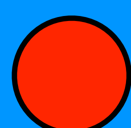
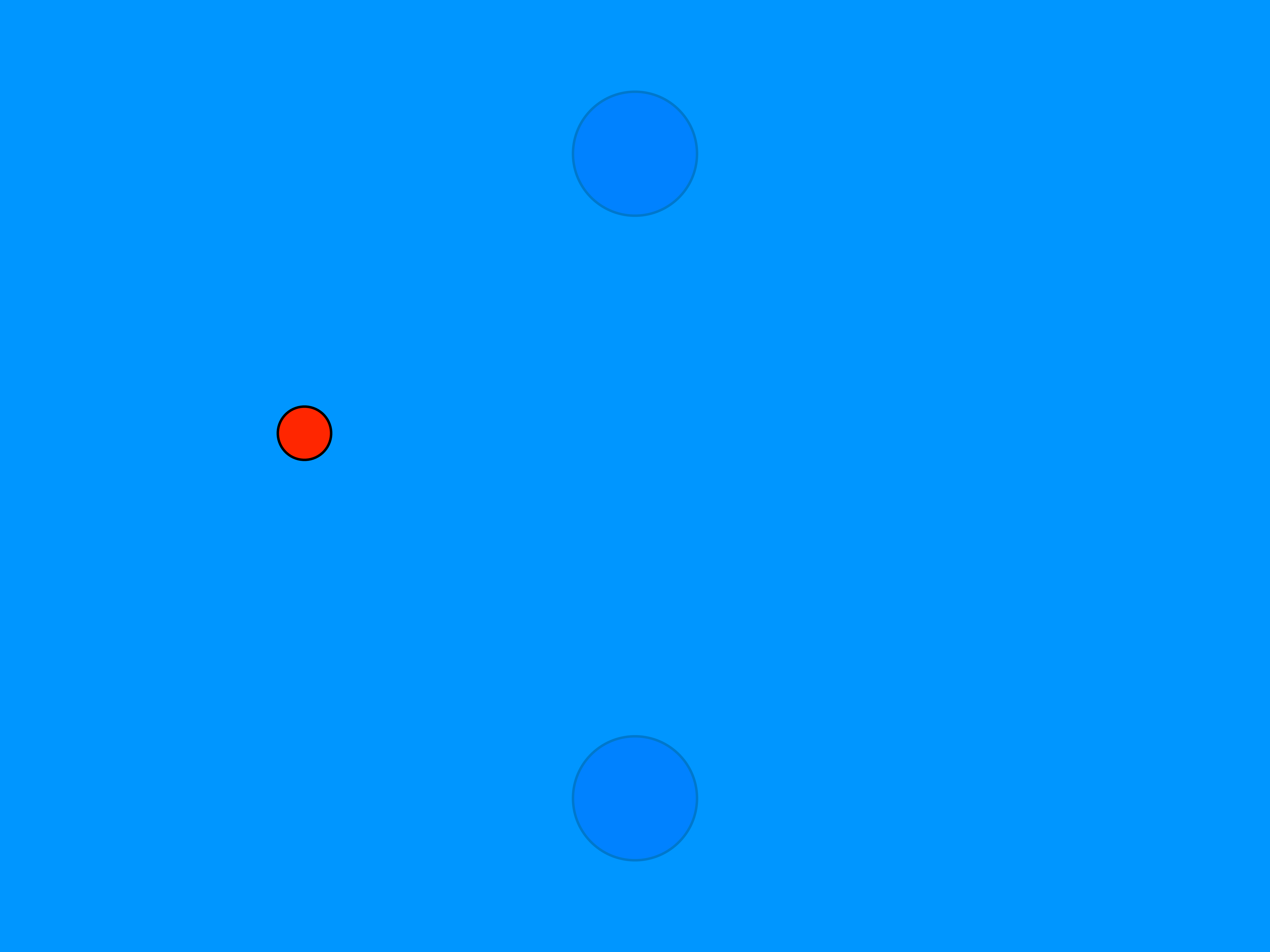
Lagrangian Data Assimilation and Manifold Detection for a Point-Vortex Model

Final Presentation

David Darmon, AMSC

Kayo Ide, AOSC, IPST, CSCAMM, ESSIC





Project Goals

Phase I: Data Assimilation

Implement and validate the extended Kalman, local ensemble transform Kalman, and particle filters for a point-vortex model with N_v vortices and N_d drifters.

Compare the performance of the filters in the two vortex, single drifter case.

Project Goals

Phase II: Observing System Design

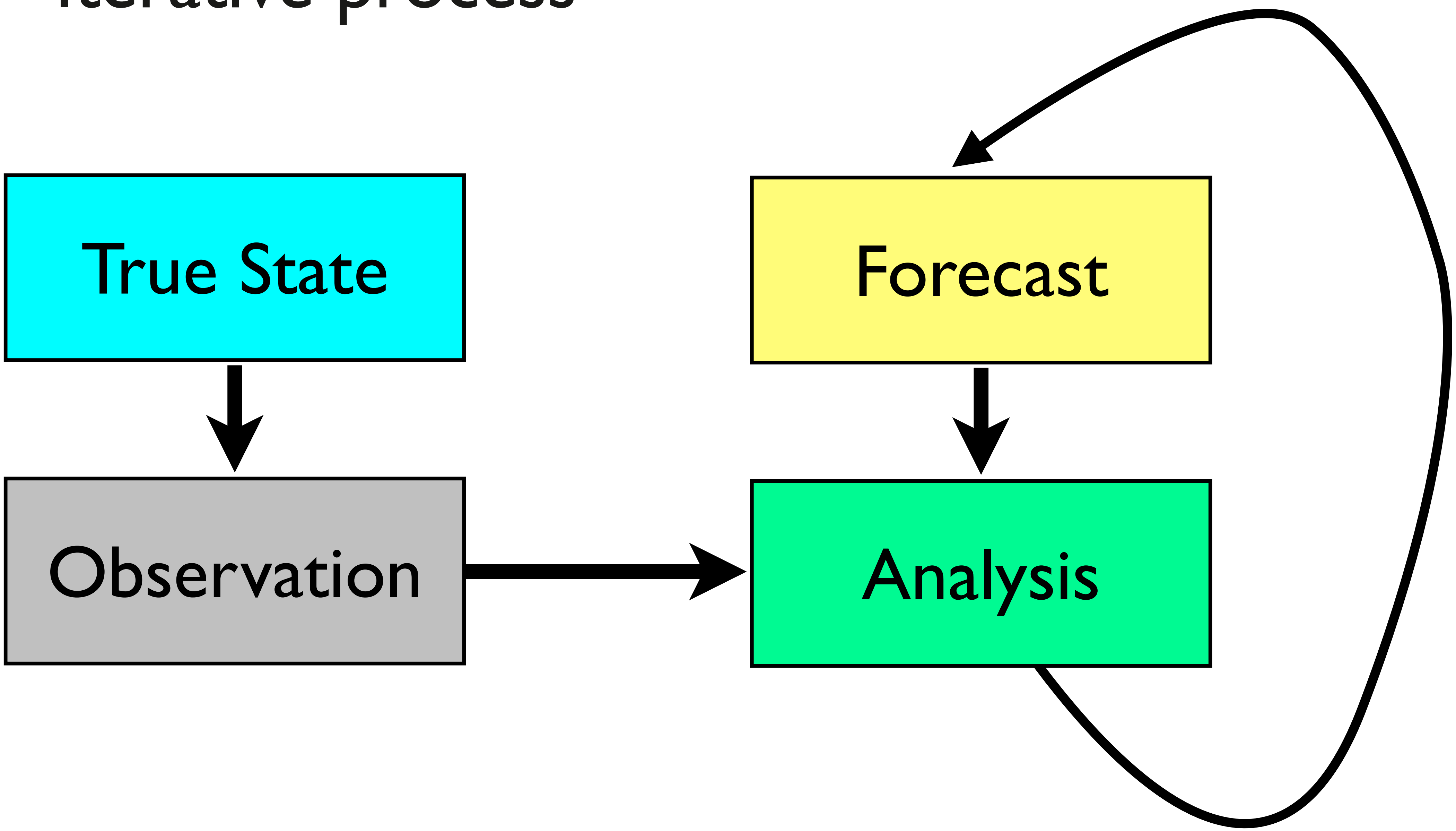
Implement methods for computing Mendoza and Mancho's *M* function and Finite-Time Lyapunov Exponents for manifold detection.

Use detected manifolds to intelligently place drifters *a priori* in a chaotic 3 vortex, single drifter case.

Background

Data Assimilation

Iterative process



Background

Data Assimilation

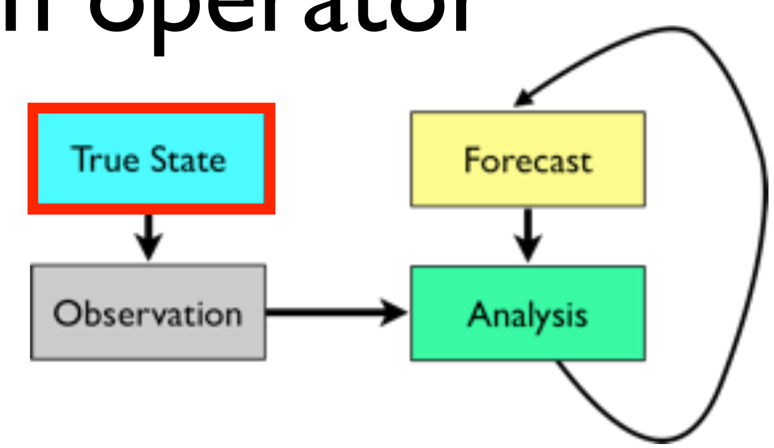
True System Dynamics

$$d\mathbf{x}^t = M(\mathbf{x}^t, t)dt + d\boldsymbol{\eta}^t$$

$\mathbf{x}^t$ – system state

$\boldsymbol{\eta}^t$ – Brownian motion

$M(\cdot, \cdot)$ – deterministic evolution operator



Background

Data Assimilation

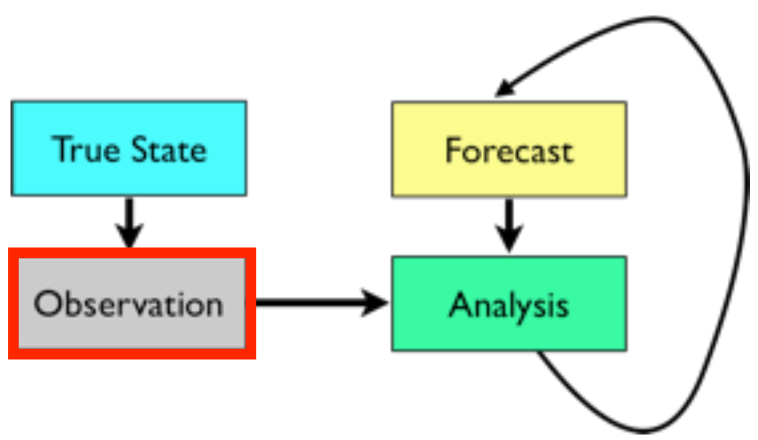
Observations

$$\mathbf{y}_k^o = h(\mathbf{x}^t(t_k)) + \epsilon_k^t$$

$\mathbf{y}_k^o$ – observation of system at time t_k

ϵ_k^t – Gaussian noise

$h(\cdot)$– observation operator



Background

Data Assimilation

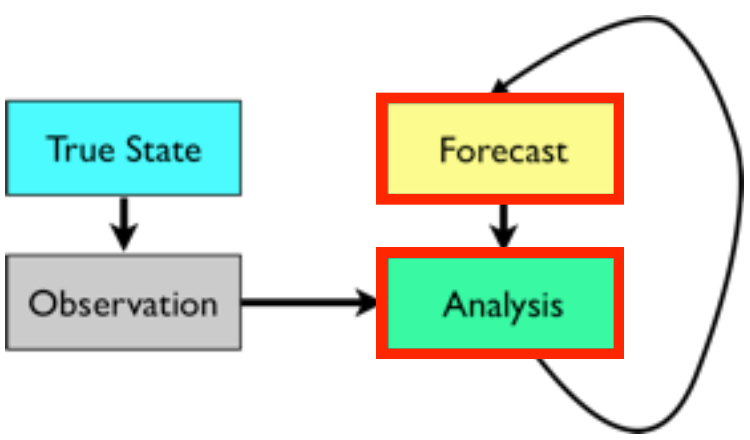
Forecast and Analysis

Evolve a forecasted state forward *somehow*

$$\frac{d}{dt}\mathbf{x}^f = f(\mathbf{x}^f, t, \boldsymbol{\eta}^f)$$

Perform analysis step upon receiving observation *somehow*

$$\mathbf{x}^a = g(\mathbf{x}^f, \mathbf{y}_k^o, \epsilon_k)$$



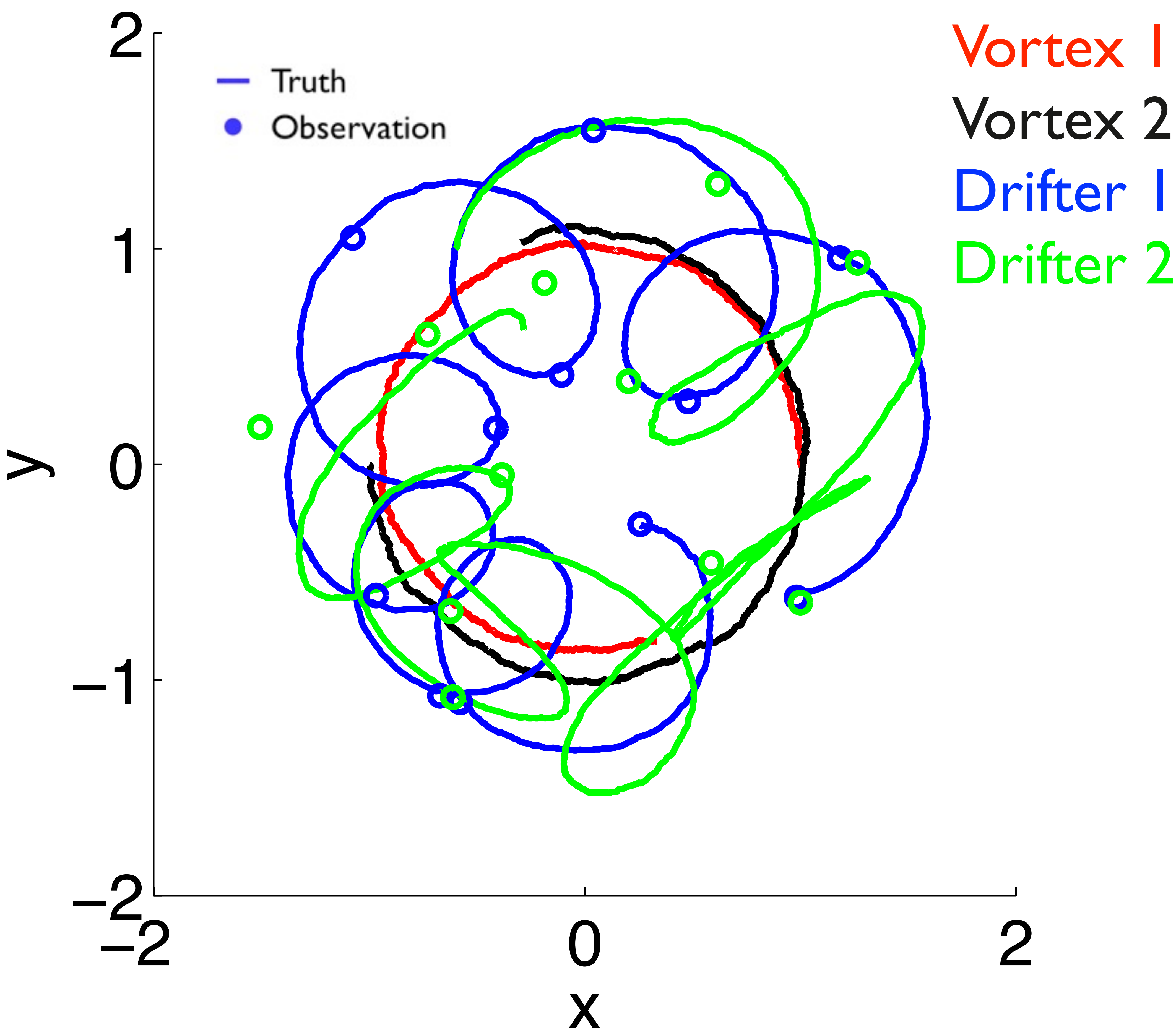
Background : Phase I

Lagrangian Data Assimilation

Consider joint state of flow (F) and drifters (D):

$$\mathbf{x}^t = \begin{pmatrix} \mathbf{x}_F^t \\ \mathbf{x}_D^t \end{pmatrix}$$

Q: Can we estimate $\mathbf{x}_F^t$ and $\mathbf{x}_D^t$ by observing $\mathbf{x}_D^t$ only?



Truth
Observation
Vortex 1
Vortex 2
Drifter 1
Drifter 2
y
x
2
1
0
−1
−2
−2
0
2

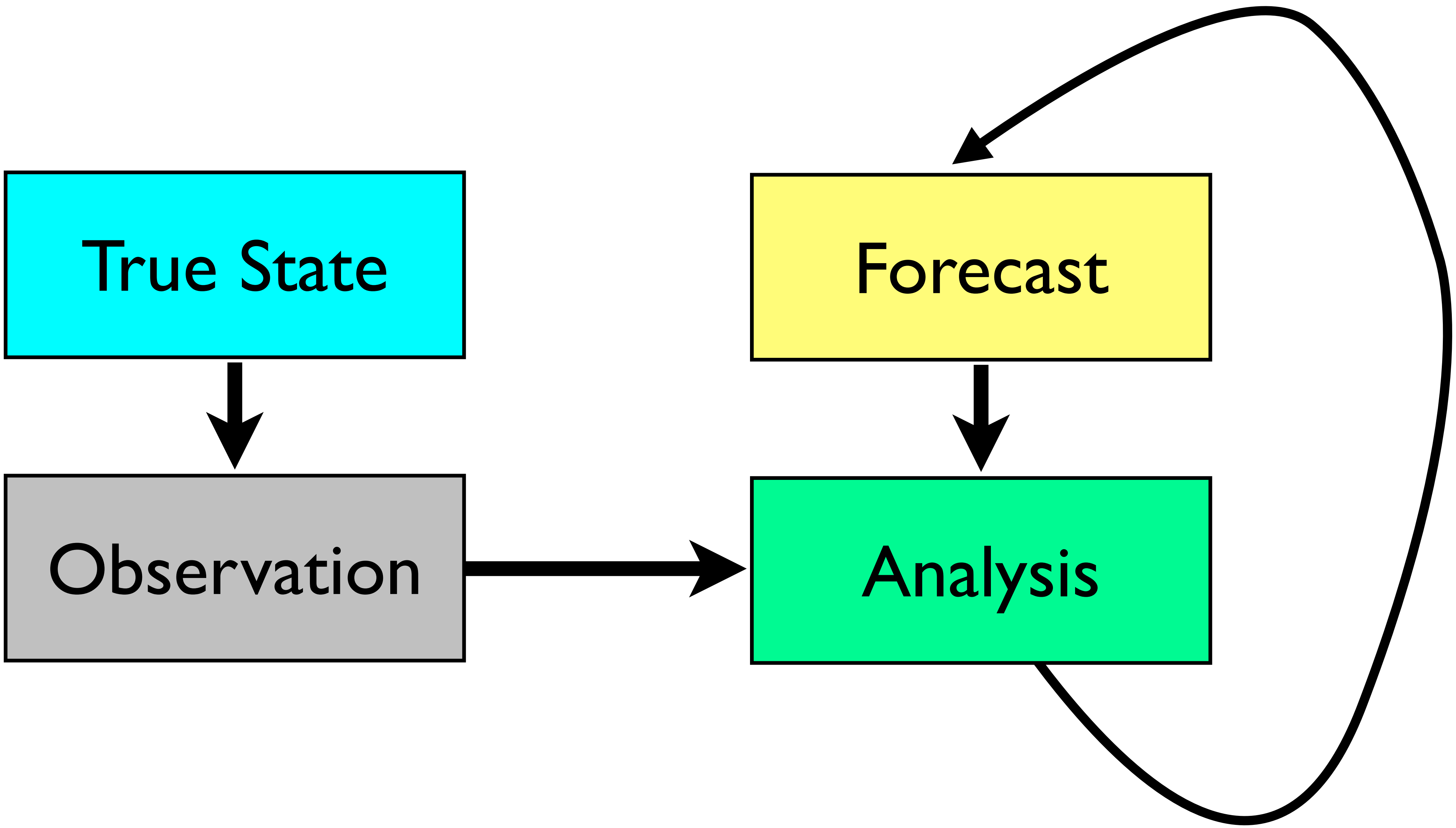
Background : Phase II

Observing System Design

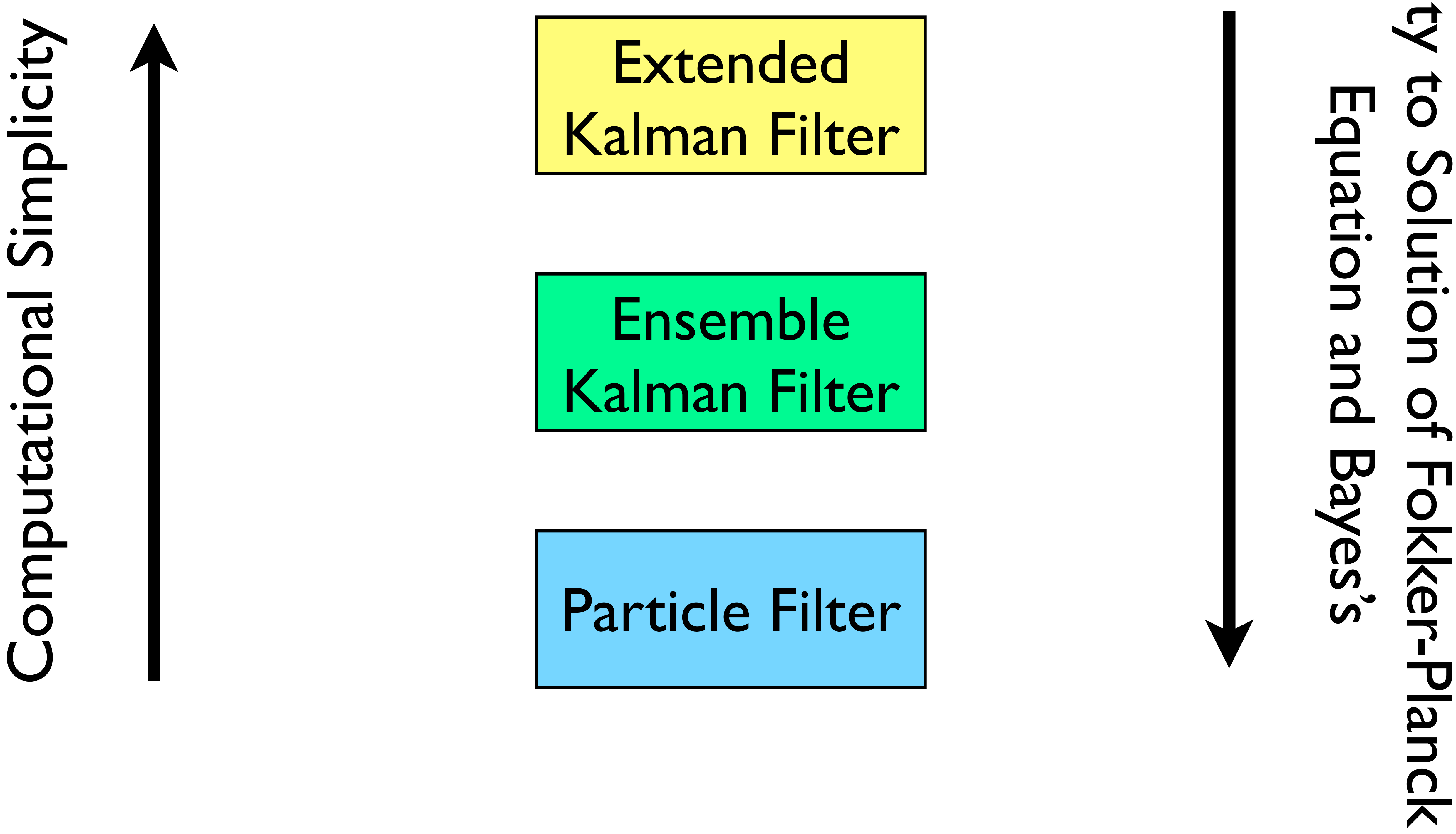
Deploying drifters is *expensive*
Obtaining measurements is *difficult*

Q: Can we design a system such that we optimally place the drifters to determine the location of the vortices?

Phase I



Approach, Phase I



Approach, Phase I

Lagrangian Data Assimilation

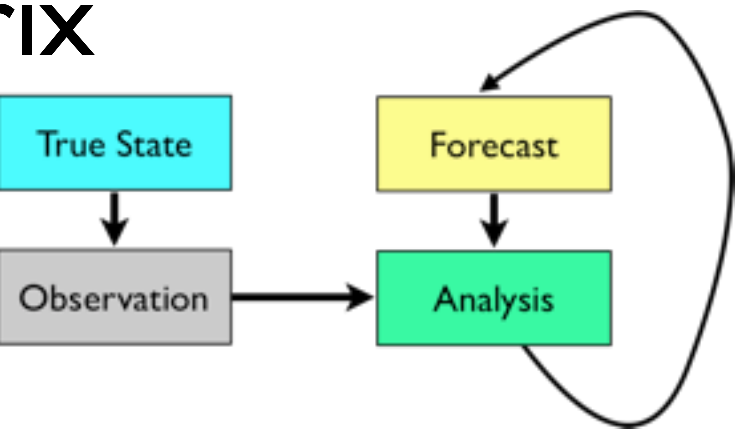
Extended Kalman Filter (EKF), Forecast

$$\frac{d}{dt}\mathbf{x}^f = M(\mathbf{x}^f, t)$$

$$\frac{d}{dt}\mathbf{P}^f = \mathbf{M}(t)\mathbf{P}^f + \mathbf{P}^f\mathbf{M}^T(t) + \mathbf{Q}$$

$\mathbf{Q}$ – covariance matrix from SDE

$\mathbf{M}(t) = J[M(\mathbf{x}, t)]\big|_{x=x^f}$ - Jacobian matrix of M



Approach, Phase I

Lagrangian Data Assimilation

Extended Kalman Filter (EKF), Analysis

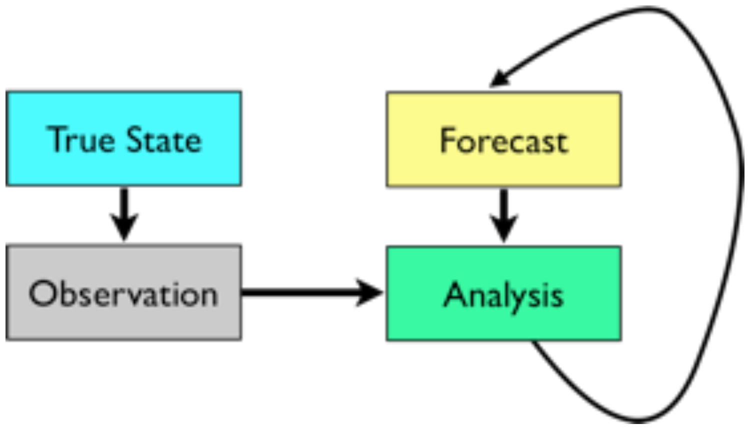
$$\mathbf{x}_k^a = \mathbf{x}^f(t_k) + \mathbf{K}_k(\mathbf{y}_k^o - h_k(\mathbf{x}^f(t_k)))$$

$$\mathbf{P}_k^a = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}^f(t_k)$$

$$\mathbf{K}_k = \mathbf{P}^f(t_k)\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}^f(t_k)\mathbf{H}_k^T + \mathbf{R}_k^o)^{-1}$$

$\mathbf{R}^o$ – covariance matrix from observation

$\mathbf{H}_k = J[h(\mathbf{x}, t_k)]\big|_{x=x^f}$ - Jacobian matrix of h_k



Approach, Phase I

Lagrangian Data Assimilation

Ensemble Kalman Filter (EnKF)

Approximate pdf by an ensemble of particles:

$$\{\mathbf{x}_i^f\}_{i=1}^N$$

Forecast: Evolve particles forward using ODE

Analysis: Perform Kalman-like analysis

Still assumes Gaussianity (in analysis)
NOT valid for nonlinear systems

Approach, Phase I

Lagrangian Data Assimilation

Particle Filter

Approximate pdf by an ensemble of *weighted* particles:

$$\{\mathbf{x}_i^f(t)\}_{i=1}^N \quad \{w_{i,k}\}_{i=1}^N$$

Forecast: Evolve particles forward using SDE

Analysis: Perform full Bayesian update at analysis

$$w_{i,k} = C w_{i,k-1} p(\mathbf{y}_k^o | \mathbf{x}_i^f(t_k)))$$

C - normalization constant

Validation and Testing, Phase I

Lagrangian Data Assimilation

Run filters with all states known up to observational noise.

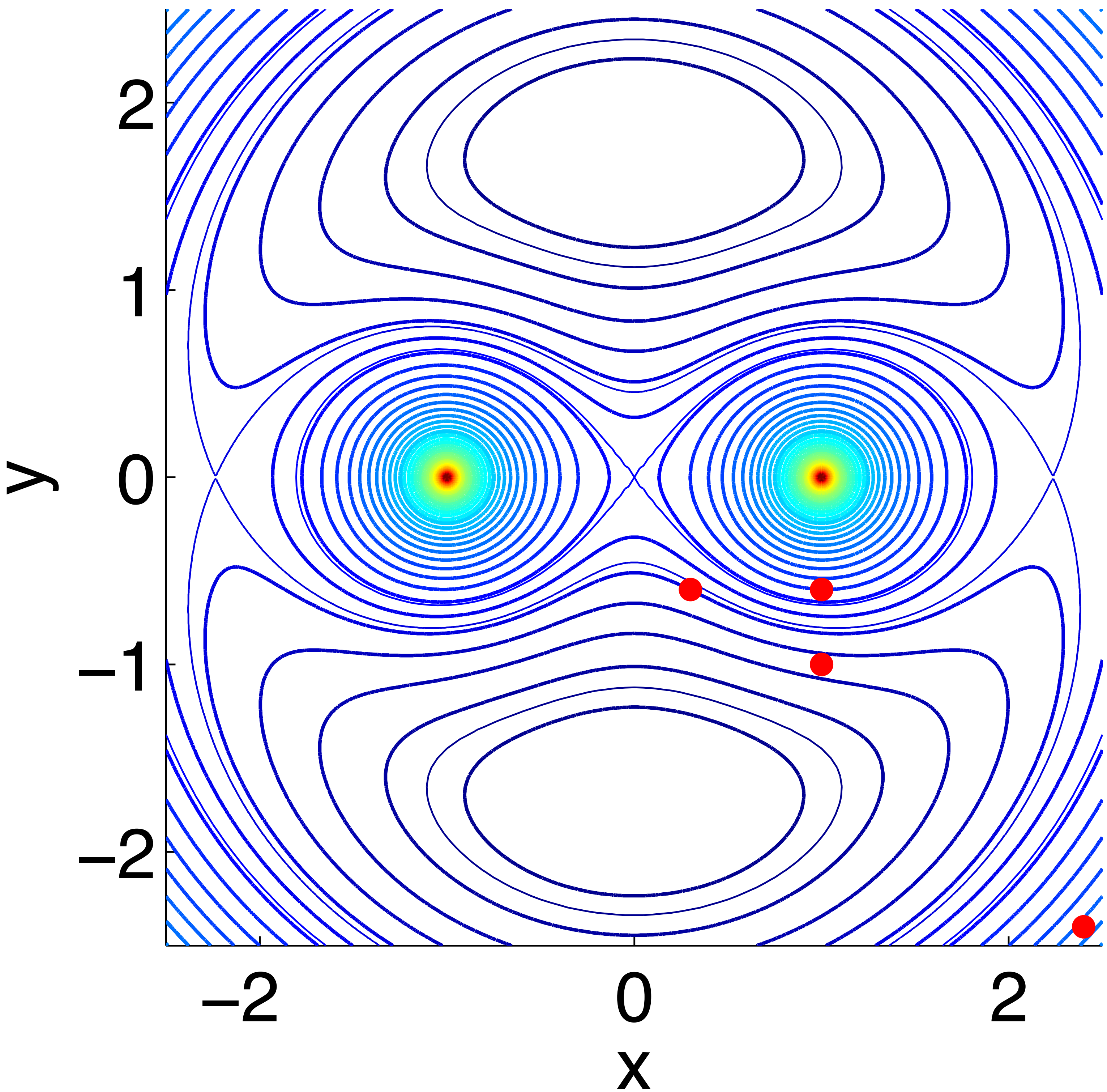
Run filters with only drifter position known. Compute failure statistics.

$$\delta^{a,f}(t) = ||\mathbf{x}^t(t) - \mathbf{x}^{a,f}(t)||_2$$

$$\hat{f}_{\delta_{div},n}(t) =$$

fraction of times $\delta(t) > \delta_{div}$ at time t in n trials

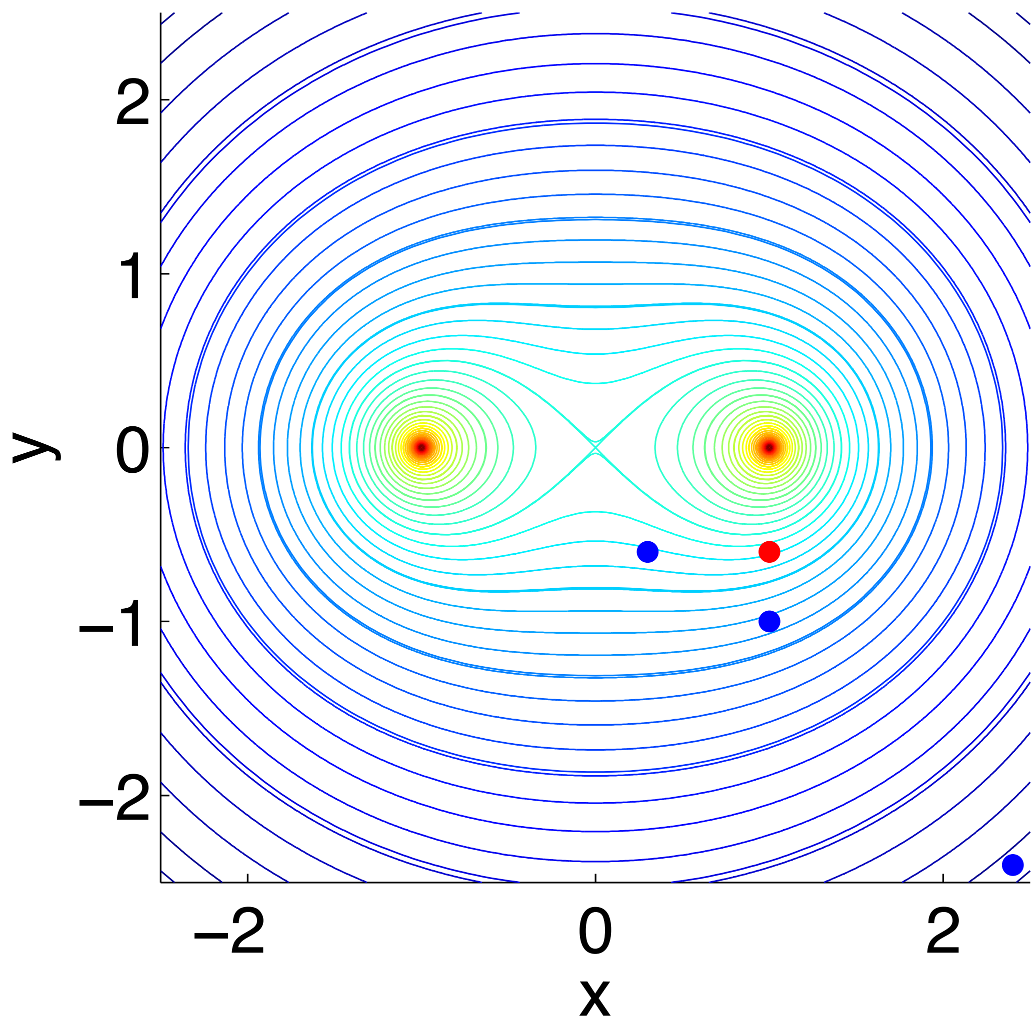
Phase II



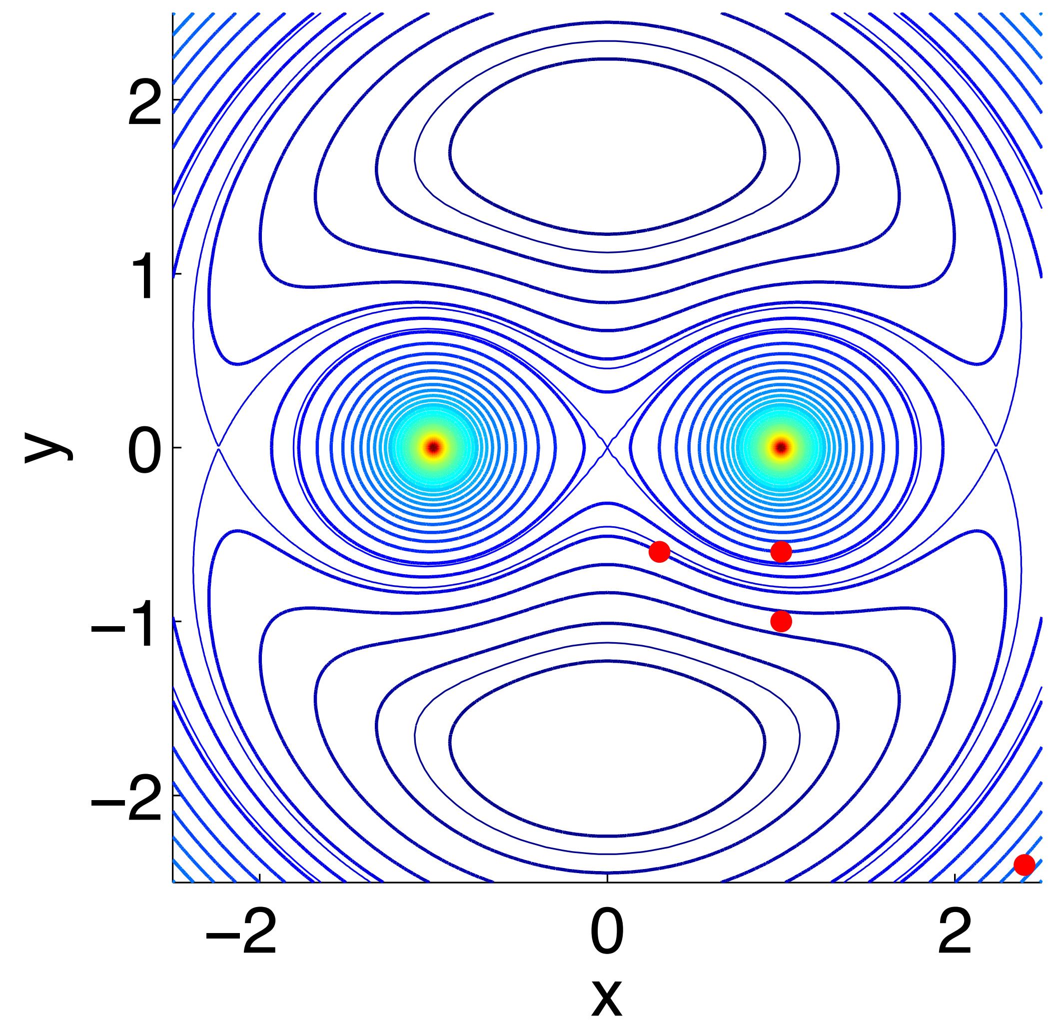
Approach, Phase II

Observing system design based on Lagrangian Coherent Structures (LCS)

streamfunction in fixed frame
[what we can compute, but not useful]



streamfunction in corotating frame
[what we want, but hidden]

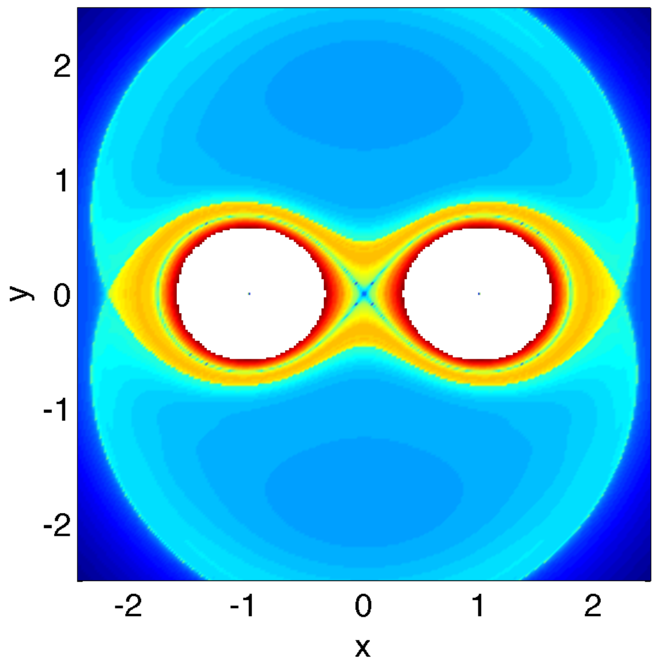


Approach, Phase II

Manifold Detection for Observing System Design

Mendoza and Mancho's Lagrangian Descriptor *M*

$$M(\mathbf{x}_D^t, t^*, \tau) = \int_{t^*-\tau}^{t^*+\tau} \left(\sum_{i=1}^{n} \left(\frac{dx_D^i(t)}{dt} \right)^2 \right)^{1/2} dt$$



Approach, Phase II

Manifold Detection for Observing System Design

Lyapunov Exponents

$\mathbf{x}_1$ - Reference Trajectory

$\mathbf{x}_2$ - Perturbed Trajectory

$\mathbf{y} = \mathbf{x}_2 - \mathbf{x}_1$ - Tangent Vector

Approach, Phase II

Manifold Detection for Observing System Design

Lyapunov Exponents

$\mathbf{x}_1$ - Reference Trajectory

$\mathbf{x}_2$ - Perturbed Trajectory

$\mathbf{y} = \mathbf{x}_2 - \mathbf{x}_1$ - Tangent Vector

$$\lambda = \lim_{t \to \infty} \frac{1}{t} \log \frac{||\mathbf{y}(t)||}{||\mathbf{y}(0)||}$$

Approach, Phase II

Manifold Detection for Observing System Design

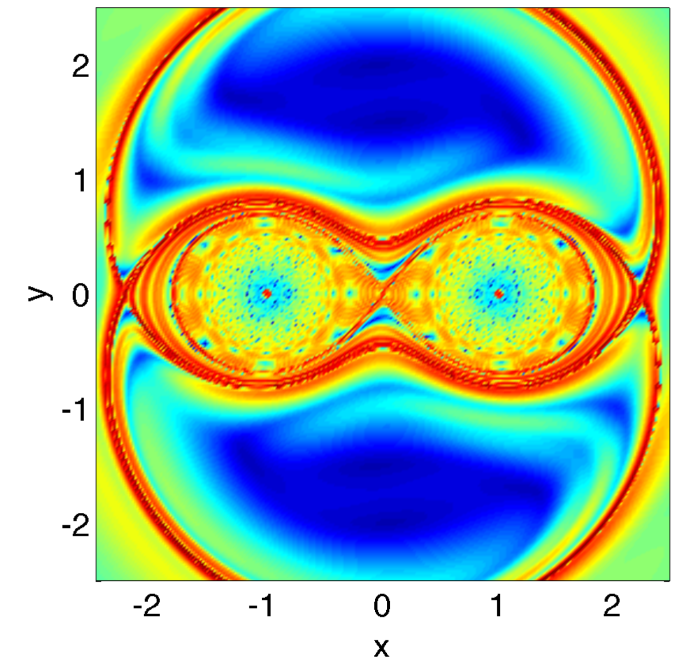
Finite-Time Lyapunov Exponents

$\mathbf{x}_1$ - Reference Trajectory

$\mathbf{x}_2$ - Perturbed Trajectory

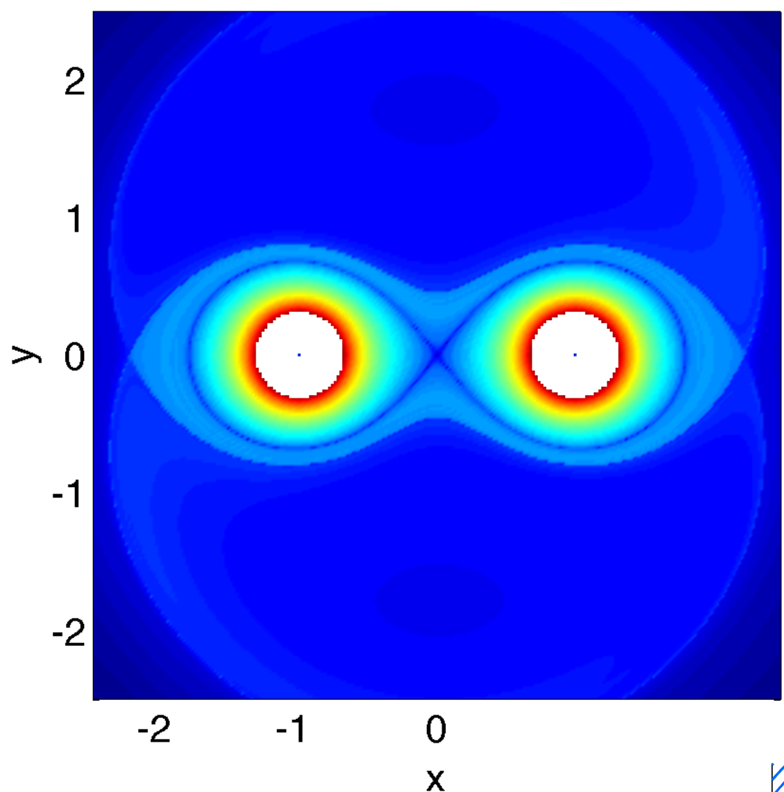
$\mathbf{y} = \mathbf{x}_2 - \mathbf{x}_1$ - Tangent Vector

$$\lambda(T) = \frac{1}{T} \log \frac{||\mathbf{y}(T)||}{||\mathbf{y}(0)||}$$

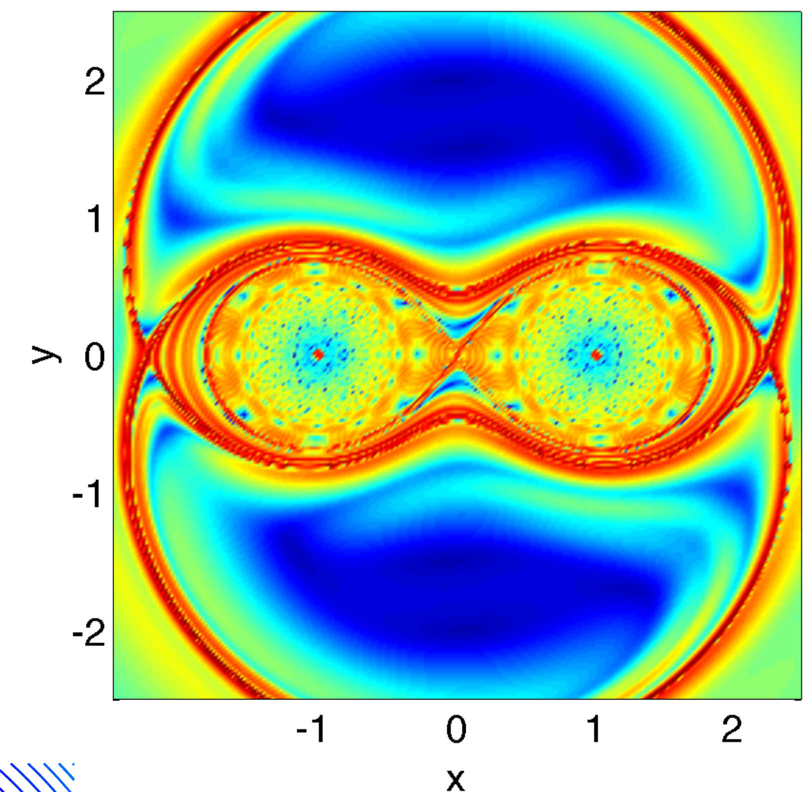


Validation, Phase II

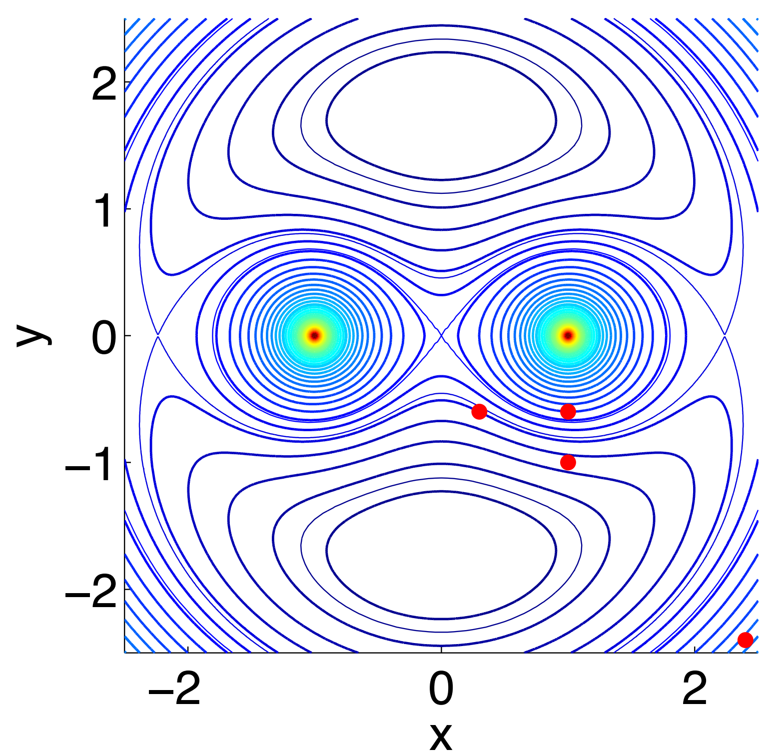
Manifold Detection for Observing System Design



M



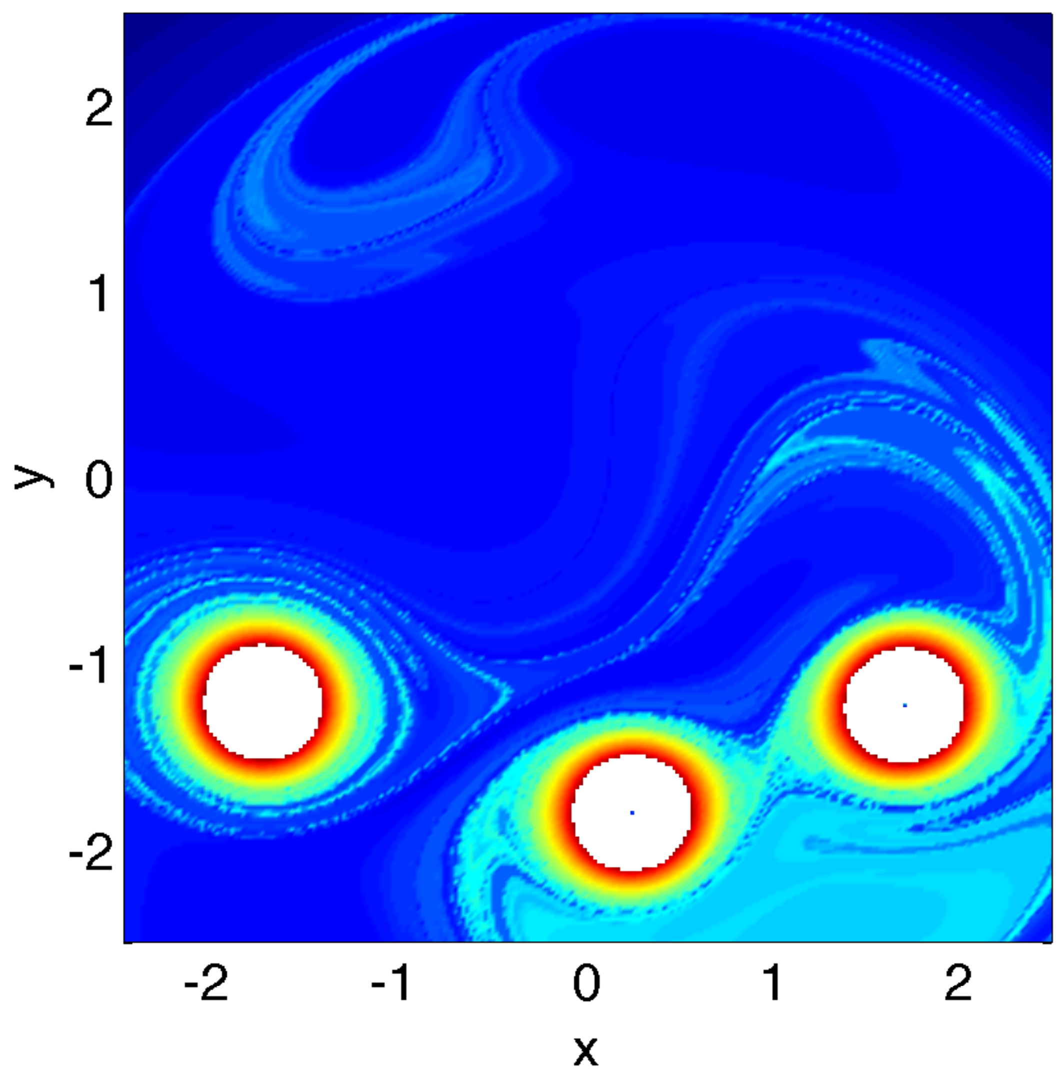
Finite-Time Lyapunov Exponents



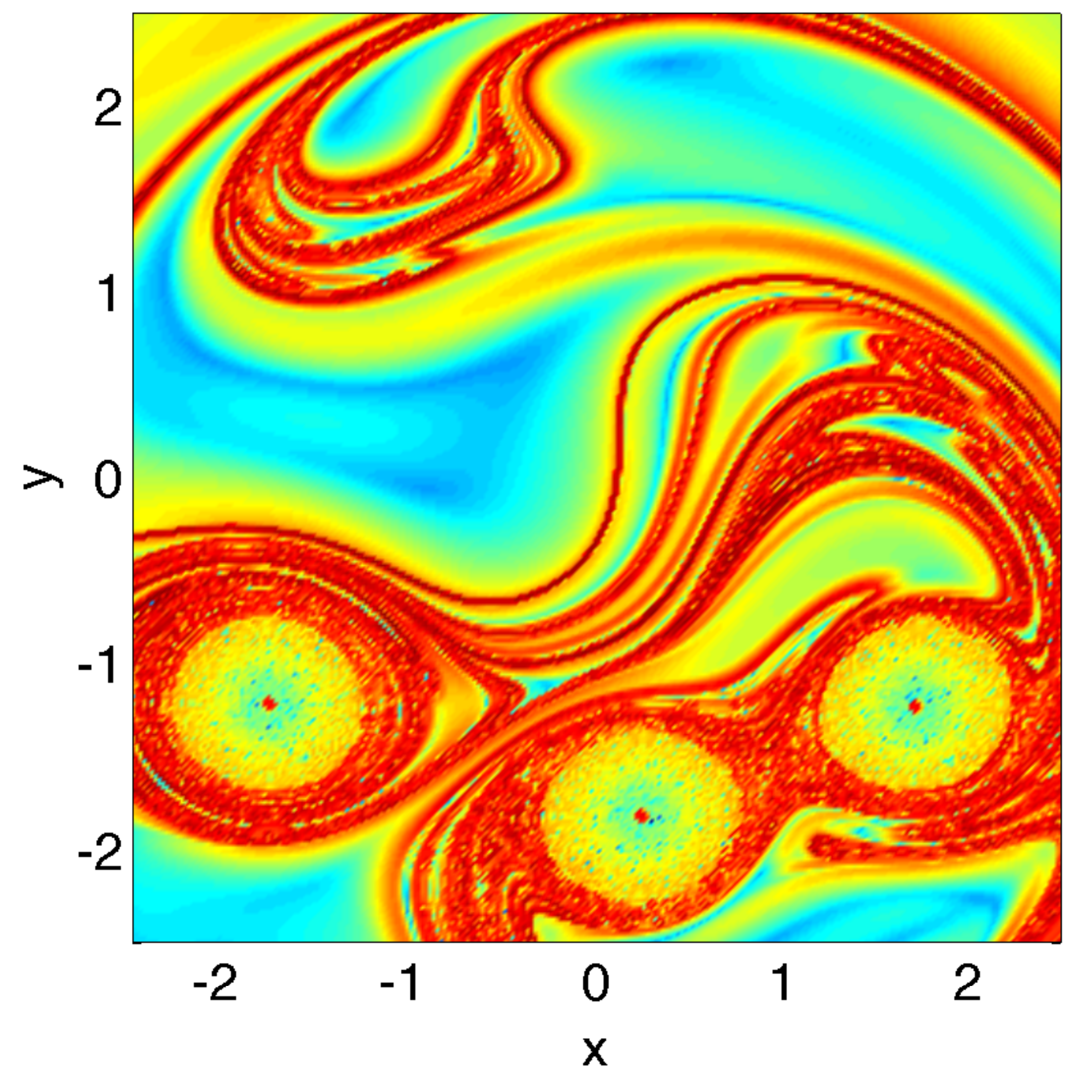
Analytical Streamfunction

Results, Phase II

Manifold Detection for Observing System Design



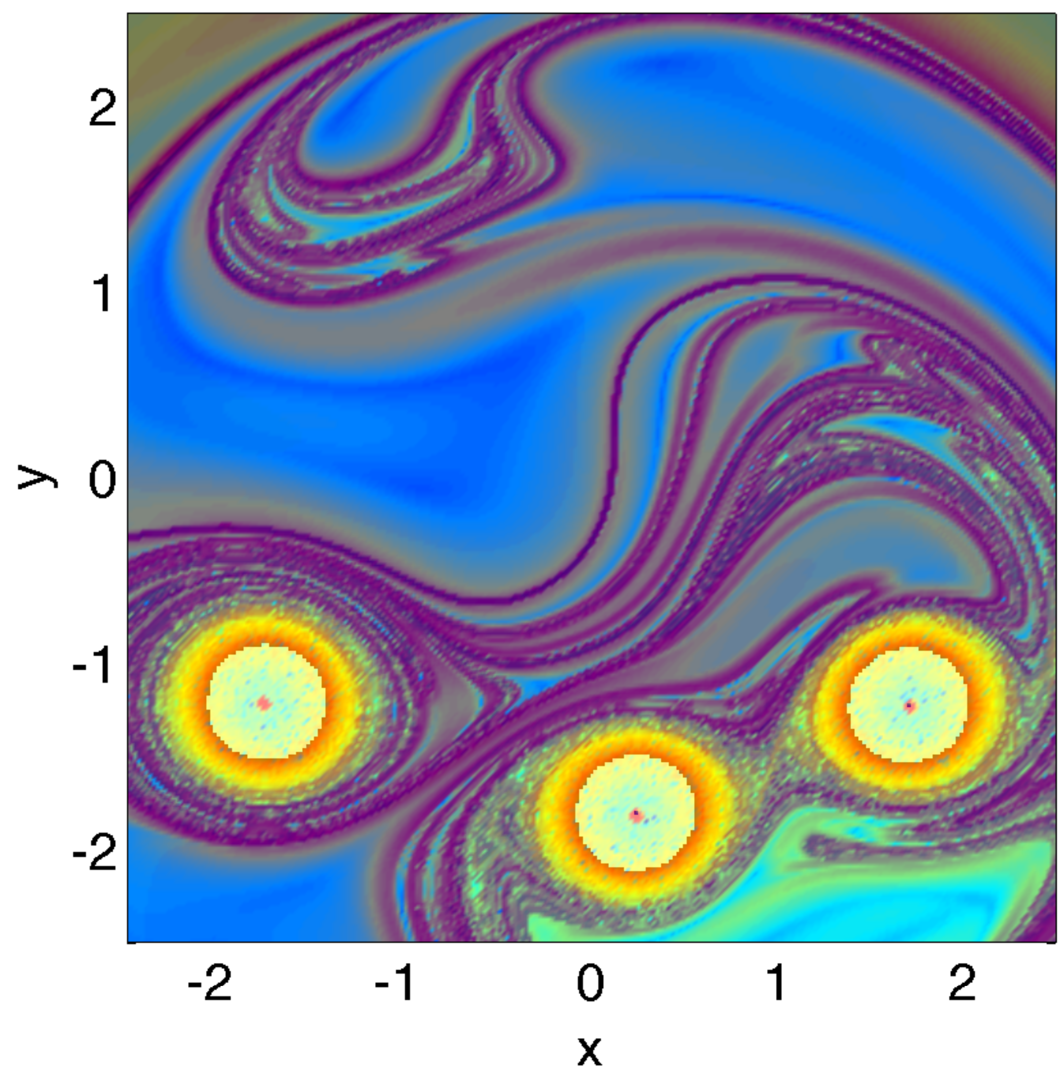
M



Finite-Time Lyapunov Exponents

Results, Phase II

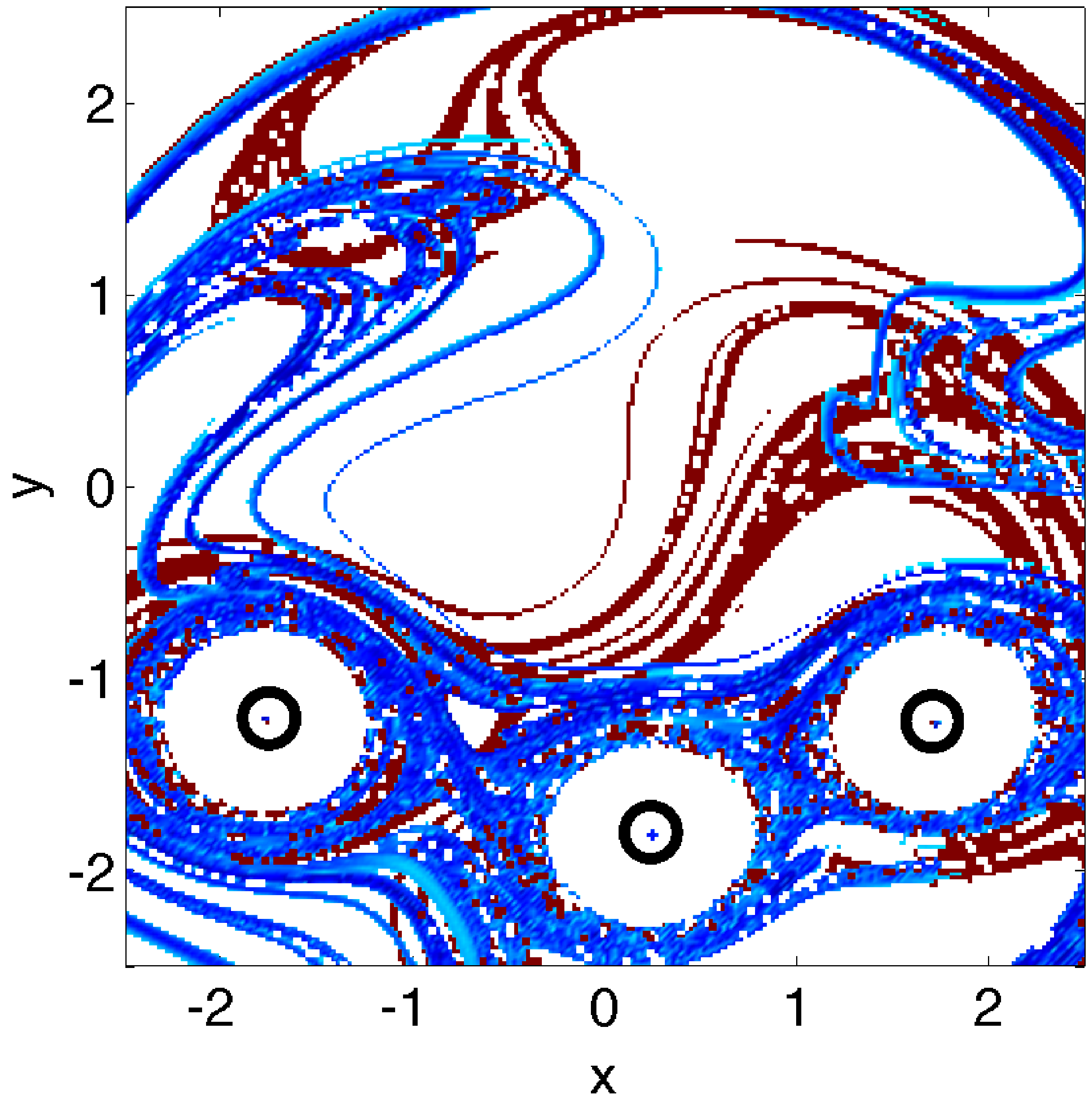
Manifold Detection for Observing System Design



Results, Phase II

Manifold Detection for Observing System Design

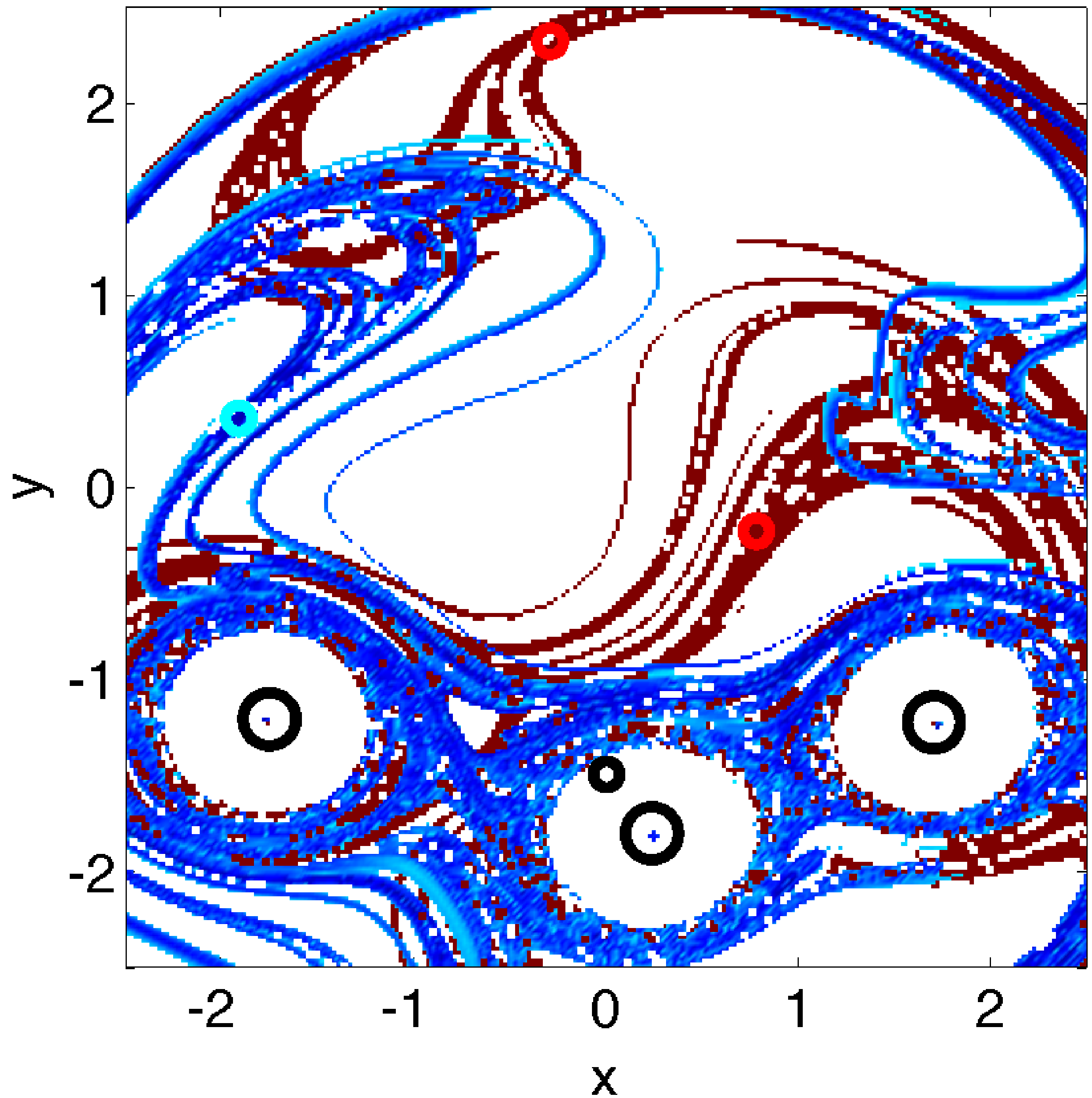
Stable Manifolds
Unstable Manifolds



Results, Phase II

Manifold Detection for Observing System Design

Stable Manifolds
Unstable Manifolds



Results, Phase II

Manifold Detection for Observing System Design

Compare three filters (EKF, LETKF, particle)

True system dynamics **deterministic**

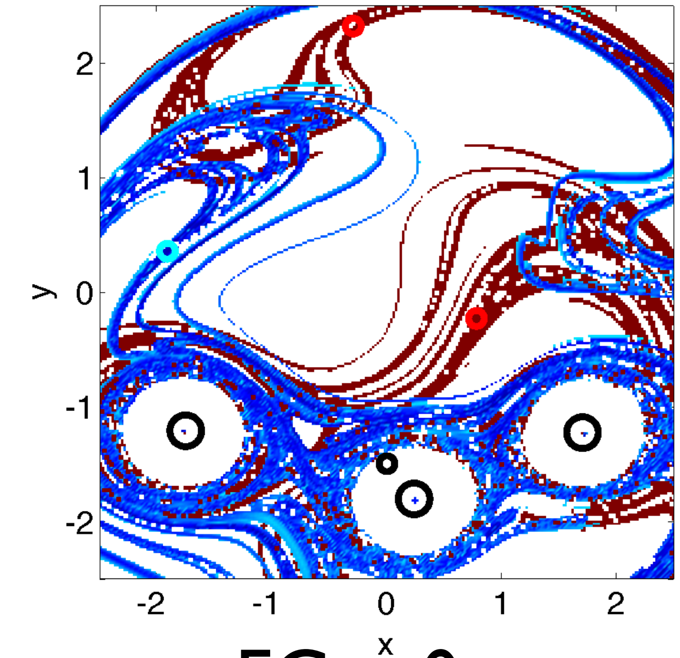
Guarantees presence of observed manifolds

Observational noise present

Play Movie

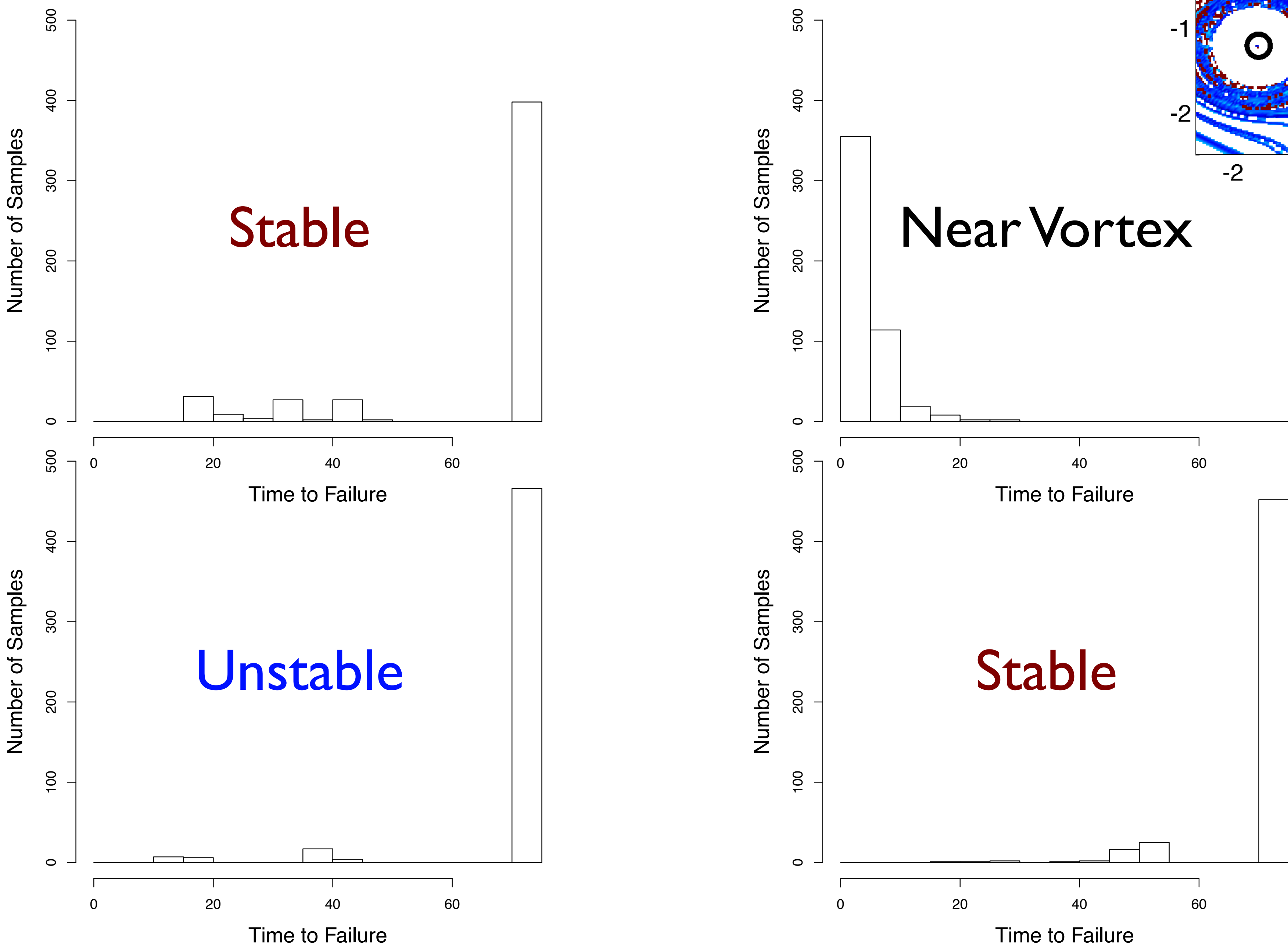
Results, Phase II

EKF



FC = 0.8

FC = 0



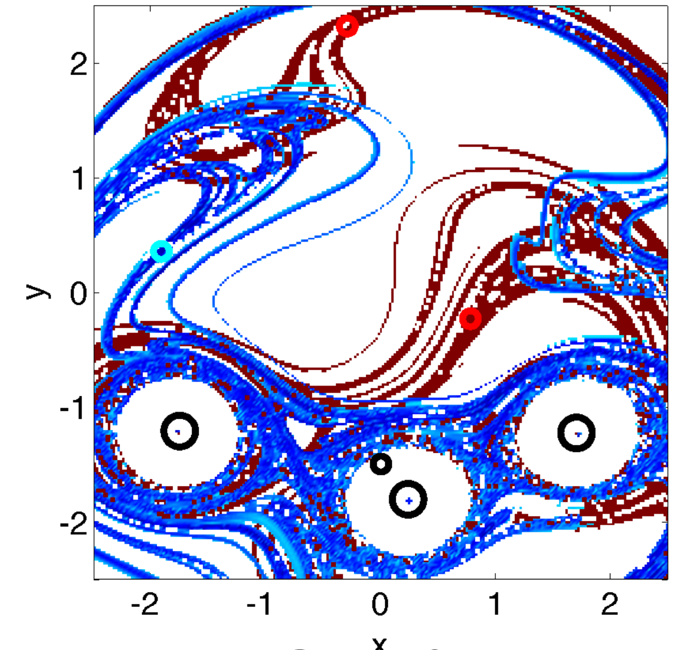
FC = 0.9

FC = 0.9

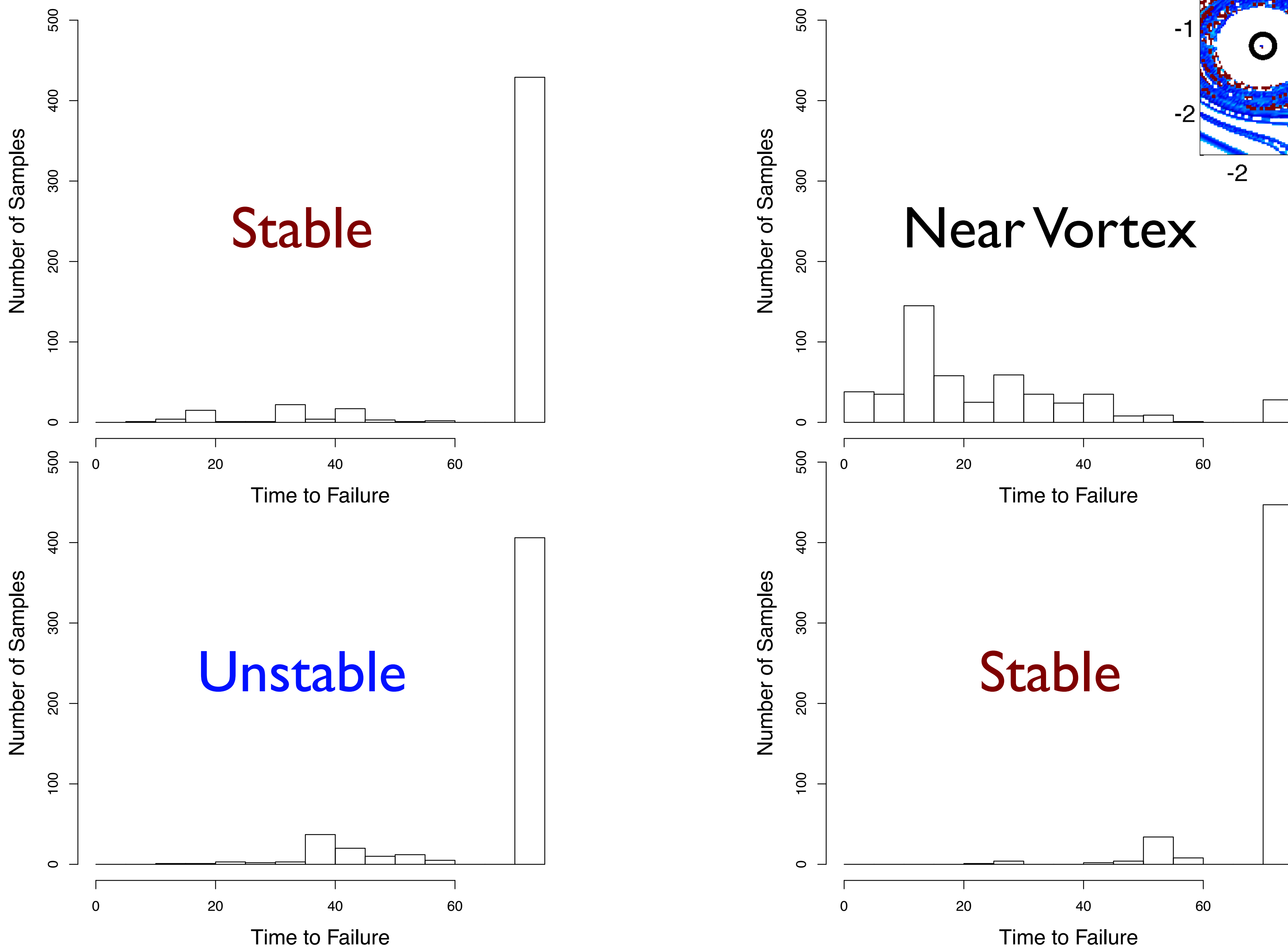
FC – fraction completed

Results, Phase II

LETKF



FC = 0.9



FC = 0.1

FC = 0.8

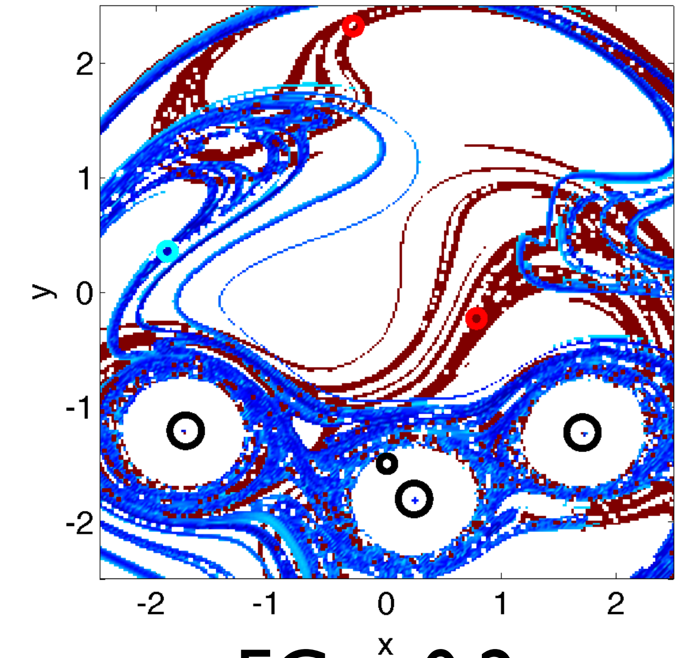
FC = 0.9

FC – fraction completed

N = 8 members

Results, Phase II

Particle

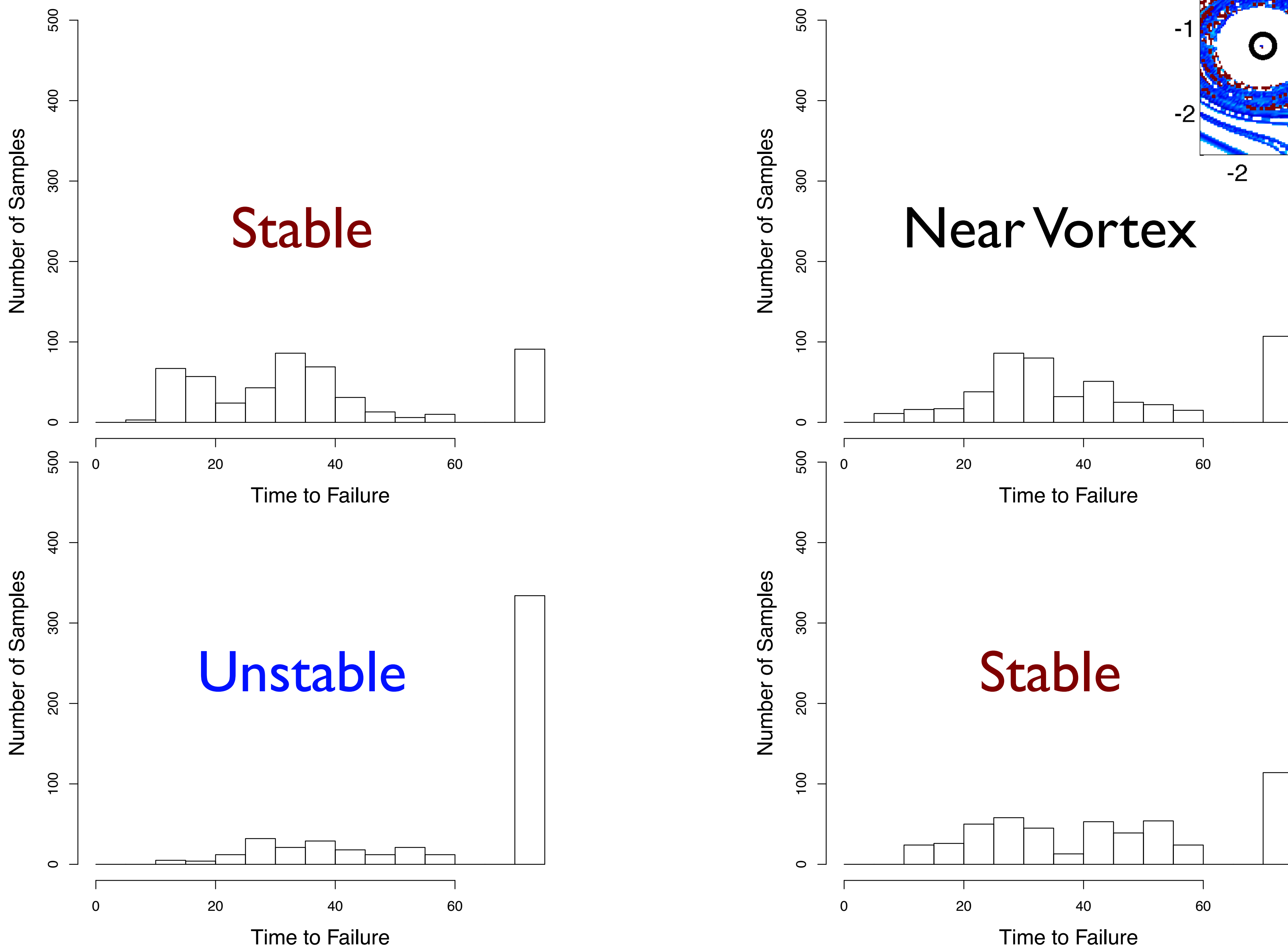


FC = 0.2

FC = 0.2

FC = 0.7

FC = 0.2

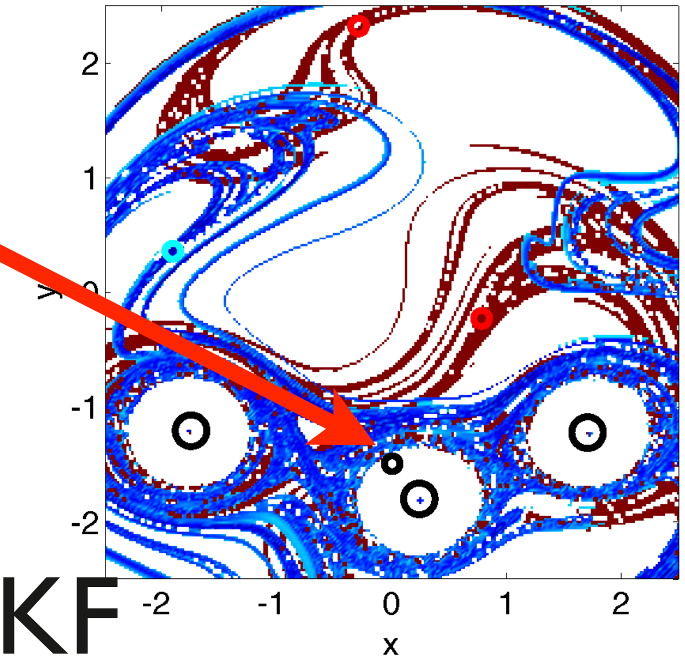


FC – fraction completed

N = 250 particles

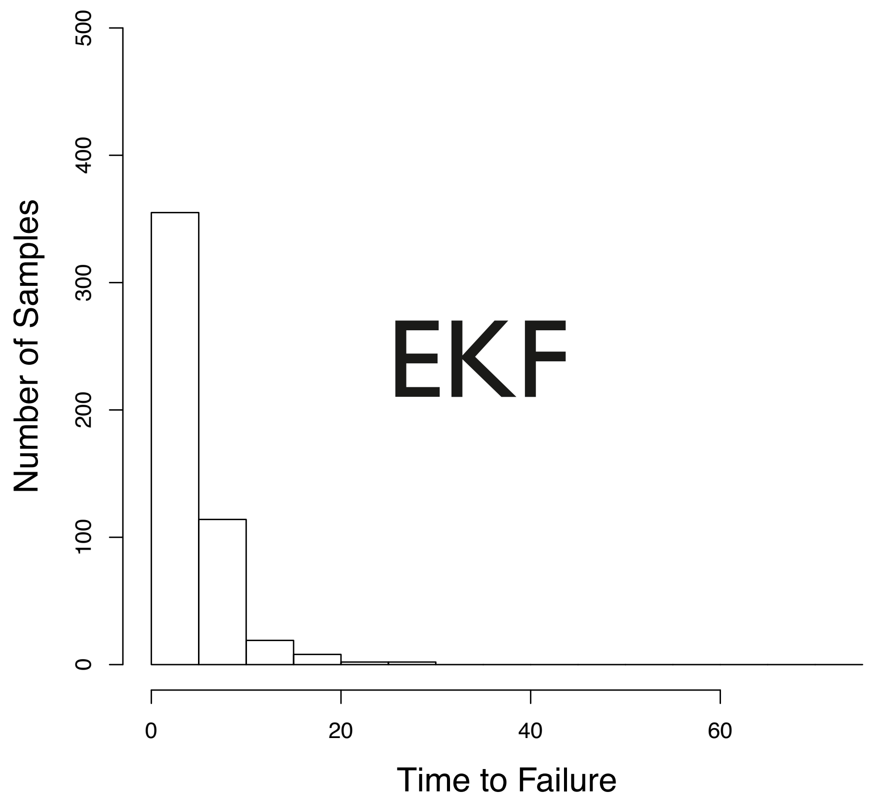
Results, Phase II

Near Vortex



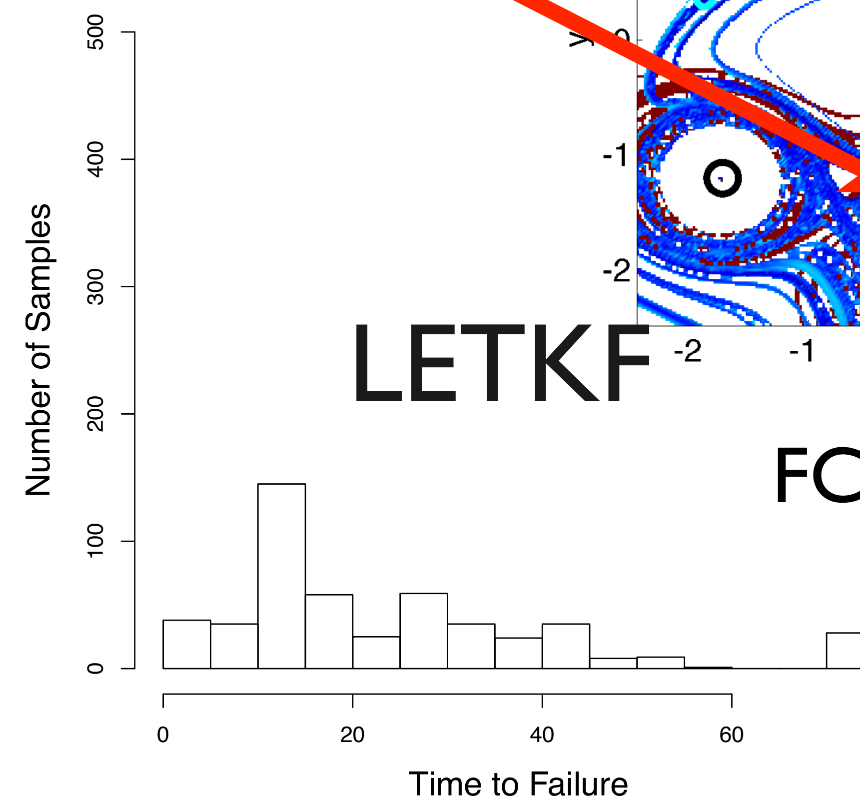
FC = 0

EKF



LETKF

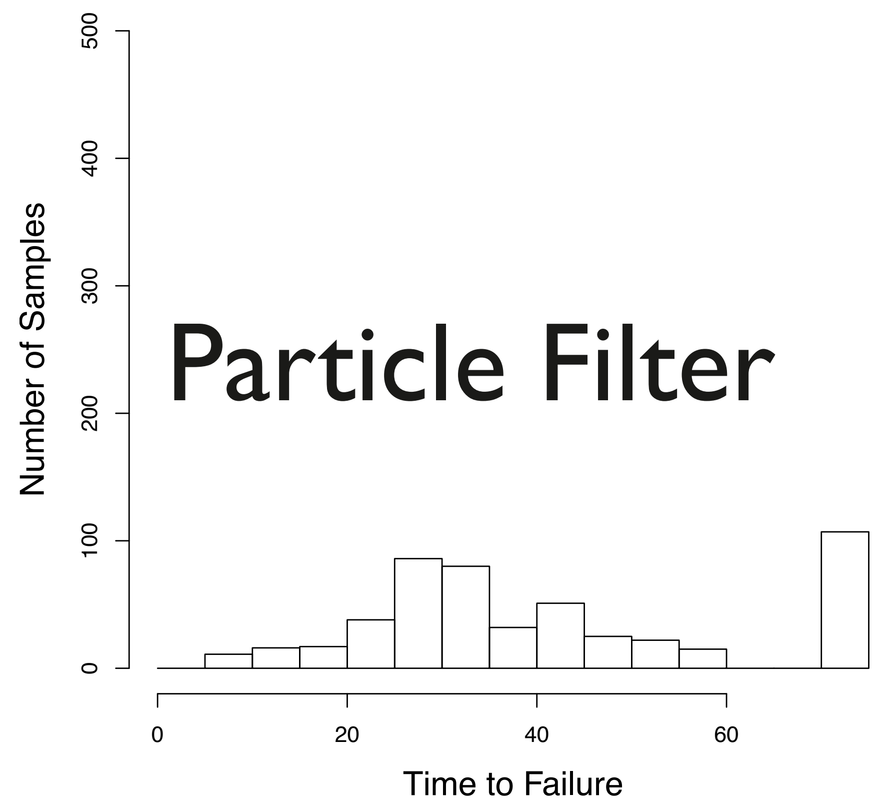
FC = 0.1



N = 8 members

FC = 0.2

Particle Filter



N = 250 particles

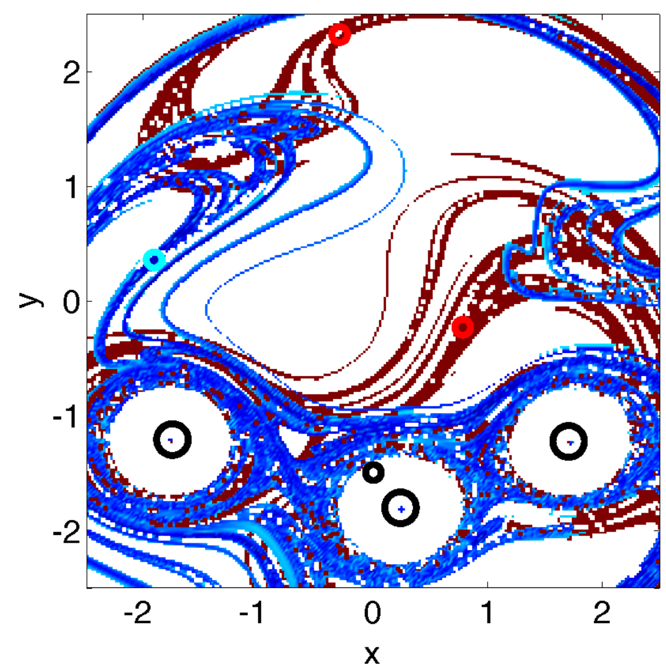
Results, Phase II

Conclusions

As before, EKF fails in the case of large non-linearities.

A priori placement emphasizes avoiding drifters near a vortex.

An artifact of the **point** vortex model.



Project Goals

Data Assimilation

Implement and validate the extended Kalman, local ensemble transform Kalman, and particle filters for a point-vortex model with N_v vortices and N_d drifters.

Compare the performance of the filters in the two vortex, single drifter case.

Observing System Design

Implement methods for computing Mendoza and Mancho's *M* function and Finite-Time Lyapunov Exponents for manifold detection.

Use detected manifolds to intelligently place drifters *a priori* in a chaotic, 3 vortex, single drifter case.

Timeline

Phase I

- Produce database: now through mid-October
- Develop extended Kalman Filter: now through mid-October
- Develop ensemble Kalman Filter: mid-October through mid-November
- Develop particle filter: mid-November through end of January
- Validation and testing of three filters (serial): Beginning in mid-October, complete by February

Timeline

Phase II

- Complete validation and testing of (serial) manifold detection: mid-March
- Parallelize ensemble methods: beginning of April
- Parallelize manifold detection algorithm: end of April

Deliverables

Database of sample trajectories used for validation and testing

Suite of parallelized software for performing EKF, **LETKF**, and particle filtering for point-vortex model

Software to compute *M* **and FTLEs** for point-vortex model

References (See Final Report for More)

A.H. Jazwinski. *Stochastic processes and filtering theory*. Dover Publications, 2007.

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

A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice.* Springer Verlag, 2001.

C. Mendoza and A.M. Mancho. Hidden geometry of ocean flows. *Physical review letters*, 105(3):38501, 2010.

Questions???