Deployment algorithms for dynamically constrained mobile robots

Sonia Martínez

Abstract The use of unmanned vehicles in exploration and surveillance operations has become evermore pervasive in today's world. The development of cooperative motion strategies has been fueled by this increasing demand. However, many dynamical models for these autonomous vehicles remain simple and are not accurate representations of a vehicle where such cooperative motion strategies may be physically implemented. This paper reviews complementary solutions to the problem of cooperative deployment of autonomous vehicles using multi-center functions. In particular, vehicles are subject to three types of dynamic constraints, such as those due to remaining power supplies, nonholonomic dynamics, and constraints due to external environmental forces. Simulations illustrate the convergence properties of the algorithms when applicable.

1 Introduction

The study of coordination mechanisms in multi-agent systems is relevant for both the understanding of scientific phenomena and the development of new technologies. A main class of examples from nature is given by swarming in animal species such as ant colonies, termites, flocking birds, and schooling fish. Emergence and self-organization is also a characteristic of human politics, societies, and economic groups. In these groups, each member makes decisions repeatedly based on local information signals sent or left by other members of the group and the environment. This decentralized process with no leaders yields complex emerging behaviors that translate into robust and efficient global structures, patterns, or organizations.

Complex systems in biology and society can help us understand, model, and design large-scale engineered systems composed of autonomous and semi-autonomous

Sonia Martínez
MAE - UC San Diego, 9500 Gilman Dr, La Jolla, CA 92093-0411 e-mail: soniamd@ucsd.edu

agents. Their potential advantages are those found in their biological counterparts—robustness to failure thanks to system redundancy, and increased efficiency in the number and quality of the global tasks that can be accomplished. However, the realization of multi-agent systems poses new challenges induced by scalability problems, agent heterogeneity, and intermittent interactions in uncertain, dynamically changing environments.

Research in mobile robotics has helped enormously in the understanding of these challenges through the study of several important benchmark problems including *rendezvous, formation control, deployment, and task assignment*. In particular, the problem of robot deployment to provide better coverage or task servicing in an environment lends itself to geometric optimization formulations, which have been extensively studied. An incomplete list of references on coverage includes [1, 2], based on potential field methods, [3] using the theory of coverage point processes, [4, **?**] making use of non-smooth analysis techniques, and [5, 6, 7] based on behavioral control approaches. More recently, the robotics community is developing new non-model based algorithms for coordinated deployment and map building [8].

An alternative, popular approach makes use of multi-center performance metrics and Voronoi-based control algorithms to stabilize multi-robot systems to locally optimal positions [9, 10]. This approach has been adapted to deal with non-convex environments with obstacles [11, 12, 13], distributed environment learning [14, 15], and equitable partitions [16, 17]. The resulting algorithms can be adapted so that limited-range, distributed interactions are possible as well [18].

More recently, different researchers have started to reconsider the difficulties introduced by vehicle dynamics. Coordination algorithms typically assume simple dynamics for vehicles. The idea is to implement this strategy as a high-level plan, together with low-level local motion plans that each vehicle uses to reconfigure to the prescribed upper-level positions. If synchronization is possible, each vehicle can wait for others to reach their positions before moving forward. However, in asynchronous regimes this strategy can be just infeasible. On the other hand, dynamic constraints may require a re-definition of the deployment objectives in order to produce more meaningful solutions. However, the inclusion of constraints in the coordination objectives can impose additional computational and control challenges.

In this chapter, we review and summarize several extensions of Voronoi-based deployment to account for different types of dynamical constraints. These include power constraints, and vehicle controllable and uncontrollable dynamics. The paper is organized as follows. In Section 2, we review the basic Lloyd's algorithm to the Locational Optimization problem for coverage control. In Section 3 we address the problem of deployment under power limitation to vehicle motion. Section 4 adapts Lloyd's algorithm to deal with Dubin's type of unicycles via a hybrid coordinated motion law. Finally, Section 5 introduces a heuristic to deal with underactuated and uncontrollable vehicles in river environments.

2 Benchmark problem: coverage control

In this section, we present a basic coverage control problem formulation together with gradient-based algorithmic solutions for unconstrained vehicle dynamics [9, 10]. Some of these will be extended in the following sections to account for different types of dynamical constraints.

Basic coverage and task-assignment objectives can be formulated by a meaningful class of Locational Optimization or multicenter performance metrics. Let $Q \subseteq \mathbb{R}^2$ be a convex, bounded environment, and $\phi : Q \to \mathbb{R}_{\geq 0}$ be a scalar field with bounded support Q. Here, ϕ represents an *a priori* measure of information on Q—the higher the value of $\phi(q)$, $q \in Q$, the more attention that should be afforded to $q \in Q$. Let $P = (p_1, \dots, p_n)$ denote the agent positions in Q. In the following, we interchangeably refer to the elements of the network as sensors, agents, vehicles, or robots. Let $f : \mathbb{R} \to \mathbb{R}$ be a non-decreasing and piecewise differentiable function relating the Euclidean distance from p_i to $q \in Q$, $\|p_i - q\|$, to coverage performance from p_i on q, for all $i \in \{1, \dots, n\}$. For example, the function f can encode the signal-to-noise ratio between a source with location q and the sensor located at p_i. Or it can define the cost of servicing a location q by an agent placed at p_i; e.g. the traveling time from p_i to q when moving on a straight line with constant velocity. With these elements, a coverage metric can be defined as:

$$\mathscr{H}(P) = \int_Q \min_{p_i} f(\|p_i - q\|) dq \,, \tag{1}$$

where $\min_{p_i} f(\|p_i - q\|)$ has the interpretation of the best coverage of q provided by the multi-robot system. The minimization of this metric results into a minimum average cost to cover Q using the multi-robot group.

For the purpose of defining a distributed algorithm that optimizes this metric, it is helpful to restate (1) in terms of the individual contribution that each agent in the network adds to $\mathscr{H}$. For example, assume that $f(x) = x^2$ for all $i \in \{1, \dots, n\}$ and denote the associated $\mathscr{H}$ by $\mathscr{H}_{\mathrm{centr}}$. Let $\mathscr{V}(P) = (V_1, \dots, V_n)$ be the so-called *Voronoi partition* of Q, where

$$V_i = \{q \in Q \mid \|q - p_i\| \leq \|q - p_i\|, \forall\, j \neq i\} \,, \qquad \forall\, i \in \{1, \dots, n\}, \tag{2}$$

satisfy $\cup_{i=1}^n V_i = Q$. Then $\mathscr{H}$ may be rewritten as

$$\mathscr{H}(P) \equiv \mathscr{H}(P, \mathscr{V}(P)) = \sum_{i=1}^{N} \int_{V_i} \|q - p_i\|^2 \phi(q) dq \,. \tag{3}$$

Given a region $W \subseteq Q$, one can define its *mass* and *centroid*, as follows:

$$M_W = \int_W \phi(q) dq \,, \qquad \mathrm{CM}_W = \int_W q\phi(q) dq \,. \tag{4}$$

It can be shown, see [19, 9], that if agents are in a *centroidal Voronoi configuration*; that is, $p_i = \mathrm{CM}_{V_i}$ for all i, then the cost function $\mathscr{H}$ is at a local minimum.

An alternative metric that considers flat and limited sensor footprints (resp. travel ranges) R can be obtained by taking $f(\|p-q\|) = -1_{[0,R)}(\|p-q\|)$, which leads to:

$$\mathscr{H}_{\mathrm{area}}(P) = -\int_{\cup_{i=1}^{n} B(p_i,R)} \phi(q)dq,.$$

The minimization of this metric results into a maximization of the area covered by the group of agents. This objective can be combined with the previous one, leading to a mixed metric of the form $\mathscr{H}_{\mathrm{mixed}}(P) = \alpha\mathscr{H}_{\mathrm{centr}} + \beta\mathscr{H}_{\mathrm{area}}(P)$, for α, $\beta \geq 0$.

Once a metric is chosen, a gradient-based distributed control algorithm can be implemented by each agent to asymptotically reach the corresponding set of local minima. The following law is a continuous-time version of the algorithm in [9]:

$$\dot{p}_i = -\mathrm{sat}\left(\frac{\partial \mathscr{H}_{\mathrm{centr}}}{\partial p_i}\right) = -M_{V_i}(\mathrm{CM}_{V_i} - p_i), \qquad i \in \{1,\dots,n\},$$

where the function $\mathrm{sat}(v) = v$, if $\|v\| \leq 1$, otherwise $\mathrm{sat}(v) = \frac{v}{\|v\|}$. Essentially, agents need to be able to compute the regions V_i, and follow the corresponding centroid. If Q is compact, agents will converge to centroidal locations. The corresponding Voronoi regions may be computed by agents using information of a limited set of other agents, the Delaunay neighbors. Thus, the algorithm is distributed in the sense of the Delaunay graph. Correspondingly, the metrics $\mathscr{H}_{\mathrm{area}}$ and $\mathscr{H}_{\mathrm{mixed}}$ give rise to algorithms that are distributed in the sense of the $2R$-disk graph (for appropriate α and β). Discrete-time versions of these algorithms can be seen to be convergent even if partial asynchronous behavior is permitted [9, 10].

A simulation of the above gradient algorithm is provided below.

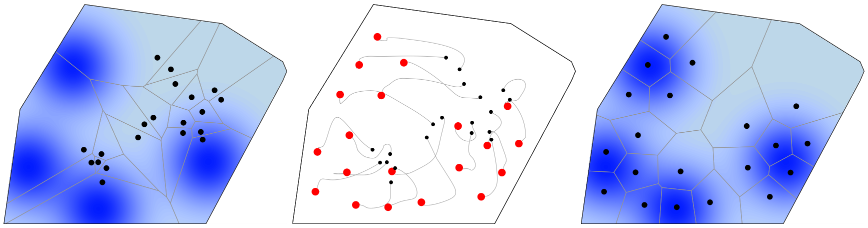

Fig. 1 A simulation run of the gradient-based algorithm associated with $\mathscr{H}_{\mathrm{centr}}$. The figure on the left depicts robots' initial positions. The figure on the right represents robots' final positions and correspond to centroidal Voronoi configurations.

3 Power-limited deployment

Power-aware algorithms have been the subject of extensive research in static sensor networks and mobile middleware, see [20, 21]. However, there is limited work on power constraints, and how these may affect cooperative control algorithms. For instance, the final agent configurations provided by Figure 1 for a task assignment objective do not seem reasonable when agents have different motion restrictions.

Here we describe a first approach [22] to deal with this problem in the context of the Locational Optimization or multicenter type of metrics of Section 2. We assume that there is enough group redundancy so that the loss of a particular agent can be afforded. This allows to account for energy limitations by means of a modified performance metric of the form of (1) and associated generalized Voronoi partitions.

Let $P=(p_1,\dots,p_n)$ be the positions of n robots in Q. The sensors have an associated energy content E_i such that $0 \le E_i \le E_{\max}$, for all $i \in \{1,\dots,n\}$. As agents move, their energy reserve will decrease. We propose the following simple agent dynamics in the augmented state $(p_i,E_i) \in Q \times \mathbb{R}_{\geq 0}$:

$$\dot{p}_i = u_i \,, \qquad \dot{E}_i = -g_i(\|\dot{p}_i\|)\,, \tag{5}$$

where $\dot{p}_i$ denotes the velocity of agent i such that $\|\dot{p}_i\| \in [0,v_{\max}]$, u_i is the control input, and $g_i : [0,v_{\max}] \to \mathbb{R}_{\geq 0}$ is any increasing function such that $g_i(x)=0$ only at $x=0$. Intuitively, $g_i(x)$ captures the fact that energy expenditure increases as velocity increases. This modeling assumption is based on the consideration that power is consumed to change absolute speed and counteract drag forces; the latter being the predominant force [22]. In the following, we will take $g_i(x)=g(x)=x^2$, for all $i \in \{1,\dots,n\}$, for simplicity.

Energy expenditure will affect the travel range that a sensor can cover with maximum velocity before running out of batteries. Suppose that agent i travels with a maximum velocity $\dot{p}_i(t) = (v_{\max},0)^T \in \mathbb{R}^2$. Then, the vehicle runs out of energy at time $T(v_{\max}) = E_i(0)/g(v_{\max})$. The associated travel range is the distance $R_* = v_{\max} T(v_{\max})$. This motivates the use of a mixed type of performance metric as in Section 2 that accounts for travel-range limits. Thus, we consider:

$$\mathscr{H}(P,E) = \int_Q \min_{i \in \{1,\dots,n\}} f_i(d_{E_i}(q,p_i))\phi(q)dq\,, \tag{6}$$

where $E=(E_1,\dots,E_n)$ are current energy levels of agents, the maps $f_i : \mathbb{R} \to \mathbb{R}$ are non-decreasing functions associated with the travel cost of each agent i, and $d_{E_i} : Q \times Q \to \mathbb{R}$ is a weighted (quasi) pseudo-metric function such as the following:

1. The power-weighted metric, $d_{E_i,\,\mathrm{pow}}(q,p_i) = \|q-p_i\|^2 - (E_i)^2$,
2. The multiplicatively-weighted metric, $d_{E_i,\,\mathrm{mult}}(q,p_i) = \frac{1}{E_i^2}\|q-p_i\|^2$,
3. The additively-weighted metric, $d_{E_i,\,\mathrm{aw}}(q,p_i) = \|q-p_i\| - (E_i)^2$.

All these metrics lead to generalized Voronoi regions [19] whose size depends on the relative energy content of neighboring robots:

$$V_i^{\text{gen}} = \{q \in Q \,|\, d_{E_i}(q,p_i) \le d_{E_i}(q,p_j), \forall\, j \ne i\}\,, \text{ where } d_{E_i} \text{ is a pseudometric.}$$

Boundaries of these Voronoi regions are (1) straight lines for the power metric, (2) circles of radii E_i, $i \in \{1,\dots,n\}$, for the multiplicatively-weighted metric, and (3) hyperbolic boundaries for the additively-weighted metric. Due to the difficulty of representing and intersecting hyperbolic boundaries, we focus on the first two types. As opposed to standard Euclidean Voronoi regions, these generalized regions can be non-convex and their generators may lie outside them. If vehicles only have a limited amount of energy to move using a maximum velocity, E_i, $i \in \{1,\dots,n\}$, we propose that in order for an agent i to be able to cover a point $q \in Q$, agent i must be able to reach q with its current energy level. This leads to new assignment regions for agents given by the intersection of Voronoi regions with circles of radii E_i, $i \in \{1,\dots,n\}$.

Let $B_i(E_i)$ be a closed ball centered at p_i with radius E_i. Then, the space that can be covered by the robots to $\cup_{i=1}^n B_i(E_i) \subseteq Q$. The new limited-Voronoi regions assigned to each agent are defined as follows:

$$V_{\text{LD},\,i}^{\text{gen}} = \{q \in Q \,|\, d_{E_i}(q,p_i) \le d_{E_i}(q,p_j), \forall\, j \ne i \text{ and } \|q-p_i\| \le E_i\}\,,$$

where d_{E_i} is one of the pseudometrics above. Figure 2 compares limited-Voronoi regions.

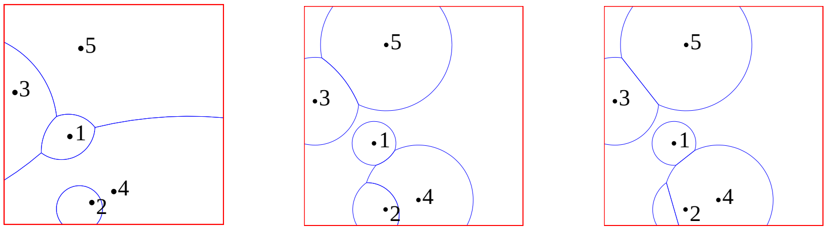

Fig. 2 Figures from left to right: (i) Voronoi partition associated with the multiplicatively weighted pseudometric, (ii) limited-range cells associated with the multiplicatively-weighted pseudometric, (iii) limited-range cells associated with the power-weighted pseudometric.

The computation of limited types of regions are spatially distributed over the Delaunay graph and the third one over the $2E_{\max}$-disk graph. The new regions emphasize different energy levels of agents. The corresponding metrics for centroidal, area, and mixed coverage are then given respectively as follows:

$$\mathscr{H}_{\text{centr}}(P,E) = \int_{\cup_{i=1}^{n} B_i(E_i)} \min_{i\in\{1,\dots,n\}} \{d_{E_i}(q,p_i)\}\phi(q)dq\,,$$
$$\mathscr{H}_{\text{area}}(P,E) = \int_{Q} \min_{i\in\{1,\dots,N\}} (-1_{[0,E_i]}(\|q-p_i\|))\phi(q)dq = -\int_{\cup_{i=1}^{n} B_i(E_i)} \phi(q)dq\,,$$
$$\mathscr{H}_{\text{mixed}}(P,E) = \kappa_{\text{area}}\mathscr{H}_{\text{area}}(P,E) + \kappa_{\text{cent}}\mathscr{H}_{\text{cent}}(P,E)\,.$$

In particular, it is still possible to rewrite $\mathscr{H}_{\text{centr}}(P,E)$ as follows:

$$\mathscr{H}_{\text{centr}}(P,E) = \sum_{i=1}^{n} \int_{V_{\text{LD, }i}^{\text{gen}}} d_{E_i}(q,p_i)\phi(q)dq\,.$$

For any of these functions $\mathscr{H}$, we can define a gradient descent control algorithm for agents as follows:

$$\dot{p}_i = -k^*(p_i,E_i)\text{sat}\left(\frac{\partial\mathscr{H}}{\partial p_i}\right), \qquad k^*(p_i,E_i) = \frac{\text{sat}\left(\frac{\partial\mathscr{H}}{\partial p_i}\right)\cdot\left(\frac{\partial\mathscr{H}}{\partial p_i}\right)}{2\left|\text{sat}\left(\frac{\partial\mathscr{H}}{\partial p_i}\right)\right|^2\frac{\partial\mathscr{H}}{\partial E_i}}$$
$$\dot{E}_i = -\|\dot{p}_i\|^2, \qquad i\in\{1,\dots,n\}. \tag{7}$$

That is, assuming that energy decreases according to $\|\dot{p}_i\|^2$, we modulate the velocity of agents via the gain k_i^* while collectively decreasing the cost function $\mathscr{H}$. In this way, vehicles with lower energy will spend less in moving toward their goal positions, while vehicles with larger energy will spend more. With this strategy, controllability to critical positions is possible for those agents that have enough energy to move. For the particular case of $\mathscr{H}_{\text{centr}}(P,E)$, the algorithm makes agents follow the centroids of the corresponding generalized Voronoi regions if they do not run out of energy. Convergence is stated in the following theorem.

Theorem 1 (Critical configurations for centroidal coverage and MWVD, [23]). *The critical points of a gradient descent flow characterized by* (7) *using an objective function* $\mathscr{H}_{\text{centr}}$ *are configurations where each agent is either:*

1. *located at the centroid,* $p_i = \text{CM}_{V_{\text{LD},i}^{gen}}$*,*
2. *has no energy,* $E_i = 0$*.*

Agents approach these critical configurations as $t\to\infty$*.*

A simulation run of the energy-aware gradient-descent algorithm for a mixed metric $\mathscr{H}_{\text{mixed}}$ is provided in Figure 3. Agents that need to travel further away, will eventually have smaller assigned regions. In this case vehicles end up at the centroids of their regions. Similar convergence results can be established for the other cost functions such as $\mathscr{H}_{\text{area}}$. The number of agents that can run out of energy depends on the initial agents' positions, the density $\phi(q)$, and size of the environment. The extension of these results to deal with asynchronous implementations can be done in a similar way to [9].

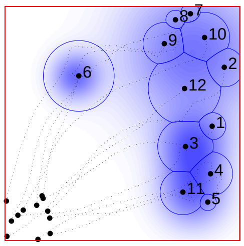

Fig. 3 Multi-robot system trajectories evolving under (7) for $\mathscr{H}_{\text{centr}}$. The final energy regions, and final agent energy levels, are also shown here.

4 Unicycle vehicle dynamics

Vehicles' actuation can severely be affected by dynamical constraints on their positions and velocities. For example, the control laws of the previous section force individual agents to move directly towards the centroid of their Voronoi regions; however, this is not always possible. When vehicles are controllable, one can consider vehicle dynamics into the design of coordination algorithms from the start. This can help avoid severe performance degradation due to lack of synchronization.

In the following we introduce control algorithms that propel a class of nonholonomic vehicles to centroidal Voronoi configurations while the minimization of $\mathscr{H}_{\text{centr}}$ is satisfied in certain sense. This results into a hybrid system that can be analyzed via the novel theory of [24, 26]. Suppose that vehicles are deployed in a convex, compact environment Q. Referencing Figure 4, each vehicle has configuration variables $(p_i, \theta_i) \in \text{SE}(2)_Q$, and a body coordinate frame with basis $e_{i,1} = (\cos\theta_i, \sin\theta_i)$ and $e_{i,2} = (-\sin\theta_i, \cos\theta_i)$. We denote $d_i = \text{CM}_{V_i} - p_i$ as in Figure 4 and define the angle $\Omega_i \in [-\pi, \pi]$ to be the angle between $e_{i,1}$ and d_i. We assume the vehicles have bounded velocity and turning rate, $|v_i| \leq v_{\max}$ and $|\omega_i| \leq \omega_{\max}$ respectively. Next, we introduce a Dubin's type of nonholonomic vehicles that we shall consider.

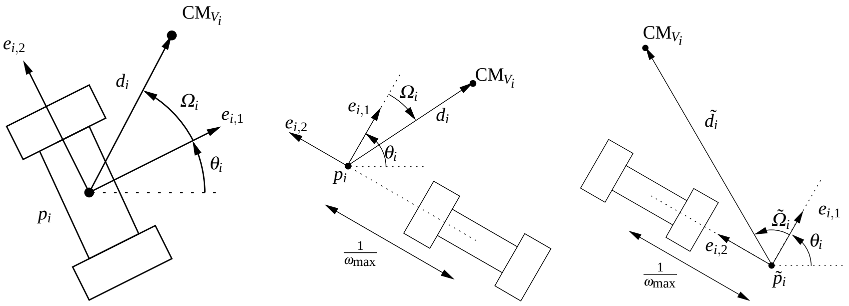

Fig. 4 Vehicle with wheeled mobile dynamics (left). The variables are redefined for a vehicle with fixed forward velocity and a left virtual center (middle). The non-active virtual center quantities are shown with a tilde, ˜ (right).

Vehicles with fixed forward velocity. Simple models for UAVs and underwater gliders can be modeled as vehicles with constant forward velocity, constant altitude and a minimum turning radius such as the following [25]:

$$\dot{p}_i = (\cos\theta_i, \sin\theta_i)^T, \qquad \dot{\theta}_i = \omega_i\,,$$

where ω_i is the only input. Define the vehicle *virtual center* as its center of rotation when the turning input is $\pm\omega_{\max}$. These centers can be on either side of the vehicle, and a strategy to switch virtual center locations will be introduced later. Our coverage objective will be formulated in terms of the virtual center of each vehicle to a desired centroid target. This target will be the centroid of the vehicle's Voronoi region calculated using the virtual centers of all vehicles in the network. Once the virtual center has arrived at the centroid, the vehicle will *hover* about it by maintaining the maximum steering input $\pm\omega_{\max}$.

The virtual centers' coordinates in the global frame are

$$p_i{}' = p_i \pm \frac{1}{\omega_{\max}}(-\sin\theta_i, \cos\theta_i)^T\,. \tag{8}$$

with time derivative:

$$\dot{p}_i{}' = \dot{p}_i \pm \frac{1}{\omega_{\max}}(-(\cos\theta_i)\dot{\theta}_i, -(\sin\theta_i)\dot{\theta}_i)^T = \left(1 \mp \frac{\omega_i}{\omega_{\max}}\right)(\cos\theta_i, \sin\theta_i)^T\,. \tag{9}$$

Indeed, with $\omega_i = \pm\omega_{\max}$, the vehicle is hovering since the virtual center remains fixed, $\dot{p}_i' = 0$. At any point in time, the current virtual center is chosen by a vehicle to be located on either side of the direction of travel. To simplify notation, let a current virtual center be p_i', and the opposite virtual center be $\tilde{p}_i' = p_i' \pm \frac{2}{\omega_{\max}} e_{i,2}$. Let $\tilde{d_i} = \mathrm{CM}_i(\mathcal{V}(P)) - \tilde{p}_i$ and let $\tilde{\Omega}_i$ denote the angle between $e_{i,1}$ and $\tilde{d_i}$, see Figure 4.

Each vehicle will either be in forward motion or hovering motion about one of the centers. This will result into four possible modes of operation for each vehicle depending on the center location: forward-left, hover-left, forward-right, and hover-right. We enumerate each mode with the state $l_i \in \{1,2,3,4\}$, thus we describe each agent by a state variable, $x_i \in \mathrm{SE}(2)_Q \times \{1,2,3,4\}$, and the multi-agent system state by $x = (x_1, \ldots, x_N) \in \mathbb{R}^{4N}$.

The choice of the center for each vehicle is based on the following observation. Starting arbitrarily with a center position p_i', we propose that each vehicle can switch to the other center, $\tilde{p}_i$, only if the actual improvement in cost satisfies $\mathcal{H}(P, \mathcal{V}(P)) - \mathcal{H}(\tilde{P}, \mathcal{V}(\tilde{P})) \geq \beta$, where $\beta \geq 0$ is a fixed constant, $P = (p_1', \ldots, p_i', \ldots, p_n')$ and $\tilde{P} = (p_1', \ldots, \tilde{p}_i, \ldots p_n')$ considers the new virtual center position $\tilde{p}_i$. This improvement can be evaluated locally by each vehicle, by knowing V_i.

We now describe more precisely the hybrid system that formalizes the cooperative algorithm for the multi-UAV group. The system state-space is $\mathrm{SE}(2)_Q \times \{1,2,3,4\} \subseteq O = \mathbb{R}^{4N}$. First, the sets $A_{i,1}, \ldots, A_{i,4}$ define the states where each vehicle i can flow continuously in each of the four modes, and are given as follows:

(1) An individual vehicle can be in $A_{i,1}$ (resp. $A_{i,3}$) if the centroid is in front of the left (resp. right) virtual center at p_i', and if p_i' is not sufficiently close to $\mathrm{CM}_i(\mathscr{V})$. Additionally, the improvement from switching between forward-left to forward-right (resp. vice-versa) must be better than β. However, if the opposite virtual center $\tilde{p}_i'$ is not in Q, then the vehicle may maintain its current virtual center despite violating the improvement threshold β:

$$\begin{aligned}
A_{i,1} = \{&x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{1\},\; e_{i,1} \cdot d_i \geq \underline{\varepsilon},\; M_i\|d_i\|^2 - M_i\|\tilde{d_i}\|^2 \leq \beta,\; \|d_i\| \geq \varepsilon\} \\
&\cup \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{1\},\; e_{i,1} \cdot d_i \geq \underline{\varepsilon},\; \tilde{p}_i' \in \overline{Q^c}, \|d_i\| \geq \varepsilon\}, \\
A_{i,3} = \{&x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{3\},\; e_{i,1} \cdot d_i \geq \varepsilon,\; M_i\|d_i\|^2 - M_i\|\tilde{d_i}\|^2 \leq \beta,\; \|d_i\| \geq \varepsilon\} \\
&\cup \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{3\},\; e_{i,1} \cdot d_i \geq \underline{\varepsilon},\; \tilde{p}_i' \in \overline{Q^c},\; \|d_i\| \geq \varepsilon\},
\end{aligned}$$

(2) A vehicle can be in $A_{i,2}$ (resp. $A_{i,4}$) if $\mathrm{CM}_i(\mathscr{V})$ is behind the left (resp. right) virtual center p_i', or if p_i' is on the boundary Q and heading outwards, or if p_i' is sufficiently close to $\mathrm{CM}_i(\mathscr{V})$:

$$\begin{aligned}
&A_{i,2} = \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{2\}, e_{i,1} \cdot d_i \leq \varepsilon, \|d_i\| \geq \varepsilon\} \cup \\
&\{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{2\}, e_{i,1} \cdot \hat{\mathbf{n}}_{\mathrm{in}} \leq 0\} \cup \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{2\}, \|d_i\| \leq \overline{\varepsilon}\}, \\
&A_{i,4} = \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{4\}, e_{i,1} \cdot d_i \leq \varepsilon, \|d_i\| \geq \varepsilon\} \cup \\
&\{x \in O \mid x_i \in \mathrm{SE}(2)_{\partial Q} \times \{4\}, e_{i,1} \cdot \hat{\mathbf{n}}_{\mathrm{in}} \leq 0\} \cup \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{4\}, \|d_i\| \leq \overline{\varepsilon}\}.
\end{aligned}$$

The hysteresis variables $0 < \underline{\varepsilon} < \varepsilon < \overline{\varepsilon}$ serve to insure that Zeno effects do not occur. Combining these sets together, the entire hybrid system flow set is $A = \bigcap_{i=1}^{N} (A_{i,1} \cup A_{i,2} \cup A_{i,3} \cup A_{i,4})$. When the system configuration $x \in A$, the state evolves under the $\dot{x} = F(x)$, where $F(x)$ is defined as follows. First, let $F_i(x)$ with:

$$\begin{aligned}
F_{i,1}(x) &= \left(\cos\theta_i, \sin\theta_i, \frac{2\Omega_i \omega_{\max}}{\pi}, 0\right)^T, & F_{i,2}(x) &= (\cos\theta_i, \sin\theta_i, \omega_{\max}, 0)^T, \\
F_{i,3}(x) &= \left(\cos\theta_i, \sin\theta_i, \frac{2\Omega_i \omega_{\max}}{\pi}, 0\right)^T, & F_{i,4}(x) &= (\cos\theta_i, \sin\theta_i, -\omega_{\max}, 0)^T.
\end{aligned}$$

Then, $F(x) = (F_1(x), \cdots, F_N(x))^T$, $F_i(x) = F_{i,k}(x)$ if and only if $l_i = k \in \{1,2,3,4\}$.

We now describe the set where discrete jumps can occur. We will consider:

1. Switching from forward-left to forward-right:
   $B_{i,1} = \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{1\}, e_{i,1} \cdot d_i \geq \underline{\varepsilon}, M_i(\|d_i\|^2 - \|\tilde{d_i}\|^2) \geq \beta, \tilde{p}_i \in Q\}$,
2. Switching from forward-right to forward-left:
   $B_{i,2} = \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{3\}, e_{i,1} \cdot d_i \geq \underline{\varepsilon}, M_i(\|d_i\|^2 - \|\tilde{d_i}\|^2) \geq \beta, \tilde{p}_i \in Q\}$,
3. Switching from forward-left to hover-left:

   $$\begin{aligned}
   &B_{i,3} = \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{1\}, e_{i,1} \cdot d_i \leq \underline{\varepsilon}\} \cup \\
   &\{x \in O \mid x_i \in \mathrm{SE}(2)_{\partial Q} \times \{1\}, e_{i,1} \cdot \hat{\mathbf{n}}_{\mathrm{in}} \leq -\varepsilon\} \cup \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{1\}, \|d_i\| \leq \varepsilon\},
   \end{aligned}$$

4. Switching from hover-left to forward-left:
   $B_{i,4} = \{x \in O \mid x_i \in SE2_Q \times \{2\}, e_{i,1} \cdot d_i \geq \varepsilon, e_{i,1} \cdot \hat{\mathbf{n}}_{\mathrm{in}} \geq 0, \|d_i\| \geq \overline{\varepsilon}\}$

5. Switching from forward-right to hover-right:

$$B_{i,5} = \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{3\},\, e_{i,1} \cdot d_i \leq \underline{\varepsilon}\} \cup$$
$$\{x \in O \mid x_i \in \mathrm{SE}(2)_{\partial Q} \times \{3\}, e_{i,1} \cdot \hat{\mathbf{n}}_{\mathrm{in}} \leq -\varepsilon\} \cup \{x \in O \mid x_i \in \mathrm{SE}(2)_Q \times \{3\},\, \|d_i\| \leq \varepsilon\},$$

6. Switching from hover-right to forward-right:
$B_{i,6} = \{x \in O \mid x_i \in SE2_Q \times \{4\},\, e_{i,1} \cdot d_i \geq \varepsilon,\, e_{i,1} \cdot \hat{\mathbf{n}}_{\mathrm{in}} \geq 0,\, \|d_i\| \geq \overline{\varepsilon}\}$.

The switching domain is the union $B = \bigcup_{i=1}^{N} \bigcup_{k=1}^{6} B_{i,k}$. The jump map G is then defined as follows. First, let $g_{i,1}(x), \ldots, g_{i,6}(x)$ be the maps for an individual vehicle i. These maps are:

$$\begin{aligned} g_{i,1}(x) &= (3, p_i - \frac{2}{\omega_{\max}} e_{i,2}, \theta_i), & g_{i,2}(x) &= (1, p_i + \frac{2}{\omega_{\max}} e_{i,2}, \theta_i), \\ g_{i,3}(x) &= (2, p_i, \theta_i), & g_{i,4}(x) &= (1, p_i, \theta_i), \\ g_{i,5}(x) &= (4, p_i, \theta_i), & g_{i,6}(x) &= (3, p_i, \theta_i). \end{aligned}$$

We combine the above functions for each vehicle and obtain

$$G_i(x) = \big\{(x_1, \ldots, g_{i,k}(x), \ldots, x_N) \mid x \in B_{i,k},\ \text{for } k \in \{1, \ldots, 6\}\big\}.$$

The complete set-valued jump map is then $G(x) = \bigcup_{i=1}^{N} G_i(x)$.
Concisely, the hybrid system of unicycles is described as

$$\begin{aligned} \dot{x} &= F(x), & x &\in A, \\ x^+ &\in G(x), & x &\in B. \end{aligned}$$

It can be seen that the system satisfies the Basic Conditions of [26], Section VI. This allows us to apply the hybrid LaSalle invariance theorem derived therein:

Theorem 2 (Goebel, Sanfelice, Teel [27]). *Given a hybrid system (F, G, A, B) on a state space $O \subseteq \mathbb{R}^M$ which satisfies the Basic Conditions, suppose that:*

1. *there is a $V : O \to \mathbb{R}$, Lyapunov function continuous on O and Locally Lipschitz on a neighborhood of A,*
2. *$\mathscr{U} \subseteq O$ is non-empty,*
3. *$u_A(x) = \max_{f \in F(x)} \mathscr{L}_f V(x) \leq 0$, for all $x \in A$,*
4. *$u_B(x) = \max_{x^+ \in G(x)} (V(x^+) - V(x)) \leq 0$, for all $x \in B$.*

Let x be precompact with $\overline{\mathrm{range}}(x) \subseteq \mathscr{U}$. Then for some constant $r \in V(\mathscr{U})$, x approaches the largest weakly invariant set in $V^{-1}(r) \cap \mathscr{U} \cap \left(\overline{u_A^{-1}(0)} \cup u_B^{-1}(0)\right)$.

A direct application of the above result leads to:

Theorem 3 (Kwok, Martínez [28]). *Let $\mathscr{U} = O$. Given the hybrid system for fixed forward velocity vehicles defined above and with virtual center dynamics* (9)*, any precompact trajectory $x(t, j)$ with* $\mathrm{rge}\, x \in \mathscr{U}$*, will approach the set of points*

$$\mathscr{M} = \{x \in O \mid \|\mathrm{CM}_{V_i} - p_i'\| \leq \overline{\varepsilon}, \forall i \in \{1, \ldots, n\}\}. \tag{10}$$

The proof makes use of $\mathscr{H}_{\text{centr}}(x_1,\dots,x_N) = \mathscr{H}(p_1',\dots,p_N')$ as a locally Lipschitz Lyapunov function. It can be seen that (i) $\dot{\mathscr{H}}_{\text{centr}}(x) \le 0$ for all $x \in A$ and, that (ii) $\mathscr{H}_{\text{centr}}(x^+) = \mathscr{H}_{\text{centr}}(x)$, for all $x \in B$. The proof follows from the analysis that the only possible set that can contain the largest invariant set is $\mathscr{M}$.

We present a simulation case below where vehicles have a fixed forward velocity in Figure 5. All vehicles begin with random positions and orientations in the lower left corner. They start with a left virtual center, but agent 5 switches to a right virtual center early in the simulation. It can be seen how the vehicles navigate their virtual centers to the centroids of their Voronoi cells. The plot in the right shows a plot of the cost function minimization to a critical value.

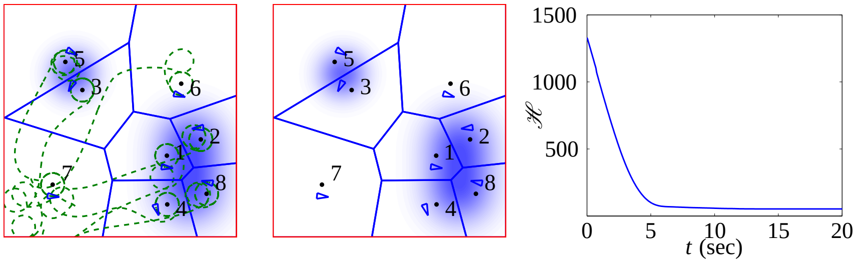

Fig. 5 Fixed forward velocity deployment simulation. The agents start in the lower left corner and path lines are shown in the left figure with final positions and orientations shown in the right figure. Virtual center locations are denoted by a dot.

5 Uncontrollable vehicles in river environments

It is generally assumed that vehicles have fully actuated, or at least controllable, dynamics. However, potential applications may involve the deployment of vehicles in hazardous environments where agents lack the actuation to counteract external forces. Example applications include the deployment of micro-UAVs in wind or gliders in a swift current. One can still aim to factor such significant environmental dynamics into the cooperative control algorithms.

In this section we summarize the results found in [29, 30] for the deployment of vehicles in fast flow environments. Assume the following kinematic model for each of the agents:

$$\dot{p}_i = u_i + V(p_i)\,, \tag{11}$$

where $u_i(t)$ is piecewise smooth, $\|u_i\| \le 1$, and $\|V\| > 1$. Time-optimal trajectories can only be obtained with maximum velocity, thus $u_i = (\cos\theta_i, \sin\theta_i)^T$, see [31].

Our notion of coverage will be associated with the set of points that an agent can travel to faster than other agents. First, let us recall the definition of reachable set:

Definition 1 (Reachable set). We define the *reachable set*, $\mathscr{R}(p_i)$, of an agent at position p_i to be the set of points $x \in X$ that an agent can reach in finite time starting from the initial position p_i and using a piecewise smooth control input $u_i(t)$ with $\|u_i\| \leq 1$. The *T-limited reachable set*, $\mathscr{R}_T(p_i)$, of an agent at position p_i, is the set of points that an agent can reach within time T using a piecewise smooth control input $u_i(t)$ with $\|u_i\| \leq 1$.

Figure 6 shows two examples of reachable sets in an affine and constant flows.

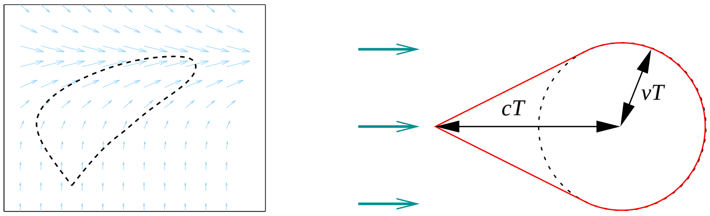

Fig. 6 Reachable sets in affine environments. The figure on the right corresponds to the T reachable set of a vehicle moving with velocity v in a constant flow of magnitude $V = (c, 0)$.

A distributed algorithm for the deployment of agents in a flow environment can be now based on the maximization of the following area coverage metric:

$$\mathscr{H}_{\text{area}}(p_1, \dots, p_n) = \int_{\bigcup \mathscr{R}_T(p_i)} 1 dx \,. \tag{12}$$

This must be done while taking into account the flow environment and how it affects the dynamics of each agent. The consideration of other metrics is still possible, however, external drifts give rise to generalized Voronoi regions with complex boundaries; see [32, 33] for some initial work in this regard. In order to maximize $\mathscr{H}_{\text{area}}$, one can follow the next steps: (a) determine minimum time trajectories in the flow environment, and (b) use knowledge of the properties of these optimal trajectories to compute a gradient direction.

In order to find $\mathscr{R}_T(p_i)$, one must solve the following optimal control problem:

$$\begin{aligned} \text{minimize:} \quad & J = \int_0^{t_f} 1 dt \,, \\ \text{subject to:} \quad & \dot{p}_i = u_i + V(p_i) \,, \|u_i\| \leq 1 \,, \\ & p_i(0) \, \text{and} \, p_i(t_f) \, \text{given} \,. \end{aligned} \tag{13}$$

For a smooth flow field V, this is known as Zermelo's problem, and a solution can be found in [31]. The optimal solution is to consider a control input of the form

$$\begin{aligned} & u_i = (\cos\theta_i, \sin\theta_i) \,, \\ & \dot{\theta}_i = \sin^2\theta_i \frac{\partial V_2}{\partial x_1} + \sin\theta_i \cos\theta_i \left(\frac{\partial V_1}{\partial x_1} - \frac{\partial V_2}{\partial x_2} \right) - \cos^2\theta_i \frac{\partial V_1}{\partial x_2} \,. \end{aligned} \tag{14}$$

The minimum-time trajectories are obtained by using this input in combination with (11). Note that a constant V produces straight-line optimal trajectories.

To obtain the T-limited boundary of $\mathscr{R}_T(p_i)$, one could integrate (11) using (14) to time T starting at the agent location p_i and initial heading $\theta_i(0) \in [\alpha - \beta - \frac{\pi}{2}, \alpha + \beta + \frac{\pi}{2}]$, where $\alpha = \arctan(V_2(x), V_1(x))$ and $\beta = \arcsin\left(\frac{1}{\|V(x)\|}\right)$. The solutions for various initial headings at time T could then be recorded and combined with the solutions $\gamma(t,-1)$ and $\gamma(t,1)$ for $t \in [0,T]$. Note that this procedure works well for affine flows, for which optimal trajectories are well behaved. That is, the trajectories do not intersect and they fill up the cone between the extreme optimal trajectories $\gamma(t,-1)$ and $\gamma(t,1)$.

The consideration of piecewise constant flows changes the nature of optimal solutions and reachable sets. We summarize some of their properties under the following assumption.

Assumption 1. The flow environment X may have obstacles and:

1. The flow V is piecewise constant. That is, $X = \bigcup_{k=1}^{m} X_k$ such that $V_{|X_k}$ is constant and satisfies $\|V_{|X_k}\| > 1$ for all k.
2. The regions X_k, $k \in \{0,\dots,m\}$, are separated by piecewise differentiable curves. Let $\psi_{k,\ell} : X_k \to \mathbb{R}$ be piecewise differentiable common boundary of X_k and X_ℓ.
3. Along the interface between two flows k and ℓ, we consider any $V(x) \in \mathrm{co}\{V_{|X_k}, V_{|X_\ell}\}$ for $\{x \mid \psi_{k,\ell}(x) = 0\}$.

Thus, the optimal paths in the interior of each X_k will be straight lines. As a path reaches X_k, several situations may arise. We briefly describe these in the following.

Catalog of optimal trajectories. For simplicity, in this chapter we assume the boundary of the environment X to be parallel to the flow in the inner region X, and there will not be obstacles present.

For the case that a trajectory intersects a boundary between two flows, defined by $\psi_{k,\ell}(x) = 0$, the intersection can again occur either transversely or tangentially. Based on this, we classify trajectories into *simple (transversal)* or *non-simple (tangential) trajectories*; see Figure 7. The transversal simple trajectories are non-pathological and undergo a direction change at the interface, following an analogous rule to that of the Snell's law in physics:

Proposition 1 (Kwok, Martínez [34]). *Let $V_- = (c_1,c_2)^T$ and $V_+ = (d_1,d_2)^T$ be the flows in two neighboring regions, and α_1, α_2 be their respective flow orientations. Let ξ be the orientation of the normal vector of the smooth curve $\psi(x) = 0$ at the point where the optimal trajectory crosses into the second flow region. A necessary condition for an optimal trajectory across the interface of the two flow regions requires that:*

$$\frac{1 + \|V_-\| \cos(\theta_- - \alpha_1)}{\sin(\theta_- - \xi)} = \frac{1 + \|V_+\| \cos(\theta_+ - \alpha_2)}{\sin(\theta_+ - \xi)}. \tag{15}$$

Given (15)*, and a fixed heading θ_-, the final heading satisfies*

$$\sin\theta_+ = \frac{B \pm C\sqrt{B^2+C^2-1}}{B^2+C^2}\,, \tag{16}$$

where $B = \frac{1+\|V_-\|\cos(\theta_- - \alpha_1)}{\sin(\theta_- - \xi)}\cos\xi - d_2$ *and* $C = \frac{1+\|V_-\|\cos(\theta_- - \alpha_1)}{\sin(\theta_- - \xi)}\sin\xi + d_1$.

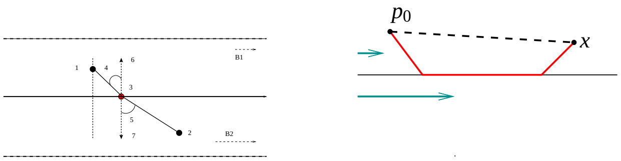

Fig. 7 Example of simple trajectories (left figure) and of non-simple trajectories (right figure.)

However, the application of (15) can also result in a trajectory (non-simple trajectory) that travels along the boundary between two different flows; see Figure 7, right picture. The use of the same result can also give a way to compute a heading back into the first region. When an agent is moving along a flow boundary, and it is possible to switch back into the first region, the agent may choose to switch back at any time, making this process indeterminate. However, the result above dictates that there is only one possible outgoing heading back into the first flow region. Furthermore, it is possible for these trajectories that flow along boundaries and later return to intersect other trajectories that remained in the original flow region. For completeness, the following result summarizes necessary conditions for the incoming and outgoing angles for these cases.

Proposition 2 (Kwok, Martínez [34]). *Assume two flow regions defined by the parameters* $\|V_-\|, \alpha_1$ *and* $\|V_+\|, \alpha_2$*, respectively, separated by an interface whose normal angle is* ξ*. If it is possible for an agent to flow along the boundary under the second flow, then* θ_+ *satisfies*

$$\theta_+ \in \Big\{\xi \pm \arccos\left[-\|V_+\|\sin(\alpha_2+\xi)\right], -\xi \pm \arccos\left[\|V_+\|\sin(\alpha_2+\xi)\right]\Big\}. \tag{17}$$

Let $D = \frac{1+\|V_+\|\cos(\theta_+ - \alpha_2)}{\sin(\theta_+ - \xi)}$*. Then, the incoming heading resulting in flow along the boundary, if it exists, satisfies*

$$\begin{aligned}\theta_- = {} & \arctan\left[\frac{\|V_-\|\sin\alpha_1 - D\cos\xi}{\|V_-\|\cos\alpha_1 + D\sin\xi}\right] \\ & \pm \arccos\left(\frac{-1}{\sqrt{(\|V_-\|\sin\alpha_1 - D\cos\xi)^2 + (D\sin\xi + \|V_-\|\cos\alpha_1)^2}}\right).\end{aligned} \tag{18}$$

Knowledge about these trajectories can be used to derive a gradient-ascent algorithm that aims to maximize $\mathscr{H}_{\text{area}}$. We begin by taking the gradient of $\mathscr{H}_{\text{area}}$ with respect to p_i in order to obtain a set directions each agent must travel in.

Proposition 3 (Kwok, Martínez [34]). *Given the area objective* (12), *let*

$$\mathring{A}_i = \partial \mathscr{R}_T(p_i) \cap \left(\bigcup_{j \in \mathscr{N}_{i,\text{flow}}} \mathscr{R}_T(p_j) \right)^c \cap X \,, \tag{19}$$

the set of points in $\partial \mathscr{R}_T(p_i)$ is not in the interior of neighboring reachable sets. Then the gradient with respect to p_i is:

$$\frac{\partial \mathscr{H}}{\partial p_i} = \int_{\mathring{A}_i} \hat{\mathbf{n}}_{\text{out}}^T(\zeta_i) \frac{\partial \zeta_i}{\partial p_i} d\zeta_i \,, \tag{20}$$

where $\zeta_i : \mathbb{S} \to \mathbb{R}^2$ is a parametrization of $\partial \mathscr{R}_T(p_i)$, and $\hat{\mathbf{n}}_{\text{out}} : \mathbb{R}^2 \to \mathbb{R}^2$ is the unit outward-pointing normal vector at ζ_i.

For piecewise constant flows, $\frac{\partial \zeta_i}{\partial p_i}$ can analytically be computed using the previous analysis of the course changes of optimal trajectories in flows; see [34]. We can further analyze the algorithm

$$\dot{p}_i = \frac{\partial \mathscr{H}_{\text{area}}}{\partial p_i}, \qquad i \in \{1, \dots, n\} \,, \tag{21}$$

above for the special case of a single constant flow field. For a constant flow the gradient according to (20) becomes $\frac{\partial \mathscr{H}_{\text{area}}}{\partial p_i} = \int_{\mathring{A}_i} \hat{\mathbf{n}}_{\text{out}}^T(\zeta_i) d\zeta_i$. This result has an intuitive interpretation. In order to maximize area covered, agents move towards locations that are not occupied by other agents' reachable sets. Now the following can be proven:

Proposition 4 (Kwok, Martínez [34]).

1. *For a constant flow field, $V = c$, if no regions intersect the boundaries ∂X, then $\mathscr{H}_{\text{area}}$ is non-decreasing if agents use the control law* (21).
2. *For constant flows, if the flow boundaries are parallel with the flow direction and X is unbounded (the flow domain is an infinitely long strip), then $\mathscr{H}_{\text{area}}$ is maximized by* (21).

Other flow cases make difficult the analysis of the evolution of $\mathscr{H}_{\text{area}}$, similarly to what happens with time-dependent coverage functions. The current strategy makes agents follow the direction of maximum ascent of $\mathscr{H}_{\text{area}}$. However, one can imagine situations in which the value of $\mathscr{H}_{\text{area}}$ decreases despite of this if, for example, boundaries of X become closer and closer. The following is a simulation showing how $\mathscr{H}_{\text{area}}$ oscillates around a given value when the flow regions force vehicles into a bounded region; see Figure 8.

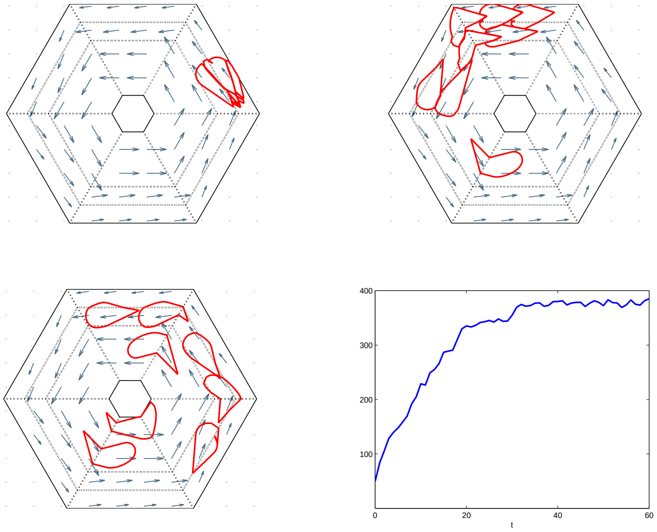

Fig. 8 The central “eye” of the storm is treated as an obstacle, or equivalently a “no-fly zone.” The simulation snapshots occur for $t = 0$ (top left), $t = 20$ (top right), and $t = 60$ (bottom left). A plot of the total reachable area is shown in the bottom right.

6 Conclusions

This chapter summarized several results concerning the deployment of vehicles subject to dynamic constraints. In general, the algorithms are distributed over the associated Delaunay graphs or, in r -disk graphs with r sufficiently large. The algorithms provide convergence guarantees to the set of local minima of different classes of Locational Optimization or multicenter metrics. Dynamic constraints were dealt with in essentially three ways: (i) in a soft manner, by modifying the Locational Optimization metric and working with easy-to-compute generalized Voronoi regions, and (ii) by resorting to controllability properties of the vehicles, and (iii) by using the dynamic constraints in the definition of generalized regions assigned to each vehicle. In general, dynamic constraints will lead to involved generalized Voronoi regions, whose boundary is hard to compute and represent, as it reduces to the solution of an optimal control problem. We are currently investigating how this can be alleviated by considering upper and lower approximations of Voronoi regions, which can be refined to any degree at a higher computational expense. By defining an algorithm that allows each agent follow the direction of an approximated gradient using the lower Voronoi region approximation, it can be seen how local minima can still be reached. We are exploring this in the context of constant river environments in the manuscript [35] with Voronoi regions given by hyperbolas, but we believe the approach can be extended to general cost functions.

7 Acknowledgments

This work surveys partially the thesis work of the graduate student Andrew N. Kwok. Andrew defended his thesis in January 2011 and was funded by grants NSF CNS-0930946, NSF CAREER CMMI-0643679.

References

1. Payton, D., Daily, M., Estowski, R., Howard, M., Lee, C.: Pheromone robotics 11, 319–324 (2001)
2. Howard, A., Matarić, M.J., Sukhatme, G.S.: Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem. In International Conference on Distributed Autonomous Robotic Systems (DARS02), pp. 299–308, Fukuoka, Japan (2002)
3. Poduri, S., Sukhatme, G.S.: Constrained coverage for mobile sensor networks. , pp. 165–172, New Orleans, LA (2004)
4. Choset, H.: Coverage for robotics - a survey of recent results. Annals of Mathematics and Artificial Intelligence 31, 113–126 (2001)
5. Arkin, R.C.: Behavior-Based Robotics. (1998)
6. Schultz, A.C., Parker, L.E. (eds.): Multi-Robot Systems: From Swarms to Intelligent Automata. (2002). Proceedings from the 2002 NRL Workshop on Multi-Robot Systems
7. Balch, T., Parker, L.E. (eds.): Robot Teams: From Diversity to Polymorphism. A K Peters Ltd., Natick, MA (2002)
8. Bulusu, N., Heidemann, J., Estrin, D.: Adaptive beacon placement. In Int. Conference on Distributed Computing Systems, pp. 489–498, Mesa, AZ (2001)
9. Cortés, J., Martínez, S., Karatas, T., Bullo, F.: Coverage control for mobile sensing networks. IEEE Trans. Rob. 20, 243–255 (2004)
10. Cortés, J., Martínez, S., Bullo, F.: Spatially-distributed coverage optimization and control with limited-range interactions. ESAIM: Control Optim. Calc. Var. 11, 691–719 (2005)
11. Zhong, M., Cassandras, C.G.: Distributed coverage control in sensor network environments with polygonal obstacles. , pp. 4162–4167, Seoul, Korea (2008)
12. Pimenta, L.C.A., Kumar, V., Mesquita, R.C., Pereira, G.A.S.: Sensing and coverage for a network of heterogeneous robots. , pp. 3947–3952, (2008)
13. Caicedo-Nùñez, C.H., Žefran, M.: Performing coverage on nonconvex domains. , pp. 1019–1024, San Antonio, TX (2008)
14. Schwager, M., Rus, D., Slotine, J.J.: Decentralized, adaptive coverage control for networked robots. Int. J. Rob. Res. 28, 357–375 (2009)
15. Martínez, S.: Distributed interpolation schemes for field estimation by mobile sensor networks. IEEE Trans. Control Syst. Technol. (2009). To appear
16. Cortés, J.: Area-constrained coverage optimization by robotic sensor networks. , pp. 1018–1023, (2008)
17. Pavone, M., Frazzoli, E., Bullo, F.: Distributed policies for equitable partitioning: Theory and applications. , pp. 4191–4197, (2008)
18. Bullo, F., Cortés, J., Martínez, S.: Distributed Control of Robotic Networks. (2009). Available at http://www.coordinationbook.info
19. Okabe, A., Boots, B., Sugihara, K., Chiu, S.N.: Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. (2000)
20. Raghunathan, V., Pereira, C., Srivastava, M., Gupta, R.: Energy-aware wireless systems with adaptive power-fidelity tradeoffs. IEEE Trans. Very Large Scale Integration Systems 13, 211–225 (2005)

21. Mohapatra, S., Dutt, N., Nicolau, A., Venkatasubramanian, N.: DYNAMO: A cross-layer framework for end-to-end QoS and Energy Optimization in Mobile Handheld Devices. IEEE J. Selected Areas in Communication (2007)
22. Kwok, A., Martínez, S.: Deployment algorithms for a power-constrained mobile sensor network (2009). To appear
23. Kwok, A., Martínez, S.: Deployment algorithms for a power-constrained mobile sensor network. International Journal of Robust and Nonlinear Control 20, 725–842 (2010)
24. Sanfelice, R.G., Goebel, R., Teel, A.R.: Results on convergence in hybrid systems via detectability and an invariance principle. In American Control Conference, pp. 551–556, (2005)
25. Dubins, L.E.: On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. American Journal of Mathematics 79, 497–516 (1957)
26. Goebel, R., Hespanha, J.P., Teel, A.R., Cai, C., Sanfelice, R.G.: Hybrid systems: generalized solutions and robust stability. , pp. 1–12, Stuttgart, Germany (2004)
27. Goebel, R., Sanfelice, R., Teel, A.: Hybrid dynamical systems. Control Systems Magazine, IEEE 29, 28–93 (2009)
28. Kwok, A., Martínez, S.: Unicycle coverage control via hybrid modeling. IEEE Transactions on Automatic Control 55, 528–532 (2010)
29. Kwok, A., Martínez, S.: A coverage algorithm for drifters in a river environment. In 2010 American Control Conference, pp. 6436–6441, Baltimore, MD, USA (2010)
30. Kwok, A., Martínez, S.: Deployment of drifters in a piecewise-constant flow environment. In IEEE International Conference on Decision and Control, pp. 6584–6589, Atlanta, GA, USA (2010)
31. Bryson, A.E., Ho, Y.: Applied Optimal Control. Hemisphere Publishing Corporation, New York (1969)
32. Bakolas, E., Tsiotras, P.: The Zermelo–Voronoi diagram: A dynamic partition problem. Automatica 46, 2059–2067 (2010)
33. Ru, Y., Martínez, S.: Energy-based Voronoi partition in constant-flow environments. IEEE Transactions on Automation Science and Engineering (2011). Submitted
34. Kwok, A., Martínez, S.: A coverage algorithm for drifters in a river environment. In American Control Conference, , Baltimore, MD (2010). To appear
35. Ru, Y., Martínez, S.: Coverage control in constant flow environments using a mixed energy-time metric. In IEEE International Conference on Decision and Control, , (2012). Submitted