UNIVERSITY OF CALIFORNIA, SAN DIEGO

Estimation techniques for large-scale turbulent fluid systems

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

in

Engineering Sciences with Specialization in Computational Science

by

Christopher Hughes Colburn

Committee in charge:

Professor Thomas R Bewley, Chair
Professor Maurício de Oliveira
Professor Yoav Freund
Professor Philip E Gill
Professor Daniel M Tartakovsky

2011

The dissertation of Christopher Hughes Colburn is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____
Chair

University of California, San Diego

2011

DEDICATION

To Jessie, for her love and support.

EPIGRAPH

For wisdom will enter your heart,
and knowledge will be pleasant to your soul.
—Proverbs 2:10

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

First, and foremost, I thank Jessie, my wife. Had it not been for your enduring support and commitment to my education, this work would not have been possible. I am incredibly thankful for the sacrifices that you have made for me to pursue this dream–it's your turn now.

Thanks to my advisor, Professor Thomas R Bewley. Your commitment to pursuing truth is inspiring, challenging, and encouraging. Thank you for thoughtfully reviewing and guiding my work over the last 5 years.

To my colleagues at UCSD: Joe and David, my office-mates, and Paul and Rob, the remaining Flow Control Laboratory contingent, thanks for the many hours of good conversation (research related, or not). I will never forget the fun we had while traveling and working together to solve interesting problems. I also thank Nick and Andrew for their tireless work during the plume project. And, Michel Claes, thank you for encouraging me to get my PhD while we were still Master students.

To my colleagues at Pfizer, SDSC, LANL: Ian Fuller, Michael Miscus, Frank Lake, Nancy Ettele, Chris Misleh, Frank Alexander, Marian Anghel, and Chuck Farrar. Thank you for your financial support and the opportunity to collaborate. I look forward to future collaborations.

Thanks to my parents: Mom and Dad, without your love I would not be here today. Thank you for setting an example of how great spouses and parents live. I look forward to seeing how well grandparents live. George and Mary Jo, without your love I would not be happily married. Thank you for raising such an amazing daughter, and thank you for welcoming me into your family.

To my brother, John, I love you. I am so excited for you and Kerry to start a new life together. Thank you for being so dedicated to our family and thoughtful in how you approach every situation. I'm humbled by your example of service, both in the community and in our relationship.

To my siblings: John (Dzundza), you're a smart SOB, and I love you for it. Thank you for advising me and Jessie on medical issues–you have calmed our spirits more than once. You are a wonderful brother to Jessie, and I'm so thankful

for how she laughs when shes with you. Jimmy/James/Jimmol, as my grandfather used to say, you're a GOG (Good 'Ole Guy). I find my world-view constantly challenged by your perspective and insight. Thanks for going out of your way to host us in Oregon. You're going to be a great uncle. Dave and Kerry, welcome to the family. It's a great family.

Thanks to the extended Colburn family: Cookie, Gramps, and all the lost parks; Pamela, Ty, and Mac; Karen, Cort, Concha, Zoe, and Atticus; Michele, Carlos, Nicholas, Ryan, and the dogs. The fun we have had over the last few years has been rejuvenating. Thank you for hosting us when we visited, and playing with us during your vacations.

Thanks to the extended Hughes family: UJ and Alice; Mike, Janie, and Cameron; Jack and April; Lindsey and Lee; John and Amy; Ed, Rebecca, Kara, and Emma; Catesby and Brad; Tookey and David; Kenney, Jennifer, Jonas, and Rebecca; Bill, Mary, Charlie, Allison, and Kirstin. We're looking forward to the next family reunion.

Lastly, I am indebted to the all the fellas: Keith, Lew, Jerry, John, Mike, Zach, Logan, Steve, Trevor, PQ, Christophe, Inouye, Brothers, Jones, Troeber, Trey, Brendyn, Stuart, Colin, Pearce, and Keaton. Thank you for striving with me to "Keep the main thing, the main thing," advising me through marital issues, and encouraging me as a friend.

Much of this dissertation is the result of collaborative work between Professor Thomas R Bewley, David D Zhang, Joseph B Cessna, and myself, at the University of California, San Diego. Chapters 1 and 2, in particular, were substantially written and refined by Professor Bewley. Similarly, Dr. Zhang contributed heavily to chapters 2 and 4. I am indebted to these colleagues for their relentless pursuit of understanding and clarity.

Chapter 1, in part, is a reprint of the material as it was accepted by the Journal of Fluid Mechanics 2011. (Colburn, Cessna & Bewley (2011*a*), "State estimation in wall-bounded flow systems. Part 3. The Ensemble Kalman Filter." *Journal of Fluid Mechanics*.) The dissertation author was the primary investigator and author of this paper.

Chapter 2, in full, has been submitted for publication of the material as it may appear in Journal of Fluid Mechanics, 2011. (Colburn, Zhang & Bewley (2011*b*), "Gradient-based optimization methods for sensor & actuator placement in LTI systems." Submitted for publication to *Journal of Fluid Mechanics.*) The dissertation author was the primary investigator and author of this paper.

Chapter 3, in full, is a reprint of material as it has been drafted for publication in 2011. (Colburn & Bewley (2011), "A hybrid Game-theoretic, Ensemble Kalman approach to online learning of non-stationary, high-dimensional sequences." Drafted for publication.) The dissertation author was the primary investigator and author of this paper.

Chapter 4, in full, is a reprint of the material as it was accepted for publication at the *IEEE American Controls Conference, June 2011* (Zhang, Colburn & Bewley (2011*a*), "Estimation and Adaptive Observation of Environmental Plumes." In *ACC 2011: Proceedings of the 2011 conference on American Control Conference.* IEEE Press, to appear.) and later augmented with additional material as it may appear in Journal of Fluid Mechanics, 2011 (Zhang, Colburn & Bewley (2011*b*), "Estimation and Adaptive Observation of Environmental Plumes." Drafted for submission to *Journal of Fluid Mechanics.*). The dissertation author contributed substantially to the authorship and theoretical development of this paper.

VITA

2004	Associate Engineer – Intern, Scientific Applications International Corporation, San Jose, CA
2006	Bachelor of Science, Engineering Sciences, Mechanical Engineering, University of California, San Diego
Fall 2006	Graduate Teaching Assistant – Numerical Methods, University of California, San Diego
2006 - 2011	Graduate Student Researcher, University of California, San Diego and Los Alamos National Lab
2006 - 2007	Project Manager, Pfizer, La Jolla, CA
2008	Master of Science, Engineering Sciences, Mechanical Engineering, University of California, San Diego
Summer 2008	Graduate Student Researcher, Center for Turbulence Research, Stanford University
2011	Doctor of Philosophy, Engineering Sciences, Mechanical Engineering, University of California, San Diego

PUBLICATIONS

Colburn, C. and Bewley, T. "Optimal Sensor Placement for Estimation of Systems with Multiplicative Uncertainty." *Under preparation for 2011 submission.*

Zhang, D., Colburn, C., and Bewley, T. "Estimation and Adaptive Observation of Environmental Plumes." *Under preparation for 2011 submission.*

Colburn, C. and Bewley, T. "GEnKF: A hybrid ensemble, game-theoretic approach to state estimation." *Under preparation for 2011 submission.*

Colburn, C., Zhang, D., and Bewley, T. (2011) "Gradient-based optimization methods for sensor & actuator placement in LTI systems." *Submitted 2011 - Journal of Fluid Mechanics.*

Zhang, D., Colburn, C., and Bewley, T. (2011) "Estimation and Adaptive Observation of Environmental Plumes." *American Controls Conference*, San Francisco, CA, USA. Accepted for publication.

Colburn, C., Cessna, J., and Bewley, T. (2010) "State estimation in wall-bounded flow systems. Part 3. The Ensemble Kalman Filter." *Journal of Fluid Mechanics.* Accepted for publication.

Bewley, T.R., Cessna, J., Colburn, C., Ham, F., Iaccarino, G., and Wang, Q. (2008) "Object-oriented implementation of the EnVE estimation/forecasting algorithm and its application to high-performance turbulence codes." *Summer Program Proceedings from Center for Turbulence Research*, Stanford, CA, USA.

Cessna, J., Colburn, C., and Bewley, T. (2008) "EnVE: A new estimation algorithm for weather forecasting and flow control." *Proceedings from 4th AIAA Flow Control Conference*, Seattle, WA, USA.

Cessna, J., Colburn, C., and Bewley, T. (2007) "Multiscale retrograde estimation and forecasting of chaotic nonlinear systems." *Proceedings from 46th IEEE Conference on Decision and Control*, New Orleans, LA, USA.

ABSTRACT OF THE DISSERTATION

Estimation techniques for large-scale turbulent fluid systems

by

Christopher Hughes Colburn

Doctor of Philosophy in Engineering Sciences with Specialization in Computational Science

University of California, San Diego, 2011

Professor Thomas R Bewley, Chair

Estimation, in general, involves the determination of a probability distribution. This probability distribution describes the likelihood that any particular point in phase space accurately represents the truth state. That is, without knowing the actual state of a system, estimation strategies attempt to represent the probability of any given state using only a time history of noisy observations and, when available, an approximate dynamic model of the system of interest. For low-dimensional linear systems with Gaussian uncertainty in the initial state, state disturbances, and measurement noise the *de facto* solution to the estimation problem has been the Kalman Filter, which provides a method to propagate the mean and covariance forward in time, making the appropriate updates to both upon the

receipt of each new measurement. Although ubiquitous within academia and industry, since many systems of interest are either of very high dimension or cannot be described by linear dynamics with Gaussian uncertainty, the Kalman Filter is inappropriately applied in many applications.

The present thesis first reviews extensions of estimation theory to high-dimensional systems and demonstrates the first successful reconstruction of 3D turbulent channel flow ($Re_\tau = 100$), using wall information only, via the Ensemble Kalman Filter. Then a new hybrid method of estimation is described which improves estimation results for such high-dimensional systems by employing recent machine learning techniques (specifically, the Normal-Hedge algorithm) to consistently combine multiple estimators.

Lastly, since the measurement operator critically determines the quality of the estimate, a gradient-base sensor/actuator placement strategy for Linear Time Invariant systems is presented. Using a test system (the Ginzburg-Landau equation) this sensor placement strategy is demonstrated by determining the optimal location for sensors in such a way that minimizes scalar metrics of the covariance matrix. With this theory clearly established, optimal sensor placements are determined for dynamic sensors in a 2D environmental plume estimation problem.

Chapter 1

State estimation in wall-bounded flow systems. Part 3. The Ensemble Kalman Filter

1.1 Abstract

State estimation of turbulent near-wall flows based on wall measurements is one of the key pacing items in model-based flow control, with low-Re channel flow providing the canonical testbed. Model-based control formulations in such settings are often separated into two subproblems: estimation of the near-wall flow state via skin friction and pressure measurements at the wall, and (based on this estimate) control of the near-wall flowfield fluctuations via actuation of the fluid velocity at the wall. In our experience, the turbulent state estimation subproblem has consistently proven to be the more difficult of the two. Though many estimation strategies have been tested on this problem (by our group and others), none have accurately captured the turbulent flow state at the outer boundary of the buffer layer ($5 \leq y^+ \leq 30$), which is deemed to be an important milestone, as this is the approximate range of the characteristic near-wall turbulent structures, the accurate estimation of which is important for the control problem. Leveraging the Ensemble Kalman Filter (an effective variant of the Kalman filter which scales

well to high-dimensional systems), the present paper achieves at least an order of magnitude improvement (in the near-wall region) over the best results available in the published literature on the estimation of low-Reynolds number turbulent channel flow based on wall information alone.

1.2 Introduction

In the past, estimation of chaotic fluid systems was motivated mostly by the need for accurate weather forecasts. Today, the prospects of potential implementation of real-time feedback control in manufacturing systems, or perhaps even aerodynamics systems, provide new motivation to study this fundamental problem. Bewley & Liu (1998), Bewley (2001) and Högberg *et al.* (2003) developed optimal feedback kernels for the *control* of the linearized Navier-Stokes equation in a channel. The dependence of these feedback control kernels on the near-wall region, where the characteristic near-wall turbulent structures are located, emphasizes the importance of accurate state estimates in this region if effective feedback control is the ultimate aim.

State estimation is a problem that has been considered at length by researchers in many distinct communities. The "controls" and "dynamic systems" communities have focused primarily, but not exclusively, on problems which (a) have numerically tractable solutions to the corresponding Riccati equations (Zhou *et al.*, 1996), Linear Matrix Inequalities (Scherer *et al.*, 2002; Boyd *et al.*, 1994), or dynamic programs (Bertsekas *et al.*, 2001) [all three of these methods usually requiring a sufficiently low state dimension, or problems that can effectively be reduced to such via standard model reduction techniques], and (b) are characterized by uncertainties in the initial state, state disturbances, and measurement noise that are well approximated as Gaussian. In most cases, these assumptions are not valid in estimation problems related to turbulent flows.

The weather forecasting community, on the other hand, has focused on estimation (a.k.a. "data assimilation") strategies that are numerically tractable for high-dimensional discretizations of PDE systems. The two primary classes of

data assimilation strategies which have been developed in this community and are available today for multiscale uncertain systems are the Ensemble Kalman Filter [EnKF; see, Evensen (2003)], and space/time variational [4DVar; see, Bouttier & Courtier (1999)] methods.

EnKF methods, which come in a few distinct variations, are particularly well suited for nonlinear multiscale systems with substantial uncertainties. Even for some low-dimensional problems, EnKF methods have been shown to provide significantly improved state estimates in certain nonlinear problems for which the more traditional Extended Kalman Filter breaks down. The statistics of the estimation error in the EnKF are not propagated via a covariance matrix, but are instead implicitly represented via the distribution of several perturbed trajectories ("ensemble members"), which themselves are propagated with the full nonlinear system model. On many problems, in practice, the collection of these ensemble members (itself called the "ensemble") accurately captures the dominant directions of uncertainty of the estimation error (in phase space) even when a relatively small number of ensemble members is used. This is the key feature that lends the EnKF method its remarkable numerical tractability in high-dimensional problems.

4DVar methods propagate state and sensitivity ("adjoint") simulations back and forth across an optimization window of interest. An optimization is performed based on these iterative marches to minimize a cost function balancing: (a) a term accounting for the misfit of the estimate with the measurements over the optimization window, with (b) a "background" term accounting for the "old" estimate (that is, based on the measurements and statistics obtained prior to the present optimization window). Though such a retrospective analysis is certainly beneficial in certain ways in the estimation of nonlinear systems, 4DVar methods are not as natural as EnKF methods for representing the principal directions of uncertainty in the estimate, which is a critical ingredient of any effective state estimation strategy.

1.2.1 Related background on state estimation

Estimation, in general, involves the determination of a probability distribution. This probability distribution describes the likelihood that any particular

point in phase space accurately represents the truth state. That is, without knowing the actual state of a system, estimation strategies attempt to represent the probability of any given state using only a time history of noisy observations and an approximate dynamic model of the system of interest. Given this statistical distribution, estimates can be inferred about the "most likely" state of the system, and how much confidence should be placed in that estimate. Unfortunately, in this most general form, the estimation problem is intractable for most systems. However, given certain justifiable assumptions about the nature of the model and its associated disturbances and uncertainties, simplifications can be applied with regards to how the probability distributions are modeled. In linear systems with Gaussian uncertainty of the initial state, Gaussian state disturbances, and Gaussian measurement noise, it can be shown that the probability distribution of the optimal estimate is itself Gaussian [see, e.g., Anderson & Moore (1979)]. Mathematically, for any linear system

$$\mathbf{x}_k = \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1} \qquad \mathbf{w} \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}) \tag{1.1}$$

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \qquad \mathbf{v} \sim \mathrm{N}(\mathbf{0}, \mathbf{R}), \tag{1.2}$$

the entire distribution of the estimate in phase space can be represented exactly by its mean $\bar{\mathbf{x}} = E[\mathbf{x}]$ and its second moment about the mean (that is, its covariance) $\mathbf{\Sigma}$ where

$$\mathbf{\Sigma} = E\big[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^H\big]. \tag{1.3}$$

In this paradigm, the notation $\mathbf{x} \sim \mathrm{N}(\bar{\mathbf{x}}, \mathbf{\Sigma})$ denotes explicitly that the random variable $\mathbf{x}$ has a normal (Gaussian) distribution about its mean and covariance. This is the essential piece of theory that leads to the traditional Kalman Filter (KF), first introduced by Kalman (1960) and Kalman & Bucy (1961).

Sequential data assimilation methods (e.g., the KF) provide a method to propagate $\hat{\mathbf{x}}$ and $\mathbf{P}$ (that is, estimates of $\bar{\mathbf{x}}$ and $\mathbf{\Sigma}$, respectively) forward in time, making the appropriate updates to both upon the receipt of each new measurement. It is useful to think of these quantities, at any given time t_k, as being

conditioned on a subset of the available measurements. The notation $\hat{\mathbf{x}}_{k|j}$ represents the highest likelihood estimate at time t_k given measurements up to and including time t_j. Similarly, $\mathbf{P}_{k|j}$ represents the covariance corresponding to this estimate. In particular, $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ are often called the prediction estimate and the prediction covariance, whereas $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$ are often called the current estimate and the current covariance. The linear, discrete-time evolution equations for the Kalman Filter are

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1} \tag{1.4}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}_{k-1}^{H} + \mathbf{Q}_{k-1} \tag{1.5}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{k|k-1}\,\mathbf{H}^{H}(\,\mathbf{H}\,\mathbf{P}_{k|k-1}\,\mathbf{H}^{H} + \mathbf{R})^{-1}(\mathbf{y}_k - \mathbf{H}\,\hat{\mathbf{x}}_{k|k-1}) \tag{1.6}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{P}_{k|k-1}\,\mathbf{H}^{H}(\,\mathbf{H}\,\mathbf{P}_{k|k-1}\,\mathbf{H}^{H} + \mathbf{R})^{-1}\mathbf{H}\,\mathbf{P}_{k|k-1} \tag{1.7}$$

where (1.4) and (1.5) propagate the state, and covariance, respectively, between measurement updates, and (1.6) and (1.7) update the state and covariance, respectively, at measurement times. Note that $\hat{\mathbf{x}}_{k|k+K}$, for some $K > 0$, is often called a smoothed estimate, and may be obtained in the sequential setting by a Kalman smoother [see, Rauch *et al.* (1965) and Anderson & Moore (1979)].

For nonlinear systems with relatively small uncertainties, a common variation on the KF known as the Extended Kalman Filter (EKF) has been developed in which the mean and covariance are propagated about a linearized trajectory of the full system. Essentially, if a Taylor-series expansion for the nonlinear evolution of the covariance is considered and all terms higher than quadratic are dropped, what is left is the differential Riccati equation associated with the EKF covariance propagation. Though this approach gives acceptable estimation performance for nonlinear systems when uncertainties are small as compared with the fluctuations of the state itself, EKF estimators often diverge when uncertainties are more substantial, and other techniques are needed.

At its core, the linear thinking associated with the uncertainty propagation in the KF and EKF breaks down in chaotic systems. Chaotic systems are characterized by stable manifolds or "attractors" in n-dimensional phase space.

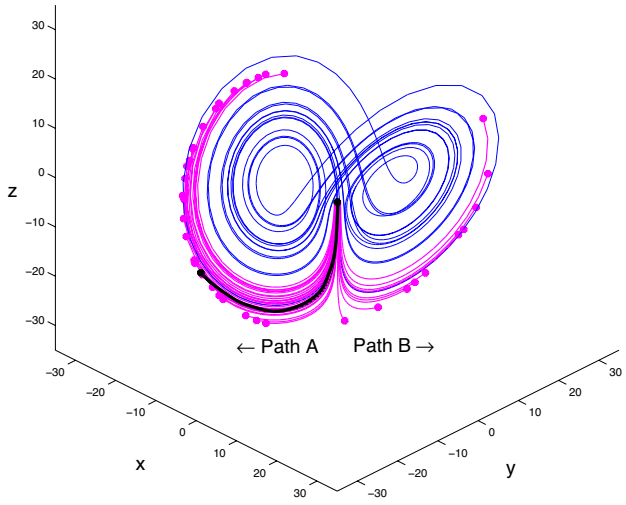


Figure 1.1: Non-Gaussian uncertainty propagation in the Lorenz system [see, Lorenz (1963)]. The black point in the center shows a typical point located in a sensitive area of this chaotic system's attractor in phase space, representing a current estimate of the state. The thick black line represents the evolution in time of the trajectory from this estimate. If the uncertainty of the estimate is modeled as a very small cloud of points, centered at the original estimate with an initially Gaussian distribution, then the additional gray lines show the evolution of each of these perturbed points in time. A Gaussian model of the resulting distribution of points is, clearly, completely invalid.

Such attractors are fractional-dimensional subsets (a.k.a. "fractal" subsets) of the entire phase space. Trajectories of chaotic systems are stable with respect to the attractor in the sense that initial conditions off the attractor converge exponentially to the attractor, and trajectories on the attractor remain on the attractor. On the attractor, however, trajectories of chaotic systems are characterized by an *exponential divergence* of slightly perturbed trajectories. That is, two points infinitesimally close on the attractor at one time will diverge exponentially from one another as the system evolves until they are effectively uncorrelated.

Just as individual trajectories diverge along the attractor, so does the uncertainty associated with them. This uncertainty can diverge in a highly non-Gaussian fashion when such uncertainties are not infinitesimal (see Figure 1.1). Estimation techniques that attempt to propagate probability distributions under linear, Gaussian assumptions fail to capture the true uncertainty of the estimate

in such settings, and thus improved estimation techniques are required. In this case, when estimating an ODE with n states, the probability density function (PDF) of the estimate in phase space must be discretized and propagated for accurate results. This converts the straightforward propagation of statistics in the KF problem (with two coupled ODEs, of order n and n^2) into a much more difficult PDE, of dimension n, governing the evolution of the PDF itself; this PDE is known as the Fokker-Plank equation [see, e.g., Jazwinski (1970), p. 164]. This PDE may be approximated and evolved with a Lagrangian method, referred to in this setting as a Particle Filter [PF; see Arulampalam *et al.* (2002)], or with a grid-based method, which may be made tractable by exploiting the sparsity of the PDF in phase space [see, Bewley & Sharma (2010)]; both approaches are numerically tractable only for extremely small values of n [i.e., $n \lesssim 5$].

1.2.2 Prior work on the estimation of turbulent channel flow

In high-dimensional estimation problems [e.g., $n \gtrsim O(10^6)$] Bayesian methods based on propagating the full Fokker-Plank PDE, such as the PF method, are completely out of the question. KF and EKF approaches are also infeasible, unless further decoupling or approximation is applied, due to their reliance on the propagation of the covariance (of order n^2) of the PDF of the estimate.

Exploiting a spectral decomposition to decouple the associated equations, Högberg *et al.* (2003) solved the full Kalman filter problem for the state estimation of near-wall flows. They did this by solving the (decoupled) estimation Riccati equations for the individual Fourier modes of the linearized system, which makes these equations tractable, then inverse-transforming the result back to physical space to obtain implementable feedback convolution kernels associated with the estimation problem. Using an improved problem formulation, Hœpffner *et al.* (2005) found effective kernels for the state estimation problem in transitional channel flow (that is, for small perturbations from the laminar state) that converged properly upon grid refinement. Chevalier *et al.* (2006) then attempted to develop a nonlinear extension of this work in order to apply it effectively to a fully-developed

turbulent flow using an Extended Kalman Filter. This work meticulously calculated a numerical model of the statistics of the nonlinear terms of a fully-developed channel flow, then used these statistics as the covariance of the state disturbances when computing the estimator feedback gains via the (linear) Kalman formulation.

Alternative methods have attempted the use of a Wiener Filter (Martinelli, 2009) and 4DVar [Bewley & Protas (2004), a vector-based variant of the Kalman Smoother reviewed] to estimate $Re_\tau = 100$ and $Re_\tau = 180$ flows, respectively. In Bewley & Protas (2004) the authors also examined the direct extrapolation of wall skin friction and pressure measurements of a turbulent flow into the flowfield via Taylor series analysis; unfortunately, it was found that the domain of convergence was much smaller than 20 viscous units. Chandrasekhar's method has also be proposed for reducing the dimension of the covariance propagation equations in the KF/EKF [see, Kailath (1973)]. This approach involves the propagation of a reduced-order factored form of the time derivative of the covariance matrix, as well as the propagation of the feedback gain matrix itself, rather than the (numerically-intractable) propagation of the full covariance matrix. This approach is promising for problems of this class, and has yet to be tried for the estimation of near-wall turbulence.

The groundwork described in the previous two paragraphs, upon which the present paper is based, is reviewed further, and put into a broader context, in Kim & Bewley (2007). The broad range of existing studies on this canonical problem provides a benchmark against which new approaches may be compared. The present paper applies the Ensemble Kalman Filter (EnKF) to estimate a $Re_\tau = 100$ turbulent flow based on wall skin friction and wall pressure measurements alone. The remainder of this paper summarizes briefly the Ensemble Kalman Filter and the principal heuristics, localization and covariance inflation, required in its application to large-scale systems. The results, presented in §1.4, might be considered the first "successful" estimation of this difficult benchmark problem.

1.3 The Ensemble Kalman Filter

The Ensemble Kalman Filter (EnKF), first proposed by Evensen (1994), is a modern stochastic alternative to Chandrasekhar's method, described above, for state estimation in high dimensional systems, and is reviewed in depth in Evensen (2003, 2009*a*,*b*). Simply stated, in standard implementations of the EnKF, a sample covariance matrix replaces the forecast of the covariance matrix in (1.6). Even though only second-order statistics of the distribution are typically used at each measurement update in the EnKF approach, the full nonlinear dynamics of the system are used to propagate each candidate realization between measurement updates.

We now review briefly the formulation of the standard EnKF, using a continuous-time representation of the system state $\mathbf{x}(t)$ and measurements $\mathbf{y}_k = \mathbf{y}(t_k)$ available at discrete times t_k:

$$\frac{\partial \mathbf{x}(t)}{\partial t} = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t)), \tag{1.8}$$

$$\mathbf{y}_k = h(\mathbf{x}_k) + \mathbf{v}_k. \tag{1.9}$$

The system dynamics $f(\cdot)$ in this formulation may be nonlinear and forced by some known function $\mathbf{u}(t)$, and are also assumed to be corrupted by random "state disturbances" $\mathbf{w}(t)$ with known statistics. Similarly, the measurement operator $h(\cdot)$ may be nonlinear, and is assumed to be corrupted by additive white "measurement noise" $\mathbf{v}_k$ with covariance $\mathbf{R}_k$.

Recall from the introduction that the EKF propagates the full covariance matrix and uses it to perform measurement updates according to Bayes' rule, assuming a Gaussian PDF. The EnKF is, in a sense, quite similar, but builds an estimate $\mathbf{P}^e$ of the covariance matrix $\mathbf{P}$ based on an outer product matrix

quantifying the deviation of the ensemble members from their mean:

$$\mathbf{P}^e_{k|k-1} = \frac{(\delta\hat{\mathbf{X}}_{k|k-1})(\delta\hat{\mathbf{X}}_{k|k-1})^H}{N-1}, \tag{1.10a}$$

$$\delta\hat{\mathbf{X}}_{k|k-1} = \left[\delta\hat{\mathbf{x}}^1_{k|k-1} \quad \delta\hat{\mathbf{x}}^2_{k|k-1} \quad \dots \quad \delta\hat{\mathbf{x}}^N_{k|k-1}\right], \tag{1.10b}$$

$$\delta\hat{\mathbf{x}}^j_{k|k-1} = \hat{\mathbf{x}}^j_{k|k-1} - \hat{\mathbf{x}}_{k|k-1}, \tag{1.10c}$$

$$\hat{\mathbf{x}}_{k|k-1} = \frac{1}{N}\sum_j \hat{\mathbf{x}}^j_{k|k-1}. \tag{1.10d}$$

Using this "sample" (that is, approximate) covariance matrix $\mathbf{P}^e$, a standard KF measurement update may be performed:

$$\hat{\mathbf{x}}^j_{k|k} = \hat{\mathbf{x}}^j_{k|k-1} + \mathbf{P}^e_{k|k-1}\,\mathbf{H}^H\left[\mathbf{H}\,\mathbf{P}^e_{k|k-1}\,\mathbf{H}^H + \mathbf{R}_k\right]^{-1}(\mathbf{y}^j_k - \mathbf{H}\,\hat{\mathbf{x}}^j_{k|k-1}), \tag{1.11}$$

where $\hat{\mathbf{x}}^j_{k|k-1}$ denotes the j'th ensemble member at timestep k based on measurements up to $\mathbf{y}_{k-1}$, $\mathbf{P}^e_{k|k-1}$ denotes the sample covariance matrix, as given in (1.10a), based on the collection of ensemble members, $\mathbf{y}^j_k$ denotes a discrete-time random vector with statistical distribution $N(\mathbf{y}_k, \mathbf{R})$, and $\mathbf{H}$ denotes a linearization of the operator $h(\cdot)$ about the mean state estimate $\hat{\mathbf{x}}_{k|k-1}$. For more information on the standard EnKF measurement update and it's properties, the reader is referred to Evensen (2003).

It is important to differentiate between the PF and the EnKF, since they are perhaps easily confused. Although both estimation methods use the governing equations to propagate sets of perturbed realizations through phase space, the measurement update in each method is fundamentally different. The PF uses a *weighted linear combination* of these perturbed candidate realizations to approximate the PDF of the estimate, with the weights being adjusted each time a measurement is taken via Bayes' rule. In contrast, the EnKF effectively uses identical weights on each realization, instead *shifting the realizations themselves*, according to a Kalman-like update formula, whenever measurements are taken.

In the computationally efficient implementation of the EnKF the outer product formula for the sample covariance matrix (1.10a) is kept in its factored form when calculating the update (1.11) [or the modified form of this update, given

below] in order to retain the numerical tractability of the result. That is, $\mathbf{P}^e$ is represented as the product of two matrices of order $n \times N$ and $N \times n$, bypassing the full computation of the $n \times n$ matrix $\mathbf{P}^e$, which is important because, in the implementation, $N \ll n$. We thus rewrite (1.11) as

$$\hat{\mathbf{x}}^j_{k|k} = \hat{\mathbf{x}}^j_{k|k-1} + \alpha\, \delta\hat{\mathbf{X}}_{k|k-1}\, \delta\hat{\mathbf{Y}}^H_{k|k-1} \times \left[\alpha\, \delta\hat{\mathbf{Y}}_{k|k-1}\, \delta\hat{\mathbf{Y}}^H_{k|k-1} + \mathbf{R}_k\right]^{-1} (\mathbf{y}^j_k - \mathbf{H}\, \hat{\mathbf{x}}^j_{k|k-1}) \tag{1.12}$$

where $\delta\hat{\mathbf{Y}}_{k|k-1} = \mathbf{H}\, \delta\hat{\mathbf{X}}_{k|k-1}$ and $\alpha = 1/(N-1)$.

Butala *et al.* (2008) established that the EnKF, when formulated correctly for systems with linear dynamics, asymptotically converges to the Kalman result as the number of ensemble members becomes sufficiently large. An abbreviated proof is provided below for convenience. Note in particular that the $\mathbf{y}^j_k$ are perturbed in a statistically consistent fashion so that the covariance term $\mathbf{KRK}^H$ is properly recovered below in (1.21).

Theorem 1.3.1 (Equivalence of the EnKF to KF) *In the limit of an infinite number of ensemble members (i.e., $N \to \infty$), the estimated ensemble mean and covariance converge to the equivalent Kalman filter equations* (1.6) *and* (1.7)*, respectively, when using the EnKF update equation* (1.11)*.*

Proof 1.3.1 *Consider the rewritten EnKF update* (1.11) *as the unique solution for the random variable $\hat{\mathbf{x}}^j_{k|k}$ conditioned on the random variables $\hat{\mathbf{x}}^j_{k|k-1}$ and $\mathbf{y}^j_k$*

$$\hat{\mathbf{x}}^j_{k|k} = \hat{\mathbf{x}}^j_{k|k-1} + \mathbf{K}^e(\mathbf{y}^j_k - \mathbf{H}\, \hat{\mathbf{x}}^j_{k|k-1}) \tag{1.13}$$

$$\mathbf{K}^e = \mathbf{P}^e_{k|k-1}\mathbf{H}^H \left[\mathbf{H}\mathbf{P}^e_{k|k-1}\mathbf{H}^H + \mathbf{R}_k\right]^{-1}. \tag{1.14}$$

After recalling the definition of the covariance matrix

$$\mathbf{P}_{k|k-1} = \lim_{N\to\infty} \frac{1}{N-1} \sum_{j=1}^{N} (\hat{\mathbf{x}}^j_{k|k-1} - \hat{\mathbf{x}}_{k|k-1})(\hat{\mathbf{x}}^j_{k|k-1} - \hat{\mathbf{x}}_{k|k-1})^H \tag{1.15}$$

$$= \lim_{N\to\infty} \mathbf{P}^e_{k|k-1}, \tag{1.16}$$

as discussed in Butala et al.*, the EnKF gain matrix $\mathbf{K}^e$ converges to the Kalman*

gain matrix $\mathbf{K}$ *as* $n \to \infty$ *by Slutsky's Theorem* (Slutsky, 1925; for explanation, see Gut, 2005, Theorem 11.4, p. 249). *Thus, the expected value of* (1.13) *can be rewritten*

$$\begin{aligned}
\hat{\mathbf{x}}_{k|k} &= \lim_{N\to\infty} \frac{1}{N}\sum_{j=1}^{N} \hat{\mathbf{x}}_{k|k}^{j} && (1.17)\\
&= \lim_{N\to\infty} \frac{1}{N}\sum_{j=1}^{N} \hat{\mathbf{x}}_{k|k-1}^{j} + \mathbf{K}\left[\frac{1}{N}\sum_{j=1}^{N} \mathbf{y}_k^j - \mathbf{H}\,\frac{1}{N}\sum_{j=1}^{N} \hat{\mathbf{x}}_{k|k-1}^{j}\right] && (1.18)\\
&= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}\big(\mathbf{y}_k - \mathbf{H}\,\hat{\mathbf{x}}_{k|k-1}\big) && (1.19)
\end{aligned}$$

which is identical to the Kalman filter state update (1.6).

By performing an equivalent analysis for the covariance of the random variable $\hat{\mathbf{x}}_{k|k}^{j}$ *the covariance update equation is recovered similarly*

$$\begin{aligned}
\mathbf{P}_{k|k} &= \lim_{N\to\infty} \frac{1}{N-1}\sum_{j=1}^{N} (\hat{\mathbf{x}}_{k|k}^{j} - \hat{\mathbf{x}}_{k|k})(\hat{\mathbf{x}}_{k|k}^{j} - \hat{\mathbf{x}}_{k|k})^{H} && (1.20)\\
&= (\mathbf{I} - \mathbf{KH})\mathbf{P}_{k|k-1}(\mathbf{I} - \mathbf{KH})^{H} + \mathbf{KRK}^{H} + \mathbf{\Phi} + \mathbf{\Phi}^{H} && (1.21)
\end{aligned}$$

where

$$\mathbf{\Phi} = \lim_{N\to\infty} \frac{1}{N-1}\sum_{j=1}^{N} \left[(\mathbf{I} - \mathbf{KH})\,(\hat{\mathbf{x}}_{k|k-1}^{j} - \hat{\mathbf{x}}_{k|k-1})\,(\mathbf{y}_k^j - \mathbf{y}_k)^{H}\,\mathbf{K}^{H}\right] = \mathbf{0} \quad (1.22)$$

which implies that (1.21) *and* (1.7) *are equivalent, and the proof is complete.*

Although this theoretical result justifies applying the EnKF to many problems, it does not provide practical guidelines for choosing ensemble size for more general applications, which is necessary during implementation. For linear problems of very high dimension [$n \geq \mathcal{O}(10^6)$] Furrer & Bengtsson (2007) show that convergence of the trace of the covariance matrix is possible when the number of ensemble members scales like the square of the order of the state [i.e., $N \sim \mathcal{O}(n^2)$]. Furrer & Bengtsson also show that there can sometimes be considerable bias in the estimator even when the number of ensemble members is of the same order as the order of the state dimension [i.e., $N \sim \mathcal{O}(n)$, also discussed in Evensen (2003,

2009*a*)], a much less strict requirement.

Based on this analysis alone, the requirements for convergence, which scale as poorly as the storage requirements for the KF and EKF, restrict applications of the EnKF to low-dimensional systems. As a result, two heuristics must be implemented in practice to reduce the negative side effects associated with a reduced ensemble size: *localization* and *covariance inflation*. As mentioned in the introduction, it is often found in practice that when these two heuristics are used with an ensemble-based approach, the dominant directions of uncertainty of the estimation error (in phase space) are captured accurately even when a relatively small number of ensemble members is used.

1.3.1 Localization

Localization is an artificial distance-based suppression of the off-diagonal components of the sample covariance matrix $\mathbf{P}^e_{k|k-1}$ as represented by (1.10a). It was first proposed by Houtekamer & Mitchell (2001), and is an essential ingredient to the success of the EnKF in practice. It is introduced to eliminate spurious correlations in the covariance matrix that arise from the fact that it is usually grossly under-sampled (that is, $N \ll n$). Note in (1.10a) that the off-diagonal components of the covariance matrix $\mathbf{P}^e_{k|k-1}$ are obtained by averaging the product of a flowfield perturbation at one point in the physical domain with a flowfield perturbation at another point in the physical domain. If these two points are separated by a large distance, it may be argued on physical grounds that this averaged product should be small; localization thus imposes this decay of correlation with distance, even if the system is so grossly under-sampled that (1.10a) does not capture this decay (which is usually the case).

The sample covariance matrix $\mathbf{P}^e_{k|k-1}$ in (1.11) may thus be replaced by

$$\mathbf{P}^e_{k|k-1} = \frac{\rho \bullet (\delta\hat{\mathbf{X}}_{k|k-1})(\delta\hat{\mathbf{X}}_{k|k-1})^H}{N-1} \tag{1.23}$$

where ρ is a distanced-based localization function, and $\bullet$ denotes the element-wise product. Using this modified sample covariance formula, (1.12) may be rewritten

as

$$\hat{\mathbf{x}}^j_{k|k} = \hat{\mathbf{x}}^j_{k|k-1} + \alpha\rho_1 \bullet (\delta\hat{\mathbf{X}}_{k|k-1}\, \delta\hat{\mathbf{Y}}^H_{k|k-1}) \times \\ \left[\alpha\rho_2 \bullet\ (\delta\hat{\mathbf{Y}}_{k|k-1}\, \delta\hat{\mathbf{Y}}^H_{k|k-1}) + \mathbf{R}_k\right]^{-1} (\mathbf{y}^j_k - \mathbf{H}\,\hat{\mathbf{x}}^j_{k|k-1}) \tag{1.24}$$

where, $\alpha = 1/(N-1)$ is the constant defined in (1.12), $\rho_1^{i,m}$ is a distance-based localization function relating the i-th state and the m-th measurement, and $\rho_2^{m_1,m_2}$ is a localization function relating m_1-th measurement and m_2-th measurement; both functions approach unity as the distance between the corresponding flow quantities approaches zero, and both approach zero as the distance between the corresponding flow quantities becomes large.

1.3.2 Covariance Inflation

Another challenge when using under-sampled representations of probability distributions in high-dimensional state-space systems is "covariance collapse." This occurs when the majority of ensemble members are distributed on a fraction of the attractor, and thus the computed statistics do not capture all of the principal directions of uncertainty. Anderson & Anderson (1999) review this phenomena and the effect it has in weather forecasting applications, and discuss its simple practical solution *covariance inflation*. Much earlier, Anderson & Moore (1979, p. 131-134) demonstrated that, even in linear KF applications, insensitivity to measurements (resulting from accounting improperly for model uncertainty) can lead to filter divergence. Furrer & Bengtsson (2007) further argue that most sources of error in ensemble filters result in underestimation of the ensemble variance; thus, covariance inflation is a natural mechanism for correcting the unknown deficiencies that lead to an underestimated prior variance. Covariance inflation, as proposed by Anderson & Anderson (1999), simply pushes the ensemble members away from the mean by some arbitrary growth factor β at each timestep.

$$\hat{\mathbf{x}}^j = \beta\,(\,\hat{\mathbf{x}}^j - \hat{\mathbf{x}}\,) + \hat{\mathbf{x}} \tag{1.25}$$

In weather forecasting applications, inflation parameters in the range of $\beta \in [1.005, 1.05]$ demonstrate significantly improved estimator performance. Attempts to develop adaptive methods for tuning β have been made in Anderson (2009, 2007) and Wang & Bishop (2003), but unfortunately the results so far do not appear to justify their complexity.

1.4 Numerical Results

We now characterize the ability of the EnKF, as described above, to estimate a 3D incompressible turbulent channel flow, given measurements of the skin friction and pressure on uniformly-spaced 16×16 array on each wall. The numerical computations presented use the standard spectral-spectral-second-order-finite-difference approach of Bewley *et al.* (2001) to simulate the uncontrolled, constant mass-flux turbulent channel flow on a 64^3 grid with $L_x = 2\pi$, $L_z = \pi$, and $L_y = 2$. [Note that this code-base uses L_x, L_z, and L_y to denote the streamwise, spanwise, and wall-normal directions, respectively. The reader is referred to Bewley *et al.* (2001) for more details.] For clarity, the (nonlinear) Navier-Stokes equation for the present application is given by

$$\left(\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} - \nu \frac{\partial^2 u_i}{\partial x_j^2} + \frac{\partial p}{\partial x_i} \right) = \mathrm{f}(x) \tag{1.26}$$

where incompressibility, uniform density and viscosity is assumed. The flow is governed by the incompressible Navier-Stokes equation with uniform density and viscosity. By defining the half-channel height δ, mean skin friction $\bar{\tau}_w = -\nu \, \partial \bar{u_1} / \partial n$, and mean friction velocity $u_\tau = (\bar{\tau}_w / \rho)^{0.5}$, this equation can be conveniently non-dimensionalized where time, space, and velocity are normalized by ν/u_τ^2, ν/u_τ, and u_τ, respectively. As a result, the Reynolds number becomes simply a function of viscosity $Re_\tau = 1/\nu$. By choosing $\nu = 0.01$ the domain is re-expressed, in non-dimensional form, with $L_x^+ = 628$, $L_z^+ = 314$, and $L_y^+ = 200$. Although the domain size is not identical to our previous work in flow estimation, it is larger than the minimal flow unit required for the onset of turbulence (Jimenez & Moin,

1991). This, along with statistics generated for validation purposes, ensures that the results are in fact representative of an estimator tracking a turbulent flow at low Reynolds number. The "truth" model is calculated as an identical simulation running in parallel with the EnKF-based estimator.

Since the simulation relies on Fourier transforms for computational efficiency, all data is stored in frequency space as the simulation advances. This conveniently allows for measurements to be extracted through high-order spectral interpolation schemes. Wall shear stress in the streamwise and spanwise directions were calculated via spectral interpolation in the wall-parallel directions and second-order interpolation in the wall-normal direction. When pressure measurements were required, a Poisson equation was solved.

As mentioned in §1.3.1 and §1.3.2, localization and inflation are ad hoc yet essential ingredients to the success of any large-scale EnKF implementation. The distance-dependent localization functions ρ used in the present work were chosen to be exponential in shape,

$$\rho^{i,j} = \rho(\mathbf{x}(i), \mathbf{x}(j)) = e^{|\mathbf{x}^i - \mathbf{x}^j|^2_{\mathbf{Q}}}, \tag{1.27}$$

where $\mathbf{Q} = \text{diag}\{\,[q_1,\, q_2,\, q_3]\,\} > 0$ is a diagonal weighting matrix related to three length scales. The appropriate selection of each q_i reflect the "trust" associated with the ability of the sample covariance matrix to construct accurate correlations in the streamwise, wall-normal, and spanwise directions. These diagonal elements of $\mathbf{Q}$ correspond to the half-height of the exponential function decay, and are generally selected to correspond to known flow statistics. These length scales were determined from correlation studies of uncontrolled turbulence, analogous to those reported in Kim *et al.* (1987) and Bewley *et al.* (2001), and are subjected to a minor amount of additional variation. The inflation parameter $\beta = 1.01$ was selected based on reported results from the weather forecasting community (Anderson & Anderson, 1999).

The simulation was performed using the Triton cluster at San Diego's Super Computing Center (SDSC), where each simulation required 70-hours of compute time on 66 parallel cores (the details of the cluster can be found at the SDSC

website[1]), with each core corresponding to an ensemble member. This algorithm design choice was determined by two unrelated but important constraints: the discretization of the domain and the size of the cluster on which the simulation was performed. The computational expense of the EnKF simulation performed and the desire to drive it all the way to statistical steady state (2200 ensemble updates were performed during each simulation) prevented us from performing more extensive parametric studies on these three length parameters at the present time, or repeating the study at higher Reynolds numbers, both of which are left for future work.

The quality of the reconstruction is determined by comparing the perturbation component of the true velocity field with the same perturbation component of the estimated velocity field. Normalized error and correlation measures, as defined by Bewley & Protas (2004), are used for comparison with previous work.

$$\mathrm{Errn}(\, q'_{est},\, q'_{tru}\,) = \frac{\int_0^{L_x}\int_0^{L_z}(\, q'_{est} - q'_{tru}\,)^2 dz\, dx}{\int_0^{L_x}\int_0^{L_z}(q'_{tru})^2 dz\, dx} \tag{1.28}$$

$$\mathrm{Corr}(\, q'_{est},\, q'_{tru}\,) = \frac{\int_0^{L_x}\int_0^{L_z} q'_{est} q'_{tru} dz\, dx}{\sqrt{\int_0^{L_x}\int_0^{L_z}(q'_{est})^2 dz\, dx}\sqrt{\int_0^{L_x}\int_0^{L_z}(q'_{tru})^2 dz\, dx}}. \tag{1.29}$$

Note that quantities are primed to emphasize that the perturbation component of the velocity field (that is, the instantaneous velocity component minus its planar average) is being used in the comparison. The subscripts $()_{est}$ and $()_{tru}$ correspond to the "estimated" and "truth" values, respectively. These two normalized measures account for the (x, z)-plane averaged statistics as a function of time and distance from the wall. The long-time average of these measures provides a rigorous quantification of the quality of the state estimate as a function of distance from the wall, approximating their corresponding expected values, $E[\mathrm{Errn}(y,t)]$ and $E[\mathrm{Corr}(y,t)]$, at statistical steady state.

The error norm defined above is perhaps the more sensitive of the two criteria. It is normalized by the planar-averaged mean-squared energy of the truth simulation, which makes it a particularly sensitive measure near the wall, where

[1] http://tritonresource.sdsc.edu/cluster.php

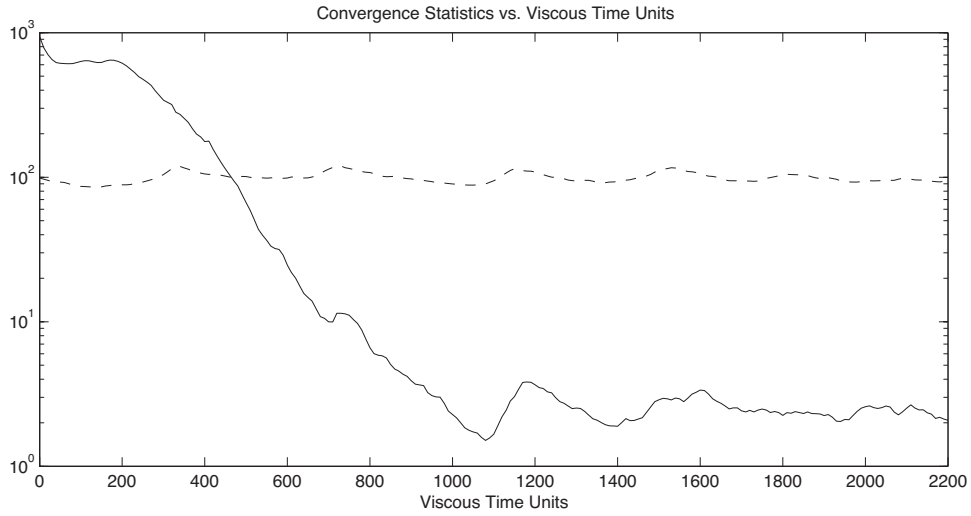


Figure 1.2: L_2 energy of the error of the estimation, solid, and total wall drag, dashed, as a function of t^+. Time averaged results in Figures 1.3 and 1.4 are averaged from time $t^+ = 1000$ to $t^+ = 2200$. The magnitude and variations in total wall drag are consistent with those of an $Re_\tau = 100$ turbulent flow.

this quantity approaches zero. (Note that an error norm near unity indicates that the estimate is completely decoupled from the truth, whereas an error norm near zero indicates that the estimate is in perfect agreement with the truth.) When significant error is present, the correlation is useful to quantify the planar-averaged phase error, as distinct from the planar-averaged amplitude error; an error in the amplitude of the estimate (but not its phase) will adversely affect the error norm, but not the correlation. Note also that a correlation near unity indicates perfect phase alignment of the estimate with the truth.

Figure 1.2 shows the L_2 energy of the difference between the estimate and the truth as a function of t^+. Note that the error norm and correlation were calculated, and averaged, from $t^+ = 1000$ to $t^+ = 2200$ (1200 viscous time units after apparently converging to statistical steady state). Through observation of the variations in averaged skin-friction (skin friction averaged over the top/bottom walls) it is clear that temporal averages are taken over multiple flow-throughs with substantial statistical averaging.

The most significant test of the estimator in this problem, of course, is to quantify its convergence starting from arbitrary initial conditions. Figures 1.3 and

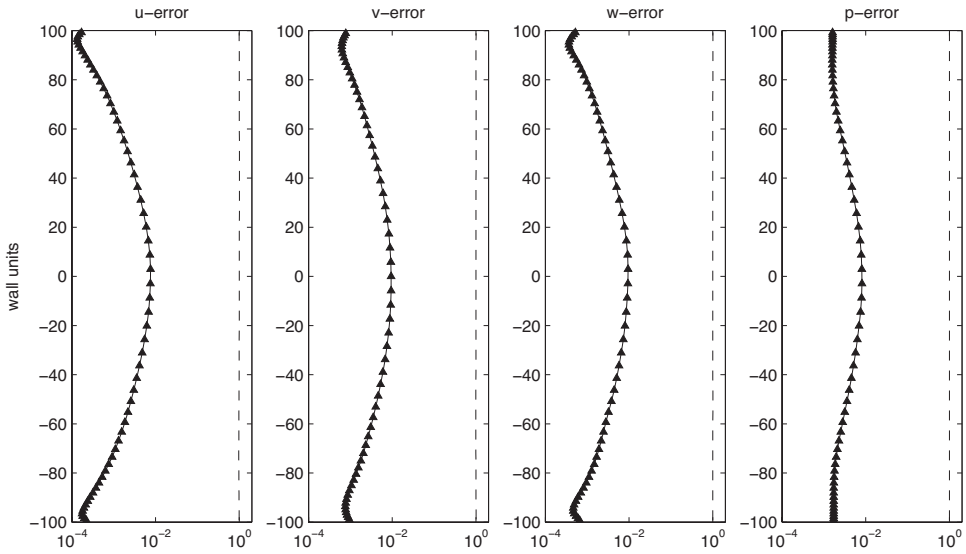


Figure 1.3: Error norm in the estimation of an $Re_\tau = 100$ flow using the EnKF, starting from bad initial conditions. The localization used is given in (1.27), with localization length scales of $\mathbf{Q} = \mathrm{diag}([50, 50, 25])$ and localization constant $\beta = 1.01$. The $- - -$ line represents the statistics of the error before estimation was attempted (averaged over 2400 viscous time units), and the $-\blacktriangleleft-$ line represents the error norm of the EnKF at statistical steady state.

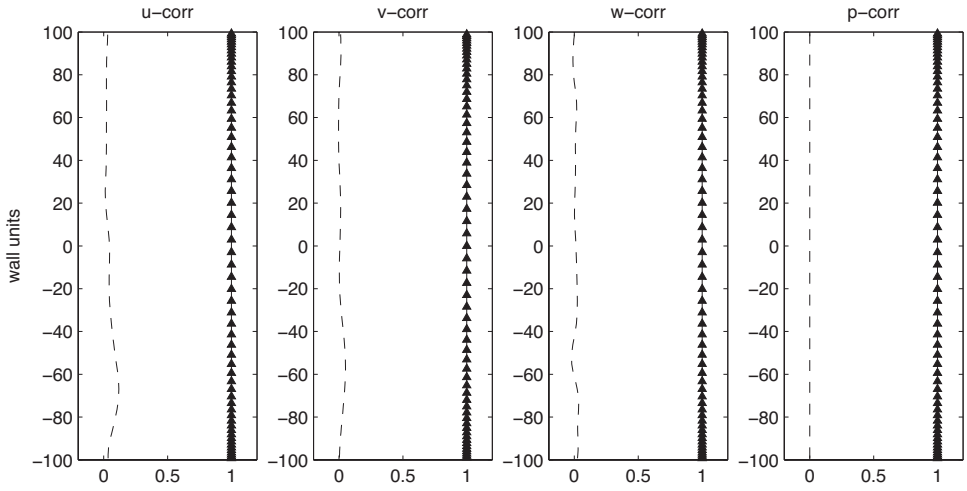


Figure 1.4: As in Figure 1.3, but reporting the correlation.

1.4 thus report such a test, using the most suitable values of the localization parameters identified thus far, $\mathbf{Q} = \text{diag}\{[50, 50, 25]\}$. An initial condition for the estimator was generated from a fully-developed turbulent flow simulation (independent of the truth model) and snapshots of this flow were taken every $\Delta t^+ = 200$ viscous time units to initialize each ensemble member. Symmetry across the channel centerline in Figures 1.3–1.4 reflect the approximate statistical convergence in the simulation. This symmetry would be perfect if statistical steady state were in fact reached.

In this case, near perfect synchronization of the EnKF with the truth is observed. The EnKF estimation was found to perform with at least an *order of magnitude less error* at 20 viscous units from the wall than the previous best estimation results reported in the literature on this problem (see, §1.2.2). In particular, we observe error measurements of 0.001, 0.001, 0.001 and correlation measurements of 0.99, 0.99, and 0.99 (in the u, v, and w velocity components, respectively). As a point of comparison Chevalier *et. al.* (2006, when using the Extended Kalman Filter) reported error measurements of 0.5, 0.88 and 0.9 and correlation measurements of 0.87, 0.59, and 0.59 at the same location.

1.5 Conclusions and future work

The EnKF depends on the sample covariance matrix $\mathbf{P}^e$, which is a low-rank approximation of the estimated covariance matrix $\mathbf{P}$. It is well known by those who use the EnKF in weather forecasting applications that the finite size of the ensemble in the EnKF causes spurious correlations and covariance collapse. In practice these phenomena must be compensated for through a distance-dependent localization function and covariance inflation in order to ensure adequate convergence of the estimator.

This paper has presented the first near-perfect state estimation of an $Re_\tau = 100$ turbulent flow using wall information only; that is, we have demonstrated a sustained synchronization of the state estimate with the truth when a random initial condition is used in the estimator, and the localization function is tuned

appropriately. When comparing with previous results in the published literature, at least an order of magnitude less error was observed at 20 viscous units from the wall.

Previous unreported results found that the ability to achieve such estimator convergence is apparently fairly sensitive to the localization parameter used in the wall-normal direction, and apparently fairly insensitive to the localization parameters used in the streamwise and spanwise directions. To a certain extent, these results reflect the law of the wall, the fundamental idea that any turbulence near-wall flow varies in a predictable statistical fashion as a function of the distance from the wall. Exploring the estimator performance as a function of these localization parameters may reveal inherent properties of the flow not yet known and warrants further investigation.

Besides further tuning of these heuristics, one interesting possibility for improving the present estimation strategy is to implement a "Rogallo transform" (Rogallo, 1981; Rogallo & Moin, 1984) for the quantities being estimated. In his pioneering work, Rogallo showed that, in regions of high shear (in this case, near a wall), a convenient transformation on the domain may be defined that moves something like a windshield wiper. Such a transformation on the domain might in the present problem provide a slower evolution of the individual discretized flow perturbation quantities being estimated, thus creating an easier problem for the EnKF to estimate. Another promising idea is to explore recent hybrid methods for state estimation that consistently combine the strengths (and numerical tractability) of the EnKF and 4DVar approaches (see, Cessna, 2010).

Once the best estimator possible for this problem has been developed, of course, the problem of controlling a turbulent flow based on this estimate must be revisited, as well as the extension of this approach to higher Reynolds numbers. The present investigation, which represents the first reasonable high-fidelity estimate of a turbulent flow based on wall information only, represents a significant step in this direction.

1.6 Appendix: Tracking for 3D Channel Flow

These results appeared in an early version of the paper submitted to the *Journal of Fluid Mechanics*, and have since been superceeded by the results presented above. Since they are an an intermediate result, and were eventually replaced by the results presented in §1.4, they have been moved to an appendix.

To characterize the dependence of the EnKF on the length scales parameterizing the localization (see §1.3.1), we first studied the problem of *tracking*. That is, the estimation problem when the estimator is initialized with good (but not perfect) initial estimate of a fully-developed turbulent flow. An initial condition was arbitrarily chosen for the truth simulation using a random, instantaneous flow-field from a simulation performed off-line. Each ensemble member was then initialized with a random, divergence-free perturbation to the initial condition of the truth simulation. Thus the statistical properties of the resulting initial condition for the estimator (used for all four tracking cases) were ideal: an ensemble mean equal to the truth, and a Gaussian distribution of perturbed ensemble members.

The estimation results summarized in Figures 1.5 and 1.6 depict the error norm and correlation, respectively, of four data assimilation cases, which differ only in the choice of localization parameters. They indicate that, when the three parameters are tuned appropriately, $\mathbf{Q} = \text{diag}\{\,[50,\, 50,\, 25]\,\}$, the EnKF *synchronizes* with the turbelent flow indefinitely. This result is remarkable; nothing like it has ever been achieved on this canonical problem.

The estimation results demonstrate a high sensitivity to the localization parameter in the wall-normal direction, and a low sensitivity to localization parameters in the stream/span-wise directions. From an information perspective, this suggests that knowledge of the relationship between measurements at the wall and velocity field just outside the wall-induced turbulent structures is critical in the estimation process. Even when the three parameters are tuned such that the EnKF does not synchronize to the turbulent flow, we note that the error norm and correlation of the estimate are still quite good, as discussed further below.

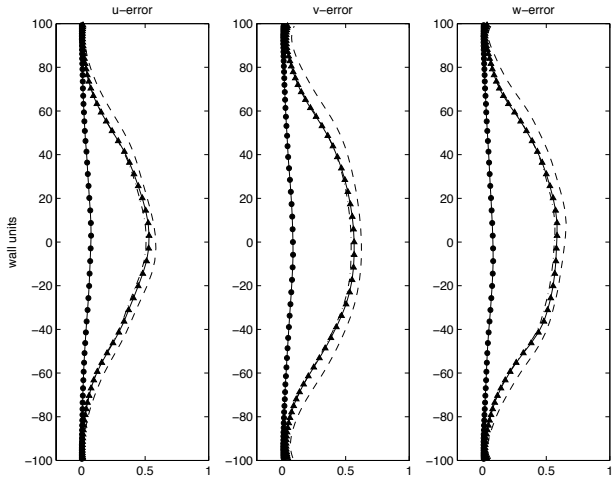


Figure 1.5: The effect of the localization parameters in (1.27) on the error norm, in an attempt to *track* (that is, to estimate, starting from accurate but not perfect initial conditions) the turbulent flowfield fluctuations with the EnKF. The $-\blacktriangleleft-$ lines are with localization length scales of $\mathbf{Q} = \mathrm{diag}([75, 25, 37.5])$ viscous units in the streamwise, wall-normal, and spanwise directions, respectively; the $-.-$ lines are with localization length scales of $\mathbf{Q} = \mathrm{diag}([50, 25, 25])$, the $---$ lines are with localization length scales of $\mathbf{Q} = \mathrm{diag}([25, 25, 12.5])$, and the $-\bullet-$ lines are with localization length scales of $\mathbf{Q} = \mathrm{diag}([50, 50, 25])$.

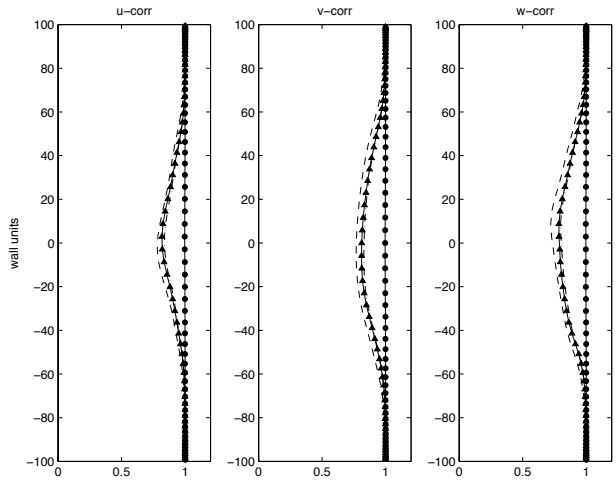


Figure 1.6: As in Figure 1.5, but reporting the correlation.

1.7 Acknowledgments

The authors gratefully acknowledge the generous financial support of the National Security Education Center (NSEC) at Los Alamos National Laboratory (LANL), the many helpful conversations with Frank Alexander (LANL), and the Triton Cluster support team at the San Diego Supercomputing Center at University of California, San Diego.

This chapter, in part, is a reprint of the material as it was accepted by the Journal of Fluid Mechanics 2011. (Colburn, Cessna & Bewley (2011*a*), "State estimation in wall-bounded flow systems. Part 3. The Ensemble Kalman Filter." *Journal of Fluid Mechanics*.) The dissertation author was the primary investigator and author of this paper.

Chapter 2

Gradient-based optimization methods for sensor & actuator placement in LTI systems

2.1 Abstract

This paper develops efficient techniques for calculating gradient information which may be used to optimize the placement of sensors & actuators of a given precision for the effective estimation and control of high-dimensional discretizations of infinite-dimensional linear time-invariant (LTI) systems. The necessary gradients are determined in this setting via adjoint analyses which quantify the effects of small variations of the observation and control operators. The approach can be modified appropriately to fit a variety of specific objectives within the Linear Quadratic Gaussian (LQG) estimation/control framework. Unlike other work in this area, we work directly with the covariance of the estimation error $\mathbf{P}$, rather than working with the Fischer information matrix $\mathbf{M}$, which is, in a sense, a best-case estimate of $\mathbf{P}^{-1}$ that neglects the impact of the state disturbances on the evolution of the state estimation error. The method is tested by optimizing the placement of two sensors and two actuators in a 1D complex Ginzburg-Landau system.

2.2 Introduction

Sensor and actuator placement techniques for state estimation and control problems have broad applications in environmental studies, finance, and engineering. Significant applications include: actuator placement in vibration control of flexible structures (Hiramoto *et al.*, 2000), sensor placement in environmental applications (Majumdar *et al.*, 2002*a*), explosion detection and contaminant plume tracking (Zhang *et al.*, 2011*a*), and estimation/control of chemical production/mixing procedures (Alonso *et al.*, 2004). Although it is clear that the fidelity of the estimator in such problems is strongly dependent on the sensor locations chosen in addition to the sensor precisions used, there has been surprisingly little work on the development of rigorous, model-based, numerically-tractable algorithms for optimizing sensor placement in such high-dimensional systems.

In low-dimensional systems, this class of problems may be addressed effectively using the linear matrix inequality (LMI) formulation of Li *et al.* (2009). This approach does not actually address the placement of sensors and actuators of fixed precision, but rather assigns a cost associated with the precision of the sensors and actuators used (in preassigned locations), then optimizes these precisions in order to minimize this cost. By so doing, the problem is made convex. One can thus formulate and solve a problem that begins with a large number of sensors and actuators of undetermined precisions in candidate positions, then perform an LMI-based optimization of the precisions of the sensors and actuators used to minimize the cost. One can then, in an ad hoc fashion, eliminate those sensors and actuators with the smallest impact on the problem at hand, and reoptimize the precisions of the sensors and actuators that remain. The scaling of the complexity of this formulation with dimension of the system under consideration is poor; the present gradient-based formulation might thus prove to be superior for the optimization of the placement of sensors & actuators of a given precision in high-dimensional discretizations of infinite-dimensional systems.

The majority of existing model-based sensor placement approaches considered in the literature are based on minimizing various measures of the Fisher information matrix, which is particularly convenient when considering problems of

this sort (for an introduction, see, e.g., Cover & Thomas 2006, p. 392). In short, if a random variable x depends on an unknown parameter θ, then the Fisher information is a characterization of the "information" provided about θ via samples of x. In particular, if the probability density function (PDF) of the random variable x depends only weakly on θ, then the derivative of the conditional PDF $p(x;\theta)$ with respect to θ will be relatively small over all possible values of x, whereas if this dependence is strong, this derivative will, at least for some x, be large. Normalizing this derivative by the value of the conditional PDF itself, we define the *score* $v(x,\theta)$ as the sensitivity of the conditional PDF with respect to variation of θ for given values of x and θ; that is,

$$v(x,\theta) = \frac{\frac{\partial}{\partial\theta}p(x;\theta)}{p(x;\theta)} = \frac{\partial}{\partial\theta}\ln p(x;\theta).$$

Note that the expected value of the score $v(x,\theta)$ over all possible values of x is zero:

$$\mathrm{E}\{v(x,\theta)\} = \int_{-\infty}^{\infty} v(x,\theta)\, p(x;\theta)\, dx = \int_{-\infty}^{\infty} \frac{\partial}{\partial\theta}p(x;\theta)\, dx = \frac{\partial}{\partial\theta}\underbrace{\int_{-\infty}^{\infty} p(x;\theta)\, dx}_{=1} = 0.$$

It is thus the variance of the score $v(x,\theta)$ that is useful in characterizing the overall magnitude of the sensitivity of the conditional PDF with respect to variation of θ; the *Fisher information* $M(\theta)$ is thus defined in this (scalar) case as the variance of the score:

$$M(\theta) = \mathrm{E}\{[v(x,\theta)]^2\} = \int_{-\infty}^{\infty} [v(x,\theta)]^2\, p(x;\theta)\, dx.$$

Extension of this concept to vectors of random variables $\mathbf{x}$ and vectors of unknown parameters $\boldsymbol{\theta}$ is straightforward, and leads immediately to the *Fisher information matrix* (FIM) $\mathbf{M}(\boldsymbol{\theta})$ via the appropriate outer product:

$$\mathbf{M}(\boldsymbol{\theta}) = \mathrm{E}\{[\mathbf{v}(\mathbf{x},\boldsymbol{\theta})]^H[\mathbf{v}(\mathbf{x},\boldsymbol{\theta})]\} = \int [\mathbf{v}(\mathbf{x},\boldsymbol{\theta})]^H[\mathbf{v}(\mathbf{x},\boldsymbol{\theta})]\, p(\mathbf{x};\boldsymbol{\theta})\, d\mathbf{x},$$

where $\mathbf{v}(\mathbf{x},\boldsymbol{\theta}) = \frac{\partial}{\partial\boldsymbol{\theta}}\ln p(\mathbf{x};\boldsymbol{\theta})$ is considered to be a row vector and $[\cdot]^H$ denotes the conjugate transpose.

The inverse of the Fisher information matrix provides a lower bound of the estimation error covariance matrix $\mathbf{P}$ of the Kalman filter via the Cramér-Rao inequality

$$\mathbf{P} \geq \mathbf{M}^{-1}; \tag{2.1}$$

derivation of this important bound is given in Goodwin & Payne 1977, Theorem 1.3.1.

As mentioned previously, scalar measures of the Fisher information matrix are typically considered when optimizing sensor locations. Three common such measures are

- A-optimality (trace):

$$J_A(\mathbf{M}) = \text{trace}(\mathbf{M}^{-1}), \tag{2.2a}$$

- D-optimality (determinant):

$$J_D(\mathbf{M}) = -\ln\det(\mathbf{M}), \tag{2.2b}$$

- E-optimality (max eigenvalue):

$$J_E(\mathbf{M}) = \lambda_{max}(\mathbf{M}^{-1}). \tag{2.2c}$$

Uciński (2005) reviews these cost functions and summarizes the impact of this choice on the overall optimization problem: "A D-optimum design minimizes the volume of the uncertainty ellipsoid for the estimates. An E-optimum design minimizes the length of the largest axis of the same ellipsoid. An A-optimum design suppresses the average variance of the estimates."

Work on the sensor placement problem has focused heavily on the three cost functions listed above. For example, Faulds & King (2000) considered A-optimality measures of the FIM to analyze (but not optimize) a model-free method for placing sensors in the domain of the 2D heat equation using Centroidal Voronoi Tesselations (CVT). Similarly, Martínez & Bullo (2006) found methods for minimizing D-optimality measures of the FIM in target tracking problems. These

two formulations are particularly attractive because they can be solved in a distributed framework (Cortes *et al.*, 2004; Bullo & Cortes, 2004; Kwok & Martinez, 2010). However attractive these distributed formulations are, computational experiments in 2D Navier-Stokes systems (Zhang *et al.*, 2011*a*) indicate that centralized formulations which optimize sensor vehicle trajectories specifically targeting regions of high estimation uncertainty in a model predictive control setting generally provide superior estimator performance than CVT-based formulations. Porat & Nehorai (1996) propose a source-seeking estimation/tracking algorithm based on A-optimality measures of the FIM which seek to optimize sensor locations for estimating a contaminant source location via a relatively inefficient global search over feasible future measurement locations. Because this method scales poorly with problem size, they augment the algorithm by calculating gradients of the FIM at select locations within the feasible set for each sensor; the authors propose this optimization in a receding horizon setting, where measurement locations eventually converge to stationary points in the domain. In the robust setting, Flaherty *et al.* (2006) use E-optimality measures of the FIM to estimate parameter values in models of biological systems.

Although optimization of sensor locations via consideration of the FIM is convenient, it is somewhat unfortunate that Cramér-Rao only relates the FIM to a *lower* bound on the quantity of interest (that is, the covariance of the estimation error); though it is evident that the covariance of the estimation error and the inverse of FIM are somehow related, optimizing the sensor locations based on the FIM in fact provides no gaurantees (upper bounds) on the resulting covariance of the estimation error. In fact, the covariance of the estimation error of the Kalman Filter only approaches the lower bound provided by the FIM in the limit that the state disturbances of the system model are made small. To illustrate this, Taylor (1979) puts the Kalman Filter for continuous time systems with discrete time measurements in context with the FIM and the Cramér-Rao bound. Assuming a continuous-time state transition matrix satisfying the differential equation

$$\frac{\mathrm{d}\mathbf{\Phi}(t,t_0)}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{\Phi}(t,t_0), \tag{2.3a}$$

defining $\mathbf{\Phi}_{k+1,k} = \mathbf{\Phi}(t_{k+1}, t_k)$, and taking the initial condition $\mathbf{\Phi}(t,t) = \mathbf{I}$, Taylor (1979) showed that the FIM for the discrete-time Kalman Filter can be written in the form

$$\mathbf{M}(t_k) = \mathbf{\Phi}_{k,k-1}^{-H} \mathbf{M}(t_{k-1}) \mathbf{\Phi}_{k,k-1}^{-1} + \mathbf{H}^H \mathbf{V}^{-1} \mathbf{H}. \tag{2.4}$$

Comparing this result with the propagation of the discrete-time information filter[1],

$$\mathbf{P}_{k|k}^{-1} = \left(\mathbf{\Phi}_{k,k-1} \mathbf{P}_{k-1|k-1} \mathbf{\Phi}_{k,k-1}^H + \mathbf{W} \right)^{-1} + \mathbf{H}^H \mathbf{V}^{-1} \mathbf{H}, \tag{2.5}$$

where $\mathbf{W}$ is the covariance of the state disturbances added to the state evolution equation, it is clear that, in the limit in which the state evolution is deterministic ($\mathbf{W} \to 0$), the covariance of the state estimation error approaches the lower bound predicted by the Crámer-Rao inequality.

Adjoint-based variational methods provide a powerful and broadly extensible framework for optimization problems of this sort, and scale well to high-dimensional discretizations of infinite-dimensional systems. Via successive linearization, they can also be used to optimize problems outside of the somewhat restrictive linear/quadratic setting. Note that adjoint-based optimizations have been applied broadly for shape design in aerodynamic systems (Jameson *et al.*, 1998; Giles & Pierce, 2000), adaptive grid refinement for error reduction in CFD simulations (Giles, 1998), and a host of other practical applications. However, such methods have not yet been extended to optimize sensor distributions in fluid systems; the present work seeks to fill this void. Furthermore, only a few investigations have used adjoint methods to evaluate the sensitivity of solutions to Riccati equations. Specifically, De Farias *et al.* (2001, Appendix A) propose a strategy similar to that used here for extracting gradients while performing optimizations of LMIs, and Kenney & Hewer (1990) examined how solutions to Riccati equations change as a result of modeling errors in the actuation/measurement covariance matrix.

The remainder of this paper develops various adjoint-based methods for minimizing relavent scalar measures in the control and estimation problems: §2.3

[1]The information filter is not to be confused with the FIM. The information filter is the propagation and update of the information matrix, which is defined as the inverse of the Kalman Filter covariance $\mathbf{P}$; see, Anderson & Moore (1979).

presents this analysis for continuous-time systems, §2.4 performs the equivalent discrete-time analysis, and §2.5 presents an application of the continuous-time theory to the 1D complex Ginzburg-Landau equation.

2.3 Continuous-time Analyses

Consider a continuous-time Linear Time Invariant (LTI) system described by

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{A}\mathbf{x} + \mathbf{B}(\mathbf{q_a})\mathbf{u} + \mathbf{w}, \tag{2.6a}$$

$$\mathbf{y} = \mathbf{C}(\mathbf{q_s})\mathbf{x} + \mathbf{v}, \tag{2.6b}$$

where[2] $\mathbf{x}(t) \in \mathbb{C}^n$ is the state, $\mathbf{u}(t) \in \mathbb{C}^\ell$ is the control, $\mathbf{y}(t) \in \mathbb{C}^m$ is the measurement, $\mathbf{w}(t) \in \mathbb{C}^n$ is the state disturbance, $\mathbf{v}(t) \in \mathbb{C}^m$ is the measurement noise, $\mathbf{q_a}(t)$ parameterizes the actuator positions, and $\mathbf{q_s}(t)$ parameterizes the sensor positions. For simplicity below, we make the standard modeling assumptions that $\mathbf{w}(t)$ and $\mathbf{v}(t)$ are *uncorrelated, zero-mean, white continuous-time random processes* with spectral densities $\mathbf{W} \geq \mathbf{0}$ and $\mathbf{V} > \mathbf{0}$ respectively. The (bounded) functional dependence of the operators $\mathbf{B}$ and $\mathbf{C}$ on the actuator and sensor positions $\mathbf{q_a}$ and $\mathbf{q_s}$ is emphasized explicitly above, but is suppressed below for notational clarity. In the discussion below, we first treat the optimization of the sensor positions $\mathbf{q_s}$, then the optimization of the actuator positions $\mathbf{q_a}$; the development of the gradient information necessary to optimize both actuator and sensor positions simultaneously follows similarly, and is discussed further in §2.3.3.

[2] For generality, our formulations are developed in §2.3 and §2.4 and tested in §2.5 on complex systems; it is trivial to restrict these formulations to the (more typical) setting of real systems.

2.3.1 Computing a gradient with respect to the sensor positions

Via standard (continuous-time) Kalman-Bucy filter theory, the best linear unbiased estimate $\hat{\mathbf{x}}(t)$ of the system (2.6) is given by

$$\frac{\mathrm{d}\hat{\mathbf{x}}(t)}{\mathrm{d}t} = (\mathbf{A} - \mathbf{LC})\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}(t)\mathbf{y}(t), \qquad \mathbf{L}(t) = \mathbf{P}(t)\mathbf{C}^H\mathbf{V}^{-1}, \tag{2.7}$$

where the covariance $\mathbf{P}(t) = \mathrm{E}\{\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^H(t)\} \geq 0$ of the estimation error $\tilde{\mathbf{x}}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$ evolves forward in time from given initial conditions $\mathbf{P}(0)$ according to the differential Riccati equation (DRE)

$$\frac{\mathrm{d}\mathbf{P}(t)}{\mathrm{d}t} = \mathbf{A}\,\mathbf{P}(t) + \mathbf{P}(t)\,\mathbf{A}^H + \mathbf{W} - \mathbf{L}(t)\,\mathbf{V}\,\mathbf{L}^H(t). \tag{2.8}$$

This evolution equation for $\mathbf{P}(t)$ reveals that, as the estimator (2.7) evolves in time, the estimation error covariance $\mathbf{P}(t)$ is driven larger by the unmodelled state disturbances $\mathbf{w}(t)$ in the system (2.6), and is driven smaller by the feedback term $\mathbf{L}(t)\mathbf{y}(t)$ in the estimator (2.7). Given that the DRE (2.8) marches to a finite value at $t = T$ as $T \to \infty$ [that is, that the system (2.6) is *detectable*], the infinite-horizon solution of the DRE (2.8) may be computed directly by setting $\mathrm{d}\mathbf{P}/\mathrm{d}t = 0$, thus transforming the DRE (2.8) into the continuous-time algebraic Riccati equation (CARE)

$$\mathbf{0} = \mathbf{A}\,\mathbf{P} + \mathbf{P}\,\mathbf{A}^H + \mathbf{W} - \mathbf{L}\,\mathbf{V}\,\mathbf{L}^H, \qquad \mathbf{L} = \mathbf{P}\,\mathbf{C}^H\,\mathbf{V}^{-1}. \tag{2.9a}$$

Closed-form solutions to a CARE such as (2.9a) are generally unavailable, and thus iterative methods based on the Schur decomposition of a $2n \times 2n$ Hamiltonian matrix are typically used to solve them (Kailath, 1980).

The matrices $\mathbf{A}$, $\mathbf{W}$, and $\mathbf{V}$ in this LTI formulation are assumed to be given. The remaining matrix which affects $\mathbf{P}$ in (2.9a) is $\mathbf{C}$, which is, in turn, a function of the sensor positions $\mathbf{q_s}$. Thus, an optimization problem may be posed to minimize some measure of $\mathbf{P}$ in the infinite-horizon problem (2.9a) with respect to the (stationary) sensor locations $\mathbf{q_s}$. In particular, we will seek the optimal $\mathbf{q_s}$

which minimizes the cost

$$J(\mathbf{q_s}) = \text{trace}(\mathbf{P}); \tag{2.9b}$$

alternative formulations based on different measures of $\mathbf{P}$ are considered in §2.3.3.

The gradient-based optimization problem we develop here focuses on the selection of $\mathbf{q_s}$ to minimize the cost $J(\mathbf{q_s})$ in (2.9b), where J is related to $\mathbf{q_s}$ via solution of the CARE (2.9a). We first select an initial $\mathbf{q_s}$ essentially arbitrarily, subject only to the technical condition that (2.6) be detectable. Local gradients of the cost J with respect to the sensor positions $\mathbf{q_s}$ are then iteratively optimized via a standard gradient-based minimization algorithm. The algebraically difficult step of this formulation is the efficient computation of the gradient $\nabla_{\mathbf{q_s}} J$.

A simple approach to computing the necessary gradient in this problem might be to apply a finite difference method or the (more accurate) complex-step derivative method to each component of each of the m sensor locations individually, solve a perturbation problem for each, then synthesize the results to assemble the gradient (see, e.g., Chen & Rowley, 2010). An accurate and significantly more computationally efficient approach to compute the gradient is to instead perform a single adjoint computation, as discussed in detail below.

Starting from an initial set of sensor locations $\mathbf{q_s}$ and the corresponding observation matrix $\mathbf{C}$, associated CARE solution $\mathbf{P}$, and cost J, consider the following Taylor series expansion of (2.9b) about $\mathbf{q_s}$:

$$J(\mathbf{q_s} + \mathbf{q_s'}) = J(\mathbf{q_s}) + (\nabla_{\mathbf{q_s}} J)^H \mathbf{q_s'} + \ldots = J(\mathbf{q_s}) + J'(\mathbf{q_s}, \mathbf{q'}) + \ldots \tag{2.10}$$

The constraint given in (2.9a) implies that a small perturbation of $\mathbf{q_s}$ yields a small perturbation of J; this expansion can also be written explicitly as a function of $\mathbf{P'}$:

$$J(\mathbf{q_s} + \mathbf{q_s'}) = \text{trace}(\mathbf{P} + \mathbf{P'}) = \text{trace}(\mathbf{P}) + \text{trace}(\mathbf{P'}) \quad \Rightarrow \quad J'(\mathbf{q_s}, \mathbf{q'}) = \text{trace}(\mathbf{P'}). \tag{2.11}$$

The corresponding equations for the perturbation matrix $\mathbf{P}'$ are

$$\mathbf{A}\mathbf{P}' + \mathbf{P}'\mathbf{A}^H - \mathbf{P}'\mathbf{C}^H\mathbf{L}^H - \mathbf{L}\mathbf{C}\mathbf{P}' = \mathbf{P}(\mathbf{C}')^H\mathbf{L}^H + \mathbf{L}\mathbf{C}'\mathbf{P} \tag{2.12a}$$

where

$$\mathbf{C}' = \left(\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\mathbf{q_s}}\right)^H \mathbf{q}'_\mathbf{s}.$$

In general, the matrix $\mathbf{C}'$ is a contraction of the rank-3 tensor $\mathrm{d}\mathbf{C}/\mathrm{d}\mathbf{q_s}$ with a vector of sensor perturbations $\mathbf{q}'_\mathbf{s}$. For notational convenience, (2.12a) is written as two linear operations $\mathrm{U}(\mathbf{P}')$ and $\mathrm{V}(\mathbf{C}')$ such that

$$\mathrm{U}(\mathbf{P}') = \mathrm{V}(\mathbf{C}') \tag{2.12b}$$

where

$$\begin{aligned}\mathrm{U}(\mathbf{P}') &= \mathbf{A}\mathbf{P}' + \mathbf{P}'\mathbf{A}^H - \mathbf{P}'\mathbf{C}^H\mathbf{L}^H - \mathbf{L}\mathbf{C}\mathbf{P}',\\ \mathrm{V}(\mathbf{C}') &= \mathbf{P}(\mathbf{C}')^H\mathbf{L}^H + \mathbf{L}\mathbf{C}'\mathbf{P}.\end{aligned}$$

Comparing the right-hand-sides of (2.10) and (2.11) reveals that

$$(\nabla_{\mathbf{q_s}} J)^H \mathbf{q}'_\mathbf{s} = \mathrm{trace}(\mathbf{P}'). \tag{2.13}$$

The relationship between P' and q'_s can thus be used to compute the gradient. To proceed, define an appropriate matrix inner-product[3]

$$\langle \mathbf{X}, \mathbf{Z} \rangle = \Re[\mathrm{trace}(\mathbf{X}^H\mathbf{Z})] \tag{2.14}$$

(where $\Re[\cdot]$ and $\Im[\cdot]$ denote the real and imaginary imaginary part, respectively) along with a matrix adjoint variable $\mathbf{S}$ and an adjoint operator $\mathrm{U}^*(\cdot)$ defined such

[3]Note that, if all matrices are real, (2.14) and (2.16) simplify, and the conjugate operation may be dropped from (2.17a).

that

$$\langle\, \mathbf{S}, \mathrm{U}(\mathbf{P}')\,\rangle = \langle\, \mathrm{U}^*(\mathbf{S}), \mathbf{P}'\,\rangle \tag{2.15}$$
$$\Rightarrow \quad \mathrm{U}^*(\mathbf{S}) = \mathbf{A}^H\mathbf{S} + \mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{L}\mathbf{C} - \mathbf{C}^H\mathbf{L}^H\mathbf{S} = (\mathbf{A} - \mathbf{L}\mathbf{C})^H\mathbf{S} + \mathbf{S}(\mathbf{A} - \mathbf{L}\mathbf{C}).$$

Recognizing that the perturbation to the cost function (2.9b) may be expressed using (2.14), it follows from (2.12b) and (2.15) that, if $\mathrm{U}^*(\mathbf{S}) = \mathbf{I}$, then the first-order perturbation to the cost is exactly

$$\begin{aligned} J'(\mathbf{q_s}, \mathbf{q}') &= \mathrm{trace}(\mathbf{P}') = \langle\, \mathbf{I}, \mathbf{P}'\,\rangle = \langle\, \mathrm{U}^*(\mathbf{S}), \mathbf{P}'\,\rangle = \langle\, \mathbf{S}, \mathrm{U}(\mathbf{P}')\,\rangle = \langle\, \mathbf{S}, \mathrm{V}(\mathbf{C}')\,\rangle \\ &= \Re\left[\mathrm{trace}\left(2\mathbf{P}\mathbf{S}\mathbf{L}\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\mathbf{q_s}}\mathbf{q_s}'\right)\right] \\ &= \mathrm{trace}\left(\Re\left[2\mathbf{P}\mathbf{S}\mathbf{L}\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\mathbf{q_s}}\right]\Re[\mathbf{q_s}']\right) - \mathrm{trace}\left(\Im\left[2\mathbf{P}\mathbf{S}\mathbf{L}\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}\mathbf{q_s}}\right]\Im[\mathbf{q_s}']\right). \end{aligned} \tag{2.16}$$

Thus, the gradient of the cost function (2.9b) with respect to the i'th element of the sensor positions vector q_s^i can be extracted:

$$\nabla_{q_s^i} J = \mathrm{trace}\overline{\left(2\mathbf{P}\mathbf{S}\mathbf{L}\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}q_s^i}\right)}, \tag{2.17a}$$

where the overbar denotes the complex conjugate, and where $\mathbf{S}$ satisfies the associated continuous-time algebraic Lyapunov equation (CALE)

$$(\mathbf{A} - \mathbf{L}\mathbf{C})^H\mathbf{S} + \mathbf{S}(\mathbf{A} - \mathbf{L}\mathbf{C}) = \mathbf{I}. \tag{2.17b}$$

2.3.2 Computing a gradient with respect to the actuator positions

Following an analogous approach as that developed above for the sensor placement problem, we now consider the corresponding actuator placement problem. Standard continuous-time optimal control theory applied to the linear system

(2.6) establishes that the cost function

$$J = \frac{1}{2}\int_0^T \left[\mathbf{x}^H(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^H(t)\mathbf{R}\mathbf{u}(t)\right] \mathrm{d}t + \frac{1}{2}\mathbf{x}^H(T)\mathbf{Q}_T\mathbf{x}(T) \tag{2.18}$$

is minimized by the full-state feedback control policy

$$\mathbf{u}(t) = -\mathbf{K}(t)\mathbf{x}(t), \qquad \mathbf{K}(t) = \mathbf{R}^{-1}\mathbf{B}^H\mathbf{Y}(t), \tag{2.19}$$

where the "cost-to-go" matrix $\mathbf{Y}(t) \geq 0$ evolves backward in time from the terminal condition $\mathbf{Y}(T) = \mathbf{Q}_T$ according to the DRE [cf. (2.8)]

$$-\frac{\mathrm{d}\mathbf{Y}(t)}{\mathrm{d}t} = \mathbf{A}^H\,\mathbf{Y}(t) + \mathbf{Y}(t)\,\mathbf{A} + \mathbf{Q} - \mathbf{K}^H(t)\,\mathbf{R}\,\mathbf{K}(t). \tag{2.20}$$

We identify $\mathbf{Y}(t)$ as a "cost-to-go" matrix because it can be shown that

$$\begin{aligned} J(\tau) &= \frac{1}{2}\int_\tau^T \left[\mathbf{x}^H(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^H(t)\mathbf{R}\mathbf{u}(t)\right] \mathrm{d}t + \frac{1}{2}\mathbf{x}^H(T)\mathbf{Q}_T\mathbf{x}(T) \\ &= \frac{1}{2}\mathbf{x}^H(\tau)\mathbf{Y}(\tau)\mathbf{x}(\tau). \end{aligned} \tag{2.21}$$

Given that the DRE (2.20) marches to a finite value at $t = 0$ as $T \to \infty$ [that is, that the system (2.6) is *stabilizable*], the infinite-horizon solution of the DRE (2.20) may be computed directly by setting $\mathrm{d}\mathbf{Y}/\mathrm{d}t = 0$, thus transforming the DRE (2.20) into the CARE [cf. (2.9a)]

$$\mathbf{0} = \mathbf{A}^H\,\mathbf{Y} + \mathbf{Y}\,\mathbf{A} + \mathbf{Q} - \mathbf{K}^H\mathbf{R}\mathbf{K}, \qquad \mathbf{K} = \mathbf{R}^{-1}\,\mathbf{B}^H\,\mathbf{Y}. \tag{2.22a}$$

The matrices $\mathbf{A}$, $\mathbf{Q}$, and $\mathbf{R}$ in this LTI formulation are assumed to be given. The remaining matrix which affects $\mathbf{Y}$ in (2.22a) is $\mathbf{B}$, which is, in turn, a function of the actuator positions $\mathbf{q_a}$. Note that (2.21) evalutated at $\tau = 0$ in the infinite-horizon limit $T \to \infty$ implies that the original cost metric in (2.18) is minimized when the eigenvalues of the symmetric matrix $\mathbf{Y}$ are minimized. Towards this end,

the following cost function may be proposed [cf. (2.9b)]

$$\min_{\mathbf{q_a}} J = \text{trace}(\mathbf{Y}). \tag{2.22b}$$

The actuator positions $\mathbf{q_a}$ can now be iteratively optimized following an essentially identical analysis to that presented in §2.3.1. The resulting expression for the gradient [cf. (2.17a)] is

$$\nabla_{q_a^i} J = \text{trace}\overline{\left(2\mathbf{KTY}\frac{\mathrm{d}\mathbf{B}}{\mathrm{d}q_a^i}\right)}, \tag{2.23a}$$

where the matrix adjoint $\mathbf{T}$ satisfies the associated CALE [cf. (2.17b)]

$$(\mathbf{A}-\mathbf{BK})\mathbf{T}+\mathbf{T}(\mathbf{A}-\mathbf{BK})^H = \mathbf{I}. \tag{2.23b}$$

2.3.3 Discussion

As mentioned previously, the gradient-based optimizations discussed in §2.3.1 and §2.3.2 first select initial sensor and actuator positions essentially arbitrarily, subject only to the technical conditions that (2.6) be detectable and stabilizable[4]. Local gradients of a relevant cost J with respect to the sensor and actuator positions are then successively calculated and used to efficiently (but locally) optimize the sensor and actuator locations via a standard minimization algorithm such as steepest descent or the nonquadratic conjugate gradient method. The algebraically difficult step is the efficient computation of the necessary gradients, which has been shown in both cases to arise in a straightforward fashion from the standard CARE for the estimation or control problem, together with an associated CALE to compute an adjoint matrix upon which the required gradient is based.

[4]Lauga & Bewley (2003) showed that detectability and stabilizability are lost gradually in systems of this short when the sensors and actuators are moved outside of the physical domain of interest (that is, where the significant dynamics of the open-loop PDE system take place), which can lead to numerical problems when using finite-precision arithmetic. It is thus advisable to chose reasonable initial placements of the sensors and actuators, well within the regions of significant dynamics of the open-loop system.

From the analyses performed in §2.3.1 and §2.3.2, it is clear that the RHS forcing in (2.17b) and (2.23b) is determined solely by the definition of the cost function. Alternative cost functions can also easily be considered, such as those appearing in (2.2a)-(2.2c) with the estimation error covariance $\mathbf{P}$, which the quantitiy of interest here, replacing the inverse of the FIM. Taking $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$ as the eigenvalues of $\mathbf{P}$, a summary outlining the key results is sufficient to clarify[5]:

- A-optimality (trace):

$$\begin{aligned} & J_A = \text{trace}(\mathbf{P}) = \lambda_1 + \ldots + \lambda_n \\ \Rightarrow \quad & J'_A = \text{trace}(\mathbf{P'}) = \langle \mathbf{I}, \mathbf{P'} \rangle \\ \Rightarrow \quad & \text{U}^*(\mathbf{S}) = \mathbf{I}. \end{aligned} \tag{2.24a}$$

- D-optimality (determinant):

$$\begin{aligned} & J_D = -\ln\det(\mathbf{P}^{-1}) = \ln\det(\mathbf{P}) \\ \Rightarrow \quad & J'_D = \text{trace}(\mathbf{P}^{-1}\mathbf{P'}) = \langle \mathbf{P}^{-1}, \mathbf{P'} \rangle \\ \Rightarrow \quad & \text{U}^*(\mathbf{S}) = \mathbf{P}^{-1}. \end{aligned} \tag{2.24b}$$

- E-optimality (max eigenvalue, λ_1, with corresponding eigenvector $\mathbf{r}_1$)

$$\begin{aligned} & J_E = \lambda_1(\mathbf{P}) \\ \Rightarrow \quad & J'_E = \text{trace}(\mathbf{r}_1\mathbf{r}_1^H\mathbf{P'}) = \langle \mathbf{r}_1\mathbf{r}_1^H, \mathbf{P'} \rangle \\ \Rightarrow \quad & \text{U}^*(\mathbf{S}) = \mathbf{r}_1\mathbf{r}_1^H. \end{aligned} \tag{2.24c}$$

[5]Matrix identities from Petersen & Pedersen (2008) and Horn & Johnson (1990) are used in the analyses leading to (2.24b) and (2.24c), respectively.

Another metric of interest is the square of the Frobenius norm of $\mathbf{P}$:

$$\begin{aligned} J_F = \|\mathbf{P}\|_F^2 = \mathrm{trace}(\mathbf{PP}) = \sum_i \sum_j |p_{ij}|^2 = \lambda_1^2 + \ldots + \lambda_n^2 \\ \Rightarrow \quad J_F' = \mathrm{trace}(2\mathbf{PP}') = \langle 2\mathbf{P}, \mathbf{P}' \rangle \\ \Rightarrow \quad \mathrm{U}^*(\mathbf{S}) = 2\mathbf{P}. \end{aligned} \tag{2.24d}$$

It has been shown that the adjoint method of computing the gradient in this class of problems is readily extensible to a broad range of different cost functions, with the only difference between the various cases being the RHS forcing of the associated adjoint.

The perhaps most notable extension to consider is the cost function associated with the $\mathcal{H}_2$ control problem. This problem is well known and has been studied extensively (see, e.g., Kwakernaak & Sivan (1972), Zhou & Doyle (1998), and Hassibi *et al.* (1999)). We proceed by appending the state equation with an additional output $\mathbf{z}$ identifying the states of interest in the control problem

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{Ax} + \mathbf{B}(\mathbf{q_a})\mathbf{u} + \mathbf{B}_1\mathbf{w}, \tag{2.25a}$$

$$\mathbf{y} = \mathbf{C}(\mathbf{q_s})\mathbf{x} + \mathbf{v}, \tag{2.25b}$$

$$\mathbf{z} = \mathbf{C}_1\mathbf{x}. \tag{2.25c}$$

Following the $\mathcal{H}_2$ approach, an estimate $\hat{\mathbf{x}}$ of the state $\mathbf{x}$ is first developed, based on the measurements $\mathbf{y}$, as discussed in §2.3.1, then a full state feedback controller $\mathbf{u} = -\mathbf{K}\mathbf{x}$ is developed, as discussed in §2.3.2. These two components are then connected by making the control feedback depend on the state estimate, $\mathbf{u} = -\mathbf{K}\hat{\mathbf{x}}$, rather than the state itself. Consolidating the disturbance vector $\mathbf{d} = [\mathbf{w}; \mathbf{v}]$ for the purpose of analysis, a new cost function can be written to characterize the $\mathcal{H}_2$-norm of the closed-loop transfer function:

$$J_{\mathcal{H}_2} = ||T_{zd}||_2^2 = \mathrm{trace}(\mathbf{C}_1\mathbf{P}\mathbf{C}_1^H) + \mathrm{trace}(\mathbf{V}^{-1}\mathbf{CPYPC}^H), \tag{2.26a}$$

$$= \mathrm{trace}(\mathbf{B}_1^H\mathbf{YB}_1) + \mathrm{trace}(\mathbf{R}^{-1}\mathbf{B}^H\mathbf{YPYB}), \tag{2.26b}$$

where $\mathbf{P}$ and $\mathbf{Y}$ are the solutions to the CAREs (2.9a) and (2.22a), respectively. [As briefly mentioned above, this sensor/actuator placement problem was addressed by Chen & Rowley (2010), but will be reconsidered here since the gradient calculation is different.] By averaging (2.26a) and (2.26b) and then performing perturbation analysis on the averaged equations, the perturbation of $J_{\mathcal{H}_2}$ can be written

$$\begin{aligned} J'_{\mathcal{H}_2} &= \operatorname{trace}\left(\mathbf{PYPC}^H\mathbf{V}^{-1}\mathbf{C}'\right) + \operatorname{trace}\left(\mathbf{R}^{-1}\mathbf{B}^H\mathbf{YPYB}'\right) \\ &\quad + \frac{1}{2}\operatorname{trace}\left(\left[\mathbf{C}_1^H\mathbf{C}_1 + \mathbf{C}^H\mathbf{V}^{-1}\mathbf{CPY} + \mathbf{YPC}^H\mathbf{V}^{-1}\mathbf{C} + \mathbf{YBR}^{-1}\mathbf{B}^H\mathbf{Y}\right]\mathbf{P}'\right) \\ &\quad + \frac{1}{2}\operatorname{trace}\left(\left[\mathbf{B}_1\mathbf{B}_1^H + \mathbf{BR}^{-1}\mathbf{B}^H\mathbf{YP} + \mathbf{PYBR}^{-1}\mathbf{B}^H + \mathbf{PC}^H\mathbf{V}^{-1}\mathbf{CP}\right]\mathbf{Y}'\right) \\ &= \left\langle \left[\mathbf{PYPC}^H\mathbf{V}^{-1}\right]^H, \mathbf{C}' \right\rangle + \left\langle \left[\mathbf{R}^{-1}\mathbf{B}^H\mathbf{YPY}\right]^H, \mathbf{B}' \right\rangle \\ &\quad + \frac{1}{2}\left\langle \left[\mathbf{C}_1^H\mathbf{C}_1 + \mathbf{C}^H\mathbf{V}^{-1}\mathbf{CPY} + \mathbf{YPC}^H\mathbf{V}^{-1}\mathbf{C} + \mathbf{YBR}^{-1}\mathbf{B}^H\mathbf{Y}\right]^H, \mathbf{P}' \right\rangle \\ &\quad + \frac{1}{2}\left\langle \left[\mathbf{B}_1\mathbf{B}_1^H + \mathbf{BR}^{-1}\mathbf{B}^H\mathbf{YP} + \mathbf{PYBR}^{-1}\mathbf{B}^H + \mathbf{PC}^H\mathbf{V}^{-1}\mathbf{CP}\right]^H, \mathbf{Y}' \right\rangle \end{aligned} \tag{2.27}$$

Thus, via a slight change of the RHS forcing of the adjoint CALEs (2.17b) and (2.23b),

$$\begin{aligned} (\mathbf{A}-\mathbf{LC})^H\mathbf{S} + \mathbf{S}(\mathbf{A}-\mathbf{LC}) &= \mathbf{C}_1^H\mathbf{C}_1 + \mathbf{YBR}^{-1}\mathbf{B}^H\mathbf{Y} \\ &\quad + \mathbf{C}^H\mathbf{V}^{-1}\mathbf{CPY} + \mathbf{YPC}^H\mathbf{V}^{-1}\mathbf{C}, \end{aligned} \tag{2.28}$$

$$\begin{aligned} (\mathbf{A}-\mathbf{BK})\mathbf{T} + \mathbf{T}(\mathbf{A}-\mathbf{BK})^H &= \mathbf{B}_1\mathbf{B}_1^H + \mathbf{PC}^H\mathbf{V}^{-1}\mathbf{CP} \\ &\quad + \mathbf{BR}^{-1}\mathbf{B}^H\mathbf{YP} + \mathbf{PYBR}^{-1}\mathbf{B}^H, \end{aligned} \tag{2.29}$$

the gradient can be reëxpressed as a function of both the sensor and actuator positions

$$\nabla_{q_s^i} J = \operatorname{trace}\overline{\left(\mathbf{P}(\mathbf{Y}+\mathbf{S})\mathbf{L}\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}q_s^i}\right)}, \tag{2.30}$$

$$\nabla_{q_a^i} J = \operatorname{trace}\overline{\left(\mathbf{K}(\mathbf{P}+\mathbf{T})\mathbf{Y}\frac{\mathrm{d}\mathbf{B}}{\mathrm{d}q_a^i}\right)}. \tag{2.31}$$

2.4 Discrete-Time Analyses

We now present the discrete-time analogs of the derivations given in §2.3 in the continuous-time case, as there are some subtle differences. Consider a discrete-time linear system described by

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}(\mathbf{q_a})\mathbf{u}_k + \mathbf{w}_k, \tag{2.32a}$$

$$\mathbf{y}_k = \mathbf{H}(\mathbf{q_s})\mathbf{x}_k + \mathbf{v}_k, \tag{2.32b}$$

where $\mathbf{x}_k \in \mathbb{C}^n$, $\mathbf{u}_k \in \mathbb{C}^\ell$, $\mathbf{y}_k \in \mathbb{C}^m$, $\mathbf{w}_k \in \mathbb{C}^n$, $\mathbf{v}_k \in \mathbb{C}^m$ are the discrete-time equivalents of the corresponding quantities in (2.6). Similarly, $\mathbf{q_a}$ and $\mathbf{q_s}$ parametrize the locations of sensors and actuators at each timestep k. We again make the standard modeling assumptions that $\mathbf{w}_k$ and $\mathbf{v}_k$ are *uncorrelated, zero-mean, white continuous-time random processes* with covariance $\mathbf{W} \geq \mathbf{0}$ and $\mathbf{V} > \mathbf{0}$ respectively. The (bounded) functional dependence of the operators $\mathbf{G}$ and $\mathbf{H}$ on the actuator and sensor positions $\mathbf{q_a}$ and $\mathbf{q_s}$ is again emphasized explicitly above, but is suppressed below for notational clarity.

2.4.1 Computing a gradient with respect to the sensor positions

Via standard (discrete-time) Kalman filter theory, the best linear unbiased estimate of the system (2.32) is given by a two-step update [cf. (2.7)]

$$\text{time update:} \qquad \hat{\mathbf{x}}_{k+1|k} = \mathbf{F}\hat{\mathbf{x}}_{k|k} + \mathbf{G}\mathbf{u}_k, \tag{2.33a}$$

$$\text{measurement update:} \qquad \hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{L}_{k+1}(\mathbf{y}_{k+1} - \mathbf{H}\hat{\mathbf{x}}_{k+1|k}), \tag{2.33b}$$

where

$$\mathbf{L}_{k+1} = \mathbf{P}_{k+1|k}\mathbf{H}^H(\mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}^H + \mathbf{V})^{-1}. \tag{2.33c}$$

The discrete-time estimation error covariance obeys a similar two-step update known as the Riccati difference equation (RDE) [cf. (2.8)]

$$\text{time update:} \qquad \mathbf{P}_{k+1|k} = \mathbf{F}\mathbf{P}_{k|k}\mathbf{F}^H + \mathbf{W}, \tag{2.34a}$$

$$\text{measurement update:} \qquad \mathbf{P}_{k+1|k+1} = (\mathbf{I} - \mathbf{L}_{k+1}\mathbf{H})\mathbf{P}_{k+1|k}. \tag{2.34b}$$

Using this standard notation for the discrete-time estimation setting, the notation $\hat{\mathbf{x}}_{k|j}$ denotes the maximum likelihood estimate of $\mathbf{x}$ at time t_k given all measurements up to and including time t_j, and $\mathbf{P}_{k|j}$ denotes the covariance corresponding to this estimate. In particular, $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ are often called the *prior estimate* and *prior covariance*, whereas $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$ are often called the *posterior estimate* and *posterior covariance*.

As in §2.3.1, we now consider the minimization of the trace of the infinite-horizon covariance matrix. Because the discrete-time Kalman filter is characterized as a two-step process, there are two possible choices to make as to whether the covariance should be minimized before or after the measurement update, as shown below. Both formulations are presented; which is more appropriate to use in practice is application dependant.

The *infinite-horizon prior covariance* of (2.34) is computed by substituting (2.34b) into (2.34a) and applying (2.33c), then defining $\mathbf{P}_- = \mathbf{P}_{k+1|k} = \mathbf{P}_{k|k-1}$, thus transforming the RDE (2.34) into the prior form of the discrete-time algebraic Riccati equation (DARE) [cf. (2.9a)]

$$\mathbf{P}_- = \mathbf{F}\mathbf{P}_-\mathbf{F}^H + \mathbf{W} - \mathbf{F}\mathbf{P}_-\mathbf{H}^H(\mathbf{H}\mathbf{P}_-\mathbf{H}^H + \mathbf{V})^{-1}\mathbf{H}\mathbf{P}_-\mathbf{F}^H \tag{2.35a}$$

Alternatively, the *infinite-horizon posterior covariance* of (2.34) is computed by substituting (2.34a) into (2.34b) and (2.33c) and combining, then defining $\mathbf{P}_+ = \mathbf{P}_{k+1|k+1} = \mathbf{P}_{k|k}$ and applying the Matrix-Inversion Lemma

$$(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1} = \mathbf{D}^{-1} + \mathbf{D}^{-1}\mathbf{C}(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}\mathbf{B}\mathbf{D}^{-1},$$

thus transforming the RDE (2.34) into the posterior form of the DARE

$$\mathbf{P}_+^{-1} = (\mathbf{F}\mathbf{P}_+\mathbf{F}^H + \mathbf{W})^{-1} + \mathbf{H}^H\mathbf{V}^{-1}\mathbf{H}. \tag{2.35b}$$

The matrices $\mathbf{F}$, $\mathbf{W}$, and $\mathbf{V}$ in this LTI formulation are assumed to be given. The remaining matrix which affects $\mathbf{P}_-$ in (2.35a), and $\mathbf{P}_+$ in (2.35b), is $\mathbf{H}$, which is a function of the sensor positions $\mathbf{q_s}$. Thus, an optimization problem may be posed to minimize some measure of $\mathbf{P}_-$, or $\mathbf{P}_+$, with respect to the (stationary) sensor locations $\mathbf{q_s}$.

Prior covariance optimization

We first seek the optimal $\mathbf{q_s}$ which minimizes the cost [cf. (2.9b)]

$$J_- = \operatorname{trace}(\mathbf{P}_-), \tag{2.36}$$

subject to (2.35a). The associated first-order perturbations are

$$\begin{aligned}
J'_- &= \operatorname{trace}(\mathbf{P}'_-) = \langle \mathbf{I}, \mathbf{P}'_- \rangle, \\
\mathbf{P}'_- &= \mathbf{F}(\mathbf{I} - \mathbf{L}\mathbf{H})\mathbf{P}'_-(\mathbf{I} - \mathbf{L}\mathbf{H})^H\mathbf{F}^H \\
&\qquad + \mathbf{F}\mathbf{L}(\mathbf{H}'\mathbf{P}_-\mathbf{H}^H + \mathbf{H}\mathbf{P}_-(\mathbf{H}')^H)\mathbf{L}^H\mathbf{F}^H - \mathbf{F}\mathbf{P}_-(\mathbf{H}')^H\mathbf{L}^H\mathbf{F}^H - \mathbf{F}\mathbf{L}\mathbf{H}'\mathbf{P}_-\mathbf{F}^H \\
\mathbf{H}' &= \left(\frac{\mathrm{d}\mathbf{H}}{\mathrm{d}\mathbf{q_s}}\right)^H \mathbf{q_s}'.
\end{aligned}$$

The above relations are derived in a manner analogous to the continuous-time case. Note that the perturbation of $(\mathbf{H}\mathbf{P}_-\mathbf{H}^H + \mathbf{V})^{-1}$ is determined leveraging the identity $(\mathbf{\Phi}^{-1})' = -\mathbf{\Phi}^{-1}\mathbf{\Phi}'\mathbf{\Phi}^{-1}$ (see Petersen & Pedersen (2008)), thus leading to

$$\big((\mathbf{H}\mathbf{P}_-\mathbf{H}^H + \mathbf{V})^{-1}\big)' = (\mathbf{H}\mathbf{P}_-\mathbf{H}^H+\mathbf{V})^{-1}(\mathbf{H}'\mathbf{P}_-\mathbf{H}^H+\mathbf{H}\mathbf{P}'_-\mathbf{H}^H+\mathbf{H}\mathbf{P}_-(\mathbf{H}')^H)(\mathbf{H}\mathbf{P}_-\mathbf{H}^H+\mathbf{V})^{-1}.$$

Defining an adjoint matrix $\mathbf{S}_-$ and the inner product (2.14) and performing the necessary rearrangements in a manner analogous to the continuous-time case, one

ultimately arrives at the gradient of J_-,

$$\nabla_{q_s^i} J_- = \overline{\mathrm{trace}\left(2\mathbf{P}_-(\mathbf{I}-\mathbf{LH})^H\mathbf{F}^H\mathbf{S}_-\mathbf{FL}\frac{\mathrm{d}\mathbf{H}}{\mathrm{d}q_s^i}\right)}, \tag{2.37a}$$

where $\mathbf{S}_-$ satisfies the associated discrete-time algebraic Lyapunov equation (DALE)

$$(\mathbf{I}-\mathbf{LH})^H\mathbf{F}^H\mathbf{S}_-\mathbf{F}(\mathbf{I}-\mathbf{LH})-\mathbf{S}_- = \mathbf{I}. \tag{2.37b}$$

Posterior covariance optimization

We now seek the optimal $\mathbf{q_s}$ which minimizes the cost [cf. (2.9b)]

$$J_+ = \mathrm{trace}(\mathbf{P}_+), \tag{2.38}$$

subject to (2.35b). The associated first-order perturbations in this case are

$$-\mathbf{P}_+^{-1}\mathbf{P}'_+\mathbf{P}_+^{-1} = -(\mathbf{FP}_+\mathbf{F}^H+\mathbf{W})^{-1}\mathbf{FP}'_+\mathbf{F}^H(\mathbf{FP}_+\mathbf{F}^H+\mathbf{W})^{-1}+(\mathbf{H}^H)'\mathbf{V}^{-1}\mathbf{H}+\mathbf{H}^H\mathbf{V}^{-1}\mathbf{H}'.$$

Using a similar procedure as before, one ultimately arrives at the graident of J_+,

$$\nabla_{q_s^i} J_+ = \overline{\mathrm{trace}\left(2\mathbf{S}_+\mathbf{H}^H\mathbf{V}^{-1}\frac{\mathrm{d}\mathbf{H}}{\mathrm{d}q_s^i}\right)}, \tag{2.39a}$$

where $\mathbf{S}_+$ satisfies the DALE

$$\mathbf{P}_+\mathbf{F}^H(\mathbf{FP}_+\mathbf{F}^H+\mathbf{W})^{-1}\mathbf{S}_+(\mathbf{FP}_+\mathbf{F}^H+\mathbf{W})^{-1}\mathbf{FP}_+-\mathbf{S}_+ = \mathbf{P}_+^2, \tag{2.39b}$$

The gradients in (2.37a) and (2.39a) are slightly different, because the cost functions they minimize are different. Noting (2.34a), it is evident that

$$J_- = \mathrm{trace}(\mathbf{P}_-) = \mathrm{trace}(\mathbf{FP}_+\mathbf{F}^H)+\mathrm{trace}(\mathbf{W}),$$

whereas $J_+ = \mathrm{trace}(\mathbf{P}_+)$. The gradients and optimal solutions of these two formulations thus coincide only if $\mathbf{F}^H\mathbf{F} = \mathbf{I}$.

2.4.2 Computing a gradient with respect to the actuator positions

In a final analysis analogous to those of the previous sections, we now consider the corresponding discrete-time actuator placement problem. Standard discrete-time optimal control theory applied to the linear system (2.32) establishes that the cost function

$$J = \frac{1}{2}\sum_{k=1}^{N-1}\left[\mathbf{x}_k^H\mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^H\mathbf{R}\mathbf{u}_k\right] + \frac{1}{2}\mathbf{x}_N^H\mathbf{Q}_N\mathbf{x}_N. \tag{2.40}$$

is minimized by the full-state feedback control policy

$$\mathbf{u}_k = -\mathbf{K}_k\mathbf{x}_k, \qquad \mathbf{K}_k = \left(\mathbf{R} + \mathbf{G}^H\mathbf{Y}_{k+1}\mathbf{G}\right)^{-1}\mathbf{G}^H\mathbf{Y}_{k+1}\mathbf{F}, \tag{2.41}$$

where the matrix $\mathbf{Y}_k \geq 0$ evolves backward in time from the terminal condition $\mathbf{Y}_N = \mathbf{Q}_N$ according to the RDE [cf. (2.8)]

$$\mathbf{Y}_k = \mathbf{F}^H\mathbf{Y}_{k+1}\mathbf{F} - \mathbf{F}^H\mathbf{Y}_{k+1}\mathbf{G}\left(\mathbf{R} + \mathbf{G}^H\mathbf{Y}_{k+1}\mathbf{G}\right)^{-1}\mathbf{G}^H\mathbf{Y}_{k+1}\mathbf{F}^H + \mathbf{Q}. \tag{2.42}$$

Similar to the continuous-time case, we identify $\mathbf{Y}_k$ as the "cost-to-go" matrix because it can be shown [cf. (2.21)] that

$$J(\kappa) = \frac{1}{2}\sum_{k=\kappa}^{N-1}\left[\mathbf{x}_k^H\mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^H\mathbf{R}\mathbf{u}_k\right] + \frac{1}{2}\mathbf{x}_N^H\mathbf{Q}_N\mathbf{x}_N \tag{2.43}$$

$$= \frac{1}{2}\mathbf{x}_\kappa^H\mathbf{Y}_\kappa\mathbf{x}_\kappa. \tag{2.44}$$

Given that the RDE (2.42) marches to a finite value at $k = 0$ as $N \to \infty$ [that is, that the system (2.32) is *stabilizable*], the infinite-horizon solution may be computed directly by setting $\mathbf{Y} = \mathbf{Y}_k = \mathbf{Y}_{k+1}$, thus transforming the RDE (2.42) into a DARE [cf. (2.9a)]

$$\mathbf{Y} = \mathbf{F}^H\mathbf{Y}\mathbf{F} - \mathbf{F}^H\mathbf{Y}\mathbf{G}(\mathbf{R} + \mathbf{G}^H\mathbf{Y}\mathbf{G})^{-1}\mathbf{G}^H\mathbf{Y}\mathbf{F} + \mathbf{Q}. \tag{2.45a}$$

As in §2.3.2, for $\kappa = 0$ in the infinite-horizon limit $N \to \infty$, the cost matrix (2.40) is minimized when the eigenvalues of $\mathbf{Y}$ are minimized. With this in mind, the following cost function may be considered [cf. (2.9b)]

$$\min_{\mathbf{q_a}} J = \mathrm{trace}(\mathbf{Y}). \tag{2.45b}$$

The steps for gradient calculation via perturbation analysis are essentially identical to those presented previously. The resulting expression for the gradient [cf. 2.17a] is

$$\nabla_{q_a^i} J = \mathrm{trace}\overline{\left(2\mathbf{KT}(\mathbf{F}-\mathbf{GK})^H\mathbf{Y}\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}q_a^i}\right)} \tag{2.46a}$$

where the matrix adjoint $\mathbf{T}$ must satisfy the associated DALE [cf. 2.17b]

$$(\mathbf{F}-\mathbf{GK})\mathbf{T}(\mathbf{F}-\mathbf{GK})^H - \mathbf{T} = \mathbf{I}. \tag{2.46b}$$

2.5 Application to the complex Ginzburg-Landau equation

The 1D complex Ginzburg-Landau (CGL) system (Chomaz *et al.*, 1987; Roussopoulos & Monkewitz, 1996) shares some interesting dynamic features of 3D Navier-Stokes (NS) systems. Notable similarities include transient energy growth (due to non-normality of the system eigenvectors) and extensively-studied stability characteristics (including well-identified thresholds between stability, convective instability, and global instability). For this reason, and its relative computational simplicity, CGL systems are a useful 1D PDE testbed for estimation and control strategies being developed for ultimate application in 3D NS systems.

The linear CGL equation for a flow perturbation variable $\phi(\xi, t)$ may be written

$$\frac{\partial\phi}{\partial t} = \left(-U\frac{\partial}{\partial\xi} + \mu(\xi) + \gamma\frac{\partial^2}{\partial\xi^2}\right)\phi \tag{2.47}$$

where U, $\mu(\xi)$, γ are complex coefficients which parameterize the advection, amplification, and diffusion properties of the flow, respectively, and ξ denotes the

streamwise coordinate of the system. This flexible parameterization has been tuned to match a variety of physical phenomena; for example, Roussopoulos & Monkewitz (1996) tuned the parameters to model vortex shedding behind a circular cylinder. A recent review of the CGL model by Bagheri *et al.* (2009) surveys several such studies of this system.

In the results presented below, the parameters were selected to coincide with the convectively unstable case mentioned by Bagheri *et al.* (2009), with $U = 2 + 0.2i$, $\mu(\xi) = 0.38 - 0.01\xi^2/2$, $\gamma = 1 - i$. The resulting variable-coefficient PDE has a locally unstable domain with $\mu(\xi) > 0$ for all $\xi \in (-8.72, 8.72)$.

Bagheri *et al.* (2009) also provide a convenient codebase for discretization and simulation of the CGL equation using a collocation approach based on a Hermite polynomial expansion. Following this approach, the state $\phi(\xi, t)$ in (2.47) is considered as a linear combination of n orthogonal polynomials[6] defined on $\xi \in (-\infty, \infty)$,

$$\phi(\xi, t) \triangleq \sum_{j=1}^{n} \hat{\phi}_j(t)\, H_j(\xi) \qquad \text{where} \qquad H_j(\xi) = (-1)^j\, e^{\xi^2}\, \frac{d^j(e^{-\xi^2})}{d\xi^j}.$$

The perturbation variable $\phi(\xi, t)$ may now be discretized on a set of n collocation points ξ_j, for $j = 1, \ldots, n$, and assembled as a state vector $\mathbf{x}$, where the n collocation points are selected as roots of $H_n(\xi)$. With this discretization, the transformation given above, and the relationships between the derivatives of the Hermite polynomials $H_n(\xi)$, it is straightforward to write the discretized system (2.47) in collocation form with the appropriate forcing and measurement variables added:

$$\frac{\partial \mathbf{x}}{\partial t} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \bar{\mathbf{B}}\bar{\mathbf{w}}, \tag{2.48}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{v}, \tag{2.49}$$

where the matrix $\mathbf{A}$ approximates the spatially-varying linear operator on the

[6]Note that the Hermite polynomials are orthogonal on $\xi \in (-\infty, \infty)$ using the weighting function $w(\xi) = e^{-\xi^2}$.

RHS of (2.47), $\mathbf{B}$ models the effect of the control inputs $\mathbf{u}$ on the system near the actuators located at $\xi = q_a^j$ for $j = 1, \ldots, n_a$, and $\mathbf{C}$ models the measurements $\mathbf{y}$ of the system taken from the sensors located at $\xi = q_s^j$ for $j = 1, \ldots, n_s$. The random vectors $\bar{\mathbf{w}}$ and $\mathbf{v}$ are independent and normally distributed with covariances $\bar{\mathbf{W}}$ and $\mathbf{V}$ respectively. The matrix $\bar{\mathbf{B}}$ models the effect of the disturbance inputs $\bar{\mathbf{w}}$ applied to the system near $\xi = q_d^j$ for $j = 1, \ldots, n_d$; note that it is straightforward to cast this system in the standard continuous-times state-space form given in (2.6) by considering a new disturbance vector $\mathbf{w}$ with covariance $\mathbf{W} = \bar{\mathbf{B}}\bar{\mathbf{W}}\bar{\mathbf{B}}^H$.

In the results presented below, we actuate the system with $n_a = 1$ or 2 actuators (at locations q_a^j that we will optimize), we sense the system with $n_s = 1$ or 2 sensors (at locations q_s^j that we will optimize), and we disrupt the system with $n_d = 1$ disturbance (at $q_d^1 = -11.0$); the corresponding matrices are all chosen to approximate narrow Gaussians in space:

$$
\begin{aligned}
[\mathbf{B}]_{i,j} &= \exp(-(q_a^j - \xi_i)^2/2\sigma^2), \\
[\mathbf{C}]_{i,j} &= m_i \exp(-(q_s^j - \xi_i)^2/2\sigma^2), \\
[\bar{\mathbf{B}}]_{i,1} &= \exp(-(q_d^1 - \xi_i)^2/2\sigma^2),
\end{aligned}
$$

where the width of the Gaussians used in the simulations reported below is $\sigma^2 = 1/2$, and where a trapezoidal integration weighting factor m_i is used in the definition of $\mathbf{C}$,

$$
m_i = \begin{cases} (\xi_2 - \xi_1)/2 & i = 1, \\ (\xi_{i+1} - \xi_{i-1})/2 & 1 < i < n, \\ \xi_n - \xi_{n-1} & i = n, \end{cases}
$$

so that the sum of the elements on any row of $\mathbf{C}$ approximates the integral of the corresponding Gaussian, independent of the sensor locations q_s^j. By selecting a parameterization of this sort, the input and output operators represent sensors and actuators of a given sensitivity, it is only their *locations* that change when q_a^j and q_s^j are modified.

Before analyzing the influence of measurements and control on the statistics of the estimation error and the statistics of the disturbed system, it is important

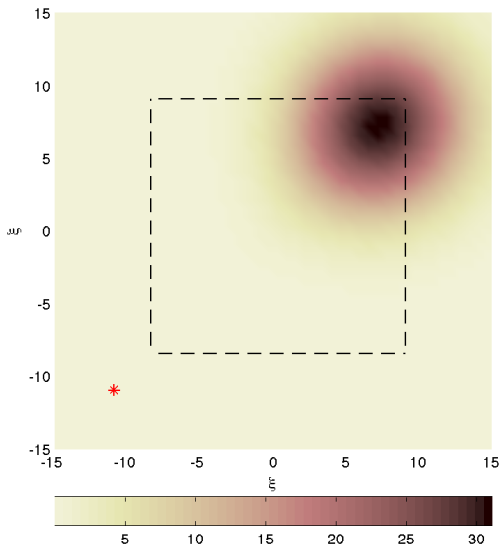


Figure 2.1: The modulus of the covariance of the state itself in the disturbed CGL system [that is, $\mathrm{E}\{\mathbf{x}(t)\mathbf{x}^H(t)\}$; see also Bagheri *et al.* (2009), Figure 16]. Note that this coincides with the covariance $\mathbf{P} = \mathrm{E}\{\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^H(t)\}$ of the state estimation error when no measurement information is used. The red star indicates the location of the disturbance forcing, $q_w^1 = -11.0$, and the dashed box indicates the region of local instability, $\mu(\xi) > 0$.

to consider first the statistics of the disturbed CGL system itself. Figure 2.1 thus depicts the modulus of the covariance of the state itself, as given by the solution to the infinite-horizon Lyapunov equation [that is, (2.8) with $\mathbf{L} = 0$], when the CGL system (2.47) is forced with the disturbances $\mathbf{w}$, but no measurements are used for state estimation. As expected, it is seen in these statistics that disruptions of the state tend to grow as they convect through the locally unstable region of the domain and then decay after that; thus, the peak in these statistics is on the diagonal near $\xi = 8.72$.

2.5.1 Optimal sensor placement in the estimation problem

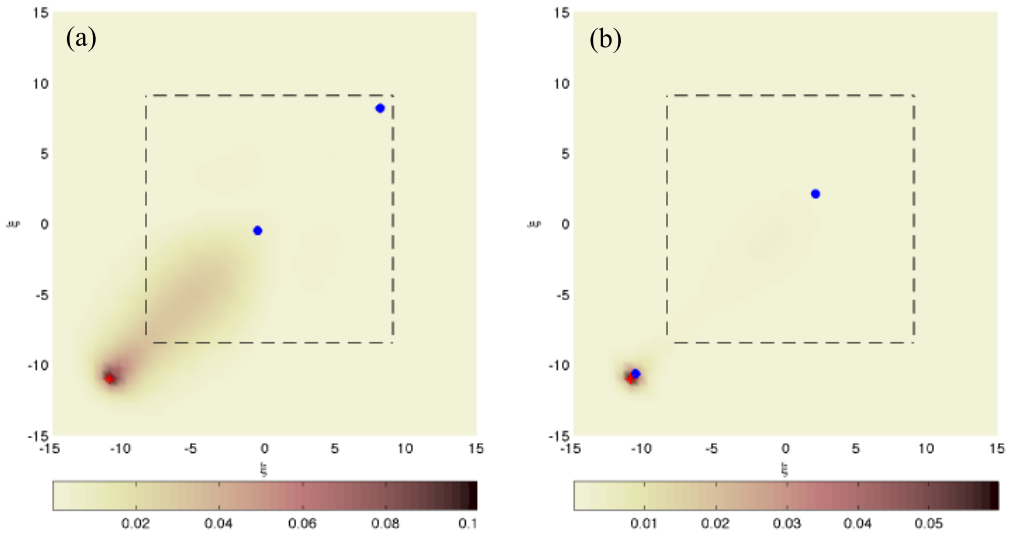
Finding the optimal placement of a single sensor $q_s^1 \in (-\infty, \infty)$ is a relatively straightforward task that may be achieved with a simple line search. The problem becomes more interesting when considering the simultaneous placement of two or more sensors with $q_s^i \in (-\infty, \infty)$, as the dimension of the optimization

space is increased and thus a gradient-based optimization approach is motivated. We thus consider now the optimization of the placement of two sensors in this system.

Figure 2.2 depicts the modulus of the covariance of the state estimation error for two different configurations of a pair of sensors. The sensor configuration in Fig. 2.2(a) was chosen heuristically, first placing one sensor at the location of maximum covariance in Figure 2.1, then placing the other sensor at the location of maximum state estimation error in the estimator that results. The sensor configuration in Fig. 2.2(b), on the other hand, was optimized using the algorithm described in §2.3.1. Figure 2.3 depicts of the full optimization surface as a function of the locations of the two sensors, indicating the path taken during the optimization process from the initial configuration at $\{q_s^1, q_s^2\} = \{12.60, -0.46\}$ [see Fig. 2.2(a)] to the optimized configuration at $\{q_s^1, q_s^2\} = \{2.10, -10.65\}$ [see Fig. 2.2(b)]. Note at each step that the path taken is downhill (normal to the isocontours), which is consistent with the fact that a steepest descent method was used in the optimization. Also, Figure 2.3 is symmetric about the diagonal $q_s^2 = q_s^1$, as the sensors in this case are identical; had sensors of different precision been used (i.e., $\mathbf{V} = \text{diag}\{[v_1, v_2]\}$ with $v_1 \neq v_2$), the symmetry in Figure 2.3 would be broken, and the gradient-based optimization algorithm would converge to a local minimum.

2.5.2 Optimal actuator placement in the full information control problem

The two-actuator placement problem is analogous to the two-sensor placement problem discussed in the previous section. Figure 2.4 depicts the full optimization surface as a function of the configuration of the two actuators, indicating the path taken during the optimization process to minimize the cost (2.9b). The initial configuration of the actuators in this case was taken simply as the optimized sensor configuration found in the previous section. As in the estimation problem, at each step the path taken is downhill (normal to the isocontours). Also, Figure 2.4 is symmetric about the diagonal $q_a^2 = q_a^1$, as the actuators in this case are identical. In this full-information setting, the optimized solution at $\{q_a^1, q_a^2\} = \{-4.66, 2.36\}$



(a) (b)

Figure 2.2: The modulus of the covariance of the state estimation error, $\mathbf{P} = \mathrm{E}\{\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^H(t)\}$, for two different placements of a pair of sensors (blue dots). Figure 2.2(a) uses a heuristic sequential method of placing the sensors (see text), thereby reducing the covariance depicted in Fig. 2.1 by nearly 4 orders of magnitude. Figure 2.2(b) uses a gradient-based method of optimizing the placement of both sensors simultaneously, as described in §2.3, thereby further reducing the covariance depicted in Fig. 2.2(a) by another order of magnitude.

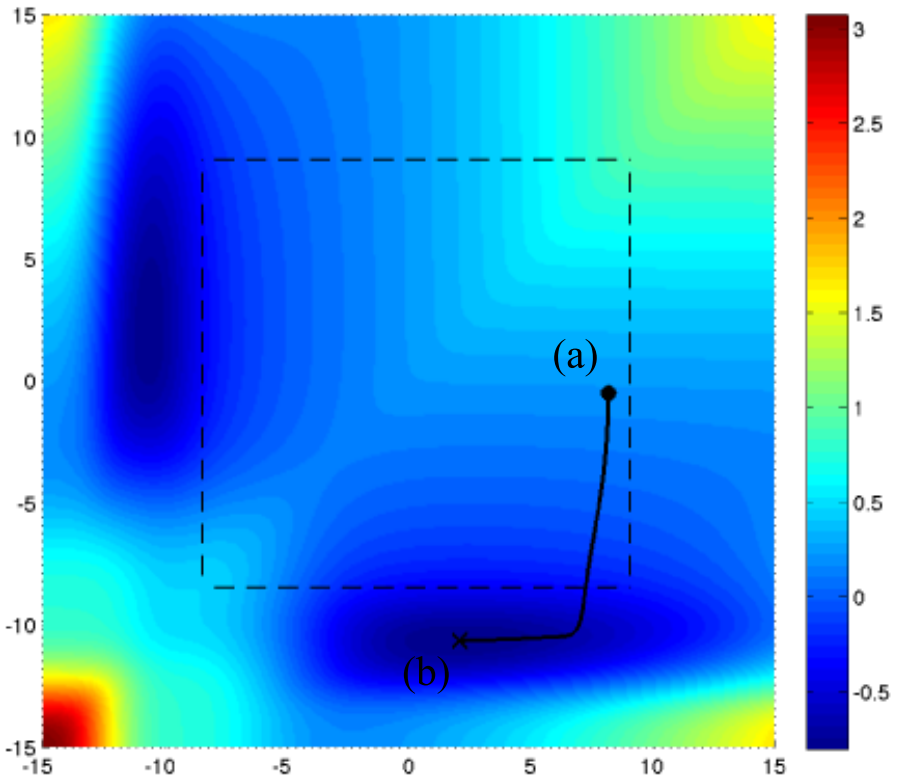


Figure 2.3: A $\log_{10}$ plot of the optimization surface for the two-sensor placement problem in the CGL system with the optimization path superposed, with the axes representing the positions of the two sensors. The optimization was initialized as depicted in Fig. 2.2(a), and converged to the solution depicted in Fig. 2.2(b). Since the two sensors are identical, the plot is symmetric across the diagonal.

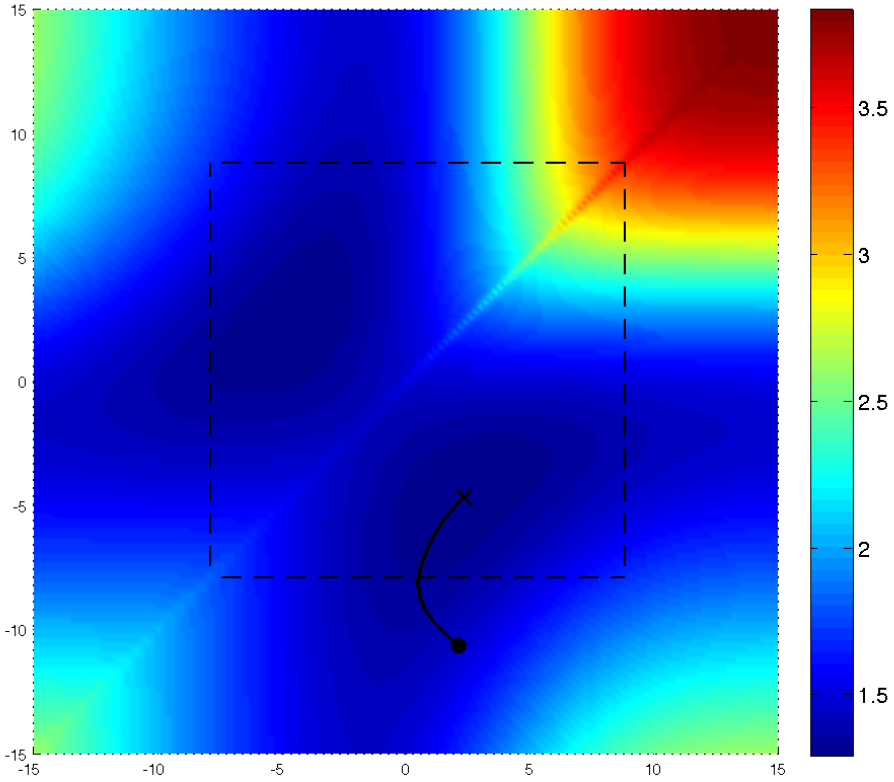


Figure 2.4: A $\log_{10}$ plot of the optimization surface for the full-information two-actuator placement problem in the CGL system with the optimization path superposed, with the axes representing the positions of the two actuators.

places both actuators inside the locally unstable region, thus effectively leveraging the positive local amplification term of the CGL; this is in contrast with the optimized sensor configuration presented previously, in which the upstream sensor is actually placed outside the unstable domain, relatively close to where the disturbance is introduced into the system.

Further understanding of this result is given by Fig. 2.5, which depicts the diagonal of the "cost-to-go" matrix $\mathbf{Y}$ of the controlled CGL system. It is reasonable that the area of the three lobes of the optimized configuration are approximately equal, indicating essentially that contributions to the cost function are, effectively, evenly distributed over the physical domain when the actuator positions are properly optimized.

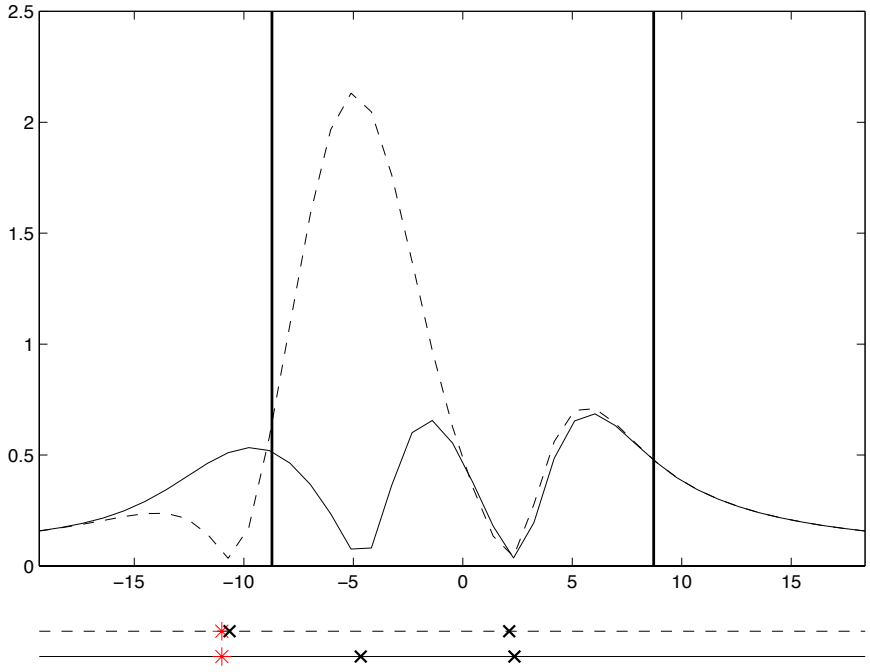


Figure 2.5: The diagonal of $\mathbf{Y}$ (the "cost-to-go" matrix) as a function of streamwise coordinate ξ in the CGL system. The integral of each line is the total cost (the trace of of the covariance matrix). The two lines represent solutions for different actuator configurations: the dashed line corresponds to the initial configuration of the optimization, $\{q_a^1, q_a^2\} = \{2.10, -10.65\}$, and the solid line corresponds to the solution of the optimization, $\{q_a^1, q_a^2\} = \{2.36, -4.66\}$. For clarity, below the plot, the disturbance location is denoted by $*$ and the actuator positions are denoted by $\times$ in the initial (dashed) and final (solid) configurations.

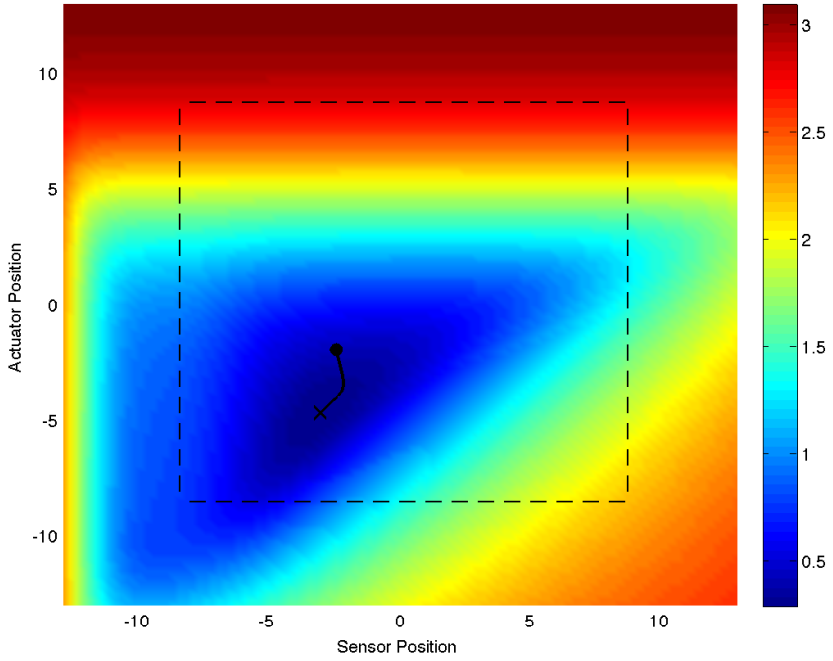
2.5.3 $\mathcal{H}_2$ optimal actuator/sensor placement

Though many control-oriented studies of the CGL system have appeared in the literature, only recently has the question of optimizing sensor and actuator placements in such problems been considered. In particular, Chen & Rowley (2010) found acuator/sensor configurations that minimize the H_2 norm in such problems by simultaneously optimizing both sensor and actuator positions. As intuition suggests, optimizing the sensor and actuators positions separately leads to reasonable but not optimal performance in the full H_2 problem.

As established in §2.3.3, the gradient-based procedure outlined in §2.3.1 and §2.3.2 may easily be extended to optimize sensor and actuator locations simultaneously in the full H_2 setting; results are depicted in Fig. 2.6. The optimization surface is depicted in Fig. 2.6(a) for the one-actuator, one-sensor $\mathcal{H}_2$ problem, and superposed is the path taken by the full optimization algorithm. Note again at each step that the path taken is downhill (normal to the isocontours), thus indicating the correctness of the gradient computation. The initial configuration, $\{q_s, q_a\} = \{-2.47, -1.94\}$, was generated by solving separately the optimal sensor placement problem for one sensor and the (full-information) optimal actuator placement problem for one actuator. The gradient-based method discussed in §2.3.3 was then used to find the optimized configuration $\{q_s, q_a\} = \{-3.08, -4.66\}$, as depicted in Fig. 2.6(b).

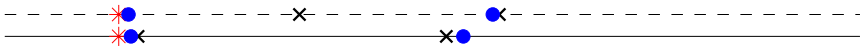
A similar procedure was performed for the two-actuator, two-sensor case, as depicted in Fig. 2.6(c), where the initial configuration (based on solving the actuator and sensor placement problems separately) and the optimized configuration $\{q_s^1, q_s^2, q_a^1, q_a^2\} = \{1.09, -10.57, -10.32, 0.49\}$ are compared side-by-side. Though this problem is four-dimensional and thus difficult to visualize, optimizations from the various random initial conditions tested all appear to converge to the same optimized configuration in this case.

The present gradient-based formulation appears to extend naturally to high-dimensional discretizations of various 2D and 3D Navier-Stokes systems, which is left for future studies.



(a) The optimization surface for the one-sensor, one-actuator $\mathcal{H}_2$ problem. Also shown is the path taken during the optimization process. The initial condition, $\bullet$, was generated by performing independent optimizations of the sensor and actuator placements. The combined gradient-based optimization formulation converges from this (suboptimal) inital guess to a (significantly improved) optimal solution, $\mathtt{x}$, of the combined problem.

(b) Initial condition (dashed) and optimal solution (solid) for the one-sensor, one-actuator $\mathcal{H}_2$ optimization problem depicted in Fig. 2.6(a). Note that, through the optimization, the sensor and actuator actually swap their relative positions



(c) As in Fig. 2.6(b), but for the two-sensor, two-actuator $\mathcal{H}_2$ problem.

Figure 2.6: Optimized sensor/actuator placements for the combined $\mathcal{H}_2$ estimation/control problem. The disturbance (indicated with $*$) is located at $q_w = -11.0$ in all instances. Actuators and sensors are denoted with the symbols $\mathtt{x}$, and $\bullet$, respectively.

2.6 Acknowledgments

The authors gratefully acknowledge the generous financial support of the National Security Education Center (NSEC) at Los Alamos National Laboratory (LANL), the many helpful conversations with Frank Alexander (LANL), and the Triton Cluster support team at the San Diego Supercomputing Center at University of California, San Diego.

This chapter, in full, has been submitted for publication of the material as it may appear in Journal of Fluid Mechanics, 2011. (Colburn, Zhang & Bewley (2011*b*), "Gradient-based optimization methods for sensor & actuator placement in LTI systems." Submitted for publication to *Journal of Fluid Mechanics*.) The dissertation author was the primary investigator and author of this paper.

Chapter 3

GEnKF: The Game-theoretic, Ensemble-Kalman Filter. A new class of Gaussian Sum Filters

3.1 Abstract

This paper explores Decision Theoretic Online-Learning (DTOL) and how it can be used in concert with standard probabilistic estimation methods (most notably, the Kalman Filter). Specifically, estimates of chaotic, high-dimensional systems (equivalent to the best available) are achieved with fewer communication and computational requirements. Estimation and forecasting of high-dimensional systems requires special attention because of the computational requirements. This paper contains a new algorithm which seeks computationally efficient methods for estimating such systems, while maintaining a high confidence in the solution.

3.2 Introduction

It is well known that the Kalman Filter (KF; see, Kalman, 1960) is the optimal filter for linear systems with Gaussian uncertainty in the initial state and process/measurement noise. Besides being relatively simple to implement, the KF

is attractive because it provides an *exact* posterior probability distribution (in fact, it should be interpreted as a common Bayesian update) for linear estimators. No equivalent solution exists for nonlinear filtering.[1]

As a result, many reasonable heuristics have been proposed to "close the gap" between linear and nonlinear filter theory. For example, the Extended Kalman Filter (EKF) uses the linearized governing equations to evolve the first and second moments of the distribution, and then performs KF updates when measurements are available. The Unscented (UKF) and Ensemble (EnKF) Kalman Filters use Monte-Carlo approximations for these propagation steps, and then build sample statistics for the KF update. And, while there may not be an immediate resolution to this void in estimation theory, there is clearly the need for improved heuristics as systems of interest become large (e.g., weather forecasting models track $\mathcal{O}(10^6)$ variables). In these situations memory or communication, or both, often effectively limit the implementable approximations to the Kalman solution.

This work is specifically interested in improving these heuristics for estimating high-dimensional system by reducing the communication required by the Ensemble Kalman Filter. Specifically, we consider the Gaussian Sum Filter (GSF; see, Sorenson & Alspach, 1971; Alspach & Sorenson, 1972) where the Bayesian weight assignment is replaced with the weight assignment determined by the Normal-Hedge algorithm (Chaudhuri *et al.*, 2009). The communication requirements in this setting are substantially reduced and an improvement from the EnKF to GEnKF is numerically demonstrated. Before being able to completely describe the Game-theoretic Ensemble Kalman Filter (GEnKF) we first briefly review the relevant components of sequential games and the Gaussian Sum Filter (GSF).

In existing literature only Prudêncio & Ludermir (2005) have used machine learning to combine explicit forecasts in dynamic systems. There the authors used a Multi-Layer Perceptron (MLP) network in combination with a "feature extractor" (a heuristic tool which uses linear trends, correlations, and turning points to forecast system dynamics) to improve forecast quality.

[1]The only exception is the use of the Fokker-Plank equation (see, e.g., Jazwinski, 1970, p. 164) in systems with only a few parameters to evolve the probability distribution in phase space (making Bayesian updates as measurement become available).

3.3 Decision Theoretic On-line Learning (DTOL)

The theory of sequential games provides a mechanism for selecting individuals who perform well with respect to their peers. It is the basic framework used for the prediction of sequences in computer science. Although game-theory is traditionally attributed to Von Neumann *et al.* (1953), in their seminal work on economic games, modern applications of multi-player sequential games to machine learning have been largely shaped by Blackwell & Girshick (1954), Blackwell (1956), Hannan (1957), and later substantially by Littlestone & Warmuth (1989), Vovk (1990), Freund & Schapire (1995) and Cesa-Bianchi *et al.* (1997). An attractive strength of the game theoretic formulation is the lack of assumptions regarding linearity and noise. That is to say, the bound determined for many sequential games are robust to non-Gaussian disturbances provided the observations contain sufficient information. A wonderful review of the DTOL framework and the associated contributions can be found in Cesa-Bianchi & Lugosi (2006).[2]

3.3.1 Review of DTOL framework

As mention above, DTOL is the framework for using "expert advice" in the prediction of sequences. In most problems, this framework can be broken into three parallel sub-processes: First, the *environment* is responsible for sequentially revealing an unknown sequence $y_1,\ y_2, \cdots y_t$ of elements from a known, and convex, outcome space $\mathcal{Y}$ which represent the "true outcomes" of a sequential process. Before each element in the sequence is revealed *experts* make a predictions[3] $\hat{y}_t^e$ at each round based on some unique apriori "knowledge" they have about the sequence and the sequence history $y_1,\ y_2, \cdots y_{t-1}$. After each expert makes a prediction the *forecaster/algorithm* looks at each experts past performance (which will be defined formally) and based on that performance determines how to use each experts

[2] Actually, Cesa-Bianchi & Lugosi (2006) review the more general field of "prediction with expert advice." The difference between this and the DTOL framework is subtle and briefly discussed in Vovk (2009).

[3] Incidentally, the most general DTOL formulation allows for the outcome space of the unknown sequential process and the experts to be different from each other (meaning that each expert might not have knowledge of all possible elements in the unknown sequence) but for the results presented below this more general formulation is not necessary.

advice.

At each time time-step, the performance is determined by a non-negative loss function $\ell(\hat{y}_t, y_t) \ : \ \mathcal{Y} \times \mathcal{Y} \rightarrow \mathcal{R}^+$, where large expert loss $\ell_t^e = \ell(\hat{y}_t^e, y_t)$ characterizes poor performance. It is essential to emphasize the assumption that this loss function is convex in it's first argument. By comparing each experts' loss with the algorithm's loss $\ell_t^A = \ell(\hat{y}_t^A, y_t)$ a relative performance metric can be defined which will allow the algorithm to assign weights to each expert. This relative performance metric is traditionally called the instantaneous regret $r_t^e = \ell_t^A - \ell_t^e$. (The term "regret" originates from the idea that a positive regret implies that the loss suffered by the algorithm was greater than the loss of a particular expert [mathematically speaking, $\ell_t^A > \ell_t^e$], thus, the algorithm feels "regret" for not following that experts advice.) By looking at the cumulative regret of each expert

$$R_t^e = \sum_{s=1}^{t} r_s^e = \sum_{s=1}^{t} \big(\ell(\hat{y}_s^A, y_s) - \ell(\hat{y}_s^e, y_s) \big) = L_t^A - L_t^e, \tag{3.1}$$

where $L_t = \sum_{s=1}^{t} \ell(\hat{y}_s^A, y_s)$ is used to denote the algorithm's cumulative loss and $L_t^e = \sum_{s=1}^{t} \ell(\hat{t}_t^e, y_t)$ is used to denote the cumulative loss of expert e, the algorithm can monitor integral measures of an expert's performance. The ultimate goal of the algorithm is to use the cumulative regret to determine how weight (trust) should be allocated between the experts.

By keeping the cumulative regret small the algorithm ensures good performance with respect to the defined loss function. Various performance bounds may be achieved; for example, the algorithm may achieve a vanishing per-round regret, that is,

$$\lim_{s \rightarrow \infty} \frac{1}{s} \Big(L_s^A - \min_{i=1,\cdots,N} L_s^i \Big) \rightarrow 0, \tag{3.2}$$

which is to say that the algorithm places majority weighting on the expert who performs the best. By definition the instantaneous regret for this situation is zero, and if summed indefinitely would result in the average cumulative regret converging to zero.

After the algorithm has assigned weight to a particular expert the logical approach for combining predictions is to take a weighted average of the experts.

Littlestone & Warmuth (1989) first proposed "The Weighted Majority Algorithm" as a strategy for consolidating advice for binary decision making.[4] In this paradigm the forecaster/algorithm uses the weights assigned at time $t-1$ to predict events at time t; that is,

$$\hat{y}_t^A = \frac{\sum_{e=1}^N w_{t-1}^e \hat{y}_t^e}{\sum_{e=1}^N w_{t-1}^e} \tag{3.3}$$

where $w_{t-1}^1, w_{t-1}^2, \cdots, w_{t-1}^N \geq 0$ are the weights assigned to the experts. Since cumulative regret is representative of some integral measure of historical performance, it makes sense to determine expert weighting at time t based on the cumulative regret at time $t-1$. To do this Cesa-Bianchi & Lugosi (2003) introduce a *potential function* which scales the effect of the cumulative regret. This potential function should assign substantial weight to an expert when the cumulative regret is large, and little weight to an expert with little cumulative regret. In their notation, this scaling is written as the derivative of a nonnegative, convex, and increasing function $\phi : \mathcal{R} \to \mathcal{R}$. The algorithm uses ϕ' to determine the weight $w_{t-1}^e = \phi'(R_{t-1}^e)$ assigned to each expert. Therefore, the prediction $\hat{y}_t^A$ can be rewritten as

$$\hat{y}_t^A = \frac{\sum_{e=1}^N \phi'(R_{t-1}^e)\hat{y}_t^e}{\sum_{e=1}^N \phi'(R_{t-1}^e)}. \tag{3.4}$$

As mentioned above, since the loss function is convex with respect to it's first argument, an important mathematical trick can be performed using Jensen's inequality (Jensen, 1906), so that the algorithm can appropriately choose it's action.

$$\ell(\,\hat{y}_t^A\,,\,y_t) = \ell\Big(\frac{\sum_{e=1}^N \phi'(R_{t-1}^e)\,\hat{y}_t^e}{\sum_{e=1}^N \phi'(R_{t-1}^e)}\,,\,y_t\Big) \;\leq\; \frac{\sum_{e=1}^N \phi'(R_{t-1}^e)\,\ell(\,\hat{y}_t^e\,,\,y_t)}{\sum_{e=1}^N \phi'(R_{t-1}^e)} \tag{3.5}$$

$$\sum_{e=1}^N \phi'(R_{t-1}^e)\,\ell(\,\hat{y}_t^A\,,\,y_t) - \sum_{e=1}^N \phi'(R_{t-1}^e)\,\ell(\,\hat{y}_t^e\,,\,y_t) \;\leq\; 0 \tag{3.6}$$

$$\sum_{e=1}^N \phi'(R_{t-1}^e)\,r_t^e \;\leq\; 0 \tag{3.7}$$

[4]Since their pioneering work it has been extended to more general applications.

Equation (3.7) satisfies *Blackwell's condition for asymptotic stability*, and is an essential when using potential function to prove the convergence of game-theoretic predictors. A rigorous discussion of Blackwell's condition is well outside the scope of this work, but for more information on it's application to the convergence of sequences and machine learning in general, the reader is referred to Blackwell (1956), Hart & Mas-Colell (2000, 2001), and Cesa-Bianchi & Lugosi (2003).

Cesa-Bianchi & Lugosi (2003) contains the analysis for some of the most widely used machine-learning algorithms. The main result of that papers bounds the cumulative regret at round t of the forecaster/algorithm with respect to the cumulative regret at round $t-1$ and the instantaneous regret at round t. For convenience, an alternative version of this theorem is summarized from Cesa-Bianchi & Lugosi (2006) before reviewing the bounds for an exponentially weighted forecaster using potential functions.[5]

Theorem 3.3.1 *Assume that a forecaster satisfies the Blackwell condition for a potential function* $\Phi(\mathbf{u}) = \psi\Big(\sum_{i=1}^{N} \phi(u_i)\Big)$*, where* $\psi : \mathcal{R}^+ \rightarrow \mathcal{R}^+$ *is an increasing, concave, and twice differentiable auxiliary function. Then, for all* $t = 1, 2, \cdots$

$$\Phi(\mathbf{R}_t) \leq \Phi(\mathbf{0}) + \frac{1}{2} \sum_{s=1}^{t} \mathrm{C}(\mathbf{r}_s) \tag{3.8}$$

where

$$\mathrm{C}(\mathbf{r}_s) = \sup_{\mathbf{u} \in \mathcal{R}^n} \psi'\Big(\sum_{i=1}^{N} \phi(u_i)\Big) \sum_{i=1}^{N} \phi''(u_i)\,(r_t^i)^2. \tag{3.9}$$

The reader is referred to Cesa-Bianchi & Lugosi (2006, p. 11) or Cesa-Bianchi & Lugosi (2003, p. 242) for a proof of this theorem. It should be briefly mentioned that, the auxiliary function in this theorem allows for tighter performance bounds to be achieved.

[5]For examples of how polynomial functions can be analyzed using this theorem the reader is referred to Cesa-Bianchi & Lugosi (2003, 2006).

3.3.2 Exponentially Weighted Forecasters

The class of exponential functions fits naturally into the framework described above. An exponentially weighted forecaster is one whose potential function takes the form

$$\phi\left(R_t^e\right) = \exp(\eta R_t^e). \tag{3.10}$$

The parameter η describes the "learning rate" of the forecaster, and conditions the speed that the algorithm converges to the bound. In fact, it is quite easy to show that an exponentially weighted forecaster with the potential function

$$\Phi(\mathbf{R}_t) = \sum_{e=1}^{N} \frac{1}{\eta} \ln\left|\exp(\eta R_t^e)\right| \tag{3.11}$$

can be bounded above by

$$\Phi(\mathbf{R}_t) \leq \Phi(\mathbf{R}_0) + \frac{1}{2}\sum_{s=1}^{t} \eta \max_e (r_t^e)^2. \tag{3.12}$$

Further, if $\max_e (r_t^e)^2 < c$, where c is constant, then

$$r_t^e \leq r_0^e + \frac{\eta t c}{2}. \tag{3.13}$$

(This requires that Φ^{-1} and ψ^{-1} exist.) This is a vanishing-per-round regret as described in (3.2).

Actually, lower bounds have been proved for the exponentially weighted forecaster which require a time-varying learning-rate η. Specifically, one can show that if $\eta_t = \sqrt{8\ln(N)/t}$ the tightest possible lower bound is $R_T^e \leq \sqrt{T\ln(N)/2}$. Logically, it makes sense that the learning-rate would be time-varying; the forecaster should be as aggressive as possible early in the sequence, when little information is known, and less aggressive as it converges, when much information is known. Since, with the proper auxiliary function and choice of η, the learning rate effects only the transient behavior (and not the upper bound), there have been efforts to completely eliminate this parameter from the analysis. One such method is call "Normal-Hedge."

3.3.3 Normal-Hedge

Normal-Hedge (NH) is a self-balancing, exponentially weighted online-learning algorithm proposed by Chaudhuri *et al.* (2009). As mentioned above, DTOL algorithms are conditioned by a handful of parameters, the most important of which, for the exponentially weighted class of forecasters, characterizes the "learning-rate" of the algorithm. Normal-Hedge differentiates itself from other DTOL algorithms in that the learning rate parameter is eliminated through a normalization step which effectively balances the information gained at each iteration. Given a bounded loss function $\ell_t^e \in \{0, 1\}$, Chaudhuri *et al.* prove an upper bound on the regret to be

$$O\left(\sqrt{T \ln \frac{1}{\epsilon}} + \ln^2 N\right)$$

where N is the number of experts, and ϵ is the "top ϵ-quantile for the distribution." If we are interested in bounding the algorithm's performance with respect to the best expert ($\epsilon = 1/N$) the bound can be rewritten $O(\sqrt{T \ln N} + \ln^2 N)$, which compares quite nicely with the optimal bound presented at the end of §3.3.2.

Normal-hedge belongs to a special class of exponentially weighted hedging algorithms, and is discussed at depth in Chaudhuri *et al.* (2009). The algorithm is initialized with a uniform probability distribution over the model class, $p_{i,0} = 1/N$, and a regret of zero, $R_{i,0} = 0$. Given the loss function $l_t^e(\hat{y}_t^e, y_t)$ which can be evaluated for each expert e at time t, as measurements become available, the following steps are taken:

1. The loss function l_t^e is evaluated for each expert.
2. The algorithm incurs a loss $l_t^A = \sum_{e=1}^N p_t^e l_t^e$ where p_t^e is the probability at time t, that the e-th expert is the best performer.
3. The cumulative regret is updated $R_t^e = R_{t-1}^e + (l_t^A - l_t^e)$ for each e.
4. The normalization constant c_t is updated so that $\frac{1}{N} \sum_{e=1}^N \exp(\frac{([R_t^e]_+)^2}{2c_t}) = e$ is satisfied.

5. The updated distribution for round t + 1 is prescribed $p^e_{t+1} \propto \frac{[R^e_t]_+}{c_t} \exp(\frac{([R^e_t]_+)^2}{2c_t})$ for each e.

The operator $[\cdot]_+$ is defined as $[\cdot]_+ = \max(\cdot, 0)$.

Although the weights are determined in a manner very similar to those of the exponentially weighted class, a potential function similar to (3.11) cannot be written explicitly for Normal-Hedge which means that the analysis discussed in §3.3.1 cannot be directly applied to determine its performance. Instead Chaudhuri *et al.* (2009) have established a handful of theorems, which allow for the bounding of the algorithm's regret. At the very least, though, one should observe that the estimation policy generated by Normal-Hedge is distributed according to (3.4), which implies that Blackwell's condition is satisfied.

3.4 Mixed-model Kalman Filters

It is well known that any probability distribution can be decomposed arbitrarily accurately[6] as a sum of Gaussian distributions. The following lemma, taken directly from Anderson & Moore (1979), summarizes the expansion.

Lemma 3.4.1 *Any probability density* $\mathrm{p}(x)$ *can be approximated as closely as desired in the space* $\mathcal{L}_1(\mathcal{R}^n)$ *by a density of the form*

$$\mathrm{p}_A(x) \approx \sum_{e=1}^{m} w^e \, \mathrm{N}(\mathbf{m}^e, \mathbf{\Sigma}^e) \tag{3.14}$$

for some number of modes m*, positive scalars* w^e*,* n*-vectors* $\mathbf{m}^e$*, and covariance matrices* $\mathbf{\Sigma}^e$*.*

For proofs the reader is referred to Lo (1972) and Feller (1968). Intuitively speaking, as the eigenvalues of $\mathbf{\Sigma}^e$ approach zero, the Gaussian bump begins to approximate the Dirac delta $\delta(\mathbf{x} - \mathbf{m}^e)$. As the number of modes (m in the analysis above) is increased, the sum of an arbitrarily large number of Dirac deltas with arbitrarily small positive weighting should be able to approximate any positive function.

[6]The approximation is such that $\mathcal{L}_1(\mathcal{R}^n) = \int_{\mathcal{R}^n} |\mathrm{p}(\mathbf{x}) - \mathrm{p}_A(\mathbf{x})| d\mathbf{x}$ can be made arbitrarily small.

Because of this, the natural development in filtering theory is to approximate estimators of a nonlinear process with a sum of Gaussian probability distributions evolving in time, and being updated as measurements become available. This is known as the Gaussian Sum Filter (GSF; see, Sorenson & Alspach, 1971; Alspach & Sorenson, 1972) and is often implemented as a collection of Extended Kalman Filters. The moments of this convex combination of filters describes the best estimate, and the uncertainty associate with that estimation. The mean and covariance for the combined estimate are written (see, Bar-Shalom *et al.*, 2001)

$$\hat{\mathbf{x}}^m_{t|t} \triangleq \sum_{e=1}^{N} w_t^e \, \hat{\mathbf{x}}^e_{t|t} \tag{3.15a}$$

$$\mathbf{P}^m_{t|t} \triangleq \sum_{e=1}^{N} w_t^e \left[\mathbf{P}^e_{t|t} + \left(\hat{\mathbf{x}}^e_{t|t} - \hat{\mathbf{x}}^m_{t|t} \right) \left(\hat{\mathbf{x}}^e_{t|t} - \hat{\mathbf{x}}^m_{t|t} \right)^H \right] \tag{3.15b}$$

where $w_t^1, w_t^1, \cdots, w_t^N \geq 0$ and $\sum_{e=1}^{N} w_t^e = 1$ are the weights assigned in Lemma 3.4.1.

Not surprisingly, the evolution equations for this mean and covariance can also be described explicitly as matrix operations on a collection of Gaussian distributions. For clarity we reintroduce the nonlinear governing equations of §1.3

$$\mathbf{x}_{t+1} = \mathrm{f}(\mathbf{x}_t) + \mathbf{w}_t \qquad \mathbf{w}_t \sim \mathrm{N}(\mathbf{0}, \mathbf{W}) \tag{3.16a}$$

$$\mathbf{y}_t = \mathrm{h}(\mathbf{x}_t) + \mathbf{v}_t \qquad \mathbf{v}_t \sim \mathrm{N}(\mathbf{0}, \mathbf{V}) \tag{3.16b}$$

with an approximate probability distribution

$$\mathrm{p}_A(\mathbf{x}_{t|t}) \approx \sum_{e=1}^{m} w_t^e \, \mathrm{N}(\bar{\mathbf{x}}^e_{t|t}, \mathbf{\Sigma}^e_{t|t}). \tag{3.17}$$

For this system, the probability distribution for the one-step-ahead prediction can be described by

$$\mathrm{p}_A(\mathbf{x}_{t+1|t}) \approx \sum_{e=1}^{m} w_t^e \, \mathrm{N}(\bar{\mathbf{x}}^e_{t+1|t}, \mathbf{\Sigma}^e_{t+1|t}) \tag{3.18}$$

where

$$\bar{\mathbf{x}}^e_{t+1|t} = \mathrm{f}(\bar{\mathbf{x}}^e_{t|t}) \tag{3.19a}$$

$$\boldsymbol{\Sigma}^e_{t+1|t} = \mathbf{F}_t \boldsymbol{\Sigma}^e_{t|t} \mathbf{F}_t^H + \mathbf{W} \qquad \mathbf{F}_t \triangleq \frac{\partial \mathrm{f}(\mathbf{x}_{t|t})}{\partial \mathbf{x}_{t|t}} \tag{3.19b}$$

which, incidentally, are also the equations for the Extended Kalman Filter.

Similarly, the update to the mean, covariance, and weights can be determined analytically for these first two moments. That is, the posterior probability distribution for the estimator is approximated by

$$\mathrm{p}_A(\mathbf{x}_{t+1|t+1}|\mathbf{y}_{t+1}) \approx \sum_{e=1}^{m} w^e_{t+1}\, \mathrm{N}(\bar{\mathbf{x}}^e_{t+1|t+1}, \boldsymbol{\Sigma}^e_{t+1|t+1}) \tag{3.20}$$

where

$$\bar{\mathbf{x}}^e_{t+1|t+1} = \bar{\mathbf{x}}^e_{t+1|t} + \boldsymbol{\Sigma}^e_{t+1|t} \mathbf{H}^H_{t+1|t} \boldsymbol{\Theta}^{-1}_{t+1|t} \left[\mathbf{y}_{t+1} - \mathrm{h}(\bar{\mathbf{x}}^e_{t+1|t})\right] \tag{3.21a}$$

$$\boldsymbol{\Sigma}^e_{t+1|t+1} = \boldsymbol{\Sigma}^e_{t+1|t} - \boldsymbol{\Sigma}^e_{t+1|t} \mathbf{H}^H_{t+1|t} \boldsymbol{\Theta}^{-1}_{t+1|t} \mathbf{H}_{t+1|t} \boldsymbol{\Sigma}^e_{t+1|t} \tag{3.21b}$$

$$w^e_{t+1} \sim w^e_t\, p(\mathbf{y}_{t+1} \,|\, \bar{\mathbf{x}}^e_{t+1|t}, \boldsymbol{\Sigma}^e_{t+1|t}) \tag{3.21c}$$

and

$$\mathbf{H}_{t+1|t} \triangleq \frac{\partial \mathrm{h}(\mathbf{x}_{t+1|t})}{\partial \mathbf{x}_{t+1|t}} \tag{3.21d}$$

$$\boldsymbol{\Theta}_{t+1|t} \triangleq \mathbf{H}_{t+1|t} \boldsymbol{\Sigma}_{t+1|t} \mathbf{H}^H_{t+1|t} + \mathbf{V}. \tag{3.21e}$$

Effectively, the GSF can be considered an exponentially weighted forecaster (discussed in §3.3.2) where each expert is an Extended Kalman Filter and the loss function is

$$\ell^e_{t+1}(\mathrm{h}(\mathbf{x}^e_{t+1|t}), \mathbf{y}_{t+1}) = \frac{1}{2} \left[\mathbf{y}_{t+1} - \mathrm{h}(\mathbf{x}^e_{t+1|t})\right]^H \boldsymbol{\Theta}^{-1}_{t+1|t} \left[\mathbf{y}_{t+1} - \mathrm{h}(\mathbf{x}^e_{t+1|t})\right]. \tag{3.22}$$

In the limit that the number of modes considered is infinite it becomes obvious that this is a good strategy for estimating nonlinear systems. But since this limit is not

achievable in practice alternatives must be considered. Also, as mentioned in the introduction, this method still relies on a sufficiently small number of states such that the EKF can be implemented; this is not the case for this paper. In the following section we present the Game-theoretic Ensemble Kalman Filter (GEnKF), a method which seeks to improve upon these two potential weaknesses.

3.4.1 Game-theoretic Ensemble Kalman Filtering

The Game-theoretic Ensemble Kalman Filter improves upon standard GSF through two simple, but subtle, changes. First, the Extended Kalman Filter is replaced with the Ensemble Kalman Filter (as discussed in §1.3). This effectively replaces the unattractive restriction to low-dimensional systems required by the EKF with the efficient scalability of the EnKF. Given the improved performance in high-dimensional (nonlinear) systems reported in Colburn *et al.* (2011*a*) and the relatively simple implementation required by the EnKF this is a reasonable substitution.[7]

A second change replaces the Bayesian update in (3.21c) with the weight update provided by the Normal-Hedge algorithm. In a typical implementation of NH, the experts themselves need not be filters, and could instead be single simulations of the system (which is something considered in the results below), but the idea of allowing estimators to compete with each other in a game-theoretic setting is quite attractive since estimators will perform better in the long-term than other near-by particles.

To summarize, the novel idea presented is an implementation of Normal-Hedge, where each expert is itself an Ensemble Kalman Filter. Recalling from the introduction, that although an estimation algorithm should be consistent with the optimal solution (the Kalman filter) under standard linear assumptions, the ulti-

[7]Incidentally, for linear systems with Gaussian uncertainty in the initial state and process/measurement noise the EnKF is known to converge to the KF solution as the number of ensemble members become arbitrarily large. Thus, it's important to emphasize that this formulation is equivalent to the Kalman Solution under the appropriate assumptions. In fact, it can be seen clearly that the KF statistics are recovered perfectly for the GEnKF if each estimator in (3.15) is considered an EnKF and the number of ensemble members is taken to infinity. This is independent of how weights are selected.

mate goal is to develop a strategy which scales well to high-dimensional systems and performs well with respect to the best alternative (an idea that fits naturally into the decision-theoretic online learning framework). The GEnKF conveniently hybridizes these two algorithms and is validated in the numerical examples presented below.

3.5 Results and Numerical Analysis

The Lorenz equation (Lorenz, 1963) is used here as a simple model of a nonlinear system with self-sustained chaotic unsteadiness[8] in order to perform this comparison. Solutions of these equations approach a well-defined manifold or *attractor* of dimension slightly higher than two. Perturbed trajectories converge exponentially back onto the attractor, while adjacent trajectories diverge exponentially within the attractor, creating the familiar chaotic motion in the Lorenz system (see, Fig. 1.1). Note that, for convenience, the system equations of (3.23) are transformed slightly from the traditional form such that the attractor is approximately centered at the origin. Lorenz (1963) developed the model to mimic the advective state of water in a closed, circular tube heated on the bottom and cooled on the top. The Lorenz equation is a system of three coupled, nonlinear ordinary differential equations given by

$$\frac{d\mathbf{x}(t)}{dt} = \begin{pmatrix} \sigma\left(x_2 - x_1\right) \\ -x_2 - x_1 x_3 \\ -\beta x_3 + x_1 x_2 - \beta r \end{pmatrix}, \tag{3.23}$$

where σ, β, and r are tunable parameters.

This 3-state ODE is ideal for validating nonlinear estimators because of the chaotic nature of the equations; specifically, the presence of an energy conserving, 2nd-order nonlinearity, similar to that of the Kuramoto-Sivashinsky and Navier-Stokes equation, make it a particularly attractive testbed. Appealing to the ergodic

[8]That is, the system considered maintains its nonperiodic, finite, bounded unsteady motion with no externally-applied stochastic forcing.

nature of the Lorenz system, estimates can be performed sequentially and historical errors can be averaged without having to re-initialize the system at each iteration. This dramatically reduces the computational burden required when calculating infinite-time statistics.

The parameters were selected to align with classic studies of the equations ($\sigma = 10$, $\beta = 4$, $r = 48$) and are consistent with results published elsewhere. Generally speaking, they were selected so that the governing equations will exhibit a sufficiently chaotic nature for the simulations considered. When measurements were used in the measurement update process, only for first state x_1 was measured with additive, zero-mean, white noise corrupting the signal. For all numerical experiments the variance of this additive noise was $V = 4.0$.

When applying game-theoretic methods to dynamic system the sampling of phase space must be constantly re-adjusted so that the regions of high-probability retain sufficient precision. This process of "resampling" moves the majority of particles near the best estimate, and is reviewed coherently in Arulampalam *et al.* (2002). Resampling is the necessary heuristic introduced in particle filters to keep the probability distribution from resolving to the stationary distribution which best describes the attractor (that is the probability distribution associated with the infinite-time statistics of the attractor). This is achieved most simply by "copying" experts which are performing well and slightly perturbing them so that each particle evolves uniquely in time. This requires selecting a parameter (or a handful of parameters) which determine how re-sampled particles will be perturbed. Resampling becomes even less clear in the PDE setting.

In the results below the performance of the Normal-Hedge algorithm was characterized extensively as a function of measurement period and repopulation covariance. Then, after finding the best parameter settings, these results were compared with the Ensemble Kalman Filter and Game-theoretic Ensemble Kalman Filter. Typically, the results are characterized by the trajectories in Fig. 3.1, which are generated by averaging update periods until at least 4 decimal places precision is achieved with a minimum iteration count of 1.5 million updates. The "saw-tooth" behavior in these images characterizes two phenomena: (i) the exponential

divergence of the Lorenz attractor and (ii) the average effect of the update. In Fig. 3.1, where the effect of measurement period is quantified for the Normal-Hedge algorithm, notice that the exponential behavior is evident for cases were the measurement period is long, and less obvious for shorter measurement periods. This is consistent with a chaotic system. Similarly, the effect of the update is much more substantial in simulations with long measurement periods. Figures 3.2 and 3.3 represent the average error of the estimator (that is, the integral measure of the "saw-tooth" plots). These averages more clearly capture the trends in parameter changes.

Figure 3.1 compares plots of the update performed over a window $t = 0.2$ time-units. To guarantee that the algorithm has been implemented properly results for the shortest measurement update period $\Delta t = 0.01$ have been checked, and are consistent with those published in Freund (2011). Although an important check, since the ultimate goal in estimation is to increase the predictive capabilities of each algorithm, we are mostly interested evaluating performance as the measurement period increases. Figure 3.2 shows the mean L2-error of the state estimate as a function of measurement update period. Each entry in this image is an integral average of the error from Figure 3.1. As expected, the mean error increases as the measurement frequency decreases. This is consistent with an "information-based" interpretation of this data: estimation error should decrease as update frequency increases since each measurement provides more information about the "truth." The only exception of this is at the shortest time-scales; the error is larger for a smallest measurement update period. At this time-scale and repopulation covariance, the dynamics do not have sufficient time to remove (that is smooth) noise introduced through expert repopulation process. Figuratively speaking, the repopulation procedure dominates the estimation process and is pulling the estimate off the attractor.

By comparing Fig. 3.3(a) and Fig. 3.3(b) the effect of repopulation covariance can understood. Although the measurement period in Fig. 3.3(b) is 1/10-th of that in Fig. 3.3(a) the error is only halved. One might expect that the error would be reduce further by choosing a better repopulation strategy, and it is likely

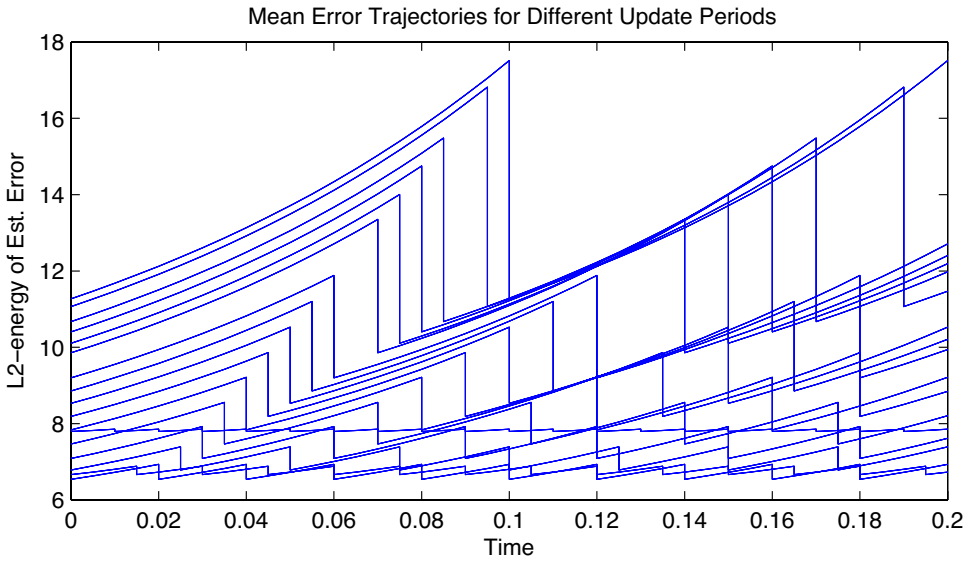


Figure 3.1: Plots of the update performed over a window $t = 0.2$ time-units. Results for the shortest measurement update period $\Delta t = 0.01$ are consistent with those published in Freund (2011). The ultimate goal in estimation is to increase the predictive capabilities of the algorithms. Thus, we are interested evaluating performance as a function of measurement period.

that the repopulation variance is arbitrarily inflating the results for $\Delta t = 0.01$.

3.5.1 Results: GEnKF

The Ensemble Kalman Filter is strongly dependent on the resolution of the sample covariance matrix, Fig. 3.4 demonstrates the convergence properties of the EnKF performance as a function of number of ensemble members. As the number of ensemble members increase the sample covariance more faithfully approximates the true covariance. As a result the cloud of particles is more accurately shifted in phase space to improve the estimate. It is also clear that the reduction of error approaches some fundamental performance limit asymptotically.

The results in Fig. 3.5 examine the relationship between effects of ensemble size, number of EnKF experts competing in parallel, and fundamental estimation limitations. First, the results demonstrate the advantage of using Normal-Hedge in the setting. Clearly, a convex combination of EnKFs performs better than a single EnKF of the same size. Unfortunately, they do not perform as well as the case

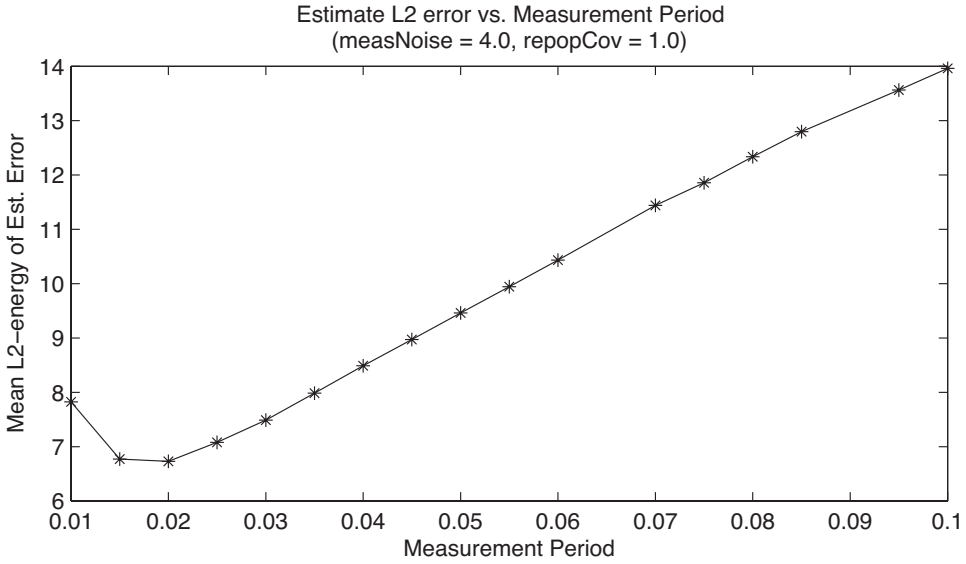
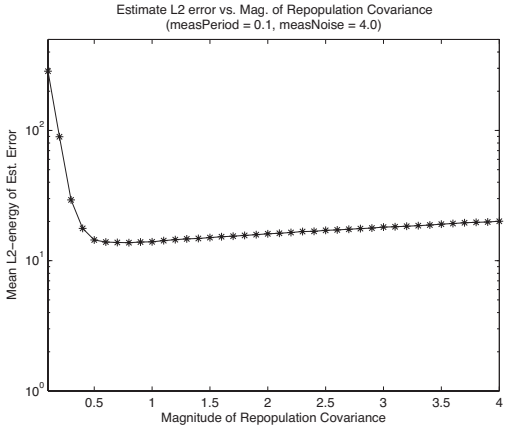
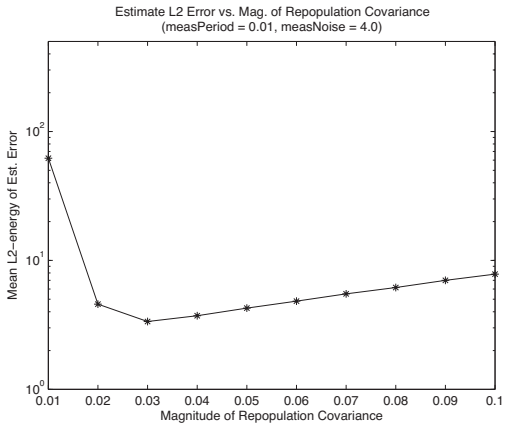


Figure 3.2: Mean L2-error of the state estimate as a function of measurement update period. Each entry in this image is an integral average of the error from Figure 3.1. As expected, the mean error increases as the measurement frequency decreases. This is consistent with an "information-based" interpretation of this data: estimation error should decrease as update frequency increases since each measurement provides more information about the "truth." The only exception of this is at the shortest time-scales; the error is larger for a smallest measurement update period. At this time-scale and repopulation covariance, Normal-Hedge does not have sufficient time to compensate (that is remove) noise introduced through expert repopulation.



(a) Mean L2-error of the state estimate versus magnitude of the repopulation covariance for a measurement period $\Delta t = 0.1$.



(b) Mean L2-error of the state estimate versus magnitude of the repopulation covariance for a measurement period $\Delta t = 0.01$. Note that for all resampling "volumes" the error is about half the magnitude of that in Figure 3.3(a).

Figure 3.3: The mean L2-norm of the error of the state estimation when using Normal-Hedge at two measurement periods [Figures 3.3(a) and 3.3(b) use a measurement update period of $\Delta t = 0.1$ and $\Delta t = 0.01$, respectively.] for various resampling distributions. The results converged after being averaged over at least 1.6 million iterations.

when all resources are allocated to the EnKF alone. The diminished performance is offset, though, by the decrease communication requirements (not visualized) for the GEnKF. Computationally speaking, the GEnKF in this example requires the same computational effort as the 720 member EnKF in the forecast step, but has the same communication requirements as the 15 member EnKF in the update step. Practically speaking, this results parallelism which allows for more than 48X speedup in execution time.

Perhaps the most important trend captured requires holding constant the total computation resources and measuring estimation error as a function of resources allocated to each EnKF. The x-axis in Fig. 3.6 determines the number of ensemble members in each EnKF, and correspondingly the number of EnKF experts used in the learning algorithm. For example, a point along this line, totalPercentage = 5, represents 20 experts, where each expert is an EnKF with 36 ensemble members. The far left end (totalPercentage = 0) quantifies the estimate using Normal-Hedge only, and the far right end (totalPercentage = 100) quantifies the estimate using only the Ensemble Kalman Filter. Not surprisingly, the best performance is achieved when all resources are allocated to the EnKF and all information is shared uniformly between ensemble members. But, there is also a local minimum at totalPercentage = 5.55 percent (corresponding to 18 EnKFs with 40 ensemble members each). Surprisingly, there is a situation where information is *not* shared uniformly across all simulations, and the L_2-norm of the error is better than when that information is shared.

3.6 Conclusion, Future Work

This work presents a novel estimation algorithm, the Game-theoretic Ensemble Kalman Filter (GEnKF), which intelligently switches between estimators competing in a game to provide the "best" estimate. The proposed algorithm reduces communication (a limiting factor as the state size becomes large) because although each filter requires full state information locally to update estimates, the competing filters are only required to broadcast one double precision value to the

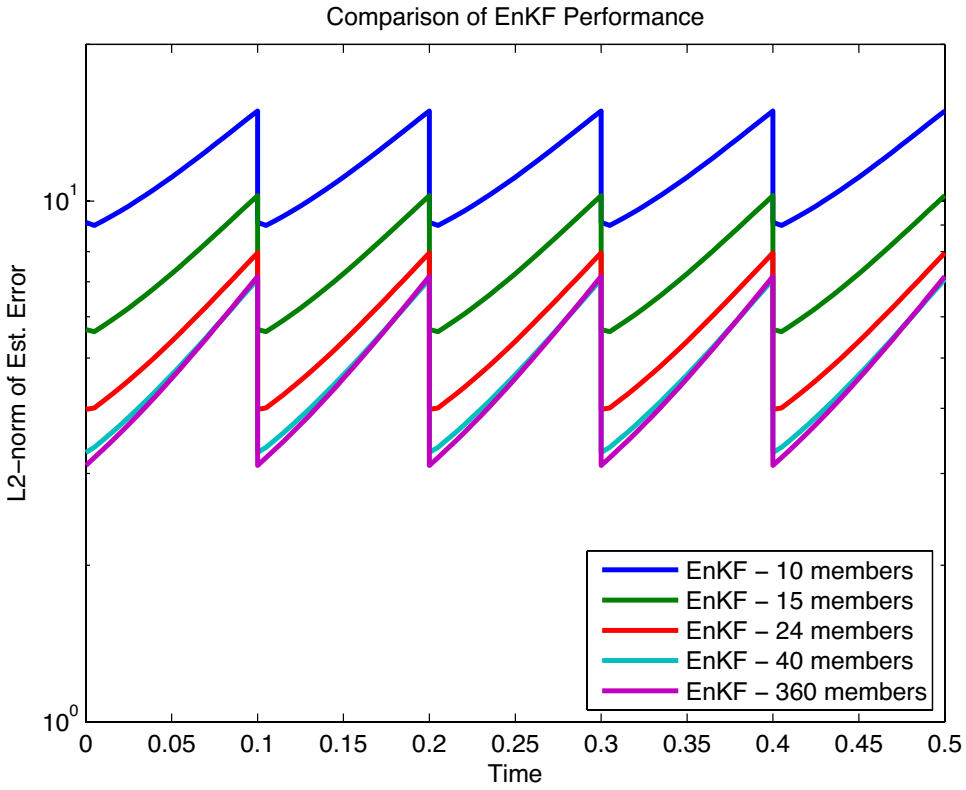


Figure 3.4: A demonstration of the increased fidelity as a result of an increased number of ensemble members. These "saw-tooth" plots are EnKF updates averaged over 2,000,000 sequential updates. As the number of ensemble members increases the fidelity of the estimate increases.

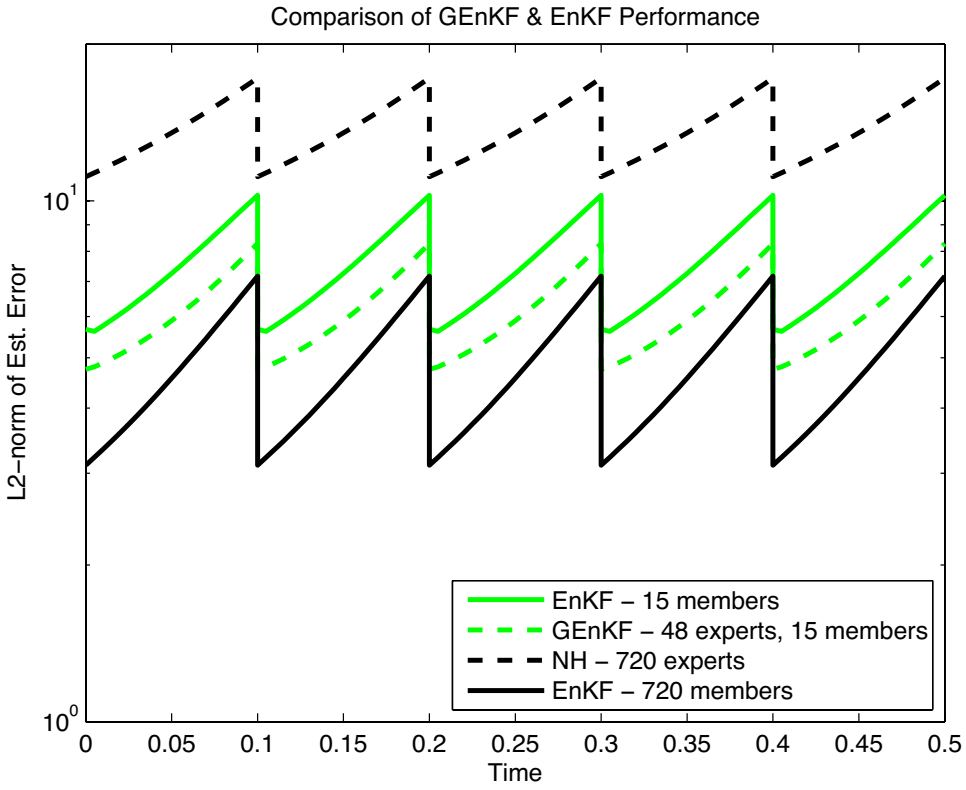


Figure 3.5: This numerical experiment examines the balance between the number of experts (EnKFs) and the number of ensemble members in each EnKF. In practical problems only a finite amount of computational resources are available. Determining how to best use those resources is a problem of paramount importance. This balance is probably problem specific.

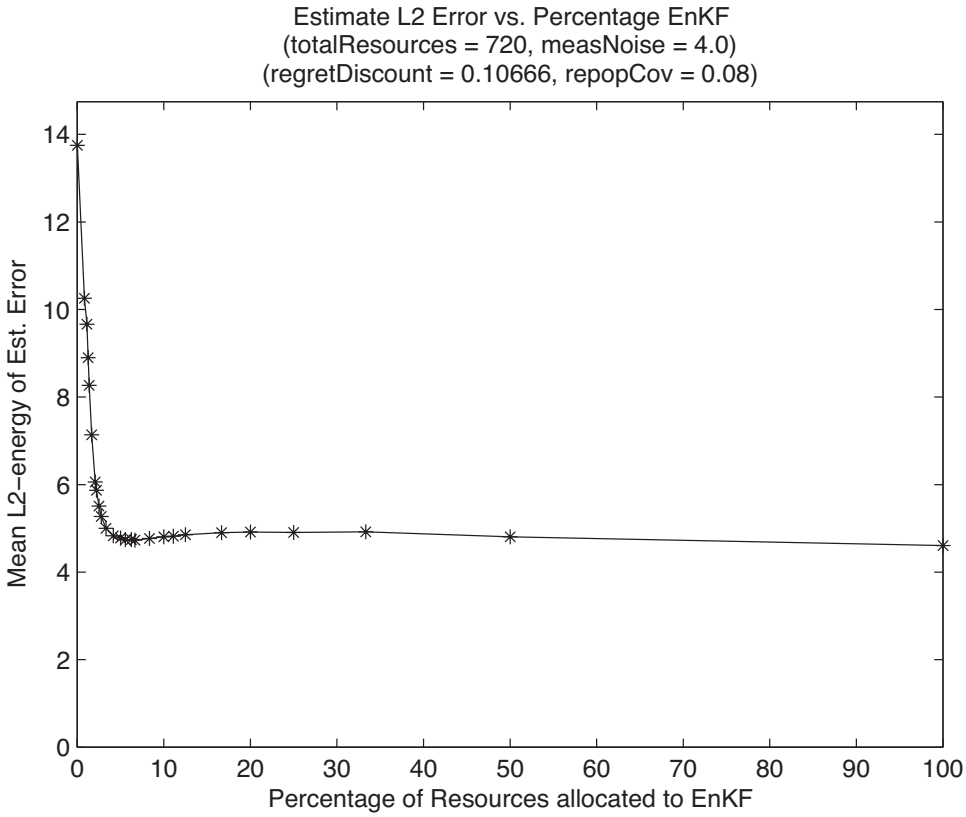


Figure 3.6: A comparison of the mean L2-norm of the error of the state estimation when using different implementations of the hybrid Game-theoretic Ensemble Kalman Filter proposed in §3.4.1. By holding constant the total number of resources (that is the total number of CPUs, which directly determines the feasible number of total experts and ensemble members) the best allocation can be found. In this example the maximum number of ensemble members permitted was $N = 720$. The x-axis (Percent of resources allocated to EnKF) determines the number of ensemble members (and thus the number of experts) used in the learning algorithm. The far left end (totalPercentage = 0) quantifies the estimate using Normal-Hedge only, and the far right end (totalPercentage = 100) quantifies the estimate using only the Ensemble Kalman Filter. A point along this line, totalPercentage = 10, represents 10 experts, where each expert is an EnKF with 72 ensemble members.

Normal-Hedge algorithm – which determines the best combination of estimates.

As a follow-up paper, and in light of the results presented in Colburn *et al.* (2011*a*), we propose applying this algorithm to a problem of interest, namely near-wall turbulent channel flow estimation. This problem requires that communication be as efficient as possible while providing sufficient information to track a complex flow-field.

Lastly, although the DTOL framework provides an attractive method for combining estimators, it does not allow for the individual expert's EnKF to be improved by the global algorithm. Developing a framework to feedback information to experts which are performing poorly, so that they might "fix" themselves, is an open research topic.

3.7 Acknowledgments

The authors gratefully acknowledge the generous financial support of the National Security Education Center (NSEC) at Los Alamos National Laboratory (LANL) and the many helpful conversations with Yoav Freund (UCSD).

Chapter 3, in full, is a reprint of material as it has been drafted for publication in 2011. (Colburn & Bewley (2011), "A hybrid Game-theoretic, Ensemble Kalman approach to online learning of non-stationary, high-dimensional sequences." Drafted for publication.) The dissertation author was the primary investigator and author of this paper.

Chapter 4

Estimation and Adaptive Observation of Environmental Plumes

4.1 Abstract

Here we present our new Adaptive Observation (AO) method, dubbed Dynamic Adaptive Observation (DAO). Unlike current AO methods, DAO rigorously incorporate vehicle dynamics to compute optimal feasible trajectories (waypoints) of a swarm of sensor vehicles in order to minimize the forecast uncertainty. A numerical experiment is performed, motivated by the recent Gulf coast oil and the Icelandic ash problems, where the well-known Ensemble Kalman Filter (EnKF) is combined with DAO for the estimation of a convection-driven environmental plume problem.

4.2 Introduction

The task of adaptive observation (AO) is to determine the future sensor positions for systems where uncertainty is distributed (non-trivially) in a region of interest. This class of problems is considered a hybrid of problems from control and

estimation theory, which, generally speaking, are either distributed or centralized in nature.

In a typical distributed AO strategy (see Martínez *et al.* (2007), Laventall & Cortés (2009), Stanković & Stipanović (2009), and Zhang & Leonard (2010)) each sensor vehicle has little knowledge of the sensed system, and deployment is planned locally. Provided that the system dynamics are sufficiently smooth, and that communication is sufficiently efficient, the collective simple behaviors lead to global actions that improve estimation. This also guarantees that the solutions are easily deployable, scale nicely, and inherently satisfy vehicle dynamic constraints. The majority of existing distributed AO methods reduce to an optimal coverage, extremum seeking, or level-set tracking problem. While these approaches work adequately for certain applications, their performance is degraded in convection-driven (that is, wind-dominated rather than diffusion dominated) problems with complicated level sets and state couplings, such as those encountered in atmospheric and oceanographic applications. Furthermore, the distribution of measurements is quite sparse in these applications, and it is not clear how distributed methods perform as the density of measurements becomes very small.

In such problems, it is beneficial to plan the sensor distributions more deliberately with a centralized AO strategy, where the sensed system model is leveraged to optimize the sensor positions. As a consequence, the centralized AO strategy is computationally intensive, and thus cannot be computed locally on the individual vehicles. Rather, the bulk of necessary computations must be off-loaded to a centralized supercomputer cluster, where optimized vehicle trajectories (or waypoints selected along these trajectories) are periodically broadcasted back to the sensor vehicles. As we are interested in applying AO to large-scale systems found in atmospheric and oceanographic applications, we focus the present work on the centralized AO strategy.

The centralized AO strategy may be further divided into "sensitivity-based" and "uncertainty-based" approaches. In sensitivity-based AO (see Langland & Rohaly (1996) and Buizza & Montani (1999)), a system adjoint is used to reveal "sensitive regions" of the domain that contribute significantly to the estima-

tion/forecast uncertainty. The strength of these approaches centers on the relative speed and computational efficiency required for the calculation of the adjoint, as the sensitivity computation cost is on the same order as the forward system propagation computational cost. However, most methods following such an approach do not address how such "sensitive regions" should be optimally probed by sensing vehicles (this issue is partially addressed in Bishop (2000 (submitted) and Baker & Daley (2000)). Furthermore, these adjoint methods base their sensitivity calculations on a *single* forecast, which is prone to exponential divergence and bifurcation in chaotic systems prevalent in atmospheric and oceanographic science. As a consequence, the sensitive regions computed may be incorrect due to a wrong forecast.

Uncertainty-based AO methods (see Bishop *et al.* (2001) and Yilmaz *et al.* (2008)) take a different approach. Rather than computing the sensitivity of the forecast, they seek a measurement location sequence that minimizes the estimation/forecast uncertainty. This is usually achieved by considering a set of all possible sensor location sequences, and computing the anticipated forecast uncertainty associated with each. Although this strategy scales poorly with large domain size and/or number of sensors, Bishop *et al.* (2001) recognizes that each anticipated forecast is computed independently, making the approach *embarrassingly parallel* (the total computation time is inversely proportional to the computational resources available). Furthermore, with sub-optimality, Bishop *et al.* (2001) noted the set can be significantly reduced by searching for each measurement location one at a time. In Yilmaz *et al.* (2008), because vehicle motion constraints are part of the formulation, the resulting feasible set is much smaller.

As far as we know, existing centralized AO methods do not fully incorporate vehicle dynamics. This is perhaps because the various atmospheric and oceanographic models to which centralized AO has traditionally been applied are essentially static when compared with the motion time scales of the sensor vehicles. In order to overcome this problem, for example when implementing EnKF on a practical system, Majumdar *et al.* (2002*b*) simply considered approximately 40 pre-approved feasible flight paths and select among them. Vehicle dynamics are partially incorporated in Yilmaz *et al.* (2008) as linear constraints, but because

linear constraints cannot model complex vehicle dynamics, overly conservative linear constraints are imposed to ensure a dynamically feasible solution, resulting in unnecessarily sluggish optimized vehicle trajectories.

We appreciate the steps toward computational efficiency that have been achieved by both sensitivity-based and uncertainty-based AO approaches but seek a method which rigorously accounts for vehicle dynamics. To this end, we propose a new centralized AO method, dubbed Dynamic Adaptive Observation (DAO), that fills this void by combining various features from existing centralized AO approaches and incorporates full vehicle dynamics. DAO uses the Kalman Filter to predict the future estimation/forecast error covariance and to compute the best control, subjected to the vehicle dynamic constraints, to minimize this covariance. This is achieved by minimizing a relevant cost function balancing a metric of the estimation/forecast quality with another metric measuring the cost of the control applied to the vehicles. Because explicit formulation of the optimal control with respect to the cost function is difficult to derive analytically, we use adjoint analysis to calculate the local gradient, and compute the optimal control iteratively. Once the optimal solution is found, either the control trajectories, vehicle trajectories, or waypoints extracted from the vehicle trajectories can be transmitted to the sensor vehicles.

The rest of the paper is as follows: in §4.3 we formulate the AO problem, where our objective is to minimize a cost balancing a measure of the estimation/forecast quality with vehicle-related penalties. Adjoint analysis is perform on the cost to reveal local gradient information in §4.4. Various generalizations to DAO and extensions to discrete- and continuous-time are discussed in §4.5. A numerical experiment is shown in §4.6 which combines DAO with an Ensemble Kalman Filter (EnKF) to estimate a convection-driven environmental plume.

4.3 AO problem formulation

Suppose an uncontrolled, linear, spatially discretized PDE, $\mathbf{x}$, on a physical domain of interest, written in the continuous-time ODE form

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{w}(t), \qquad \mathbf{w}(t) \sim N(0, \mathbf{W}), \tag{4.1}$$

where $\mathbf{w}(t)$ has zero mean and covariance $\mathbf{W}$. Within the physical domain of interest, there are M sensor vehicles. The i'th vehicle's (linearized) continuous-time dynamics, with state $\mathbf{q}^i(t)$ and control $\mathbf{u}^i(t)$, may be written

$$\frac{d\mathbf{q}^i(t)}{dt} = \mathbf{F}\mathbf{q}^i(t) + \mathbf{G}\mathbf{u}^i(t). \tag{4.2}$$

Note $\mathbf{A}$, $\mathbf{B}$, $\mathbf{F}$, and $\mathbf{G}$ in (4.1) and (4.2) can be time-varying. For simplicity we shall restrict them to this simpler linear, time-invariant case in the following analysis. The extension to the nonlinear and time-varying case is discussed in §4.5.1.

The vehicles move about the domain *continuously*, while taking measurements *discretely* at times t_k. Vehicle states such as position, heading, and velocity affect both the measurements and the noise, and thus both the (linearized) measurement matrix $\mathbf{H}(\mathbf{q}^i(t_k))$ and the noise covariance matrix $\mathbf{R}(\mathbf{q}^i(t_k))$ depend on the vehicle state. For convenience, the notation $\mathbf{H}_k^i$ and $\mathbf{R}_k^i$ is adopted, with the understanding that the dependence of these matrices on the vehicle states is implied. The i'th vehicle measurement vector at time t_k is thus written

$$\mathbf{y}_k^i = \mathbf{H}_k^i \mathbf{x}_k + \mathbf{v}_k^i, \qquad \mathbf{v}_k^i \sim N(0, \mathbf{R}_k^i), \tag{4.3}$$

and the collection of all measurement is as follows:

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{y}_k^1 \\ \vdots \\ \mathbf{y}_k^M \end{bmatrix}, \ \mathbf{H}_k = \begin{bmatrix} \mathbf{H}_k^1 \\ \vdots \\ \mathbf{H}_k^M \end{bmatrix}, \ \mathbf{R}_k = \begin{bmatrix} \mathbf{R}_k^1 & & 0 \\ & \ddots & \\ 0 & & \mathbf{R}_k^M \end{bmatrix}.$$

The state estimate $\hat{\mathbf{x}}$ is updated at each measurement with estimation error

covariance $\mathbf{P}$. We adopt a mixed continuous/discrete-time scheme for the evolution of $\mathbf{P}$, where the error covariance propagates between measurements in continuous-time according to

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{A}\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^T + \mathbf{B}\mathbf{W}\mathbf{B}^T, \tag{4.4}$$

and updated at the measurements according to the discrete-time formula

$$\begin{aligned} \mathbf{P}_k^+ &= (\mathbf{I} - \mathbf{L}_k\mathbf{H}_k)\mathbf{P}_k^-, \\ \mathbf{L}_k &= \mathbf{P}_k^-\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \end{aligned} \tag{4.5}$$

where $()^+$ and $()^-$ denote the estimate *a posteriori* and *a priori*, respectively.

The DAO problem statement is framed as follows: *At time t_0, the vehicle states $\mathbf{q}_0^i$ and estimation error covariance $\mathbf{P}_0$ are specified. Design a set of control trajectories $\mathbf{u}^i(t)$ for all sensor vehicles over the time window $[0, t_K^-]$ that balance the control effort and a metric quantifying the forecast quality at the final time t_F, where $t_F \geq t_K$, conditioned on the measurements taken by the vehicles at times $\{t_1, t_2, \cdots, t_K\}$.* For simplicity, here we choose a quadratic measure for the control effort, and a weighted sum of the variances to quantify the forecast accuracy (both choices could certainly be generalized, as demonstrated in §4.5.1):

$$\min_{\mathbf{u}^i(t)} J = \mathsf{trace}(\mathbf{T}\mathbf{P}_F) + \frac{1}{2}\sum_{i=1}^{M}\int_0^{t_K^-} \mathbf{u}^i(t)^T\mathbf{Q}_\mathbf{u}\mathbf{u}^i(t)\,dt, \tag{4.6}$$

where $\mathbf{T}$ is a diagonal matrix that targets specific regions of the domain, and $\mathbf{Q}_\mathbf{u} > 0$. The cartoon in Fig. 4.1 illustrates this mixed continuous/discrete-time formulation, and the relationships between the several quantities involved. Note that, as $\mathbf{P}(t)$ is updated discretely at each measurement time, the trajectory $\mathbf{P}(t)$ is *piecewise smooth*. Also note that, as the effect of $\mathbf{u}^i(t)$ is nonlinear, the cost function J is, in general, nonconvex; thus the global optimization of this cost function can not be guaranteed with a computationally tractable algorithm.

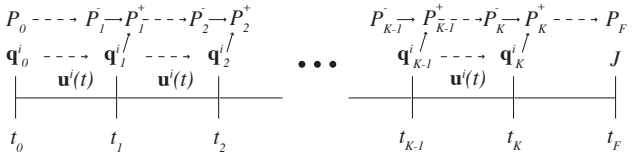


Figure 4.1: Cartoon illustrating the problem formulation. The controls $\mathbf{u}^i(t)$ affect the (continuous-time) evolution of sensor vehicle trajectories $\mathbf{q}^i(t)$. The sensor vehicle positions at the measurement times, $\mathbf{q}^i_k$, in turn, affect the (discrete-time) updates to the covariance of the estimation error $\mathbf{P}_k$; this covariance otherwise evolves continuously between the measurements, and between t_K and t_F. The cost J depends on $\mathbf{P}_F$ and the $\mathbf{u}^i(t)$ for $t \in [0, t_K^-]$; a set of controls $\mathbf{u}^i(t)$ is sought to minimize this cost. Dashed arrows denote continuous-time propagation; solid arrows denote discrete-time updates.

4.4 Computing the optimal control

Given the complex dependence of J on $\mathbf{u}^i(t)$, the minimizing solution is sought via an iterative approach, where a nominal control trajectory for each vehicle is assumed initially, then a local gradient $\nabla_{\mathbf{u}^i(t)} J$ is computed using *adjoint analysis*. Applying a perturbation to the assumed nominal control trajectories causes a chain reaction that perturbs other variables; the first-order perturbations of the relevant variables may be written:

$$\frac{d\mathbf{P}(t)'}{dt} = \mathbf{A}\mathbf{P}(t)' + \mathbf{P}(t)'\mathbf{A}^T, \qquad \mathbf{P}'_0 = 0,\ \mathbf{W}' = 0 \tag{4.7a}$$

$$\frac{d\mathbf{q}^i(t)'}{dt} = \mathbf{F}\mathbf{q}^i(t)' + \mathbf{G}\mathbf{u}^i(t)', \qquad \mathbf{q}^i_0{}' = 0, \tag{4.7b}$$

$$\begin{aligned}\mathbf{P}_k^{+\prime} = \mathbf{P}_k^{-\prime} &- (\mathbf{P}_k^{-\prime}\mathbf{H}_k^T + \mathbf{P}_k^-(\mathbf{H}'_k)^T)\mathbf{L}_k^T - \mathbf{L}_k(\mathbf{H}_k\mathbf{P}_k^{-\prime} + \mathbf{H}'_k\mathbf{P}_k^-) \\ &+ \mathbf{L}_k(\mathbf{H}'_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{H}_k\mathbf{P}_k^{-\prime}\mathbf{H}_k^T + \mathbf{H}_k\mathbf{P}_k^-(\mathbf{H}'_k)^T + \mathbf{R}'_k)\mathbf{L}_k^T,\end{aligned} \tag{4.7c}$$

$$J' = \mathsf{trace}(\mathbf{T}\mathbf{P}'_F) + \sum_{i=1}^{M} \int_0^{t_K^-} \mathbf{u}^i(t)^T \mathbf{Q}_u \mathbf{u}^i(t)'\, dt, \tag{4.7d}$$

$$\mathbf{R}'_k = \begin{bmatrix} \mathbf{R}_k^{1\prime} & & 0 \\ & \ddots & \\ 0 & & \mathbf{R}_k^{M\prime} \end{bmatrix}, \quad \mathbf{H}'_k = \begin{bmatrix} \mathbf{H}_k^{1\prime} \\ \vdots \\ \mathbf{H}_k^{M\prime} \end{bmatrix}, \tag{4.7e}$$

$$\mathbf{R}_k^{i\prime} = \left(\frac{d\mathbf{R}_k^i}{d\mathbf{q}_k^i}\right)\mathbf{q}_k^{i\prime}, \quad \mathbf{H}_k^{i\prime} = \left(\frac{d\mathbf{H}_k^i}{d\mathbf{q}_k^i}\right)\mathbf{q}_k^{i\prime}, \tag{4.7f}$$

where $\mathbf{q}_0^i{}'$, $\mathbf{P}_0'$, and $\mathbf{W}'$ are zero because they are not affected by $\mathbf{u}^i(t)$. Note that $\frac{d\mathbf{R}_k^i}{d\mathbf{q}_k^i}$ and $\frac{d\mathbf{H}_k^i}{d\mathbf{q}_k^i}$ are rank-3 tensors that contract by the inner product with $\mathbf{q}_k^i{}'$ to yield the matrices $\mathbf{R}_k^i{}'$ and $\mathbf{H}_k^i{}'$. The first-order term in the Taylor series expansion of J about $\mathbf{u}^i(t)$ is written

$$J' = \sum_{i=1}^{M} \int_0^{t_K^-} \left(\nabla_{\mathbf{u}^i(t)} J\right)^T \mathbf{u}^i(t)'\, dt, \tag{4.8}$$

which is very similar to (4.7d) except for the $\mathsf{trace}(\mathbf{T}\mathbf{P}_F')$ term. The rest of the derivation focuses on rewriting $\mathsf{trace}(\mathbf{T}\mathbf{P}_F')$ in a same form as (4.8).

We now simplify the description of (4.7b) and (4.7a) by introducing the linear operators $\mathcal{L}(\mathbf{P}')$, $\mathcal{M}(\mathbf{q}')^i$, and $\mathcal{B}(\mathbf{u}')^i$:

$$\mathcal{L}(\mathbf{P}') \triangleq \frac{d\mathbf{P}(t)'}{dt} - \mathbf{A}\mathbf{P}(t)' - \mathbf{P}(t)'\mathbf{A}^T, \tag{4.9a}$$

$$\mathcal{M}(\mathbf{q}')^i \triangleq \frac{d\mathbf{q}^i(t)'}{dt} - \mathbf{F}\mathbf{q}^i(t)', \tag{4.9b}$$

$$\mathcal{B}(\mathbf{u}')^i \triangleq \mathbf{G}\mathbf{u}^i(t)', \tag{4.9c}$$

so that $\mathcal{L}(\mathbf{P}') = 0$ by (4.7a) and $\mathcal{M}(\mathbf{q}')^i = \mathcal{B}(\mathbf{u}')^i$ by (4.7b).

An adjoint variable $\mathbf{S}(t)$ and adjoint operator $\mathcal{L}^*(\cdot)$ is now defined over the window $[t_K^+, t_F]$ and considered within an adjoint identity based on a relevant inner product:

$$\langle \mathbf{S}, \mathcal{L}(\mathbf{P}') \rangle_{t_K^+, t_F} = \langle \mathcal{L}^*(\mathbf{S}), \mathbf{P}' \rangle_{t_K^+, t_F} + a, \tag{4.10a}$$

$$\langle \mathbf{X}, \mathbf{Y} \rangle_{t_K^+, t_F} \triangleq \int_{t_K^+}^{t_F} \mathsf{trace}(\mathbf{X}(t)^T \mathbf{Y}(t))\, dt. \tag{4.10b}$$

Using integration by parts, it can be shown that

$$\mathcal{L}^*(\mathbf{S}) = -\frac{d\mathbf{S}(t)}{dt} - \mathbf{A}^T\mathbf{S}(t) - \mathbf{S}(t)\mathbf{A}, \tag{4.11a}$$

$$a = \mathsf{trace}(\mathbf{S}_F^T \mathbf{P}_F'{}) - \mathsf{trace}((\mathbf{S}_K^+)^T \mathbf{P}_K^+{}'). \tag{4.11b}$$

Taking $\mathcal{L}^*(\mathbf{S}) = 0$ and $\mathbf{S}_F^T = \mathbf{T}$, (4.7d) is manipulated using the relationships

established in (4.10) and (4.11) into

$$J' = \mathsf{trace}((\mathbf{S}_K^+)^T \mathbf{P}_K^+{}') + \sum_{i=1}^{M} \int_0^{t_K^-} \mathbf{u}^i(t)^T \mathbf{Q}_\mathbf{u} \mathbf{u}^i(t)' \, dt. \tag{4.12}$$

Note that setting $\mathcal{L}^*(\mathbf{S}) = 0$ and $\mathbf{S}_F^T = \mathbf{T}$ is equivalent to defining a backward-in-time evolution equation for $\mathbf{S}$ such that $\frac{d\mathbf{S}(t)}{dt} = -\mathbf{A}^T\mathbf{S}(t) - \mathbf{S}(t)\mathbf{A}$ with starting condition $\mathbf{S}_F^T = \mathbf{T}$. By the special structure of the evolution equation for $\mathbf{S}$ and the symmetric definition of $\mathbf{S}_F$, it is clear that $\mathbf{S}_K^+$ is also symmetric in (4.12); and it will become apparent later in the formulation, that in fact the entire trajectory $\mathbf{S}(t)$ is symmetric.

Substituting (4.7c), (4.7e), and (4.7f) into $\mathbf{P}_K^+{}'$ in (4.12) and leveraging the trace identity $\mathsf{trace}(\mathbf{AB}) = \mathsf{trace}(\mathbf{BA}) = \mathsf{trace}(\mathbf{A}^T\mathbf{B}^T)$, the $\mathbf{P}_K^-{}'$ and $\mathbf{q}_K^i{}'$ terms are gathered to the right. Leveraging the structure of $\mathbf{H}_K'$ and $\mathbf{R}_K'$, (4.12) becomes

$$\begin{aligned}
J' &= \mathsf{trace}((\mathbf{I} - \mathbf{H}_K^T\mathbf{L}_K^T)\mathbf{S}_K^+(\mathbf{I} - \mathbf{L}_K\mathbf{H}_K)\mathbf{P}_K^-{}') \\
&\quad + \mathsf{trace}(2\mathbf{P}_K^-(\mathbf{H}_K^T\mathbf{L}_K^T - \mathbf{I})\mathbf{S}_K^+\mathbf{L}_K\mathbf{H}_K') \\
&\quad + \mathsf{trace}(\mathbf{L}_K^T\mathbf{S}_K^+\mathbf{L}_K\mathbf{R}_K') + \sum_{i=1}^{M} \int_0^{t_K^-} \mathbf{u}^i(t)^T \mathbf{Q}_\mathbf{u} \mathbf{u}^i(t)' \, dt, \\
&= \mathsf{trace}((\mathbf{I} - \mathbf{H}_K^T\mathbf{L}_K^T)\mathbf{S}_K^+(\mathbf{I} - \mathbf{L}_K\mathbf{H}_K)\mathbf{P}_K^-{}') \\
&\quad + \sum_{i=1}^{M} \mathsf{trace}\left[(2\mathbf{P}_K^-(\mathbf{H}_K^T\mathbf{L}_K^T - \mathbf{I})\mathbf{S}_K^+\mathbf{L}_K)_i \left(\frac{d\mathbf{H}_K^i}{d\mathbf{q}_K^i}\right)\right]^T \mathbf{q}_K^i{}' \\
&\quad + \sum_{i=1}^{M} \mathsf{trace}\left[(\mathbf{L}_K^T\mathbf{S}_K^+\mathbf{L}_K)_{ii} \left(\frac{d\mathbf{R}_K^i}{d\mathbf{q}_K^i}\right)\right]^T \mathbf{q}_K^i{}' \\
&\quad + \sum_{i=1}^{M} \int_0^{t_K^-} \mathbf{u}^i(t)^T \mathbf{Q}_\mathbf{u} \mathbf{u}^i(t)' \, dt,
\end{aligned} \tag{4.13}$$

where $(\mathbf{L}_k^T\mathbf{S}_k^T\mathbf{L}_k)_{ii}$ denotes the (i,i) block of the $M \times M$ block matrix $\mathbf{L}_k^T\mathbf{S}_k^T\mathbf{L}_k$ and $(2\mathbf{A}\mathbf{P}_k\mathbf{A}^T(\mathbf{H}_k^T\mathbf{L}_k^T - \mathbf{I})\mathbf{S}_k\mathbf{L}_k)_i$ denotes the i'th column block of the $1 \times M$ block matrix $2\mathbf{A}\mathbf{P}_k\mathbf{A}^T(\mathbf{H}_k^T\mathbf{L}_k^T - \mathbf{I})\mathbf{S}_k\mathbf{L}_k$. The $()^-$ superscript on $\mathbf{q}_K^i{}'$ is dropped because the trajectory $\mathbf{q}^i(t)$ is smooth.

Now, if the same inner product and adjoint identity as in (4.10) are defined,

but over the time window $[t_{K-1}^+, t_K^-]$, and if in addition M adjoint vectors $\mathbf{r}^i(t)$ are defined over the same time window with the new adjoint identity

$$\langle\!\langle \mathbf{r}^i, \mathcal{M}(\mathbf{q}')^i \rangle\!\rangle_{t_{K-1}^+, t_K^-} = \langle\!\langle \mathcal{M}^*(\mathbf{r})^i, \mathbf{q}^{i\,\prime} \rangle\!\rangle_{t_{K-1}^+, t_K^-} + b^i, \tag{4.14a}$$

$$\langle\!\langle \mathbf{x}, \mathbf{y} \rangle\!\rangle_{t_{K-1}^+, t_K^-} \triangleq \int_{t_{K-1}^+}^{t_K^-} \mathbf{x}(t)^T \mathbf{y}(t)\, dt, \tag{4.14b}$$

$$\mathcal{M}^*(\mathbf{r})^i = -\frac{d\mathbf{r}^i(t)}{dt} - \mathbf{F}^T \mathbf{r}^i(t), \tag{4.14c}$$

$$b^i = (\mathbf{r}_K^{i-})^T \mathbf{q}_K^{i\,\prime} - (\mathbf{r}_{K-1}^{i+})^T \mathbf{q}_{K-1}^{i\,\prime}, \tag{4.14d}$$

then by taking

$$\mathcal{L}^*(\mathbf{S}) = 0, \qquad \mathcal{M}^*(\mathbf{r})^i = 0, \tag{4.15a}$$

$$(\mathbf{S}_K^-)^T = (\mathbf{I} - \mathbf{H}_K^T \mathbf{L}_K^T) \mathbf{S}_K^+ (\mathbf{I} - \mathbf{L}_K \mathbf{H}_K), \tag{4.15b}$$

$$\begin{aligned} \mathbf{r}_K^{i-} = \mathsf{trace} \left[(2\mathbf{P}_K^- (\mathbf{H}_K^T \mathbf{L}_K^T - \mathbf{I}) \mathbf{S}_K^+ \mathbf{L}_K)_i \left(\frac{d\mathbf{H}_K^i}{d\mathbf{q}_K^i} \right) \right] \\ + \mathsf{trace} \left[(\mathbf{L}_K^T \mathbf{S}_K^+ \mathbf{L}_K)_{ii} \left(\frac{d\mathbf{R}_K^i}{d\mathbf{q}_K^i} \right) \right], \end{aligned} \tag{4.15c}$$

and applying (4.10), (4.11), and (4.14), (4.13) is transformed to:

$$\begin{aligned} J' = \mathsf{trace}((\mathbf{S}_{K-1}^+)^T \mathbf{P}_{K-1}^{+\,\prime}) + \sum_{i=1}^{M} \int_0^{t_K^-} \mathbf{u}^i(t)^T \mathbf{Q}_\mathbf{u} \mathbf{u}^i(t)'\, dt \\ + \sum_{i=1}^{M} \Big(\int_{t_{K-1}^+}^{t_K^-} \mathbf{r}^i(t)^T \mathcal{B}(\mathbf{u}')^i\, dt + (\mathbf{r}_{K-1}^{i+})^T \mathbf{q}_{K-1}^{i\,\prime} \Big). \end{aligned} \tag{4.16}$$

Equation (4.16) bears a strong resemblance to (4.12), the only differences being the shifted time indices on the first term and the additional third and fourth terms in (4.16). Thus, the procedures between (4.12) and (4.16) can be repeated sequentially, backward in time, to repeatedly transform J'. In general for a given

measurement interval $[t_{k-1}^+, t_k^+]$,

$$\mathbf{S}_k^- = (\mathbf{I} - \mathbf{H}_k^T \mathbf{L}_k^T)\mathbf{S}_k^+(\mathbf{I} - \mathbf{L}_k \mathbf{H}_k), \tag{4.17a}$$

$$\begin{aligned}\mathbf{r}_k^{i-} = \mathbf{r}_k^{i+} &+ \mathsf{trace}\left[(2\mathbf{P}_k^-(\mathbf{H}_k^T\mathbf{L}_k^T - \mathbf{I})\mathbf{S}_k^+\mathbf{L}_k)_i\left(\frac{d\mathbf{H}_k^i}{d\mathbf{q}_k^i}\right)\right] \\ &+ \mathsf{trace}\left[(\mathbf{L}_k^T\mathbf{S}_k^+\mathbf{L}_k)_{ii}\left(\frac{d\mathbf{R}_k^i}{d\mathbf{q}_k^i}\right)\right],\end{aligned} \tag{4.17b}$$

$$\mathbf{S}_{k-1}^+ \leftarrow \mathbf{S}_k^- \quad \text{via } \frac{d\mathbf{S}(t)}{dt} = -\mathbf{A}^T\mathbf{S}(t) - \mathbf{S}(t)\mathbf{A}, \tag{4.17c}$$

$$\mathbf{r}_{k-1}^{i+} \leftarrow \mathbf{r}_k^{i-} \quad \text{via } \frac{d\mathbf{r}^i(t)}{dt} = -\mathbf{F}^T\mathbf{r}^i(t), \tag{4.17d}$$

and the transformed J' is

$$\begin{aligned}J' = \mathsf{trace}((\mathbf{S}_{k-1}^+)^T\mathbf{P}_{k-1}^{+\,\prime}) &+ \sum_{i=1}^{M}\int_0^{t_K^-}\mathbf{u}^i(t)^T\mathbf{Q}_\mathbf{u}\mathbf{u}^i(t)'\,dt \\ &+ \sum_{i=1}^{M}\Big(\int_{t_{k-1}^+}^{t_K^-}\mathbf{r}^i(t)^T\mathcal{B}(\mathbf{u}')^i\,dt + (\mathbf{r}_{k-1}^{i+})^T\mathbf{q}_{k-1}^{i\,\prime}\Big).\end{aligned} \tag{4.18}$$

Eventually, the J' transformation reaches time $t = 0$, where $\mathbf{P}_0' = 0$ and $\mathbf{q}_0^{i\,\prime} = 0$. The final J' equation is

$$\begin{aligned}J' &= \sum_{i=1}^{M}\int_0^{t_K^-}\mathbf{r}^i(t)^T\underbrace{\mathcal{B}(\mathbf{u}')^i}_{\mathbf{G}\mathbf{u}^i(t)'} + \mathbf{u}^i(t)^T\mathbf{Q}_\mathbf{u}\mathbf{u}^i(t)'\,dt \\ &= \sum_{i=1}^{M}\int_0^{t_K^-}(\underbrace{\mathbf{G}^T\mathbf{r}^i(t) + \mathbf{Q}_\mathbf{u}\mathbf{u}^i(t)}_{\nabla_{\mathbf{u}^i(t)}J})^T\mathbf{u}^i(t)'\,dt,\end{aligned} \tag{4.19}$$

which is in the necessary form to obtain the local gradient information. This gradient information is then used by an iterative optimization method such as steepest decent, conjugate gradient, or limited memory BFGS to update the initial nominal control trajectories.

After the optimal solution is found, either the optimal control trajectories or the resulting optimal vehicle trajectories can be sent to the sensor vehicles. However typically it is more practical to broadcast the waypoints taken along the

optimal vehicle trajectories, and let the sensor vehicle's on-board auto-pilot decides the control. Since the waypoints are taken from feasible vehicle trajectories, the control calculated by the auto-pilot would also be feasible.

To recap, we started with (4.7d), which was not in the correct form as in (4.8) to obtain the gradient information. Through defining the proper adjoint identities (4.10), (4.11), and (4.14), leveraging (4.7c), (4.7e), and (4.7f), trace identities, and through the correct setting of $\mathcal{L}^*(\mathbf{S})$, $\mathbf{S}_k^-$, $\mathcal{M}^*(\mathbf{r})^i$, and $\mathbf{r}_k^{i-}$ in (4.17), the J' equation is transformed a piece at a time until the final form in (4.19) is achieved, allowing the necessary gradient extraction.

Note that $\mathbf{P}(t)$ is updated at each measurement time in the forward march; it is thus natural that the adjoint variables $\mathbf{S}(t)$ and $\mathbf{r}^i(t)$ are similarly updated at each measurement time in the backward march. Also note that even though the initial $\mathbf{u}^i(t)$ and $\mathbf{q}^i(t)$ are continuous, because the update to $\mathbf{u}^i(t)$ is derived from $\mathbf{r}^i(t)$, which is now piece-wise continuous like $\mathbf{P}(t)$, the newly updated $\mathbf{u}^i(t)$ in the second iteration of the optimization will also be piece-wise continuous, and so will $d\mathbf{q}^i(t)/dt$. Nevertheless, $\mathbf{q}^i(t)$ is still continuous, and the continuity assumption on $\mathbf{q}^i(t)$ in the formulation still holds.

4.5 DAO extensions

4.5.1 Generalization

For clarity sake, we restricted the DAO formulation in §4.4 to a specific cost function, linear dynamics, and identical dynamics and sensors in all vehicles. The generalizations of the these restrictions are discussed here.

Generalize Cost Function

The vehicle penalty portion in (4.6) is not restricted to be quadratic and penalizes only $\mathbf{u}^i(t)$, other types of penalties can be incorporated. In general,

$$J = \mathsf{trace}(\mathbf{TP}_F) + \sum_{i=1}^{M} \Big(\int_0^{t_K^-} g^i(\mathbf{q}^i(t), \mathbf{u}^i(t))\, dt + h^i(\mathbf{q}_K^i) \Big). \tag{4.20}$$

Note if $g^i(\cdot,\cdot)$ and $h^i(\cdot)$ are quadratic, one would have the standard Linear Quadratic Regulator familiar in the controls community, where $g^i(\cdot,\cdot)$ is the state and control trajectory penalties and $h^i(\cdot)$ the terminal state penalty. Also this formulation allows the possibility to penalize each vehicle differently. Without re-deriving, the modifications to DAO are described in the following.

Suppose the perturbation of $g^i(\mathbf{q}^i(t),\mathbf{u}^i(t))$ and $h^i(\mathbf{q}^i_K)$ can be written as

$$\begin{aligned} g^i(\mathbf{q}^i(t),\mathbf{u}^i(t))' &= \left(\frac{\partial g^i(\mathbf{q}^i(t),\mathbf{u}^i(t))}{\partial \mathbf{q}^i(t)}\right)^T \mathbf{q}^i(t)' \\ &\quad + \left(\frac{\partial g^i(\mathbf{q}^i(t),\mathbf{u}^i(t))}{\partial \mathbf{u}(t)^i}\right)^T \mathbf{u}^i(t)', \\ h^i(\mathbf{q}^i_K)' &= \left(\frac{dh^i(\mathbf{q}^i_K)}{d\mathbf{q}^i_K}\right)^T {\mathbf{q}^i_K}'. \end{aligned}$$

The local cost function gradient is now expressed as

$$\nabla_{\mathbf{u}^i(t)} J = \mathbf{G}^T \mathbf{r}^i(t) + \frac{\partial g^i(\mathbf{q}^i(t),\mathbf{u}^i(t))}{\partial \mathbf{u}^i(t)}. \tag{4.21}$$

The terminal state penalty simply changes the starting condition of $\mathbf{r}^{i-}_K$, and the state trajectory penalty introduces an additional forcing to the $\mathbf{r}^i$ evolution equation. The evolution equation for $\mathbf{r}^i(t)$ now changed into

$$\frac{d\mathbf{r}^i(t)}{dt} = -\mathbf{F}^T \mathbf{r}^i(t) - \frac{\partial g^i(\mathbf{q}^i(t),\mathbf{u}^i(t))}{\partial \mathbf{u}^i(t)} \tag{4.22a}$$

$$\begin{aligned} \mathbf{r}^{i-}_K &= \frac{dh^i(\mathbf{q}^i_K)}{d\mathbf{q}^i_K} + \mathsf{trace}\left[(\mathbf{L}^T_K \mathbf{S}^+_K \mathbf{L}_K)_{ii}\left(\frac{d\mathbf{R}^i_K}{d\mathbf{q}^i_K}\right)\right] \\ &\quad + \mathsf{trace}\left[(2\mathbf{P}^-_K(\mathbf{H}^T_K \mathbf{L}^T_K - \mathbf{I})\mathbf{S}^+_K \mathbf{L}_K)_i \left(\frac{d\mathbf{H}^i_K}{d\mathbf{q}^i_K}\right)\right]. \end{aligned} \tag{4.22b}$$

Nonlinearities

So far we assumed linear time-invariant models and used the Kalman Filter for the covariance update. However DAO can be easily extended to work with nonlinear time-varying models.

Dealing with nonlinear vehicle model is simple, one needs to propagate and store the trajectory of each $\mathbf{q}^i(t)$. During the vehicle adjoint propagation and cost

function gradient evaluation, the linearized $\mathbf{F}(\mathbf{q}^i(t), \mathbf{u}^i(t))$ and $\mathbf{G}(\mathbf{q}^i(t), \mathbf{u}^i(t))$ are evaluated along the trajectory of $\mathbf{q}^i(t)$ and $\mathbf{u}^i(t)$.

However having a nonlinear system model introduces additional complexities. Namely, this requires linearized $\mathbf{A}(\hat{\mathbf{x}}(t))$ and $\mathbf{B}(\hat{\mathbf{x}}(t))$ during the time-update phase and the linearized measurement operator $\mathbf{H}(\hat{\mathbf{x}}_k^-)$ during the measurement-update phase. Since the best estimate at current time is $\hat{\mathbf{x}}_0^+$, naturally a forecast made from $\hat{\mathbf{x}}_0^+$ would be used as $\hat{\mathbf{x}}(t)$. Furthermore, the best guess on the future measurements $\mathbf{y}$ at time $k = 1$ is obtained from the $\hat{\mathbf{x}}(t)$ forecast at t_1, $\hat{\mathbf{x}}_1^-$, and apply the output operator $\mathbf{H}_1$ to predict $\mathbf{y}_1$. Using this predicted measurement in state update equation would yield $\hat{\mathbf{x}}_1^+ = \hat{\mathbf{x}}_1^-$, since the innovation would be zero. In general if $\hat{\mathbf{x}}_k^+$ is given, one would follow similar logics to forecast $\hat{\mathbf{x}}_k^+$ and update with $\mathbf{y}_{k+1} = \mathbf{H}_{k+1}\hat{\mathbf{x}}_{k+1}^-$, and show that $\hat{\mathbf{x}}_{k+1}^+ = \hat{\mathbf{x}}_{k+1}^-$. Thus by induction it is clear to see a forecast of $\hat{\mathbf{x}}_0^+$ is sufficient to produces $\hat{\mathbf{x}}_k^+$ and $\hat{\mathbf{x}}_k^-$ needed for linearizing $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{H}$. Therefore with nonlinear models, one also needs to know $\hat{\mathbf{x}}_0$ in order to propagate and store $\mathbf{P}_k$ $\mathbf{q}^i(t)$, *and* $\hat{\mathbf{x}}(t)$.

Multiple Vehicle and Sensor Types

In some instances, vehicles with different dynamical properties carrying different instruments are deployed. For example, in weather forecasting one aircraft may carry a Doppler Radar to give a global view of the weather system while several UAVs are carrying barometers, temperature sensors, and humility sensors to provide pin-point measurements. By replacing $\mathbf{F}$ and $\mathbf{G}$ with $\mathbf{F}^i$ and $\mathbf{G}^i$, allowing $\mathbf{H}^i$ and $\mathbf{R}^i$ to be different for each vehicle, and adjust the size of block matrices of $(\mathbf{L}_k^T\mathbf{S}_k^T\mathbf{L}_k)_{ii}$ and $(2\mathbf{A}\mathbf{P}_k\mathbf{A}^T(\mathbf{H}_k^T\mathbf{L}_k^T - \mathbf{I})\mathbf{S}_k\mathbf{L}_k)_i$ appropriately based on the size of $\mathbf{R}^i$ and $\mathbf{H}^i$, DAO is capable in handling multiple vehicles with different vehicle dynamics and sensor types.

Routine Measurements

In some situations supplemental routine measurements are also available in the future. These measurements typically come from existing stationary sensor networks that makes routine measurements (e.g. sensor buoys), while some other

times these measurements come from non-controllable sources (i.e. wind data from boats). Routine measurements should be incorporated into the AO formulation to avoid redundant measurements. This is done by augmenting $\mathbf{H}_k$ and $\mathbf{R}_k$ such that

$$\mathbf{H}_k \triangleq \begin{bmatrix} \mathbf{H}_k^r & 0 \\ 0 & \mathbf{H}_k^{AO} \end{bmatrix}, \qquad \mathbf{R}_k \triangleq \begin{bmatrix} \mathbf{R}_k^r & 0 \\ 0 & \mathbf{R}_k^{AO} \end{bmatrix}. \tag{4.23}$$

Note when performing perturbation analysis, the perturbation of $\mathbf{H}_k^r$ and $\mathbf{R}_k^r$ to $\mathbf{q}_k^i$ are zero since we cannot control the routine measurement placements.

4.5.2 Extension to discrete- and continuous-time

The mixed continuous/discrete time DAO formulation can be extended to pure discrete or continuous time. In both cases, the generalizations in §4.5.1 equally applied with appropriate modifications.

Discrete time

If the continuous propagation of $\mathbf{P}(t)$ in (4.4) and $\mathbf{q}^i(t)$ in (4.2) from time t_k to t_{k+1} are allowed to be propagated discretely in one time-step, then the propagation and update can be combined into a discrete evolution equation for the posterior estimation covariance and vehicle states. The conversion to discrete time propagation is done through an explicit Euler approximation of the continuous time propagation. Similarly, this can be done for adjoint propagations. Using the same explicit Euler approximation and bearing in mind the propagations are *backward* in time, the combined adjoint propagation and update equations are

$$\mathbf{S}_{k-1} = \mathbf{A}_D^T(\mathbf{I} - \mathbf{H}_k^T\mathbf{L}_k^T)\mathbf{S}_k(\mathbf{I} - \mathbf{L}_k\mathbf{H}_k)\mathbf{A}_D, \tag{4.24a}$$

$$\begin{aligned}\mathbf{r}_{k-1}^i = \mathbf{F}_D^T\mathbf{r}_k^i &+ \mathsf{trace}\left[(\mathbf{L}_k^T\mathbf{S}_k\mathbf{L}_k)_{ii}\left(\frac{d\mathbf{R}_k^i}{d\mathbf{q}_k^i}\right)\right] \\ &+ \mathsf{trace}\left[(2\mathbf{A}_D\mathbf{P}_k\mathbf{A}_D^T(\mathbf{H}_k^T\mathbf{L}_k^T - \mathbf{I})\mathbf{S}_k\mathbf{L}_k)_i\left(\frac{d\mathbf{H}_k^i}{d\mathbf{q}_k^i}\right)\right],\end{aligned} \tag{4.24b}$$

where $()_D$ denotes the discretized version of the continuous-time counterpart. The cost function gradient is also appropriately redefined as

$$J' = \sum_{i=1}^{M} \sum_{k=0}^{K-1} (\nabla_{\mathbf{u}_k^i} J)^T \mathbf{u}_k^{i\,\prime} \tag{4.25a}$$

$$\nabla_{\mathbf{u}_k^i} J = \mathbf{G}_D^T \mathbf{r}_k^i + \mathbf{Q}_{uD} \mathbf{u}_k^i, \tag{4.25b}$$

where final time index is $K-1$ because the zero-order-hold assumption for the control $\mathbf{u}_k^i$. These results are consistent with our earlier work in Zhang & Bewley (2011).

Equation (4.24) only holds within the time interval $[t_0, t_K]$, where measurement updates are performed. It is necessary to propagate $\mathbf{S}_F$ to $\mathbf{S}_K$ using the discretized version of (4.17c)

$$\mathbf{S}_{k-1} = \mathbf{A}^T \mathbf{S}_k \mathbf{A} \tag{4.26}$$

and set $\mathbf{S}_{K-1} = \mathbf{S}_K$ before using (4.24). The starting conditions are $\mathbf{S}_F = \mathbf{T}$, and $\mathbf{r}_{K-1}^i = 0$. Note the starting conditions for $\mathbf{S}_{K-1}$ and $\mathbf{r}_{K-1}^i$ are one time-step off. This is due to a theoretical gap that exists when converting a continuous-time adjoint equation to discrete-time. This inconsistency vanishes as the time-step becomes small, as $\mathbf{S}_{K-1}$ and $\mathbf{r}_{K-1}^i$ approach $\mathbf{S}_K$ and $\mathbf{r}_K^i$.

Continuous time

The continuous-time Kalman Filter is rarely used in practice for practical reasons. Nevertheless, for theoretical completeness a DAO formulation also exists for a continuous-time KF. There are two approaches to this problem. The first approach is to perform the entire DAO formulation presented before with the continuous-time KF propagation of $\mathbf{P}(t)$. A perhaps simpler second approach, inspired by the work in Smith & Robers (1978) which reconciles the discrete and continuous-time KF, is by substituting

$$\mathbf{A}_D \to \mathbf{I} + \Delta t \mathbf{A}, \qquad \mathbf{F}_D \to \mathbf{I} + \Delta t \mathbf{F},$$

$$\mathbf{H}_k \to \mathbf{H}(t_k), \qquad \mathbf{R}_k \to \mathbf{R}(t_k)/\Delta t,$$

into (4.24). After pulling the $\mathbf{S}_k - \mathbf{S}_{k-1}$ and $\mathbf{r}_k^i - \mathbf{r}_{k-1}^i$ term to the L.H.S. and dividing by Δt, the limit $\Delta t \to 0$ is taken to yield the continuous-time equivalent of DAO:

$$\frac{d\mathbf{S}}{dt} = -\mathbf{A}^T\mathbf{S} - \mathbf{S}\mathbf{A} + \mathbf{S}\mathbf{P}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}\mathbf{S}, \tag{4.27a}$$

$$\begin{aligned}\frac{d\mathbf{r}^i}{dt} = &-\mathbf{F}^T\mathbf{r}^i + \mathsf{trace}\left[(2\mathbf{P}\mathbf{S}\mathbf{P}\mathbf{H}^T\mathbf{R}^{-1})_i\left(\frac{d\mathbf{H}^i}{d\mathbf{q}^i}\right)\right] \\ &- \mathsf{trace}\left[(\mathbf{R}^{-1}\mathbf{H}\mathbf{P}\mathbf{S}\mathbf{P}\mathbf{H}^T\mathbf{R}^{-1})_{ii}\left(\frac{d\mathbf{R}^i}{d\mathbf{q}^i}\right)\right].\end{aligned} \tag{4.27b}$$

Again (4.27) only holds within the time interval $[t_0, t_K]$. It is necessary to propagate $\mathbf{S}_F$ to $\mathbf{S}_K$ using (4.17c). The starting conditions remain unchanged, $\mathbf{S}_F = \mathbf{T}$, and $\mathbf{r}_K^i = 0$.

4.6 Experimental Results for an Environmental Plume

We shall demonstrate the capabilities of DAO through a numerical experiment, an environmental plume problem, where DAO is used to plan sensor vehicle waypoints for estimating a convection driven contaminant plume.

Estimating and forecasting the evolution of environmental flows is one of today's most visible grand challenge problems. Such algorithms must grapple with a myriad of challenges involved in extending these concepts to high-dimensional discretizations of infinite-dimensional systems.

In problems of this scale, classical estimation solutions like the Kalman Filter (KF; see Kalman (1960) and Anderson & Moore (1979)) and the Extended Kalman Filter (EKF) simply do not work, due both to their poor scaling with problem size [they require the propagation of an $n \times n$ covariance matrix] as well as their inability to represent non-Gaussian statistics. Ensemble methods, specifically the Ensemble Kalman Filter (EnKF), are standard–though (apparently) not well known–algorithms for addressing these challenges in an efficient stochastic

manner: by propagating several perturbed candidate system trajectories and inferring the principle directions of the state uncertainties from the distribution of these ensemble members in phase space. In practice, a remarkably small ensemble sample [$O(100)$] is typically sufficient for capturing the principle directions of the state estimate uncertainties even in high-dimensional problems [with $n \gtrsim O(10^6)$]. A brief EnKF overview is presented in §1.3.

Motivated by the recent Gulf coast oil spill and the Icelandic ash cloud, the EnKF and DAO are combined in this experiment to test the estimation of a convection-driven environmental plume problem: the 2D Navier-Stokes Equation (NSE) couple with a passive scalar equation. By exciting the fluid system with additive low-frequency forcing, this passive scalar equation exhibits the same characteristic motion as an environmental plume. The governing equations are written

$$\frac{\partial \mathbf{v}}{\partial t} = -\mathbf{v} \cdot \nabla \mathbf{v} + \nu \nabla^2 \mathbf{v} + \frac{1}{\rho} \nabla p + \mathbf{w}_{\mathbf{v}} \tag{4.28a}$$

$$\frac{\partial \phi}{\partial t} = -\mathbf{v} \cdot \nabla \phi + \kappa \nabla^2 \phi + \mathbf{w}_{\phi}, \tag{4.28b}$$

with velocity vector field $\mathbf{v}$, passive scalar field ϕ, density ρ, kinematic viscosity ν, scalar pressure field p, and diffusion constant κ. $\mathbf{w}_{\mathbf{v}}$ and $\mathbf{w}_{\phi}$ are external forcing. The measurement operator (linear with respect to the flow state) is a function of the sensor positions, in the point-mass vehicle states $\mathbf{q}_k^i$:

$$h(\mathbf{q}_k^1, \cdots, \mathbf{q}_k^M, \mathbf{v}_k) = \mathbf{H}(\mathbf{q}_k^1, \cdots, \mathbf{q}_k^M)^T \mathbf{v}_k. \tag{4.29}$$

The pseudo-spectral code developed in Bewley *et al.* (2001) is used for numerical simulation on a 64×64 uniform square grid. The EnKF assimilates the measurements and provide an estimate, while DAO determines where the measurements should be taken in the future. In all numerical experiments the "truth" simulation uses an identical model (with different initial conditions and random forcing) running in parallel with the EnKF, enabling us to gage the estimator performance.

Although simulations are done using periodic domains, the estimation al-

gorithm is isolated to an "region of interest,"; which is effectively defined through the weighting matrix $\mathbf{T}$. The square, periodic domain is nondimensionalized with width $L_x = L_y = 2$, and a estimation subregion of interest of width $R_x = R_y = 1.35$. In DAO, the targeting matrix $\mathbf{T}$ is chosen to focus on the same subregion. To maintain numerical computation stability in the flow calculation, a relatively small time-step ($\Delta t = 0.005$) is used, whereas measurements are taken every $\tau_{meas} = 0.15$ time units. Given the large difference between Δt and τ_{meas}, the mixed continuous/discrete time version of DAO is used.

In this application DAO is structured in a receding-horizon framework: every three measurements DAO receives a new ensemble approximation of $\mathbf{P}$ from the estimator. Using the current vehicle states and assumed initial control trajectories (both prescribed during initialization), DAO optimizes the vehicle waypoints over the next six measurement times (because $\tau_{meas} = 0.15$, the event horizon is $T_F = T_K = 0.9$); however, only the waypoints for the first three measurement times are used before DAO is called again to compute new waypoints. Because of the complex dynamics of this system, a few additional computation approximations are introduced to make this problem feasible. Since the propagation of $\mathbf{P}$ using the full system dynamics in (4.28) would be too computationally intensive, we approximate full covariance propagation with the simplified continuous-time differential equation

$$\frac{d\mathbf{P}(t)}{dt} = 0.2\mathbf{P}(t). \tag{4.30}$$

This, with an additional assumption that $\mathbf{P}$ is diagonally dominant (effectively, that the spatial cross-correlations are negligible), dramatically reduces the computational and storage requirements for $\mathbf{P}$, resulting in a feasible implementation of DAO. These approximations are perhaps physically justified by the small event horizon considered in the present simulations.

To quantify the quality of the estimate, we consider the steady-state, infinite-time averaged statistics of the absolute error, defined by Bewley & Protas (2004) as follows:

$$\mathrm{Errn}(\hat{d}, d_{tru}) = \int_{R_x}\int_{R_y} (\,\hat{d} - d_{tru}\,)^2 dy\, dx, \tag{4.31}$$

where $()_{tru}$ corresponding the "truth" values. Long-time averages of this measure (applied to both the velocity field and the scalar) are used to approximate the expected value, $\mathcal{E}[\mathrm{Errn}(\hat{d}(t), d_{tru}(t))] \triangleq 1/T \int_0^T \mathrm{Errn}(\hat{d}(t), d(t))\, dt$, at statistical steady state.

Fig. 4.2 and 4.3 compare the error statistics using three different adaptive observation strategies: sensors following a random walk (dot-dashed), a decentralized AO strategy in which sensors gravitate to the centroid of their voronoi cells (dashed), and the present DAO strategy (solid). Fig. 4.4 provides typical examples of the estimate, determined by the EnKF, and the waypoints optimized by DAO.

The results demonstrate that a significant improvement in the estimate can be accomplished via path planning: 42% in the flow [a reduction of the error norm defined in (4.31) from 0.388 to 0.206] and 47% in the scalar [a reduction from 0.449 to 0.259]. Furthermore, this improvement is not just a consequence of the movement, as the case with randomly walking sensor vehicles (with the same mean velocity) significantly underperforms as compared with the optimized DAO solution.

4.7 Conclusions and Future Work

This paper presents DAO, a new Adaptive Observation method that rigorously incorporates vehicle dynamics in the formulation. The DAO formulation is based on mixed continuous/discrete-time scheme, where the system propagations are continuous, but measurement updates are discrete. By minimizing a cost containing both a measure of the forecast quality and vehicle control effort penalties *iteratively* using adjoint analysis, DAO is able to balance control effort with uncertainty reduction. Various DAO generalization including generalized cost, nonlinear time-varying systems, and taking extraneous routine measurements into account are discussed. Also for theoretical completeness, the mixed DAO continuous/discrete time scheme is extended to pure discrete- and continuous time. The results for a randomly forced 2D environmental plume demonstrate a significant reduction in the error of the forecast over less deliberate sensor routing strategies.

A promising new Hybrid (variational/Kalman) Ensemble Smoother (HEnS) algorithm for state estimation in large-scale systems has recently been proposed by our group (see Cessna & Bewley (2010)). As DAO effectively combines an ensemble representation with variational sweeps to perform optimizations over the *near future* to solve the adaptive optimization problem, HEnS similarly combines an ensemble representation with variational sweeps to perform optimizations over the *recent past* to solve the data assimilation problem. Preliminary tests have show that this new algorithm for data assimilation outperforms the more traditional EnKF in the presence of substantial non-Gaussian uncertainties. Both HEnS and DAO will thus be applied in concert to the present representative test problem in the near future.

The ultimate goal of this work is to use DAO in conjunction with a data assimilation scheme such as HEnS, or the EnKF, in real-time, to estimate and forecast large complex system. It is our vision that one day small, deployable sensor vehicles would be deployed in an environmental contaminant disaster as first-responders to adaptively take environmental readings and facilitate the forecast of the contaminant movement.

4.8 Acknowledgments

The authors gratefully acknowledge the generous financial support of the National Security Education Center (NSEC) at Los Alamos National Laboratory (LANL), the many helpful conversations with Frank Alexander (LANL), and Yoav Freund (UCSD) for providing access to the Greenlight Cluster at the University of California, San Diego.

This chapter, in full, is a reprint of the material as it was accepted for publication at the *IEEE American Controls Conference, June 2011* (Zhang, Colburn & Bewley (2011*a*), "Estimation and Adaptive Observation of Environmental Plumes." In *ACC 2011: Proceedings of the 2011 conference on American Control Conference.* IEEE Press, to appear.) and later augmented with additional material as it may appear in Journal of Fluid Mechanics, 2011 (Zhang, Colburn & Bew-

ley (2011*b*), "Estimation and Adaptive Observation of Environmental Plumes." Drafted for submission to *Journal of Fluid Mechanics*.). The dissertation author contributed substantially to the authorship and theoretical development of this paper.

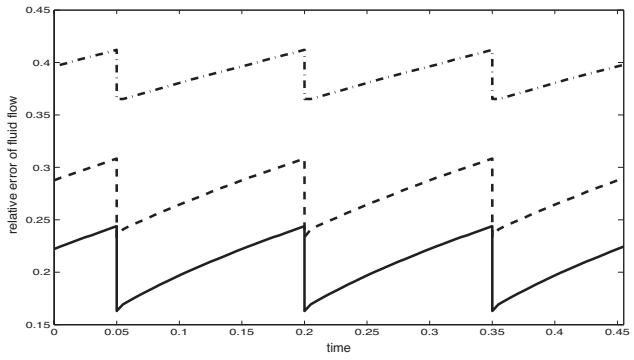


Figure 4.2: The norm of the relative error of the flow velocities, as a function of time, following three AO strategies: (1) sensors following a random walk (dot-dashed) (2) stationary sensors uniformly distributed over the domain (dashed), and (3) sensor trajectories provided by DAO (solid). The error increases between measurements, and decreases at the EnKF measurement updates, thus creating the "saw-tooth" shape in the error plot.

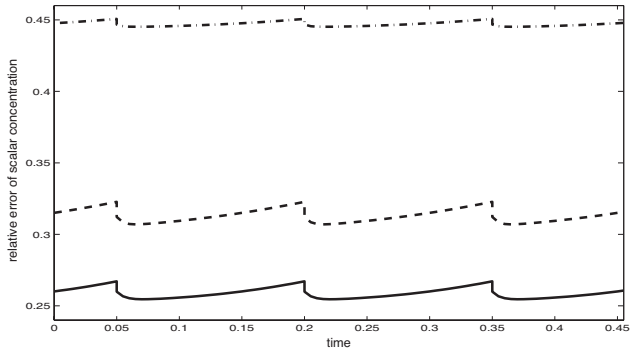


Figure 4.3: The norm of the relative error of the scalar component, as a function of time, with the sensor motion as described in Fig. 4.2. The evolution of the scalar is primarily "convective" (that is, it is primarily driven by the flow velocities); note that the error of the scalar component thus dips slightly after each measurement update, due apparently to the improved velocity estimate.

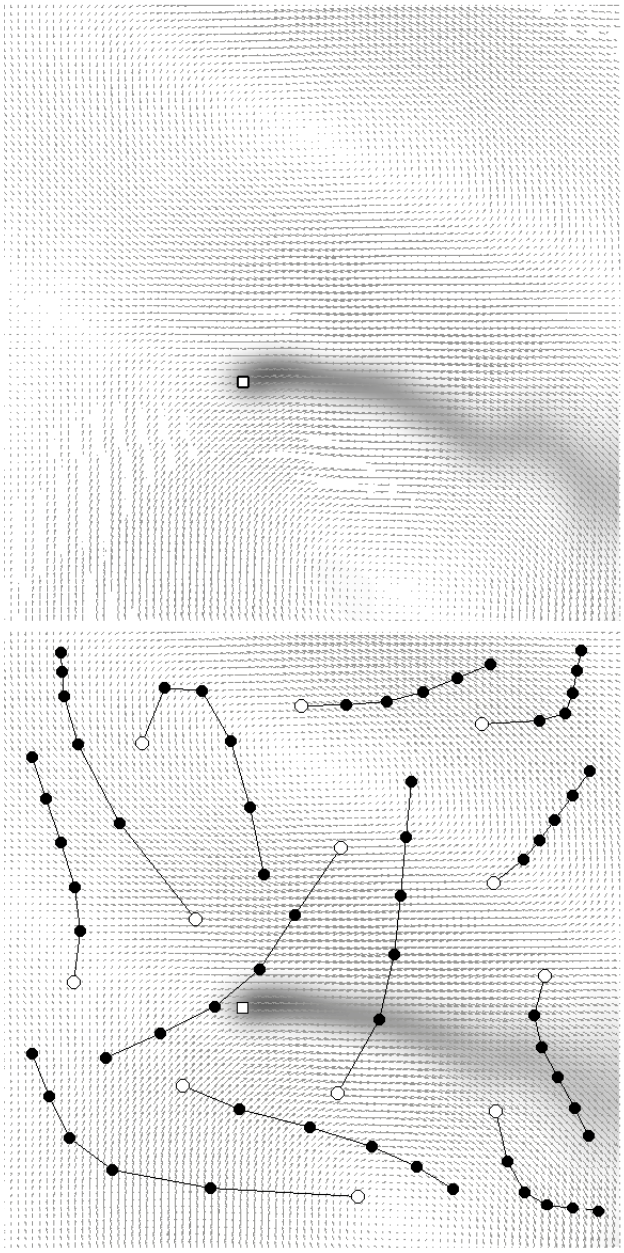


Figure 4.4: A typical example of the truth simulation (top), reconstruction (bottom, provided by EnKF), and sensor trajectories generated by DAO. The contaminant source (an empty square) is located near the center of the domain. Note that, by eye, the velocities and the scalar distributions of the estimate and truth are indeed quite similar. Future waypoints for sensors are all well distributed over the domain following the present approach, and are designed in particular to fill in valuable pieces of information in areas of concentrated uncertainty.

Appendix A

Relevant Theorems from Probability

A.1 Slutsky's Theorem

Theorem A.1.1 (Slutsky, 1925; Gut, 2005, Theorem 11.4, p. 249) *Let* X_1, X_2, ... *and* Y_1, Y_2, ... *be sequences of random variables. Suppose that*

$$X_n \to X \quad \textit{and} \quad Y_n \to a \quad \textit{as} \quad n \to \infty, \tag{A.1}$$

where a *is some constant. Then*

$$X_n + Y_n \to X + a, \tag{A.2}$$

$$X_n - Y_n \to X - a, \tag{A.3}$$

$$X_n \bullet Y_n \to X \bullet a, \tag{A.4}$$

$$\frac{X_n}{Y_n} \to \frac{X}{a} \quad \textit{for } a \neq 0, \tag{A.5}$$

as $n \to \infty$.

A.2 Cramér-Rao Lower Bound

The appropriate interpretation of equation (2.1) is critical to understanding the differences in cost functions (2.2a) and (2.24a). The following theorem clarifies the interpretation and is reprinted here for convenience.

Theorem A.2.1 [Goodwin & Payne 1977, Theorem 1.3.1] *(The Cramér-Rao Inequality) Let* $\{P_\theta : \theta \in \Theta\}$ *be a family of distributions on a sample space* Ω, $\Theta \subset \mathbb{C}^p$, *and suppose that, for each* Θ, P_θ *is defined by a density* $p_{_{Y|\Theta}}(\cdot|\theta)$. *Then subject to certain regularity conditions, the covariance of any unbiased estimator* $g(Y)$ *of* Θ *satisfies the inequality*

$$\operatorname{cov}(g) \geq \mathbf{I}_F^{-1} \tag{A.6}$$

where

$$\operatorname{cov}(g) = \mathrm{E}_{_{Y|\Theta}}\{(\mathrm{g}(Y) - \theta)(\mathrm{g}(Y) - \theta)^H\} \tag{A.7}$$

and where $\mathbf{I}_F$, *known as Fisher's Information Matrix (FIM), is defined by*

$$\mathbf{I}_F = \mathrm{E}_{_{Y|\Theta}}\left\{\left(\frac{\partial \ln \mathrm{p}(Y|\theta)}{\partial \theta}\right)^H \left(\frac{\partial \ln \mathrm{p}(Y|\theta)}{\partial \theta}\right)\right\}. \tag{A.8}$$

Proof A.2.1 *Since* $\mathrm{g}(Y)$ *is an unbiased estimator of* θ, *we have*

$$\mathrm{E}_{_{Y|\Theta}}\{\mathrm{g}(Y)\} = \theta, \tag{A.9}$$

i.e.,

$$\int_\Omega \mathrm{g}(y)p(y|\theta)\,dy = \theta, \quad so \quad \frac{\partial}{\partial\theta}\int_\Omega \mathrm{g}(y)p(y|\theta)\,dy = I \tag{A.10}$$

Assuming sufficient regularity to allow differentiation under the integral sign,

$$\int_\Omega \mathrm{g}(y)\frac{\partial p(y|\theta)}{\partial\theta}\,dy = I, \quad so \quad \int_\Omega \mathrm{g}(y)\frac{\partial \ln p(y|\theta)}{\partial\theta}p(y|\theta)\,dy = I, \tag{A.11}$$

i.e.,

$$\mathrm{E}_{_{Y|\Theta}}\left\{\mathrm{g}(Y)\frac{\partial \ln p(Y|\theta)}{\partial\theta}\right\} = I. \tag{A.12}$$

Also

$$\mathrm{E}_{Y|\Theta}\left\{\frac{\partial \ln p(Y|\theta)}{\partial\theta}\right\} = \int_{\Omega} \frac{\partial \ln p(y|\theta)}{\partial\theta} p(y|\theta)\, dy = \int_{\Omega} \frac{\partial p(y|\theta)}{\partial\theta}\, dy$$
$$= \frac{\partial}{\partial\theta}\int_{\Omega} p(y|\theta)\, dy = \frac{\partial}{\partial\theta}(1) = 0. \tag{A.13}$$

Thus, using equations (A.7), (A.8), (A.9), (A.12), *and* (A.13), *the covariance of* $\mathrm{g}(Y)$ *and* $\partial \ln p(y|\theta)/\partial\theta$ *can be written as*

$$\mathrm{E}_{Y|\Theta}\left\{ \begin{bmatrix} (\mathrm{g}(Y)-\theta) \\ \left(\dfrac{\partial \ln p(Y|\theta)}{\partial\theta}\right)^{H} \end{bmatrix} \left[(\mathrm{g}(Y)-\theta)^{H} \left(\frac{\partial \ln p(Y|\theta)}{\partial\theta}\right)\right] \right\} = \begin{bmatrix} \mathrm{cov}(g) & I \\ I & \mathbf{I}_F \end{bmatrix}. \tag{A.14}$$

By the definition of covariance matrices, (A.14) *is positive semi-definite; hence*

$$\begin{bmatrix} I : & -\mathbf{I}_F^{-1} \end{bmatrix} \begin{bmatrix} \mathrm{cov}(g) & I \\ I & \mathbf{I}_F \end{bmatrix} \begin{bmatrix} I \\ -\mathbf{I}_F^{-1} \end{bmatrix} \geq 0$$
$$\mathrm{cov}(g) - \mathbf{I}_F^{-1} \geq 0. \tag{A.15}$$

Bibliography

Alonso, A.A., Kevrekidis, I.G., Banga, J.R. & Frouzakis, C.E. 2004 Optimal sensor location and reduced order observer design for distributed process systems. *Computers & Chemical Engineering* **28** (1-2), 27–35.

Alspach, D. & Sorenson, HW 1972 Nonlinear Bayesian estimation using Gaussian sum approximations. *Automatic Control, IEEE Transactions on* **17** (4), 439–448.

Anderson, B.D.O. & Moore, J.B. 1979 *Optimal filtering*. Prentice-Hall Englewood Cliffs, NJ.

Anderson, J.L. 2007 An adaptive covariance inflation error correction algorithm for ensemble filters. *Tellus* **59** (2), 210–224.

Anderson, J.L. 2009 Spatially and temporally varying adaptive covariance inflation for ensemble filters. *Tellus A* **61** (1), 72–83.

Anderson, Jeffrey L. & Anderson, Stephen L. 1999 A Monte Carlo Implementation of the Nonlinear Filtering Problem to Produce Ensemble Assimilations and Forecasts. *Monthly Weather Review* **127** (12), 2741–2758.

Arulampalam, S., Maskell, S., Gordon, N. & Clapp, T. 2002 A tutorial on particle filters for on-line non-linear/nongaussian Bayesian tracking. *IEEE Transactions on Signal Processing* **50**, 174–188.

Bagheri, S., Henningson, D.S., Hœpffner, J. & Schmid, P.J. 2009 Input-output analysis and control design applied to a linear model of spatially developing flows. *Applied Mechanics Reviews* **62**, 020803.

Baker, N.L. & Daley, R. 2000 Observation and background adjoint sensitivity in the adaptive observation-targeting problem. *Quarterly Journal of the Royal Meteorological Society* **126** (565), 1431–1454.

Bar-Shalom, Y., Li, X.R. & Kirubarajan, T. 2001 *Estimation with applications to tracking and navigation*. Wiley-Interscience.

Bertsekas, D. P. *et al.* 2001 *Dynamic Programming and Optimal Control*, 2nd edn. Athena Scientific.

Bewley, T.R. 2001 Flow control: new challenges for a new renaissance. *Progress in Aerospace Sciences* **37** (1), 21–58.

Bewley, T.R. & Liu, S. 1998 Optimal and robust control and estimation of linear paths to transition. *Journal of Fluid Mechanics* **365**, 305–349.

Bewley, T.R., Moin, P. & Temam, R. 2001 DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms. *Journal of Fluid Mechanics* **447**, 179–225.

Bewley, T.R. & Protas, B. 2004 Skin friction and pressure: the "footprints" of turbulence. *Physica D: Nonlinear Phenomena* **196** (1-2), 28–44.

Bewley, T. & Sharma, A.S. 2010 A tractable framework for grid-based Bayesian estimation of nonlinear low-dimensional systems with sparse nongaussian PDFs. Availible at http://fccr.ucsd.edu/.

Bishop, C.H. 2000 (submitted) Estimation theory and adaptive observing techniques. *Quarterly Journal of the Royal Meteorological Society* .

Bishop, C.H., Etherton, B.J. & Majumdar, S.J. 2001 Adaptive Sampling with the Ensemble Transform Kalman Filter. Part I: Theoretical Aspects. *Monthly Weather Review* **129** (3), 420–436.

Blackwell, D. 1956 An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics* **6** (1), 1–8.

Blackwell, D. & Girshick, M.A. 1954 *Theory of games and statistical decisions*. John Wiley & Sons, Inc.

Bouttier, F. & Courtier, P. 1999 Data assimilation concepts and methods. *ECMWF Meteorological Training Course Lecture Series* **14**.

Boyd, S., El Ghaoui, L., Feron, E. & Balakrishnan, V. 1994 *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial Mathematics.

Buizza, R. & Montani, A. 1999 Targeting Observations Using Singular Vectors. *Journal of the Atmospheric Sciences* **56** (17), 2965–2985.

Bullo, F. & Cortes, J. 2004 Adaptive and distributed coordination algorithms for mobile sensing networks. *Cooperative Control* pp. 431–434.

Butala, M.D., Yun, J., Chen, Y., Frazin, R.A. & Kamalabadi, F. 2008 Asymptotic convergence of the ensemble Kalman filter. In *15th IEEE International Conference on Image Processing.*, pp. 825–828.

Cesa-Bianchi, N., Freund, Y., Haussler, D., Helmbold, D.P., Schapire, R.E. & Warmuth, M.K. 1997 How to use expert advice. *Journal of the ACM (JACM)* **44** (3), 427–485.

Cesa-Bianchi, N. & Lugosi, G. 2003 Potential-Based Algorithms in On-Line Prediction and Game Theory. *Machine Learning* **51**, 239–261.

Cesa-Bianchi, N. & Lugosi, G. 2006 *Prediction, learning, and games*. Princeton University Press. Princeton, NJ.

Cessna, J. & Bewley, T. 2010 A hybrid (variational/Kalman) ensemble smoother for the estimation of nonlinear high-dimensional discretizations of PDE systems. Availible at http://fccr.ucsd.edu.

Cessna, Joseph B. 2010 The Hybrid Ensemble Smoother (HEnS) & Noncartesian Computational Interconnects. PhD thesis, University of California, San Diego.

Chaudhuri, K., Freund, Y. & Hsu, D. 2009 A parameter-free hedging algorithm. *Advances in Neural Information Processing Systems* **22**, 297–305.

Chen, K. & Rowley, C. 2010 Optimal actuator and sensor placement in the linearized complex Ginzburg-Landau system. *Bulletin of the American Physical Society* **55**.

Chevalier, M., Hœpffner, J., Bewley, T.R. & Henningson, D.S. 2006 State estimation in wall-bounded flow systems. Part 2. Turbulent flows. *Journal of Fluid Mechanics* **552**, 167–187.

Chomaz, J.M., Huerre, P. & Redekopp, L.G. 1987 Models of hydrodynamic resonances in separated shear flows. In *Proc. 6th Symposium on Turbulent Shear Flows*, pp. 321–326.

Colburn, C.H. & Bewley, T.R. 2011 GEnKF: A hybrid game-theoretic, ensemble-kalman approach to online learning of non-stationary, high-dimensional sequences. Drafted for publication 2011.

Colburn, C.H., Cessna, J.B. & Bewley, T.R. 2011*a* State estimation in wall-bounded flow systems. Part 3. The Ensemble Kalman Filter. *Journal of Fluid Mechanics* Accepted for publication 2011.

Colburn, C.H., Zhang, D.D. & Bewley, T.R. 2011*b* Gradient-based optimization methods for sensor & actuator placement in LTI systems. Submitted for publication 2011.

Cortes, J., Martinez, S., Karatas, T. & Bullo, F. 2004 Coverage control for mobile sensing networks. *Robotics and Automation, IEEE Transactions on* **20** (2), 243–255.

Cover, Thomas M. & Thomas, Joy A. 2006 *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience.

De Farias, DP, De Oliveira, MC & Geromel, JC 2001 Mixed H_2/H_∞ Control of Flexible Structures. *Mathematical Problems in Engineering* **6** (6), 557–598.

Evensen, G. 1994 Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of geophysical research* **99** (C5), 10143.

Evensen, G. 2003 The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics* **53** (4), 343–367.

Evensen, G. 2009*a* *Data assimilation: The ensemble Kalman filter*. Springer Verlag.

Evensen, G. 2009*b* The ensemble Kalman filter for combined state and parameter estimation. *IEEE Control Systems Magazine* **29** (3), 83–104.

Faulds, A.L. & King, B.B. 2000 Sensor location in feedback control of partial differential equation systems. In *Proceedings of the 2000 IEEE International Conference on Control Applications, 2000.*, pp. 536–541. IEEE.

Feller, William 1968 *An Introduction to Probability Theory and Its Applications*. Wiley-Interscience.

Flaherty, P., Jordan, M., Arkin, A. *et al.* 2006 Robust design of biological experiments. *Advances in Neural Information Processing Systems* **18**, 363–370.

Freund, Y. 2011 http://seed.ucsd.edu/mediawiki/index.php/TrackingChaos.

Freund, Y. & Schapire, R. 1995 A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pp. 23–37. Springer.

Furrer, R. & Bengtsson, T. 2007 Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *Journal of Multivariate Analysis* **98** (2), 227–255.

Giles, M.B. 1998 On adjoint equations for error analysis and optimal grid adaptation in CFD. *Frontiers of Computational Fluid Dynamics* pp. 155–169.

Giles, M.B. & Pierce, N.A. 2000 An introduction to the adjoint approach to design. *Flow, Turbulence and Combustion* **65** (3), 393–415.

Goodwin, G.C. & Payne, R.L. 1977 *Dynamic system identification: experiment design and data analysis*. Academic Press.

Gut, A. 2005 *Probability: A Graduate Course*. Springer Verlag.

Hannan, J. 1957 Approximation to Bayes risk in repeated play. *Contributions to the Theory of Games* **3**, 97–139.

Hart, S. & Mas-Colell, A. 2000 A simple adaptive procedure leading to correlated equilibrium. *Econometrica* **68** (5), 1127–1150.

Hart, S. & Mas-Colell, A. 2001 A General Class of Adaptive Strategies. *Journal of Economic Theory* **98** (1), 26 – 54.

Hassibi, B., Sayed, A.H. & Kailath, T. 1999 *Indefinite-Quadratic Estimation and Control: A Unified Approach to H2 and H[infinity] Theories*. Society for Industrial Mathematics.

Hiramoto, K., Doki, H. & Obinata, G. 2000 Optimal sensor/actuator placement for active vibration control using explicit solution of algebraic Riccati equation. *Journal of Sound and Vibration* **229** (5), 1057–1075.

Hœpffner, J., Chevalier, M., Bewley, T.R. & Henningson, D.S. 2005 State estimation in wall-bounded flow systems. Part 1. Perturbed laminar flows. *Journal of Fluid Mechanics* **534**, 263–294.

Högberg, M., Bewley, T.R. & Henningson, D.S. 2003 Linear feedback control and estimation of transition in plane channel flow. *Journal of Fluid Mechanics* **481**, 149–175.

Horn, R.A. & Johnson, C.R. 1990 *Matrix Analysis*. Cambridge University Press.

Houtekamer, P.L. & Mitchell, H.L. 2001 A sequential ensemble Kalman filter for atmospheric data assimilation. *Monthly Weather Review* **129** (1), 123–137.

Jameson, A., Martinelli, L. & Pierce, N.A. 1998 Optimum aerodynamic design using the Navier–Stokes equations. *Theoretical and Computational Fluid Dynamics* **10** (1), 213–237.

Jazwinski, A.H. 1970 *Stochastic Processes and Filtering Theory*. Academic Press.

Jensen, J.L.W.V. 1906 Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica* **30** (1), 175–193.

Jimenez, J. & Moin, P. 1991 The minimal flow unit in near-wall turbulence. *Journal of Fluid Mechanics* **225**, 213–240.

Kailath, T. 1973 Some New Algorithms for Recursive Estimation in Constant Linear Systems. *IEEE Trans. Information Theory* **19**, 750–760.

Kailath, T. 1980 *Linear systems*. Prentice-Hall Englewood Cliffs, NJ.

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

Kalman, R.E. & Bucy, R.S. 1961 New results in linear filtering and prediction theory. *Journal of Basic Engineering* **83** (3), 95–108.

Kenney, C. & Hewer, G. 1990 The sensitivity of the algebraic and differential Riccati equations. *SIAM Journal on Control and Optimization* **28**, 50.

Kim, J. & Bewley, T.R. 2007 A linear systems approach to flow control. *Annu. Rev. Fluid Mech.* **39**, 383–417.

Kim, J., Moin, P. & Moser, R. 1987 Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics* **177** (1), 133–166.

Kwakernaak, H. & Sivan, R. 1972 *Linear Optimal Control Systems*. Wiley-Interscience New York.

Kwok, A. & Martinez, S. 2010 Unicycle coverage control via hybrid modeling. *Automatic Control, IEEE Transactions on* **55** (2), 528–532.

Langland, R.H. & Rohaly, G.D. 1996 Adjoint-Based Targeting of Observations for FASTEX Cyclones. *Defense Technical Information Center* .

Lauga, Eric & Bewley, Thomas R 2003 The decay of stabilizability with Reynolds number in a linear model of spatially developing flows. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **459** (2036), 2077–2095.

Laventall, Katie & Cortés, Jorge 2009 Coverage control by multi-robot networks with limited-range anisotropic sensory. *International Journal of Control* **82**, 1113–1121.

Li, Faming, Oliveira, Maurcio C. & Skelton, Robert E. 2009 Designing Instrumentation for Control. In *Model-Based Control: Bridging Rigorous Theory and Advanced Technology*, pp. 71–88. Springer US.

Littlestone, N. & Warmuth, MK 1989 The weighted majority algorithm. In *Proceedings of the 30th Annual Symposium on Foundations of Computer Science*, pp. 256–261. IEEE Computer Society.

Lo, J. 1972 Finite-dimensional sensor orbits and optimal nonlinear filtering. *Information Theory, IEEE Transactions on* **18** (5), 583–588.

Lorenz, E.N. 1963 Deterministic Nonperiodic Flow. *Journal of the Atmospheric Sciences* **20**, 130–141.

Majumdar, S.J., Bishop, C.H., Etherton, B.J. & Toth, Z. 2002*a* Adaptive Sampling with the Ensemble Transform Kalman Filter. Part II: Field Program Implementation. *Monthly Weather Review* **130** (5), 1356–1369.

Majumdar, S. J., Bishop, Craig H., Etherton, B. J. & Toth, Z. 2002*b* Adaptive Sampling with the Ensemble Transform Kalman Filter. Part II: Field Program Implementation. *Monthly Weather Review* **130**, 1356–1369.

Martinelli, F. 2009 Feedback Control of Turbulent Wall Flows. PhD thesis, Politecnico di Milano.

Martínez, S. & Bullo, F. 2006 Optimal sensor placement and motion coordination for target tracking. *Automatica* **42** (4), 661–668.

Martínez, Sonia, Cortés, Jorge & Bullo, Francesco 2007 Motion coordination with distributed information. *IEEE Control Systems Magazine* **27** (4), 75–88.

Petersen, K. B. & Pedersen, M. S. 2008 The matrix cookbook. Technical University of Denmark.

Porat, B. & Nehorai, A. 1996 Localizing vapor-emitting sources by moving sensors. *IEEE Transactions on Signal Processing* **44** (4), 1018 –1021.

Prudêncio, R. & Ludermir, T. 2005 Using machine learning techniques to combine forecasting methods. *AI 2004: Advances in Artificial Intelligence* pp. 277–286.

Rauch, H.E., Tung, F. & Striebel, C.T. 1965 Maximum Likelihood Estimates of Linear Dynamic Systems. *AIAA journal* **3** (8), 1445–1450.

Rogallo, R.S. 1981 Numerical Experiments in Homogeneous Turbulence. *Tech. Rep.*. NASA.

Rogallo, R.S. & Moin, P. 1984 Numerical simulation of turbulent flows. *Annual Review of Fluid Mechanics* **16** (1), 99–137.

Roussopoulos, K. & Monkewitz, P.A. 1996 Nonlinear modelling of vortex shedding control in cylinder wakes. *Physica D: Nonlinear Phenomena* **97** (1-3), 264–273.

Scherer, C., Gahinet, P. & Chilali, M. 2002 Multiobjective Output-feedback Control via LMI Optimization. *Automatic Control, IEEE Transactions on* **42** (7), 896–911.

Slutsky, E. 1925 Über stochastische Asymptoten und Grenzwerte. *Metron* **5** (3), 3–90.

Smith, M.W.A. & Robers, A.P. 1978 An exact equivalence between the discrete- and continuous-time formulations of the kalman filter. *Mathematics and Computers in Simulations* **20**, 102–109.

Sorenson, H.W. & Alspach, D.L. 1971 Recursive Bayesian estimation using Gaussian sums. *Automatica* **7** (4), 465–479.

Stanković, Miloš S. & Stipanović, Dušan M 2009 Stochastic extremum seeking with applications to mobile sensor networks. In *ACC'09: Proceedings of the 2009 conference on American Control Conference*, pp. 5622–5627. Piscataway, NJ, USA: IEEE Press.

Taylor, J. 1979 The Cramér-Rao Estimation Error Lower Bound Computation for Deterministic Nonlinear Systems. *Automatic Control, IEEE Transactions on* **24** (2), 343–344.

Uciński, D. 2005 *Optimal measurement methods for distributed parameter system identification*. CRC.

Von Neumann, J., Morgenstern, O., Kuhn, H.W. & Rubinstein, A. 1953 *Theory of games and economic behavior*. Princeton University Press. Princeton, NJ.

Vovk, V.G. 1990 Aggregating strategies. In *Annual Workshop on Computational Learning Theory: Proceedings of the third annual workshop on Computational learning theory*. Association for Computing Machinery, Inc, One Astor Plaza, 1515 Broadway, New York, NY, 10036-5701, USA,.

Vovk, V. 2009 Some comments on A parameter-free hedging algorithm by Chaudhuri, Freund, and Hsu.

Wang, X. & Bishop, C.H. 2003 A Comparison of Breeding and Ensemble Transform Kalman Filter Ensemble Forecast Schemes. *Journal of the Atmospheric Sciences* **60** (9), 1140–1158.

Yilmaz, Namik Kemal, Evangelinos, Constantinos, Lermusiaux, Pierre F.J. & Patrikalaks, Nicholas M. 2008 Path planning of autonomous underwater vehicles for adaptive sampling using mixed integer linear programming. *IEEE Journal of Oceanic Engineering* **33** (4), 522–537.

Zhang, D. & Bewley, T. 2011 Adaptive observation of stationary fields with mobile robotic systems. In *ACC 2011: Proceedings of the 2011 conference on American Control Conference*. IEEE Press, to appear.

Zhang, D., Colburn, C.H. & Bewley, T.R. 2011*a* Estimation and Adaptive Observation of Environmental Plumes. In *ACC 2011: Proceedings of the 2011 conference on American Control Conference*. IEEE Press, to appear.

Zhang, D.D., Colburn, C.H. & Bewley, T.R. 2011*b* Estimation and Adaptive Observation of Environmental Plumes. Drafted for submission to Journal of Fluid Mechanics.

Zhang, Fumin & Leonard, Naomi Ehrich 2010 Cooperative filters and control for cooperative exploration. *IEEE Transactions on Automatic Control* **55** (3), 650–663.

Zhou, K. & Doyle, J.C. 1998 *Essentials of Robust Control*. Prentice Hall New Jersey.

Zhou, K., Doyle, J.C. & Glover, K. 1996 *Robust and Optimal Control*. Prentice Hall Englewood Cliffs, NJ.