Robotica (2014) volume 32, pp. 305–324. © Cambridge University Press 2014
doi:10.1017/S0263574714000022

Stochastic optimal enhancement of distributed formation control using Kalman smoothers

Ross P. Anderson† and Dejan Milutinović‡*

†*Department of Applied Mathematics and Statistics, Mail Stop SOEGrad, University of California, Santa Cruz, 1156 High St, Santa Cruz, CA 95064, USA*
‡*Computer Engineering Department, Mail Stop SOE2, University of California, Santa Cruz, 1156 High St, Santa Cruz, CA 95064, USA*

(Accepted December 20, 2013. First published online: January 31, 2014)

SUMMARY
Beginning with a deterministic distributed feedback control for nonholonomic vehicle formations, we develop a stochastic optimal control approach for agents to enhance their non-optimal controls with additive correction terms based on the Hamilton–Jacobi–Bellman equation, making them optimal and robust to uncertainties. In order to avoid discretization of the high-dimensional cost-to-go function, we exploit the stochasticity of the distributed nature of the problem to develop an equivalent Kalman smoothing problem in a continuous state space using a path integral representation. Our approach is illustrated by numerical examples in which agents achieve a formation with their neighbors using only local observations.

KEYWORDS: Multi-robot systems; Control of robotic systems; Motion planning; Mobile robots.

1. Introduction

The focus of this work is on an approach to optimally enhance a given distributed feedback control for nonholonomic agents, such as mobile robots,[1] or vehicles. We are motivated by the problem of distributed formation control, in which each agent is tasked with attaining and maintaining pre-specified distances from its neighbors. Various control approaches have been proposed for formation control, including proportional-integral-derivative (PID) control,[2,3] artificial potentials,[4–6] navigation functions,[7] constraint forces,[8] geometric methods,[9] adaptive control design,[10,11] sliding mode control,[12] and consensus algorithms,[13,14] for example. These approaches are attractive due to their often intuitive design principles, the transparent, analytic forms of the resulting control laws, and their generally satisfactory results. However, developing a control law that is both optimal and robust to uncertainty presents a greater challenge.

Here we begin with a *reference* distributed control policy[4] for formations of nonholonomic vehicles consisting of a feedback-controlled turning rate and acceleration and based on an artificial potential function. Next, we numerically compute the additional additive control input necessary to drive the non-optimal system into a formation *optimally* and in a manner that is *robust to uncertainty*. Consequently, our control approach is optimal, and, due to the adopted artificial potential function,[4] it provides collision-free agent trajectories. Moreover, in some sense to be defined later, the computed optimal feedback control is "close" to the reference feedback control, preserving many of the nicer properties of the latter, including collision avoidance and, to some extent, qualitative behavior of the controlled agents. We are not aware of other works that improve upon existing, rather than build from the ground up, mobile robot feedback control laws in this manner.

To compute an optimal feedback control, one must solve the Hamilton–Jacobi–Bellman (HJB) equation, which is a nonlinear partial differential equation (PDE), and, accordingly, the solution of the optimal control problem is necessarily numerical. However, the computational complexity of the solution to the HJB equation grows exponentially with the state space dimension of the system

* Corresponding author. E-mail: dejan@soe.ucsc.edu

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

(i.e., with the size of the robotic team), making conventional approaches to computing the stochastic optimal feedback control intractable. In this work, we exploit the distributed nature of the problem at hand in order to make the solution to the HJB equation computationally feasible. The distributed formation control problem is inherently stochastic – from the perspective of one agent, its neighbors' observations and control inputs are unknown, as well as the effects of these inputs on agent trajectories due to model uncertainties. Accordingly, this work considers the problem of controlling one agent based on its own observations of its neighbors in a way that anticipates the probability distribution of their future motion. This probability distribution arises from an assumption that a prior for the unknown control input of an agent can be robustly described as Brownian motion,[15] which, for the agent model considered in this paper, results in a so-called "banana distribution" prior.[16] Based on our prior and the system kinematics, we can induce a probability distribution of the relative state $\mathbf{x}$ to all neighbors in an interval $(\mathbf{x}, \mathbf{x} + d\mathbf{x})$ at a particular future time.[17] It follows that the cost function to be minimized by an agent not only computes the optimal control with respect to the current system state as viewed by that agent but also with respect to the distribution of possible trajectories originating from the current state.

Besides aiding in the creation of a robust control law, this distribution over the future system trajectories serves several additional purposes. For example, the distribution encodes the same type of information that must be repeatedly transmitted among neighbors in some other distributed formation control approaches,[18] e.g., assumed future trajectories of a neighbor. Perhaps more importantly, this distribution over future system trajectories can be used to statistically infer the probability distribution of the *control*, and hence the optimal additive control required for the considered reference feedback control. In particular, the relations between the solutions to optimal control PDEs and the probability distribution of stochastic differential equations[19–21] allow certain stochastic optimal control problems to be written as an estimation problem on the distribution of optimal trajectories in continuous state space in a manner known as the path integral (PI) approach.[22–26] (1)

Related works incorporating the PI framework for multi-agent systems[29–32] designed control for systems in which agents cooperatively compute their control from a marginalization of the joint probability distribution of the group's trajectory. In this paper(2) we develop a method by which agents independently compute their controls without explicit communication. Moreover, previous works using the PI approach have formulated an unconstrained receding horizon optimal control problem for which stability is difficult to guarantee.[35] We therefore consider two formulations admitting time-invariant feedback control policies. The first is based on a planning horizon that ends only when the formation is reached, while the other is based on an infinite-horizon, discounted cost control problem.

In addition, we establish a connection between the presented optimal feedback control problem and nonlinear Kalman smoothing algorithms so that each agent can compute its control in real-time in a way that preserves the optimality and stability properties of the HJB equation solution. That a Kalman smoother can be used to compute an optimal control is related to the well-established duality between linear-quadratic-Gaussian (LQG) control and linear-Gaussian estimation.[36] Although other robotic control algorithms have employed Kalman filters or smoothers for motion planning and path planning,[37–41] the relation of the nonlinear Kalman smoothing algorithm to the HJB equation has not been exploited. Moreover, since our Kalman smoothing control computations are based on the current system state and do not require a global HJB equation solution, we describe how our algorithm may be used to test, pointwise in state space, the possibility for analytic improvements to the reference feedback control for optimality and robustness to uncertainties.

In our approach, after each agent computes and applies the optimal control computed by its Kalman smoother, it observes the new state of its neighbors, and then the process repeats. This method is similar in spirit to receding horizon control/model predictive control (MPC)[42] and related suboptimal algorithms,[43] some of which have been used previously for vehicle formation control.[18,44,45] However, our approach differs in three respects. First, as previously discussed, the control solution is derived from the HJB equation instead of an open-loop, numerical optimization problem. In addition, for reasons that will become clear in the sequel, the system trajectories to which the Kalman smoother is applied are uncontrolled, whereas MPC treats the control along these paths as decision variables. Finally, our Kalman smoother solution is generally computationally fast compared with nonlinear

(1) There is also an analogous approach in the open-loop control case.[27,28]
(2) Preliminary versions of some portions of this work have been presented previously elsewhere.[33,34]

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

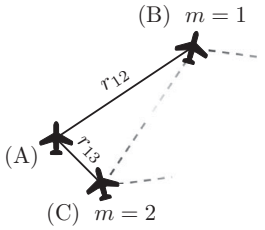

Fig. 1. In this scenario, agent A observes neighbors B and C, labeled by agent A as $m = 1$ and 2, and attempts to achieve the inter-agent spacings $r_{12} = \delta_{12}$ and $r_{13} = \delta_{13}$, as well as alignment of heading angles and speeds. Agent A is unaware of any other observation (dash lines). Consequently, both the reference control and the optimal controls computed by agent A do not include the observation B–C.

optimization methods for systems with large dimensional state and control space, and the algorithm quite naturally handles stochasticity. We point out that without the notion of uncertainty in a neighbor's control, an optimization problem resulting from distributed MPC may predict a neighbor's trajectory incorrectly and without an explicit remedy for this issue.

This paper is organized as follows. Section 2 introduces the formation control problem as viewed by a single agent in the group, followed by a derivation of a PI representation in Section 3. Section 4 presents a Kalman smoother method for computing individual agent control. Section 5 compares the optimal feedback control computed by Kalman smoothers against that computed using a numerical method involving discretization of the state space and examines the areas of the state space where the analytical reference feedback control could be improved. We continue in Section 5 to illustrate our method with simulations of five agents achieving formations, and conclude with Section 6.

2. Control Problem Formulation

We consider a team of agents, each described by a Cartesian position (x_m, y_m), a heading angle θ_m, a speed v_m, $m = 1, \ldots, M_{\text{tot}}$, and the kinematic model:

$$\begin{aligned} \mathrm{d}x_m(t) &= v_m \cos\theta_m \mathrm{d}t \\ \mathrm{d}y_m(t) &= v_m \sin\theta_m \mathrm{d}t \\ \mathrm{d}\theta_m(t) &= \omega_m \mathrm{d}t + \sigma_{\theta,m}\mathrm{d}w_{\theta,m} \\ \mathrm{d}v_m(t) &= a_m \mathrm{d}t + \sigma_{v,m}\mathrm{d}w_{v,m}, \end{aligned} \tag{1}$$

where ω_m and a_m are the feedback-controlled turning rate and acceleration, respectively, and $\mathrm{d}w_{\theta,m}$ and $\mathrm{d}w_{v,m}$ are mutually independent Wiener process increments with corresponding intensities $\sigma_{\theta,m}$ and $\sigma_{v,m}$, respectively. Our goal is for the distances separating the agents to reach a set of predefined nominal distances δ_{mn}, where $m, n = 1, \ldots, M_{\text{tot}}$, $m \neq n$, and for the heading angles and speeds to be equal.

To achieve a distributed control, the problem is formulated from the perspective of just one agent whose state (x, y, θ, v) is notated without subscript. This agent observes M neighbors with subscripts $m = 1, \ldots, M$, $M \leq M_{\text{tot}}$, regardless of the total number of agents M_{tot} in the team, and computes an optimal feedback control to reach a formation with respect to its observed neighbors. No further information is assumed about the observations made by its neighbors. In other words, the only interactions modeled by an agent are undirected (bilateral) and only include that agent's observed neighbors. This is illustrated by Fig. 1.

We introduce collision avoidance by adding to the original kinematic model an artificial potential function-based control[4] for collision-free velocity vector alignment of groups of vehicles described by a noiseless version of (1). The evolution equations for $\theta(t)$ and $v(t)$ become

$$\mathrm{d}\theta(t) = \omega_R \mathrm{d}t + \omega \mathrm{d}t + \sigma_\theta \mathrm{d}w_\theta, \tag{2}$$

$$\mathrm{d}v(t) = a_R \mathrm{d}t + u \mathrm{d}t + \sigma_v \mathrm{d}w_v, \tag{3}$$

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

so that the controls ω and a are interpreted as the optimal *correction* terms to the reference turning rate feedback control ω_R and acceleration feedback control a_R, respectively, in the presence of uncertainty. For brevity, we omit details[4] of ω_R and a_R equations. Suffice it to say that in the deterministic case ($\sigma_{\theta,m} = \sigma_{v,m} = 0$), the reference feedback control ω_R and a_R ensures collision avoidance, that is, the inter-agent distances remain strictly positive,[(3)] and it aligns the agent velocity vectors. It guarantees that the group will tend toward a minimum of the an artificially constructed potential $V(r_m)$, where $r_m = \sqrt{(x - x_m)^2 + (y - y_m)^2}$, $m = 1, \ldots, M$. Our specific choice of $V(r_m) = \delta_m^2 ||r_m||^{-2} + 2 \log ||r_m||$ causes the potential[4] to reach minimum value when all inter-agent distances $r_m \to \delta_m$.

Evolutions of the heading angle $\theta_m(t)$ and speed $v_m(t)$ of a neighbor m are based on its own observations that are unknown to any other agent. Therefore, their increments for an agent m are modeled as Gaussian random variables:

$$\mathrm{d}\theta_m(t) = \omega_{R,m}\mathrm{d}t + N\left(\omega_m \mathrm{d}t, \sigma_{\theta,m}^2 \mathrm{d}t\right) \tag{4}$$

$$\mathrm{d}v_m(t) = a_{R,m}\mathrm{d}t + N\left(a_m \mathrm{d}t, \sigma_{v,m}^2 \mathrm{d}t\right), \tag{5}$$

where $\omega_{R,m}$ and $a_{R,m}$ are the reference controls computed for neighboring agents (see Fig. 1 caption), and where $\sigma_{\theta,m}$ and $\sigma_{v,m}$ take into account both kinematic uncertainty *and control uncertainty* so that $\sigma_{\theta,m} \geq \sigma_\theta$ and $\sigma_{v,m} \geq \sigma_v$. In summary, the kinematic model for one agent with M observed neighbors takes the form

$$\mathrm{d}\Delta x_m(t) = \mathrm{d}(x(t) - x_m(t)) = v\cos\theta \mathrm{d}t - v\cos\theta_m \mathrm{d}t, \tag{6}$$

$$\mathrm{d}\Delta y_m(t) = \mathrm{d}(y(t) - y_m(t)) = v\sin\theta \mathrm{d}t - v\sin\theta_m \mathrm{d}t, \tag{7}$$

$$\mathrm{d}\theta(t) = \omega_R \mathrm{d}t + \omega \mathrm{d}t + \sigma_\theta \mathrm{d}w_\theta, \tag{8}$$

$$\mathrm{d}v(t) = a_R \mathrm{d}t + a\mathrm{d}t + \sigma_v \mathrm{d}w_v, \tag{9}$$

$$\mathrm{d}\theta_m(t) = \omega_{R,m}\mathrm{d}t + \omega_m \mathrm{d}t + \sigma_{\theta,m}\mathrm{d}w_{\theta,m}, \tag{10}$$

$$\mathrm{d}v_m(t) = a_{R,m}\mathrm{d}t + a_m \mathrm{d}t + \sigma_{v,m}\mathrm{d}w_{v,m}, \qquad m = 1, \ldots, M,\ M \leq M_{\mathrm{tot}}. \tag{11}$$

This model can be written in a general form:

$$\mathrm{d}\mathbf{x}(t) = f(\mathbf{x})\mathrm{d}t + B\mathbf{u}\mathrm{d}t + \Gamma \mathrm{d}\mathbf{w}, \tag{12}$$

where the state vector $\mathbf{x}$ includes the system state from the perspective of just one agent, $f(\mathbf{x})$ captures the kinematics, including the deterministic, collision-avoiding reference controls, and $\mathbf{u} = [\omega, a, \omega_1, a_1, \ldots, \omega_M, a_M]^{\mathrm{T}}$ is a vector of optimal feedback controls to be computed by the agent. The Wiener process $\mathrm{d}\mathbf{w}$ captures the uncertainty due to model kinematics for each agent, as well as the uncertainty due to the control executed by neighboring agents $m = 1, \ldots, M$.

We consider two types of cost functionals that aim to bring the agents into formation. In the first, our goal is to compute the feedback controls $\mathbf{u}(\mathbf{x})$ that minimize the total accumulated cost up until the formation is reached (a problem sometimes called control until a target set is reached). The formation is achieved when the inter-agent distances $r_m = \sqrt{(\Delta x_m)^2 + (\Delta y_m)^2}$, $m = 1, \ldots, M$, reach a set of predefined nominal distances δ_m, the agents' heading angles are aligned, and the speeds are equal. Since this is a stochastic problem, we must define the target set $\mathbf{x} \in \mathcal{F}$ as a small ball about the formation. We define the following cost functional:

$$J(\mathbf{x}) = \min_{\mathbf{u}} \mathbb{E}\left\{\int_0^\tau \frac{1}{2}(k(\mathbf{x}) + \mathbf{u}^{\mathrm{T}} R\mathbf{u})\mathrm{d}s\right\}, \tag{13}$$

where $\tau = \inf\{t > 0 : \mathbf{x}(t) \in \mathcal{F}\}$ is a (finite) first exit time, i.e., the first time that the state reaches the formation $\mathcal{F}$. We note that, unlike previous works that either use a receding horizon approach or

[(3)] Ensuring collision avoidance among agents with non-zero collision radii would require a different deterministic control or artificial potential function than is considered in this work.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

fix a final time, here the final time τ is not known in advance. The positive semi-definite matrix R in (13) provides a quadratic control penalty, and the instantaneous state cost $k(\mathbf{x})$,

$$k(\mathbf{x}) = (h(\mathbf{x}) - \boldsymbol{\mu})^{\mathrm{T}} Q (h(\mathbf{x}) - \boldsymbol{\mu}), \tag{14}$$

is a quadratic that reaches minimum value when the inter-agent distances r_m equal the predefined nominal distances δ_m, the heading angles are equal, and the speeds are equal:

$$h(\mathbf{x}) = [r_1, \ldots, r_M, \theta - \theta_1, \ldots, \theta - \theta_M, v - v_1, \ldots, v - v_M]^{\mathrm{T}}, \tag{15}$$

$$\boldsymbol{\mu} = [\delta_1, \ldots, \delta_M, 0, \ldots, 0]^{\mathrm{T}}, \tag{16}$$

and where Q is a diagonal positive definite matrix. The form of the cost function relieves the agents' dependence on a global coordinate frame. Since all couplings between pairs of agents in the cost function are relative, the coordinate frame implied by the Cartesian components Δx_m and Δy_m and angles θ and θ_m in (6)–(11) do not need to be shared among the agents. Further note that this instantaneous state cost (14) reaches minimum value when the potential $V(r_m)$ associated with the reference control also reaches minimum value. However, the reference control also prevents collisions using an infinite potential energy when inter-agent distances approach $r_m = 0$. Since the correction control $\mathbf{u}$ is penalized, it will not overcome this barrier, and collision avoidance is still ensured.

The second type of control problem that we consider is to minimize an infinite horizon cost functional with a discounting factor $\beta > 0$:

$$J(\mathbf{x}) = \min_{\mathbf{u}} \mathbb{E} \left\{ \int_0^\infty \frac{e^{-\beta s}}{2} (k(\mathbf{x}) + \mathbf{u}^{\mathrm{T}} R \mathbf{u}) \mathrm{d}s \right\}, \tag{17}$$

where the terms Q, R, and $k(\mathbf{x})$ are as in (13).

3. Path Integral Representation

In this section we show how the optimal control problem can be represented as a PI over possible system trajectories. The derivation is similar to that used in previous works,[29] but the new types of cost functionals used in this paper warrant a new derivation. We will first consider the cost functional for control until the formation is reached (13), hereinafter abbreviated as CUF.

3.1. Control until formation (CUF)

The (stochastic) HJB equation for model (12) and cost functional (13) is

$$0 = \min_{\mathbf{u}} \left\{ (f + B\mathbf{u})^{\mathrm{T}} \partial_{\mathbf{x}} J + \frac{1}{2} \mathrm{Tr} \left(\Sigma \partial_{\mathbf{x}}^2 J \right) + \frac{1}{2} k(\mathbf{x}) + \frac{1}{2} \mathbf{u}^{\mathrm{T}} R \mathbf{u} \right\}, \tag{18}$$

where $\Sigma = \Gamma \Gamma^{\mathrm{T}}$, and the boundary condition for this PDE is

$$J(\mathbf{x}(\tau)) = 0, \quad \mathbf{x} \in \mathcal{F}. \tag{19}$$

The HJB equation must typically be solved numerically in a discretized state space until a steady state is reached (see Kushner and Dupuis,[52] for example). However, the structure of the problem at hand allows us to avoid discretization through a suitable transformation.

The optimal control $\mathbf{u}(\mathbf{x})$ that minimizes (18) is

$$\mathbf{u}(\mathbf{x}) = -R^{-1} B^{\mathrm{T}} \partial_{\mathbf{x}} J, \tag{20}$$

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

which, when substituted back into the HJB equation, yields:

$$0 = f^{\mathrm{T}}\partial_{\mathbf{x}}J - \frac{1}{2}\left(\partial_{\mathbf{x}}J\right)^{\mathrm{T}} BR^{-1}B^{\mathrm{T}}\partial_{\mathbf{x}}J + \frac{1}{2}\mathrm{Tr}\left(\Sigma\partial_{\mathbf{x}}^2 J\right) + \frac{1}{2}k(\mathbf{x}(t)). \tag{21}$$

Next, we apply a logarithmic transformation[46] $J(\mathbf{x}) = -\lambda \log \Psi(\mathbf{x})$ for constant $\lambda > 0$ to obtain a new PDE

$$\begin{aligned} 0 = \frac{k(\mathbf{x})}{2\lambda} - \frac{f^{\mathrm{T}}}{\Psi}\partial_{\mathbf{x}}\Psi - \frac{1}{2}\frac{1}{\Psi}\mathrm{Tr}\left(\Sigma\partial_{\mathbf{x}}^2\Psi\right) \\ - \frac{1}{2}\frac{\lambda}{\Psi^2}\left(\partial_{\mathbf{x}}\Psi\right)^{\mathrm{T}} BR^{-1}B^{\mathrm{T}}\partial_{\mathbf{x}}\Psi + \frac{1}{2}\frac{1}{\Psi^2}\left(\partial_{\mathbf{x}}\Psi\right)^{\mathrm{T}}\Sigma\partial_{\mathbf{x}}\Psi. \end{aligned} \tag{22}$$

In models (6)–(11), it can be seen that the optimal controls $\mathbf{u}(\mathbf{x})$ act as a correction term to the deterministic controls and the stochastic noise. Penalizing this control (13) suggests that the optimal control is that which is "close" (in terms of the Kullback–Leibler divergence[25]) to the reference control. Moreover, this implies that the possibility of a large stochastic disturbance (either due to neighbors' unknown controls or model kinematics) requires the possibility of a greater control input. Because of this, we assume that the noise in the controlled components is inversely proportional to the control penalty, or

$$\Sigma = \Gamma\Gamma^{\mathrm{T}} = \lambda BR^{-1}B^{\mathrm{T}}. \tag{23}$$

This selects the value of the control penalty that we shall use in the sequel as

$$R = \lambda\,\mathrm{diag}\left(\sigma_\theta^{-2}, \sigma_v^{-2}, \sigma_{\theta,1}^{-2}, \sigma_{v,1}^{-2}, \ldots, \sigma_{\theta,M}^{-2}, \sigma_{v,M}^{-2}\right), \tag{24}$$

and also causes the quadratic terms on the second line of (22) to cancel so that the remaining PDE for Ψ is linear:

$$0 = f^{\mathrm{T}}\partial_{\mathbf{x}}\Psi(\mathbf{x}) + \frac{1}{2}\mathrm{Tr}\left(\Sigma\partial_{\mathbf{x}}^2\Psi\right) - \frac{k(\mathbf{x})}{2\lambda}\Psi(\mathbf{x}), \tag{25}$$

$$\Psi(\mathbf{x}) = 1, \qquad \mathbf{x} \in \mathcal{F}. \tag{26}$$

As before, this could be solved numerically until a steady state is reached. However, the Feynman–Kac equations[20,21] connect certain linear differential operators to adjoint operators that describe the evolution of a *forward* diffusion process beginning from the current state $\widetilde{\mathbf{x}}(0) = \widetilde{\mathbf{x}}_0 = \mathbf{x}$. From the Feynman–Kac equations, the solution to (25) is[19]

$$\Psi(\mathbf{x}) = \mathbb{E}_{\widetilde{\mathbf{x}},\tau|\widetilde{\mathbf{x}}_0}\left\{\Psi(\widetilde{\mathbf{x}}(\tau))\exp\left(-\frac{1}{2\lambda}\int_0^\tau k(\widetilde{\mathbf{x}}(s))ds\right)\right\}, \tag{27}$$

where $\widetilde{\mathbf{x}}(t)$ satisfies the PI-associated, uncontrolled dynamics (cf. (12)),

$$\mathrm{d}\widetilde{\mathbf{x}}(t) = f(\widetilde{\mathbf{x}}(t))\mathrm{d}t + \Gamma\mathrm{d}\mathbf{w}, \tag{28}$$

with initial condition $\widetilde{\mathbf{x}}(0) = \mathbf{x}$, and which, as before, includes the reference control inputs for one agent and its observed neighbors (see Fig. 1). The expectation in (27) is taken with respect to the joint distribution of $(\widetilde{\mathbf{x}}, \tau)$ of sample paths $\widetilde{\mathbf{x}} = \widetilde{\mathbf{x}}(t)$ that begin at $\widetilde{\mathbf{x}}_0 = \mathbf{x}$ and evolve as (28) until hitting the formation $\widetilde{\mathbf{x}}(\tau) \in \mathcal{F}$ at time τ. Unlike previous PI works, where the terminal time is fixed and known in advance, this stopping time is a property of the set of stochastic trajectories $\widetilde{\mathbf{x}}(t)$.

The distribution $(\widetilde{\mathbf{x}}, \tau)$ is difficult to obtain. Monte Carlo techniques may be used to sample trajectories $\widetilde{\mathbf{x}}$, but hitting the formation is a rare event unless there is a mechanism to "guide" the

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

trajectory into the formation. In this work, we determine the trajectory $\widetilde{\mathbf{x}}|\tau, \mathbf{x}_0$ conditioned on its hitting time. From the law of total expectation,

$$\Psi(\mathbf{x}) = \mathbb{E}_{\tau|\mathbf{x}_0}\left\{\mathbb{E}_{\widetilde{\mathbf{x}}|\tau,\widetilde{\mathbf{x}}_0}\left[\Psi(\widetilde{\mathbf{x}}(\tau))\exp\left(-\frac{1}{2\lambda}\int_0^{\tau} k(\widetilde{\mathbf{x}}(s))ds\right)\right]\right\} \tag{29}$$

$$= \mathbb{E}_{\tau|\mathbf{x}_0}\left\{\Psi(\mathbf{x}|\mathbf{x}_0, \tau)\right]\right\}. \tag{30}$$

In practice, we find that the inner distribution $\Psi(\mathbf{x}|\mathbf{x}_0, \tau)$ exhibits small tails for most τ and has high probability for just a small range of τ. Moreover, the range of τ with higher likelihood $\Psi(\mathbf{x}|\mathbf{x}_0, \tau)$ is the one that appears to equally balance state and control costs. Therefore, we consider a discrete set $(\tau_1, \ldots, \tau_{N_\tau})$ of N_τ possible values for τ with non-informative, uniform prior probabilities. This implies that the distribution of the hitting times is implicitly encoded in the length (and the ensuing cost) of the path $\widetilde{\mathbf{x}}|\tau_i, \mathbf{x}_0$. Since $\Psi(\mathbf{x}(\tau_i)) = 1$ from (26), the solution (29) can be expanded as follows:

$$\Psi(\widetilde{\mathbf{x}}) = \frac{1}{N_\tau}\sum_{i=1}^{N_\tau}\mathbb{E}_{\widetilde{\mathbf{x}}|\widetilde{\mathbf{x}}_0,\tau_i}\left\{\Psi(\widetilde{\mathbf{x}}(\tau_i))\exp\left(-\frac{1}{2\lambda}\int_0^{\tau_i} k(\widetilde{\mathbf{x}}(s))ds\right)\right\} \tag{31}$$

$$= \frac{1}{N_\tau}\sum_{i=1}^{N_\tau}\mathbb{E}_{\widetilde{\mathbf{x}}|\widetilde{\mathbf{x}}_0,\tau_i}\left\{\exp\left(-\frac{1}{2\lambda}\int_0^{\tau_i} k(\widetilde{\mathbf{x}}(s))ds\right)\right\}. \tag{32}$$

By discretizing the interval $[0, \tau_i]$ into N_i intervals of equal length Δt, $t_0 < t_1 < \ldots < t_{N_i} = \tau_i$, we can consider a sample of the discretized trajectory $\widetilde{\mathbf{x}}^N|\mathbf{x}_0, \tau_i = (\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_{N_i})$. Under this discretization in time, solution (32) with a right-hand Riemann sum approximation to the integral can be written as

$$\Psi(\widetilde{\mathbf{x}}) = \frac{1}{N_\tau}\lim_{\Delta t\to 0}\sum_{i=1}^{N_\tau}\int \mathrm{d}\widetilde{\mathbf{x}}^N P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)\exp\left[-\frac{\Delta t}{2\lambda}\sum_{k=1}^{N_i} k(\widetilde{\mathbf{x}}_k)\right], \tag{33}$$

where $\mathrm{d}\widetilde{\mathbf{x}}^N = \prod_{k=1}^{N_i}\mathrm{d}\widetilde{\mathbf{x}}_k$, and where $P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)$ is the probability of a discretized sample path, conditioned on the starting state $\widetilde{\mathbf{x}}_0$ and hitting time τ_i, given by

$$P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) = \prod_{k=0}^{N_i-1} p(\widetilde{\mathbf{x}}_{k+1}|\widetilde{\mathbf{x}}_k, \tau_i). \tag{34}$$

Since the uncontrolled process (28) is driven by Gaussian noise with zero mean and covariance $\Sigma = \Gamma\Gamma^{\mathrm{T}}$, the transition probabilities may be written as

$$p(\widetilde{\mathbf{x}}_{k+1}|\widetilde{\mathbf{x}}_k, \tau_i) \propto \exp\left(-\frac{1}{2}\left(\widetilde{\mathbf{x}}_{k+1} - \widetilde{\mathbf{x}}_k - f(\widetilde{\mathbf{x}}_k)\,\Delta t\right)^{\mathrm{T}}\right.$$
$$\left.\times\left(\Delta t\lambda B R^{-1} B^{\mathrm{T}}\right)^{-1}\left(\widetilde{\mathbf{x}}_{k+1} - \widetilde{\mathbf{x}}_k - f\left(\widetilde{\mathbf{x}}_k\right)\Delta t\right)\right) \tag{35}$$

for $k < N_i - 1$, and $p(\widetilde{\mathbf{x}}_{k+1}|\widetilde{\mathbf{x}}_k, \tau_i) = \mathbb{1}_{h(\widetilde{\mathbf{x}}_{k+1})=\boldsymbol{\mu}}(\widetilde{\mathbf{x}}_{k+1})$ for $k = N_i - 1$.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

The PI representation of $\Psi(\widetilde{\mathbf{x}})$ is obtained from Eqs. (33)–(35), and can be written as an exponential of an "action"[47] $S(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)$ along the time-discretized sample trajectories $(\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_N)$:

$$\Psi(\widetilde{\mathbf{x}}) \propto \frac{1}{N_\tau} \lim_{\Delta t \to 0} \sum_{i=1}^{N\tau} \int d\widetilde{\mathbf{x}}^N \exp(-S(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)) \tag{36}$$

$$S(\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_N|\widetilde{\mathbf{x}}_0, \tau_i) = \sum_{k=1}^{N_i} \frac{\Delta t}{2\lambda} k(\widetilde{\mathbf{x}}_k) + \sum_{k=0}^{N_i-1} \frac{1}{2} (\widetilde{\mathbf{x}}_{k+1} - \widetilde{\mathbf{x}}_k - \Delta t f(\widetilde{\mathbf{x}}_k))^{\mathrm{T}} \times (\lambda \Delta t B R^{-1} B^{\mathrm{T}})^{-1} (\widetilde{\mathbf{x}}_{k+1} - \widetilde{\mathbf{x}}_k - \Delta t f(\widetilde{\mathbf{x}}_k)) . \tag{37}$$

Differentiating (36) with respect to $\widetilde{\mathbf{x}}_0$, we can obtain the optimal control (20) as[29]

$$\begin{aligned} \mathbf{u}(\widetilde{\mathbf{x}}) &= \lim_{\Delta t \to 0} \lambda R^{-1} B^{\mathrm{T}} \partial_{\widetilde{\mathbf{x}}} \log \Psi \\ &= \lim_{\Delta t \to 0} \frac{1}{N_\tau} \sum_{i=1}^{N_\tau} \int d\widetilde{\mathbf{x}}^N P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) \mathbf{u}_L(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) \qquad (38) \\ &= \lim_{\Delta t \to 0} \frac{1}{N_\tau} \sum_{i=1}^{N_\tau} \mathbb{E}_{P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)} \left\{ \mathbf{u}_L(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) \right\}, \qquad (39) \end{aligned}$$

where $\lim_{\Delta t \to 0} P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) = P(\widetilde{\mathbf{x}}|\widetilde{\mathbf{x}}_0, \tau_i)$ is the probability of an *optimal* trajectory conditioned to hit the formation at time τ_i,

$$P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) \propto e^{-S(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)}, \tag{40}$$

which weights the local controls $\mathbf{u}_L(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i)$ in (38), defined by

$$\mathbf{u}_L(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau_i) = R^{-1} B^T (B R^{-1} B^T)^{-1} \left(\frac{\widetilde{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_0}{\Delta t} - f(\widetilde{\mathbf{x}}_0) \right). \tag{41}$$

Although the resulting control law is stationary, the state space is too large for it to be computed off-line. Because of this, after computing $\mathbf{u}(\mathbf{x}) = \mathbf{u}(\widetilde{\mathbf{x}}_0)$, each agent executes only the first increment of that control, at which point the optimal control is recomputed. Then (39) is

$$\begin{aligned} \mathbf{u}(\mathbf{x}) &= R^{-1} B^T (B R^{-1} B^T)^{-1} \left(\frac{\mathbb{E}_{P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0)} \{\widetilde{\mathbf{x}}_1\} - \mathbf{x}}{\Delta t} - f(\mathbf{x}) \right) \\ &= R^{-1} B^T (B R^{-1} B^T)^{-1} \left(\frac{\mathbb{E}_\tau \mathbb{E}_{P(\widetilde{\mathbf{x}}^N|\widetilde{\mathbf{x}}_0, \tau)} \{\widetilde{\mathbf{x}}_1\} - \mathbf{x}}{\Delta t} - f(\mathbf{x}) \right). \end{aligned} \tag{42}$$

In other words, control (42) applied by an agent in state $\mathbf{x}$ is constructed from a realization of the unknown or random dynamics of the system that maximizes the probability of the trajectory that starts from $\mathbf{x}$ and evolves until hitting the formation. This probability is weighted by the cost accumulated along the path.

3.2. Discounted cost infinite-horizon control

For the discounted infinite-horizon costs, we will develop the derivation separately while showing the relation to the CUF problem. We begin by writing the cost functional (17) in terms of a finite-horizon

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

cost functional and an error term ε:

$$J(\mathbf{x}, t) = \min_{\mathbf{u}} \mathbb{E}\left\{\int_0^T \frac{e^{-\beta s}}{2}\left(k(\mathbf{x}) + \mathbf{u}^{\mathrm{T}} R \mathbf{u}\right) ds\right\} + \varepsilon$$
$$= \min_{\mathbf{u}} \mathbb{E}\left\{\int_0^T \frac{1}{2}\left((h(\mathbf{x}) - \boldsymbol{\mu})^{\mathrm{T}} Q_t (h(\mathbf{x}) - \boldsymbol{\mu}) + \mathbf{u}^{\mathrm{T}} R_t \mathbf{u}\right) ds\right\} + \varepsilon, \tag{43}$$

where it is assumed that $T > \overline{T}$ is sufficiently large so that ε may be neglected, and where $Q_t = e^{-\beta t} Q$ and $R_t = e^{-\beta t} R$. The feedback control $\mathbf{u}$ that minimizes (43) will be used in a receding horizon manner over the horizon $[0, T]$, i.e., $\mathbf{u}(\mathbf{x}) = \mathbf{u}(\mathbf{x}(0))$.

The (stochastic) HJB equation for the system kinematics (12) and cost functional (43), assuming $\varepsilon = 0$, is

$$0 = \partial_t J + \min_{\mathbf{u}} \left((f + B\mathbf{u})^{\mathrm{T}} \partial_{\mathbf{x}} J + \frac{1}{2} \mathrm{Tr}\left(\Sigma \partial_{\mathbf{x}}^2 J\right) \right.$$
$$\left. + \frac{e^{-\beta t}}{2} k(\mathbf{x}(t)) + \frac{1}{2} \mathbf{u}(\mathbf{x})^{\mathrm{T}} R_t \mathbf{u}(\mathbf{x}) \right). \tag{44}$$

Substituting the optimal control $\mathbf{u}(\mathbf{x}, t) = -R_t^{-1} B^{\mathrm{T}} \partial_{\mathbf{x}} J(\mathbf{x}, t)$ and applying the transformation $J(\mathbf{x}, t) = -\lambda \log \Psi(\mathbf{x}, t)$, we obtain

$$\frac{1}{\Psi} \partial_t \Psi = e^{-\beta t} \frac{k(\mathbf{x})}{2\lambda} - \frac{f^{\mathrm{T}}}{\Psi} \partial_{\mathbf{x}} \Psi - \frac{1}{2} \frac{1}{\Psi} \mathrm{Tr}\left(\Sigma \partial_{\mathbf{x}}^2 \Psi\right)$$
$$- \frac{1}{2} \frac{\lambda}{\Psi^2} \left(\partial_{\mathbf{x}} \Psi\right)^{\mathrm{T}} B R_t^{-1} B^{\mathrm{T}} \partial_{\mathbf{x}} \Psi + \frac{1}{2} \frac{1}{\Psi^2} \left(\partial_{\mathbf{x}} \Psi\right)^{\mathrm{T}} \Sigma_t \partial_{\mathbf{x}} \Psi. \tag{45}$$

In order for the nonlinear terms to cancel as in (22), we use assumption (23), but with time dependence due to the discounting factor included,

$$\Sigma_t = e^{\beta t} \Gamma \Gamma^{\mathrm{T}} = \lambda B R_t^{-1} B^{\mathrm{T}}. \tag{46}$$

Note that in a feedback setting, the control $\mathbf{u}(\mathbf{x})$ is based on the state $\mathbf{x}(0)$, and Σ_0 reduces to $\lambda B R^{-1} B^{\mathrm{T}}$. The remaining PDE,

$$\partial_t \Psi = \left(e^{-\beta t} \frac{k(\mathbf{x})}{2\lambda} - f^{\mathrm{T}} \partial_{\mathbf{x}} - \frac{1}{2} \mathrm{Tr}\left(\Sigma_t \partial_{\mathbf{x}}^2\right)\right) \Psi, \tag{47}$$

is linear and has solution

$$\Psi(\widetilde{\mathbf{x}}_0, 0) = \mathbb{E}_{\widetilde{\mathbf{x}}|\widetilde{\mathbf{x}}_0} \left\{ \exp\left(-\frac{1}{2\lambda} \int_0^T e^{-\beta s} k(\mathbf{x}(s)) ds \right) \right\}. \tag{48}$$

The reader will note that this solution also appears in the case of the CUF problem in (32), but with a few changes. First, only one horizon T is considered, and so the summation over τ_i is removed. Next, the discounting factor β is included inside the expectation. Finally, the uncontrolled process $\widetilde{\mathbf{x}}$ is (28) but with an increasing noise intensity due to (46):

$$\mathrm{d}\widetilde{\mathbf{x}}(t) = f(\widetilde{\mathbf{x}}(t))\mathrm{d}t + e^{\frac{\beta t}{2}} \Gamma \mathrm{d}\mathbf{w}. \tag{49}$$

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

The PI representation of $\Psi(\widetilde{\mathbf{x}}_0, 0)$ corresponding to the discounted infinite-horizon problem is obtained in the same manner as (36)–(37), and is

$$\Psi(\widetilde{\mathbf{x}}_0, 0) \propto \lim_{\Delta t \to 0} \int \mathrm{d}\widetilde{\mathbf{x}}^N \exp(-S(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0)) \tag{50}$$

$$\begin{aligned} S(\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_N | \widetilde{\mathbf{x}}_0) &= \sum_{k=1}^{N} \frac{e^{-\beta t_k} \Delta t}{2\lambda} k(\widetilde{\mathbf{x}}_k) \\ &+ \sum_{k=0}^{N-1} \frac{1}{2} (\widetilde{\mathbf{x}}_{k+1} - \widetilde{\mathbf{x}}_k - \Delta t f(\widetilde{\mathbf{x}}_k))^{\mathrm{T}} \\ &\times (\lambda \Delta t B R^{-1} B^{\mathrm{T}} e^{\beta t_k})^{-1} (\widetilde{\mathbf{x}}_{k+1} - \widetilde{\mathbf{x}}_k - \Delta t f(\widetilde{\mathbf{x}}_k)) . \end{aligned} \tag{51}$$

Along similar lines, the control may be computed as (cf. (42))

$$\mathbf{u}(\mathbf{x}) = R^{-1} B^T (B R^{-1} B^T)^{-1} \left(\frac{\mathbb{E}_{P(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0)} \{\widetilde{\mathbf{x}}_1\} - \mathbf{x}}{\Delta t} - f(\mathbf{x}) \right), \tag{52}$$

where the path probability is

$$P(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0) \propto e^{-S(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0)}. \tag{53}$$

Both for the CUF problem and the infinite horizon problem, the matrix coefficients multiplying (42) and (52) may be dropped since control is only affecting the noisy states.[4] One may compute the optimal control (42) (resp. (52)) once the path probability $P(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0, \tau_i)$ (40) (resp. (53)) has been computed, a nontrivial task to be discussed in the following section.

4. Computing the Control with Kalman Smoothers

In this section we present our approach to compute the control in (42) and (52). Although Monte Carlo techniques can be used to generate samples of the maximally likely trajectory $P(\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_N | \widetilde{\mathbf{x}}_0)$, we find them to be slow in practice due to the high dimension of this problem ($\widetilde{\mathbf{x}}^N \in \mathbb{R}^{4NM}$). Moreover, in the case of CUF, when sampling a trajectory $\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0, \tau_i$, the trajectory must be conditioned to hit the formation at τ_i. Finally, it is not necessary to sample the entire distribution $P(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0)$ since only the estimate $\widehat{\mathbf{x}}_1 \equiv \mathbb{E}_{P(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0)} \{\widetilde{\mathbf{x}}_1\}$ is needed.

Therefore, in this work we treat the temporal discretization of the optimal trajectory $\widetilde{\mathbf{x}}^N$ as the hidden state of a stochastic process where appropriately chosen measurements of this hidden state are related to the system goal $\boldsymbol{\mu}$ (16). The optimal control can then be computed from the optimal estimate $\widehat{\mathbf{x}}_1$, given the process and measurements over a fixed interval $t_1, \ldots, \tau_i$ for the CUF problem and $t_1, \ldots, T$ for the infinite-horizon problem. We define the following nonlinear smoothing problem.

Nonlinear smoothing problem: Given measurements $\mathbf{y}_k = \mathbf{y}(t_k)$ for $t_k = t_1, \ldots, t_N = \tau_i$ (or T), where $t_{k+1} - t_k = \Delta t$, compute the estimate $\widehat{\mathbf{x}}_{1:N}$ of the hidden state $\widetilde{\mathbf{x}}_{1:N}$ from the nonlinear hidden state-space model:

$$\widetilde{\mathbf{x}}_{k+1} = \widetilde{\mathbf{x}}_k + \Delta t f(\widetilde{\mathbf{x}}_k) + \varepsilon_k, \tag{54}$$

$$\mathbf{y}_k = h(\widetilde{\mathbf{x}}_k) + \eta_k, \tag{55}$$

[4] The reader may examine the form of $B\mathbf{u}$, taking into account that the components of the final multiplicative factor in (42) or (52) are zero for uncontrolled states.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

where $f(\cdot)$ and $h(\cdot)$ are as in Section 2, and ε_k and η_k are independent multivariate Gaussian random variables with zero mean and with covariances chosen as follows:

CUF:

$$\mathbb{E}\left(\varepsilon_k \varepsilon_k^{\mathrm{T}}\right) = \lambda \Delta t B R^{-1} B^{\mathrm{T}}, \tag{56}$$

$$\mathbb{E}\left(\eta_k \eta_k^{\mathrm{T}}\right) = \begin{cases} \frac{\lambda}{\Delta t} Q^{-1} & k = 1, \ldots, N-1 \\ 0 & k = N \end{cases}. \tag{57}$$

Discounted infinite horizon:

$$\mathbb{E}\left(\varepsilon_k \varepsilon_k^{\mathrm{T}}\right) = \lambda \Delta t B R^{-1} B^{\mathrm{T}} e^{\beta t_k}, \tag{58}$$

$$\mathbb{E}\left(\eta_k \eta_k^{\mathrm{T}}\right) = \frac{\lambda}{\Delta t} Q^{-1} e^{\beta t_k}. \tag{59}$$

The smoothing is initialized from $\widetilde{\mathbf{x}}_0 = \mathbf{x}$, the current state of the system as viewed by the agent. Measurements $\mathbf{y}_k$ are always exactly $\mathbf{y}_k = \boldsymbol{\mu}$. □

Note that the measurement and process covariances have the same structure for both cost function types, but the noise intensity increases during the smoothing horizon in the discounted infinite horizon case. As the covariance becomes large, the measurements and state predictions contain so little information that the change in the estimated hidden state (the trajectory) is negligible, leading to a stationary control policy.

To show the relation between the nonlinear smoothing problem and the stochastic optimal control problem, we write the probability of a hidden state sequence $\widetilde{\mathbf{x}}^N$ given measurements $\mathbf{y}_k$ and the initial state $\widetilde{\mathbf{x}}_0$, which is[48]

$$P(\widetilde{\mathbf{x}}^N | \widetilde{\mathbf{x}}_0, \mathbf{y}_1, \ldots, \mathbf{y}_N) \propto \prod_{k=1}^{N} p(\mathbf{y}_k | \widetilde{\mathbf{x}}_k) p(\widetilde{\mathbf{x}}_k | \widetilde{\mathbf{x}}_{k-1}), \tag{60}$$

where, in the case of the discounted infinite-horizon cost function,[(5)]

$$\begin{aligned} p(\mathbf{y}_k | \widetilde{\mathbf{x}}_k) \equiv p(\boldsymbol{\mu}_k | \widetilde{\mathbf{x}}_k) &= N\left(h(\widetilde{\mathbf{x}}_k), \eta_k \eta_k^{\mathrm{T}}\right) \\ &\propto \exp\left\{-\frac{\Delta t}{2\lambda}(h(\widetilde{\mathbf{x}}_k) - \boldsymbol{\mu})^{\mathrm{T}} Q e^{-\beta t}(h(\widetilde{\mathbf{x}}_k) - \boldsymbol{\mu})\right\}, \end{aligned} \tag{61}$$

$$\begin{aligned} p(\widetilde{\mathbf{x}}_k | \widetilde{\mathbf{x}}_{k-1}) &= N\left(\widetilde{\mathbf{x}}_{k-1} + \Delta t f(\widetilde{\mathbf{x}}_{k-1}), \Delta t \Sigma\right) \\ &\propto \exp\left\{-\frac{1}{2}\left(\widetilde{\mathbf{x}}_k - \widetilde{\mathbf{x}}_{k-1} - \Delta t f(\widetilde{\mathbf{x}}_{k-1})\right)^{\mathrm{T}}\right. \\ &\quad \left.\times \left(\lambda \Delta t B R^{-1} B^{\mathrm{T}} e^{\beta t}\right)^{-1} \left(\widetilde{\mathbf{x}}_k - \widetilde{\mathbf{x}}_{k-1} - \Delta t f(\widetilde{\mathbf{x}}_{k-1})\right)\right\}. \end{aligned} \tag{62}$$

Comparing the right-hand sides of (61) and (62) with (50) and (51) (resp. (36) and (37)), it can be seen that these are identical to those in the stochastic optimal control problem.

Since the optimal control (42) is based on the probability of a full trajectory of fixed length and values $\boldsymbol{\mu}$ are available in advance, the expected value of the trajectory originating from state $\widehat{\mathbf{x}}_0$ conditioned to hit the formation at time τ_i or T, that is, the hidden states $\widehat{\mathbf{x}}_k$, $k = 1, \ldots, N$, can be found by filtering and then smoothing the process, given the values $\boldsymbol{\mu}_k$ using a nonlinear fixed-interval Kalman smoother. Such an algorithm assumes that the increments given by (61) and (62) are

[(5)] Set $\beta = 0$ for the CUF case.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

Gaussian[6] to some extent, but the algorithm is sufficiently fast to be applied in *real-time* by each unicycle in a potentially large group with an even larger state space, motivating its use in this work.

In the case of the first-hitting time problem, the estimated trajectory is first computed for each τ_i, the point at which the expectation over τ_i may be computed. This results in an average of the controls $u_L(\widetilde{\mathbf{x}}|\widetilde{\mathbf{x}}_0, \tau_i)$ to be applied. In other words, each agent estimates both the optimal system trajectory (from its perspective) given the time the formation will hit *and* the hitting time of the formation. Hitting the target sooner would save on state costs, but may cause an increase in control costs and *vice versa*.

When the smoothing is complete and agents have applied their computed control, each agent must then observe the actual states of its neighbors so that the next iteration begins with the correct initial condition. We provide a pseudo code for our computations in Algorithm 1 for the CUF case and in Algorithm 2 for the infinite horizon case. For clarification, we also provide a flowchart in Fig. 2.

Algorithm 1 Formation control algorithm applied by each agent: control to formation case

$\mathbf{x}(t) \leftarrow$ measured state of system from agent's viewpoint
$\boldsymbol{\mu} \leftarrow$ nominal distances
while $\mathbf{x}(t) \notin \mathcal{F}$ **do**
  for $i = 1, \ldots, N_\tau$ **do**
    $\mathbb{E}\{\widetilde{\mathbf{x}}_1\}, \ldots, \mathbb{E}\{\widetilde{\mathbf{x}}_N\} \leftarrow$ KalmanSmoother (initial state $= \mathbf{x}_0 = \mathbf{x}$, horizon $= [0, \tau_i]$, predictions from (28) or (49), measurements $= \boldsymbol{\mu}$)
    $\mathbb{E}\left\{\mathbf{u}_L(\widetilde{\mathbf{x}}^N|\mathbf{x}_0, \tau_i)\right\} \leftarrow (\mathbb{E}(\widetilde{\mathbf{x}}_1) - \mathbf{x})/\Delta t - f(\mathbf{x})$ ▷ from (41)
  end for
  $\mathbf{u}(\mathbf{x}) = \frac{1}{N_\tau}\sum_{i=1}^{N_\tau} \mathbb{E}\left\{\mathbf{u}_L(\widetilde{\mathbf{x}}^N|\mathbf{x}_0, \tau_i)\right\}$
  Apply computed corrective control $\mathbf{u}(\mathbf{x})$ ▷ using (12)
  $\mathbf{x}(t) \leftarrow$ measured state of system from agent's viewpoint
end while

Algorithm 2 Formation control algorithm applied by each agent: infinite horizon case

$\mathbf{x}(t) \leftarrow$ measured state of system from agent's viewpoint
$\boldsymbol{\mu} \leftarrow$ nominal distances
loop
  $\mathbb{E}\{\widetilde{\mathbf{x}}_1\}, \ldots, \mathbb{E}\{\widetilde{\mathbf{x}}_N\} \leftarrow$ KalmanSmoother (initial state $= \mathbf{x}_0 = \mathbf{x}$, horizon $= [0, T]$, predictions from (28) or (49), measurements $= \boldsymbol{\mu}$)
  $\mathbf{u}(\mathbf{x}) = \mathbb{E}\left\{\mathbf{u}_L(\widetilde{\mathbf{x}}^N|\mathbf{x}_0)\right\} \leftarrow (\mathbb{E}(\widetilde{\mathbf{x}}_1) - \mathbf{x})/\Delta t - f(\mathbf{x})$ ▷ from (52)
  Apply computed corrective control $\mathbf{u}(\mathbf{x})$ ▷ using (12)
  $\mathbf{x}(t) \leftarrow$ measured state of system from agent's viewpoint
end loop

In practice, the controller/smoother must be capable of efficiently smoothing over the horizon $[t_0, \tau_i]$ (or $[t_0, T]$). The computational complexity (see Särkkä[50] for Matlab toolbox) of the smoother used in this work[49] roughly scales with the number of neighbors as M^3. As such, for implementation on mobile robotic controllers, the computational requirements may prescribe a two-level approach – a high-level nonlinear fixed-interval Kalman smoothing algorithm that computes reference inputs for lower-level robot control loops. However, various implementations of the smoothing algorithm can streamline the computation. For instance, a sigma point-based approximation to the two-filter smoothing equations[51] could produce reference control values using only twice the computing power of an unscented filtering algorithm. Moreover, through linearization of the measurement model (55) or of both the process and the measurement models (54) and (55), the control could be computed using extended Kalman smoother or the linear unscented Kalman smoother, respectively. However, through full linearization, we would effectively only be solving an LQG control problem, as mentioned in Section 1.

[6] See Long *et al.*[16] for a discussion on the Gaussianity of these increments.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

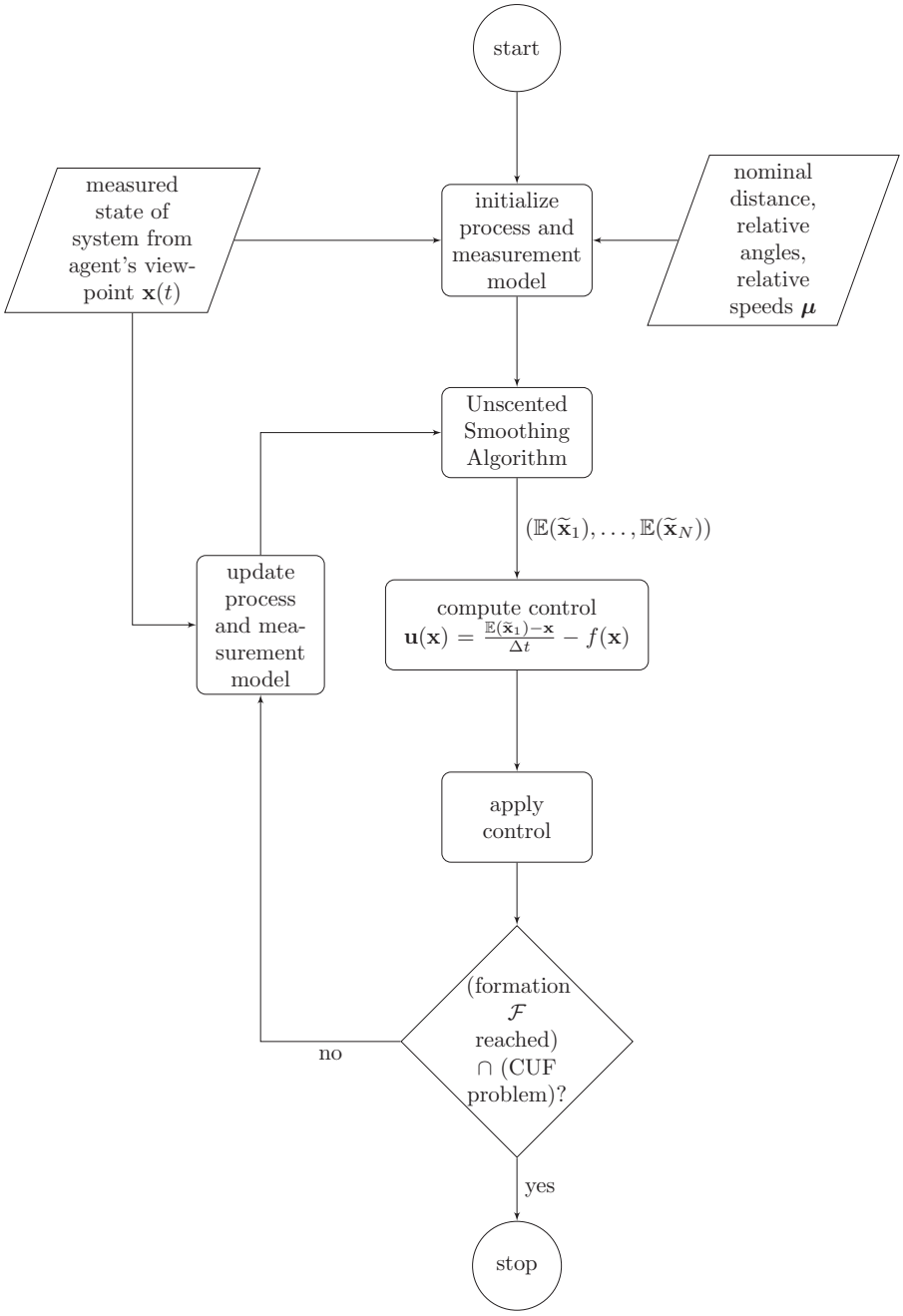

Fig. 2. Flowchart for formation control algorithm.

5. Results

In this section we compare the control computed by our Kalman smoothing algorithm with that computed by a numerical stochastic optimal control method,[52] and then provide simulations in which five agents achieve formations with equal and aligned velocities. The system and algorithm parameters were chosen as $\lambda = 100$, $\sigma_\theta = \sigma_{\theta,m} = 0.1$ (see (24) for R), $Q = 100I$, and $\Delta t = 0.1$, with all units relative to meters and seconds. In the case of CUF, $N_\tau = 10$, $\tau = 1, \ldots, 10$, $\epsilon_r = \epsilon_v = 0.1$, $\epsilon_\theta = 10^\circ$. For the infinite-horizon case, $\beta = 0.5$ and $T = 10$. With these parameters, $\exp(-\beta T) \approx 0.007$, and so the cost function (43) used for the algorithm is a good approximation to a discounted infinite-horizon cost function (17). In each case, the control was computed using a Discrete-time Unscented Kalman Rauch–Tung–Striebel Smoother[50] (see Särkkä[49] for Matlab toolbox).

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

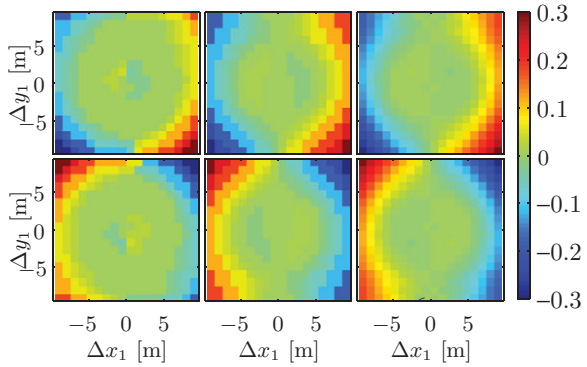

Fig. 3. (Colour online) Optimal feedback control $\mathbf{u}(\Delta x_1, \Delta y_1, \theta_1, \theta_2)$ based on a discounted infinite horizon for two agents with fixed speed $v = 1$, evaluated at $\theta_1 = \theta_2 = 0$. The top row is the control ω and the bottom is ω_1 (i.e., the control assumed for a neighbor $m = 1$). (Left column) Control based on Markov chain approximation with Δx step size $= \Delta y$ step size $= 1.05$, θ_1 step size $= \theta_2$ step size $= 0.078$, and control $\mathbf{u}$ step size 0.85 for both ω and ω_1. (Right column) Control evaluated using a Kalman smoother on the same grid, but without control discretization. (Center column) To compare, Kalman smoother control down-sampled to the same control step sizes 0.85 for ω and ω_1 that are used in the left column's control computation.

We first apply the Markov chain approximation method[52] to compute the feedback control from the HJB equation for cost function (17) so that we may compare the computation time and computed controls with our proposed method. In order to reduce the dimension of the state space for the Markov chain approximation method, which requires discretization of the state space, we consider the problem of just two agents, fix $v_1 = v_2 = 1$ m/s, and aim for a nominal distance between agents of $\delta_1 = 5$ m. Then the controls $\omega(\Delta x_1, \Delta y_1, \theta, \theta_1)$ and $\omega_1(\Delta x_1, \Delta y_1, \theta, \theta_1)$ may be computed, although this is computationally intensive due to the discretization of the state space ($\mathbf{x} \in \mathbb{R}^4$) and the control variable $\mathbf{u} \in \mathbb{R}^2$, leading to 25,600 grid cells and 225 control possibilities for the chosen discretization. The left column of Fig. 3 shows the optimal controls as computed by the Markov chain approximation method, which, for the chosen parameters, required 3.7 h of computation time in Matlab® on an Intel i7 at 2.7 GHz with 8-GB RAM (0.51 s per grid cell visible in Fig. 3). To compare, the Kalman smoother algorithm described in the previous section was applied to the same grid locations in state space, but without the control discretization. The Kalman smoother-based method required only 37.26 s of computation time, an average of 0.09 s per grid cell, although it is not actually necessary to compute the control for the entire state space grid.

Figure 3 serves two purposes. First, it is seen that the controls computed by our method are similar to those from the Markov chain approximation method, although with such an aggressive discretization required by the latter, there are no guarantee that the control solutions should be identical. Moreover, recalling that the displayed controls are additive corrections to the non-optimal feedback controls, this figure allows us to examine the regions in state space where the non-optimal feedback control is most deficient in terms of optimality and robustness to uncertainty. For example, Fig. 3 shows that little correction control is needed in regions $r \lesssim 5$ (i.e., $\mathbf{u} \approx 0$). However, as agents become further separated, the reference feedback controls are not sufficiently strong. Of course, the described corrections are only for the case $\theta_1 = \theta_2 = 0$ and with fixed speed, and we do not attempt in this paper to analytically improve the reference control nor its underlying artificial potential function. However, we envision our method being useful for control designers to analyze a "slice" or marginalization of their feedback control function without having to solve the HJB PDE in the entire state space.

Next, the Kalman smoothing method is employed so that five agents achieve the formation of a regular pentagon, where each agent is individually estimating the hidden optimal trajectory based on the relative kinematics of all its neighbors. The agents observe all others, but, as described in Section 2, only the inter-agent connections *known* by an agent are used when computing the control. The instantaneous state cost (14) penalizes the mean squared distance from the unicycle to all of its $M = 4$ neighbors in excess of the side length of the pentagon (5 m) or the diagonal of the pentagon, depending on the relative configuration of the pentagon encoded in δ_m, $m = 1, \ldots, 4$. The system

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

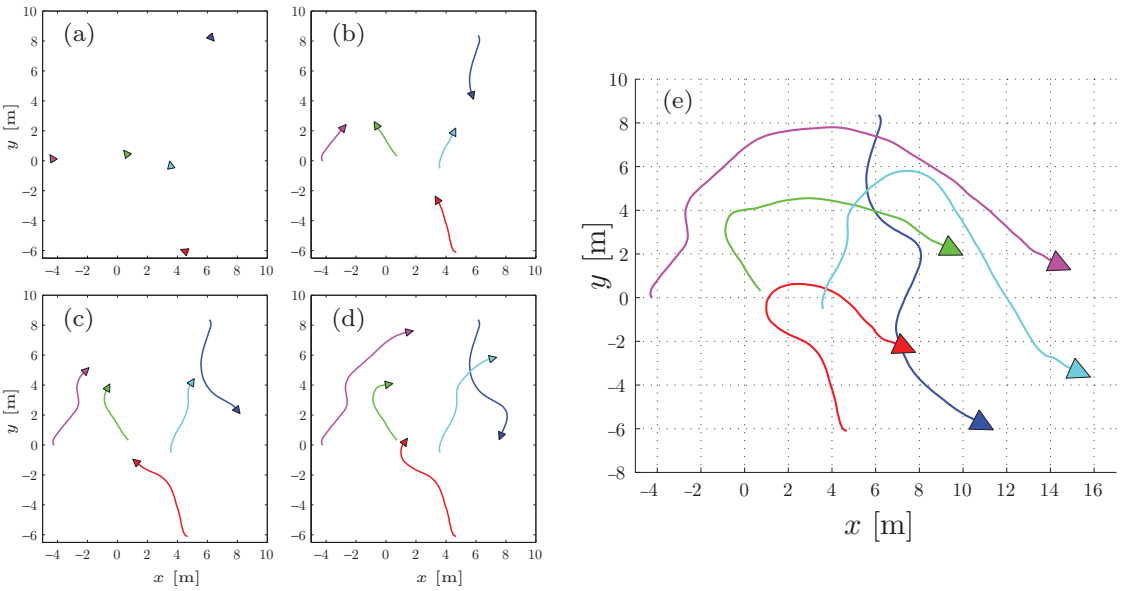

Fig. 4. (Colour online) Five agents, starting from random initial positions and a common speed $v = 2.5$ m/s, achieve a regular pentagon formation by an individually optimal choice of acceleration and turning rate, without any active communication. Frames (a) through (e) correspond to the times, $t = 0$ s, 2 s, 4 s, 6 s, and 16.4 s. An example of collision avoidance between the two upper-left agents is seen in frames (b) and (c).

parameters are the same as the previous example, with the addition of $\sigma_v = \sigma_{v,m} = 0.1$ (see (24) for R).

For the case of CUF, we define the formation, i.e., stopping condition, as

$$\begin{aligned}\mathcal{F} = \big\{\mathbf{x} : &|r_m - \delta_m| \le \epsilon_r, \\ &|\theta - \theta_m| \le \epsilon_\theta, \\ &|v - v_m| \le \epsilon_v, \quad m = 1, \ldots, M\big\}\end{aligned} \tag{63}$$

based on tolerances ϵ_r, ϵ_θ, and ϵ_v. Figure 4 shows the trajectories of all the agents, while the inter-agent distances and agents' angles and speeds can be seen in Fig. 5. The actual stopping time was $\tau = 16.1$ s, and the agents did not collide. In the last second of the simulation, a minor deviation in the agents' heading angles and speeds was seen. This was due to a final correction in agents' relative distances that was needed after having converged in heading angle and speed. Without the addition of the optimal controls, the collision-avoiding controls acting alone did not lead to a formation (63) within the first 60 s of the simulation (Fig. 5). Although the non-optimal control aligned heading angles and, to some extent, speeds, they led to oscillatory trajectories, which is not uncommon in the artificial potential function approach.

The infinite-horizon case may be seen in Fig. 6, and its corresponding cost is in Fig. 7. The agents achieve and maintain a pentagon using the computed optimal controls, while a simulation using only the reference feedback controls again leads to oscillations, as seen in Fig. 7.

As discussed in Section 1, the implementation of the Kalman smoothing algorithm is in some ways similar to receding horizon control/MPC. To compare our results with this strategy, we next consider an Euler discretization of system (12) with noise intensity $\Gamma = 0$. At each time step, each agent must compute the controls $(\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{N-1})$ that minimize the total accumulated cost (43) over the horizon, at which point the first control $\mathbf{u}_0$ is executed, and then the process repeats. Using the same time step (0.1 s) that was employed with the Kalman smoothers was computationally prohibitive, and so we chose $\Delta t = 0.5$ s, resulting in 200 decision variables. We computed the control using IPOPT,[53] which required approximately 30 min for each time step. Figure 8(a) shows that the agents under MPC control converge to a pentagon that moves in a different direction than the Kalman

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

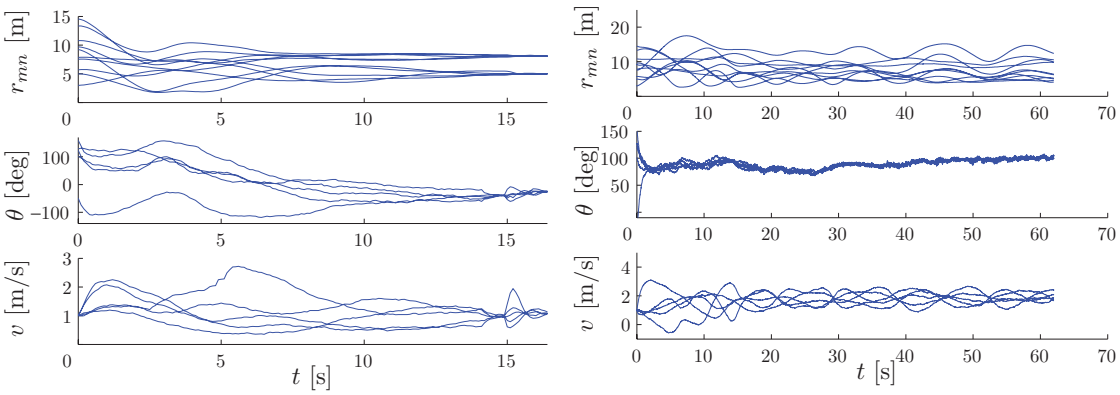

Fig. 5. (Colour online) Inter-agent distances r_{mn}, agent heading angles θ, and agent speeds v as a function of time using the (left) stochastic optimal control (CUF) and the (right) deterministic feedback control.

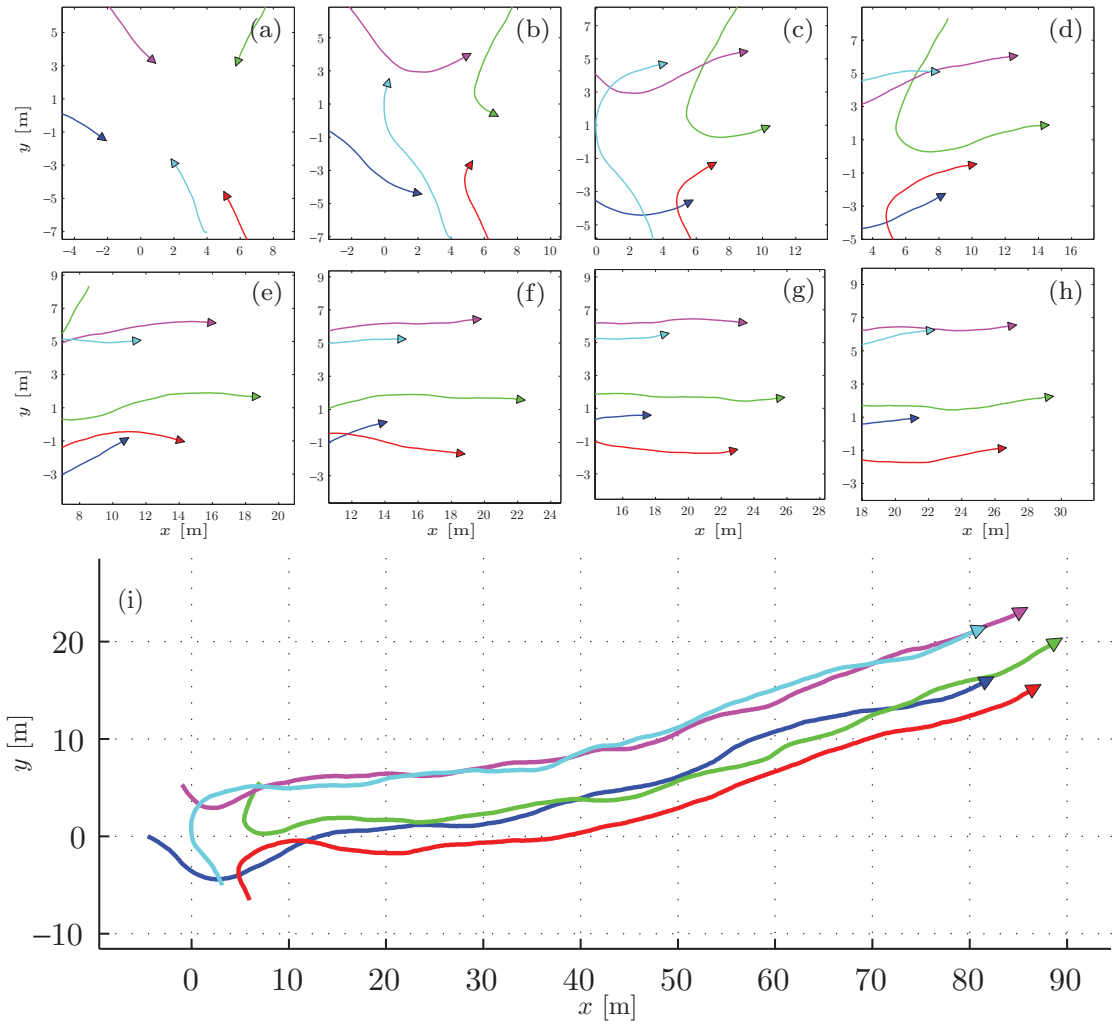

Fig. 6. (Colour online) Five agents, starting from random initial positions and a common speed $v = 1$ m/s, achieve a regular pentagon formation by an individually optimal choice of acceleration and turning rate, without any active communication. Frames from (a) to (h) are 2.5 s to 20 s, incrementing by 2.5 s. Frame (i) shows the end of the simulated trajectories at 60.6 s.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

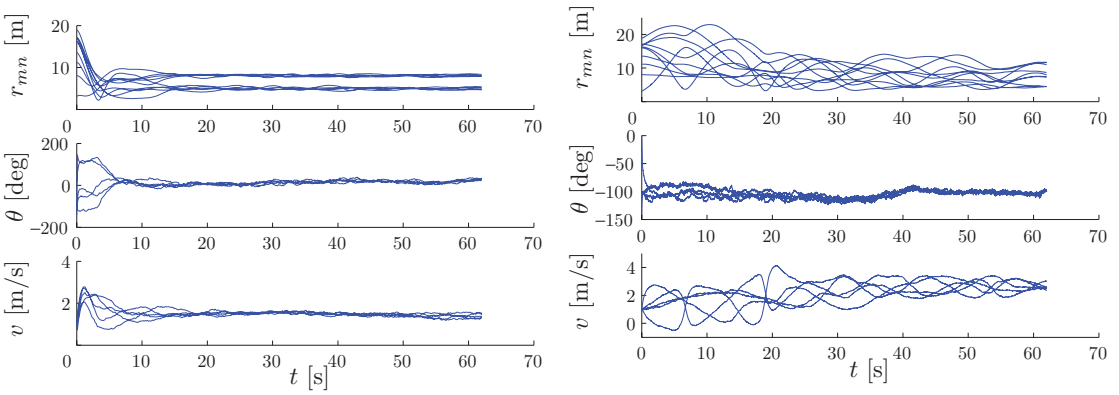

Fig. 7. (Colour online) Inter-agent distances r_{mn}, agent heading angles θ, and agent speeds v as functions of time using (left) the stochastic optimal control (infinite-horizon), and (right) the deterministic feedback control.

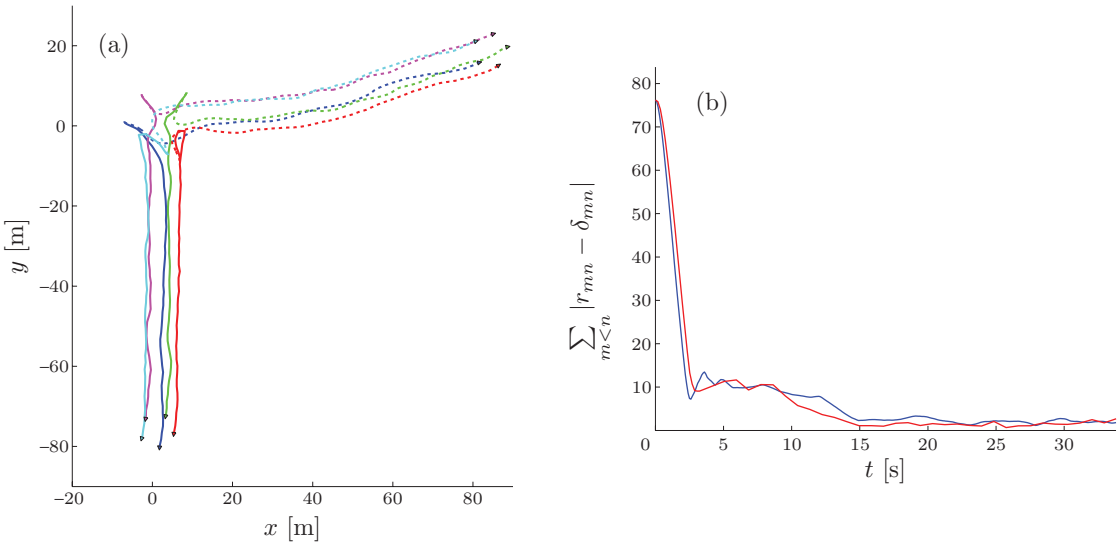

Fig. 8. (Colour online) Comparison with MPC-computed control for the same problem. (a) The dashed trajectories are those from Fig. 6 computed using Kalman smoothing algorithms, while the solid trajectories were computed using IPOPT.[53] (b) The total error between the instantaneous formation distances and the nominal distances for the MPC control (red) and the Kalman smoother control (blue).

smoothing-based pentagon, but the direction of the agents' motion was not part of the optimization problem. The difference between the actual inter-agent distances and the nominal distances for the MPC-based trajectories are close to that from our Kalman smoothing methods (Fig. 8(b)), but the latter approach requires considerably less computational time (approximately 1.7 s to compute five agents' controls).

Finally, we show how the choice of measurements $\boldsymbol{\mu}$ supplied to the smoothing algorithm can allow for dynamic morphing between formations. At the end of the simulation in Fig. 6, we update the nominal inter-agent distances so that agents form a line, as seen in Fig. 9.

6. Discussion

This work considers the problem of unicycle formation control in a distributed optimal feedback control setting. Since this gives rise to a system with huge state space, we exploit the stochasticity inherent in distributed multi-agent control problems in order to apply a PI method. By relating the resulting estimation problem to Kalman smoothing algorithms, each agent can compute its optimal control using a nonlinear Kalman smoother, greatly reducing the computational complexity associated with multi-robot optimal control problems. The measurement and process noise of the smoothing

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

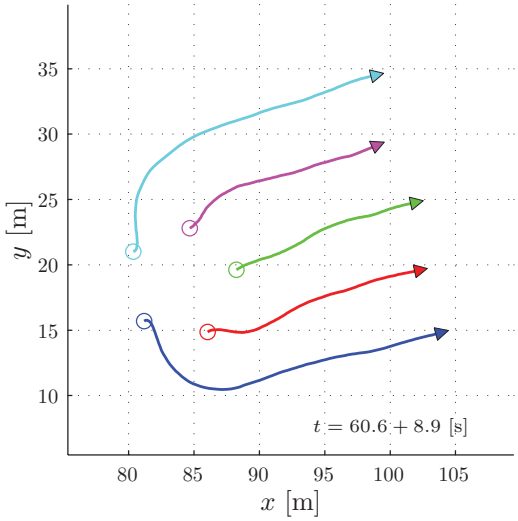

Fig. 9. (Colour online) Following the simulation in Fig. 6, the set of nominal distances between agents m and n was modified to $\delta_{mn} = 5|m - n|$, and the reference control was dropped. The agents are seen forming a line.

problem are created using the structure of the cost function and stochastic kinematics. Since agents share a common prior probability distribution describing the future trajectories of their neighbors, the formation can be attained without any communication among agents aside from instantaneous observations of neighbors.

The typical Lyapunov function approach to problems of this type offers several benefits, including transparency and analyticity, that are often lost in the numerical solution of the HJB equation, but the latter is also advantageous or in some instances preferable. Our approach aims to balance the attractiveness of an analytic feedback control for deterministic systems with the robustness to stochasticity and optimality provided by the HJB equation. Therefore, the optimal turning rate and acceleration controls affect the system alongside a non-optimal reference feedback control law based on an artificial potential function.

Not only do the optimal correction terms improve the performance of the reference control, leading the agents into formation faster and optimally, but the existence of an underlying non-optimal control complements the optimal enhancement controls. For instance, since the reference controls help to move the agents toward a formation, a Kalman smoother measurement (recall that a measurement in our work is the nominal distance) does not constitute a rare event in the measurement probability model, and, consequently, the reference controls reduce numerical issues involved in Kalman smoothing algorithms. Moreover, without the reference controls, collision avoidance would require complex state constraints to be added to the Kalman smoothing algorithm.

Since the Kalman smoothing-based control is the additional input required alongside the non-optimal control in order to achieve optimality and robustness, we can envision in a future line of research using these algorithms to test feedback control laws for multi-robot systems against various scenarios in order to derive analytical improvements.

References

1. L. E. Parker, "Multiple Mobile Robot Systems," **In**: *Springer Handbook of Robotics* (B. Sciliano and O. Khatib, eds.) (Springer-Verlag, Berlin, Germany, 2008), Ch. 40, pp. 921–941.
2. A. W. Proud, M. Pachter and J. J. D'Azzo, "Close Formation Flight Control," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR (1999) pp. 1231–1246.
3. B. Anderson, B. Fidan, C. Yu and D. Walle, "UAV Formation Control: Theory and Application," **In**: *Recent Advances in Learning and Control* (V. Blondel, S. Boyd, and H. Kimura, eds.) (Springer-Verlag, London, 2008) pp. 15–34.
4. H. Tanner, A. Jadbabaie and G. Pappas, "Coordination of Multiple Autonomous Vehicles," *Proceedings of the IEEE Mediterranean Conference on Control and Automation*, Rhodes, Greece (2003) pp. 3448–3453.
5. G. Elkaim and R. Kelbley, "A Lightweight Formation Control Methodology for a Swarm of Non-Holonomic Vehicles," *IEEE Aerospace Conference*, Big Sky, MT (2006) pp. 1–8.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

6. T. Paul, T. R. Krogstad and J. T. Gravdahl, "UAV Formation Flight Using 3D Potential Field," *Proceedings of the 16th Mediterranean Conference on Control and Automation*, Ajaccio, France (2008) pp. 1240–1245.
7. G. Roussos, D. V. Dimarogonas and K. J. Kostas, "3D navigation and collision avoidance for nonholonomic aircraft-like vehicles," *Int. J. Adapt. Control* **24**, 900–920 (2010).
8. Y. Zou and P. R. Pagilla, "Distributed formation flight control using constraint forces," *J Guide. Control Dynam.* **32**(1), 112–120 (2009).
9. F. Bullo, J. Cortes and S. Martinez, *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms* (Princeton University Press, Princeton, NJ, 2009).
10. S. N. Singh, P. Chandler, C. Schumacher, S. Banda and M. Pachter , "Nonlinear adaptive close formation control of unmanned aerial vehicles," *Dynam. Control* **10**(2), 179–194 (2000).
11. R. Sattigeri, A. J. Calise and J. H. Evers, "An Adaptive Vision-Based Approach to Decentralized Formation Control," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, RI (2004) pp. 2575–2798.
12. D. Galzi and Y. Shtessel, "UAV Formations Control Using High Order Sliding Modes," *Proceedings of the 2006 American Control Conference*, Minneapolis, MN (2006) pp. 4249–4254.
13. W. Ren and R. Beard, *Distributed Consensus in Multi-Vehicle Cooperative Control: Theory and Applications* (Springer-Verlag, New York, NY, 2007).
14. D. V. Dimarogonas, "On the rendezvous problem for multiple nonholonomic agents," *IEEE T. Automat. Control* **52**(5), 916–922 (2007).
15. N. G. van Kampen, *Stochastic Processes in Physics and Chemistry*, 3rd ed. (Elsevier B.V., Amsterdam, Netherlands, 2007).
16. A. W. Long, K. C. Wolfe, M. J. Mashner and G. S. Chirikjian, "The Banana Distribution is Gaussian : A Localization Study with Exponential Coordinates," *Proceedings of Robotics: Science and Systems*, Syndey, Australia (2012) pp. 265–272.
17. M. C. Wang and G. Uhlenbeck, "On the theory of Brownian Motion II," *Rev. Mod. Phys.* **17**(2–3), 323–342 (1945).
18. W. B. Dunbar and R. M. Murray, "Distributed receding horizon control for multi-vehicle formation stabilization," *Automatica* **42**(5), 549–558 (2006).
19. M. Freidlin, *Functional Integration and Partial Differential Equations* (Princeton University Press, Princeton, NJ, 1985).
20. B. Oksendal, *Stochastic Differential Equations: An Introduction with Applications*, 6th ed. (Springer-Verlag, Berlin, Germany, 2003).
21. J. Yong, "Relations among ODEs, PDEs, FSDEs, BDSEs, and FBSDEs," *Proceedings of the 36th IEEE Conference on Decision and Control*, San Diego, CA (1997) pp. 2779–2784.
22. H. Kappen, "Linear theory for control of nonlinear stochastic systems," *Phys. Rev. Lett.* **95**(20), 1–4 (2005).
23. H. J. Kappen, "Path integrals and symmetry breaking for optimal control theory," *J. Stat. Mech. Theory E.* **2005**(21), (2005) pp. 1–25.
24. E. Todorov, "General Duality Between Optimal Control and Estimation," *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico (2008) pp. 4286–4292.
25. E. Todorov, "Efficient computation of optimal actions." *P. Natl. Acad. Sci. USA* **106**(28), 11478–11483 (2009).
26. H. J. Kappen, V. Gómez and M. Opper, "Optimal control as a graphical model inference problem," *Mach. Learn.* **87**(2), 159–182 (2012).
27. D. Milutinović, "Utilizing Stochastic Processes for Computing Distributions of Large-Size Robot Population Optimal Centralized Control," *Proceedings of the 10th International Symposium on Distributed Autonomous Robotic Systems*, Lausanne, Switzerland (2010).
28. A. Palmer and D. Milutinović, "A Hamiltonian Approach Using Partial Differential Equations for Open-Loop Stochastic Optimal Control," *Proceedings of the 2011 American Control Conference*, San Francisco, CA (2011) pp. 2056–2061.
29. B. van den Broek, W. Wiegerinck and B. Kappen, "Graphical model inference in optimal control of stochastic multi-agent systems," *J. Artif. Intell. Res.* **32**(1), 95–122 (2008).
30. B. van den Broek, W. Wiegerinck and B. Kappen, "Optimal Control in Large Stochastic Multi-Agent Systems," *Adaptive Agents and Multi-Agent Systems III Adaptation and Multi-Agent Learning*, LNAI, vol. 4865. Springer-Verlag, Berlin, Germany, pp. 15–26 (2008).
31. W. Wiegerinck, B. Broek and H. Kappen, "Stochastic Optimal Control in Continuous Space-Time Multi-Agent Systems," *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*, Cambridge, MA (2006) pp. 528–535.
32. W. Wiegerinck, B. van den Broek and B. Kappen, "Optimal On-Line Scheduling in Stochastic Multiagent Systems in Continuous Space-Time," *Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems* (2007).
33. R. P. Anderson and D. Milutinović, "A Stochastic Optimal Enhancement of Feedback Control for Unicycle Formations," *Proceedings of the 11th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, Baltimore, MD (2012) pp. 608–615.
34. R. P. Anderson and D. Milutinović, "Distributed Path Integral Feedback Control Based on Kalman Smoothing for Unicycle Formations," *Proceedings of the 2013 American Control Conference*, Washington, DC (2013) pp. 4611–4616.

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press

35. A. Jadbabaie and J. Hauser, "On the Stability of Unconstrained Receding Horizon Control with a General Terminal Cost," *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, FL (2001) pp. 4826–4831.
36. R. F. Stengel, *Optimal Control and Estimation* (Dover, New York, NY, 1994).
37. B. Bouilly, T. Simeon and R. Alami, "A Numerical Technique for Planning Motion Strategies of a Mobile Robot in Presence of Uncertainty," *Proceedings of the 1995 IEEE International Conference on Robotics and Automation*. Nagoya, Japan (1995) pp. 1327–1332.
38. T. Fraichard and R. Mermond, "Path Planning with Uncertainty for Car-Like Robots," *Proceedings of the 1998 IEEE International Conference on Robotics & Automation*, Leuven, Belgium (1998) pp. 27–32.
39. A. Lambert and D. Gruyer, "Safe Path Planning in an Uncertain-Configuration Space," *Proceedings of the 2003 International Conference on Robotics and Automation*, Taipei, Taiwan (Sep. 14–19, 2003) pp. 85–90.
40. J. van den Berg, P. Abbeel and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," *Int. J. Robot. Res.* **30**(7), 895–913 (2011).
41. J. van den Berg, S. Patil, R. Alterovitz, P. Abbeel and K. Goldberg , "LQG-Based Planning, Sensing, and Control of Steerable Needles," Vol. 68, *Springer Tracts in Advanced Robotics: Algorithmic Foundations of Robotics IX* (D. Hsu, V. Isler, J.-C. Latombe, M. C. Lin, eds.) (Springer-Velag, Berlin, Germany, 2011), pp. 373–389.
42. D. Q. Mayne, J. B. Rawlings, C. V. Rao and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica* **36**(6), 789–814 (2000).
43. D. Bertsekas, "Dynamic programming and suboptimal control: A survey from ADP to MPC," *Eur. J. Control* **11**(4–5), 310–334 (2005).
44. Z. Chao, S.-L. Zhou, L. Ming and W.-G. Zhang, "UAV formation flight based on nonlinear model predictive control," *Math. Probl. Eng.* **2012**, (2012) pp. 1–15 (and references therein).
45. T. P. Nascimento, A. P. Moreira and A. G. S. Conceção, "Multi-robot nonlinear model predictive formation control: Moving target and target absence," *Robot. Auton. Syst.* **61**(12), 1502–1515 (2013).
46. W. Fleming and H. Soner, "Logarithmic Transformations and Risk Sensitivity," **In**: *Controlled Markov Processes and Viscosity Solutions* (Springer, Berlin, Germany, 1993), Ch. 6, pp. 227–259.
47. H. Goldstein, C. P. Poole, Jr. and J. L. Safko, *Classical Mechanics*, 3rd ed. (Addison-Wesley, San Francisco, CA, 1980).
48. A. Gelb, *Applied Optimal Estimation*. (MIT Press, Cambridge, MA, 1974).
49. S. Särkkä, "Continuous-time and continuous-discrete-time unscented Rauch-Tung-Striebel smoothers," *Signal Process.* **90**(1), 225–235 (2010).
50. S. Särkkä, "EKF/UKF Toolbox for Matlab V1.3," available at: `http://becs.aalto.fi/en/research/bayes/ekfukf/`. Accessed November 2011.
51. E. A. Wan and R. van der Merwe, "Unscented Kalman Filter," **In**: *Kalman Filtering and Neural Networks* (S. Haykin, ed.) (John Wiley, New York, NY, 2000), Ch. 7, pp. 221–282.
52. H. J. Kushner and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, 2nd ed. (Springer, New York, NY, 2001).
53. A. Wächter and L. T. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Math. Program.* **106**(1), 25–57, (2006).

https://doi.org/10.1017/S0263574714000022 Published online by Cambridge University Press