

HOW TO DISPATCH OBSERVERS TO TRACK AN EVOLVING BOUNDARY

Tingting Jiang and Carlo Tomasi

Duke University
Department of Computer Science
Durham, NC 27708, USA

Scott C. Schmidler

Duke University
Institute of Statistics and Decision Sciences
Durham, NC 27708, USA

ABSTRACT

Some distributed-sensing applications make it necessary to dispatch a limited number of observers (ships, vehicles, or airplanes with cameras; field workers with chemical kits; high-flying balloons with atmospheric sensors) to track the evolving boundary of a large phenomenon such as an oil spill, a fire, a hurricane, air or water pollution, or *EL Niño*. This paper develops a new framework for controlling the movements of the observers to maximize the information gained about the boundary's shape and position. To this end, we represent boundary uncertainty by a particle filter where each particle is a binary indicator function. This makes our dispatch algorithms applicable to arbitrary boundary representations from which indicator functions can be computed, including level sets and polygonal approximations. We demonstrate the benefits of optimal dispatch on both synthetic and real data. These benefits are most apparent when the observers are sparse relative to the boundary size.

1. INTRODUCTION

The increasing availability of new sensor types and of sensor networks yields new opportunities and challenges in distributed sensing. Often sensors are few and expensive, and need to be carried to appropriate, usually changing locations. For instance, air temperature, atmospheric pressure, cloud patterns, or wind speeds can be tracked by sensors on high-flying balloons. Water currents, temperatures, or pollution levels are recorded over vast areas by vessels or movable buoys. Oil spills or fires are often monitored by airplanes with cameras.

Applications like these can be abstracted into a complex, plane boundary, whose shape and topology (the boundary's *state* for short) changes constantly, tracked by a small number of moving observers. The complexity of the set of possible boundaries suggests using particle filters [1, 2] to track and refine the state of evolving objects from sparse measurements. These filters maintain a random sample (a set of particles) drawn from the probability distribution that represents the current knowledge about the boundary's state. Particles

are propagated as the state changes, and are updated when new measurements become available.

In the standard particle filter theory [3], the measurement locations over time are assumed to be given as inputs. However, the small number of observers and the cost of moving them in applications like the ones discussed above implies that observers must be controlled with care, and in a way that depends on the evolving phenomenon being monitored. We call this the observer *dispatch* problem.

Assume that each observer is mobile, controllable and able to observe local information around some region of the evolving boundary. An optimal dispatch algorithm must consider the change of the boundary, the time and energy cost of reallocations, and the distribution of the observers to maximize the benefit of their measurements. Such a problem is naturally addressed in a decision-theoretic framework.

To this end, we formulate dispatch as a utility optimization problem for which we propose a real-time algorithm. Our method can work in concert with any boundary tracking algorithm based on particle filters, regardless of the curve representation used by the latter. For instance, boundaries can be represented as splines or level sets of functions [4, 2]. We obtain this generality by introducing a new, probabilistic boundary indicator function that captures the spatial aspects of boundary uncertainty that are relevant to the dispatch problem. Our experiments on synthetic and real data sets show that our dispatch method yields a more balanced observer placement and lower overall boundary uncertainty in comparison with a random dispatch strategy, even if the latter is guided by the current best estimate of the boundary.

In the next section, we briefly review previous work. Section 3 introduces the boundary representation. Section 4 introduces the dispatch algorithm and its analysis. Section 5 presents experimental results, and Section 6 concludes with a summary and plans for future work.

2. PREVIOUS WORK

Particle filters [3, 5] have been used in the context of boundary tracking mainly in the area of active contours [6, 7]. These approaches capture the uncertain position of a boundary at

This research was partially funded under NSF grant IIS-0534897.

time t by a probability distribution represented by a random sample of boundaries (*particles*). Each boundary in the sample is represented explicitly, *e.g.* with splines [4] and *propagated* forwards in time through an assumed, uncertain motion model. Measurements *update* the particles by weighing each of them by its posterior probability given the measurements. *Resampling* then draws new particles from the posterior to be ready for a new step of propagation. This cycle is analogous to the estimation loop of a Kalman filter [8], but maintains a sample-based representation of the distribution rather than a Gaussian one. Recently, the approach of “level-set curve particles” was proposed [2] where each boundary particle is represented by a level set function, instead of a spline. This approach is based on the notion of “adding” two boundary curves by adding the level set functions that represent them. This insight leads to a superposition effect that lets one track exponential number of boundary particles at linear cost in the framework of a particle filter and makes a combination of level sets and particle filters computationally affordable.

The problem of *sensor placement* has been studied recently by the community of robotics and machine learning [9, 10, 11]. Guestrin et al. [9] proposed a mutual information criterion to place sensors based on a Gaussian process model (GP). Instead of maximizing the joint entropy of the Gaussian unknown variables inside “sensing area”, they maximize the mutual information between the unknown variables in the sensing area and those in the rest of the space. The key assumptions are that the prior (before-observation) and posterior (post-observation) distributions of the unknown variables both have the form of a Gaussian process which has a fixed kernel function and further implicitly that the observations are “perfect” (noise-free). Schwager et al. [10] proposed a control strategy for mobile robots to optimize the measurement of sensory information. The target function is based on an unreliability function of observer measurements and a scalar sensory function that indicates the relative importance of different areas in the region. They use the centroidal Voronoi Diagrams [12] to distribute the sensors simultaneously.

However, the existing particle filter techniques for tracking dynamic boundaries do not address the issue of optimal observer placement. Conversely, current sensor placement work does not accommodate moving boundaries. Our proposed dispatch algorithm is based on known techniques in location optimization, modified to apply to a different target function, and to include a specific account of the geometry of the boundary being tracked without any distribution assumptions. This requires devising a general boundary representation that simultaneously captures the aspects of boundary uncertainty that are relevant to tracking and can interface with whatever boundary encoding is used in the tracker of choice. This new representation is described next.

3. INDICATOR FUNCTIONS FOR BOUNDARIES

Given a boundary \mathcal{B} on the plane, define an indicator function $b(\mathbf{x}) : \mathbb{R}^2 \rightarrow \{0, 1\}$ as :

$$b(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \text{ inside the boundary,} \\ 0 & \text{otherwise.} \end{cases}$$

When \mathcal{B} is uncertain, let b be a random indicator function whose realizations could be any boundary indicator on the plane, and let $p(b)$ be a probability density function over the space of all possible indicator functions. The mean of b is denoted as

$$\bar{b}(\mathbf{x}) = \int b(\mathbf{x})p(b) db \approx \sum_{i=1}^P b_i(\mathbf{x})w_i. \quad (1)$$

The p.d.f. $p(b)$ in (1) is to be understood in a very abstract sense, as b is a point in the space of all boundary indicator functions. Concretely, $p(b)$ can be represented approximately by a random sample of weighted particles $\{b_i, w_i\}_{i=1}^P$ of some parametric form. Each particle b_i is an indicator function for a sample boundary \mathcal{B}_i and w_i is the weight of this particle with the normalization constraint that $\sum_{i=1}^P w_i = 1$. Then the mean of b can be approximated by the sample mean. Given any point \mathbf{x} on the plane, the variance of the indicator random variable $b(\mathbf{x})$ is denoted as the following function:

$$v(\mathbf{x}) = \int [b(\mathbf{x}) - \bar{b}(\mathbf{x})]^2 p(b) db \approx \sum_{i=1}^P [b_i(\mathbf{x}) - \bar{b}(\mathbf{x})]^2 w_i. \quad (2)$$

When the boundary is evolving and deforming over time, the random indicator function for the boundary $\mathcal{B}^{(t)}$ at time t is denoted by $b^{(t)}$. A particle set $\{b_i^{(t)}, w_i^{(t)}\}_{i=1}^P$ is then used to represent the probability distribution of the boundary $\mathcal{B}^{(t)}$. For each time step t , define the variance function $v^{(t)}(\mathbf{x})$ of the indicator random variable $b^{(t)}(\mathbf{x})$ as in (2).

The above discussion is not rigorous in that the boundary space is infinite-dimensional and we have not specified an underlying measure. However, we can avoid these technicalities since in practice, the region of the plane of interest can be discretized into N grid points, so that each indicator function $b_i^{(t)}(\mathbf{x})$ is transformed to a binary vector $\mathbf{b}_i^{(t)}$ of size N and the function $v^{(t)}(\mathbf{x})$ becomes a vector $\mathbf{v}^{(t)}$ of size N .

To summarize, a probability density function $p(b)$ for the random indicator function b of a dynamic boundary at time t can be approximately represented by a particle set $\chi^{(t)} = \{\mathbf{b}_i^{(t)}, w_i^{(t)}\}_{i=1}^P$ on a discretized plane. The sample mean (1) and variance (2) are a first order summary of the density.

4. DISPATCH PROBLEM

Based on the above representation of the boundary, we formulate observer dispatch as the problem of maximizing expected

utility of the observer placements. We solve this problem using centroidal Voronoi diagrams. We also consider some of the constraints that arise in real applications.

4.1. Problem Statement

As mentioned earlier, a particle filter works in three steps: propagation, update and resampling. In addition, we assume that there are M mobile and controllable observers each of which gathers some local information around its own location. To exploit the resulting freedom to choose where to take the measurements at each time step t , we insert a new step called *dispatch* between propagation and update to redistribute the mobile observers for the new measurements. Now the particle filter has four steps: propagation, dispatch, update and resampling.

4.2. Observer Model

Our observer model includes two aspects: how each observer works, and how different observers interact. The first aspect specifies the probability that an observer at \mathbf{z} can *observe* a point \mathbf{x} , that is, determine whether it is inside or outside the boundary:

$$f(\mathbf{x}, \mathbf{z}) = Pr[b(\mathbf{x}) \text{ is observed by observer at } \mathbf{z}]. \quad (3)$$

To reflect decreasing acuity with distance, the function f is assumed to decrease as the Euclidean distance $d(\mathbf{x}, \mathbf{z})$ between the observer position \mathbf{z} and the point \mathbf{x} increases. In this model, the observer returns one bit of information about point \mathbf{x} if the point is observed. In this case, the posterior variance of the indicator function value at \mathbf{x} is zero. With no observation, the posterior variance remains equal to its prior value.

Concerning observer interaction, we note that observers are precious and carefully dispatched, so they are likely to be far apart enough to eliminate overlap between their observation ranges. This implies that whether a point \mathbf{x} can be observed only depends on the observer nearest to \mathbf{x} . With M observer locations $Z = \{\mathbf{z}_1, \dots, \mathbf{z}_M\}$, let

$$\mathbf{z}(\mathbf{x}) = \arg \min_{\mathbf{z}_m} d(\mathbf{x}, \mathbf{z}_m) \quad (4)$$

be the location of the observer nearest to \mathbf{x} . Then equation (3) can be rewritten as follows in the multi-observer case:

$$Pr[b(\mathbf{x}) \text{ is observed} | Z] = f(\mathbf{x}, \mathbf{z}(\mathbf{x})). \quad (5)$$

4.3. Optimization Function

To formalize the quality of a given choice of observer dispatch, we define a utility function for the placement of observers. First, define an L_2 metric to measure the distance between any two boundaries as

$$D(b_i, b_j) = \int_{\mathbf{x}} [b_i(\mathbf{x}) - b_j(\mathbf{x})]^2 d\mathbf{x}. \quad (6)$$

Given a probability distribution $p(b)$ and an estimate of the boundary \hat{b} , the expected loss function is:

$$L(p(b), \hat{b}) = \int D(b, \hat{b}) p(b) db.$$

The optimal decision-theoretic estimate of the boundary given $p(b)$ is

$$b^* = \operatorname{argmin}_{\hat{b}} L(p(b), \hat{b}). \quad (7)$$

For a given dispatch choice Z , the expected utility is defined as the reduction in expected loss:

$$U(Z) = L(p(b), b^*) - L(p(b)|_Z, b^*_{|Z}) \quad (8)$$

where $p(b)|_Z$ denotes the conditional probability distribution on the observer placement Z and $b^*_{|Z}$ means the best estimate given $p(b)|_Z$.

To maximize the utility function in (8), the best choice of the placement would be

$$Z^* = \operatorname{argmax}_Z U(Z) = \operatorname{argmin}_Z L(p(b)|_Z, b^*_{|Z}). \quad (9)$$

The second equality holds because the loss function value $L(p(b), b^*)$ before dispatch is fixed.

Computing Z^* for $\hat{b} \in \{0, 1\}^N$ for general p and f is an exponential computation. However, if we relax the range of the estimate \hat{b} to be $[0, 1]^N$ instead of $\{0, 1\}^N$, then the best estimate for $p(b)$ is easily shown to be the mean of b in Eqn. (1) and therefore the loss function becomes

$$\begin{aligned} L(p(b), \bar{b}) &= \int D(b, \bar{b}) p(b) db \\ &= \int \int [b(\mathbf{x}) - \bar{b}(\mathbf{x})]^2 p(b) d\mathbf{x} db \\ &= \int v(\mathbf{x}) d\mathbf{x}. \end{aligned}$$

Therefore, the optimization function $L(p(b)|_Z, b^*_{|Z})$ can be defined as

$$F(Z) = \int_{\mathbf{x}} v_{|Z}(\mathbf{x}) d\mathbf{x}, \quad (10)$$

where $v_{|Z}(\mathbf{x})$ is the posterior variance of the point \mathbf{x} after the observation with observer placement Z .

With the observer model in Section 4.2, the expected posterior variance of point \mathbf{x} can be approximated by

$$\begin{aligned} E(v_{|Z}(\mathbf{x})) &\approx Pr[b(\mathbf{x}) \text{ observed} | Z] \cdot 0 + \\ &\quad Pr[b(\mathbf{x}) \text{ not observed} | Z] \cdot \hat{v}(\mathbf{x}) \\ &= [1 - f(\mathbf{x}, \mathbf{z}(\mathbf{x}))] \hat{v}(\mathbf{x}) \end{aligned} \quad (11)$$

where $\hat{v}(\mathbf{x})$ is the estimated prior variance of point \mathbf{x} from propagation result. The optimal dispatch strategy is to send

the observers to locations which minimizes the total expected posterior loss, approximated as follows:

$$\begin{aligned}
F(Z) &= \int_{\mathbf{x}} E(v|Z(\mathbf{x})) \, d\mathbf{x} \\
&= \int_{\mathbf{x}} [1 - f(\mathbf{x}, \mathbf{z}(\mathbf{x}))] \hat{v}(\mathbf{x}) \, d\mathbf{x} \\
&= \sum_{m=1}^M \int_{C_m} [1 - f(\mathbf{x}, \mathbf{z}_m)] \hat{v}(\mathbf{x}) \, d\mathbf{x}. \quad (12)
\end{aligned}$$

Here, C_m denotes the cell of point \mathbf{z}_m in the Voronoi diagram of the observer locations $\mathbf{z}_{1:M}$. This is because each $f(\mathbf{x}, \mathbf{z}(\mathbf{x}))$ only depends on the observer nearest to \mathbf{x} . Finding a set of \mathbf{z}_m that minimizes the target function value F is an unconstrained optimization problem. The grid-discretized form of (12) is

$$F(Z) = \sum_{m=1}^M \sum_{\mathbf{x} \in C_m} [1 - f(\mathbf{x}, \mathbf{z}_m)] \hat{v}(\mathbf{x}). \quad (13)$$

4.4. Dispatch Algorithm

When the observer model f is a function of the Euclidean distance between \mathbf{x} and $\mathbf{z}(\mathbf{x})$, it is geometrically appealing to cast the optimization problem defined in Section 4.3 into the form of a centroidal Voronoi diagram computation [12]. To this end, we choose $f(\mathbf{x}, \mathbf{z}(\mathbf{x})) = \phi(\|\mathbf{x} - \mathbf{z}(\mathbf{x})\|^2)$ and assume that $\phi(d)$ is first-order differentiable with (non-positive) derivative $\phi'(d)$. Let $\mathbf{z}_m = \{z_m^1, z_m^2\}$ and $\mathbf{x} = \{x_1, x_2\}$. The partial derivative of F with respect to z_m^j for $j = 1, 2$ is

$$\frac{\partial F}{\partial z_m^j} = - \sum_{\mathbf{x} \in C_m} 2\phi'(\|\mathbf{x} - \mathbf{z}(\mathbf{x})\|^2) \hat{v}(\mathbf{x}) (z_m^j - x_j). \quad (14)$$

Define a nonnegative function $\rho(\mathbf{x}, \mathbf{z}(\mathbf{x}))$ such that

$$\rho(\mathbf{x}, \mathbf{z}(\mathbf{x})) = -\phi'(\|\mathbf{x} - \mathbf{z}(\mathbf{x})\|^2) \hat{v}(\mathbf{x}). \quad (15)$$

At a minimum of F , all terms of the form (14) are equal to zero. This implies that the optimal position \mathbf{z}_m of each observer satisfies

$$\mathbf{z}_m = \frac{\sum_{\mathbf{x} \in C_m} \rho(\mathbf{x}, \mathbf{z}_m) \mathbf{x}}{\sum_{\mathbf{x} \in C_m} \rho(\mathbf{x}, \mathbf{z}_m)} \quad (16)$$

which is the centroid of the cell of \mathbf{z}_m in the Voronoi diagram with respect to the density function $\rho(\mathbf{x}, \mathbf{z}_m)$.

From this analysis, we see that the dispatch problem can be solved by finding the centroids of a Voronoi diagram of the observers, in which the ‘‘mass distribution’’ ρ depends on both the observation model ϕ and the estimated prior variance $\hat{v}(\mathbf{x})$. Based on Lloyd’s method [13], which iterates between constructing Voronoi diagrams and finding centroids, we obtain the intuitively simple dispatch Algorithm 1 in the table. The initial observer positions are randomly generated. The

Algorithm 1 : Dispatch algorithm

INPUT: Observer model ϕ and estimated prior variance $\hat{v}(\mathbf{x})$

OUTPUT: Observer locations $\mathbf{z}_{1:M}$.

0. Let $\mathbf{z}_{1:M}$ be the current observer positions;

1. Construct the Voronoi diagram of points $\mathbf{z}_{1:M}$;

2. Set new values of $\mathbf{z}_{1:M}$ as the centroids of the Voronoi cells C_m to the right-hand sides of equation (16);

3. If the new set of points meet a convergence criterion, terminate; otherwise, go to step 1.

convergence criterion depends on the specific application. It is easy to show that the function value of F decreases during each iteration, but there is no guarantee that it will converge to the global minimum. The running time of each iteration is $O(M \log M + N)$.

4.5. Constraints

In real applications, the observers are typically subject to constraints like the following: (a) A speed limit for each observer, which limits the range of variation of each \mathbf{z}_m in a fixed time interval; (b) A need for a bounded travel cost (e.g., energy consumption), which is an increasing function of the travel distance; (c) Mobility constraints, e.g. each observer might only be allowed to move in a local area. Speed limits and mobility constraints can be incorporated by turning the optimization problem into a constrained one with the same target function F . Travel cost can be included as an additive item of the target function F , and its importance can be regulated by a weight λ :

$$F' = F + \lambda F_c \quad (17)$$

where F_c is the travel cost. All these constraints reduce the feasible regions of \mathbf{z}_m ’s.

5. EXPERIMENTS

We test the dispatch algorithm on two kinds of data sets, synthetic and real. Real data refer to the problem of tracking sea surface temperature. Before showing experimental results, we illustrate how the form of the estimated prior variance function $\hat{v}(\mathbf{x})$ affects the dispatch results with the same observer model. In all the experiments, we choose the observer model

$$\phi(d) = \exp(-d^2/R^2) \quad (18)$$

where the parameter R models the (soft) observation range of each observer.

5.1. The Influence of Variance Functions

With the same initialization of observer positions \mathbf{z}_m and observation model ϕ , different prior variance estimate functions

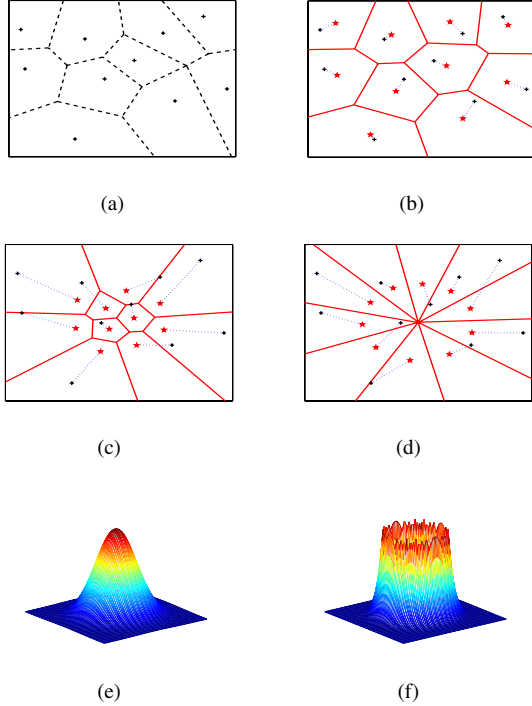


Fig. 1. (a) The initial positions of observers (black dots) and their Voronoi diagram (dotted line). (b) The dispatch result when $\hat{v}(\mathbf{x})$ is uniform. Red pentagams are the new positions of the observers. Red lines display the corresponding Voronoi diagram. The blue dotted line denotes the travel path of each observer. (c) The dispatch result when $\hat{v}(\mathbf{x})$ is Gaussian. (d) The dispatch result when $\hat{v}(\mathbf{x})$ is like a crater. (e) $\hat{v}(\mathbf{x})$ is a Gaussian function. (f) $\hat{v}(\mathbf{x})$ is like a crater.

$\hat{v}(\mathbf{x})$ can result in different outputs by the proposed dispatch algorithm.

In Fig. 1(a), the black dots denote the initial positions of the observers and the dotted lines display the Voronoi diagram. As for $\hat{v}(\mathbf{x})$, three different cases are tested: (i) uniform. All the points have the same value of $\hat{v}(\mathbf{x})$. The dispatch result is shown in Fig. 1(b). The red pentagams are the new positions of the observers and the red lines display the corresponding Voronoi diagram. The blue dotted line denotes the travel path of each observer; (ii) Gaussian. $\hat{v}(\mathbf{x})$ decreases from the center to the edges (See Fig. 1(e)). The dispatch result is shown in Fig. 1(c). The observers are clustered in the center of the plane where the predicted variance function value is higher. Therefore the target function F is minimized; (iii) crater. Suppose the boundary under tracking is a circle and the predicted variance function is like a crater of a volcano (See Fig. 1(f)), which means the variance is higher if the point is closer to the circle. Fig. 1(d) shows the corresponding dispatch result. All the observers are dis-

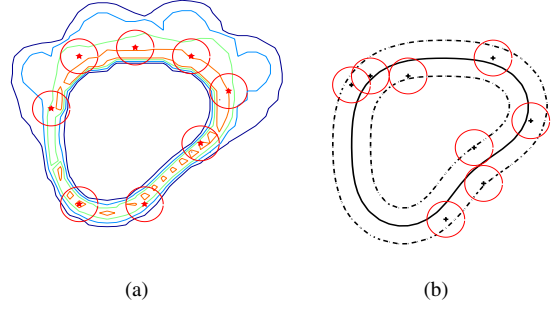


Fig. 2. (a) shows the variance-based optimal dispatch result where the red pentagams are the observers and the contours are the contours of the estimated prior variance function value of $\hat{v}(\mathbf{x})$ (from orange, green, cyan to blue, the variance value is decreasing). The red circles display the observation range of the observers. (b) shows the random dispatch result where the black dots are the observers. The black solid line is the last tracking result and the dotted lines enclosing a narrow band show the range where the observers are randomly dispatched.

patched uniformly along the circle. In our experiments, the convergence criterion is set as a threshold on the square sum of the distances between the new and old observer positions. If the observers move too little between iterations, the dispatch algorithm will stop. The number of iterations for the above three cases are 21, 23, 19 respectively.

This simulation test tells us that the dispatch algorithm distributes observers to the places with higher predicted variance values as expected.

5.2. Synthetic Test

We apply the proposed dispatch strategy in tracking dynamic planar boundaries in a synthetic data set. The ground truth is a 30 frame sequence of moving and deforming 2D boundaries. These are created by taking level sets of mixture of 2D Gaussian functions. The motion model is derived from the difference between the ground truth boundaries at consecutive time steps by an optical flow method [14]. The grid size is 50×50 and the total number of grid points is $N = 2500$.

The initial particle set is generated by different Gaussian perturbations based on the ground truth of the first frame. The total number of particles is $P = 80$. We test the tracking algorithm [2] with the proposed dispatch strategy for different number of measurements M and different observation range R . Accuracy is measured by the error rate defined as the ratio of the total symmetric differences between the tracking results and the ground truth over the total area enclosed by the ground truth boundaries from all frames. Specifically, it is defined as

$$\text{Error Rate} = \frac{\sum_t \|\tilde{\mathbf{b}}^{(t)} - \hat{\mathbf{b}}^{(t)}\|_2}{\sum_t \|\tilde{\mathbf{b}}^{(t)}\|_2} \quad (19)$$

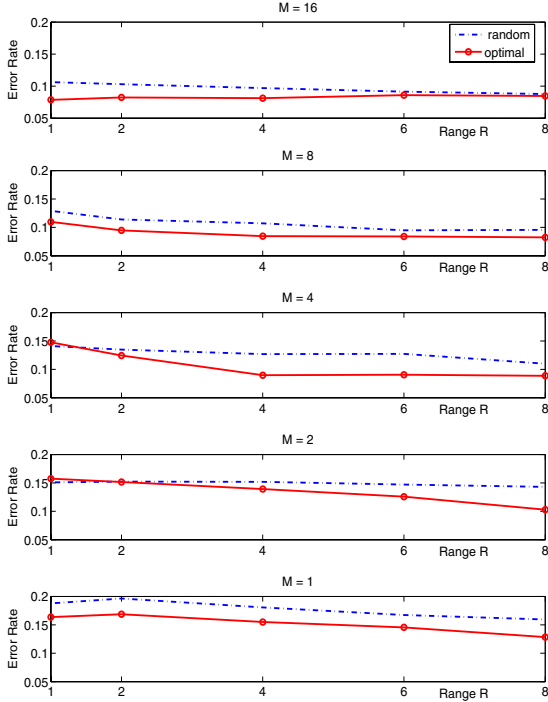


Fig. 3. Comparison of the error rates generated by the optimal dispatch and random dispatch strategies with the number of measurements $M = 1, 2, 4, 8, 16$ and the observation range $R = 1, 2, 4, 6, 8$ for boundary tracking in a synthetic data set. Dotted lines show the results from random dispatch strategy and solid lines show the results from optimal dispatch strategy. The grid size is 50×50 .

where $\tilde{\mathbf{b}}^{(t)}$ and $\hat{\mathbf{b}}^{(t)}$ denote the binary indicator vectors of the ground truth boundary and tracking result at time step t respectively. The tracking result $\hat{\mathbf{b}}^{(t)}$ is defined as the MAP particle after update.

To demonstrate the performance of the proposed dispatch strategy, we also experiment with the same tracking algorithm but a different dispatch strategy. We use a naive random dispatch strategy that reallocates the observers randomly inside a narrow band along the tracking result from previous time step. The bandwidth is set equal to $2R$. A comparison of these two different dispatch strategies is shown in Fig. 2 where (a) shows how the variance-based optimal dispatch strategy reallocate the observers to the area corresponding to high value of $\hat{v}(\mathbf{x})$ and (b) shows how the random dispatch strategy works with the same probability distribution of the boundary. Based only on the tracking result at last time step, the random strategy aimlessly dispatches observers inside the narrow band denoted by the dotted lines. The intersection between red circles implies that some observation resource is wasted.

These two experiments only differ by the dispatch strategy. All the other parts including the initial particle set and

observation model are same. The comparison of the experimental results from these two approaches are shown in Fig. 3. The displayed error rate for each M and each R is the average error rate of 20 trials. It is easy to see that the error rate has been improved by the optimal dispatch strategy compared to random dispatch strategy, especially when the total coverage by all the observers is small, such as $M = 16$ and $R = 1$, $M = 4$ and $R = 4, 6$, or $M = 2$ and $R = 8$. In other words, optimal dispatch uses less observers to achieve the same accuracy with same R compared to random dispatch.

5.3. Sea Surface Temperature Tracking

Sea surface temperature (SST) is the water temperature at the sea surface. SSTs above 26.5 degrees C are favorable for the formation and sustaining of tropical cyclones. In general, the higher the SST, the stronger the storm. So tracking the isotherms of high SSTs is very helpful in forecasting tropical cyclones and hurricanes.

Experimental validation of our proposed dispatch framework has two conflicting requirements: On one hand, sparseness of the observers is the main issue we are addressing. On the other hand, ground truth about the true location of the boundary is required to measure the performance. To meet both requirements, we have tracked the SST isotherms in a data set where we have dense information of the boundary, but where we simulate the sparseness of observers by withholding the dense information from our tracking and dispatch algorithm. Specifically, we obtain NOAA_ERSST_V2 data¹ which is an extended reconstruction of historical SST monthly mean values using improved statistical methods from 1854 to present and take it as ground truth. In particular, we track the isotherm at 25 degrees C. One example of the boundary (January 1990) is shown in Fig. 5. The observers are simulated by the points on the map and each observer can report the SSTs for the grid points within the observation range. In real applications, the observers could be a set of ships or buoys with temperature sensors. The underlying motion model for the isotherms of SST is very complicated because there are many factors which could affect SST. Since SST changes periodically, for simplicity, we learned a motion model for each calendar month (Jan. to Dec.) from the historical data between 1990 and 1999. For example, to get the motion model for January, we compare the data in January and February in every year and find a motion model by the optical flow method [14], and then take the average of the motion models over the ten years (Fig. 5).

We started tracking from January 2000 and ended in December 2000. The particle set was initialized by small perturbations based on the boundary in December 1999. The size of the particle set is $P = 200$. The grid size is 89×180 (2.0 degree latitude \times 2.0 degree longitude global grid,

¹Provided by the NOAA/OAR/ESRL PSD, Boulder, Colorado, USA, from their web site at <http://www.cgd.noaa.gov/>

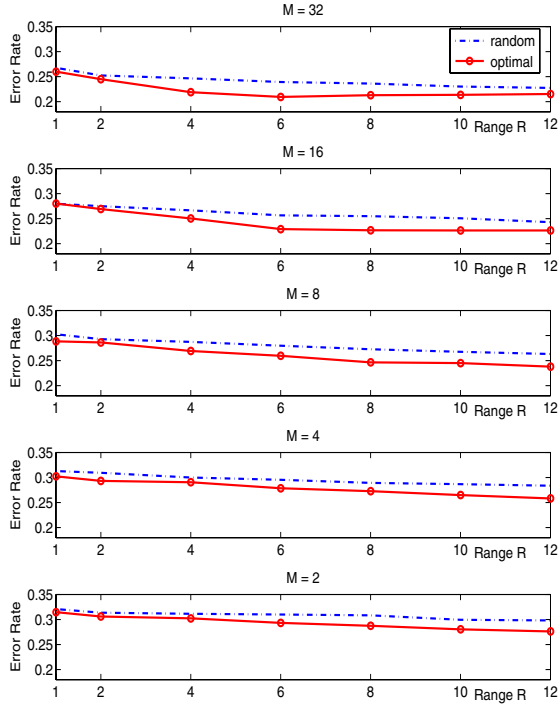


Fig. 4. Comparison of the error rates generated by the optimal dispatch and random dispatch strategies with the number of measurements $M = 2, 4, 8, 16, 32$ and the observation range $R = 1, 2, 4, 6, 8, 10, 12$ for SST tracking. Dotted lines show the results from random dispatch strategy and solid lines show the results from optimal dispatch strategy. The grid size is 89×180 .

$88.0N - 88.0S, 0.0E - 358.0E$). We have applied both the proposed dispatch strategy and the random dispatch strategy described in Section 5.2. The definition of the error rate is as in (19). The comparison of the tracking performance of these two strategies is shown in Fig. 4. The error rate shown for each M and each R is the average of 30 trials. The improvement of the error rate by the optimal dispatch strategy is apparent for most M and R unless the observation range R is too small ($R = 1, 2$).

6. SUMMARY AND FUTURE WORK

We have introduced a new framework for dispatching a set of mobile observers during boundary tracking to maximize the information gained about the boundary's state. The probability distribution of the boundary is represented by a particle filter where each particle is a binary indicator function of one possible boundary. We show that the dispatch problem can be formulated as the minimization of the sum of the posterior variances of the individual binary indicator values at each point. We propose an optimal dispatch strategy and compare

it to a random dispatch strategy with same initialization and observation resources for both synthetic data and real data. The comparison results show that our approach performs better when the observation resource is sparse but not rare.

For future work, we intend to generalize the proposed dispatch solution to include constraints on the motions of the observers. Our experiments suggest that there is a relationship between the performance of the proposed dispatch approach and the sparseness of the observers, where the sparseness depends on both the number of observers and the observation range. This suggests the presence of an optimal point of sparseness to maximize the benefit of the proposed dispatch algorithm.

7. REFERENCES

- [1] Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi, "Particle filtering for geometric active contours with application to tracking moving and deforming objects," *CVPR*, vol. 2, pp. 2–9, June 2005.
- [2] T. Jiang and C. Tomasi, "Level-set curve particles," in *ECCV*, A. Leonardis, H. Bischof, and A. Pinz, Eds. 2006, vol. 3953 of *LNCS*, pp. 633–644, Springer.
- [3] A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo in Practice*, Springer, New York, NY, 2001.
- [4] C. De Boor, *A Practical Introduction to Splines*, Springer, New York, NY, 2001.
- [5] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking," *IEEE Trans. of Signal Processing*, vol. 50, no. 2, pp. 174–188, February 2002.
- [6] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking," *IJCV*, vol. 29(1), pp. 5–28, 1998.
- [7] A. Blake and M. Isard, *Active Contours*, Springer, New York, 1999.
- [8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. of the ASME J. on Basic Eng.*, vol. 82, pp. 34–45, 1960.
- [9] C. Guestrin, A. Krause, and A. Singh, "Near-optimal sensor placements in Gaussian processes," in *ICML*, L. Raedt and S. Wrobel, Eds. 2005, pp. 265–72, ACM.
- [10] M. Schwager, J. McLurkin, and D. Rus, "Distributed coverage control with sensory feedback for networked robots," in *Proceedings of Robotics: Science and Systems*, Philadelphia, USA, August 2006.
- [11] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," in *ICRA*. 2002, pp. 1327–1332, IEEE.
- [12] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: Applications and algorithms," *SIAM Review*, vol. 41(4), pp. 637–676, 1999.
- [13] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 129–137, Mar. 1982.
- [14] B. Horn and B. Schunk, "Determining optical flow," *Artificial Intelligence*, vol. 20, 1981.

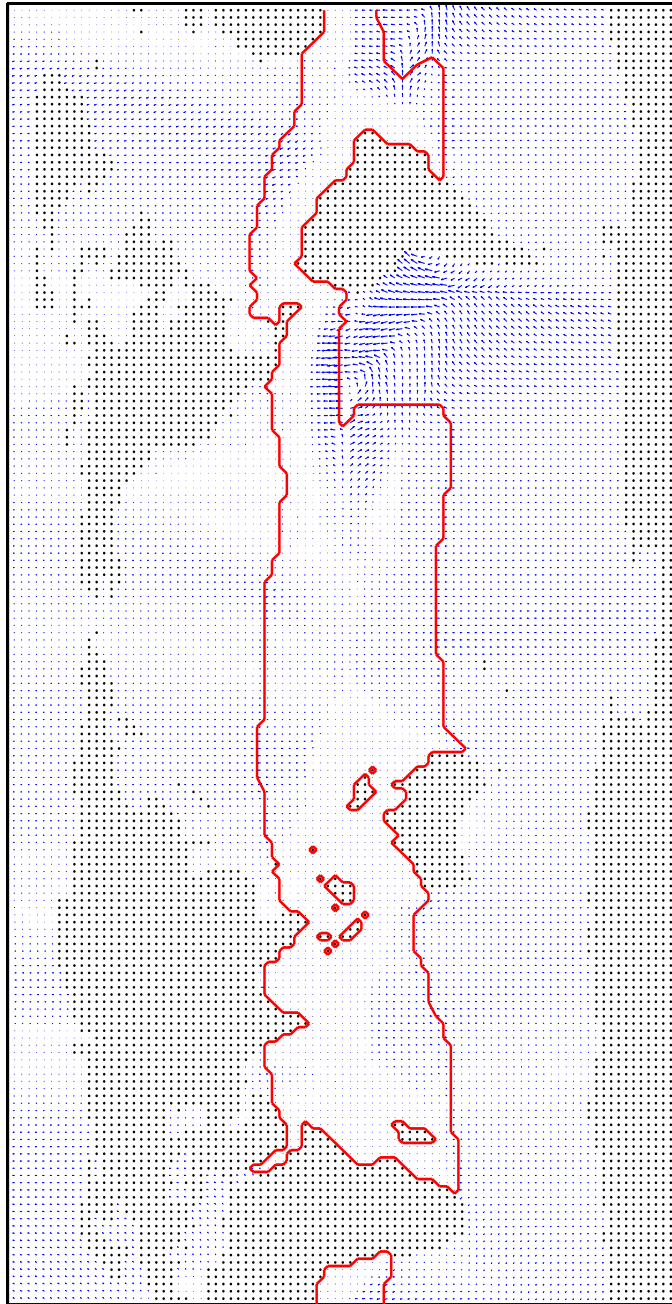


Fig. 5. The red curve is the isotherm of SST at 25 degrees C in January 1990 and the blue arrows denote the learned motion model for the isotherm at 25 degrees C in January from the SST data between 1990 and 1999. The black dots denote land.