

Short Papers

A Generalized Reduced Gradient Method for the Optimal Control of Very-Large-Scale Robotic Systems

Keith Rudd, *Member, IEEE*, Greg Foderaro, *Member, IEEE*, Pingping Zhu, *Member, IEEE*, and Silvia Ferrari, *Senior Member, IEEE*

Abstract—This paper develops a new indirect method for distributed optimal control (DOC) that is applicable to optimal planning for very-large-scale robotic (VLSR) systems in complex environments. The method is inspired by the nested analysis and design method known as generalized reduced gradient (GRG). The computational complexity analysis presented in this paper shows that the GRG method is significantly more efficient than classical optimal control or direct DOC methods. The GRG method is demonstrated for VLSR path planning in obstacle-populated environments in which robots are subject to external forces and disturbances. The results show that the method significantly improves performance compared to the existing direct DOC and stochastic gradient methods.

Index Terms—Distributed optimal control (DOC), distributed systems, mobile robotic networks, path planning, very-large-scale robotic (VLSR).

I. INTRODUCTION

As the costs of robots and embedded systems diminish, very-large-scale robotic (VLSR) systems comprised of hundreds of autonomous robots become an important solution to many industrial and military applications [1]. Miniaturized manufacturing and new sensing and communication technologies are enabling the implementation of cooperative VLSR for tasks ranging from hazardous inspection and exploration to monitoring and surveillance [2], [3]. Unlike swarms, which are comprised of thousands of simple units, VLSR systems consist of robots that are each highly capable and autonomous and, together, can plan cooperative behaviors to optimize common objectives and achieve common goals.

Several approaches have been proposed for the control of cooperative VLSR systems [4]. Because optimization of the plans of N cooperative robots is PSPACE-hard [5], when N is very large the problem is typically decoupled into independent components for which solutions can be found quickly, typically at the expense of optimality and completeness. Prioritized planning techniques [6], [7] and path-coordination methods [8], for example, plan the robot trajectories independently and then adjust the control laws to avoid collisions. Behavior-based control specifies a set of simple behaviors for each robot, and their relative importance, in order to achieve a desired macroscopic behavior [1]. In distributed control methods each robot plans its motion based on local

information and properly designed rules, such that the VLSR system displays a desired global behavior or achieves a set of predefined goals.

It was recently shown that the performance of cooperative agents can be formulated as a function of a restriction operator such as a time-varying probability density function (PDF), its lower order moments, or maximum likelihood estimator [1], [9]–[12]. Based on this principle, if robot dynamics and costs are weakly coupled, useful decentralized control strategies for multiscale dynamical systems can be obtained via the nash certainty equivalence (NCE), or mean field, methodology [13], [14]. Similarly to distributed optimal control (DOC), the NCE methodology relies on identifying a consistency relationship between the robot kinodynamic equations and a macroscopic description, which in NCE is the mass of all robots. However, while in NCE methods couplings are produced by the averaging of the microscopic robot kinodynamics and costs, in the DOC approach the couplings do not need to be weak and arise as a result of cooperative objectives.

By extending the optimal control (OC) formulation to VLSR systems, DOC optimizes network-level objectives subject to robot kinodynamic constraints that are possibly nonlinear and stochastic, and may depend on distributed environmental conditions such as obstacles, winds, or currents. Also, DOC affords objectives of a more general form than existing methods and computes optimal VLSR densities that are guaranteed reachable and may entail strong couplings. Previous work showed that, if the robot motion is purely deterministic, DOC optimality conditions consist of parabolic partial differential equations (PDEs) that can be solved by parameterizing the restriction operator via time-varying Gaussian mixtures [9], [10], [15]. To further reduce the computation required, an indirect method of solution was proposed in [15] and demonstrated on a multiagent formation problem.

This paper builds on the preliminary results in [15] to solve the DOC optimality conditions for VLSR systems described by stochastic differential equations (SDEs). A new generalized reduced gradient (GRG) method of solution is presented that brings about both improved VLSR performance and significant computational savings compared to both direct DOC [9], [10] and classical OC [16] solutions. The GRG solution is demonstrated on a new VLSR planning problem that optimizes multiple cooperative objectives for 500 cooperative robots in the presence of wind currents and obstacles, outperforming existing DOC and stochastic gradient planning algorithms.

II. PROBLEM FORMULATION

Consider the problem of planning the actions of a VLSR system comprised of N cooperative robots that are each described by a kinodynamic SDE

$$\dot{\mathbf{x}}_i(t) = \mathbf{f}[\mathbf{x}_i(t), \mathbf{u}_i(t), t] + \mathbf{G}\mathbf{w}(t)$$

$$\mathbf{x}_i(T_0) = \mathbf{x}_{i,0}, \quad i = 1, \dots, N \quad (1)$$

Manuscript received July 7, 2016; revised November 17, 2016; accepted January 27, 2017. Date of publication September 12, 2017; date of current version October 2, 2017. This paper was recommended for publication by Associate Editor L. Pallottino and Editor T. Murphey upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Grant ECCS-1556900.

The authors are with Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14850 USA (e-mail: krudd@ccei.com; gfoderaro@ara.com; pingping.zhu@cornell.edu; ferrari@cornell.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TRO.2017.2686439

where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^n$ denotes the i th robot configuration or state, $\mathbf{u}_i \in \mathcal{U} \subset \mathbb{R}^m$ denotes the i th robot action or control, \mathcal{X} denotes the robot state space, \mathcal{U} is the space of m admissible robot actions, and $\mathbf{x}_{i,0}$ denotes the robot initial conditions. The robot kinodynamic equation (1) is characterized by an additive Gaussian disturbance vector of independent and identically distributed random variables, denoted by $\mathbf{w} \in \mathbb{R}^n$, and $\mathbf{G} \in \mathbb{R}^{n \times n}$ is a constant matrix. In this paper, \mathbf{w} is assumed to obey a standard Gaussian process, but the approach can easily be extended to any other diffusion process. The state \mathbf{x}_i , defined with respect to an inertial coordinate system \mathcal{F}_W , is assumed fully observable and error free for any robot.

Because the robots obey the same SDE (1) and are interchangeable, the VLSR system can be viewed as a multiscale dynamical system with macroscopic state $X(t) = \varphi(\mathbf{x}_i, t)$, where $\varphi: \mathcal{X} \times \mathbb{R} \rightarrow \mathbb{R}$ is a restriction operator. In this paper, the restriction operator is chosen to be a time-varying PDF. Then, at any time t , the probability that any robot i is at a configuration \mathbf{x}_i in a subset B of the configuration space \mathcal{X} is

$$P(\mathbf{x}_i \in B) = \int_B \varphi(\mathbf{x}_i, t) d\mathbf{x}_i \quad (2)$$

where φ is a nonnegative function that satisfies the normalization property

$$\int_{\mathcal{X}} \varphi(\mathbf{x}_i, t) d\mathbf{x}_i = 1 \quad (3)$$

and $N\varphi(\mathbf{x}_i, t)$ is the density of robots in \mathcal{X} .

For any N , the macroscopic kinodynamic equation describing the spatio-temporal evolution of the VLSR system can be derived from the continuity equation, assuming $\mathbf{x}_i(t) \in \mathcal{X}$ for any t . From (1), the PDF φ is advected by a known velocity field $\mathbf{v}_i = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i, t)$ and diffused by the additive Gaussian noise $\mathbf{G}\mathbf{w}$ [17]. From the continuity equation and Gauss's theorem, the time-rate of change of φ can be defined as the sum of the negative divergence of the advection vector ($\varphi\mathbf{v}_i$) and the divergence of the diffusion vector ($\mathbf{G}\mathbf{G}^T \nabla\varphi$) [18]. Then, the VLSR kinodynamics are governed by a parabolic PDE known as an advection–diffusion equation

$$\begin{aligned} \frac{\partial\varphi(\mathbf{x}_i, t)}{\partial t} &= -\nabla \cdot [\varphi(\mathbf{x}_i, t)\mathbf{v}_i] + \frac{1}{2}\nabla \cdot [(\mathbf{G}\mathbf{G}^T)\nabla\varphi(\mathbf{x}_i, t)] \\ &= -\nabla \cdot [\varphi(\mathbf{x}_i, t)\mathbf{f}(\mathbf{x}_i, \mathbf{u}_i, t)] + \nu\nabla^2\varphi(\mathbf{x}_i, t) \end{aligned} \quad (4)$$

where the gradient ∇ is a row vector, $\nu \triangleq \nabla(\mathbf{G}\mathbf{G}^T)$, and (\cdot) denotes the dot product.

Like optimal planning with kinematic and dynamic constraints for a single robot, optimal planning for VLSR systems with kinodynamic constraints can be formulated as an OC problem, as follows. The macroscopic VLSR performance over a fixed time interval $(T_0, T_f]$ is expressed as an integral cost functional of the i th robot PDF and control inputs

$$J = \phi[\varphi(\mathbf{x}_i, T_f)] + \int_{T_0}^{T_f} \int_{\mathcal{X}} \mathcal{L}[\varphi(\mathbf{x}_i, t), \mathbf{u}_i(\mathbf{x}_i, t), t] d\mathbf{x}_i dt \quad (5)$$

where $\mathcal{L}[\cdot]$ denotes the Lagrangian and $\phi[\cdot]$ denotes the terminal cost. Similarly to classical OC [16], the Lagrangian term represents the instantaneous VLSR system performance at any time in $(T_0, T_f]$, such as obstacle avoidance and fuel consumption. The terminal cost represents the system objectives at the end time, such as reaching a goal destination. Because the robot initial conditions are typically given, the initial robot distribution is a known PDF $\varphi_0(\mathbf{x}_i)$ and the VLSR kinodynamic equation (4) is subject to the initial conditions

$$\varphi(\mathbf{x}_i, T_0) = \varphi_0(\mathbf{x}_i) \quad (6)$$

and boundary conditions

$$[\nabla\varphi(\mathbf{x}_i, t)] \cdot \hat{\mathbf{n}} = 0, \quad \forall t \in (T_0, T_f] \quad (7)$$

where $\hat{\mathbf{n}}$ is a unit vector normal to the workspace boundary $\partial\mathcal{X}$. The zero-flux condition (7) prevents the robots from entering or leaving the state space \mathcal{X} such that the continuity equation assumptions are satisfied. Additionally, φ must obey the normalization condition (3), and the state constraint

$$\varphi(\mathbf{x}_i, t) = 0, \quad \forall \mathbf{x}_i \notin \mathcal{X} \quad \text{and} \quad \forall t \in (T_0, T_f]. \quad (8)$$

The VLSR optimal planning problem considered in this paper consists of finding the optimal robot distribution φ^* and optimal robot actions \mathbf{u}_i^* (for all i) that minimize the macroscopic cost J over the time interval $(T_0, T_f]$, subject to the kinodynamic constraint (4), the normalization condition (3), the initial and boundary conditions (6), (7), and the state constraint (8).

III. DOC OPTIMALITY CONDITIONS

The VLSR optimal planning problem consists of optimizing a functional (5) with respect to a function comprised of the robot distribution φ . Thus, the optimality conditions can be derived using calculus of variations [19]. While in previous works the optimality conditions were used to validate the numerical solutions obtained by direct numerical methods, the GRG algorithm presented in this paper seeks to solve the optimality conditions numerically.

As a first step, an augmented cost function is obtained by adjoining the VLSR kinodynamic constraint (4) to the integral cost function (5) using a time-varying Lagrange multiplier $\lambda(\mathbf{x}_i, t)$ as follows

$$\begin{aligned} J_A &= \phi[\varphi(\mathbf{x}_i, T_f)] + \int_{T_0}^{T_f} \int_{\mathcal{X}} \left\{ \mathcal{L}[\varphi(\mathbf{x}_i, t), \mathbf{u}_i, t] + \lambda(\mathbf{x}_i, t) \right. \\ &\quad \left. \times \left[\frac{\partial\varphi(\mathbf{x}_i, t)}{\partial t} + \nabla \cdot [\varphi(\mathbf{x}_i, t)\mathbf{f}(\cdot)] - \nu\nabla^2\varphi(\mathbf{x}_i, t) \right] \right\} d\mathbf{x}_i dt \end{aligned} \quad (9)$$

where $J_A = J_A(\varphi, \mathbf{u}_i, \lambda)$. Then, state and control trajectories that optimize (5) while also satisfying the kinodynamic constraint (4) can be determined by optimizing (9) with respect to φ , \mathbf{u}_i , and λ , where \mathbf{u}_i and λ are both implicit functions of φ by virtue of the control feedback $\mathbf{u}_i = \mathbf{c}[\varphi(\mathbf{x}_i, t)]$.

The effect of variations in the robot PDF, $\delta\varphi$, on the augmented cost function is

$$\begin{aligned} \delta J_A(\delta\varphi) &= \lim_{\epsilon \rightarrow 0} \frac{J_A(\varphi + \epsilon\delta\varphi, \mathbf{u}_i, \lambda) - J_A(\varphi, \mathbf{u}_i, \lambda)}{\epsilon} \\ &= \int_{\mathcal{X}} \frac{\partial\phi}{\partial\varphi} \Big|_{t_f} \delta\varphi d\mathbf{x}_i + \int_{T_0}^{T_f} \int_{\mathcal{X}} \left\{ \frac{\partial\mathcal{L}}{\partial\varphi} \delta\varphi \right. \\ &\quad \left. + \lambda \left[\frac{\partial(\delta\varphi)}{\partial t} + \nabla \cdot (\delta\varphi\mathbf{f}) - \nabla^2\delta\varphi \right] \right\} d\mathbf{x}_i dt \end{aligned} \quad (10)$$

where $\partial\phi/\partial\varphi$ is a functional derivative, and $\mathcal{L}[\cdot]$, $\phi[\cdot]$, $\mathbf{f}[\cdot]$, and $\varphi(\cdot)$ are assumed to be of class C^2 . From the fundamental theorem of variational calculus (FTVC) and integration by parts, the variation in (10) at optimality can be written as

$$\begin{aligned} \delta J_A(\delta\varphi) &= \int_{\mathcal{X}} \left(\frac{\partial\phi}{\partial\varphi} + \lambda \right) \delta\varphi \Big|_{T_f} d\mathbf{x}_i \\ &\quad + \int_{T_0}^{T_f} \int_{\partial\mathcal{X}} [\lambda(\mathbf{f} \cdot \hat{\mathbf{n}}) + \nu\nabla\lambda \cdot \hat{\mathbf{n}}] \delta\varphi d\mathbf{x}_i dt \\ &\quad + \int_{T_0}^{T_f} \int_{\mathcal{X}} \left(\frac{\partial\mathcal{L}}{\partial\varphi} - \frac{\partial\lambda}{\partial t} - \nabla\lambda \cdot \mathbf{f} - \nu\nabla^2\lambda \right) \delta\varphi d\mathbf{x}_i dt = 0. \end{aligned} \quad (11)$$

In order for (11) to hold for arbitrary nonzero variations $\delta\varphi$, the PDE problem

$$\begin{aligned} \frac{\partial\lambda}{\partial t} &= \frac{\partial\mathcal{L}}{\partial\varphi} - \nabla\lambda \cdot \mathbf{f} - \nu\nabla^2\lambda \\ \text{s.t. : } \lambda(\mathbf{x}_i, T_f) &= -\frac{\partial\phi}{\partial\varphi}\Big|_{T_f}, \quad \forall \mathbf{x}_i \in \mathcal{X} \\ \lambda(\mathbf{f} \cdot \hat{\mathbf{n}}) + \nu \nabla\lambda \cdot \hat{\mathbf{n}} &= 0, \quad \forall \mathbf{x}_i \in \partial\mathcal{X} \end{aligned} \quad (12)$$

referred to as *adjoint equation* must be satisfied.

The effect of robot input variations, $\delta\mathbf{u}_i$, by virtue of feedback control is given by

$$\begin{aligned} J_A(\delta\mathbf{u}_i) &= \lim_{\epsilon \rightarrow 0} \frac{J_A(\varphi, \mathbf{u}_i + \epsilon\delta\mathbf{u}_i, \lambda) - J_A(\varphi, \mathbf{u}_i, \lambda)}{\epsilon} \\ &= \int_t \int_{\mathcal{X}} \frac{\partial\mathcal{L}}{\partial\mathbf{u}_i} + \lambda \left[\nabla \cdot \left(\varphi \frac{\partial\mathbf{f}}{\partial\mathbf{u}_i} \delta\mathbf{u}_i \right) \right] d\mathbf{x}_i dt = 0. \end{aligned} \quad (13)$$

From the FTVC and integration by parts, (13) can be written as

$$\begin{aligned} &\int_t \int_{\mathcal{X}} \left\{ \frac{\partial\mathcal{L}}{\partial\mathbf{u}_i} + \lambda \left[\nabla \cdot \left(\varphi \frac{\partial\mathbf{f}}{\partial\mathbf{u}_i} \delta\mathbf{u}_i \right) \right] \right\} d\mathbf{x}_i dt \\ &= \int_t \int_{\mathcal{X}} \left\{ \frac{\partial\mathcal{L}}{\partial\mathbf{u}_i} - \nabla\lambda \cdot \left(\varphi \frac{\partial\mathbf{f}}{\partial\mathbf{u}_i} \right) \delta\mathbf{u}_i \right\} d\mathbf{x}_i dt \\ &\quad + \int_t \left\{ \int_{\partial\mathcal{X}} \lambda \left(\varphi \frac{\partial\mathbf{f}}{\partial\mathbf{u}_i} \cdot \hat{\mathbf{n}} \right) \delta\mathbf{u}_i \right\} d\mathbf{x}_i dt = 0. \end{aligned} \quad (14)$$

Using the no-flux boundary condition (7), the velocity field at $\partial\mathcal{X}$ is either perpendicular to $\hat{\mathbf{n}}$ or zero. Thus, (14) can be simplified to the following *control equation*:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{u}_i} - \nabla\lambda \cdot \left(\varphi \frac{\partial\mathbf{f}}{\partial\mathbf{u}_i} \right) = 0. \quad (15)$$

Finally, at optimality, the effect of variations in the costate, $\delta\lambda$, can be written as

$$\begin{aligned} \delta J_A(\delta\lambda) &= \lim_{\epsilon \rightarrow 0} \frac{J_A(\varphi, \mathbf{u}_i, \lambda + \epsilon\delta\lambda) - J_A(\varphi, \mathbf{u}_i, \lambda)}{\epsilon} \\ &= \frac{\partial\varphi}{\partial t} + \nabla \cdot (\varphi\mathbf{f}) - \nu\nabla^2\varphi = 0 \end{aligned} \quad (16)$$

amounting to a PDE referred to as *forward equation*. Then, the necessary conditions for optimality for the VLSR optimal planning problem in (3)–(8) consist of the parabolic PDEs given by the forward (16) and adjoint (12) equations, and an algebraic equation given by the control equation (15). Although sufficient conditions for optimality can be similarly derived, numerical evaluation of J_A at the extremal (φ^* , \mathbf{u}_i^* , λ^*) typically is sufficient to verify that the extremal is a minimum. A numerical method of solution for the above DOC optimality conditions is presented in Section IV.

IV. GRG DOC SOLUTION

The GRG method belongs to the class of numerical methods referred to as nested analysis and design, which solve PDE-constrained optimization problems by treating the PDE state variables as implicit functions of the PDE parameters [20]. The GRG method was first proposed in [21] for solving nonlinear programs (NLPs) with linear constraints, and was later extended to nonlinear constraints by using penalty function techniques [22], [23]. This paper builds on the results in [15] to develop a DOC solution method that parameterizes the robot controls \mathbf{u}_i , for $i = 1, \dots, N$. Then, approximations of the costate λ and robot distribution φ are obtained by solving the optimality conditions (12) and (16), respectively. Holding these two approximations fixed, the robot control approximations are updated by a gradient-based

algorithm that minimizes the augmented Lagrangian and, ultimately, satisfies the third and final optimality condition (15). After a satisfactory numerical solution is obtained, the parameterized control values are discarded, and the optimal distribution is used to compute the robot control feedback $\mathbf{u}_i^* = \mathbf{c}[\varphi^*(\mathbf{x}_i, t)]$.

Assume every element $u_{i,j}$ of the robot control or action vector \mathbf{u}_i can be parameterized as the sum of M linearly independent Fourier basis functions, $\phi_1(\cdot), \dots, \phi_M(\cdot)$, i.e.,

$$u_{i,j}(\mathbf{x}_i, t) \approx \sum_{\ell=1}^M \phi_{\ell}(\mathbf{x}_i) \alpha_{ij\ell}(t), \quad j = 1, \dots, m, \quad i = 1, \dots, N \quad (17)$$

where $\alpha_{ij\ell}$ are control coefficients to be determined. By holding the control approximation in (17) fixed, the costate λ and macroscopic state φ are obtained from the numerical solution of the adjoint and forward equations, (12) and (16), respectively.

Equations (12) and (16) amount to two coupled parabolic PDEs that can be solved efficiently using a modified Galerkin method, referred to as constrained integration (CINT) [24], and chosen here for its non-dissipative property [25], [26]. In CINT, the PDE solution is approximated by a linear combination of polynomial basis used to satisfy the PDE operator, and Gaussian or radial basis functions (RBFs) used to enforce the boundary conditions at each point of the integration [24]. Given (17), the forward equation (16) becomes a parabolic PDE with Neumann boundary conditions that can be solved numerically for the robot distribution

$$\varphi(\mathbf{x}_i, t) \approx \sum_{l=1}^L \sigma_l(\mathbf{x}_i) \beta_l(t) + \sum_{q=1}^Q \psi_q(\mathbf{x}_i) \gamma_q(t) \quad (18)$$

obtained in terms of L sigmoidal functions $\sigma_1(\cdot), \dots, \sigma_L(\cdot)$, and Q polynomial basis functions $\psi_1(\cdot), \dots, \psi_Q(\cdot)$, where β_l and γ_q are coefficients to be determined. Once the above approximate solution is obtained from (16), the adjoint equation (12) becomes a parabolic PDE in $\lambda(\mathbf{x}_i, t)$, with Dirichlet boundary conditions. Therefore, a numerical solution for $\lambda(\mathbf{x}_i, t)$ can be obtained by a superposition of polynomial basis and RBFs, similarly to (18).

Based on the GRG approach [27], the control coefficients $\alpha_{ij\ell}$ are obtained holding the state and costate approximations fixed, and minimizing the augmented cost function (9) with respect to \mathbf{u}_i . Assume the latest approximations of the macroscopic state φ and costate λ obtained by the CINT method, satisfy (16) and (12), respectively. From (9), an analytical representation of the cost function gradient can be derived using integration by parts as follows

$$\frac{\partial J_A}{\partial \mathbf{u}_i} \Big|_{\varphi, \lambda} = \int_{T_0}^{T_f} \int_{\mathcal{X}} \left[\frac{\partial\mathcal{L}}{\partial\mathbf{u}_i} - \nabla\lambda \cdot \left(\varphi \frac{\partial\mathbf{f}}{\partial\mathbf{u}_i} \right) \right] \delta\mathbf{u}_i d\mathbf{x}_i dt \quad (19)$$

where the gradient is evaluated at given values of φ and λ .

Now, because the basis functions in (17) are given, J_A can be minimized with respect to the control coefficients $\alpha_{ij\ell}$, in lieu of \mathbf{u}_i . Let the time interval (T_0, T_f) be discretized into K equal time steps $\Delta t = (T_f - T_0)/K$, and let $t_k = T_0 + k\Delta t$ denote the discrete-time index with $k = 0, \dots, K$. Using the chain rule of differentiation and (19), the cost function gradient can be approximated as

$$\frac{\partial J_A}{\partial \alpha_{ij\ell}} \Big|_{t=t_k} = \left(\frac{\partial J_A}{\partial u_{i,j}} \frac{\partial u_{i,j}}{\partial \alpha_{ij\ell}} \right) \Big|_{t=t_k} \quad (20)$$

$$\approx \Delta t \int_{\mathcal{X}} \left\{ \left[\frac{\partial\mathcal{L}}{\partial u_{i,j}} - \nabla\lambda \cdot \left(\varphi \frac{\partial\mathbf{f}}{\partial u_{i,j}} \right) \right]_{t=t_k} \phi_{\ell}(\mathbf{x}_i) \right\} d\mathbf{x}_i$$

and, then, used to minimize J_A using a gradient-based optimization algorithm [28]. Once a new and improved control input approximation is obtained, it is held fixed and used to obtain new CINT approximations

Algorithm 1: Pseudocode of GRG Algorithm.

```

1: initialize  $\alpha_{ij\ell}(t)$ 
2:  $\mathbf{u}_i(t) \leftarrow$  evaluate (17)
3: while  $\|\partial J_A / \partial \mathbf{u}_i\| > TOL$  do
4:    $\varphi \leftarrow$  solve forward equation (16)
5:    $\lambda \leftarrow$  solve adjoint equation (12)
6:    $\partial J_A / \partial \mathbf{u}_i \leftarrow$  evaluate (19)
7:   for all  $k$  do
8:      $\alpha_{ij\ell} \leftarrow$  solve control equation (15) using (20)
9:   end for
10: end while

```

TABLE I
COMPUTATIONAL COMPLEXITY RESULTS

	GRG Method	Direct DOC	Classical OC
Hessian update	$O(MXK)$	$O(zXK^2)$	$O(nmN^2K^2)$
QP Solution	$O(M^2XK^3)$	$O(z^2XK^3)$	$O(nm^2N^3K^3)$
Line Search	$O(XK)$	$O(XK)$	$O(nNK)$

for the macroscopic state φ and costate λ . This iterative process is repeated until all three optimality conditions are satisfied within a user-specified tolerance. The pseudocode of the GRG algorithm is shown in Algorithm 1.

V. COMPUTATIONAL COMPLEXITY ANALYSIS

The computational complexity of the indirect GRG-DOC approach presented in Section IV is analyzed and compared to that of the direct DOC method presented in [10], and to that of a direct method for classical OC presented in [29]. Direct methods for classical OC obtain a nonlinear program representation of the kinodynamic equation (1) and of the integral cost function, such that approximations to the optimal state and control trajectories can be computed using a sequential quadratic programming (SQP) algorithm [29]. The direct DOC method in [10] discretizes the VLSR kinodynamic equation (4) and the integral cost function (5) by using a finite volume approach, and by parameterizing the robot PDF φ using a finite Gaussian mixture [10]. Subsequently, a nonlinear program representation of the DOC problem is obtained, and the optimal PDF and control trajectories are computed using SQP [30].

The computational complexity of the three methods is analyzed by assuming that they utilize the same SQP algorithm [28]. The three most expensive steps of this algorithm are the Hessian update, performed using the Broyden–Fletcher–Goldfarb–Shanno method [31], the solution of the quadratic programming (QP) subproblem, and the line-search minimization of the merit function. The order of the computations required by all three steps in GRG, direct DOC, and direct classical OC is summarized in Table I. N is the number of robots, M is the number of GRG basis functions in (17), X is the number of spatial collocation points in \mathcal{X} , K is the number of temporal collocation points in $(T_0, T_f]$, z is the number of Gaussian mixture components in the direct DOC method, and n and m are the microscopic state and control vector dimensions, respectively.

For every method, the solution of the QP subproblem, carried out by a QR decomposition of the active constraints using Householder triangularization [28], is the most expensive computation. For classical OC, the QP solution exhibits cubic growth with respect to K and N , and thus becomes prohibitive for $N \gg 1$. For direct DOC, the QP solution exhibits cubic growth only with respect to K , and quadratic growth with respect to z or M . Thus, for systems with $z \ll N$, the direct DOC

method can bring about considerable computational savings compared to classical OC. In addition to solving a minimization problem via SQP, the indirect GRG method requires solving the parabolic PDEs (16) and (12) at every iteration. Because the CINT method adapts the PDE solution incrementally over time [24], the integration in (20) can be accurately approximated with $O(X)$ computation time, and the Hessian update can be performed with $O(MXK)$ computation time. Therefore, it can be seen from Table I that the GRG method provides significant computational savings compared to direct DOC when $M \lesssim z$.

VI. SIMULATION RESULTS

The indirect GRG method presented in Section IV is demonstrated here on a VLSR system comprised of $N = 500$ mobile robots with single-integrator dynamics

$$\dot{\mathbf{x}}_i(t) = \mathbf{u}_i(t) + \sigma \mathbf{I}_2 \mathbf{w}(t) + \mathbf{v}_i(\mathbf{x}_i), \quad \mathbf{x}_i(t_0) = \mathbf{x}_{i,0} \quad (21)$$

where $\mathbf{x}_i = [x_i \ y_i]^T$ is the robot state, x_i and y_i are the robot xy -coordinates in inertial frame, $\mathbf{u}_i = [u_{i,1} \ u_{i,2}]^T$ is the microscopic control input, and $u_{i,1}$ and $u_{i,2}$ represent the linear velocities in the x - and y -direction, respectively. The disturbance vector is defined as $\mathbf{w} = [\eta_x \ \eta_y]^T$, where η_x and η_y are independent random variables sampled from a standard Gaussian process, \mathbf{I}_2 is the identity matrix, and $\sigma = 0.01$. The robots are also influenced by distributed environmental conditions, such as external wind or current forcing, represented by a velocity field, $\mathbf{v}_i(\mathbf{x}_i) = [v_x(\mathbf{x}_i) \ v_y(\mathbf{x}_i)]^T$, plotted in Fig. 1.

The objectives of the VLSR system are to travel from a given initial distribution $\varphi_0(\mathbf{x}_i)$ to a goal distribution $g(\mathbf{x}_i)$, plotted in Fig. 1, in a fixed time interval $(T_0, T_f]$. During this time, the robots must minimize energy consumption and avoid collisions in an obstacle-populated workspace, $\mathcal{X} = [L_x, 0] \times [0, L_y]$, where $L_x = 20$ km and $L_y = 16$ km. The obstacle geometries are assumed known without error and are used to generate a repulsive potential $U_{\text{rep}}(\mathbf{x}_i)$ over \mathcal{X} , as in the well-known potential field method [32]. Because φ and g are non-negative scalar functions, the error between them is computed as the integral of the squared difference over the workspace. The energy consumption is modeled as an exponential function of the control usage. Thus, the VLSR system performance can be formulated as an integral function of the robot density and control inputs

$$\begin{aligned}
J &= \int_{\mathcal{X}} w_{\varphi} [g(\mathbf{x}_i) - \varphi(\mathbf{x}_i, T_f)]^2 d\mathbf{x}_i + \int_{T_f}^{T_0} \int_{\mathcal{X}} [w_r \varphi(\mathbf{x}_i, t) \\
&\quad \times U_{\text{rep}}(\mathbf{x}_i) + \varphi(\mathbf{x}_i, t) e^{\{w_u [u_{i,1}(\mathbf{x}_i, t)^2 + u_{i,2}(\mathbf{x}_i, t)^2]/2\}}] d\mathbf{x}_i dt \\
&\triangleq \phi[\varphi(\mathbf{x}_i, T_f)] + \int_{T_f}^{T_0} \int_{\mathcal{X}} \mathcal{L}[\cdot] d\mathbf{x}_i dt \quad (22)
\end{aligned}$$

that is to be minimized with respect to the functions φ and \mathbf{u}_i , for $T_0 = 0$ and $T_f = 15$ h. The weighting coefficients w_{φ} , w_r , and w_u are based on relative importance of objectives chosen by the user.

The GRG method presented in Section IV is implemented to obtain the optimal, time-varying robot distribution φ^* that minimizes (22) subject to the robot kinodynamic constraints (21). In particular, the GRG control approximation (17) is obtained in terms of Fourier sine basis functions

$$\phi_{\ell}(\mathbf{x}_i) \triangleq \sum_{a=1}^A \sin \left[\frac{\ell \pi x_i}{2L} \right] \sin \left[\frac{a \pi y_i}{2L} \right] \quad (23)$$

such that $\mathbf{u}_i = 0$ on the boundary, and $\mathbf{f} \cdot \hat{\mathbf{n}} = 0$, which simplifies the boundary condition in (12) to $\nabla \lambda \cdot \hat{\mathbf{n}} = 0$. The GRG macroscopic state approximation (18) is obtained in terms of sigmoidal functions in

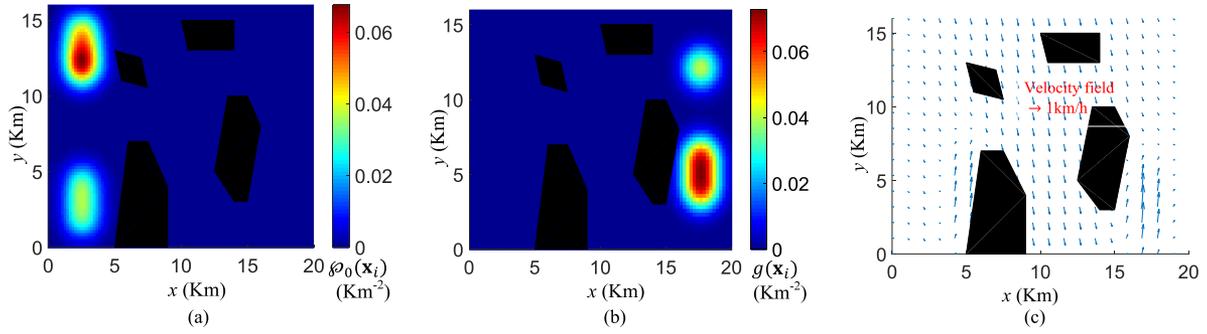


Fig. 1. (a) VLSR system is to travel from an initial robot distribution, (b) to a goal robot distribution, (c) in a simulated wind velocity vector field.

the form

$$\sigma_l(\mathbf{x}_i) \triangleq \exp[-\gamma\|\mathbf{x}_i - \mathbf{x}\|^2] \left\{ \frac{\exp[(\mathbf{x}_i - \mathbf{x})^T \hat{\mathbf{n}}] - 1}{\exp[(\mathbf{x}_i - \mathbf{x})^T \hat{\mathbf{n}}] + 1} \right\} \quad (24)$$

used to satisfy Neumann boundary conditions [24], where $\hat{\mathbf{n}}$ is the unit vector normal to the boundary at any collocation point $\mathbf{x} \in \partial\mathcal{X}$, and in terms of the polynomial basis

$$\psi_q(\mathbf{x}_i) = \sum_{b=1}^B x_i^q y_i^b \quad (25)$$

where $Q = 14$ and $B = 14$. The GRG costate approximation takes a form that is analogous to (18), but in place of the sigmoidal basis functions, (24) employs RBFs in the form

$$\tilde{\sigma}_l(\mathbf{x}_i) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}\|^2) \quad (26)$$

in order to satisfy the Dirichlet boundary conditions [24], where γ is the shape parameter.

From (17) and (23), the gradient approximation (20) is given by the vector

$$\left[\frac{\partial J}{\partial \alpha_{i1\ell}} \Big|_{t=t_k} \quad \frac{\partial J}{\partial \alpha_{i2\ell}} \Big|_{t=t_k} \right]^T \approx \Delta t \times \left[\int_{\mathcal{X}} \left\{ [w_u u_{i,1} e^{\frac{w_u}{2}(u_{i,1}^2 + u_{i,2}^2)} - \varphi \frac{\partial \lambda}{\partial x_i}] \sin\left(\frac{\ell \pi x_i}{2L}\right) \sin\left(\frac{a \pi y_i}{2L}\right) \right\} dx_i \right. \\ \left. \int_{\mathcal{X}} \left\{ [w_u u_{i,2} e^{\frac{w_u}{2}(u_{i,1}^2 + u_{i,2}^2)} - \varphi \frac{\partial \lambda}{\partial y_i}] \sin\left(\frac{\ell \pi x_i}{2L}\right) \sin\left(\frac{a \pi y_i}{2L}\right) \right\} dx_i \right] \quad (27)$$

and is used to update the control coefficients $\alpha_{ij\ell}$ (for $j = 1, 2$) at every t_k . Then, holding the control coefficients fixed, the parameters of the robot distribution and costate approximations are updated by solving the optimality conditions (16) and (12), respectively. Starting with the initial guess $\alpha_{ij\ell} = 0$ for all i, j , and ℓ , the optimal solution is reached using $K = 100$ discrete time steps.

The optimal time-varying robot distribution φ^* , obtained by the GRG algorithm, is used by each robot to evaluate the feedback control law $\mathbf{u}_i^* = \mathbf{c}[\varphi^*(\mathbf{x}_i, t)]$. Approaches such as Voronoi diagrams, Delaunay triangulations, and potential navigation functions can be used to design a control law based on a given robot PDF [1], [4]. In this paper, the potential function approach in [9] and [10] is adopted so as to minimize deviations between the observed robot PDF $\hat{\varphi}$ and the optimal PDF φ^* , thus preventing robots' intracollisions, and compensating for modeling errors and disturbances. Let the attractive potential of agent i be defined as

$$U_i(\mathbf{x}_i, t) \triangleq \frac{1}{2} [\hat{\varphi}(\mathbf{x}_i, t + \delta t) - \varphi^*(\mathbf{x}_i, t + \delta t)]^2 \quad (28)$$

where δt is a time-shift parameter that allows the control law to look ahead in time to prevent agents from lagging behind. The estimate $\hat{\varphi}(\mathbf{x}_i, t + \delta t)$ is computed by stepping the VLSR PDE (4) forward in time by δt from the PDF $\hat{\varphi}(\mathbf{x}_i, t)$, obtained from observations of robot

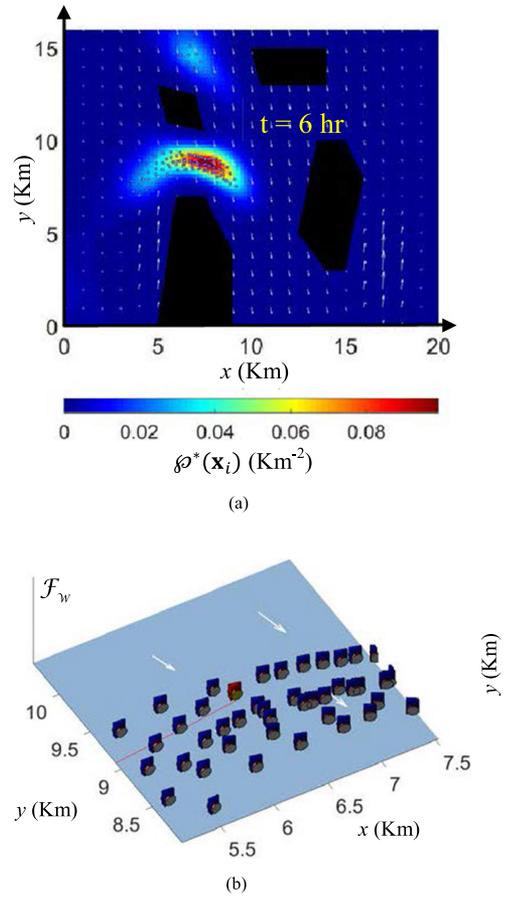


Fig. 2. Close-up view of select robots (a) selected from the VLSR system with optimal robot distribution and configurations (b) obtained by the GRG DOC algorithm in the presence of winds, at $t = 6$ h.

positions at time t using kernel density estimation [33]. Let the desired heading angle expressed by the direction of the negative gradient of the attractive potential be denoted by $\Theta(-\nabla U_i)$, and the agent actual heading angle be denoted by $\hat{\theta}_i$. Then, a feedback control law that follows the navigation function (28) is

$$\mathbf{u}_i = [v_c \quad Q(\hat{\theta}_i, -\nabla U_i)]^T \quad (29)$$

where v_c is the agent speed and

$$Q(\cdot, \cdot) \triangleq \{a(\hat{\theta}_i) - a(\Theta(-\nabla U_i))\} \text{sgn}\{a(\Theta(-\nabla U_i)) - a(\hat{\theta}_i)\} \quad (30)$$

where $\text{sgn}(\cdot)$ is the sign function and $a(\cdot)$ is an angle wrapping function.

The VLSR behavior obtained by the GRG algorithm subject to a wind velocity field is shown in Figs. 2 and 3, where the optimal robot

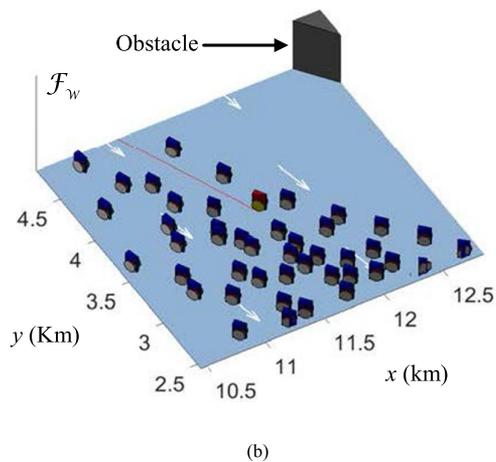
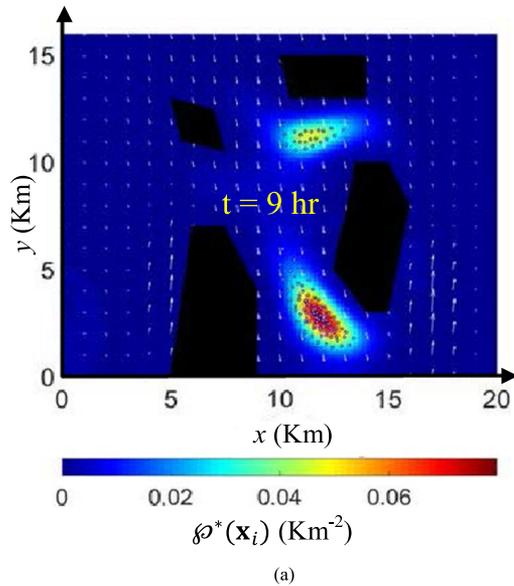


Fig. 3. Close-up view of select robots (a) selected from the VLSR system with optimal robot distribution and configurations (b) obtained by the GRG DOC algorithm in the presence of winds, at $t = 9$ h.

positions and distribution, as well as a close-up robot animation, are shown at two moments in time. The close-up robot animation is shown from a fixed viewpoint in three-dimension (3-D). The optimal robot distribution, along with all N robot positions (circles), is plotted with respect to the 2-D workspace \mathcal{X} .

A. Influence of Environmental Conditions

The difference in the robot behaviors under different environmental conditions can be seen from the paths planned by the robots with and without the external (wind) forcing plotted in Figs. 4 and 5, respectively. The paths of robots indexed by $i = 1, \dots, N$ are plotted in colors from coolest to warmest, as the index increases in size to illustrate the robot behaviors over time. It can be seen that, in order to minimize energy consumption while also reaching the target PDF, the robots that are subject to the wind velocity field travel in larger number below the bottom-right obstacle (see Fig. 4), while in the absence of winds, the robots travel in larger number above the obstacle to follow a shorter path (see Fig. 5). These results show that the GRG DOC method can account for distributed environmental conditions and optimize the global performance of the network accordingly.

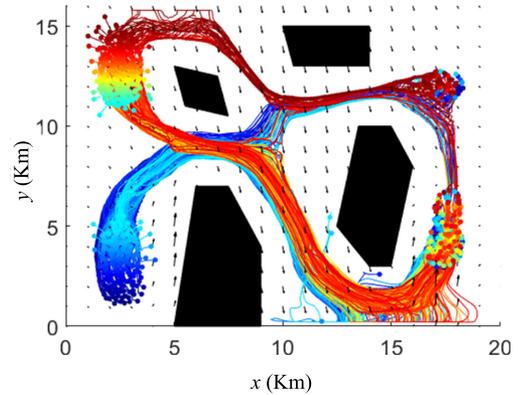


Fig. 4. GRG results for VLSR path planning in the presence of winds.

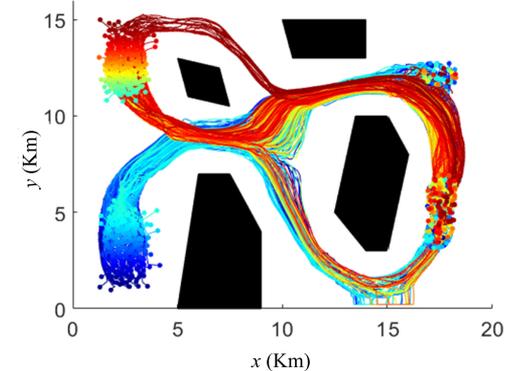


Fig. 5. GRG results for VLSR path planning in the absence of winds.

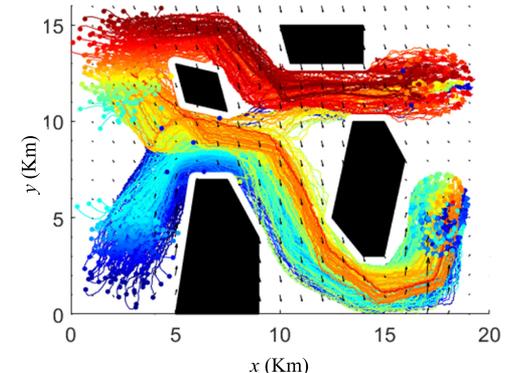


Fig. 6. Trajectories of robots obtained by the direct DOC method in the presence of winds.

B. Performance Comparison

The VLSR system performance obtained by the indirect GRG method presented in this paper is compared to that of the direct DOC method presented in [10] and to the stochastic gradient descent method presented in [34]. The stochastic gradient method obtains control laws for mobile robotic networks by expressing the goal configuration as the minimum of an objective function, and by using a gradient descent algorithm to obtain a local motion plan for each robot that is decoupled from other robots. While it is inspired by potential field methods, stochastic gradient can also be implemented for robots with stochastic dynamic effects, such as (21). For comparison, the goal configurations are obtained by sampling the goal robot PDF $g(\mathbf{x}_i)$ and the initial configurations are obtained by sampling $\wp_0(\mathbf{x}_i)$. The VLSR paths planned by the direct DOC method and the stochastic gradient method are shown in Figs. 6 and 7, respectively. The values of the

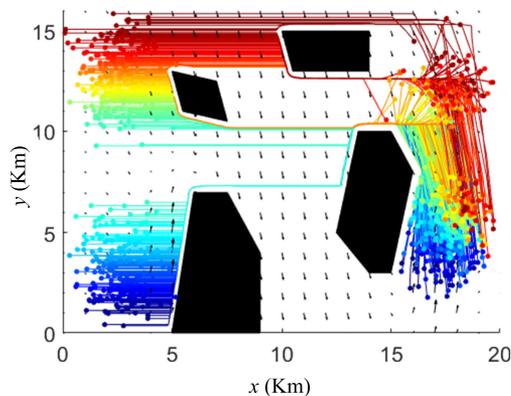


Fig. 7. Trajectories of robots obtained by the stochastic gradient method in the presence of winds.

VLSR cost function (22) obtained by the GRG method, direct DOC, and stochastic gradient method, are $J = 67.16, 78.79,$ and $104.25,$ respectively. Thus, it can be seen that the GRG method obtains a lower total cost than direct DOC thanks to its ability to approximate a more general robot distribution, while also requiring less computation (see Section V). Furthermore, the GRG method reduces the cost by 35.5% when compared to the stochastic gradient method.

VII. CONCLUSION

This paper presents an indirect GRG method for optimal planning in VLSR systems with multiple cooperative objectives and subject to external forces and disturbances. The method is derived using a DOC problem formulation with stochastic robot kinodynamic equations. Unlike previous DOC algorithms, the GRG method seeks to solve the optimality conditions obtained from calculus of variations numerically. Because the stochastic DOC optimality conditions consist of parabolic PDEs, the optimal robot distribution and behaviors are obtained by solving the forward and adjoint PDEs numerically, and then minimizing the augmented Lagrangian by an analytical gradient approximation. It is shown that the GRG solution displays a computational complexity that is significantly reduced compared to both classical OC and direct DOC. Also, the GRG algorithm is shown to outperform both existing DOC and stochastic gradient algorithms for VLSR path planning in obstacle-populated environments, subject to distributed environmental conditions.

REFERENCES

- [1] J. H. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robots," *Robot. Auton. Syst.*, vol. 27, no. 3, pp. 171–194, 1999.
- [2] A. T. Baisch, O. Ozcan, B. Goldberg, D. Ithier, and R. J. Wood, "High speed locomotion for a quadrupedal microrobot," *Int. J. Robot. Res.*, vol. 33, pp. 1063–1082, 2014.
- [3] M. Steinberg, J. Stack, and T. Paluszkiwicz, "Long duration autonomy for maritime systems: Challenges and opportunities," *Auton. Robots*, vol. 40, no. 7, pp. 1119–1122, 2016.
- [4] C. C. Cheah, S. P. Hou, and J. J. E. Slotine, "Region-based shaped control for a swarm of robots," *Automatica*, vol. 45, pp. 2406–2411, 2009.
- [5] L. E. Parker, "Path planning, and motion coordination in multiple mobile robot teams," *Encyclopedia of Complexity and System Science*. Heidelberg, Germany: Springer, 2009.
- [6] J. Ota, T. Arai, C. Ferrari, and E. Pagello, "Multirobot motion coordination in space and time," *Robot. Auton. Syst.*, vol. 25, pp. 219–229, 1998.
- [7] S. Thrun, M. Bennewitz, and W. Burgard, "Finding and optimizing solvable priority schemes for decoupled path planning techniques for teams of mobile robots," *Robot. Auton. Syst.*, vol. 41, no. 2, pp. 89–99, 2002.
- [8] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *Int. J. Robot. Res.*, vol. 5, no. 3, pp. 72–89, 1986.
- [9] G. Foderaro and S. Ferrari, "Necessary conditions for optimality for a distributed optimal control problem," in *Proc. IEEE Conf. Decision Control*, Atlanta, GA, USA, 2010, pp. 4831–4838.
- [10] G. Foderaro, S. Ferrari, and T. A. Wettergren, "Distributed optimal control for multi-agent trajectory optimization," *Automatica*, vol. 50, no. 1, pp. 149–154, 2014.
- [11] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," *IEEE Trans. Robot. Autom.*, vol. 14, no. 6, pp. 926–939, Dec. 1998.
- [12] J. P. Desai, J. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Trans. Robot. Autom.*, vol. 17, no. 6, pp. 905–908, Dec. 2001.
- [13] M. Huang, P. E. Caines, and R. P. Malhame, "The NCE (mean field) principle with locality dependent cost interactions," *IEEE Trans. Automat. Control*, vol. 55, no. 12, pp. 2799–2805, Dec. 2010.
- [14] M. Huang, R. P. Malhame, and P. E. Caines, "Large population stochastic dynamic games: Closed-loop Mckean-Vlasov systems and the nash certainty equivalence principle," *Commun. Inf. Syst.*, vol. 6, no. 3, pp. 221–252, 2006.
- [15] K. Rudd, G. Foderaro, and S. Ferrari, "A generalized reduced gradient method for optimal control of multiscale dynamical systems," in *Proc. IEEE Conf. Decision Control*, Firenze, Italy, 2013, pp. 3857–3863.
- [16] R. F. Stengel, *Optimal Control and Estimation*. New York, NY, USA: Dover, 1986.
- [17] J. P. Boyd, *Chebyshev and Fourier Spectral Methods*, 2nd ed. New York, NY, USA: Dover, 2001.
- [18] X. Mao, *Stochastic Differential Equations and Applications*. Sawston, U.K.: Horwood, 1997.
- [19] C. Fox, *An Introduction to the Calculus of Variations*. New York, NY, USA: Dover, 1987.
- [20] L. Biegler, *Large-Scale PDE-Constrained Optimization* (ser. Lecture Notes in Computational Science and Engineering), vol. 30. New York, NY, USA: Springer-Verlag, 2003.
- [21] P. Wolfe, "Methods of nonlinear programming," in *Recent Advances in Mathematical Programming*. New York, NY, USA: McGraw-Hill, 1963.
- [22] J. Abadie and J. Carpentier, "Generalization of the Wolfe reduced gradient method to the case of nonlinear constraints," in *Optimization*. London, U.K.: Academic, 1969, pp. 37–47.
- [23] H. Dommel and W. Tinney, "Optimal power flow solutions," *Trans. Power App. Syst.*, vol. 87, pp. 1866–1876, 1968.
- [24] K. Rudd and S. Ferrari, "A constrained integration (CINT) approach to solving partial differential equations using artificial neural networks," *Neurocomputing*, vol. 155, pp. 277–285, 2015.
- [25] M. Kaazempur-Mofrad and C. Ethier, "An efficient characteristic Galerkin scheme for the advection equation in 3-d," *Comput. Methods Appl. Mech. Eng.*, vol. 191, no. 46, pp. 5345–5363, 2002.
- [26] T. Sengupta, S. Talla, and S. Pradhan, "Galerkin finite element methods for wave problems," *Sadhana*, vol. 30, pp. 611–623, 2005.
- [27] W. Sun and Y. Yuan, *Optimization Theory and Methods: Nonlinear Programming* (Springer Optimization and Its Applications Series). New York, NY, USA: Springer Science+Business Media, 2006.
- [28] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations," *Numer. Anal.*, vol. 630, pp. 144–157, 1978.
- [29] J. Betts, "Survey of numerical methods for trajectory optimization," *J. Guidance, Control, Dyn.*, vol. 21, no. 2, pp. 193–207, 1998.
- [30] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.
- [31] R. Fletcher, "A new approach to variable metric algorithms," *Comput. J.*, vol. 13, no. 3, pp. 317–322, 1970.
- [32] J. C. Latombe, *Robot Motion Planning*. Norwell, MA, USA: Kluwer, 1991.
- [33] J. S. Simonoff, *Smoothing Methods in Statistics*. New York, NY, USA: Springer-Verlag, 1996.
- [34] J. L. Ny and G. J. Pappas, "Adaptive deployment of mobile robot networks," *IEEE Trans. Automat. Control*, vol. 58, no. 3, pp. 654–666, Mar. 2013.