Distributed optimal transport for the deployment of swarms

Vishaal Krishnan and Sonia Martínez

Abstract— **The analysis and design of scalable distributed algorithms for spatial deployment is an important problem in the area of multi-robot systems. For very large swarms, this can be prescribed via macroscopic objectives on the behavior of the swarm, and accomplished by local sensing and communication between agents. In this paper, we address the problem of distributed optimal transport, with the aim of minimizing the cost of deployment of large swarms. Working with a macroscopic PDE model of swarms given by the continuity equation, we first formulate a general deployment objective, and formulate deployment algorithms as convergent gradient flows. Then, we design and analyze a novel Laplacian-based distributed algorithm and a corresponding weighted gradient flow for optimal transport. We conclude the manuscript with simulations that illustrate our results.**

I. Introduction

The development of low-cost sensor, communication and computational systems makes foreseeable in the near future the deployment of large teams of multi-robot systems in diverse areas such as remote monitoring, manufacturing, and construction. As the number of robotic agents increases, the design of efficient control algorithms for these poses new challenges, starting with the choice of appropriate mathematical abstractions for them. The need for parsimonious descriptions of swarms, together with the fact that tasks for these systems are more likely to be specified at a high level, calls for the use of macroscopic models. This is more so in scenarios that involve interaction with human operators, wherein the complexity of the description has to be kept minimal. However, such a setting introduces new theoretical challenges in the analysis and control of these systems, and we place this work in this broader context.

Literature review: The problems of deployment and formation control of groups of robots have been extensively studied in the past [1], [2]. The approaches range from Voronoi-based deployment [3] to deployment using potential fields [4], among other methods. An important characteristic of these works is the use of discrete models, where the system is seen as the finite collection of N robots, and their evolution is modeled by a system of ODEs. Having a limited set of resources makes the specific configuration of each robot important. However, as the size of the group increases, the relative importance of a single robot decreases, and a macroscopic description is sufficient to capture the quality of swarm deployment. This motivates the macroscopic or PDE-based approach adopted in this paper. One notable approach to large swarm deployment problem makes use of Markov chains [5]–[7], where the swarm configuration is described through the partitioning of the spatial domain into a finite number of disjoint subregions, on which a probability distribution is defined. Other works on large-scale swarm deployment have used continuum models for swarms. In [8], [9], the authors address the problem of deployment in the absence of position information. In related subsequent work [10], where agents have access to position information, the authors address the problem of local density estimation and velocity field generation in the context of robot swarms. Another notable approach in this regard involves modeling the swarm collective dynamics by the reaction-advection-diffusion PDE [11]–[13], wherein the desired coverage profiles are obtained by tuning the parameters in the model and/or by boundary control.

Optimal transport theory [14] deals with the problem of optimally rearranging a given probability distribution onto another while minimizing the cost of transport. The solutions to this problem are referred to as optimal transport plans, and specified as a joint distribution with marginals equal to the original distributions. Although the original problem has a static description, it allows for a time/path-dependent formulation [15], which also takes the form of a stochastic optimal control problem [16]. Optimal transport theory provides a natural framework for the problem of minimizing the cost of deployment of swarms. However, despite the availability of powerful mathematical tools in this area, its adaptation to the control of robot swarms has remained limited. The papers [17] and [18] approach this very question of optimal swarm motion. While in the first paper the problem is formulated as one of optimal control, the second is placed in the framework of optimal transport. However, the previous works present certain limitations, either because they require a centralized offline planning phase [17], or because of costly information exchange required among agents [18].

Contributions: This paper contributes to the body of work on large-scale swarm deployment in the following ways. First, we begin by presenting a PDE-based swarm model while providing a formal justification for it. Then, we encode the general deployment objective via suitable performance metrics, which leads to the design of deployment algorithms via gradient flows. We point out how these flows are naturally distributed for some standard metrics over the space of probability measures, and prove formally their convergence to the target distributions of interest. This is followed by the main contribution of this paper, which is a scalable Laplacian-based distributed algorithm that solves the Monge-Kantorovich optimal transport problem for a wide class

This work has been partially supported by grant FA9550-18-1-0158.

The authors are with the Department of Mechanical and Aerospace Engineering, University of California at San Diego, La Jolla CA 92093 USA (email: v6krishn@ucsd.edu; soniamd@ucsd.edu).

of transport cost functions. This algorithm, in conjunction with a weighted gradient flow is shown to achieve swarm deployment while minimizing the transport cost.

II. Notation and Preliminaries

In what follows, we let $\nabla = \left(\frac{\partial}{\partial x_1}, \dots \frac{\partial}{\partial x_n}\right)$ denote the gradient operator in $\mathbb{R}^n$ when acting on real-valued functions. As a shorthand, we let $\frac{\partial}{\partial z}(\cdot) = \partial_z(\cdot)$ for a variable z. For any $x \in \Omega \subset \mathbb{R}^d$, we denote by $B_r(x)$ an open d-ball of radius $r > 0$. Let $\mu \in \mathscr{P}(\Omega)$ be an absolutely continuous probability measure on $\Omega \subset \mathbb{R}^d$. We denote by ρ the corresponding density function, where $\mathrm{d}\mu = \rho\,\mathrm{dvol}$, with vol being the Lebesgue measure. Let $F : \mathscr{P}(\Omega) \to \mathbb{R}$ be a functional such that $F(\mu) = \int_\Omega g(x) \ \mathrm{d}\mu(x)$, we denote by $\frac{\delta F}{\delta \mu}(x) = g(x)$ the derivative of F with respect to μ, in the distributional sense [19]. The set of functions on a measurable space U, given by $L^p(U) = \{f : U \to \mathbb{R} \,|\, \|f\|_{L^p(U)} = (\int_U |f|^p\,\mathrm{dvol})^{1/p} < \infty\}$, constitute the L^p space, where $\|\cdot\|_{L^p(U)}$ is the L^p norm. Of particular interest is the L^2 space, or the space of square-integrable functions. In this paper, we denote by $\|f\|_{L^2(U)}$ the L^2 norm of f with respect to the Lebesgue measure, and by $\|f\|_{L^2(U,\mu)} = \left(\int_\Omega |f|^2\,\mathrm{d}\mu\right)^{1/2}$ the weighted L^2 norm. We denote by $\langle f, g\rangle$ the inner product of functions $f, g : \Omega \to \mathbb{R}$, given by $\langle f, g\rangle = \int_\Omega fg\,\mathrm{dvol}$. The Sobolev space $W^{1,p}(U)$ over a measurable space U is defined as $W^{1,p}(U) = \{f : U \to \mathbb{R} \,|\, \|f\|_{W^{1,p}} = (\int_U |f|^p + \int_U |\nabla f|^p)^{1/p} < \infty\}$. Of particular interest is the space $W^{1,2}$, also called the H^1 space.

III. Swarm model and deployment objective

In this section, we present a macroscopic model for large swarms, starting with the dynamics of a single agent and making a continuum approximation to the associated PDE dynamics. We then specify the macroscopic deployment objective, for which we present novel distributed algorithms in following sections.

Let the configuration of the swarm at any given instant of time t be denoted by the tuple $(\mathscr{I}, \{x_i(t)\}_{i\in\mathscr{I}}, \{\mathbf{v}_i(t)\}_{i\in\mathscr{I}})$ consisting of agent indices, their positions and velocities (with $|\mathscr{I}| = N$). We assume that agents are distributed across a domain $\Omega \subset \mathbb{R}^d$, where Ω is a bounded and open set. That is, $x_i(t) \in \Omega$ and $\mathbf{v}_i(t) \in \mathbb{R}^d$ for all $i \in \mathscr{I}$ and any $t \in \mathbb{R}_{\geq 0}$. For simplicity, the dynamics for an agent i at $x_i \in \Omega$ is given by $\dot{x}_i = \mathbf{v}_i$. In this work, we consider large swarms, where N is large, and are interested not in the relative positions of individual agents—as they are not considered significant—but in the macroscopic description of the swarm. Therefore, we abstract a multi-agent swarm at any instant t by means of a swarm distribution $\mu_t \in \mathscr{P}(\Omega)$, where $\mathscr{P}(\Omega)$ is the space of absolutely continuous probability distributions on Ω. In other words, we consider the distribution to be normalized by the number of agents, that is $\int_\Omega \mathrm{d}\mu = 1$, so that μ can be interpreted as a probability distribution.

We assume $\mathbf{v}_i = \mathbf{v}(x_i)$, where $\mathbf{v}$ is a velocity field over Ω. Let Φ be the flow associated with this field $\mathbf{v}$, such that $\frac{\mathrm{d}}{\mathrm{d}t}\Phi_t(x) = \mathbf{v}(t, \Phi_t(x))$ and $\Phi_0(x) = x$, for $x \in \Omega$. The flow Φ describes the trajectories of the agents in the swarm, in the sense that $\Phi_t(x)$ represents the position at time t of the agent starting from x at $t = 0$. The evolution of the swarm distribution $\mu_t \in \mathscr{P}(\Omega)$ in the limit $N \to \infty$, subject to the flow Φ, is given by the continuity equation:

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{1}$$

where $\mathrm{d}\mu_t = \rho_t$ dvol, and vol is the Lebesgue measure. We will also require that $\rho\mathbf{v}\cdot\mathbf{n} = 0$ on the boundary $\partial\Omega$, in order to confine the swarm in the domain Ω.

Our objective is to design a class of distributed algorithms in the continuous domain for swarm deployment, which transform the initial distribution μ_0 of the swarm into a desired distribution μ^*, as $t \to \infty$, subject to the dynamics (1). As usually done in the discrete formulation, we aim to design these distributed algorithms by means of gradient flows minimizing an appropriate potential function.

IV. Gradient flow as a deployment mechanism

In this section, we present deployment algorithms as gradient flows of over the space of probability distributions. We then show formally how these algorithms converge to the desired swarm distribution. We take as examples the L^2 distance and KL-divergence which lead to a distributed gradient flow, and helps set the stage for next sections.

Given a desired distribution $\mu^* \in \mathscr{P}(\Omega)$, and an arbitrary other one $\mu \in \mathscr{P}(\Omega)$ the following potential functions serve to quantify how close μ is to μ^*:

$$V_{L^2}(\mu) = \frac{1}{2}\int_\Omega |\rho - \rho^*|^2 \ \mathrm{dvol},$$
$$V_{\mathrm{KL}}(\mu) = \int_\Omega \rho \log\left(\frac{\rho}{\rho^*}\right) \ \mathrm{dvol}.$$

where recall that ρ and ρ^*, are such that $\rho\,\mathrm{dvol} = \mathrm{d}\mu$, respectively. The previous functionals are convex with a minimizer at μ^*, and the corresponding expressions for their gradient flows are the following:

$$\mathbf{v}_{L^2}(x) = -\nabla(\rho - \rho^*), \qquad \mathbf{v}_{\mathrm{KL}}(x) = -\nabla\left(\frac{\rho}{\rho^*}\right), \quad x \in \Omega.$$

These expressions (or a discretization of them) only require local communication of each agent at x with others at y such that $|x - y| \leq r$, for $r > 0$. Therefore, they result in a naturally distributed computation. In addition, agents need knowledge of the target swarm distribution μ^* as well as access to their position information.

The following theorem establishes a fundamental convergence result for gradient flows on convex potentials. We do not present proofs of the results in this paper, for the sake of brevity. We refer the reader to an extended version [20] for detailed proofs, which will appear in a forthcoming publication.

Assumption 4.1: ***(Well-posedness of solutions).*** We assume that the desired distribution μ^* is absolutely continuous (with density function ρ^* in $H^1(\Omega)$) and supported in a compact subset of Ω. Further, we assume that the solutions ρ_t

to (1) for the gradient flow are well-posed and lie the the Sobolev space $H^1(\Omega)$. •

Theorem 4.2: ***(Convergence of gradient flow).*** Let $V : \mathscr{P}(\Omega) \to \mathbb{R}$ be a convex potential functional with minimizer at μ^* satisfying Assumption 4.1. Define the gradient flow $\mathbf{v} = -\nabla\left(\frac{\delta V}{\delta \mu}\right)$ and consider the swarm dynamics (1) together with the condition $\rho \mathbf{v} \cdot \mathbf{n} = 0$ on the boundary $\partial\Omega$. Then, μ_t converge to μ^* as $t \to \infty$ in the L^2 sense, i.e., the solutions ρ_t to (1) satisfying Assumption 4.1 on well-posedness converge in the L^2-norm to ρ^*. •

V. Distributed Optimal Transport

In this section, we develop a distributed algorithm for optimal transport, which is implemented by the swarm as a gradient flow. We begin by presenting the standard formulation of the Monge-Kantorovich optimal transport problem, which we then reduce to a form that allows us to design our distributed algorithm. For more information about optimal transport we refer the reader to [14].

The cost of optimal transport from a probability measure μ on Ω to another measure ν on Ω, is given by:

$$C(\mu,\nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{\Omega\times\Omega} c(x,y)\, \mathrm{d}\pi(x,y), \tag{2}$$

where $c(x,y)$ is the unit cost of transport between x and y. Here, the cost $c(x,y)$ is a non-negative, lower semicontinuous function. The subset $\Pi(\mu,\nu) \subset \mathscr{P}(\Omega\times\Omega)$ consists of the joint measures with marginals μ and ν, and $\mathrm{d}\pi(x,y)$ is the amount of mass transported from x to y. This is commonly referred to as the Monge-Kantorovich optimal transport problem. We state the following lemma on the convexity of the cost functional C.

Lemma 5.1: ***(Convexity of*** $C(\mu,\nu)$***, Theorem 4.8 [14]).*** The optimal transport cost $C(\mu,\nu)$ in (2) is convex in μ and ν. •

The primal optimization problem (2) is a linear programming problem (note that the minimization is in π and the objective function is linear in π). The dual problem, called Kantorovich duality, is given by:

$$\begin{aligned} K(\mu,\nu) = \sup_{\phi\in L^1_\mu(\Omega);\psi\in L^1_\nu(\Omega)} & \int_\Omega \phi(x)d\mu(x) + \int_\Omega \psi(y)d\nu(y) \\ \text{s.t} \quad & \phi(x)+\psi(y) \le c(x,y), \quad \forall x,y\in\Omega. \end{aligned}$$

and the duality gap is zero, $C(\mu,\nu) = K(\mu,\nu)$ (Theorem 5.10, [14]). The maximizers of this problem are pairs of functions (ϕ,ψ) that occur at the boundary of the inequality constraint and are such that:

$$\begin{aligned} \phi(x) &= \inf_{y\in\Omega}\left(c(x,y) - \psi(y)\right), \\ \psi(y) &= \inf_{z\in\Omega}\left(c(z,y) - \phi(z)\right). \end{aligned} \tag{3}$$

In other words, a solution pair consists of conjugate functions, which by definition satisfy the previous equations. We write $\psi = \phi^c$, to denote that ψ is conjugate of ϕ, and we have:

$$\phi(x) = \inf_{y\in\Omega}\left[c(x,y) - \inf_{z\in\Omega}\left(c(y,z)-\phi(z)\right)\right]. \tag{4}$$

The dual problem can be now reduced to:

$$K(\mu,\nu) = \sup_{\phi\in L^1_\mu(\Omega)} \int_\Omega \phi(x)d\mu(x) + \int_\Omega \phi^c(y)d\nu(y). \tag{5}$$

A. Gradient Flow based on Kantorovich duality

In this section, we first reformulate the dual optimal transport problem for metric costs $c(x,y)$. We then exploit this formulation to design a distributed algorithm that achieves optimal transport. We essentially assume that the cost of transport is non-negative, symmetric and that the cost of transport between any two points, while stopping by multiple waypoints is at least as much as the cost of direct transport. The following lemma will allow us to reduce the Kantorovich dual problem for metric costs.

Lemma 5.2: ***(Metric costs).*** If the cost c is a metric and ϕ satisfies (4), then $|\phi(x)-\phi(y)| \le c(x,y)$ for all $x,y\in\Omega$. Also, $\phi^c = -\phi$ and $|\nabla\phi| \le |\nabla c|$, where $\nabla c|_x = \nabla_y c(x,y)\big|_{y=x}$. •

We can now rewrite the Kantorovich dual problem (5) in the following form:

$$\begin{aligned} K(\mu,\nu) = \sup_\phi & \int_\Omega \phi(\rho-\rho^*)\,\mathrm{dvol} \\ \text{s.t.} \quad & |\nabla\phi| \le |\nabla c|. \end{aligned} \tag{6}$$

Clearly, the objective function in the above problem is linear in ϕ and the feasible set corresponding to the constraint $|\nabla\phi| \le |\nabla c|$ is convex.

The Lagrangian corresponding to the Kantorovich dual problem (6) is given by:

$$\mathscr{L}(\phi,\lambda) = \int_\Omega \phi(\rho-\rho^*) - \frac{1}{2}\int_\Omega \lambda(|\nabla\phi|^2 - |\nabla c|^2), \tag{7}$$

where all the integrals are with respect to the Lebesgue measure, and $\lambda \ge 0$ is the Lagrange multiplier for the constraint, which we have rewritten as $|\nabla\phi|^2 \le |\nabla c|^2$.

Here, we obtain a crucial insight that the stationary condition for the above problem is a Poisson equation with a weighted Laplace operator (and a Neumann boundary condition), as stated in the following lemma. This will eventually allow us to implement a Laplacian-based distributed algorithm for optimal transport.

Lemma 5.3: ***(Optimality conditions).*** The necessary and sufficient conditions for a feasible solution $\bar\phi$ of (6) to be optimal are:

$$\begin{aligned} -\nabla\cdot\left(\bar\lambda\nabla\bar\phi\right) &= \rho-\rho^*, \\ \bar\lambda\nabla\bar\phi\cdot\mathbf{n} &= 0, \quad \text{on } \partial\Omega, \\ \bar\lambda \ge 0, \quad |\nabla\bar\phi| \le |\nabla c|, & \quad \bar\lambda(|\nabla\bar\phi| - |\nabla c|) = 0, \end{aligned} \tag{8}$$

where $\bar\lambda$ is the optimal Lagrange multiplier λ for the constraint $|\nabla\phi| \le |\nabla c|$, and they correspond to the saddle point of the Lagrangian (7). •

We now define the primal-dual dynamics to converge to the saddle point of the Lagrangian (7), where we use second-order dynamics for the dual variable. The primal-dual dynamics, owing to its structure (involving the Laplacian) is suitable for a distributed implementation.

To obtain the second order primal-dual dynamics, we consider an augmented Lagrangian:

$$\mathscr{L}_a(\phi,\lambda,\partial_t\lambda) = \mathscr{L}(\phi,\lambda) + \frac{1}{2}\int_\Omega |\partial_t\lambda|^2,$$

where $\mathscr{L}$ is as defined in (7) and the added penalty term is for the time derivative of λ which is introduced as an additional variable. With $\gamma = (\lambda, \partial_t\lambda)$ as the new dual variable, we construct the dynamics for a gradient ascent on $\mathscr{L}_a$ w.r.t ϕ (the primal variable) and a weighted gradient descent on $\mathscr{L}_a$ w.r.t γ (the new dual variable), given by:

$$\begin{aligned}
\partial_t\phi &= \nabla\cdot(\lambda\nabla\phi) + (\rho-\rho^*),\\
\lambda\nabla\phi\cdot\mathbf{n} &= 0, \quad \text{on } \partial\Omega,\\
\partial_t\lambda &= [\theta]^+_\lambda,\\
\partial_t\theta &= -\beta(t,x)\theta + \frac{1}{2}\left(|\nabla\phi|^2 - |\nabla c|^2\right),
\end{aligned} \tag{9}$$

where $\beta(t,x)$ is a tunable parameter controlling the rate of convergence for the second-order dual dynamics, and

$$[f]^+_\lambda = \begin{cases} f, & \text{if } \lambda > 0,\\ \max\{0, f\}, & \text{if } \lambda = 0,\end{cases} \tag{10}$$

is the projection operator. We have used the projection operator to explicitly impose the constraint $\lambda \geq 0$ for all $t \in \mathbb{R}_{\geq 0}$ and $x \in \Omega$.

It can be seen that $\partial_t\phi = \frac{\delta\mathscr{L}_a}{\delta\phi}$ and $\partial_t\gamma = A\frac{\delta\mathscr{L}_a}{\delta\gamma}$ with,

$$A = \begin{bmatrix} 0 & 1_\lambda \\ -1 & -\beta \end{bmatrix},$$

where m_λ is an operator such that $m_\lambda f = m[f]^+_\lambda$. We note that for any $\mathbf{y}\in\mathbb{R}^2$, $\mathbf{y}^\top A\mathbf{y} \leq 0$ (since $A_s = \frac{A+A^\top}{2}$ is negative semi-definite). Thus, we have a gradient ascent w.r.t. ϕ and a weighted gradient descent w.r.t. $\gamma = (\lambda, \partial_t\lambda)$.

Assumption 5.4: ***(Well-posedness of solutions).*** We assume that the PDE (9) is well-posed and that the solutions $(\phi_t, \lambda_t, \theta_t)$ lie in the Sobolev space $H^1(\Omega)$ for every time t. •

Lemma 5.5: ***(Convergence of primal-dual dynamics).*** The solutions $(\phi_t, \lambda_t, \theta_t)$ to the primal-dual dynamics (9) satisfying Assumption 5.4, converge in the L^2-norm to the optimality condition (8) as $t\to\infty$, for any given ρ, ρ^*. •

The condition on $\beta(t,x)$ at any $t\in\mathbb{R}_{\geq 0}$ and $x\in\Omega$ which allows for the convergence of the primal-dual based gradient flow is specified in Theorem 5.8 that follows later in this section. However, we state here that the condition on $\beta(t,x)$ is one of a lower bound on its value, which can be computed by the agents using only local information (this will be clear from the statement of Theorem 5.8).

Lemma 5.6: The velocity field:

$$\mathbf{v} = -\nabla\bar{\phi}, \tag{11}$$

where $\bar{\phi}$ is a solution to (6), implemented in the continuity equation (1), provides a gradient flow on $C(\mu,\mu^*)$ w.r.t μ. •

Even though the algorithm (9) results in a distributed implementation of optimal transport, we would like the agents to implement a solution online. In this way, we do not wait for ϕ to converge to $\bar{\phi}$ and then implement the gradient flow velocity $\mathbf{v} = -\nabla\bar{\phi}$, but instead set the velocity control law as $\mathbf{v} = -\frac{\lambda}{\rho}\nabla\phi$, weighted by $\frac{\lambda}{\rho}$, where ϕ and λ are the current local estimates supplied by the distributed algorithm (9) and ρ is the current local density measurement/estimate. We note that since the velocity is for an agent, $\mathbf{v}$ is defined only where $\rho > 0$, and thus $\mathbf{v} = -\frac{\lambda}{\rho}\nabla\phi$ is well-defined.

Assumption 5.7: ***(Well-posedness of solutions).*** We assume that the desired distribution μ^* is absolutely continuous (with density function ρ^* in $H^1(\Omega)$) and supported in a compact subset of Ω. Further, we assume that the solutions $(\phi_t, \lambda_t, \theta_t)$ to (9) and ρ_t to (1) for the gradient flow are well-posed, continuous and lie the the Sobolev space $H^1(\Omega)$ for every $t\in\mathbb{R}_{\geq 0}$. •

Theorem 5.8: ***(Convergence of optimal transport).*** Let the primal-dual dynamics (9) be such that $\beta(t,x)|\theta(t,x)| \geq \max\left\{0, \frac{1}{2}|\nabla\phi|^2 \;\; + \frac{1}{2}\mathrm{sgn}(\theta)\left(|\nabla\phi|^2 - |\nabla c|^2\right)\right\}$. Consider the gradient flow with the velocity $\mathbf{v} = -\frac{\lambda}{\rho}\nabla\phi$, with λ and ϕ from the primal-dual dynamics (9). Then, the solutions of (1) with this velocity field satisfying the Assumption 5.7, converge to μ^* in the L^2 sense. That is, the solutions ρ_t to (1) satisfying Assumption 5.7 converge in the L^2-norm to ρ^*, while the solutions to the primal-dual dynamics (9) converge in the L^2-norm to the optimality condition (8). •

Remark We note that the condition on the adaptive tunable parameter β, that $\beta(t,x)|\theta(t,x)| \geq \max\left\{0, \frac{1}{2}|\nabla\phi|^2 + \frac{1}{2}\mathrm{sgn}(\theta)\left(|\nabla\phi|^2 - |\nabla c|^2\right)\right\}$, suggests that as $|\theta|\to 0$, we could potentially have $\beta\to\infty$. However, we point out here that we need to interpret the whole term $\beta\theta$ as the actual input to the algorithm (9), with $u(t,x) = \beta(t,x)\theta(t,x)$ and the condition will be that $|u| > \max\left\{0, \frac{1}{2}|\nabla\phi|^2 + \frac{1}{2}\mathrm{sgn}(\theta)\left(|\nabla\phi|^2 - |\nabla c|^2\right)\right\}$ when $\theta \neq 0$, which can stay bounded provided the expression on the right hand side stays bounded. In other words, there is no unboundedness issue introduced close to the equilibrium.

B. Discrete Implementation

We now present a consistent discretization scheme for (9) that can be implemented by a robotic swarm. The PDEs on the spatial domain are discretized onto the graph underlying the network of robots in the swarm, where any two robots at $x\in\Omega$ and $y\in\Omega$ separated by a distance $|x-y| < r$ are neighbors in the network graph.

Gradient: To estimate the local gradient of a function φ, robot i at x_i receives the value of φ from its neighbors j, estimates the directional derivatives by $\frac{\varphi(x_j)-\varphi(x_i)}{|x_j-x_i|} = \nabla\varphi\cdot\frac{x_j-x_i}{|x_j-x_i|}$. An estimate of the local gradient is obtained by combining multiple directional derivative estimates.

Weighted Laplacian: We now present a discretization scheme for the term $\nabla\cdot(\lambda\nabla\varphi)$ as in the primal dynamics, which is a λ-weighted Laplace operator acting on φ. We refer the reader to Section 5.2 in [9] for a detailed treatment.

We first consider a symmetric positive definite kernel $\Lambda : \Omega\times\Omega\to\mathbb{R}_{\geq 0}$. The Λ-weighted average variation in ϕ around a point $x\in\Omega$, averaged over a ball $B_r(x)$ of radius $r > 0$ and centered at x is given by:

$$\frac{1}{|B_r(x)|}\int_{B_r(x)}\Lambda(x,y)(\varphi(y)-\varphi(x))\,\mathrm{dvol}(y).$$

The weighted Laplace operator is obtained as the limit of the above average as $r\to 0$. We first let $\lambda(x)=\Lambda(x,x)$ and $\nabla\lambda(x)=\frac{1}{2}(\partial_1\Lambda+\partial_2\Lambda)(x,x)$ (∂_1 and ∂_2 refer to the partial derivatives w.r.t. the first and second arguments respectively), and we get:

$$\nabla\cdot(\lambda\nabla\varphi)=\lim_{r\to 0}\frac{1}{|B_r(x)|}\int_{B_r(x)}\Lambda(x,y)(\varphi(y)-\varphi(x))\,\mathrm{dvol}(y).$$

We employ the above fact; that is, that the λ-weighted Laplace operator can be viewed as the limit of the weighted average of the local variation, to construct a discretization scheme.

Owing to the symmetry of Λ, we have $\partial_1\Lambda(x,x)=\partial_2\Lambda(x,x)$, and by a Taylor expansion around (x,x), we get:

$$\Lambda(y,y)=\Lambda(x,x)+2\partial_1\Lambda(x,x)\cdot(y-x)+\mathscr{O}(|y-x|^2)$$
$$\Rightarrow\ \partial_1\Lambda(x,x)\cdot(y-x)\approx\frac{1}{2}\left(\Lambda(y,y)-\Lambda(x,x)\right).$$

From the above, we obtain:

$$\Lambda(x,y)=\Lambda(x,x)+\partial_1\Lambda(x,x)\cdot(y-x)+\mathscr{O}(|y-x|^2)$$
$$\approx\Lambda(x,x)+\frac{1}{2}\left(\Lambda(y,y)-\Lambda(x,x)\right)=\frac{1}{2}\left(\Lambda(y,y)+\Lambda(x,x)\right). \tag{12}$$

We now construct a discrete (matrix) version of the kernel Λ, which will be a symmetric matrix Λ^d. For an agent i at $x\in\Omega$ and a j at $y\in\Omega$, let $\Lambda^d_{ij}=\Lambda(x,y)$. In implementing the discretized version of the primal-dual dynamics, an agent i will update and use the weights Λ^d_{ij} for its neighbors j. By means of this discretization scheme, we approximate $\nabla\cdot(\lambda\nabla\varphi)(x)\approx(L_{\Lambda^d}\varphi)_i$, where L_{Λ^d} is the weighted Laplacian matrix of the underlying network graph.

To update the weights Λ^d_{ij}, every agent i first updates Λ^d_{ii} according to (9) (where the weights Λ_{ii} correspond to the dual variable λ and (9) is discretized in time by a finite difference method), communicates with neighbors j in the network graph, acquires Λ^d_{jj} and sets:

$$\Lambda^d_{ij}=\frac{1}{2}\left(\Lambda^d_{ii}+\Lambda^d_{jj}\right).$$

The equation above is simply the discrete version of (12).

Time update: We note that the computation of the derivative in time of any function, say φ, on-board the robots while they are in motion, needs to be changed to the total derivative $\frac{d\varphi}{dt}=\partial_t\varphi+\nabla\varphi\cdot\mathbf{v}$. For any time update specified in (9) (which are specified as partial derivatives in time), while being implemented on-board, the transport term $\nabla\phi\cdot\mathbf{v}$ must be added.

The Neumann boundary condition imposed on the PDE is naturally attained in a graph discretization and does not require additional specification. The Neumann boundary condition $\lambda\nabla\phi\cdot\mathbf{n}=0$ implies that $\int_\Omega\nabla\cdot(\lambda\nabla\phi)=\int_{\partial\Omega}\lambda\nabla\phi\cdot\mathbf{n}=0$, by the divergence theorem. What this means for the Laplacian matrix of the graph is that the all-ones vector is in the null space of the Laplacian matrix, which is naturally obtained without any additional specification.

Algorithm 1 Distributed optimal transport

Input: Desired density ρ^*, $g(x)=|\nabla c(x,x)|$, integration time step h

Each agent i at time t:

Requires: Current position x_i, local density estimate $\rho(x_i)$

Communicate with neighbors j, acquire ϕ_j,Λ^d_{jj}, and θ_j

Weight update:

For every neighbor j, set $\Lambda^d_{ij}\to\frac{1}{2}\left(\Lambda^d_{ii}+\Lambda^d_{ij}\right)$

Primal update:

$\phi_i\to\phi_i+h\left[\sum_{j\in\mathscr{N}_i}\Lambda^d_{ij}(\phi_j-\phi_i)+\rho(x_i)-\rho^*(x_i)+(\nabla\phi)_i\cdot\mathbf{v}\right]$

Dual update:

$\Lambda^d_{ii}\to\max\{0,\Lambda^d_{ii}+h(\theta_i+(\nabla\Lambda)_i\cdot\mathbf{v})\}$

$\theta_i\to\theta_i+h\left(\frac{1}{2}|(\nabla\phi)_i|-\frac{1}{2}g(x_i)-\beta\theta_i+(\nabla\theta)_i\cdot\mathbf{v}\right)$

Update current velocity:

$\mathbf{v}_i\to\frac{\Lambda^d_{ii}}{\rho(x_i)}(\nabla\phi)_i$

VI. Simulation Results

We now present simulation results for the gradient flow with the primal-dual dynamics (9) for distributed optimal transport. We note that the results presented are from the simulation of the PDE (1) under the gradient flow corresponding to the primal-dual dynamics (9), over a stationary grid and not an agent-based simulation of the swarm. Figure 1 shows the evolution of the distribution of the swarm over time. The grayscale images show the distribution of the swarm in the domain, with darker shades representing higher density of agents at any given location. The domain is a 50×50 grid, and the PDE (9) was discretized over the grid. The initial distribution value was randomly generated (a random number was generated by the *rand* function in MATLAB for each cell of the grid and then normalized to obtain the probability distribution over the grid). The target density was for the swarms to converge to a bounded polygonal domain, as seen in the final subfigure in Figure 1, where the distribution is uniform in the interior of the polygon and zero outside. The cost of transport was chosen to be $c_{ij}=1$ between neighboring cells i and j in the grid.

As we had noted in the previous section, there exists a fundamental trade-off between optimality and an on-the-fly implementation of the distributed optimal transport. We sought to investigate the extent of this trade-off in simulation by running the primal-dual algorithm (9) for multiple iterations n between two consecutive motion steps. The assumption is that the distributed computation is many times faster than the motion of the agents. Figure 2 is the plot of the L^1-density error $e(t)=\int_\Omega|\rho-\rho^*|\,\mathrm{dvol}$ as a function of time, for various iteration steps of the primal-dual dynamics (to converge to the optimal gradient flow velocity) per motion step. We notice an improvement in the tracking performance (as measured by $e(t)$) with only a few steps of the primal-dual dynamics per motion step, and the convergence to

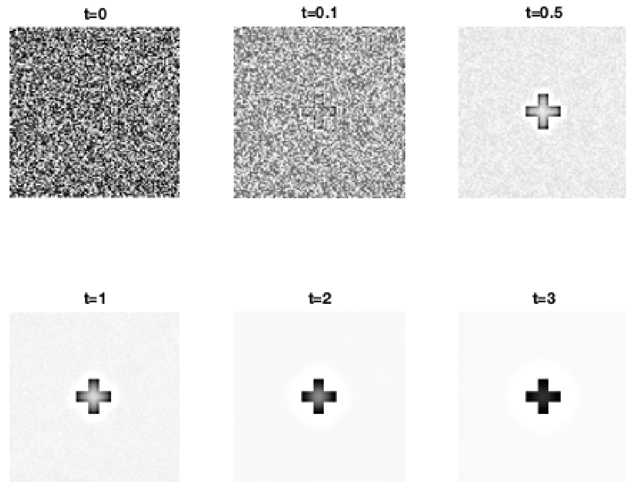

Fig. 1: Distribution of the swarm at various time instants of deployment

true optimal transport (in the sense of decay rate of the error $e(t)$) is obtained with approximately an order ($n \approx 10^1$) of magnitude time scale separation between computation and motion.

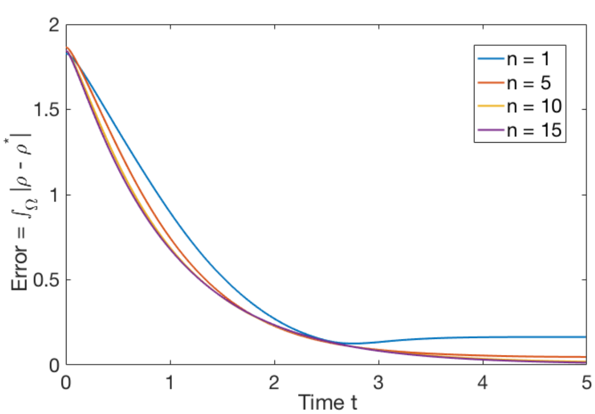

Fig. 2: Plot of density error $\int_\Omega |\rho - \rho^*|\text{dvol}$ with time for various iteration steps n of the primal-dual dynamics per motion step. The plot shows the rate of convergence to the optimal transport gradient flow (represented by $n = 15$)

VII. Conclusions

Working with a macroscopic PDE model of large swarms specified by the continuity equation, we formulated general deployment problems for swarms as a gradient flow and proved a fundamental convergence result. We then designed and analyzed a novel scalable distributed algorithm for gradient flow based on optimal transport theory.

There remain many open lines of inquiry for future work. Firstly, we note that the second-order PDE-based primal-dual dynamics introduced in this paper is novel, and we proved the convergence of solutions satisfying some reasonable assumptions. Future work involves a detailed investigation of the notion and regularity of solutions to the second-order PDE-based primal-dual dynamics. Characterizing analytically the trade-off on optimality due to the on-the-fly implementation of the distributed algorithm for optimal transport is also an interesting open problem of current focus.

References

[1] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*. Applied Mathematics Series, Princeton University Press, 2009.

[2] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton Series in Applied Mathematics, Princeton University Press, 2010.

[3] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[4] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Distributed Autonomous Robotic Systems*, pp. 299–308, Springer, 2002.

[5] S. Bandyopadhyay, S. J. Chung, and F. Y. Hadaegh, "Inhomogeneous Markov chain approach to probabilistic swarm guidance algorithms," in *Int. Conf. on Spacecraft Formation Flying Missions and Technologies*, 2013.

[6] N. Demir, U. Eren, and B. Açıkmeşe, "Decentralized probabilistic density control of autonomous swarms with safety constraints," *Autonomous Robots*, vol. 39, no. 4, pp. 537–554, 2015.

[7] S. Bandyopadhyay, S. J. Chung, and F. Hadaegh, "Probabilistic and distributed control of a large-scale swarm of autonomous agents," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1103–1123, 2017.

[8] V. Krishnan and S. Martínez, "Self-organization in multi-agent swarms via distributed computation of diffeomorphisms," in *Mathematical Theory of Networks and Systems*, (Minneapolis, MN, USA), pp. 706–713, July 2016.

[9] V. Krishnan and S. Martínez, "Distributed control for spatial self-organization of multi-agent swarms," *SIAM Journal on Control and Optimization*, 2018. To appear. A preprint is available at `https://arxiv.org/abs/1705.03109`.

[10] U. Eren and B. Açıkmeşe, "Velocity field generation for density control of swarms using heat equation and smoothing kernels," vol. 50, no. 1, pp. 9405–9411, 2017.

[11] P. Frihauf and M. Krstic, "Leader-enabled deployment onto planar curves: A PDE-based approach," *IEEE Transactions on Automatic Control*, vol. 56, 2011.

[12] K. Elamvazhuthi, H. Kuiper, and S. Berman, "PDE-based optimization for stochastic mapping and coverage strategies using robotic ensembles," *Automatica*, vol. 95, pp. 356–367, 2018.

[13] F. Zhang, A. Bertozzi, K. Elamvazhuthi, and S. Berman, "Performance bounds on spatial coverage tasks by stochastic robotic swarms," *IEEE Transactions on Automatic Control*, vol. 63, no. 6, pp. 1563–1578, 2018.

[14] C. Villani, *Optimal transport: old and new*, vol. 338. Springer, 2008.

[15] J. D. Benamou and Y. Brenier, "A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem," *Numerische Mathematik*, vol. 84, no. 3, pp. 375–393, 2000.

[16] T. Mikami and M. Thieullen, "Optimal transportation problem by stochastic optimal control," *SIAM Journal on Control and Optimization*, vol. 47, no. 3, pp. 1127–1139, 2008.

[17] S. Ferrari, G. Foderaro, P. Zhu, and T. A. Wettergren, "Distributed optimal control of multiscale dynamical systems: a tutorial," *IEEE Control Systems*, vol. 36, no. 2, pp. 102–116, 2016.

[18] S. Bandyopadhyay, S. J. Chung, and F. Y. Hadaegh, "Probabilistic swarm guidance using optimal transport," in *IEEE Conf. on Control Applications*, pp. 498–505, 2014.

[19] L. Evans, *Partial differential equations*. Graduate studies in mathematics, Providence (R.I.): American Mathematical Society, 1998.

[20] V. Krishnan and S. Martínez, "Distributed optimal transport for the deployment of swarms," *arXiv preprint arXiv:1804.01572*, 2018. Extended version.