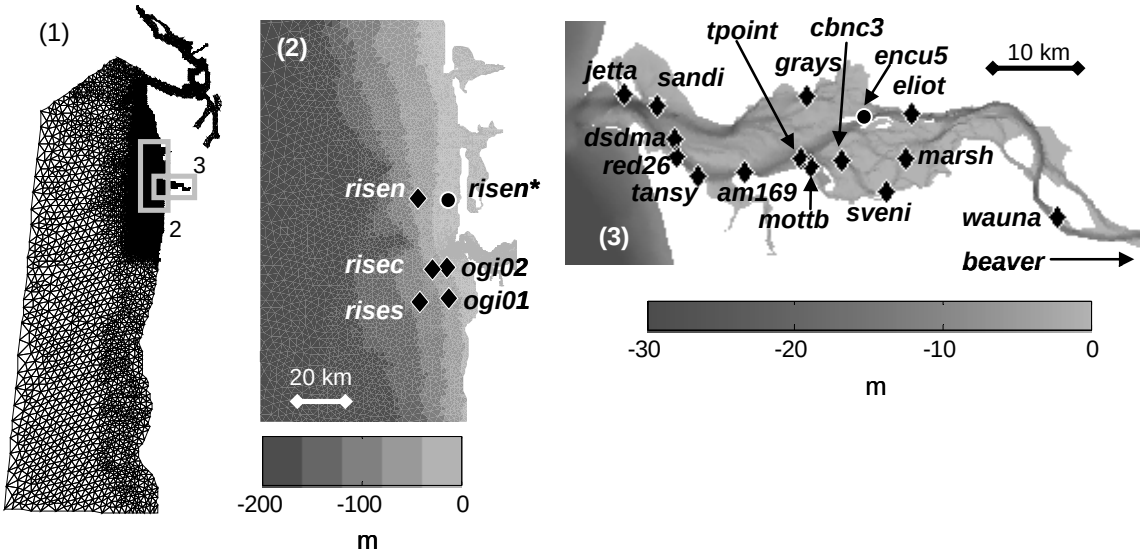


Figure 15: Maps of the computational domain and observation stations.
(1) The computational grid for DB14. Maps of observation stations in the CR plume (2) and the CR estuary (3). The map of stations in panels 2 and 3 is overlaid over the bathymetry. Stations *risen*, *risec*, and *rises* were operated by the RISE project (www.ocean.washington.edu/rise/). Tide gages *tpoint* and *wauna* were operated by National Oceanographic and Atmospheric Agency. Stations *risen** and *encu5* on panels 2 and 3 are locations of virtual stations that were used in our experiments.

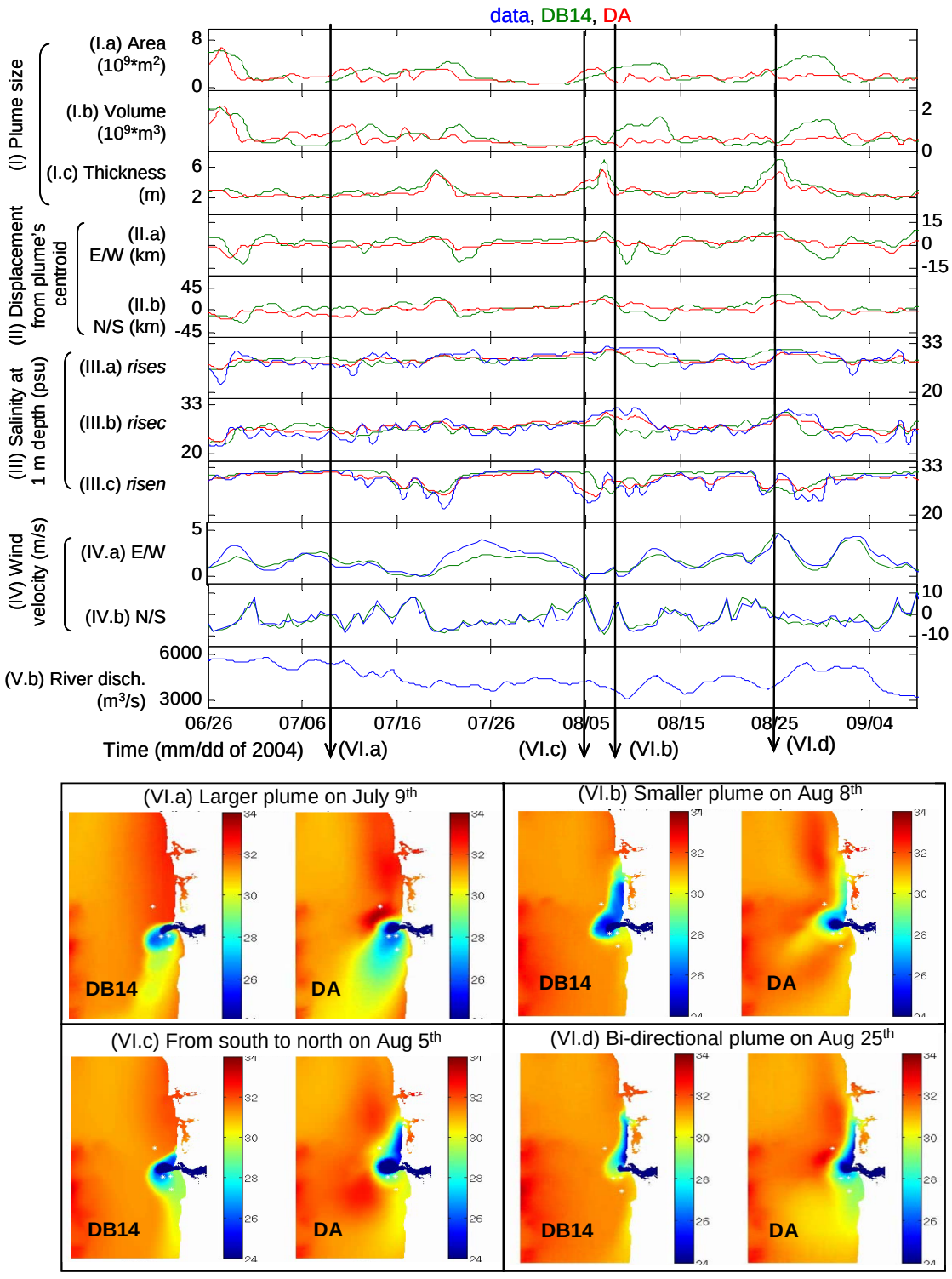


Figure 16: Time series of CR plume parameters and forcings.
(I) Size of the CR plume. (II) Displacement of the plume's centroid from the seasonal mean, which in summer of 2004 was located within 1.7 km of *risec*. (III) Salinity at stations *rises*, *risec*, and *risen*. (IV) Wind velocities at the NOAA buoy 46029. (V) River discharge at the USGS station Beaver Army. (VI) Snapshots of surface salinity from DB14 and DA. On panels (I-V) observational data is in blue, DB14 is in green, and DA is in red. Time series in panels (I-V) were filtered using low-pass filter (T>30h).

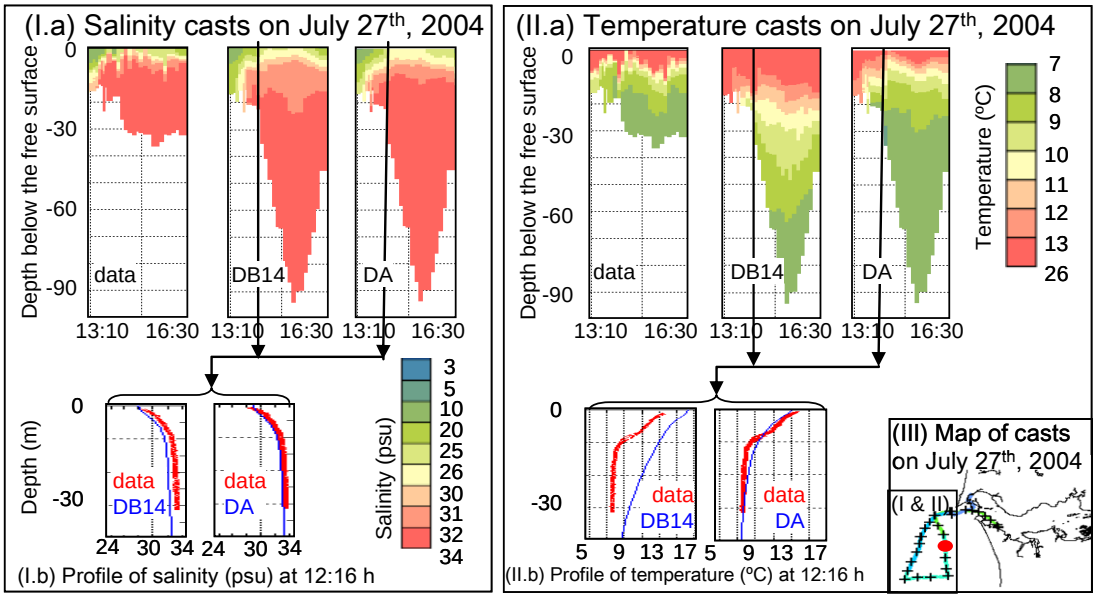


Figure 17: Data from research cruises on July 27th, 2004.
(I) Salinity and (II) temperature profiles from research cruise on July 27th, 2004. (III) Map of the research cruise. (I.a and II.a) Vertical profiles of salinity and temperature in the CR plume. (I.b and II.b) A selected profile of salinity and temperature in the CR plume. Times of the casts are in PST. The wire frame in panel (III) highlights the period of the research cruise shown in panel (I and II).

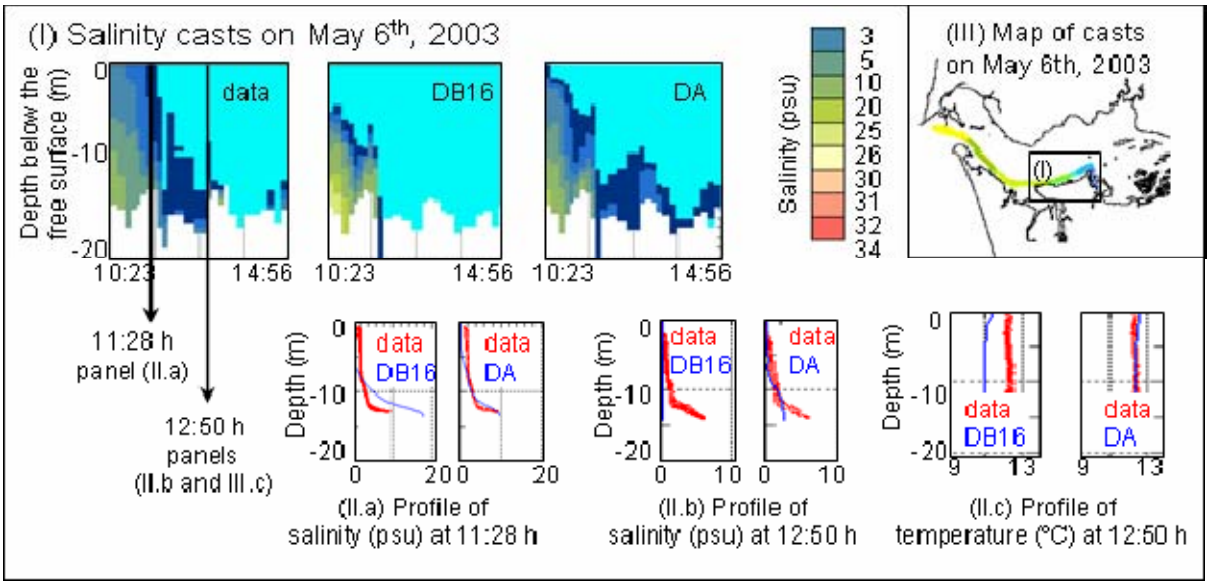


Figure 18: Data from research cruises on July 27th, 2004.
(I) Salinity profiles from research cruise on May 6th, 2003. (II.a) Salinity profile at 11:28. (II.b) Salinity and (II.c) and temperature profiles at 12:50. (III) Map of the research cruise. The wire frame in panel (III) highlights the period of the research cruise shown in panel (I).

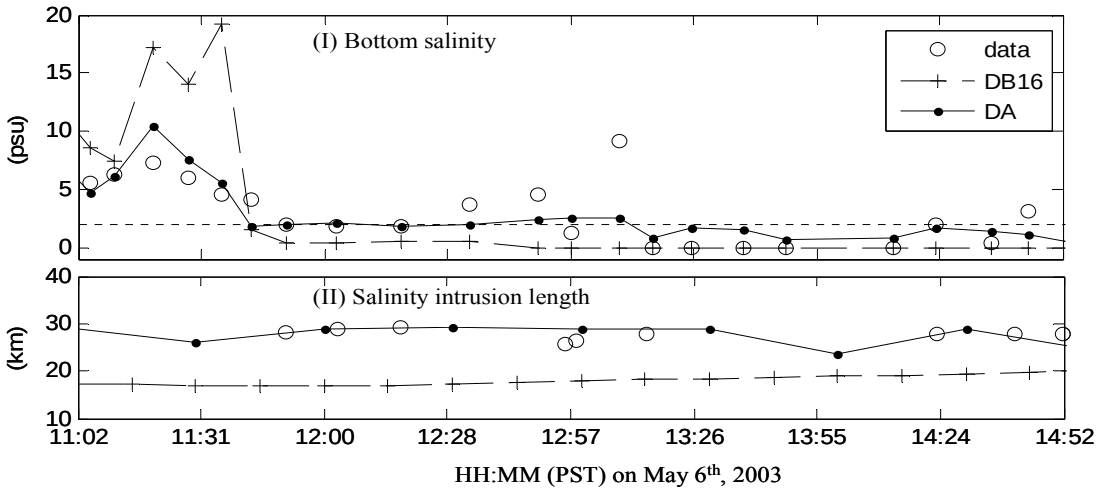


Figure 19: Salinity intrusion length interpolated from measurements of bottom salinity. (I) Bottom salinity measured by research cruise on May 6th, 2003. For the map of the cruise see Figure 18, panel III. (II) Salinity intrusion length during the cruise on May 6th, 2003. Observational data is marked with (o), DB16 data is marked with (+), and DA data is marked with (.). Observations of salinity intrusion were estimated from the time series of the bottom salinity using linear interpolation.

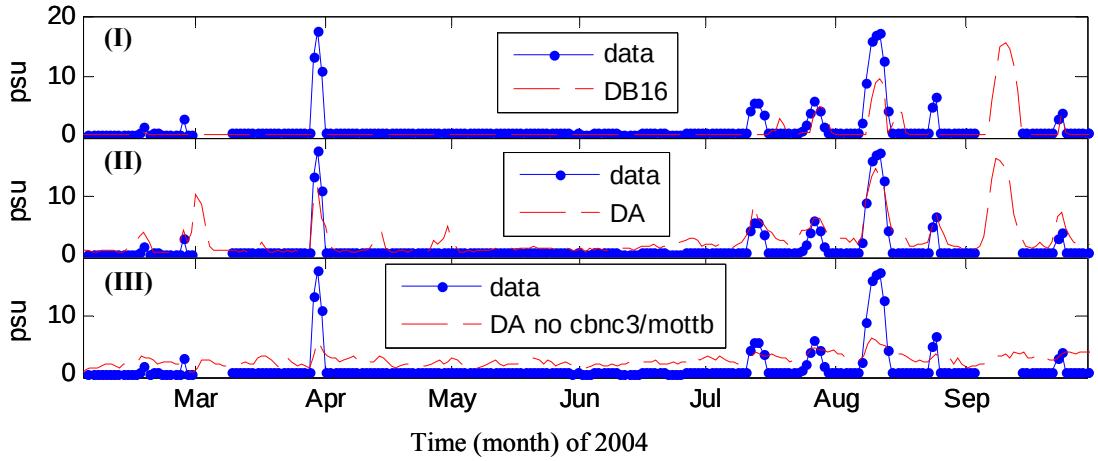


Figure 20: Daily salinity maximums observed at eliot and simulated at encu5.
(I) Model data is from DB16; (II) model data is from data assimilation; and (III) model data is from data assimilation run, where data from *cbnc3* and *mottb* were not used in assimilation.

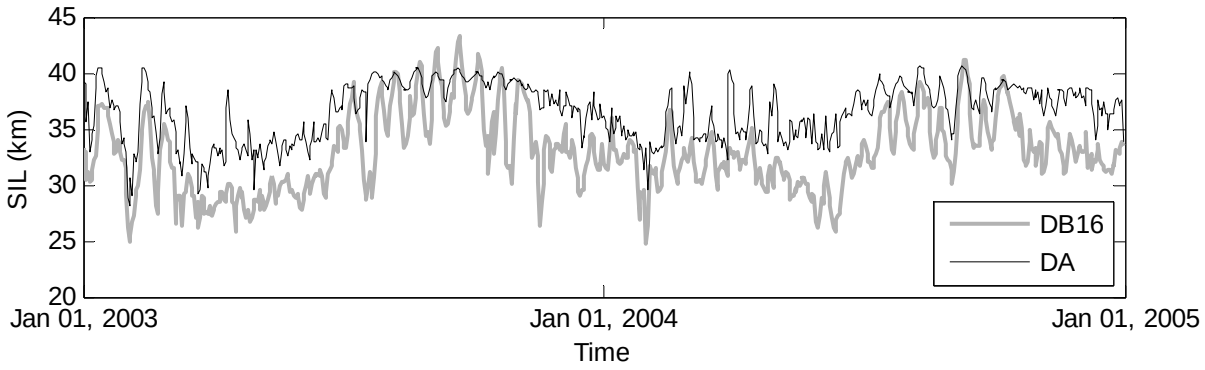


Figure 21: Daily salinity maximum of salinity intrusion length (SIL) before (gray) and after data assimilation (black)

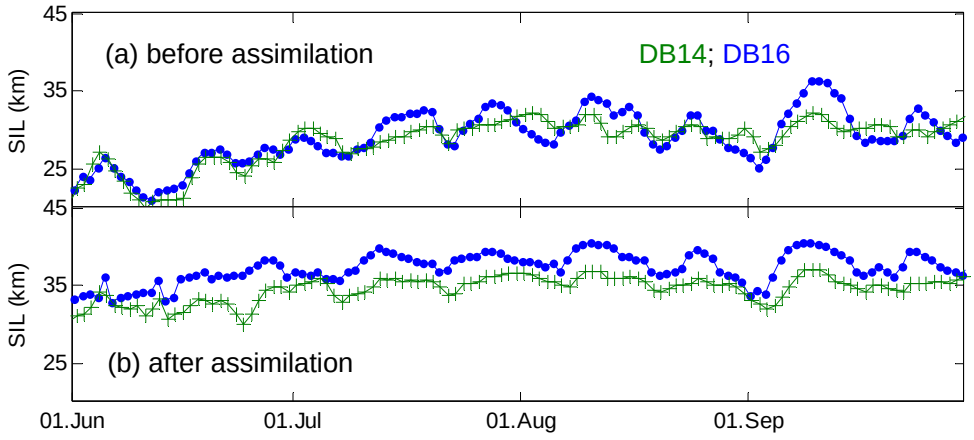


Figure 22: Daily salinity maximum of salinity intrusion length (SIL) as represented by two hindcast databases: DB14 (marked with '+') and DB16 (marked with '.'). Top panel, SIL before assimilation. Bottom panel, SIL after assimilation.

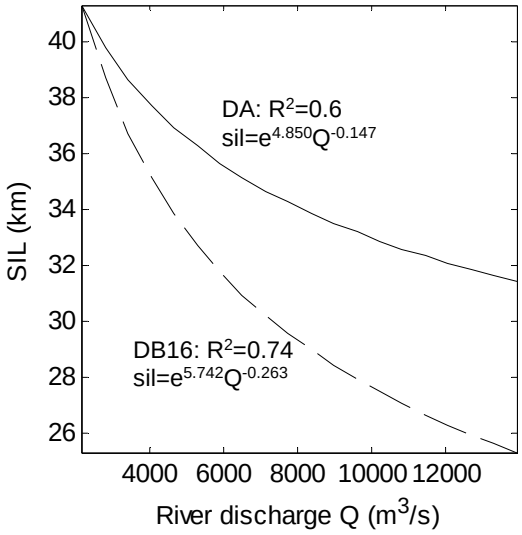


Figure 23: Regression relationships for river discharge (Q) vs. salinity intrusion length (SIL).
Regression line for DA is plotted in solid line and for DB16, in dashed line.

Tables

Table 1: Root mean square errors (RMSE) for cast data in the CR plume.

Date	RMSE			cruise description	
	Before DA (DB14)	After DA	Change (%)	location	# of casts
7/10/04	2.4	2.1	9	ogi02	15
7/11/04	1.6	1.3	19	ogi02	53
7/12/04	1.2	1.5	-27	mouth	11
7/13/04	2.0	1.7	12	north/center/south	89
7/14/04	1.5	1.3	11	mouth	14
7/15/04	1.6	1.3	20	north	78
7/20/04	1.9	1.7	10	mouth	13
7/24/04	1.9	1.7	13	ogi02	25
7/25/04	2.0	1.8	10	ogi02/south	51
7/27/04	4.7	3.6	24	estuary + plume south	72
all	2.5	2.0	19		

References:

Baptista, A. M. (2006), "CORIE: The First Decade of a Coastal Margin Collaborative Observatory," in *Oceans'06 MTS/IEEE*, Boston.

Baptista, A. M., Zhang, Y.-L., Chawla, A., Zulauf, M. A., Seaton, C., Myers, E. P., Kindle, J., Wilkin, M., Burla, M., and Turner, P. J. (2005), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: Ii. Application to the Columbia River," *Continental Shelf Research*, 25, 935-972.

Bottom, D. L., Simenstad, C. A., Burke, J., Baptista, A. M., Jay, D. A., Jones, K. K., Casillas, E., and Schiewe, M. H. (2005), "Salmon at River's End. The Role of the Estuary in the Decline and Recovery of Columbia River Salmon," NOAA Technical Memorandum NMFS-NWFSC-68, U.S. Dept. Commer.

Burla, M., Baptista, A. M., Casillas, E., and Williams, J. G. (submitted), "The Influence of the Columbia River Plume on the Survival of Steelhead (Oncorhynchus Mykiss) and Chinook Salmon (O. Tshawytscha): A Numerical Exploration.," *Can. J. Fish. Aquat. Sci.*

Burla, M., Baptista, A. M., Zhang, Y.-L., Casillas, E., Bottom, D. L., and Simenstad, S. A. (2007), "Salmon Habitat Opportunity in the Columbia River Estuary: Modeling the Physical Environment to Inform Management Decisions," in *Coastal Zone '07*, Portland, OR.

Dee, D. P. (1995), "On-Line Estimation of Error Covariance Parameters for Atmospheric Data Assimilation," *Monthly Weather Review*, 123, 1128-1145.

Foreman, M. G. G. (1977), "Manual for Tidal Heights Analysis and Prediction," vol. 58, Institute of Ocean Sciences.

Garcia-Berdeal, I., Hickey, B., and Kawase, M. (2002), "Influence of Wind Stress and Ambient Flow on a High Discharge River Plume," *Journal of Geophysical Research*, 107 (C9), 3130.

Hamilton, P. (1990), "Modeling Salinity and Circulation for the Columbia River Estuary," *Progress in Oceanography*, 25, 113-156.

Hickey, B., Geier, S., Kachel, N., and MacFadyen, A. (2005), "A Bi-Directional River Plume: the Columbia in Summer," *Continental Shelf Research*, 25 (14), 1631-1656.

Hickey, B. M., Pietrafesa, L. J., Jay, D. A., and Boicourt, W. C. (1998), "The Columbia River Plume Study: Subtidal Variability in the Velocity and Salinity Fields," *Journal of Geophysical Research*, 103 (C5), 10,339-10,369.

Jay, D., and Smith, J. D. (1990), "Circulation, Density Distribution and Neap-Spring Transitions in the Columbia River Estuary," *Progress in Oceanography*, 25 (1-4), 81-112.

Jay, D. A., and Flinchem, E. P. (1997), "Interaction of Fluctuating River Flow with a Barotropic Tide: A Demonstration of Wavelet Tidal Analysis Methods," *Journal of Geophysical Research*, 102 (C3), 5705-5720.

Lu, Z., Leen, T. K., van der Merwe, R., Frolov, S., and Baptista, A. M. (2007), "Sequential Data Assimilation with Sigma-Point Kalman Filter on Low-Dimensional Manifold," *submitted to Journal of Inverse Problems*, 30, also available at http://www.stccmop.org/files/CMOP-TR-07-001.pdf [Viewed on September 14, 2007].

Martin, P. J. (2000), *Description of the Navy Coastal Ocean Model Version 1.0.*, Naval Research Laboratory, Technical Report.

Myers, E. P., and Baptista, A. M. (2001), "Inversion for Tides in the Eastern North Pacific Ocean," *Advances in Water Resources*, 24 (5), 505-519.

van der Merwe, R. (2004), "*Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*," Ph.D., OHSU.

van der Merwe, R., Leen, T. K., Lu, Z., Frolov, S., and Baptista, A. M. (2007), "Fast Neural Network Surrogates for Very High Dimensional Physics-Based Models in Computational Oceanography," *Neural Networks*.

van der Merwe, R., and Wan, E. (2003), "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models," in *Workshop on Advances in Machine Learning*, Montreal, Canada.

Zhang, Y.-L., and Baptista, A. M. (submitted), "A Semi-Implicit Eulerian-Lagrangian Finite-Element Model for Cross-Scale Ocean Circulation," *Ocean Modeling*.

Chapter 4

A real-time data assimilative nowcast and forecast of the baroclinic circulation in the Columbia River estuary

Authors: S. Frolov, Baptista A. M., Turner P., Seaton C., Wilkin M.

To: ASCE journal of waterway, portal, and coastal engineering

Abstract

A very fast nowcast-forecast system was developed for the circulation of the Columbia River (CR) estuary. The computational efficiency of the developed system was due to the new technology for model-surrogate prediction and reduced-dimension Kalman filtering. The developed system was used daily to assimilate three days of observational data and to predict the state of the estuary up to 3 days in the future. The assimilated data included measurements of salinity, temperature, and water levels from fixed *in situ* sensors in the CR estuary. Our validation studies showed that the model-surrogate forecast was as accurate as the forecast using the traditional circulation model, and, when observational data were assimilated, the errors in the simulation of the CR estuary were reduced substantially. The computational cost of the developed nowcast-forecast system was only a fraction (1/100) of the cost of the traditional non-assimilated circulation model.

4.1 Introduction

Accurate descriptions of the current and future state of the coastal ocean are in demand by government agencies, industries, researchers, and recreational users[18]. Some of this descriptions are already available from operational observation systems like PORTS®[19] and from forecast and nowcast of the ocean state simulated with circulation models like POM (Blumberg, et al. 1987), ROMS (Haidvogel, et al. 2000), and SELFE (Zhang, et al. submitted). However, observations of the ocean tend to be sparse in space or time, and the circulation models may have erroneous predictions. Data assimilation (DA) is a modeling technique that addresses these limitations by constraining the dynamics of a circulation model using sparse observations of the ocean. Recently, an operational assimilative nowcast of a coastal ocean circulation became a reality (Robinson 1999, Martin 2000, Kindle 2005, Lermusiaux 2007). However, an assimilative nowcast of circulation in a coastal margin—*e.g.,* in estuaries, tidal inlets, and river plumes—is yet to be developed.

A major impediment for developing DA nowcasts for coastal margin circulation is the high computational cost of assimilative models. For example, (Ngodock, et al. 2006) reported that computational cost of an advanced DA algorithm can be equivalent to 500 simulations of a traditional circulation model. To enable fast DA in coastal margin models, we recently developed a technology for model-surrogate prediction (van der Merwe, et al. 2007), which enables fast DA by approximating a slow traditional circulation model with a fast neural network. To train the neural network, a long hindcast simulation of the coastal ocean is usually used. Model surrogates were previously used by us (in Chapter 2) to expedite DA of the baroclinic circulation in the CR estuary. In this article, we report on the extension of this hindcast system to a real-time nowcast-forecast system and on the application of the developed nowcast-forecast system in support of research cruises in the CR estuary.

---

[18] For an overview of possible applications see: http://www.mercator-ocean.fr/html/applications/index_en.html

[19] Physical Oceanographic Real-Time System® operated by National Oceanic and Atmospheric Administration and accessible at http://tidesandcurrents.noaa.gov/ports.html.

4.2 Coastal observatory for the CR estuary

The application of our DA nowcast-forecast system was for the CR estuary. The CR estuary, a classic river-dominated estuary, is a highly energetic and dynamic system that responds quickly to changes in ocean tides, regulated river discharge, and coastal winds. The tidal dynamics in the CR are significant (tidal amplitudes of up to 3.6 m) and are driven by the non-linear interaction of astronomic tides with complicated bathymetry, non-stationary river discharge, and coastal wind (Jay, et al. 1997). Compressed and often stratified, the estuarine circulation in the CR is subject to extreme variations in salinity intrusion and stratification regimes (Jay, et al. 1990).

CORIE—a coastal observatory for the CR estuary and plume—studies the dynamics of the CR estuary using a real-time observation network (Baptista 2006) and a modeling system for the 3D baroclinic circulation of the CR estuary and the adjacent ocean (Baptista, et al. 2005). At the time of our experiments, CORIE observatory had three major components:

1) A real-time observation system that included a network of fixed stations and research cruises. Fixed stations utilized in our experiments are shown in Figure 24. At each station, an array of sensors measured a variable combination of parameters, including water level, salinity, temperature, velocity, air temperature, and wind velocity.
2) A modeling system for simulating the 3D baroclinic circulation of the CR estuary, plume, and adjacent coastal ocean. The model simulations used in our experiments were generated using an Eulerian-Lagrangian model SELFE (Zhang, et al. submitted) and were forced with a combination of realistic atmospheric, ocean, and river forcings.
3) A hindcast and forecast DA system, based on the RDKF algorithm (Lu, et al. 2007). The hindcast DA system was described in Chapter 2. In this paper, we describe the implementation of the RDKF algorithm in a nowcast system.

4.3 Data assimilation method

To assimilate observational data into a model of the CR estuary and plume, we used the RDKF algorithm (Lu, et al. 2007)—a recently developed DA method, which is fast, model-independent, adjoint-free, and nonlinear. The computational efficiency of the new method comes, in part, from the use of a neural network model surrogate (van der Merwe, et al. 2007) that executes forward simulations three to four orders of magnitude faster than a traditional numerical circulation code. The new method was successfully applied for hindcast DA in the CR estuary (Chapter 2 of this thesis) and, in this research, was extended to the real-time nowcast. Following is the short overview of the framework for the RDKF that we used in our experiments:

1) A long, statistically representative hindcast simulation of the CR system was generated using the forward model SELFE (Zhang, et al. submitted). Similarly to Chapter 2 of this thesis, we used hindcast database DB16 for training of the model surrogate.
2) Samples drawn from the hindcast database in step (1) were used to train a model surrogate, using training algorithms developed in (van der Merwe, et al. 2007). Because it was not computationally feasible to train the model surrogate in the high-dimensional space of the forward model, the model surrogate was trained in the subspace spanned by the empirical orthogonal functions (EOFs) computed from the hindcast database.
3) The state of the model surrogate was estimated using a Kalman filter of choice. Because the estimated state of the model surrogate was small (50 degrees of freedom), we choose to use the sigma point Kalman filter–a state-of-the-art, full-rank Kalman filter (van der Merwe, et al. 2003).
4) The output of the Kalman filter was analyzed by reconstructing the estimated state and the estimated model error from the EOF subspace, where the model surrogate and the Kalman filter operated, to the full space of the model, where many analysis and visualization tools operated.

Since all DA experiments in step (3) above were performed in the reduced EOF subspace, CPU, memory, and storage requirements for these DA experiments were very low, as will be demonstrated later in section 4.4.2.

4.4 Implementation of the real-time nowcast system

The real-time nowcast-forecast system was implemented using the RDKF algorithm, which requires the following list of inputs for its computation:

1) A model surrogate, which is used to expediently update the ensemble of model states;
2) An initial condition for the state and the error covariance;
3) A description of the process noise model that characterizes uncertainty in the prediction of the model surrogate and in the accuracy of the model forcings;
4) A times series of systems forcings, such as wind stress, river discharge, and ocean tides; and
5) Observations of the environment.

The implementation of the real-time nowcast was conducted in two stages. In the first, off-line, stage, we trained a model surrogate and calibrated process-noise models of the Kalman filter, items 1-3 above. For a brief summary of experimental procedures used in this off-line stage see section 4.4.1, or see Chapter 2 of this thesis for a detailed description of methods. In the second, on-line, stage, we estimated the state of the CR estuary using a RDKF, which was implemented as a real-time nowcast-forecast system. The real-time nowcast system, described in section 4.4.2, was responsible for the timely collection of inputs and observational data that were used by the DA system, items 4 and 5 above.

4.4.1 Off-line training of the model surrogate and process-noise models

To implement the nowcast system, we used the model surrogate and the process-noise model that were trained and validated in our hindcast experiments, described in detail in Chapter 2 of this thesis. Here, we briefly summarize the training procedures and the results of the validation studies.

To train the model surrogate, required by the RDKF for fast computation of the ensemble forecast, we used training algorithms described in (van der Merwe, et al. 2007). The model surrogate was trained to approximate the variability of the CR estuary, as was represented by a six-year-long hindcast simulation of the CR estuary (Baptista, et al. 2005, Baptista 2006). We will later refer to this simulation as DB16. To enable training of the model surrogate, the simulated dynamics of the CR estuary were projected on 50 leading EOFs, computed from the hindcast DB16. Both the EOF and the model surrogate were trained on five years of simulation data (from 1999 to 2003) and were validated for prediction on 1.5 years of simulation data (from 2004-2005). As documented in Chapter 2 of this thesis, the EOFs were equally accurate, within 2% of the normalized mean square error, on both training and test datasets. The model surrogate was slightly more accurate on training dataset (1.5%) than on test dataset (2.3%).

To calibrate the process-noise model, we used the *n*-fold empirical cross validation (Bishop 1995)—a machine learning techniques that reuses observational data to effectively calibrate and validate a DA algorithm. Our process-noise models included uncertainty in model forcings and model prediction. The uncertainty in model forcings were characterized by a scaled variance of the forcings, which included: river discharge and temperature; ocean tides, temperature, and salinity; wind stress; and atmospheric fluxes. The uncertainty in model prediction was approximated by the scaled variance of the model surrogate error. The scaling parameters for the forcing and model errors were optimized using an *n*-fold empirical cross validation.

4.4.2 Real-time nowcast system

To generate operational nowcasts of the CR estuary, we implemented machinery that ensured timely execution of programs that prepared the inputs for the DA algorithm, executed the DA algorithms, and post-processed outputs of the DA algorithm.

The dataflow diagram for the developed nowcast machinery is shown in Figure 25. As seen from this diagram, the developed DA nowcast model relied not only on the

external forcings, like the atmospheric weather forecast, but also on the output of the regional ocean model. We used the output of this regional ocean model to provide ocean boundary conditions for the DA model, which was embedded in the domain of the regional ocean model. The five main components of the nowcast machinery were:

1) Acquisition of the model forcings, which included atmospheric forcings from the National Center for Environmental Prediction, measured river discharge and temperature from the USGS station Beaver Army[20], and ocean boundary conditions from the Navy Coastal Ocean Model (Martin 2000) and from the tidal model of Foreman (Foreman 1977). The acquisition time for all forcing files was ~ 2 hours.
2) Regional ocean forecast with a traditional circulation model SELFE. This 7-day forecast was for the CR estuary, CR plume, and the adjacent coastal ocean. It took ~30 hours to generate this 7-day forecast using a single AMD Athlon© processor.
3) Acquisition of observational data. Observational data from the CORIE network were acquired using near real-time telemetry and for the station *tpoint*, using automated downloads from the NOAA website[21]. The acquisition time for observational data was ~15 minutes.
4) Assimilative nowcast and forecast. Once the inputs to the assimilative model were ready, a DA run was executed, which assimilated the previous two and a half days of observational data. The initial conditions from this assimilated run were then used for a three day forecast using model surrogate. It took under 5 minutes to generate both the hindcast and the forecast of the CR estuary.
5) Post processing of model results. The outputs of the DA nowcast and model surrogate forecast were processed using automated scripts and displayed as static and dynamic images on the web. The complete post-processing cycle took ~45 minutes.

[20] USGS stands for U.S. Geological Survey. Data for station Beaver Army is available at: http://waterdata.usgs.gov/usa/nwis/uv?site_no=14246900.
[21] NOAA stands for National Oceanic and Atmospheric Administration. Data for the station *tpoin* available at http://www.co-ops.nos.noaa.gov/.

The timing diagram for the developed nowcast-forecast system is presented in Figure 26. For each of the tasks, the filled rectangle shows the timing of the task execution, and the solid black line shows the length of the dataset produced by this task. As seen from the timing diagram, the DA model was executed once a day, resulting in a hindcast assimilation of the past two days of observational data, a single nowcast state estimate, and a three-day forecast. Because it took only five minutes to run the nowcast-forecast model, it is possible in the future to run the assimilative model more frequently than once a day, hence providing constant updates to the nowcast estimate of the CR estuary.

4.5 Evaluation of results

To determine the accuracy of the developed DA system, we computed an estimate of the average error for a hindcast-nowcast-forecast. The errors were averaged over 365 hindcast-nowcast-forecast cycles and over 8 water level sensors, 13 salinity sensors, and 15 temperature sensors. In each cycle of this experiment the state of the CR estuary was estimated for one previous day, using a hindcast-nowcast cycle, and was forecasted using model surrogate for two more days. The time series of the average errors are presented in Figure 27. In addition to the error of the developed DA system (in red), we plotted average errors for the non-assimilated forecast with the model surrogate (in green) and with the traditional circulation model SELFE (in blue).

The time series of the average RMS errors in Figure 27 show that:

1) When observational data were not assimilated, the model-surrogate forecast was as accurate as the forecast with the traditional circulation model, but only at a fraction of the computational cost, ~10,000 faster.
2) When observational data were assimilated, the accuracy of the CR estuary simulation was substantially improved, while the computational cost remained very low, ~100 faster than the non-assimilated forecast with the traditional circulation model SELFE.
3) When DA estimates were used to initialize the model-surrogate forecast, the positive impact of DA reduced quickly and the skill of the model-surrogate forecast did not differ substantially from the non-assimilated forecast. For

example, for salinity errors, the forecast skill improved only slightly, by <1 psu over the next 10 hours. In contrast, for water levels errors, the forecast skill degraded by ~0.1m for ~10 hours after the forecast initialization, possibly due to an initialization shock.

4.6 Application for adaptive data collection

Both the assimilated and the non-assimilated forecast of the CR estuary are routinely used to support adaptive field sampling programs in the CR estuary. An example of such program is the study of the microbiological diversity in the estuary turbidity maximum—a biologically unique region of the estuary, located close to the moving edge of salinity intrusion. During such sampling an ensemble of physical and microbiological data are collected using research vessel Forerunner, which is guided using model forecasts that are accessed using Internet-enabled computer on board of the research vessel.

An example of the salinity transect collected during one of such sampling programs on June 5th is displayed in Figure 28. Qualitatively, the assimilated forecast reproduced observations salinities better. For example, assimilated forecast represented the observed non-zero salinities after 13:34 hours, which non-assimilated forecast failed to represent.

4.7 Conclusions

A real-time assimilative hindcast-nowcast-forecast system was implemented for the CR estuary. The DA system used the RDKF as a DA algorithm and a model surrogate as a forecast algorithm. The DA system was validated and applied in a field exercise that involved adaptive data collection. The DA system was successful in assimilating observational data and reducing errors in simulated circulation of the CR estuary. When data were not assimilated, model-surrogate forecast was as accurate as the non-assimilated forecast with the traditional circulation model SELFE. However, the computational cost of the DA system was only a fraction of the cost of the traditional circulation model: ~100 times faster when observational data was assimilated, and ~10,000 when no data were assimilated in the model-surrogate forecast. The very low

computational cost of the our DA system shows great promise for enabling computationally inexpensive, real-time DA nowcasts and forecasts in challenging coastal environments, such as the baroclinicaly driven circulation in a river-dominated estuary.

Acknowledgments

We thank Professor Todd Leen, Rudolph van der Merwe, and Zhengdong Lu for help with the developing the code of the original data assimilation system. The National Science Foundation (ACI-0121475, OCE-0424602) and National Oceanic and Atmospheric Administration (AB133F-04-CN-0033) provided financial support for this research.

Figures

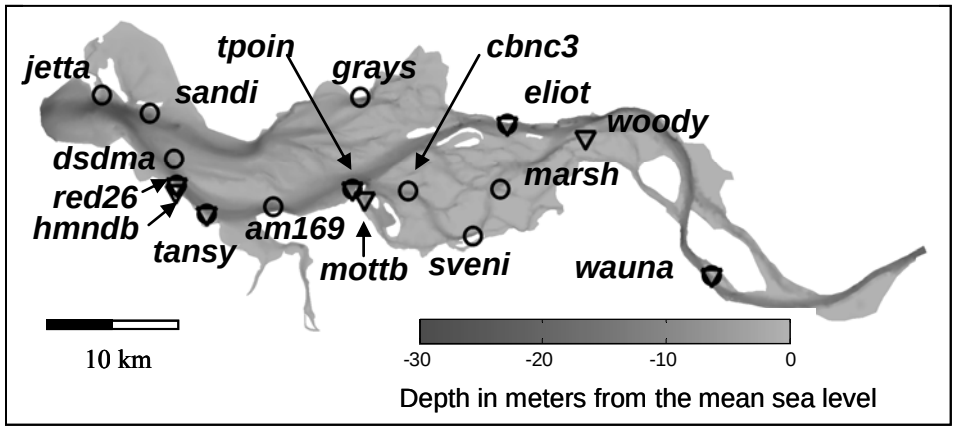


Figure 24: Map of the bathymetry in the computational domain.
Overlaid are the names and locations of the observation stations. Stations marked with the circles were used in the calibration of the DA algorithm. Stations marked with a triangle were used in the real-time forecasts.

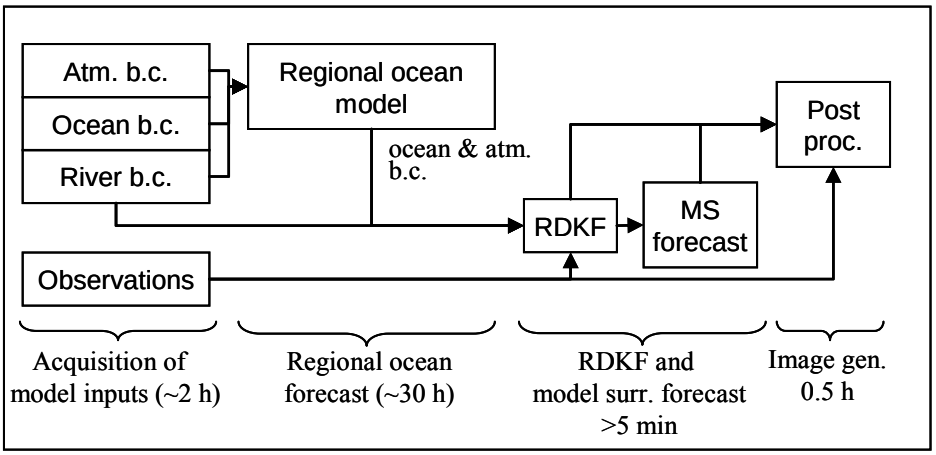


Figure 25: Data flow in the nowcast/forecast system.
MS—model surrogate. RDKF—reduced-dimension Kalman filter. b.c.—boundary conditions.

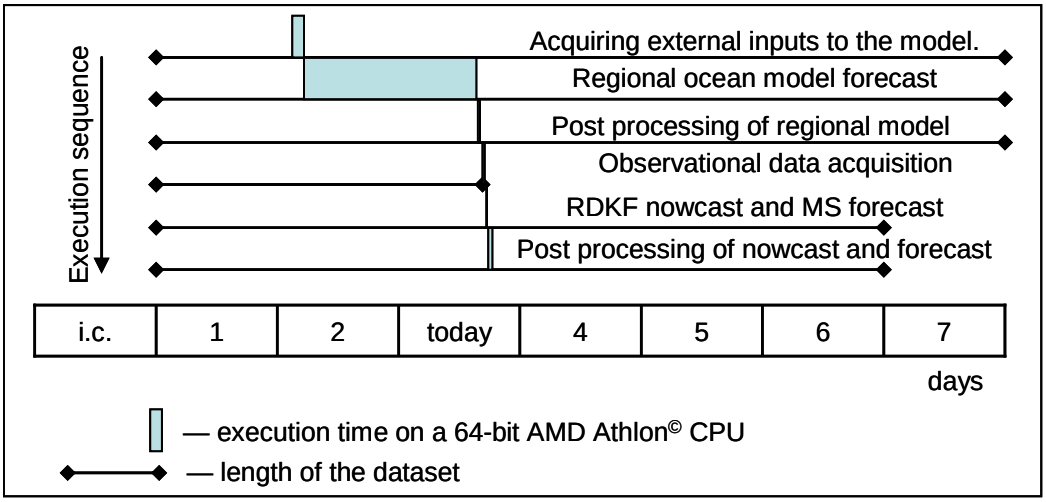


Figure 26: Timing diagram for the nowcast/forecast system.
MS—model surrogate. RDKF—reduced-dimension Kalman filter.

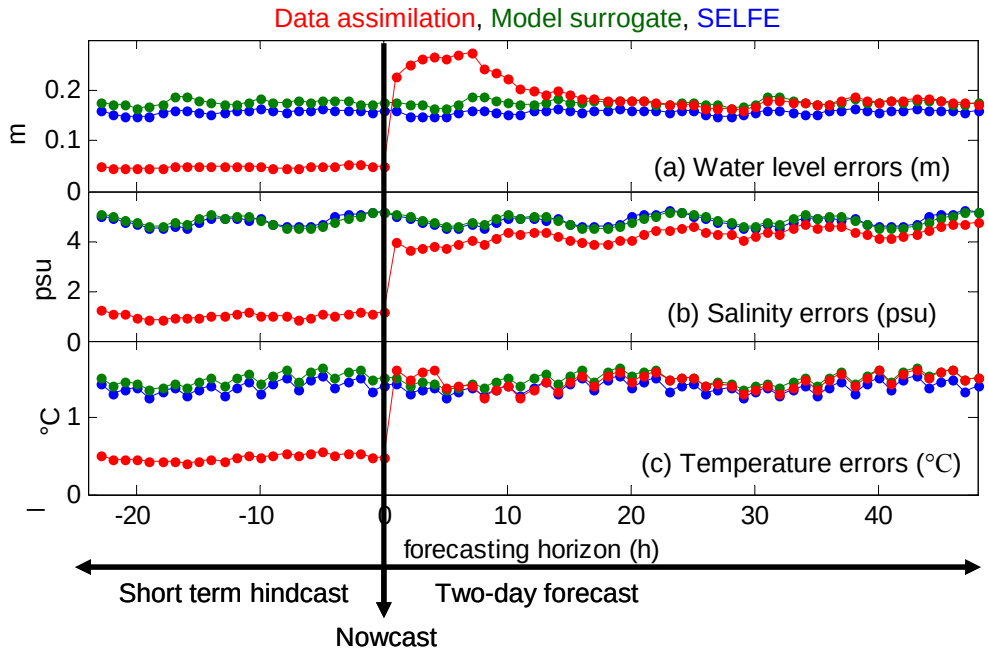


Figure 27: Root mean square error (RMSE) averaged over 365 hindcast-nowcast-forecast cycles. (a) RMSE for water levels, (b) RMSE for salinity, (c) RMSE for temperature.

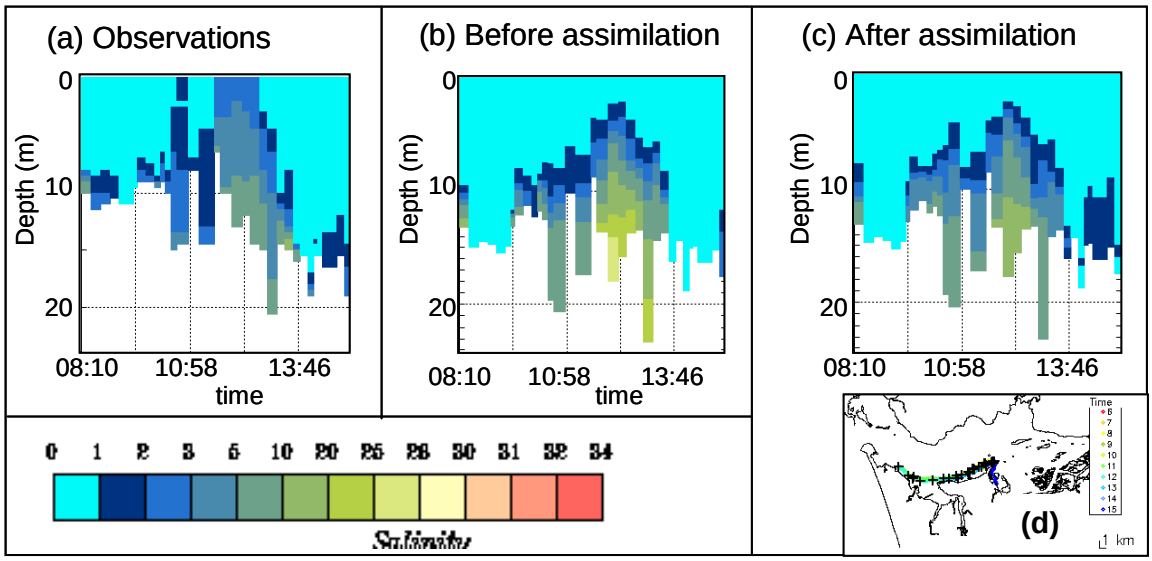


Figure 28: Profiles of observed (a) and modeled (b, c) salinity in the Columbia River estuary on June 5th.
(b) profiles before assimilation. (c) profiles after observations from fixed stations were assimilated. (d) map of the transect.

References

Baptista, A. M. (2006), "CORIE: The First Decade of a Coastal Margin Collaborative Observatory," in *Oceans'06 MTS/IEEE*, Boston.

Baptista, A. M., Zhang, Y.-L., Chawla, A., Zulauf, M. A., Seaton, C., Myers, E. P., Kindle, J., Wilkin, M., Burla, M., and Turner, P. J. (2005), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: Ii. Application to the Columbia River," *Continental Shelf Research*, 25, 935-972.

Bishop, C. (1995), *Neural Networks for Pattern Recognition*, Oxford University Press.

Blumberg, A. F., and Mellor, G. L. (1987), "A Description of a Three dimensional Coastal Ocean Circulation Model.," in *Three-Dimensional Coastal Ocean Models* (Vol. 4), ed. N. Heaps, Washington, DC, USA: AGU, pp. 1-16.

Foreman, M. G. G. (1977), "Manual for Tidal Heights Analysis and Prediction," vol. 58, Institute of Ocean Sciences.

Haidvogel, D. B., Arango, H. G., Hedstrom, K., Beckmann, A., Malanotte-Rizzoli, P., and Shchepetkin, A. F. (2000), "Model Evaluation Experiments in the North Atlantic Basin: Simulations in Nonlinear Terrain-Following Coordinates.," *Dynamics of Atmosphere and Oceans*, 32, 239–281.

Jay, D., and Smith, J. D. (1990), "Circulation, Density Distribution and Neap-Spring Transitions in the Columbia River Estuary," *Progress in Oceanography*, 25 (1-4), 81-112.

Jay, D. A., and Flinchem, E. P. (1997), "Interaction of Fluctuating River Flow with a Barotropic Tide: A Demonstration of Wavelet Tidal Analysis Methods," *Journal of Geophysical Research*, 102 (C3), 5705-5720.

Kindle, J. (2005), "Near Real-Time Depiction of the California Current System," in *AMS Conference on Coastal Atmospheric and Oceanic Prediction and Processes*, San Diego, CA.

Lermusiaux, P. F. J. (2007), "Adaptive Modeling, Adaptive Data Assimilation and Adaptive Sampling," *Physica D*, 230 (1-2), 172-196.

Lu, Z., Leen, T. K., van der Merwe, R., Frolov, S., and Baptista, A. M. (2007), "Sequential Data Assimilation with Sigma-Point Kalman Filter on Low-Dimensional Manifold," *submitted to Journal of Inverse Problems*, 30, also available at http://www.stccmop.org/files/CMOP-TR-07-001.pdf [Viewed on September 14, 2007].

Martin, P. J. (2000), *Description of the Navy Coastal Ocean Model Version 1.0.*, Naval Research Laboratory, Technical Report.

Ngodock, H., E., Jacobs, G., A., and Chen, M. (2006), "The Representer Method, the Ensemble Kalman Filter and the Ensemble Kalman Smoother: A Comparison Study Using a Nonlinear Reduced Gravity Ocean Model," *Ocean Modeling*, 12 (3-4), 378-400.

Robinson, A. R. (1999), "Forecasting and Simulating Coastal Ocean Processes and Variabilities with the Harvard Ocean Prediction System," in *Coastal Ocean Prediction*, ed. C.N.K. Moers, American Geophysical Union, pp. 77-100.

van der Merwe, R., Leen, T. K., Lu, Z., Frolov, S., and Baptista, A. M. (2007), "Fast Neural Network Surrogates for Very High Dimensional Physics-Based Models in Computational Oceanography," *Neural Networks*.

van der Merwe, R., and Wan, E. (2003), "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models," in *Workshop on Advances in Machine Learning*, Montreal, Canada.

Zhang, Y.-L., and Baptista, A. M. (submitted), "A Semi-Implicit Eulerian-Lagrangian Finite-Element Model for Cross-Scale Ocean Circulation," *Ocean Modeling*.

Chapter 5

Optimizing placement of fixed observational sensors in a coastal observatory

Sergey Frolov, António Baptista, Michael Wilkin

Target journals: Continental Shelf Research, Journal of Marine Science, JGR-oceans, Ocean Modeling

Abstract

Proliferation of coastal observatories necessitates an objective approach to managing of observational assets. In this article, we used our experience in the coastal observatory for the Columbia River estuary and plume to identify and address common problems in managing of fixed observational assets, such as salinity temperature and water level sensors attached to pilings and moorings. Specifically, we address the following problems: assessing the quality of an existing array, adding stations to an existing array, removing stations from an existing array, validating an array design, and targeting of an array toward data assimilation or monitoring.

Our analysis was based on a combination of methods from oceanographic and statistical literature, mainly on the statistical machinery of the best linear unbiased estimator. The key information required for our analysis was the covariance structure for a field of interest, which was computed from the output of assimilated and non-assimilated models of the Columbia River estuary and plume. The network optimization experiments in the Columbia River estuary and plume proved to be successful, largely withstanding the scrutiny of sensitivity and validation studies, and hence providing valuable insight into optimization and operation of the existing observational network. Our success in the CR estuary and plume suggest that algorithms for network optimization can play a significant role in the design of other ocean observatories, such as the Ocean Observation Initiative and the Integrated Ocean Observing System that are currently under development in the U.S.

5.1 Introduction

A series of reports identified integrated ocean observatories as the backbone of future scientific exploration and science-based management for coastal resources (National Research Council 2003, Pew Oceans Commission 2003). As a result, the U.S. is planning to spend upward of $500 million a year on developing two complementary ocean observatories: the Ocean Observation Initiative (OOI) (National Research Council 2003), with the goal to advance the scientific understanding of the ocean, and the Integrated Ocean Observing System (IOOS) (U.S. Commission on Ocean Policy 2004), with the goal to support scientifically informed management of the coastal ocean. Although the design-objectives of the two observatories are different, they both are expected to involve sensors deployed on fixed and mobile observational platforms, such as moorings, gliders, costal radars, and satellite sensors. Hence, many common challenges are expected in the design of these observatories. One such common challenge is finding such an optimal co-placement of fixed sensors that will maximize the amount of collected information, exploit the synergy with existing observational networks, and minimizing the development and operational costs of the observatory.

A possible strategy for the optimal sensor placement can be found using methods from the theory of optimal experiment design (Fedorov 1972, Silvey 1980, Pukelsheim 1993)—a subfield of formal statistics. In the past, many of these methods were applied to design observational arrays in oceanography (Bretherton, et al. 1976, Hackert, et al. 1998, Oke, et al. 2007) and meteorology (Berliner, et al. 1999, Bishop, et al. 2000). However, the application of these methods to design of coastal observatories is not common (She, et al. 2006). Recent advances in modeling of the coastal ocean circulation show great promise in providing information necessary for extending the application of optimal design methods to design of coastal observatories. In the research reported in this article, we studied how outputs of a circulation model and a data assimilation system can be used to find an optimal placement of sensors in a coastal observatory for the Columbia River (CR) estuary and plume.

The CR estuary and plume provide great examples of challenging dynamical environments, where an increasing number of observational and modeling assets are emerging, hence necessitating an objective approach for optimal placement of new and existing sensors. Recent observational assets in the CR estuary and plume include several multi-annual observational arrays, such as the CORIE observational arrays for the estuary and plume (Baptista 2006), the RISE observational array (Hickey 2004) for the CR plume, and an array of tide gages and atmospheric buoys operated by National Atmospheric and Oceanic Administration. Complementary to these observational assets, several realistic models were recently developed for the CR estuary and plume (Baptista, et al. 2005, MacCready, et al. 2007) and Chapter 2 of this thesis.

To understand how much of the CR estuary and plume variability is already known from the existing observation and modeling studies, and how much remains to be learned from deploying new observational assets, we sought answers to six specific questions regarding optimal placement of fixed observational assets in the CR estuary and plume:

1) How informative are the existing observational networks about the variability of salinity, temperature, and water levels in the CR estuary and plume?
2) How robust is the existing observational network in the CR estuary and plume to data dropouts?
3) Which sensors in the existing CR network collect redundant information and, hence, can be removed from the network?
4) Where in the CR estuary and plume should new sensors be placed?
5) How can the CR network be optimized for monitoring or for data assimilation?
6) How can the validity of proposed optimal sensor placements be validated?

Many of these questions are common to other observatories with fixed observational assets and, hence, can be extended for optimal placement of fixed observational assets in the national OOI and IOOS observatories.

To understand how to best place existing and future observational assets in the CR estuary and plume, we adopted a set of classical algorithms from the statistical and

oceanographic literature, including the best linear unbiased estimator (BLUE) (Ripley 1987), the exchange-type optimization algorithm (Fedorov 1994), and the array modes of Bennett (Bennett 1985, 1992). The BLUE estimator , our main analysis algorithm, uses the correlation length-scales for the field of interest (e.g. salinity in the CR estuary) to quantify how observing this field at locations of fixed stations reduces the uncertainty in the remaining, unobserved, locations in the CR estuary. To characterize these correlation length-scales in the CR, we used the temporally-averaged statistics about simulated fields (salinity, temperature, and water levels) and errors in these simulated fields, which were computed from non-assimilative (Baptista, et al. 2005, Baptista 2006) and assimilative (Chapters 2 and 3 of this thesis) models of the CR estuary and plume. To verify optimality of the proposed sensor-placements, we compared the predicted utility of the sensor-placement (predicted with the BLUE estimator) against the true utility (computed using a data assimilation experiment).

Our sensor-placement experiments in the CR estuary and plume proved to be successful, largely withstanding the scrutiny of sensitivity and validation studies and, hence, providing valuable insight into optimization and operation of the existing observational network. For example, we found that the existing water level network in the CR was very informative and robust, unlike the salinity network that was too sparse and too prone to data dropouts to reliably measure errors in the simulated salinity of the CR estuary. Using an optimization algorithm, we developed strategies for removing redundant stations and adding new, more informative stations in the CR estuary and plume. For example, the optimized array in the CR plume was predicted to be more informative than the historic array, even though the optimized array used only three moorings instead of five historic moorings.

5.2 Observatory for the CR estuary and plume

The test-bed for our network optimization experiments was CORIE—an observatory for the CR estuary and plume. The CR estuary, a classic river-dominated estuary, is a highly energetic and dynamic system that responds quickly to changes in ocean tides, regulated river discharge, and coastal winds. The tidal dynamic in the CR is significant (tidal amplitudes of up to 3.6 m) and is driven by the non-linear interaction of

astronomic tides with complicated bathymetry, non-stationary river discharge, and coastal wind (Jay, et al. 1997). Compressed and often stratified, the estuarine circulation in the CR is subject to extreme variations in salinity intrusion and stratification regimes (Jay, et al. 1990, Baptista, et al. 2005). The CR plume is a dominant hydrographic feature on the U.S. west coast that plays an important role in the transport of dissolved and particulate matter for hundreds of kilometers along and across the continental shelf (Barnes, et al. 1972, Grimes, et al. 1996). The far field of the CR plume has a predominant orientation towards north in fall and winter, and towards southwest in spring and summer (Hickey, et al. 1998). The near field of the CR plume has a shorter response time (hours to days) and can react quickly to changes in the local wind (Garcia-Berdeal, et al. 2002, Baptista, et al. 2005, Hickey, et al. 2005).

CORIE studies the dynamics of the CR estuary using a real-time observation network (Baptista 2006) and a modeling system for 3D baroclinic circulation of the CR estuary and the adjacent ocean (Baptista, et al. 2005). CORIE data and modeling products (Baptista, et al. 2005, Baptista 2006) are used to support the research and development of novel modeling techniques (Zhang, et al. 2004, Zhang, et al. submitted), fisheries research (Bottom, et al. 2005, Burla, et al. 2007, Burla, et al. submitted), and the CR ecosystem management (USACE 2001). At the time of our experiments, CORIE observatory had three major components:

1) A real-time observation system that included a network of fixed stations and research cruises. Fixed stations utilized in our experiments in sections 5.4 and 5.5 are shown in Figure 29. At each station, an array of sensors measured a variable combination of parameters, including water level, salinity, temperature, velocity, air temperature, and wind velocity. Stations *risen*, *risec*, and *rises* in Figure 29 were operated by the RISE project (Hickey 2004), and stations *tpoint* and *wauna* in Figure 29 were operated by National Atmospheric and Oceanic Administration.
2) A modeling system for simulating 3D baroclinic circulation of the CR estuary, plume, and adjacent coastal ocean. The model simulations used in our experiments were generated using an Eulerian-Lagrangian model SELFE (Zhang,

et al. submitted) and were forced with a combination of realistic atmospheric, ocean, and river forcings.

3) A data assimilation system (Chapter 2 of this thesis), which was based on a recently developed, reduced-dimension Kalman filter (Lu, et al. 2007)—a fast, nonlinear extension to the classical Kalman filter.

Results of recent hindcast simulations (Baptista, et al. 2005, Baptista 2006) showed good predictive skill in both the CR estuary and the plume, even under such challenging conditions as high river discharge and quickly changing winds. Specifically, water levels and salinities were represented robustly in the estuary across wide spatial (from channels to tidal flats) and time (from tidal to inter-annual) scales. In the plume, models showed reliable skill in representing fronts and direction of the plume (as compared to remote sensing and field data). Recent research on the application of data assimilation to the CR estuary and plume (Chapters 2 and 3 of this thesis) showed that data assimilation was able to further improve simulated water levels, salinity, and temperature in the CR estuary and plume.

5.3 Optimal experiment design methods

In the existing oceanographic and meteorological literature, the methods of optimal experiment design fall into two large categories: (1) methods based on a framework of statistical experiment design (Fedorov 1972, Silvey 1980, Pukelsheim 1993) and (2) methods based on a framework of adjoint sensitivity fields (Langland, et al. 1996, Baker, et al. 2000). The seminal article by Berliner (Berliner, et al. 1999) reconciled the two frameworks and identified some theoretical limitations to using the adjoint sensitivity fields for guiding placement of adaptive measurements. From the two methodological frameworks, we used the framework of statistical experiment design to find optimal placement of sensors in the CR estuary and plume, since the statistical framework is closely aligned with the framework of Kalman filter, the data assimilation scheme of choice in the CORIE observatory. An additional consideration for choosing the statistical framework was the ease with which it can be applied to other computational domains and variables, and the independence of the framework from many details of a forward model and a data assimilation system.

To implement the framework of statistical experiment design in the CR estuary and plume, we used the following methodological choices and approximations:

- To evaluate how the location of a potential measurement contributed to reducing the uncertainty about the field of interest (e.g. salinity or errors in salinity), we employed the statistical machinery of the BLUE estimator (Ripley 1987).
- To characterize the covariance structure for the field of interest, required by the BLUE estimator, we used a realistic, stationary, low-rank approximation to the covariance of state and error fields. These covariance approximations were computed based on the output of non-assimilative (Baptista, et al. 2005, Baptista 2006) and assimilative (Chapters 2 and 3 of this thesis) models of the CR estuary and plume.
- To compare alternative designs of observational arrays, we used two different cost functions: the trace (mean square error) and the determinant (volume) of the posterior covariance (Fedorov 1994).
- To find optimal array configuration, we used several modified versions of the exchange-type optimization algorithm (Fedorov 1994).
- To quantify the statistical redundancy of the existing observational array, we analyzed the eigen-spectrum of the observation covariance matrix, using analysis similar to the array modes of (Bennett 1985, 1992).

5.3.1 Best Linear Unbiased Estimator

To evaluate how the location of a potential measurement contributed to reducing the uncertainty about the field of interest (e.g. salinity or errors in simulated salinity), we employed the statistical machinery of the BLUE estimator (Ripley 1987). The BLUE estimator—also know as a Gauss-Markov theorem, kriging, and an objective analysis method—was introduced as early as 1963 (Gandin 1963) in meteorology and 1976 (Bretherton, et al. 1976) in oceanography. In the following description of the BLUE estimator, we adopted the exposition of the method from (Fedorov 1994).

Consider a discretized random field $x = x(\xi, k)$ where ξ is the spatial index on the computational grid and *k* is the index of the discrete time. The statistics of the field *x* are approximated using a stationary mean $\overline{x}(\xi) = E[x(\xi, k)]$ and a stationary covariance matrix $\mathbf{C}(\xi, \xi') = \mathrm{cov}(x(\xi), x(\xi'))$. In geosciences, we are often interested in predicting the field values:

$$x_p(k) = \mathbf{H}_p x(\xi, k) \qquad (25)$$

at spatial locations $\xi_p \in \xi$, defined by the selection operator $\mathbf{H_p}$, given sparse, noisy observations of the field:

$$y(k) = \mathbf{H}_y x(\xi, k) + \varepsilon \ , \qquad (26)$$

defined by the selection operator $\mathbf{H_y}$ and the Gaussian noise $\varepsilon \sim N(0, \sigma^2 \mathbf{I})$. For example in the CR estuary, we are interested in predicting the values of salinity inside of the estuary $x_p(k)$, given sparse *in situ* measurements of salinity $y(k)$.

The solution to this prediction (interpolation) problem Eq. (25) is the well-known BLUE estimator, which is defined as:

$$x_p(k) = \overline{x}_p + \mathbf{C}_{py}\mathbf{C}_{yy}^{-1}(y(k) - H\overline{x}) \qquad (27)$$

$$\mathbf{D}_{pp} = \mathbf{C}_{pp} - \mathbf{C}_{py}\mathbf{C}_{yy}^{-1}\mathbf{C}_{yp} \qquad (28)$$

where $\overline{x}$ is the mean state, $\mathbf{C}_{pp}$ is the prior covariance, and $\mathbf{D}_{pp}$ is the posterior covariance. The cross covariance $\mathbf{C}_{py}$ and the observations covariance $\mathbf{C}_{yy}$ in (Eqs. 27-28) are defined as:

$$\begin{aligned} \mathbf{C}_{py} &= \mathbf{H}_p \mathbf{C} \mathbf{H}_y^T \\ \mathbf{C}_{yy} &= (\mathbf{H}_y \mathbf{C} \mathbf{H}_y^T + \sigma^2 \mathbf{I}) \end{aligned} \ . \qquad (29)$$

The diagonal elements of the prior and posterior covariance matrices $\mathbf{C}_{pp}(\xi, \xi)$ and $\mathbf{D}_{pp}(\xi, \xi)$ in Eq. (28) are the prior and the posterior variance of the field x_p, which characterize our uncertainty about the field values before and after the measurements are taken, e.g., the lower variance indicates a higher certainty about the field. The off-diagonal elements of the covariance matrices $\mathbf{C}_{pp}(\xi, \xi')$ and $\mathbf{D}_{pp}(\xi, \xi')$ indicate how strongly the field values are correlated for any two spatial locations ξ and ξ'. For example, the stronger the cross correlations, the fewer measurements will be required to achieve small posterior uncertainty $\mathbf{D}_{pp}$.

Eq. (28) of the BLUE estimator shows that additional information from observations reduces the posterior uncertainty $\mathbf{D}_{pp}$, as compared to the initial (prior) uncertainty $\mathbf{C}_{pp}$. Eqs. (4 and 29) also show that, for linear selection operators $\mathbf{H}_{\mathbf{y}}$ and $\mathbf{H}_{\mathbf{p}}$, the posterior uncertainty $\mathbf{D}_{pp}$ only depends on the locations of observation and prediction points and does not depend on the value of the field at these locations.

5.3.2 Optimization criteria

In optimal experiment design, we are interested in selecting, based on some criterion of optimality, the best observation strategy from a set of all possible strategies. One such criterion of optimality is the size of the posterior covariance $\mathbf{D}_{pp}$ (Eq. 28), i.e., the smaller posterior covariance indicates a higher certainty about the field. However, the set of covariance matrices does not possess a natural order; hence, a cost function $J(\mathbf{D}_{pp})$ needs to be defined in order to map a set of posterior covariance matrices to the ordered set of real numbers. In our experiments, we used two such cost functions: the minimum mean square error (MSE) and the minimum determinant (DET) of the posterior covariance $\mathbf{D}_{pp}$.

The first cost function that we used was the MSE, which for a fixed number of predictions points x_p is measured by the trace of the posterior covariance $\mathbf{D}_{\mathbf{pp}}$:

$$J(\mathbf{D}_{pp}) = \text{trace}(\mathbf{D}_{pp}) = \sum \lambda_i \,, \qquad (30)$$

where $\text{trace}(.)$ is the trace of a matrix, and λ_i is the i^{th} eigenvalue of the covariance matrix $\mathbf{D}_{\mathbf{pp}}$. The trace of the matrix $\mathbf{D}_{\mathbf{pp}}$ is defined as a summation of the diagonal elements of the matrix $\mathbf{D}_{\mathbf{pp}}$, which is equivalent to the sum of the posterior variances (posterior uncertainties) at all prediction points x_p. Eq. (30) shows that minimizing for the trace (MSE) criterion is the same as minimizing for the smallest average eigen-value of the matrix $\mathbf{D}_{\mathbf{pp}}$. The advantage of using the MSE criterion is that it is the same criterion as used by the Kalman filter and the variational data assimilation (Bennett 2002). The drawback of the MSE criterion is the high computational cost of computing Eq. (30), since it is computed in the space of prediction points x_p, which is usually larger than the space of observations y.

The second cost function that we used was the determinant (DET) of the posterior covariance:

$$J(\mathbf{D}_{pp}) = \det(\mathbf{D}_{pp}) = \prod \lambda_i \,, \qquad (31)$$

where $\det(.)$ is the determinant of the matrix and λ_i is the i^{th} eigenvalue of the covariance matrix $\mathbf{D_{pp}}$. The DET criterion is equivalent to the product of the eigen values, hence minimizing for the DET criterion is equivalent to minimizing for the smallest volume of the covariance $\mathbf{D_{pp}}$. The advantage of using the DET criterion is due to computational efficiency that comes from evaluating Eq. (31) in the space of observations y. For example, Fedorov (Fedorov 1994) showed that the minimum of $\det(\mathbf{D}_{pp})$ is equivalent to the maximum of $\det(\mathbf{C}_{yy})$. That is, the minimum volume of $\mathbf{D}_{\mathrm{pp}}$ (the posterior uncertainty) is achieved at the maximum volume of $\mathbf{C}_{\mathrm{yy}}$ (the observation covariance). Computing $\det(\mathbf{C}_{yy})$ is usually computationally more efficient than computing $\det(\mathbf{D}_{pp})$.

5.3.3 Optimization algorithm

To find the minimum of the cost function in Eqs. (30 and 31), we defined the following optimization problem:

$$\mathbf{Y}_o = \arg\min_{\mathbf{Y}_o \in \mathbf{Y}_f} \left[J(\mathbf{D}_{pp}) \right], \qquad (32)$$

where $\mathbf{Y}_f = \{y_1, ..., y_n\}$ is the set of all possible observations and $\mathbf{Y}_o \in \mathbf{Y}_f$ is the set of optimal observations that minimizes cost function $J(\mathbf{D}_{pp})$. To search for the minimum of the optimization problem in Eq. (32), we used several variants of the exchange-type algorithm. The basic exchange-type algorithm, as described in (Fedorov 1994), iterates over the following two steps:

1) Delete observation y_o^-, chosen from the set of set of candidate stations, that contributes the least to minimizing the optimization criteria $J(\mathbf{D}_{pp})$:

$$y_o^- = \arg\min_{y_o^- \in \mathbf{Y}_o} \left[J(\mathbf{D}_{pp}) \right] \qquad (33).$$

2) Add observation y_l^+, chosen from the set of all feasible observations, that contributes the most to minimizing the optimization criteria $J(\mathbf{D}_{pp})$:

$$y_o^+ = \arg\min_{y_o^+ \in \{\mathbf{Y}_f - \mathbf{Y}_o\}} \left[J(\mathbf{D}_{pp}) \right] \qquad (34).$$

Iterating over the delete and add steps continues until $J(\mathbf{D}_{pp})$ decreases. After each successful iteration, the active set of optimal observations is updated as following: $\mathbf{Y}_o = \mathbf{Y}_o - y_o^- + y_o^+$.

In practice, we implemented the delete and the add steps (Eqs. 33-34) differently for the MSE and the DET criteria. For the MSE criteria, we implemented the optimization algorithm as a direct search through the set of all active observations $\mathbf{Y}_o$ (delete step) and through the set of all feasible observations $\mathbf{Y}_f - \mathbf{Y}_o$ (add step). For the DET criteria, we used a computationally more efficient algorithm (Fedorov 1994) that operates in a smaller space of observations *y*. The modified algorithm uses the following delete and add steps to find a minimum of the $J = \det(\mathbf{D}_{pp})$:

Delete: $y_o^- = \arg\max_{y_o^- \in \mathbf{Y}_o} \left[c_{yy}^{-1} \right]$ (35)

Add: $y_o^+ = \arg\min_{y_o^+ \in \{\mathbf{Y}_f - \mathbf{Y}_o\}} \left[c_{yy}^{-1} \right]$ (36)

where c_{yy}^{-1} are the diagonal elements of matrix $\mathbf{C}_{yy}^{-1}$, and y_o^- and y_o^+ are the observations that correspond to the largest (y_o^-) and the smallest (y_o^+) value in the vector c_{yy}^{-1}.

Although the exchange-type algorithm (Eqs. 33-34) is guaranteed to converge to a fixed point, this fixed point is not guaranteed to be the global minimum of the optimization problem (Eq. 32) (Fedorov 1994). To limit the likelihood of convergence to a suboptimal local minimum, we restarted the optimization algorithm using multiple random configurations of the initial observation network. Alternatively, we often reduced the exchange-type algorithm to the pure add- or delete-algorithm, both of which have a unique, but suboptimal minimum. In section 5.1, we compare the optimization results using all three modifications of the exchange algorithm.

5.3.4 Choices of the prior covariance

To define our prior knowledge about the field of interest, we used two distinct choices of the prior covariance: the state and the error covariance. The state covariance characterizes the uncertainty about the current state of the system with respect to the mean state. The error covariance characterizes the uncertainty about the accuracy of the model prediction. The observation locations optimized using the state covariance are optimal for reconstructing the state of the system without the aid of a numerical model; that is for monitoring of the system state. The observation locations that are optimized using the error covariance are optimal for reconstructing the error field of the model; that is, for improving the model simulation through a data assimilation procedure.

The state covariance: To characterize the covariance structure for the state variables, such as salinity and temperature in the CR estuary, we used an empirical covariance:

$$\mathbf{C}^{state} = E\left[(x-\overline{x})(x-\overline{x})^T\right], \qquad (37)$$

where the samples of states x were approximated by model states extracted from a long hindcast database. The primary choice for computing the state covariance matrix $\mathbf{C}_{pp}^{state}$ was the CORIE hindcast database DB14 that included both the CR estuary and plume and that did not use data assimilation. For sensitivity experiments in section 5.5.1, we computed the state covariance based on two additional datasets: the non-assimilated DB16 and the assimilated DB14. The design of DB16 was similar to DB14; however, the numerical grid of DB16 had a higher resolution and was restricted to the CR estuary. The data assimilation method was described in Chapter 2 of this thesis.

The error covariance: To characterize the covariance structure of model errors, such as errors in simulated salinity and temperature, we used a scaled time-average of the forecast error covariance matrices:

$$\mathbf{C}^{error} = \alpha \frac{1}{n}\sum_{k=1}^{n} \mathbf{P}_{xx}^{-}(k) \qquad (38)$$

where $\mathbf{P}_{xx}^{-}(k)$ is a forecast error covariance computed by the data assimilation system described in Chapter 2 of this thesis, n is the number of averaged time steps, and α is the

inflation factor that accounts for a possible inconsistency between the predicted and the true size of the forecast error covariance. The inflation factor α, which establishes a correct balance between the model error and the observation error in the Eq. (29), was computed using the following formula:

$$\alpha = \left[trace(\mathbf{C}_{yy}^{observed}) - trace(\mathbf{R}) \right] \Big/ trace(\mathbf{C}_{yy}^{predicted})$$
$$\mathbf{C}_{yy}^{predicted} = \mathbf{H}_{\mathbf{y}} \left[\frac{1}{n} \sum_{k=1}^{n} \mathbf{P}_{xx}^{-}(k) \right] \mathbf{H}_{\mathbf{y}}^{T} \quad , (39)$$

where $\mathbf{C}_{yy}^{observed}$ and $\mathbf{C}_{yy}^{predicted}$ are the observed and the predicted error covariance at the locations of existing observation stations, and **R** is the covariance of observation errors.

Since the size of the state and error covariance (Eqs. 37-38) can be quite large (~10^6x10^6), we operated with a factorized form of these covariance matrixes:

$$\mathbf{C}_{pp} = \mathbf{H}_{\mathbf{p}} \mathbf{C} \mathbf{H}_{\mathbf{p}}^{T} = \mathbf{H}_{\mathbf{p}} \mathbf{L} \mathbf{S} \mathbf{L}^{T} \mathbf{H}_{\mathbf{p}}^{T}, \quad (40)$$

where $\mathbf{H_p}$ is the selection operator for the prediction points, **L** is the matrix of leading eigen-vectors for a covariance matrix **C**, and **S** is the diagonal matrix of the eigen-values. The eigen-spectrum **S** was truncated to improve computational performance. Typically, we retained more than ~95% of the variance, with the exact truncation value determined through a series of sensitivity studies (not presented here). The efficient implementation of the BLUE formulas (Eq. 27) with the factorized covariance matrix (Eq. 40) is detailed in Appendix 1.

5.3.5 Advantages and limitations of the proposed method

There are several advantages to using the optimal experiment design method described in sections 5.3.1-5.3.4. The method is generic, e.g., we used it for placing salinity sensors in the CR estuary and plume, but similar methods were also used for finding optimal sensor-locations in many open ocean (Bretherton, et al. 1976, Hackert, et al. 1998, Oke, et al. 2007) and meteorological (Berliner, et al. 1999, Bishop, et al. 2000) applications. The method does not require development of an adjoint model; in fact, it can be used without any model input, in cases were high resolution observational data are available. The method is closely related to the theory of data assimilation, since the MSE

cost function (Eq. 30) that we used in our experiments is the same cost function as used by most sequential and variational data assimilation algorithms (Bennett 2002).

The two main limitations of the method stem from the assumption of stationary statistics and from the choice of the optimization criteria. It is likely that the stationary (averaged in time) covariance matrix is well-suited for optimizing fixed observational assets, since averaging in time is likely to provide an adequate proxy for large-scale correlations of persistent features that can be monitored well with a fixed observational array. However, it is also likely that the stationary covariance matrix may not suffice for adaptive sampling of a fast-moving feature like a plume front. In the case of adaptive measurements, the fine-scale correlations associated with fronts are likely to be non-stationary and will require an accurate prediction of their structure, for example using a data assimilation system. For more references on adaptive measurements see (Berliner, et al. 1999, Daescu, et al. 2003, Ogren, et al. 2004, Bertozzi, et al. 2005, Leonard, et al. 2006).

The second limitation is rooted in the optimization criteria. Both the MSE and the DET criteria are designed to find measurements that are optimal for state estimation; that is, for reconstructing the field given sparse measurements. However, the observations can be targeted for applications other than the state estimation, for example for detecting extreme events or for tracking the location of a maximum concentration. In these cases, observations optimized using the MSE and the DET criteria may not be optimal, and using alternative optimization criteria can be prudent (Fedorov 1994, Berliner, et al. 1999, Chang, et al. 1999).

5.4 Design of the observational array in the CR estuary and plume

To design an optimal array for the CR estuary and plume, we applied optimal experiment design methods from section 5.3, to answer three questions regarding the optimality of the existing observational array in the CR estuary and plume and regarding the optimality of future sensor-placements:

1) How informative is the existing CORIE array, described in section 5.2?

2) Which CORIE stations provided redundant information, and hence, can be removed from the existing array without a significant loss in the collected information?
3) Where in the CR estuary and plume should new salinity sensors be placed?

In most of our experiments, we compared the array designs based on two choices of the covariance matrix: the state and the error covariance, hence optimizing sensor placements towards monitoring of the CR estuary and plume or towards data assimilation. The suggested array designs were validated using data assimilation experiments, presented in section 5.5.

5.4.1 Quality of the existing array

To characterize the quality of the existing observational array in the CR estuary, we estimated the information content and the redundancy of the existing array. The information content was estimated using the R^2 statistic and the redundancy, using the eigen-spectrum of the observation covariance matrix. From our experiments, we found that the existing array was too sparse to adequately represent the statistics of salinity errors (R^2 of ~62%). The array was also redundant; however, this redundancy was not sufficient to account for gaps in observations of salinity that occurred due to bio-fouling of salinity sensors. The details of the experiments are presented below.

Information content: To determine whether the existing array was representative of state and error variability in the CR estuary, we computed the R^2 statistic using the following formula:

$$R^2 = 100 * \left(1 - \frac{trace(\mathbf{D}_{pp})}{trace(\mathbf{C}_{pp})} \right), \qquad (41)$$

where $\mathbf{C_{pp}}$ and $\mathbf{D_{pp}}$ are the prior and the posterior covariance matrixes, and *trace(.)* is the trace of the matrix. The R^2 statistic measures the percentage of the prior variance (uncertainty), explained by the observations. For example, an R^2 of 0% indicates uninformative measurements that cannot reduce the prior uncertainty about the field of interest. An R^2 of 100% indicates highly informative measurements that can be used to

reconstruct the field of interest in its entirety, short of inaccuracies in the prior covariance $\mathbf{C_{pp}}$.

From our experiments in the CR estuary, we found that the salinity, temperature, and water level arrays were highly informative of the state variability (R^2 of >90%). However, the arrays were less informative of the error variability in the estuary; R^2 was as low as ~62% for salinity errors, and between 80% and 90% for temperature and water level errors. The results of the experiments are presented in Figure 30, where the R^2 values for the state (panel 1) and the error (panel 2) covariance are plotted against the number of sensors retained in the array. Sensors in Figure 30 are ordered from the most important (first sensor) to the least important (last sensor). The order of sensors was determined using the sensor-reduction sequences discussed in section 5.4.2. Results in Figure 30 suggest that investing in additional salinity sensors in the CR estuary is likely to yield the most informative array, since the current array can explain only ~62% of the salinity error variance, but more than 80% of the temperature and water level error variance.

Redundancy: To determine whether the existing salinity array in the CR estuary provided redundant information, we analyzed the eigen-spectrum of the observation covariance matrix[22] $\mathbf{C}_{yy}$ (Eq. 29), using the analysis method similar to the array modes of Bennett (Bennett 1985, 1992). From our experiments, we found that the existing salinity array was slightly redundant and could be further optimized at the cost of decreased robustness to data dropouts. For example, Figure 31 shows the truncation error e_i (Eq. 42) plotted against the number of retained eigen-values λ_k. From a total of 14 salinity sensors, 99% of the variance was captured by 8 modes for the state covariance and by 10 modes for the error covariance. The comparison of the truncation errors between the state and the error covariance in Figure 31 shows that more sensors were required for data assimilation than for monitoring of the CR estuary, suggesting that the error covariance had shorter correlations scales than the state covariance.

[22] The state covariance was computed based on observations of salinity. The error covariance was computed based on model-data misfits for DB14.

The truncation error e_i in Figure 31 was computed using the following formula:

$$e_i = 1 - \sum_{k=1}^{i} \lambda_k \Big/ \sum_{k=1}^{k_{\max}} \lambda_k \,, \qquad (42)$$

where λ_k is the k^{th} eigen-value of the observation covariance matrix $\mathbf{C_{yy}}$, k_{max} is the number of all eigen-values, and i is the number of retained eigen-values.

Robustness of the CORIE array to data dropouts: To study the robustness of the existing CORIE array to data dropouts, we plotted in Figure 32 the time series of the R^2 values (Eq. 41) for salinity, temperature, and water level arrays. Figure 32.a shows the number of active sensors in May-October 2004. Figure 32.b shows how the change in the number of sensors affected the R^2 statistic, which measured the ability of the CORIE array to represent salinity, temperature, and water level errors in the CR estuary. The results in Figure 32 show that the information content of the salinity array was most sensitive to data dropouts, with the R^2 values varying between 20% and 60%.

5.4.2 Removing redundant sensors from the network

To determine which salinity sensors in the CORIE network provided redundant information, and hence, can be removed from the existing array without a significant loss in the collected information, we applied the deleting procedure (Eq. 33) together with the MSE criterion (Eq. 30) to sequentially remove redundant sensors from the salinity array in the CR estuary. Table 2 presents the order in which the salinity sensors were removed and the corresponding R^2 values for the truncated arrays. In general, the suggested order in Table 2 was logical. For example, sensors that were removed first, such as *marsh*, *eliot,* and *cbnc3*, were located upriver and saw salt only occasionally. In contrast, sensors that were removed last, such as *sandi*, *dsdma*, and *jetta*, were located in the mouth of the CR estuary and saw large variation in salinity at each tidal cycle.

Analysis of R^2 values for the truncated arrays in Table 2 suggests that many sensors in the array were equally important. For example, it took removing four sensors in the error-based array (Table 2 columns IV and V) to change the value of R^2 from 62% to 61%, which suggests that these four sensors had equally small effect on the value of

R^2, and hence, the order in which this sensors were removed can be easily altered. To determine whether there was a single most important or least important salinity sensor in the CR estuary, we conducted two additional experiments described below.

Most important sensor: To determine which single salinity sensor was most important, we computed R^2 (Eq. 41) values for arrays that had only one active salinity sensor in the CR estuary. Based on our experiments, shown in Table 3 columns II and III, we could not locate a single most important salinity sensor. Instead, we found that most salinity sensors were equally important in the CR estuary, since each had similar value of R^2. For example, there were at least seven sensors, each of which was able to explain between 70% and 80% of the state variability. We also found that a single sensor can explain a much higher percentage of the state variance (as much as 79%) than the error variance (no more than 14%).

Least important sensor: To determine which single salinity sensor was least important, we computed the R^2 (Eq. 41) values for arrays with one of the salinity sensors in Table 3 removed. The results of the experiment, columns IV and V, show that, while the impact of an individual sensor varied, removing a single sensor had very little impact on the R^2 of the remaining sensors. For example, removing a single salinity sensor led to a decrease in R^2 of no more than 1.1% for the state and 4.6% for the error variance.

5.4.3 Adding salinity sensors to the existing CORIE network

To find optimal placement of new salinity sensors in the CR estuary and plume, we reduced the exchange-type algorithm to several sequential add-steps (Eq. 34). We considered two realistic scenarios: adding one salinity sensor in the CR estuary and re-organizing the existing salinity sensors in the CR plume. In the estuary, our optimization procedure placed the new salinity sensor in the North Channel, leading to an improved R^2 for the salinity errors (from 62% to 69%). In the plume, the optimization procedure placed two sensors north and one station south of the CR mouth. The three optimally placed salinity sensor in the CR plume were more informative (R^2 of 23% for salinity errors) than the five historical sensor; the R^2 value for the historical plume sensors *risen,*

risec, rises, ogi01, and *ogi02*, shown in Figure 29, was 18%. The comparisons of the arrays optimized using the error and the state covariance showed that both arrays had similar configuration in the estuary and plume. The detailed description of the logistical constraints and the experimental results follow.

Logistical constraints on the placement of sensors: To characterize some of the logistical constraints that are likely to arise in the deployment of new salinity sensors in the CR estuary and plume, we added the following constraints to the optimization problem (Eq. 32). We constrained the locations of feasible salinity sensors to areas deeper than 10 m in the plume and deeper than 4 m in the estuary. We constrained the domain of interest, defined by the operator $\mathbf{H}_p$ in Eq. (25), to the CR plume for placing the plume sensors, and to the CR estuary for placing the estuarine sensors. We further constrained our search by deploying candidate sensors at the depth of the highest variability at each horizontal location, hence reducing the three-dimensional optimization problem to a two-dimensional one. The experiments in this section were for May-September 2004.

Adding salinity sensor in the CR estuary: The results of the array-optimization in the CR estuary are presented in Figure 33. The configurations for both the state (panels *s.1, s.2,* and *s.3*) and the error covariance (panels *e.1, e.2,* and *e.3*) are presented. The top and the middle panels depict the local uncertainty reduction R^2_{local} (from Eq. 43 below) before (panels *s.1* and *e.1*) and after (panels *s.2* and *e.2*) the new salinity sensor was added. The bottom panels (*s.3* and *e.3*) depict the map of the cost function that shows the added value of placing the next salinity sensor at one of the logistically feasible locations that satisfy logistical constraints described above. The added value was characterized by an increase in the global R^2 value (Eq. 41).

The local uncertainty reduction values R^2_{local} on panels *e.1-2* and *s.1-2* were computed for each spatial location *x(i)* using the following formula:

$$R^2_{local}(i) = 100 * \left(1 - \frac{\mathbf{D}_{pp}(i,i)}{\mathbf{C}_{pp}(i,i)} \right) \qquad (43)$$

where R^2_{local} is the local uncertainty reduction, and $\mathbf{C}_{pp}(i,i)$ and $\mathbf{D}_{pp}(i,i)$ are the prior and the posterior variances at the spatial location with the index *i*. Figure 33 plots the maximum over depth R^2_{local}, hence showing the maximum extent of horizontal correlations between the observation sensors and the rest of the estuary. The correlation length scales in the CR estuary, shown in Figure 33, were spatially dependant, with longer correlation length scales along the channels of the estuary.

Results in Figure 33 show that both the state- and the error-based configurations agreed on placing the next salinity sensor in the CR estuary close to the river end of the North Channel. Both optimal locations were in the shallower water (4.9 m depth), ~300 m away from each other. The maps of the cost function in panels *s.3* and *e.3* showed that most of the location in the North Channel of the CR estuary were equally beneficial, hence provided additional information for the operational staff of the CORIE observatory. Placing the next salinity sensor in the North Channel of the CR estuary contributed slightly to uncertainty reduction for the state-based array, improving R^2 from 96% to 97%. However, the new salinity sensor had a higher impact on the error-based array, improving R^2 from 62% to 69%. Although these increases in the average R^2 were modest, the local R^2 in the North Channel of the estuary increased substantially, from ~50% to >80% in Figure 33, panel e.2.

Adding salinity sensors in the CR plume: The results of the array-optimization in the CR plume are presented in Figure 34 for the state covariance and in Figure 35 for the error covariance. The panels in a top row on both figures display the prior salinity variance (panel 1) and the local uncertainty reduction R^2_{local} (panel 2) before any salinity sensors were added in the CR plume. The panels below the top row display the added value of placing new salinity sensors in the CR plume. Panel 3 depicts the map of the cost function that shows the added value of placing the next salinity sensor at one of the feasible sensor-locations in the CR plume. The added value was characterized by the

increase in the global R^2 value (Eq. 41). Panel 4 shows the increase in local uncertainty reduction R^2_{local} (Eq. 43) after the new sensor was added in the CR plume. Panels 3 and 4 show three consecutive iterations of the algorithm, each sequentially adding another new salinity sensor in the CR plume.

Comparison of the optimal designs in Figures 34 and 35 shows a remarkable agreement between the state- and the error-based design. Both designs place two salinity sensors north and one salinity sensor south of the CR mouth. These configurations contrast the historic array designs in the CR plume that favored sampling in southern and central plume; for an example of historical array, see three RISE and two OGI sensor on Figure 29.2. A possible explanation for this difference is that historic arrays were biased towards sampling of the plume's southern branch—the dominant orientation of the CR plume in summer. In contrast to the historic designs, the proposed optimal designs in Figures 34 and 35 stress the importance of placing multiple sensors in the northern plume in order to determine the extent of the plume's northern propagation.

To compare the optimality of the proposed salinity-array designs in the CR plume with that of the historic arrays, we list the computed R^2 values (Eq. 41) for the historic and the proposed arrays in Table 4. Table 4 shows that the proposed arrays had consistently higher R^2 values than the historic arrays with comparable number of moorings. For example, the error-based array with three optimally placed moorings had higher R^2 value (23%) than the RISE array with three moorings (15%) and the combined RISE+OGI array with five moorings (18.7%).

As we mentioned earlier, it is likely that the *ogi01* and *ogi02* sensors will be redeployed in the CR plume. An optimal design shown in Figures 34 and 35 places both moorings north of the CR mouth. However, given only two moorings, it may seem beneficial to deploy one mooring north and one south of the CR mouth. It turns out that the modified design, with one mooring north and one south, is only marginally suboptimal in comparison to the suggested optimal design in Figures 34 and 35. For example, the map of the cost function (Figures 34 and 35, panel 3, iteration 2) shows that

the second location in the northern plume is only marginally better than the location in the southern plume: R^2 of 17.9% vs. 16.6% for the error array and 71.0% vs. 71.2% for the state array. Hence, we suggest that it is best to redeploy OGI moorings north and south of the CR mouth, in contrast to the existing, less informative configuration of the OGI array, displayed in Figure 29.2.

5.5 Verification of the proposed array designs

To verify the optimality of the arrays designed for CR estuary and plume in section 5.4, we answered the following four questions about the validity of our array-design methods:

1) **What were the theoretical limitations of our design algorithm?** In section 5.3.5, we identify the limitations of our design algorithm, which suggested that the arrays that we designed using the MSE criterion are likely to be optimal for data assimilation and for monitoring the variability of the CR estuary and plume. However, these arrays may not be optimal for detecting occurrence of extreme events, such as an event of unusually high salinity intrusion.
2) **Was the model that we used to compute the prior statistics representative of the physical variability in the CR estuary and plume?** From the past studies (Baptista, et al. 2005, Baptista 2006) and Chapters 2 and 3 of this thesis, we know that, at the time of our experiments, assimilated and non-assimilated models were capable of realistically representing variability of salinity, temperature, and water levels in the CR estuary and plume. To determine how differences among these models influenced the optimal placement of sensors, we conducted a sensitivity study described in section 5.5.1, which used four different datasets and three different lengths of the dataset.
3) **How sensitive was the optimal sensor placement to changes in the optimization criteria and the optimization algorithm?** To determine the sensitivity of optimal sensor placement, we conducted two studies in section 5.5.1 that examined the sensitivity to the choice of the optimization criteria, MSE vs. DET, and to the choice of the optimization algorithm, full exchange-type algorithm vs. add- and delete-only algorithms.

4) **How accurately did the BLUE estimator predict the utility of alternative sensor placements?** In section 5.5.2, we present the results of the cross-validation study, where we used data from existing salinity sensors to verify the predicted utility of several alternative designs. We verified the BLUE predictions of R^2 values using a data assimilated experiment that computed the true value of error reduction at the locations of validation sensors.

5.5.1 Sensitivity studies

To determine whether the proposed salinity arrays in section 5.4.3 were sensitive to changes in experimental procedures, we studied the sensitivity of these optimal arrays to changes in the optimization criteria, the optimization algorithm, and the length and the type of the dataset that was used for computing of the prior covariance. From our experiments, we found that the design of these salinity arrays was strongly sensitive to the choice of the optimization criteria, but was not very sensitive to the choice of the optimization algorithm or to the choice of the dataset. To select between the arrays designed using the MSE and the DET criteria, we used our scientific judgment and selected array configurations optimized using the MSE criterion, since the MSE is also the criterion used by data assimilation.

Sensitivity to optimization criteria: To determine how the choice of the optimization criteria influences the optimal placement of salinity sensors in the CR estuary and plume, we reproduced sensor-placement experiments from section 5.4.3 using two different optimization criteria: the MSE criterion (Eq. 30) and the DET criterion (Eq. 31). The reproduced experiments included adding new salinity sensor in the CR estuary, results shown in Figure 36.1, and adding three new sensors In the CR plume, results shown in Figure 36.2. Sensor locations optimized using the MSE criterion, marked with “o” in Figure 36, differed from sensor locations optimized using the DET criterion, marked with “+”. For example in the plume (Figure 36.2), the MSE criterion distributed the salinity sensor throughout the CR plume, while the DET criterion clustered the salinity sensors in the high-variance area, located close to the CR mouth. We speculate, based on the limited results from Figure 36 and more extensive

experiments not presented here, that locations selected by the DET criteria tend to cluster in the region of higher variance, while the MSE criteria tend to yield more distributed observation networks.

Sensitivity to optimization algorithm: To determine how sensitive the optimized locations of salinity sensors were to changes in the optimization algorithm, we compared optimization results using the full exchange-type algorithm, described in section 5.3.3, with the results from the modified add-only algorithm, used in sections 5.4.2, and the delete-only algorithm, used in section 5.4.3. The sensitivity experiment compared the R^2 values from 1000 random initializations of the full exchange-type algorithm (displayed as frequency bars in Figure 37) with the deterministic results of the modified algorithms (marked with vertical lines in Figure 37). To enable this computationally expensive experiment, we considered a small optimization problem, where the vertical locations of five salinity sensors at station *red26* were optimized to fully represent the vertical variability of the salinity error at the location of *red26*.

Experimental results in Figure 37 show that the exchange-type algorithm produced array designs with a range of R^2 values (from 91.5% to 92.4%), which in several occasions were somewhat higher than the R^2 values computed by the deterministic add-only (R^2=92.1%) and delete-only (R^2=92.3%) algorithms. All three optimization algorithms produced arrays with the R^2 higher than an average randomly initialized array (R^2=89%). It is very likely that some of the arrays optimized using the exchange-type algorithm were at or close to the global optimum. These experimental results suggest that multiple initializations of the full exchange-type algorithm are likely to find better array designs. However, the add-only and the delete-only algorithms are also likely to find good approximations to the globally optimal array design.

Sensitivity to the length and the type of the dataset: To determine how sensitive the location of the next salinity station in the CR estuary was to the type of the dataset that was used in computing of the prior covariance $\mathbf{C_{pp}}$, we repeated the experiment from section 5.4.3 using four different datasets: assimilated DB14, non-

assimilated DB14, non assimilated DB16, and a scaled time-average of prior error $\mathbf{C}_{\mathbf{error}}$ (Eq. 38). To determine the sensitivity to the length of the dataset, we repeated the same experiment using the following three lengths of the dataset: 1/2 year, 1 year, and 6 years. We found that the placement of the salinity sensor in the CR estuary was not very sensitive to the type or to the length of the dataset. At most, the locations of the new salinity sensor were ~1.5 km apart.

5.5.2 Validation experiments

To determine whether the BLUE estimator accurately predicted the utility of alternative observation strategies in the CR estuary, we used a validation procedure that leveraged existing observations of salinity in the CR estuary. From our experiments, we found that the BLUE estimator over predicted the R^2 values by 10-20%. However, the BLUE estimator correctly predicted relative importance of alternative salinity sensor placements, suggesting that the optimal designs of salinity arrays in section 5.4 were likely correct, even though the BLUE estimates of their utility, the R^2 values, were likely too optimistic. The details of the validation strategy and the detailed analysis of the experimental results follow.

We used the following algorithm to validate the predicted utility of a candidate salinity array using a data assimilation experiment:

1) We chose a validation station where the error reductions were computed using available observational data. We used two such stations—*red26* and *am169*—each of which had three vertically spaced salinity sensors.
2) We predicted, using the BLUE estimator (Eqs. 27-28), how assimilating data from the existing salinity sensors can reduce salinity errors at the location of the validation station. To predict the performance of a data assimilation system, our BLUE estimator used the MSE criterion, the salinity error covariance matrix (Eq. 38), and a two-week-long period in September of 2004. We also considered several alternative configurations of the observational network. In each network, one of the existing salinity sensors was removed from the assimilation.

3) We verified the BLUE predictions of salinity error reductions, computed in item 2 above, by assimilating salinity data from candidate arrays using the data assimilation method described in Chapter 2 of this thesis.

The results of the experiments for the salinity validation stations *red26* and *am169* are presented in Tables 5 and 6. The alternative arrays are ranked based on their predicted R^2 values. The arrays with lower R^2 values indicate salinity sensors with highly valuable observations, since removing them from the assimilation reduced the data assimilation accuracy at validation station significantly.

The results in Tables 5 and 6 show that the BLUE estimator consistently over predicted the R^2 values at both validation stations, by about 10-20%. For example, the BLUE estimator predicted that using all sensors (last line in Tables 5 and 6) the errors in models salinity will be reduced at the validation station *red26* by 79% and for *am169* by 51%; however, the data assimilation experiment showed that actual errors were reduced only by 51% and 41% respectively. However, the results in Tables 5 and 6 also show that the predicted ranking of the alternative observation strategies was largely correct, suggesting that the salinity error covariance matrix used in the BLUE estimator captured the correlation scales in the CR estuary well. For example, the rankings agreed for all configurations at the validation station *red26* (Table 5) and for most configurations at the validation station *am169* (Table 6). The few candidate salinity arrays for which the rankings disagreed are marked in bold in Table 6.

5.6 Summary and discussion

Algorithms for evaluating and optimizing a fixed network of sensors were developed. The developed algorithms were based on the theory of optimal experiment design, were computationally inexpensive, and were application- and model-independent. The developed algorithms optimized placement of sensors for accurate reconstruction of the ocean state with or without the help of a data assimilation system. Alternative optimization goals, e.g., detection of extreme events, were not considered in this article and are subject of future research.

In applying the developed algorithms for optimal placement of sensors to the coastal observatory for the CR estuary and plume, we determined that:

1) The existing salinity array in the CR estuary was too sparse to adequately represent the statistics of the salinity errors (R^2 of 62%);
2) The existing salinity, temperature, and water level arrays were also redundant. For example, 8 out of 14 eigen-modes of the salinity array captured 99% of the measured variance. However, this redundancy was not sufficient to account for data dropouts due to bio-fouling of salinity sensors;
3) The correlation scales for model errors were shorter than the scales for model fields, suggesting that more sensors are required for data assimilation than for monitoring of the CR estuary and plume;
4) The configuration of salinity arrays that were optimized using the state and the error covariance agreed, suggesting that an initial analysis of an observational array is possible even without access to an advanced data assimilation system;
5) Multiple initialization of the exchange-type algorithm yielded the most optimal array configuration. However, the deterministic simplifications of the exchange-type algorithm, such as the delete-only and the add-only algorithms, yielded equally optimal array designs.

Based on the analysis of the existing observatory for the CR estuary and plume, we recommended several strategies for the future management of observation arrays in the CR estuary and plume:

1) To improve the accuracy of salinity data assimilation in the CR estuary, we suggested placing the next observational sensor in the North Channel of the CR estuary, which would lead to an improvement in the R^2 value for salinity errors from 62% to 69%;
2) To improve the historical design of the observational array in the CR plume, we suggested deploying two sensors north and one sensor south of the CR mouth. This optimized array with three sensors was more informative of salinity errors (R^2 of 22.7%) than the historic RISE and OGI arrays with five sensors (R^2 of 18.7%);

3) To reduce the maintenance costs for the salinity array in the CR estuary, we suggested a strategy for reducing the existing salinity array from 14 to 10 sensors, however only in the case the remaining salinity sensors can be continuously operated. In case the continuous operation is impractical, due to bio-fouling of salinity sensors, we suggested that an increased number of sensors may be required for a robust operation.

In transitioning the developed array designs to operational staff of the CR observatory, we found that the suggested optimal observational strategies were only a part of considerations weighted by the science managers and the field staff. Alternative considerations included: logistical constraints, contractual obligations, and competing observational agendas—all of which would be impossible to express as a single cost function. However, the benefits of the proposed objective evaluation procedure were clear. Our objective procedure allowed scientists to prove or disprove their intuitions and to identify, otherwise overlooked, flaws in the design of the CORIE observatory.

5.7 Conclusions

In the research reported in this article, we demonstrated that it is possible to use an output from a circulation model and a data assimilation system to optimize the placement of sensors in such challenging dynamic environment as the CR estuary and plume. Our success with locating optimal sensors placements in the CR estuary and plume suggests that the developed algorithms can play a significant role in locating optimal placements of sensors in other observatories, such as regional coastal observatories, and national OOI and IOOS observatories. Our experience in the CR suggests that the developed algorithms for optimal sensor placement are most effective when used interactively, providing scientist, mangers, and field staff an opportunity to explore optimality of alternative sensor placements.

Appendix 1: BLUE estimates with factorized covariance matrix

To enable a fast computation of the BLUE estimate (Eq. 27), we used the factorized form of the prior covariance (Eq. 40). Following are the computational

procedures that we used to compute the factorized covariance (Eq. 40) and to compute the BLUE estimate (Eq. 27).

Computing factorized covariance: To efficiently compute the eigen-factorization of state and error covariance matrices, we used the empirical orthogonal functions (EOF) of the forward model, computed from a long, statistically representative simulation of the CR estuary and plume. The EOFs were computed using a memory-efficient EOF algorithm described in Chapter 2 of this thesis. To compute the factorized state covariance matrix $\mathbf{C}^{state} = \mathbf{LSL}^T$ for a simulation database discussed in section 5.3.4, we first computed the orthonormal EOF operator $\mathbf{\Pi}$, which describes the dominant spatial modes of this simulation database, and the time series of the EOF coefficients $x_s = \mathbf{\Pi}(x - \overline{x})$, which describe the temporal variability of this database in the EOF subspace. Given the time series of the EOF coefficients x_s, we computed the eigen-decomposition of their covariance matrix:

$$\mathrm{cov}(x_s) \xrightarrow{eig} \mathbf{L}_s\mathbf{S}\mathbf{L}_s^T, \qquad (44)$$

where **S** is the diagonal matrix of eigen-values, and $\mathbf{L_s}$ is the matrix of eigen-vectors**,** both of which are defined in the EOF subspace. To compute the eigen-vectors **L** for the full-space covariance, we reconstructed the reduced-space vectors $\mathbf{L_s}$ as the full-space vectors **L** using the EOF operator $\mathbf{\Pi}$:

$$\mathbf{L} = \mathbf{\Pi}\mathbf{L}_s. \quad (45)$$

To compute the factorized error covariance $\mathbf{C}^{error} = \mathbf{LSL}^T$, we used a similar three-step procedure as for computing the state covariance. First, we used Eq. (38) to compute a time-averaged prior covariance $\mathbf{C}_{EOF}^{error}$. Since each of the forecast error covariance matrices $\mathbf{P}_{xx}^{-}(k)$ in Eq. (38) was defined in the same low-dimensional EOF subspace, their time-average $\mathbf{C}_{EOF}^{error}$ was also defined in the same EOF subspace, characterized by the projection operator $\mathbf{\Pi}$. In the second step, we computed the eigen-decomposition of $\mathbf{C}_{EOF}^{error}$ as:

$$\mathbf{C}_{EOF}^{error} \xrightarrow{eig} \mathbf{L}_{error}\mathbf{S}\mathbf{L}_{error}^T. \quad (46)$$

In the third step, we computed the eigen-vectors **L** of the full-space error covariance $\mathbf{C}^{error}$ as:

$$\mathbf{L} = \mathbf{\Pi}\mathbf{L}_{error} . \qquad (47)$$

Computing the prior variance: To compute the prior variance $cpp = diag(\mathbf{C}_{pp})$ in Eqs. (30, 41, and 51), we used the following formula that avoided the unnecessary computation of cross-covariance terms:

$$cpp = \sum_{i=1}^{ns} \left(\left[\mathbf{H_p} \mathbf{L}(:,i) \sqrt{S(i,i)} \right]^2 \right) \qquad (48),$$

where *cpp* is a vector of prior variances, $\mathbf{L}(:,i)$ is the i^{th} eigen-vector, $S(i,i)$ is the i^{th} eigen-value, $[\ldots]^2$ is the element-wise squaring-operation, and *ns* is the number of retained eigen-vectors.

Computing posterior variance: To compute the posterior variance $dpp = diag(\mathbf{D}_{pp})$, without computing the unnecessary cross terms, we used the following two-step procedure. First, we computed the intermediate matrix product:

$$\mathbf{A} = \mathbf{C}_{py} \mathbf{C}_{yy}^{-1} , \qquad (49)$$

where, from Eqs. (29 and 40), the observation covariance $\mathbf{C}_{yy}$ and the cross-covariance $\mathbf{C}_{py}$ are:

$$\begin{aligned} \mathbf{C}_{py} &= (\mathbf{H}_p \mathbf{L}) \mathbf{S} (\mathbf{L}^T \mathbf{H}_y^T) \\ \mathbf{C}_{yy} &= (\mathbf{H}_y \mathbf{L}) \mathbf{S} (\mathbf{L}^T \mathbf{H}_y^T) + \sigma^2 \mathbf{I} \end{aligned} . \qquad (50)$$

In a case some of the observations were nonlinear, we linearized them around the mean state $\bar{x}$. In the second step, we compute the posterior variance *dpp* as:

$$dpp = cpp - \sum_{i=1}^{ns} \left(\mathbf{A}(:,i) .* \mathbf{C}_{yp}(:,i) \right) \quad (51),$$

where '.*' is the element-wise multiplication of vectors, and $(:,i)$ denotes the i^{th} column of the matrix.

List of Figures

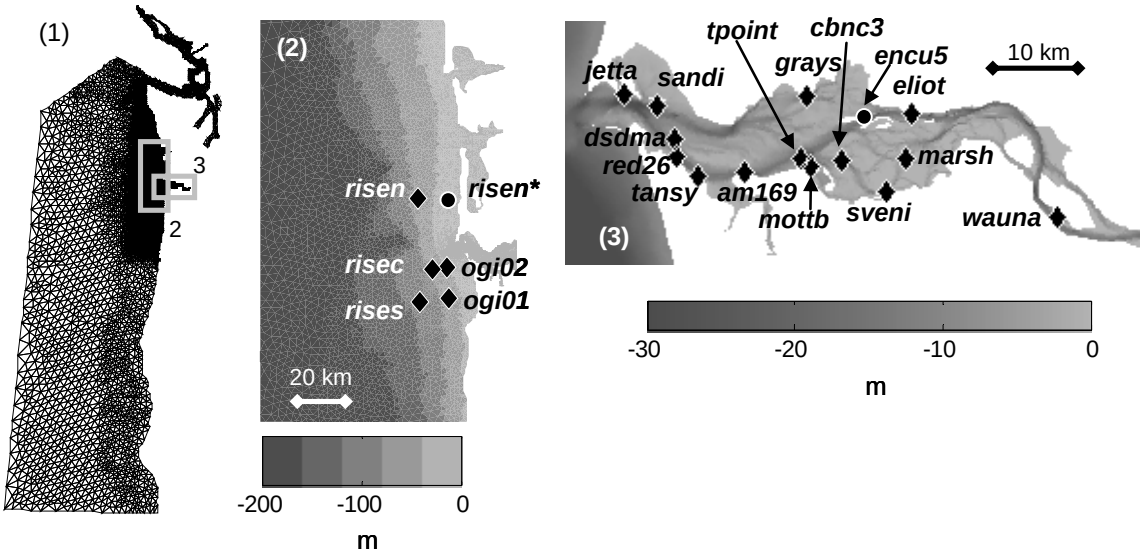


Figure 29: Map of the computational domain and observational stations.
(1) The computational grid. (2 and 3) The map of observation stations in the CR plume (2) and estuary (3). The map of stations in panels 2 and 3 is overlaid over the bathymetry. Stations *risen*, *risec*, and *rises* were operated by the RISE project (www.ocean.washington.edu/rise/). Tide gages *tpoint* and *wauna* were operated by National Oceanographic and Atmospheric Agency.

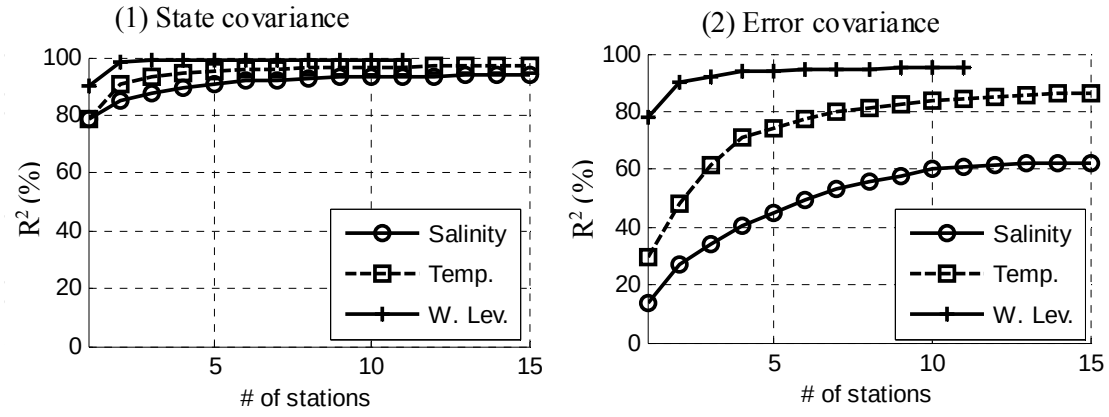


Figure 30: The R^2 values for the state (1) and the error (2) covariance as a function of the number of sensors retained in the existing observation array.

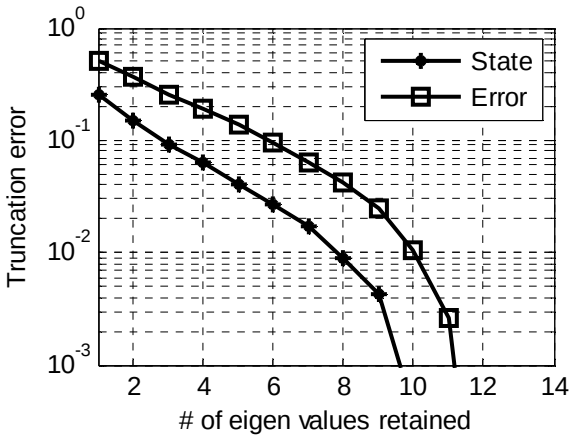


Figure 31: Eigen-spectrum of the observation covariance C_{yy} for the existing array of salinity sensors in the CR estuary.
The truncation errors e_i (Eq. 18) are plotted against the number of retained eigen-values λ_k. The observation covariance for the state (line marked with +) was computed using actual measurements and for the error (line marked with *squares*) was computed using errors for DB14 prediction.

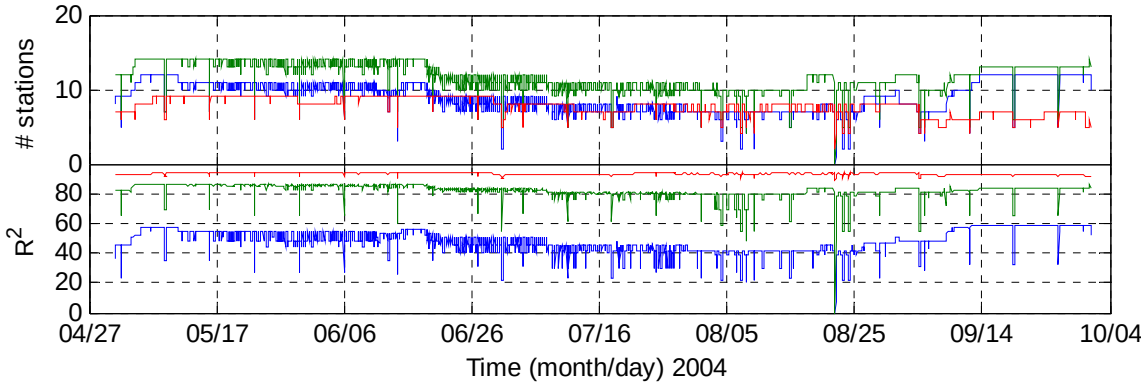


Figure 32: Redundancy of sensors to data dropouts.
(a) Number of available water level (red), salinity (blue), and temperature (green) sensors. (b) Time series of the R^2 values for the same three categories of sensors.

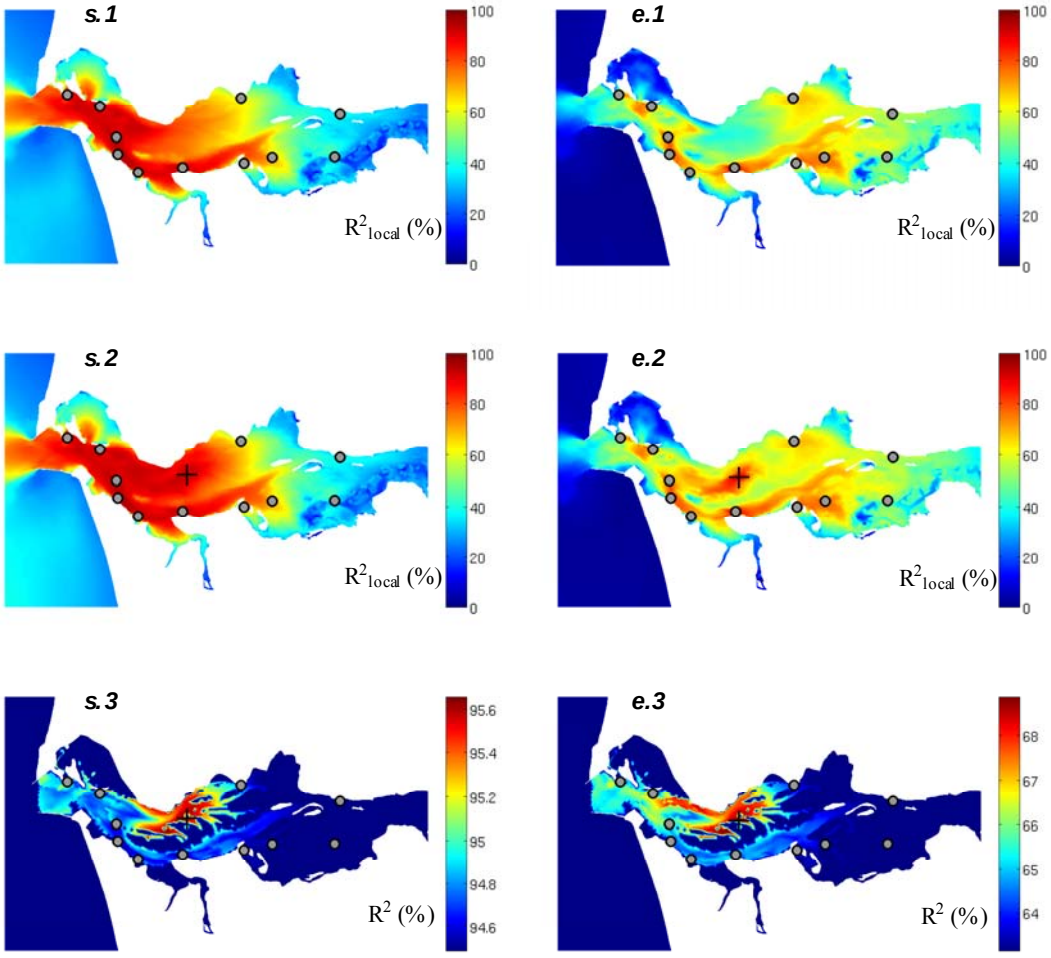


Figure 33: Optimal location of the next estuarine sensor
New sensor marked with '+'. Optimization was using the state (panels *s.1, s.2,* and *s.3*) and the error (panels *e.1, e.2,* and *e.3*) covariance. Top panels display local uncertainty reduction (R^2_{local} in %) for the existing salinity sensors (marked with circles), middle panels display R^2_{local} for augmented array (new sensor marked with +), and bottom panels display the added value of placing the next sensor in one of the feasible sensor-locations. All panels display maximum-over-depth quantities.

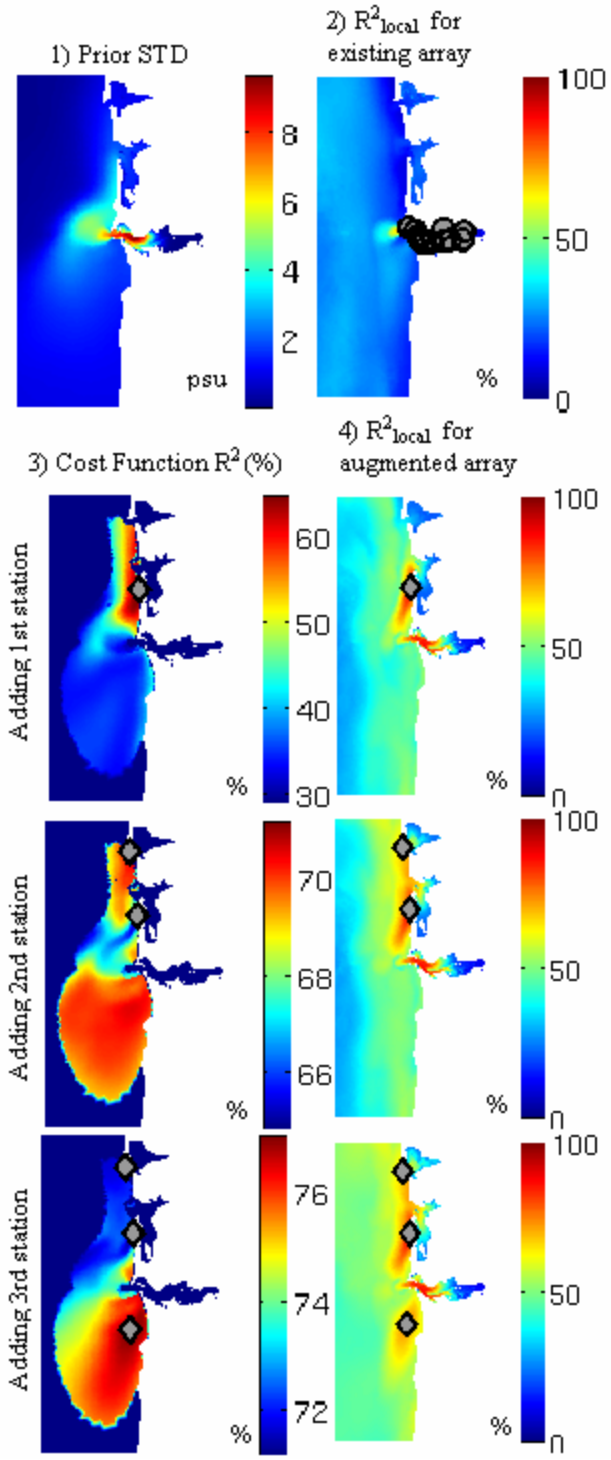


Figure 34: Optimal locations of sensors in CR plume. Optimization based on the state covariance. (1) Standard deviations (STD) of surface salinities (prior uncertainty). (2) Local uncertainty reduction R^2_{local} (%) for the existing array; existing sensors are marked with circles. (3) Value of the cost function for sensors one through three; new sensors are marked with diamonds. (4) Local uncertainty reduction R^2_{local} (%) for the augmented array. All quantities were computed on the surface, where the prior uncertainty was the largest.

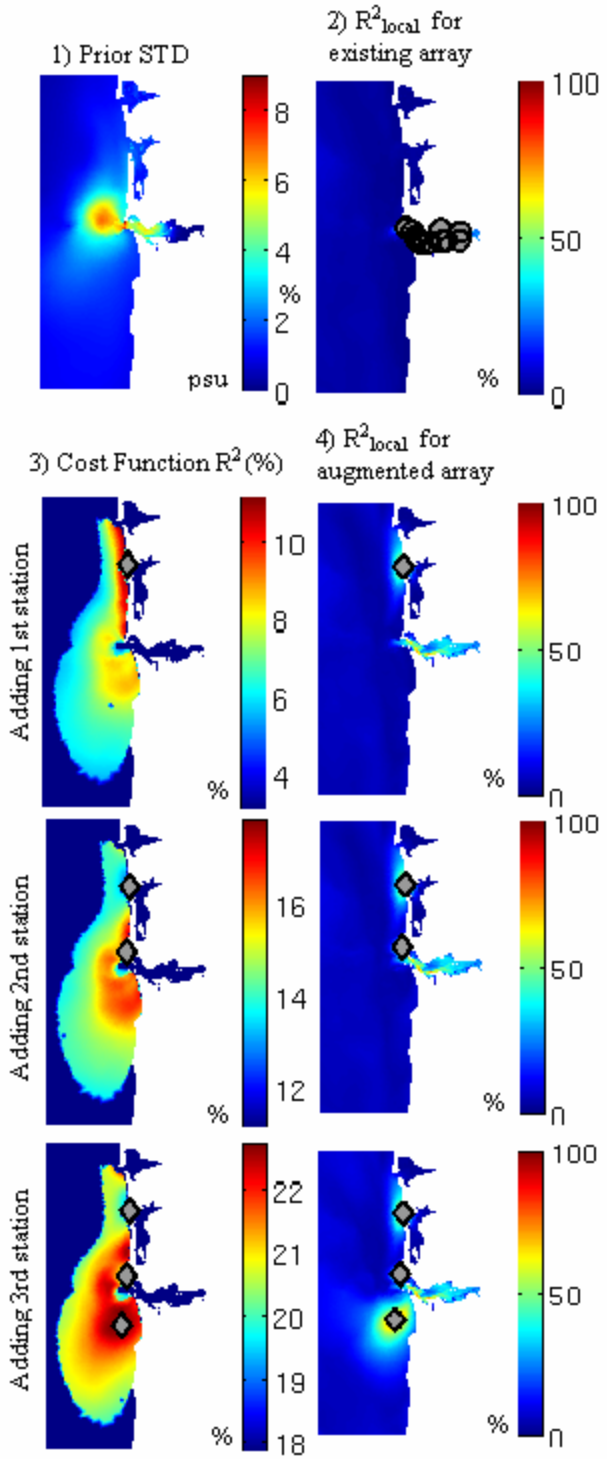


Figure 35: Same as Figure 34 but computed for the error covariance.

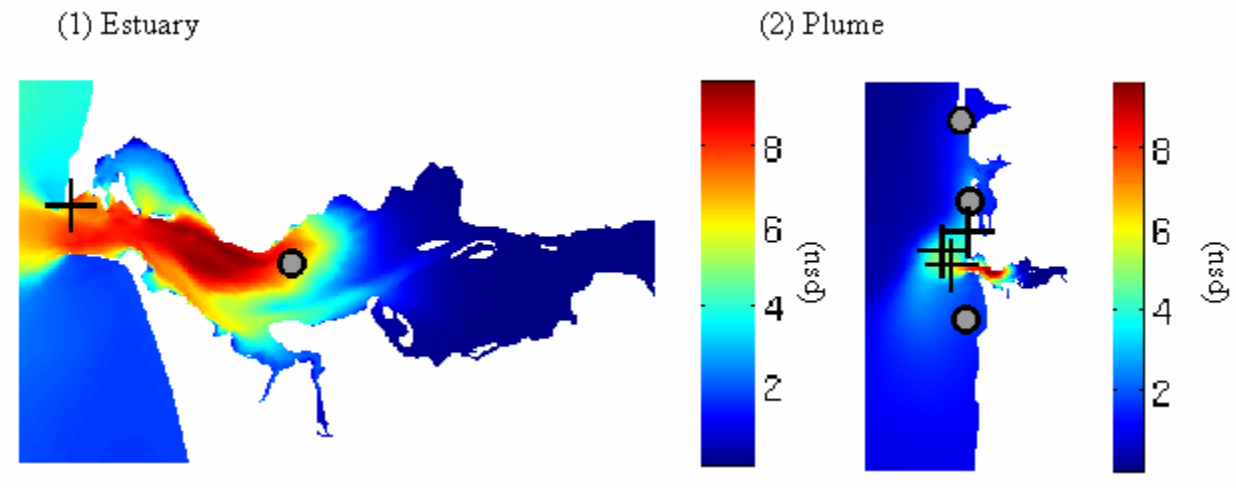


Figure 36: Sensitivity of sensor placement to optimization criteria.
Sensor locations that were located using the DET criterion are marked with (o) and using the MSE criterion, with (+). (1) Optimal placement of the next salinity sensor in the CR estuary. (2) Optimal placement of salinity sensors in the CR plume. The locations were computed using the state covariance and are overlaying the maximum-over-depth prior uncertainty.

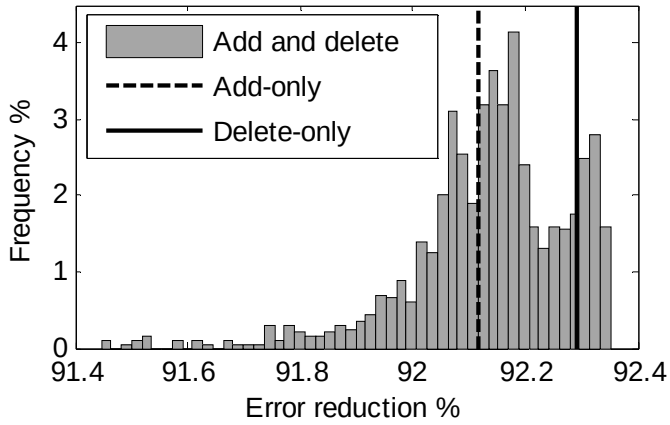


Figure 37: Optimality of sensor placement as a function of the optimization algorithm used.

The histogram shows frequency distribution of the R^2 values from 1000 randomly-initialized passes through the full exchange algorithm. Vertical lines mark the R^2 values for the add-only (dashed line) and the delete-only (solid line) algorithms. Experiments were conducted for the MSE criterion, the error covariance, and a time period of ½ years.

List of Tables

Table 2: Suggested order in which salinity sensors may be removed from the existing CR array.

Order	State		Error	
	Salinity sensor	R^2(%)	Salinity sensor	R^2(%)
I	II	II	IV	V
1	eliot	94	marsh	62
2	marsh	94	eliot	62
3	cbnc3	94	cbnc3	62
4	mottb	94	red26 (mid.)	62
5	tansy	93	am169 (bot.)	61
6	am169 (mid.)	93	tansy	60
7	red26 (mid.)	93	am169 (mid.)	58
8	dsdma	93	mottb	56
9	am169 (top)	92	red26 (bot.)	53
10	sandi	92	sandi	49
11	red26 (bot.)	91	red26 (top)	45
12	am169 (bot.)	89	grays	40
13	grays	87	jetta	34
14	jetta	85	dsdma	27
15	red26 (top)	79	am169 (top)	14

Table 3: Results of the most important and the least important salinity sensor experiment.

	Most important (%)		Least important (%)	
Salinity sensor	State	Error	State	Error
I	II	III	IV	V
jetta	74.1	12.0	-0.7	-3.9
Sandi	73.2	12.1	-1.1	-4.6
red26 (top)	78.9	11.2	-0.6	-2.2
red26 (mid)	74.4	8.7	-0.2	-0.5
red26 (bot)	72.0	7.8	-0.3	-0.7
Dsdma	77.5	13.5	-0.6	-4.0
Tansy	74.4	9.4	-0.2	-2.1
am169 (top)	64.3	13.7	-0.4	-3.6
am169 (mid)	58.5	8.9	-0.2	-0.7
am169 (bot)	51.3	7.8	-0.2	-0.7
Mottb	15.9	6.9	-0.1	-1.7
cbnc3	25.5	8.3	-0.1	-0.4
Grays	27.4	9.1	-0.5	-1.6
Eliot	0.0	0.2	0.0	0.0
Marsh	0.0	0.1	0.0	0.0

Table 4: Comparison of the salinity array designs in the CR plume.

Array name	Number of plume sensors	R^2 (%)	
		State	Error
Estuarine sensors	0	29.0	3.1
OGI	2	41.4	10.9
RISE	3	46.5	15.1
RISE+OGI	5	53.1	18.7
Optimal 2	2	71.2	17.9
Optimal 3	3	77.1	22.7
Optimal 5	5	82.7	32.0

Note: To make adequate comparison between the existing and the proposed optimal arrays, we only use information from the surface salinity sensors at OGI and RISE locations.

Table 5: Predicted and actual R^2 (%) values at the validation station *red26*.

Observation array	R^2	
	predicted	actual
no stations	0	0
– dsdma	53	43
– tansy	68	48
– jetta	78	49
– am169 (top)	78	49
– am169 (bot)	79	49
– sandi	79	50
– cbnc3	79	50
– am169 (mid)	79	50
– eliot	79	51
all stations	79	51

Note: Minus sign '–' in the first column indicates that the station was removed from the assimilation array.

Table 6: Predicted and actual R^2 (%) values at the validation station *am169*.

Observation array	R^2	
	predicted	Actual
no stations	0	0
– tansy	34	00
– cbnc3	44	40
– jetta	49	40
– red26 (top)	**50**	**47**
– sandi	**50**	**42**
– red26 (bot)	50	41
– dsdma	**51**	**40**
– eliot	51	41
– red26 (mid)	51	41
all stations	51	41

Note: Minus sign '–' in the first column indicates that the station was removed from the assimilation array. Bold indicates stations for which the predicted and actual rankings disagree.

References

Baker, N. L., and Daley, R. (2000), "Observation and Background Adjoint Sensitivity in the Adaptive Observation-Targeting Problem," *Quarterly Journal of the Royal Meteorological Society*, 126 (565B), 1431-1454.

Baptista, A. M. (2006), "CORIE: The First Decade of a Coastal Margin Collaborative Observatory," in *Oceans'06 MTS/IEEE*, Boston.

Baptista, A. M., Zhang, Y.-L., Chawla, A., Zulauf, M. A., Seaton, C., Myers, E. P., Kindle, J., Wilkin, M., Burla, M., and Turner, P. J. (2005), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: Ii. Application to the Columbia River," *Continental Shelf Research*, 25, 935-972.

Barnes, C. A., Duxbory, A. C., and Morse, B. A. (1972), "Circulation and Selected Properties of the Columbia River Effluent at Sea," in *The Columbia River Estuary and Adjacent Ocean Regions: Bioenvironmental Studies*, eds. D. L. Alverson and A. T. Pruter, Seattle, WA.: University of Washington Press, pp. 41-80.

Bennett, A. F. (1985), "Array Design by Inverse Methods," *Progress in oceanography*, 15, 129-156.

Bennett, A. F. (1992), *Inverse Methods in Physical Oceanography*, Cambridge University Press.

Bennett, A. F. (2002), *Inverse Modeling of the Ocean and Atmosphere*, Cambridge, UK: Cambridge University Press.

Berliner, M., Lu, X.-Q., and Snyder, C. (1999), "Statistical Design of Adaptive Weather Observations," *Jounral of Atmospheric Sciences*, 56 (15), 2536–2552.

Bertozzi, A. L., Kemp, M., and Marthaler, D. (2005), "Determining Environmental Boundaries: Asynchronous Communication and Physical Scales," *Lecture Notes In Control And Information Sciences*, 309, 25-42.

Bishop, C. H., Etherton, B. J., and Majumdar, S. J. (2000), "Adaptive Sampling with the Ensemble Transform Kalman Filter. Part I: Theoretical Aspects," *Monthly Weather Review*, 129, 420-436.

Bottom, D. L., Simenstad, C. A., Burke, J., Baptista, A. M., Jay, D. A., Jones, K. K., Casillas, E., and Schiewe, M. H. (2005), "Salmon at River's End. The Role of the Estuary in the Decline and Recovery of Columbia River Salmon," NOAA Technical Memorandum NMFS-NWFSC-68, U.S. Dept. Commer.

Bretherton, F. P., Davis, R. E., and Fandry, C. B. (1976), "A Technique for Objective Analysis and Design of Oceanographic Experiments Applied to Mode-73," *Deep-Sea Research*, 23, 559-582.

Burla, M., Baptista, A. M., Casillas, E., and Williams, J. G. (submitted), "The Influence of the Columbia River Plume on the Survival of Steelhead (Oncorhynchus Mykiss) and Chinook Salmon (O. Tshawytscha): A Numerical Exploration.," *Can. J. Fish. Aquat. Sci.*

Burla, M., Baptista, A. M., Zhang, Y.-L., Casillas, E., Bottom, D. L., and Simenstad, S. A. (2007), "Salmon Habitat Opportunity in the Columbia River Estuary: Modeling the Physical Environment to Inform Management Decisions," in *Coastal Zone '07*, Portland, OR.

Chang, N.-B., and Tseng, C. C. (1999), "Optimal Design of a Multi-Pollutant Air Quality Monitoring Network in a Metropolitan Region Using Kaohsiung, Taiwan as an Example," *Environmental Monitoring and Assessment*, 57, 121–148.

Daescu, D. N., and Carmichael, G. R. (2003), "An Adjoint Sensitivity Method for the Adaptive Location of the Observations in Air Quality Modeling," *Journal of the Atmospheric Sciences*, 60, 434-450.

Fedorov, V. V. (1972), *Theory of Optimal Experiments.*, Academic Press.

Fedorov, V. V. (1994), "Optimal Experimental Design: Spatial Sampling," *Calcutta Statistical Association Bulletin*, 44 (173-174).

Gandin, L. (1963), *Objective Analysis of Meteorological Fields*, Leningrad: Hydrometeoizdat.

Garcia-Berdeal, I., Hickey, B., and Kawase, M. (2002), "Influence of Wind Stress and Ambient Flow on a High Discharge River Plume," *Journal of Geophysical Research*, 107 (C9), 3130.

Grimes, C., and Kingsford, M. (1996), "How Do Riverine Plumes of Different Sizes Influence Fish Larvae: Do They Enhance Recruitment?" *Marine and Freshwater Research*, 47 (2), 191-208.

Hackert, E. C., Miller, R. N., and Busalacchi, A. J. (1998), "An Optimized Design for a Moored Instrument Array in the Tropical Atlantic Ocean," *Journal of Geophysical Research*, 103, 7491-7509.

Hickey, B. (2004), "River Influences on Shelf Ecosystems: Initial Impressions," in *AGU Fall meeting*, San Francisco.

Hickey, B., Geier, S., Kachel, N., and MacFadyen, A. (2005), "A Bi-Directional River Plume: the Columbia in Summer," *Continental Shelf Research*, 25 (14), 1631-1656.

Hickey, B. M., Pietrafesa, L. J., Jay, D. A., and Boicourt, W. C. (1998), "The Columbia River Plume Study: Subtidal Variability in the Velocity and Salinity Fields," *Journal of Geophysical Research*, 103 (C5), 10,339-10,369.

Jay, D., and Smith, J. D. (1990), "Circulation, Density Distribution and Neap-Spring Transitions in the Columbia River Estuary," *Progress in Oceanography*, 25 (1-4), 81-112.

Jay, D. A., and Flinchem, E. P. (1997), "Interaction of Fluctuating River Flow with a Barotropic Tide: A Demonstration of Wavelet Tidal Analysis Methods," *Journal of Geophysical Research*, 102 (C3), 5705-5720.

Langland, R., and G. Rohaly. (1996), "Adjoint-Based Targeting of Observations for Fastex Cyclones.," in *Preprints, Seventh Conf. on Mesoscale Processes*, Phoenix, AZ: Amer. Meteor. Soc.

Leonard, N. E., Paley, D., Lekien, F., Sepulchre, R., and Fratantoni, D. M. (2006), "Collective Motion, Sensor Networks and Ocean Sampling," *Proceedings of the IEEE, Special Issue on Networked Control Systems*.

Lu, Z., Leen, T. K., van der Merwe, R., Frolov, S., and Baptista, A. M. (2007), "Sequential Data Assimilation with Sigma-Point Kalman Filter on Low-Dimensional Manifold," *submitted to Journal of Inverse Problems*, also available at http://www.stccmop.org/files/CMOP-TR-07-001.pdf [Viewed on September 14, 2007].

MacCready, P., Banas, N. S., Hickey, B. H., and Dever, E. P. (2007), "A Model Study of Tide- and Wind-Induced Mixing in the Columbia River Estuary and Plume," *Submitted to Continental Shelf Res., April 2007.*

National Research Council. (2003), "Enabling Ocean Research in the 21st Century: Implementation of a Network of Ocean Observatories."

Ogren, P., Fiorelli, E., and Leonard, N. E. (2004), "Cooperative Control of Mobile Sensor Networks: Adaptive Gradient Climbing in a Distributed Environment," *IEEE Transactions on Automatic Control*, 49 (8), 1292-1302.

Oke, P. R., and Sakov, P. (2007), "Optimal Array Design: Application to the Tropical Indian Ocean," *Journal of Atmospheric and Oceanic Technology*, (accepted June 2007).

Pew Oceans Commission. (2003), "America's Living Oceans: Charting a Course for Sea Change. a Report to the Nation."

Pukelsheim, F. (1993), *Optimal Design of Experiments.*, Wiley.

Ripley, B. D. (1987), *Spatial Statistics*, New York: John Wiley.

She, J., Høyer, J. L., and Larsen, J. (2006), "Assessment of Sea Surface Temperature Observational Networks in the Baltic Sea and North Sea," *Journal of Marine Systems*, doi:10.1016/j.jmarsys.2005.01.004.

Silvey, S. D. (1980), *Optimal Design.*, Chapman and Hall.

U.S. Commission on Ocean Policy. (2004), "An Ocean Blueprint for the 21st Century. Final Report."

USACE. (2001), "Columbia River Channel Improvement Project: 2001 Biological Assessment," U. S. Army Corps of Engineers.

Zhang, Y.-L., and Baptista, A. M. (submitted), "A Semi-Implicit Eulerian-Lagrangian Finite-Element Model for Cross-Scale Ocean Circulation," *Ocean Modeling*.

Zhang, Y.-L., Baptista, A. M., and Myers, E. P. (2004), "A Cross-Scale Model for 3d Baroclinic Circulation in Estuary-Plume-Shelf Systems: I. Formulation and Skill Assessment," *Continental Shelf Research*, 24, 2187-2214.

Chapter 6

Conclusions

6.1 Summary of findings

In this dissertation, we demonstrated that, unlike previously thought, data assimilation (DA) does not need to be computationally challenging. Specifically, we developed and verified algorithms that enabled a fast ensemble DA that ran an estimate of 100-10,000 times faster than existing DA algorithms, and did not require development of hard-to-code adjoint models. We enabled this fast, adjoint-free DA by employing algorithms recently developed in the machine learning community, including algorithms for training neural-network model surrogates (van der Merwe, et al. 2007) and algorithms for nonlinear Kalman filtering (van der Merwe 2004, Lu, et al. 2007). The availability of these enabling algorithms allowed us to deploy our DA system with very limited personnel and computational resources, even in such a complex and nonlinear system as the Columbia River (CR) estuary and plume. As a result of this deployment, we were able to assimilate multiple years of observational data, to improve simulation of ecologically significant circulation features, to develop a real-time nowcast-forecast system, and to develop strategies for optimization of the existing and future observational arrays in coastal observatory for CR estuary and plume.

The summaries of our findings are organized in six themes presented in subsections that follow.

6.1.1 Improved representation of salinity, temperature, and water level variability in the CR estuary and plume

To verify our DA algorithms, we estimated the long-term variability of salinity, temperature, and water levels in the CR estuary and plume. Chapter 2-4 describe applications in the estuary; with chapter 3 also describing an application to the plume. The verification experiments included verifications against independent data from research cruises and leave-one-out cross-validation. These verification experiments showed that DA estimates were more accurate than a non-assimilated model in reproducing variability of water levels, salinity, and temperature in the CR estuary and plume. Our verification experiments also showed that assimilation of salinity, water levels, and temperature measurements had little effect on reducing errors in simulated velocities.

In the CR estuary, DA was effective in two ways:

1) Improving representation of episodic events, such as improving representation of water levels during periods of winter storms, salinity intrusion events during periods of low-river flow, and salinity and temperature during relaxation of upwelling events.
2) Correcting persistent modeling biases, such as under prediction of the salinity intrusion length and seasonal biases in heating and cooling of tidal flats.

In the CR plume, DA improved representation of transitions in the size and orientation of the plume, and improved the representation of vertical stratification.

The success of our experiments in the CR estuary and plume verified the of the developed DA algorithms in such a challenging application as the baroclinic circulation in the CR estuary and plume.

6.1.2 Verification of predicted uncertainty

One output of our DA algorithm was the estimate of model uncertainty. In chapter 2, we verified these uncertainty estimates against measurements of the model uncertainty. On average, our uncertainty estimates were slightly smaller (0.2 MSE units) than

measured uncertainty (0.3 MSE units), which indicated a slightly inconsistent filter. It is likely that future improvements to DA algorithms, detailed in section 6.2, will yield more accurate estimates of model uncertainty, which can be used as guidance in resource management decisions.

6.1.3 Improved estimates of ecologically significant circulation features in the CR estuary and plume

In chapters 2 and 3, we used independent data to establish that DA was more accurate in simulating the variability of salinity, temperature, and water levels in the CR estuary and plume. In chapter 3, we analyzed how these more accurate simulations represented the variability of three ecologically significant circulation features: the orientation and the size of the CR plume, and the length of the salinity intrusion in the CR estuary. Our analysis showed that:

- DA played a significant role in controlling the magnitude and timing of dynamic events in the simulated CR estuary and plume, such as events of extreme salinity intrusion and events of regime transitions in the CR plume.
- DA changed the response of the simulated salinity intrusion length to variations in the CR discharge, hence providing a different interpretation of the circulation in the CR estuary.
- DA changed the simulated size and orientation of the CR plume, usually in response to correcting for episodes of large errors in the non-assimilated model.
- DA corrected, with a limited success, for the underestimation of the salinity intrusion length in the CR estuary.

The observed improvements in representation of ecologically significant circulation features in the CR estuary and plume are encouraging. As a next step in our research, we plan to evaluate the impact of the improved state estimates on the ecological indicators of salmon survival described in (Bottom, et al. 2005, Burla, et al. 2007).

6.1.4 Real-time nowcast and forecast of the CR estuary

In the research reported in chapters 2 and 4, we studied the applicability of our DA and model surrogate algorithms to real-time nowcast and forecast of the CR estuary. In chapter 2, we showed that in the CR estuary the model surrogate, the EOF subspace, and the reduced-dimension KF were equally accurate on the data that was used for training of the model surrogate and of the EOF subspace, as on the new, unseen data. This equal accuracy on unseen data suggested that the developed DA methods can be successfully used in a real-time nowcast of the CR estuary without the need to retrain the model surrogate and the EOF subspace in real-time. In research reported in chapter 4, we verified this suggestion by implementing a real-time nowcast-forecast system for the CR estuary.

Validation of the developed real-time nowcast-forecast system showed that non-assimilated model-surrogate forecast was as accurate as non-assimilated forecast with a traditional circulation model SELFE, however only at a fraction of the computational effort (~10,000 faster). When observational data were assimilated, the accuracy of the CR estuary simulation was substantially improved, while the computational cost remained very low, ~100 faster than the non-assimilated forecast with SELFE. However, when a model-surrogate forecast was initialized with these DA estimates, the forecast skill of this model-surrogate forecast did not differ substantially from that of the non-assimilated forecast.

6.1.5 Computational efficiency of the developed DA methods

Our application of the developed DA algorithms in the CR estuary and plume showed that both the model surrogate and the DA algorithms computationally were highly efficient. In case of the CR estuary, model surrogates were ~10,000 faster than the traditional circulation code, which was ~100,000 faster than the real-time. Training of the model surrogate and the EOF basis was ~1/10 of the computational time that was required to generate the hindcast database, from which the training samples were drawn. The complete DA system was ~100 faster than non-assimilated model and ~100-10,000 faster than other ensemble-based DA methods.

6.1.6 Optimal placement of sensors in the CR estuary and plume

In chapter 5, we developed and verified algorithms for evaluating the efficiency of existing observational networks and for optimizing the placement of new observational sensors. In applying these algorithms to finding optimal placement of sensors in the CR estuary and plume, we determined that simulated fields predicted by a non-assimilated model and model error estimates predicted by a DA system can be effectively used for evaluation and optimization of observation arrays. For example, we found that the existing water level network in the CR estuary was very informative and robust, unlike the salinity network that was too sparse and too prone to data dropouts to reliably measure errors in the simulated salinity. To improve upon the existing network, we made a series of recommendations on placement of future sensors in the estuary and the plume. For example, we found that it is possible to deploy a more informative observational array in the CR plume than the historic observational array.

6.2 Conclusions and future work

Our successful verification of the developed algorithms for fast, model-independent DA and for evaluation and optimization of observational arrays suggests that these algorithms can be applied in other integrated ocean observatories. Such implementations can significantly reduce the technological and financial barriers for implementing the modeling component of an integrated ocean observatory. A successful implementation in other observatories will require training of an accurate model surrogate and a representative EOF space. It is likely that it will be possible to train such accurate model surrogates and EOF subspaces in observatories where a forward model already has high skill in representing the statistics of the modeled system, hence providing good samples for training of the model surrogate and the EOF subspace.

In verifying the developed algorithms for DA and array optimization, we identified a number of directions for future research, which may become important in future applications of the developed methods.

Further extensions to model surrogate methods:

- To enable more accurate training of very large networks, such as our network for the joint CR estuary and plume that had more than $2*10^6$ unknown weights, we suggest exploring two alternative strategies: reducing the size of the network and exploring algorithms for robust training of very large networks. To reduce the size of the network, we suggest optimizing the number of time embeddings and applying a dimension reduction postprocessor on the output of the neural network. To improve training of large networks, we suggest developing model surrogates based on the mixture of experts framework (Jacobs, et al. 1991), which are usually easier to train than a single global neural network.
- To achieve more parsimonious dimensions reduction for the forward model state, we suggest substituting linear EOF dimension reduction algorithm with nonlinear dimension reduction algorithms, such as algorithms by (Kambhatla, et al. 1997, Roweis, et al. 2000, Brand 2003).

Further extension to DA methods:

- To strictly enforce conservation laws in state estimates, we suggest incorporating conservation laws as a constraint either in the KF algorithm or in the dimension reduction algorithm.
- To estimate the dynamics that were not captured by the EOF subspace, we suggest developing a hybrid filter that will benefit from a fast model surrogate, which operates in the reduced space of the EOF functions, and from a slower forward model, which operates in the full space. One such hybridization, using an extended KF is described in (Hoteit, et al. 2003); however, a hybridization suitable for highly nonlinear systems is yet to be developed.
- To estimate poorly known model parameters, such as parameters in ecological models, we suggest extending the current model surrogate training software. At present, model surrogates are trained for a fixed choice of model parameters, which precludes their application for exploring sensitivities to changes in values of these parameters. Hence, and extension of the model surrogate software is needed that will accurately approximate the response of the model output to changes in the model parameters.

- To further improve accuracy of DA estimates in systems with very strong nonlinearity in dynamics and observations, a particle filter algorithm (Arulampalam, et al. 2002) may need to be implemented. Algorithms for several particle filters are already available in ReBel (van der Merwe 2002-2006)—the software that was used in our experiments. However, as our preliminary research showed (Lu, et al. 2007), care should be taken in adopting these algorithms to DA in high-dimensional spaces.
- To facilitate comparisons between sequential and variational methods, a model-surrogate-enabled variational DA needs to be developed. Model surrogates are well positioned to enable such comparisons at low computational and personnel cost since an adjoint and tangent linear models are always available for a model surrogate, even in cases were the original circulation model lacks tangent linear and adjoint models.

Further extension to observation network optimization methods:

- To better understand the synergies between observations with fixed, mobile, and satellite platforms, the developed algorithms for evaluation of fixed observational networks need to be extended.
- To enable targeting of adaptive observations, such as gliders, the ensemble transform KF (Bishop, et al. 2000) needs to be implemented. It is likely that model surrogates can enable fast computation of the ensemble forecasts that are required by the ensemble transform KF.

Future work with DA in the CR estuary and plume:

- To further improve accuracy of DA in the CR plume, an extension to DA software is required that will enable assimilation of data from gliders, research cruises, and coastal radars.
- To better diagnose sources of errors in the modeling systems and to improve the accuracy of model forecasts, a joint estimation of model state and model forcing may be required. Our preliminary research (Lu, et al. 2007) suggested that the developed DA algorithm is well suited for such joint estimation.

- To further improve the estimates of model uncertainty, further improvement to model error and forcing error statistics are necessary. One such suggested improvement will include adaptive tuning of the error statistics.
- To better represent the variability of velocities in the CR estuary and plume, DA of velocity measurements will be necessary.

Further verification of developed methods:

- To better characterize strength and limitations of developed algorithms, additional verification in other coastal and coastal-margin observatories will be needed.
- To facilitate application of the developed methods in other observatories, the high algorithmic complexity of the developed methods needs to be encapsulated in a well-designed software package, similar to IOM (Chua, et al. 2001), EnKF (Evensen 2004-2007), and ReBel (van der Merwe 2002-2006)—the existing packages for DA and sequential Bayesian estimation.

References

Arulampalam, S., Maskell, S., Gordon, N., and Clapp, T. (2002), "Tutorial on Particle Filters for on-Line Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transaction on Signal Processing*, 50 (2).

Bishop, C. H., Etherton, B. J., and Majumdar, S. J. (2000), "Adaptive Sampling with the Ensemble Transform Kalman Filter. Part I: Theoretical Aspects," *Monthly Weather Review*, 129, 420-436.

Bottom, D. L., Simenstad, C. A., Burke, J., Baptista, A. M., Jay, D. A., Jones, K. K., Casillas, E., and Schiewe, M. H. (2005), "Salmon at River's End. The Role of the Estuary in the Decline and Recovery of Columbia River Salmon," NOAA Technical Memorandum NMFS-NWFSC-68, U.S. Dept. Commer.

Brand, M. (2003), "Charting a Manifold," in *Neural Information Processing Systems*, MIT Press.

Burla, M., Baptista, A. M., Zhang, Y.-L., Casillas, E., Bottom, D. L., and Simenstad, S. A. (2007), "Salmon Habitat Opportunity in the Columbia River Estuary: Modeling the Physical Environment to Inform Management Decisions," in *Coastal Zone '07*, Portland, OR.

Chua, B. S., and Bennett, A. F. (2001), "An Inverse Ocean Modeling System," *Ocean Modelling*, 3, 137-165.

Evensen, G. (2004-2007), "Enseble Kalman Filter," http://enkf.nersc.no/.

Hoteit, I., and Pham, D. T. (2003), "Evolution of the Reduced State Space and Data Assimilation Schemes Based on the Kalman Filter," *Journal of Meteorological Society Japan*, 81, 21-39.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. (1991), "Adaptive Mixtures of Local Experts," *Neural Computation*(3), 79-87.

Kambhatla, N., and Leen, T. K. (1997), "Optimal Dimension Reduction by Local PCA," *Neural Computation*, 9 (1493-1516).

Lu, Z., Leen, T. K., van der Merwe, R., Frolov, S., and Baptista, A. M. (2007), "Sequential Data Assimilation with Sigma-Point Kalman Filter on Low-Dimensional Manifold," *submited to Journal of Inverse Problems*, also available at http://www.stccmop.org/files/CMOP-TR-07-001.pdf [Viewed September 14, 2007].

Roweis, S., and Saul, L. (2000), "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *SCIENCE*, 290 (5500), 2323-2326.

van der Merwe, R. (2002-2006), "Rebel: Recursive Bayesian Estimation Library," http://choosh.ece.ogi.edu/rebel/ [Viewed September 14, 2007].

van der Merwe, R. (2004), "*Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*," Ph.D. thesis, OHSU.

van der Merwe, R., Leen, T. K., Lu, Z., Frolov, S., and Baptista, A. M. (2007), "Fast Neural Network Surrogates for Very High Dimensional Physics-Based Models in Computational Oceanography," *Neural Networks*.