Partitioning Algorithms for Multi-Agent Systems Based on Finite-Time Proximity Metrics ⋆

Efstathios Bakolas [a]

[a] *Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, Texas 78712-1221, USA*

Abstract

We address a generalized Voronoi partitioning problem for a team of mobile agents with nonlinear dynamics with respect to a state-dependent proximity metric. In particular, the proximity (pseudo-) metric corresponds to the reduction of a generalized energy metric that occurs during the transfer of an agent to an arbitrary destination with zero terminal velocity, in finite time. The realization of every finite-time state transition takes place by means of a class of continuous feedback control laws that render the closed loop dynamics of each mobile agent non-Lipschitzian. The arrival time also turns out to be a state-dependent quantity, whose functional description is not prescribed a priori. We show that the partitioning problem studied in this work can admit a decentralized solution, that is, each agent can compute its own cell independently from its teammates provided that is aware of the positions and velocities of its neighboring agents. Numerical simulations that illustrate the theoretical developments are also presented.

Key words: Autonomous agents, Voronoi diagrams, partitioning algorithms, finite-time control, computational methods.

1 Introduction

The importance of using state-dependent proximity metrics in partitioning problems related to applications of multi-agent dynamical systems was first identified in [3] and subsequently extended in a series of papers [1, 2, 4, 13]. In particular, as is stressed in [3], the use of state-dependent proximity metrics leads to spatial partitions that encode proximity information that succinctly captures the dynamic characteristics of the agents; something that cannot be achieved with more standard spatial partitions [11], whose proximity metrics stem from geometric considerations primarily. A typical problem in which the proposed class of spatial partitions could play a key role is the following: A team of agents are seeking for an optimal way to subdivide a common subset of their state space such that they can share among them the load of carrying out tasks that need to be attended at different locations in this space, in a fair way.

The state-dependent proximity metrics that are typically employed in the literature correspond to the value functions of relevant optimal control problems. Unfortunately, the cost of computing the value function for even simple optimal control problems can be excessively high unless one is confined to the classes of problems involving mobile agents with single integrator kinematics. An exception to this rule is presented in [1], wherein a class of partitioning problems for multi-agent systems with linear second order dynamics and with respect to the minimum control effort metric is addressed under the assumption that the arrival time is constant throughout the configuration space of the agents. On the one hand, the spatial partitions proposed in [1] can be directly associated with a class of power or affine diagrams, which are spatial subdivisions comprised of convex polyhedra of low combinatorial complexity. On the other hand, the applicability of the results of [1] is limited by 1) the fact that the linear/affine dynamic model employed therein cannot always describe accurately the motion of a mobile agent, and 2) the assumption that the terminal time is equal to a prescribed constant for any final destination, for the value of which the agents have to somehow agree a priori, is neither practical nor in accordance with the modern requirements for decentralized operation of multi-agent systems.

The main objective of this work is to remove the two strong assumptions made in [1] regarding the linearity of the agent dynamics and the constancy of the terminal time, while keeping at the same time, if possible, the same solution structure, namely the structure of affine or power diagrams. This objective should be achieved under the requirement that the utilized proximity metric gives a meaningful assessment of the "closeness" of a mobile agent with nonlinear dynamics from an arbitrary destination, which should be reached with zero

⋆ Corresponding author E. Bakolas. Tel.: +1 512 471 4250.
Email address: bakolas@austin.utexas.edu (Efstathios Bakolas).

terminal velocity (soft landing) in finite time. Note that the final time is, in general, a state-dependent quantity whose functional description is not known a priori. Consequently, the partition space corresponds to a two-dimensional (flat) manifold embedded in the common, four-dimensional state space of the agents. It turns out that all the previous requirements for the spatial partition can be met by taking the proximity metric to be the reduction of a generalized energy metric associated with the transfer of an agent to its goal configuration. This state transition is realized with the application of the continuous feedback control law proposed in [16], which enforces finite time convergence by rendering the closed loop dynamics non-Lipschitzian [5, 8–10].

Similar to the class of problems studied in [1], the solution to the partitioning problems considered in this work can be associated with a class of affine diagrams that subdivide a two-dimensional Euclidean space into a finite collection of non-overlapping convex polygons. Interestingly, the partitioning problem can be solved in a decentralized way, in the sense that each agent can compute its own cell independently from its teammates provided that is aware of the positions and velocities of all the agents whose cells share a common face with its own cell in the affine diagram (neighboring agents). In this way, the combination of the solutions to the local partitioning problems solved independently by each agent furnishes a good approximation of the solution to the global partitioning problem. The proposed decentralized partitioning algorithm is built upon an algorithm for partitioning problems involving teams of single integrators proposed in our previous work [2], which was inspired by [12]. The new challenge in the class of problems considered in this work has to do with the fact that, due to the second order nonlinear dynamics of the agents, the generators of the partition are not necessarily interior points of their corresponding cells; something that precludes the direct application of the results presented in [2]. To address this issue, we associate the original partitioning problem with an affine diagram problem, whose generators are taken to be the minimizers of the proximity metric characterized for each corresponding agent individually. In this way, the new generators are more likely to be interior points of their associated cells. For the case when the new generators fail again to belong to the interior of their associated cells, a modified algorithm is proposed.

The rest of the paper is organized as follows. Section 2 deals with the steering problem for a single agent. The partitioning problem for the case of multiple agents is addressed in Section 3. Section 4 presents numerical simulations, and finally, Section 5 concludes the paper with a summary of remarks.

2 Problem Formulation

2.1 Notation

We denote by $\mathbb{R}^n$ the set of n-dimensional real vectors. The unit circle in $\mathbb{R}^2$ will be denoted by $\mathbb{S}^1$. We denote by $\mathbb{R}_{\geq 0}$ and $\mathbb{Z}_{>0}$, respectively, the sets of non-negative real numbers and (strictly) positive integers. We write $|\boldsymbol{\alpha}|$, to denote the vector 2-norm of a vector $\boldsymbol{\alpha} \in \mathbb{R}^n$. Given two vectors $\boldsymbol{\alpha},\ \boldsymbol{\beta} \in \mathbb{R}^n$, we denote by $[\boldsymbol{\alpha}, \boldsymbol{\beta}]$, the line segment from $\boldsymbol{\alpha}$ to $\boldsymbol{\beta}$; when $\boldsymbol{\alpha} \equiv \boldsymbol{\beta}$, we have $[\boldsymbol{\alpha}, \boldsymbol{\beta}] \equiv \boldsymbol{\alpha}$. Given two (column) vectors $\boldsymbol{\alpha} \in \mathbb{R}^{n_1}$, $\boldsymbol{\beta} \in \mathbb{R}^{n_2}$, we denote by $\mathrm{col}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ the $n_1 + n_2$ dimensional real (column) vector that corresponds to the concatenation of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The scalar-valued signum function is denoted by $\mathrm{sgn}(\cdot)$, where $\mathrm{sgn}(\cdot) : \mathbb{R} \mapsto \{-1, 0, 1\}$. In addition, we write $\mathrm{vsgn}^\gamma(\boldsymbol{\alpha})$ and $\mathrm{vnorm}^\gamma(\boldsymbol{\alpha})$, where $\gamma > 0$ and $\boldsymbol{\alpha} = \mathrm{col}(\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$, to denote the n-dimensional (column) vectors $\mathrm{col}(\mathrm{sgn}(\alpha_1)|\alpha_1|^\gamma, \ldots, \mathrm{sgn}(\alpha_n)|\alpha_n|^\gamma)$ and $\mathrm{col}(|\alpha_1|^\gamma, \ldots, |\alpha_n|^\gamma)$, respectively. Furthermore, given $\boldsymbol{\alpha} = \mathrm{col}(\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$, we denote by $\boldsymbol{\Delta}(\boldsymbol{\alpha})$ the $n \times n$ diagonal matrix whose diagonal elements are the components of $\boldsymbol{\alpha}$, that is, $\boldsymbol{\Delta}_{[ii]}(\boldsymbol{\alpha}) = \alpha_i$, $i \in \{1, \ldots, n\}$. We write $\mathbf{P} = \mathbf{P}^{\mathrm{T}} \succ \mathbf{0}$ to denote the fact that a square (symmetric) matrix is positive definite. In addition, $\mathrm{bd}(S)$ and $\mathrm{int}(S)$ denote, respectively, the boundary and the interior of a set S; the measure of S is denoted by $\mathcal{A}(S)$. Finally, we denote by $\Theta(f(n))$ the set of functions $F : \mathbb{Z}_{>0} \mapsto [0, \infty)$ for which there exist $c_1,\ c_2 > 0$ and $n_0 \in \mathbb{Z}_{>0}$ such that $c_1 f(n) \leq F(n) \leq c_2 f(n)$, for all $n \geq n_0$.

2.2 Equations of motion

We consider a team of n mobile agents that are initially located at n distinct positions $\bar{\boldsymbol{x}}_i \in \mathbb{R}^2$, where $i \in \mathcal{I}_n := \{1, \ldots, n\}$. We write $\bar{\boldsymbol{x}}_i \in \overline{\boldsymbol{\mathcal{X}}}$, where $\overline{\boldsymbol{\mathcal{X}}} := \{\bar{\boldsymbol{x}}_i,\ i \in \mathcal{I}_n\} \subsetneq \mathbb{R}^2$. Each agent has a prescribed initial velocity, which is denoted by $\bar{\boldsymbol{v}}_i \in \mathbb{R}^2$. We write $\bar{\boldsymbol{v}}_i \in \overline{\boldsymbol{\mathcal{N}}}$, where $\overline{\boldsymbol{\mathcal{N}}} := \{\bar{\boldsymbol{v}}_i,\ i \in \mathcal{I}_n\} \subsetneq \mathbb{R}^2$. The composite initial state vector of the i-th agent is denoted by $\bar{\boldsymbol{z}}_i$; we write $\bar{\boldsymbol{z}}_i := \mathrm{col}(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{v}}_i)$ and we denote the set of initial states by $\overline{\boldsymbol{\mathcal{Z}}}$, where $\overline{\boldsymbol{\mathcal{Z}}} := \overline{\boldsymbol{\mathcal{X}}} \times \overline{\boldsymbol{\mathcal{N}}}$.

The motion of the i-th agent, where $i \in \mathcal{I}_n$, is described by the following set of equations:

$$\dot{\boldsymbol{x}}_i = \boldsymbol{v}_i, \qquad \boldsymbol{x}_i(0) = \bar{\boldsymbol{x}}_i, \tag{1a}$$

$$\dot{\boldsymbol{v}}_i = \boldsymbol{\alpha}_i(\boldsymbol{z}_i) + \mathbf{B}_i(\boldsymbol{z}_i)\boldsymbol{u}_i(t), \qquad \boldsymbol{v}_i(0) = \bar{\boldsymbol{v}}_i, \tag{1b}$$

where $\boldsymbol{x}_i := \mathrm{col}(x_{i,1}, x_{i,2}) \in \mathbb{R}^2$ and $\boldsymbol{v}_i := \mathrm{col}(v_{i,1}, v_{i,2}) \in \mathbb{R}^2$ are, respectively, the position and velocity vectors of the i-th agent at time t; we will denote the state vector of the i-th agent at time t by $\boldsymbol{z}_i$, where $\boldsymbol{z}_i := \mathrm{col}(\boldsymbol{x}_i, \boldsymbol{v}_i) \in \mathbb{R}^4$. The initial velocity of the i-th agent is not necessarily zero at $t = 0$ (the i-th agent is not necessarily at rest initially). In addition, $\boldsymbol{u}_i$ is the control input of the i-th agent. It is assumed that the functions $\boldsymbol{\alpha}_i(\cdot) : \mathbb{R}^4 \mapsto \mathbb{R}^2$ and $\mathbf{B}_i(\cdot) : \mathbb{R}^4 \mapsto \mathbb{R}^{2\times 2}$ are, at least, Lipschitz continuous. In addition, we assume that $\mathbf{B}_i(\boldsymbol{z})$ is non-singular for all $\boldsymbol{z} \in \boldsymbol{\mathcal{Z}} \subseteq \mathbb{R}^4$, where $\boldsymbol{\mathcal{Z}}$ is the subset of the state space where the agents are expected to be operating; we write $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{X}} \times \boldsymbol{\mathcal{N}}$ where $\boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2$, $\boldsymbol{\mathcal{N}} \subseteq \mathbb{R}^2$. We say that the system is *fully actuated* in $\boldsymbol{\mathcal{Z}}$. For example, the motion of the i-th agent is described by the following equation:

$$\mathbf{M}_i(\boldsymbol{x}_i)\dot{\boldsymbol{v}}_i + \mathbf{C}_i(\boldsymbol{z}_i)\boldsymbol{v}_i + \boldsymbol{g}_i(\boldsymbol{x}_i) = \boldsymbol{u}_i, \tag{2}$$

where $\mathbf{M}_i(\cdot)$, $\mathbf{C}_i(\cdot)$ and $\boldsymbol{g}_i(\cdot)$ are state-dependent matrix and vector valued functions of appropriate dimensions, respectively, which can be written in the form of

Eqs. (1a)-(1b) by taking

$$\boldsymbol{\alpha}_i(\boldsymbol{z}_i) = -\mathbf{M}_i^{-1}(\boldsymbol{x}_i)\left(\mathbf{C}_i(\boldsymbol{z}_i)\boldsymbol{v}_i + \boldsymbol{g}_i(\boldsymbol{x}_i)\right),$$
$$\mathbf{B}_i(\boldsymbol{z}_i) = \mathbf{M}_i^{-1}(\boldsymbol{x}_i),$$

provided that the so-called mass inertia matrix $\mathbf{M}_i(\boldsymbol{x}) = \mathbf{M}_i^{\mathrm{T}}(\boldsymbol{x}) \succ \mathbf{0}$, for all $\boldsymbol{z} \in \boldsymbol{\mathcal{Z}}$. Note that Eq. (2) corresponds to a special case of the Euler-Lagrange equations that are commonly used to describe the motion of mobile robots and/or robotic systems (with two degrees of freedom, in our case) in the relevant literature [15]. Therefore, the nonlinear model described in Equations (1a)-(1b) is more general than the linear models considered in our previous work [1].

2.3 Finite horizon steering problem

Next, we consider the finite horizon steering problem for the i-th agent. Specifically, the i-th agent has to reach a state $\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{x})$ in the two-dimensional (flat) manifold $\boldsymbol{\mathcal{Z}}_0 := \{\mathrm{col}(\boldsymbol{x}, 0) : \ \boldsymbol{x} \in \boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2\}$, where $\boldsymbol{\mathcal{Z}}_0 \subsetneq \boldsymbol{\mathcal{Z}}$, in finite time. Equivalently, the objective of the i-th agent is to arrive at a prescribed, yet arbitrary, terminal position $\boldsymbol{x} \in \mathbb{R}^2$ with zero terminal velocity (soft landing) at a finite time $T > 0$; we write $\boldsymbol{z}_i(t) \to \boldsymbol{z}(\boldsymbol{x})$ as $t \to T$.

Problem 1 (Steering Problem) *Consider the i-th agent whose motion is described by Eqs.* (1a)-(1b)*, and let $\boldsymbol{x} \in \boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2$ be given. Determine a feedback control law $\boldsymbol{u}_i(\cdot; \boldsymbol{x}) : \mathbb{R}^4 \mapsto \mathbb{R}^2$ that will steer the i-th agent from the state $\bar{\boldsymbol{z}}_i \in \overline{\boldsymbol{\mathcal{Z}}}$, at time $t = 0$, to the state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$, as $t \to T$, for some $0 \leq T < \infty$.*

Proposition 1 *Let us consider the i-th agent, whose motion is described by Eqs.* (1a)-(1b)*, and let $\boldsymbol{x} \in \mathbb{R}^2$. In addition, let $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^{\mathrm{T}} \succ \mathbf{0}$, $\mathbf{K}_1 = \mathbf{K}_1{}^{\mathrm{T}} \succ \mathbf{0}$ and $\mathbf{K}_2 = \mathbf{K}_2{}^{\mathrm{T}} \succ \mathbf{0}$ be 2×2 diagonal matrices. Then, the feedback control law $\boldsymbol{u}_i(\cdot; \boldsymbol{x}) : \mathbb{R}^4 \mapsto \mathbb{R}^2$, where*

$$\boldsymbol{u}_i(\boldsymbol{z}_i; \boldsymbol{x}) := -\mathbf{B}_i^{-1}(\boldsymbol{z}_i)\Big(\boldsymbol{\alpha}_i(\boldsymbol{z}_i) + \frac{1}{\gamma}\boldsymbol{\Lambda}^{-1}\mathrm{vsgn}^{2-\gamma}(\boldsymbol{v}_i) + \mathbf{K}_1\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}) + \mathbf{K}_2\mathrm{vsgn}^{\rho}(\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}))\Big), \quad (3)$$

where $\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}) := \boldsymbol{x}_i - \boldsymbol{x} + \boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\boldsymbol{v}_i)$, $1 < \gamma < 2$ and $0 < \rho < 1$, will steer the i-th agent to the state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$, as $t \to T$, for some $0 < T < \infty$.

PROOF. Let us consider, for a given $\boldsymbol{x} \in \mathbb{R}^2$, the candidate quasi-Lyapunov function $V(\cdot; \boldsymbol{x}) : \mathbb{R}^4 \mapsto \mathbb{R}_{\geq 0}$, where

$$V(\boldsymbol{z}_i; \boldsymbol{x}) := \langle \boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}), \mathbf{P}\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x})\rangle, \quad (4)$$

where $\mathbf{P} = \mathbf{P}^{\mathrm{T}} \succ \mathbf{0}$ is a 2×2 diagonal matrix. By using the facts that

$$\begin{aligned}\mathrm{vsgn}^{\gamma}(\boldsymbol{v}_i) &= \mathrm{col}(\mathrm{sgn}(v_{i,1})|v_{i,1}|^{\gamma}, \mathrm{sgn}(v_{i,2})|v_{i,2}|^{\gamma}) \\ &= \mathrm{col}(v_{i,1}|v_{i,1}|^{\gamma-1}, v_{i,2}|v_{i,2}|^{\gamma-1}) \\ &= \boldsymbol{\Delta}(\mathrm{vnorm}^{\gamma-1}(\boldsymbol{v}_i))\boldsymbol{v}_i,\end{aligned}$$

and

$$2|v_{i,\ell}|\frac{\mathrm{d}|v_{i,\ell}|}{\mathrm{d}t} = \frac{\mathrm{d}|v_{i,\ell}|^2}{\mathrm{d}t} = \frac{\mathrm{d}v_{i,\ell}^2}{\mathrm{d}t} = 2v_{i,\ell}\frac{\mathrm{d}v_{i,\ell}}{\mathrm{d}t},$$

for $\ell \in \{1, 2\}$, it is easy to show that

$$\begin{aligned}\dot{\boldsymbol{s}}_i(\boldsymbol{z}_i; \boldsymbol{x}) &= \dot{\boldsymbol{x}}_i + \gamma\boldsymbol{\Lambda}\boldsymbol{\Delta}(\mathrm{vnorm}^{\gamma-1}(\boldsymbol{v}_i))\dot{\boldsymbol{v}}_i \\ &= \boldsymbol{v}_i + \gamma\boldsymbol{\Lambda}\boldsymbol{\Delta}(\mathrm{vnorm}^{\gamma-1}(\boldsymbol{v}_i))\big(\boldsymbol{\alpha}_i(\boldsymbol{z}_i) \\ &\quad + \mathbf{B}_i(\boldsymbol{z}_i)\boldsymbol{u}_i(\boldsymbol{z}_i; \boldsymbol{x})\big) \\ &= -\widehat{\mathbf{K}}_1(\boldsymbol{v}_i)\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}) - \widehat{\mathbf{K}}_2(\boldsymbol{v}_i)\mathrm{vsgn}^{\rho}(\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x})),\end{aligned}$$

where $\widehat{\mathbf{K}}_1(\boldsymbol{v}_i) := \gamma\boldsymbol{\Lambda}\boldsymbol{\Delta}(\mathrm{vnorm}^{\gamma-1}(\boldsymbol{v}_i))\mathbf{K}_1$ and $\widehat{\mathbf{K}}_2(\boldsymbol{v}_i) := \gamma\boldsymbol{\Lambda}\boldsymbol{\Delta}(\mathrm{vnorm}^{\gamma-1}(\boldsymbol{v}_i))\mathbf{K}_2$ are diagonal matrices with $\widehat{\mathbf{K}}_1(\boldsymbol{v}_i) = \widehat{\mathbf{K}}_1^{\mathrm{T}}(\boldsymbol{v}_i) \succ \mathbf{0}$ and $\widehat{\mathbf{K}}_2(\boldsymbol{v}_i) = \widehat{\mathbf{K}}_2^{\mathrm{T}}(\boldsymbol{v}_i) \succ \mathbf{0}$, for all $\boldsymbol{v}_i \in \mathbb{R}^2$ with $v_{i,1}v_{i,2} \neq 0$. The time derivative of V along the trajectories of the i-th agent driven by the feedback control law (3) is given by

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}V(\boldsymbol{z}_i; \boldsymbol{x}) &= 2\langle \boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}), \mathbf{P}\dot{\boldsymbol{s}}_i(\boldsymbol{z}_i; \boldsymbol{x})\rangle \\ &= -2\langle \boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}), \mathbf{P}\widehat{\mathbf{K}}_2(\boldsymbol{v}_i)\mathrm{vsgn}^{\rho}(\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}))\rangle \\ &\quad - 2\langle \boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}), \mathbf{P}\widehat{\mathbf{K}}_1(\boldsymbol{v}_i)\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x})\rangle. \quad (5)\end{aligned}$$

It is easy to show, after some algebraic manipulation and by utilizing Lemma 2 from [16], that there exist constants c_1, $c_2 > 0$ such that

$$\frac{\mathrm{d}}{\mathrm{d}t}V(\boldsymbol{z}_i; \boldsymbol{x}) \leq -c_1V(\boldsymbol{z}_i; \boldsymbol{x}) - c_2V^{\rho}(\boldsymbol{z}_i; \boldsymbol{x}), \quad (6)$$

provided that $v_{i,1}v_{i,2} \neq 0$, which in turn implies that the trajectories of the i-th agent will converge to the following two-dimensional manifold:

$$\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}) = \boldsymbol{x}_i - \boldsymbol{x} + \boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\boldsymbol{v}_i) = 0,$$

as $t \to T_m$, where $0 \leq T_m < \infty$. As explained in the proof of Theorem 1 from [16], the manifold $\boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}) = 0$ will be reached in finite time even in the case when $v_{i,1}v_{i,2} = 0$. Note that the manifold $\boldsymbol{s}_i = 0$ is invariant, given that $\dot{\boldsymbol{s}}_i = 0$ when $\boldsymbol{s}_i = 0$. In addition, let $\boldsymbol{p}_i(t) := \boldsymbol{x}_i(t) - \boldsymbol{x}$, for $t \geq T_m$. Then, $\boldsymbol{p}_i$ converges along the manifold $\boldsymbol{s}_i = 0$ to the origin. This convergence will take place in finite time, that is, $\boldsymbol{p}_i(t) \to 0$ as $t \to T$, where $0 \leq T_m \leq T < \infty$, given that along the manifold $\boldsymbol{s}_i = 0$, we have that

$$\boldsymbol{p}_i + \boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\dot{\boldsymbol{p}}_i) = 0,$$

where we have used the fact that $\boldsymbol{v}_i(t) = \dot{\boldsymbol{p}}_i(t)$. The last (vector) equation corresponds to a system of decoupled scalar differential equations with non-Lipschitzian vector fields, whose solutions converge to zero in finite time regardless of the initial conditions. The fact that $\boldsymbol{p}_i$ converges to the origin in finite time in turn implies that $\boldsymbol{v}_i(t) = \dot{\boldsymbol{p}}_i(t) \to 0$ as $t \to T$ (along the invariant manifold $\boldsymbol{s}_i = 0$); this is because, by definition, $\boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\boldsymbol{v}_i) = \boldsymbol{s}_i(\boldsymbol{z}_i; \boldsymbol{x}) - \boldsymbol{p}_i$. Therefore, we have shown that $\boldsymbol{p}_i(t) \to 0$, or equivalently, $\boldsymbol{x}_i(t) \to \boldsymbol{x}$ and $\boldsymbol{v}_i(t) \to 0$ as $t \to T$. The proof is now complete. ■

Remark 1 Note that the function V used in the proof of Proposition 1 is not, strictly speaking, a Lyapunov function. This is because $\boldsymbol{s}_i = 0$, or equivalently, $V(\boldsymbol{z}_i; \boldsymbol{x}) = 0$, does not necessarily imply that $\boldsymbol{z}_i = \boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, which explains why we refer to V as a quasi-Lyapunov function, following the suggestion given in [14].

2.4 The proximity metric in the state space

Next, we utilize the quasi-Lyapunov function used in the proof of Proposition 1 to define a generalized proximity metric that will measure the closeness of the i-th agent from a desired state $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$. In particular, we measure the distance of the i-th agent from a state $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$ in terms of the reduction of the generalized energy metric that occurs during the state transition of this agent from its initial state $\bar{\boldsymbol{z}}_i$ to $\boldsymbol{z}(\boldsymbol{x})$, in some finite time. Specifically, for a given $\bar{\boldsymbol{z}}_i \in \overline{\boldsymbol{\mathcal{Z}}}$, we define the generalized metric $\pi(\cdot;\bar{\boldsymbol{z}}_i) : \mathbb{R}^2 \mapsto \mathbb{R}_{\geq 0}$, where $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) := V(\bar{\boldsymbol{z}}_i;\boldsymbol{x})$, that is,

$$\begin{aligned}\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) := &\langle \bar{\boldsymbol{x}}_i - \boldsymbol{x}, \mathbf{P}(\bar{\boldsymbol{x}}_i - \boldsymbol{x})\rangle \\ &+ 2\langle \bar{\boldsymbol{x}}_i - \boldsymbol{x}, \mathbf{P}\boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\bar{\boldsymbol{v}}_i)\rangle \\ &+ \langle \boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\bar{\boldsymbol{v}}_i), \mathbf{P}\boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\bar{\boldsymbol{v}}_i)\rangle. \end{aligned} \tag{7}$$

The expression for $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i)$ can be written more compactly as follows:

$$\begin{aligned}&\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) = |\mathbf{P}^{1/2}(\bar{\boldsymbol{x}}_i - \boldsymbol{x})|^2 + 2\langle \bar{\boldsymbol{x}}_i - \boldsymbol{x}, \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i)\rangle + \sigma_i(\bar{\boldsymbol{v}}_i), \\ &\quad \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i) := \mathbf{P}\boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\bar{\boldsymbol{v}}_i), \\ &\quad \sigma_i(\bar{\boldsymbol{v}}_i) := |\mathbf{P}^{1/2}\boldsymbol{\Lambda}\mathrm{vsgn}^{\gamma}(\bar{\boldsymbol{v}}_i)|^2. \end{aligned} \tag{8}$$

It is interesting to note that $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i)$ measures the "distance" between an arbitrary state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$ and the initial state $\bar{\boldsymbol{z}}_i$ at time $t = 0$, which is fixed (*cost-to-come*), whereas $V(\boldsymbol{z}_i;\boldsymbol{x})$ measures the "distance" between the state $\boldsymbol{z}_i = \boldsymbol{z}_i(t)$ at time t and the fixed terminal state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, when $\boldsymbol{x}$ is fixed (*cost-to-go*).

3 The Partitioning Problem

Next, we formulate a generalized Voronoi partitioning problem that will furnish a subdivision of the two-dimensional (flat) manifold $\boldsymbol{\mathcal{Z}}_0$, which is embedded in $\boldsymbol{\mathcal{Z}} \subseteq \mathbb{R}^4$. The set of generators of this generalized Voronoi partition corresponds to the group of agents, which in turn are identified by their initial states in $\overline{\boldsymbol{\mathcal{Z}}}$.

Problem 2 (Partitioning Problem) *Let $\overline{\boldsymbol{\mathcal{Z}}} := \{\bar{\boldsymbol{z}}_i \in \mathbb{R}^4,\ i \in \mathcal{I}_n\}$ be given. Then, determine a partition $\boldsymbol{\mathfrak{V}} = \{\boldsymbol{\mathfrak{V}}_i,\ i \in \mathcal{I}_n\}$ of $\boldsymbol{\mathcal{Z}}_0$ such that*

(1) $\boldsymbol{\mathcal{Z}}_0 = \bigcup_{i\in\mathcal{I}_n} \boldsymbol{\mathfrak{V}}_i$ (that is, $\boldsymbol{\mathfrak{V}}$ is exhaustive),
(2) $\mathrm{int}(\boldsymbol{\mathfrak{V}}_i) \cap \mathrm{int}(\boldsymbol{\mathfrak{V}}_j) = \varnothing$, for all $i, j \in \mathcal{I}_n$, $i \neq j$ (that is, $\boldsymbol{\mathfrak{V}}$ consists of mutually disjoint sets),
(3) A state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$, belongs to $\boldsymbol{\mathfrak{V}}_i$ if, and only if, $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) \leq \pi(\boldsymbol{x};\bar{\boldsymbol{z}}_j)$, for all $j \in \mathcal{I}_n$, where $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_\ell)$, $\ell \in \{i, j\}$, is given by Eq. (8) (that is, $\boldsymbol{\mathfrak{V}}$ is optimal).

Next, we show that the solution to Problem 2 can be associated with an affine diagram in $\mathbb{R}^2$.

3.1 Analysis of the partitioning problem

Let us consider a pair of generators $(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j) \in \overline{\boldsymbol{\mathcal{Z}}} \times \overline{\boldsymbol{\mathcal{Z}}}$, $i \neq j$. We define their corresponding bisector in $\boldsymbol{\mathfrak{V}}$, which is denoted by $\mathcal{B}_{\boldsymbol{\mathfrak{V}}}(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j)$, to be the set of all states $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$ that are equidistant from $\bar{\boldsymbol{z}}_i$ and $\bar{\boldsymbol{z}}_j$ with respect to the proximity metric π, that is, $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) = \pi(\boldsymbol{x};\bar{\boldsymbol{z}}_j)$.

Proposition 2 *Let $\boldsymbol{\mathfrak{V}} := \{\boldsymbol{\mathfrak{V}}_i,\ i \in \mathcal{I}_n\}$ be the generalized Voronoi partition of $\boldsymbol{\mathcal{Z}}_0$ generated by the point-set $\overline{\boldsymbol{\mathcal{Z}}}$ that solves Problem 2. Then, the bisector $\mathcal{B}_{\boldsymbol{\mathfrak{V}}}(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j)$ of the partition $\boldsymbol{\mathfrak{V}}$ that corresponds to the pair of generators $(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j) \in \overline{\boldsymbol{\mathcal{Z}}} \times \overline{\boldsymbol{\mathcal{Z}}}$ is determined by the following equation*

$$\langle \boldsymbol{x}, \boldsymbol{\gamma}(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j)\rangle = \zeta(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j), \tag{9}$$

where

$$\boldsymbol{\gamma}(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j) := 2(\boldsymbol{r}_j(\bar{\boldsymbol{v}}_j) - \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i)) + 2\mathbf{P}(\bar{\boldsymbol{x}}_j - \bar{\boldsymbol{x}}_i), \tag{10a}$$

$$\begin{aligned}\zeta(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j) := &\sigma_j(\bar{\boldsymbol{v}}_j) - \sigma_i(\bar{\boldsymbol{v}}_i) + \langle \bar{\boldsymbol{x}}_j, 2\boldsymbol{r}_j(\bar{\boldsymbol{v}}_j) + \mathbf{P}\bar{\boldsymbol{x}}_j\rangle \\ &- \langle \bar{\boldsymbol{x}}_i, 2\boldsymbol{r}_i(\bar{\boldsymbol{v}}_i) + \mathbf{P}\bar{\boldsymbol{x}}_i\rangle. \end{aligned} \tag{10b}$$

PROOF. By definition, the bisector $\mathcal{B}_{\boldsymbol{\mathfrak{V}}}(\bar{\boldsymbol{z}}_i, \bar{\boldsymbol{z}}_j)$ consists of all the points $\boldsymbol{x}$ such that $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) = \pi(\boldsymbol{x};\bar{\boldsymbol{z}}_j)$, which implies

$$\begin{aligned}0 = &|\mathbf{P}^{1/2}\bar{\boldsymbol{x}}_i|^2 + |\mathbf{P}^{1/2}\boldsymbol{x}|^2 - 2\langle \boldsymbol{x}, \mathbf{P}\bar{\boldsymbol{x}}_i\rangle \\ &+ 2\langle \bar{\boldsymbol{x}}_i - \boldsymbol{x}, \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i)\rangle + \sigma_i(\bar{\boldsymbol{v}}_i) \\ &- |\mathbf{P}^{1/2}\bar{\boldsymbol{x}}_j|^2 - |\mathbf{P}^{1/2}\boldsymbol{x}|^2 + 2\langle \boldsymbol{x}, \mathbf{P}\bar{\boldsymbol{x}}_j\rangle \\ &- 2\langle \bar{\boldsymbol{x}}_j - \boldsymbol{x}, \boldsymbol{r}_j(\bar{\boldsymbol{v}}_j)\rangle - \sigma_j(\bar{\boldsymbol{v}}_j). \end{aligned} \tag{11}$$

It follows that

$$\begin{aligned}0 = &-2\langle \boldsymbol{x}, \mathbf{P}(\bar{\boldsymbol{x}}_i - \bar{\boldsymbol{x}}_j) + \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i) - \boldsymbol{r}_j(\bar{\boldsymbol{v}}_j)\rangle \\ &+ 2\langle \bar{\boldsymbol{x}}_i, \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i)\rangle - 2\langle \bar{\boldsymbol{x}}_j, \boldsymbol{r}_j(\bar{\boldsymbol{v}}_j)\rangle + \sigma_i(\bar{\boldsymbol{v}}_i) - \sigma_j(\bar{\boldsymbol{v}}_j) \\ &+ \langle \bar{\boldsymbol{x}}_i, \mathbf{P}\bar{\boldsymbol{x}}_i\rangle - \langle \bar{\boldsymbol{x}}_j, \mathbf{P}\bar{\boldsymbol{x}}_j\rangle, \end{aligned} \tag{12}$$

and the result follows readily. ■

Eq. (9) describes a hyperplane in $\mathbb{R}^4$, whose intersection with the flat manifold $\boldsymbol{\mathcal{Z}}_0$ determines a straight line that lies in $\boldsymbol{\mathcal{Z}}_0$.

Proposition 3 *Let $\overline{\boldsymbol{\mathcal{Z}}} := \{\bar{\boldsymbol{z}}_i = \mathrm{col}(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{v}}_i),\ i \in \mathcal{I}_n\}$. In addition, let $\boldsymbol{\mathfrak{V}} := \{\boldsymbol{\mathfrak{V}}_i,\ i \in \mathcal{I}_n\}$ be the generalized Voronoi partition of $\boldsymbol{\mathcal{Z}}_0$ that solves Problem 2. Moreover, let*

$$\bar{\boldsymbol{\xi}}_i := \bar{\boldsymbol{x}}_i + \mathbf{P}^{-1}\boldsymbol{r}_i(\bar{\boldsymbol{v}}_i), \tag{13a}$$

$$\mu_i := -\langle \boldsymbol{r}_i(\bar{\boldsymbol{v}}_i), \mathbf{P}^{-1}\boldsymbol{r}_i(\bar{\boldsymbol{v}}_i)\rangle + \sigma_i(\bar{\boldsymbol{v}}_i), \tag{13b}$$

where $\boldsymbol{r}_i(\bar{\boldsymbol{v}}_i)$ and $\sigma_i(\bar{\boldsymbol{v}}_i)$ are given by (8), and let $\boldsymbol{\mathfrak{R}} := \{\boldsymbol{\mathfrak{R}}_i,\ i \in \mathcal{I}_n\}$ be the partition of $\boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2$, which is generated by the point-set $\overline{\boldsymbol{\Xi}} := \{\bar{\boldsymbol{\xi}}_i \in \mathbb{R}^2,\ i \in \mathcal{I}_n\}$ with proximity metric $\rho(\cdot;\bar{\boldsymbol{\xi}}_i, \mu_i) : \boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2 \mapsto \mathbb{R}_{\geq 0}$, where

$$\rho(\boldsymbol{x};\bar{\boldsymbol{\xi}}_i, \mu_i) := \langle \boldsymbol{x} - \bar{\boldsymbol{\xi}}_i, \mathbf{P}(\boldsymbol{x} - \bar{\boldsymbol{\xi}}_i)\rangle + \mu_i. \tag{14}$$

Then, for any $\boldsymbol{x} \in \boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2$, the state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$, belongs to the cell $\boldsymbol{\mathfrak{V}}_i \in \boldsymbol{\mathfrak{V}}$ if, and only if, the point $\boldsymbol{x}$ belongs to $\boldsymbol{\mathfrak{R}}_i \in \boldsymbol{\mathfrak{R}}$. In addition, $\boldsymbol{\mathfrak{R}}$ is an affine diagram with combinatorial complexity $\Theta(n)$.

PROOF. Equations (8) and (14) imply that $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) = \rho(\boldsymbol{x};\bar{\boldsymbol{\xi}}_i, \mu_i)$, provided that (13a)-(13b) hold. By definition, a state $\boldsymbol{z}(\boldsymbol{x}) = \mathrm{col}(\boldsymbol{x}, 0)$, $\boldsymbol{z}(\boldsymbol{x}) \in \boldsymbol{\mathcal{Z}}_0$, belongs to the cell $\boldsymbol{\mathfrak{V}}_i \in \boldsymbol{\mathfrak{V}}$, if, and only if, $\pi(\boldsymbol{x};\bar{\boldsymbol{z}}_i) \leq \pi(\boldsymbol{x};\bar{\boldsymbol{z}}_j)$ or equivalently, $\rho(\boldsymbol{x};\bar{\boldsymbol{\xi}}_i, \mu_i) \leq \rho(\boldsymbol{x};\bar{\boldsymbol{\xi}}_j, \mu_j)$, for all $i \neq j$. Let $\mathcal{B}_{\boldsymbol{\mathfrak{R}}}(\bar{\boldsymbol{\xi}}_i, \bar{\boldsymbol{\xi}}_j)$ denote the bisector of $\boldsymbol{\mathfrak{R}}$ that corresponds to two distinct generators $\bar{\boldsymbol{\xi}}_i$, $\bar{\boldsymbol{\xi}}_j \in \overline{\boldsymbol{\Xi}}$, $i \neq j$. Then, a point $\boldsymbol{x} \in \boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2$ belongs to $\mathcal{B}_{\boldsymbol{\mathfrak{R}}}(\bar{\boldsymbol{\xi}}_i, \bar{\boldsymbol{\xi}}_j)$ if, and only if, $\rho(\boldsymbol{x};\bar{\boldsymbol{\xi}}_i, \mu_i) = \rho(\boldsymbol{x};\bar{\boldsymbol{\xi}}_j, \mu_j)$, which is equivalent to (9). Eq. (9) in turn implies that $\mathcal{B}_{\boldsymbol{\mathfrak{R}}}(\bar{\boldsymbol{\xi}}_i, \bar{\boldsymbol{\xi}}_j)$ is

a straight line in $\mathbb{R}^2$, for all the pairs of distinct generators $(\bar{\boldsymbol{\xi}}_i, \bar{\boldsymbol{\xi}}_j) \in \overline{\boldsymbol{\Xi}} \times \overline{\boldsymbol{\Xi}}$, $i \neq j$, of $\mathfrak{R}$. Consequently, $\mathfrak{R}$ is an affine Voronoi diagram in $\boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^2$. The result on the combinatorial complexity of $\mathfrak{R}$ follows immediately from Theorem 18.2.3 in [6, p. 439]. ■

3.2 A decentralized spatial partitioning algorithm

In many applications, it is important that every agent can compute its own cell independently from its teammates; in this way, every agent can use its limited computational resources frugally. This is desirable because the computation of the whole partition may be irrelevant to the objectives of an agent both at the individual and the team levels. In this section, we present a decentralized algorithm, which computes the partition $\mathfrak{R} = \{\mathfrak{R}^i,\ i \in \mathcal{I}_n\}$, which is equivalent, in the sense described in Proposition 3, to the generalized Voronoi diagram $\mathfrak{V} = \{\mathfrak{V}^i,\ i \in \mathcal{I}_n\}$ that solves Problem 2. We will assume that the partition space is a convex polygon $\mathcal{S}$, where $\mathcal{S} \subsetneq \mathbb{R}^2$, which is homeomorphic to the subset $\mathcal{S}_0$ of the terminal manifold $\boldsymbol{\mathcal{Z}}_0$, where $\mathcal{S}_0 := \{\boldsymbol{z} \in \boldsymbol{\mathcal{Z}} :\ \boldsymbol{z} = \mathrm{col}(\boldsymbol{x}, 0),\ \boldsymbol{x} \in \mathcal{S}\}$.

Next, we introduce an assumption that will allow us to compute approximations of the cells that comprise the partition $\mathfrak{R}$ by building upon the algorithm proposed in our previous work [2] for partitioning problems involving agents with single integrator kinematics. This assumption is not necessary for the decentralized computation of the spatial partition and will be removed later; however, when it is satisfied, the computation of $\mathfrak{R}$ is significantly simpler.

Assumption 1 *For all $i, j \in \mathcal{I}_n$, with $i \neq j$, it holds that $\rho(\boldsymbol{\xi}_i; \bar{\boldsymbol{\xi}}_i, \mu_i) < \rho(\bar{\boldsymbol{\xi}}_i; \bar{\boldsymbol{\xi}}_j, \mu_j)$, or equivalently,*

$$\mu_i < \langle \bar{\boldsymbol{\xi}}_i - \bar{\boldsymbol{\xi}}_j, \mathbf{P}(\bar{\boldsymbol{\xi}}_i - \bar{\boldsymbol{\xi}}_j) \rangle + \mu_j, \tag{15}$$

where $\bar{\boldsymbol{\xi}}_\ell$, μ_ℓ, $\ell \in \{i, j\}$, are defined as in (13a)-(13b).

Note that (15) along with the fact that the proximity metric ρ is a continuous function imply that the generator $\bar{\boldsymbol{\xi}}_i \in \overline{\boldsymbol{\Xi}}$ of the affine diagram $\mathfrak{R}$ is an interior point of its associated cell $\mathfrak{R}_i$, for all $i \in \mathcal{I}_n$; thus, every cell of $\mathfrak{R}$ has a non-empty interior. One should actually expect that it is likely, yet not necessary, that the point $\bar{\boldsymbol{\xi}}_i$ belongs to the interior of $\mathfrak{R}_i$ given that it corresponds, in view of (14), to the minimizer of the function $\rho(\cdot; \bar{\boldsymbol{\xi}}_i, \mu_i)$. Note that (15) also implies that $\bar{\boldsymbol{\xi}}_i \neq \bar{\boldsymbol{\xi}}_j$, for all $i \neq j$, for otherwise, $\mu_i < \mu_j$ and $\mu_j < \mu_i$ (by interchanging i and j in (15)), which is absurd.

Proposition 4 *Suppose that Assumption 1 holds and let $\mathfrak{R} = \{\mathfrak{R}_i,\ i \in \mathcal{I}_n\}$ be the affine diagram generated by the point-set $\overline{\boldsymbol{\Xi}} \subsetneq \mathbb{R}^2$, which is defined as in Proposition 3. Then, the set $\mathfrak{R}_i$, $\mathfrak{R}_i \in \mathfrak{R}$, is star-convex with respect to its generator $\bar{\boldsymbol{\xi}}_i$. In other words, the line segment $[\bar{\boldsymbol{\xi}}_i, \boldsymbol{x}]$ is a subset of $\mathfrak{R}_i$, that is, $[\bar{\boldsymbol{\xi}}_i, \boldsymbol{x}] \subsetneq \mathfrak{R}_i$, for all $i \in \mathcal{I}_n$ and for all $\boldsymbol{x} \in \mathfrak{R}_i$.*

PROOF. By definition, each cell $\mathfrak{R}_i \in \mathfrak{R}$ is determined by a finite intersection of closed half-spaces in $\mathbb{R}^2$ given that $\mathfrak{R}$ is an affine diagram in $\mathbb{R}^2$. Therefore, the cell $\mathfrak{R}_i$ is a convex set, for all $i \in \mathcal{I}_n$. In addition, (15) implies that $\bar{\boldsymbol{\xi}}_i \in \mathrm{int}(\mathfrak{R}_i)$; consequently, the interior of $\mathfrak{R}_i$ is non-empty, that is, $\mathrm{int}(\mathfrak{R}_i) \neq \varnothing$. Thus, by convexity, every line segment emanating from $\bar{\boldsymbol{\xi}}_i$ to any point $\boldsymbol{x} \in \mathfrak{R}_i$ is a subset of $\mathfrak{R}_i$, and the result follows. ■

Note that the fact that each cell $\mathfrak{R}_i \in \mathfrak{R}$ is convex, which was shown in the previous proof, holds true regardless of whether Assumption 1 is satisfied or not. Now let $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ denote the ray that emanates from $\bar{\boldsymbol{\xi}}_i$ and is parallel to a unit vector $\boldsymbol{e} \in \mathbb{S}^1$. The following proposition follows readily from the star-convexity of $\mathfrak{R}_i$.

Proposition 5 *Suppose that Assumption 1 holds and let $\mathfrak{R} = \{\mathfrak{R}_i :\ i \in \mathcal{I}_n\}$ be the affine diagram generated by the point-set $\overline{\boldsymbol{\Xi}} \subsetneq \mathbb{R}^2$, which is defined in Proposition 3. If there exists a point $\boldsymbol{y} \in \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ for which $\rho(\boldsymbol{y}; \bar{\boldsymbol{\xi}}_i, \mu_i) \leq \rho(\boldsymbol{y}; \bar{\boldsymbol{\xi}}_j, \mu_j)$, for all $j \in \mathcal{I}_n \backslash \{i\}$, then all the points that lie in the line segment from $\bar{\boldsymbol{\xi}}_i$ to $\boldsymbol{y}$ will belong to the cell $\mathfrak{R}_i$, that is, $[\bar{\boldsymbol{\xi}}_i, \boldsymbol{y}] \subsetneq \mathfrak{R}_i$. In addition, if $\rho(\boldsymbol{y}; \bar{\boldsymbol{\xi}}_i, \mu_i) = \rho(\boldsymbol{y}; \bar{\boldsymbol{\xi}}_\ell, \mu_\ell)$ for some $\ell \in \mathcal{I}_n \backslash \{i\}$, then the point $\boldsymbol{y}$ will belong to the boundary of $\mathfrak{R}_i$, that is, $\boldsymbol{y} \in \mathrm{bd}(\mathfrak{R}_i)$ and there is no point $\boldsymbol{\psi} \in \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ with $|\boldsymbol{\psi} - \bar{\boldsymbol{\xi}}_i| > |\boldsymbol{y} - \bar{\boldsymbol{\xi}}_i|$ such that $\rho(\boldsymbol{\psi}; \bar{\boldsymbol{\xi}}_i, \mu_i) \leq \rho(\boldsymbol{\psi}; \bar{\boldsymbol{\xi}}_j, \mu_j)$, for all $j \in \mathcal{I}_n \backslash \{i\}$; we write $\boldsymbol{y} = \boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$.*

The most important implications of Proposition 5 are the following: 1) If a point $\boldsymbol{x} \in [\bar{\boldsymbol{\xi}}_i, \boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)]$, then $\boldsymbol{x} \in \mathfrak{R}_i$, and 2) a point $\boldsymbol{y}$ belongs to $\mathrm{bd}(\mathfrak{R}_i)$ if, and only if, $\boldsymbol{y} = \boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$, for some $\boldsymbol{e} \in \mathbb{S}^1$; thus, $\mathrm{bd}(\mathfrak{R}_i) = \{\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) :\ \boldsymbol{e} \in \mathbb{S}^1\}$. Hence, in order to compute a discrete (finite) approximation of $\mathrm{bd}(\mathfrak{R}_i)$, we simply have to find a finite point-set $\{\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) :\ \boldsymbol{e} \in \boldsymbol{\mathcal{E}}^0\}$, where $\boldsymbol{\mathcal{E}}^0$ is a finite discretization of the unit circle $\mathbb{S}^1$. Next, we present the main steps of a decentralized partitioning algorithm that computes this discrete approximation of $\mathrm{bd}(\mathfrak{R}_i)$.

Step 0: Take $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}^0$ and set $\boldsymbol{\mathcal{E}} \leftarrow \boldsymbol{\mathcal{E}}^0$.

Step 1: Set $\boldsymbol{x}^{[0]}(\boldsymbol{e}) \leftarrow \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e}) \cap \mathrm{bd}(\mathcal{S})$ and $\rho^{[0]}(\boldsymbol{e}) \leftarrow \rho(\boldsymbol{x}^{[0]}(\boldsymbol{e}); \bar{\boldsymbol{\xi}}_i, \mu_i)$. At this step, we compute the intersection of $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ with the boundary $\mathrm{bd}(\mathcal{S})$ of the partition space $\mathcal{S}$.

Step 2: If $\rho^{[0]}(\boldsymbol{e}) \leq \rho(\boldsymbol{x}^{[0]}(\boldsymbol{e}); \bar{\boldsymbol{\xi}}_j, \mu_j)$, for all $j \in \mathcal{I}_n \backslash \{i\}$, set $\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \leftarrow \boldsymbol{x}^{[0]}(\boldsymbol{e})$ and $\boldsymbol{\mathcal{E}} \leftarrow \boldsymbol{\mathcal{E}} \backslash \{\boldsymbol{e}\}$. End if. In this case, the query point belongs to the boundary $\mathrm{bd}(\mathcal{S})$.

Step 3: Set $\boldsymbol{\mathcal{E}}^0 \leftarrow \boldsymbol{\mathcal{E}}^0 \backslash \{\boldsymbol{e}\}$. If $\boldsymbol{\mathcal{E}}^0 \neq \varnothing$, take $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}^0$ and go to **Step 1**. End if.

We then proceed with the line search process along the rays $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ for the unit vectors $\boldsymbol{e}$ for which $\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \notin \mathrm{bd}(\mathcal{S})$, that is, for all $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}$.

Step 4: Pick an error tolerance $\varepsilon > 0$ and then start the following iterative process:

```
%% Begin Iterative Process 1
for e ∈ E
    k ← 1
    x^[k](e) ← x^[k−1](e) − 1/2|x^[k−1](e) − ξ̄_i|e,
```

$\rho^{[k]}(\boldsymbol{e}) \leftarrow \rho(\boldsymbol{x}^{[k]}(\boldsymbol{e}); \bar{\boldsymbol{\xi}}_i, \mu_i)$,
$\Delta\boldsymbol{x}^{[k]}(\boldsymbol{e}) \leftarrow \boldsymbol{x}^{[k]}(\boldsymbol{e}) - \boldsymbol{x}^{[k-1]}(\boldsymbol{e})$
while $\left|\Delta\boldsymbol{x}^{[k]}(\boldsymbol{e})\right| > \varepsilon$
$k \leftarrow k+1$
if $\rho^{[k-1]}(\boldsymbol{e}) \leq \rho(\boldsymbol{x}^{[k-1]}(\boldsymbol{e}); \bar{\boldsymbol{\xi}}_j, \mu_j)$, $\forall j \in \mathcal{I}_n \backslash \{i\}$
$\boldsymbol{x}^{[k]}(\boldsymbol{e}) \leftarrow \boldsymbol{x}^{[k-1]}(\boldsymbol{e}) + 1/2|\Delta\boldsymbol{x}^{[k-1]}(\boldsymbol{e})|\boldsymbol{e}$
else
$\boldsymbol{x}^{[k]}(\boldsymbol{e}) \leftarrow \boldsymbol{x}^{[k-1]}(\boldsymbol{e}) - 1/2|\Delta\boldsymbol{x}^{[k-1]}(\boldsymbol{e})|\boldsymbol{e}$
end
$\rho^{[k]}(\boldsymbol{e}) \leftarrow \rho(\boldsymbol{x}^{[k]}(\boldsymbol{e}); \bar{\boldsymbol{\xi}}_i, \mu_i)$
$\Delta\boldsymbol{x}^{[k]}(\boldsymbol{e}) \leftarrow \boldsymbol{x}^{[k]}(\boldsymbol{e}) - \boldsymbol{x}^{[k-1]}(\boldsymbol{e})$
end
$\boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \leftarrow \boldsymbol{x}^{[k]}(\boldsymbol{e})$
end
%% End Iterative Process 1

The output of the previous iterative process is an approximation of the point $\boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ for each unit vector $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}$. Furthermore, the output of the previous algorithm is an approximation of the actual cell $\mathfrak{R}_i$ (or more precisely, of its boundary); we will denote this approximation by $\mathfrak{R}_i^{\star}$, when it is important to distinguish it from the exact cell $\mathfrak{R}_i$.

Note that the implementation of the previous algorithm requires that the i-th agent has knowledge of the location $\bar{\boldsymbol{x}}_j$ and the velocity $\bar{\boldsymbol{v}}_j$ of the j-th agent, and consequently, the point $\bar{\boldsymbol{\xi}}_j$, for every $j \in \mathcal{I}_n \backslash \{i\}$. However, in principle, it suffices for the i-th agent to know the point $\bar{\boldsymbol{\xi}}_i$ for every $j \in \mathcal{N}(i, \mathfrak{R})$, where $\mathcal{N}(i; \mathfrak{R})$ denotes the set comprised of the indices of the neighbors of the i-th agent in the affine diagram $\mathfrak{R}$, that is, all the agents whose cells in $\mathfrak{R}$ share a face with the cell $\mathfrak{R}_i$. Note that, by the definition of $\mathfrak{R}$, we have that $\mathcal{N}(i; \mathfrak{R}) = \mathcal{N}(i; \mathfrak{V})$. In the presence of sensing constraints, the i-th agent can only be assigned points of $\mathcal{S}$ that lie inside the closed ball $\overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i) := \{\boldsymbol{x} \in \mathbb{R}^2 : \ |\boldsymbol{x} - \bar{\boldsymbol{x}}_i| \leq \eta\}$, where η is a positive constant that corresponds to the *sensing radius* of the i-th agent. In addition, the i-th agent can only infer the locations and the velocities of the agents that lie inside $\overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i)$. In this case, the partitioning algorithm we previously described should be modified accordingly such that its output is the intersection of the cell $\mathfrak{R}_i$ with $\overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i)$. To this aim, it may be tempting for one to simply modify the first step of the partitioning algorithm by setting $\boldsymbol{x}^{[0]}(\boldsymbol{e}) \leftarrow \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e}) \cap \mathrm{bd}(\overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i))$, instead of $\boldsymbol{x}^{[0]}(\boldsymbol{e}) \leftarrow \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e}) \cap \mathrm{bd}(\mathcal{S})$, and proceed similarly henceforth. This approach will not, in general, give a correct approximation of $\mathfrak{R}_i \cap \overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i)$ given that $\overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i)$ does not necessarily contain all the neighbors of the i-th agent in $\mathfrak{R}$. In practice, the implementation of the previous algorithm in a distributed fashion requires that the sensing radius η of the i-th agent be sufficiently large such that $\bar{\boldsymbol{x}}_j \in \overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i)$ for all $j \in \mathcal{N}(i, \mathfrak{R})$ [7]. Only then the previous modification of the partitioning algorithm will produce a "correct" approximation of $\mathfrak{R}_i \cap \overline{\mathcal{B}}_{\eta}(\bar{\boldsymbol{x}}_i)$.

3.3 A modified partitioning algorithm

As we have already mentioned, Assumption 1 is not necessary for the decentralized computation of the affine diagram $\mathfrak{R}$. In particular, if the point $\bar{\boldsymbol{\xi}}_i$ is not an interior point of $\mathfrak{R}_i$, then of course $\mathfrak{R}_i$ cannot be star convex with respect to $\bar{\boldsymbol{\xi}}_i$. However, $\mathfrak{R}_i$ is a cell of an affine diagram and thus a convex set regardless of whether it is true or not that $\bar{\boldsymbol{\xi}}_i \in \mathrm{int}(\mathfrak{R}_i)$.

Proposition 6 *Let $E(\mathfrak{R}_i) := \{E_i^{\ell}, \ \ell \in \mathcal{L}(\mathfrak{R}_i)\}$, where $\mathcal{L}(\mathfrak{R}_i)$ is some finite index set, be the collection of edges E_i^{ℓ} that comprise the boundary $\mathrm{bd}(\mathfrak{R}_i)$ of the cell $\mathfrak{R}_i$ of the affine diagram $\mathfrak{R}$. In addition, let the interior of the cell $\mathfrak{R}_i$ be non-empty, that is, $\mathrm{int}(\mathfrak{R}_i) \neq \varnothing$, and assume that $\bar{\boldsymbol{\xi}}_i \notin \mathrm{int}(\mathfrak{R}_i)$. Then, if for some $\boldsymbol{e} \in \mathbb{S}^1$ the ray $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ intersects $\mathrm{bd}(\mathfrak{R}_i)$, there are at most two edges from $E(\mathfrak{R}_i)$ whose intersection with $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ is non-empty.*

PROOF. If intersection of $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ with $\mathrm{bd}(\mathfrak{R}_i)$ is non-empty, then there are three possibilities: 1) $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ intersects $\mathrm{bd}(\mathfrak{R}_i)$ at a single point, which is a vertex of $\mathfrak{R}_i$, 2) $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ intersects $\mathrm{bd}(\mathfrak{R}_i)$ at two points that belong to two different edges of $E(\mathfrak{R}_i)$, and 3) there exists an edge $E_i^{\ell} \in E(\mathfrak{R}_i)$ such that $E_i^{\ell} \subseteq \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$. Note that $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ cannot intersect $\mathrm{bd}(\mathfrak{R}_i)$ at more than two points that belong to different edges; this would violate the convexity of the cell $\mathfrak{R}_i$ (as we have already mentioned, the cell $\mathfrak{R}_i$ is convex regardless of whether (15) holds true or not). ■

Proposition 6 implies the following: If the intersection of $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ with $\mathfrak{R}_i$ is non-empty for some $\boldsymbol{e} \in \mathbb{S}^1$, then it necessarily corresponds to a line segment $[\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i), \boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)]$, where the point $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ does not always coincide with the point $\bar{\boldsymbol{\xi}}_i$, and, in addition, $\boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \in \mathrm{bd}(\mathfrak{R}_i)$. Specifically, if $\bar{\boldsymbol{\xi}}_i$ is an interior point of $\mathfrak{R}_i$ (Assumption 1 holds true), then $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) = \bar{\boldsymbol{\xi}}_i$; otherwise, $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \in \mathrm{bd}(\mathfrak{R}_i)$.

Proposition 7 *Suppose that $\mathfrak{R}_i$ has a nonempty interior, the intersection of a ray $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ with $\mathfrak{R}_i$ is nonempty for some $\boldsymbol{e} \in \mathbb{S}^1$, and $\bar{\boldsymbol{\xi}}_i \notin \mathrm{int}(\mathfrak{R}_i)$. Let $\boldsymbol{y}$, $\boldsymbol{y}'$ be two points (not necessarily distinct) in $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$, such that there are no other points $\boldsymbol{\psi}$, $\boldsymbol{\psi}' \in \Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ with $|\boldsymbol{\psi} - \bar{\boldsymbol{\xi}}_i| < |\boldsymbol{y} - \bar{\boldsymbol{\xi}}_i|$ and $|\boldsymbol{\psi}' - \bar{\boldsymbol{\xi}}_i| > |\boldsymbol{y}' - \bar{\boldsymbol{\xi}}_i|$, respectively, such that $\rho(\boldsymbol{\psi}; \bar{\boldsymbol{\xi}}_i, \mu_i) \leq \rho(\boldsymbol{\psi}; \bar{\boldsymbol{\xi}}_j, \mu_j)$ and $\rho(\boldsymbol{\psi}'; \bar{\boldsymbol{\xi}}_i, \mu_i) \leq \rho(\boldsymbol{\psi}'; \bar{\boldsymbol{\xi}}_j, \mu_j)$, for all $j \in \mathcal{I}_n \backslash \{i\}$. Then the line segment $[\boldsymbol{y}, \boldsymbol{y}']$ will be a subset of $\mathfrak{R}_i$ and the points $\boldsymbol{y}$, $\boldsymbol{y}'$ will belong to $\mathrm{bd}(\mathfrak{R}_i)$; we write $\boldsymbol{y} = \boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ and $\boldsymbol{y}' = \boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$.*

Note that when $\boldsymbol{y}$ and $\boldsymbol{y}'$ are not distinct then the point $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \equiv \boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ corresponds to a vertex of $\mathfrak{R}_i$. If $\boldsymbol{y}$ and $\boldsymbol{y}'$ are distinct points, there are two possibilities. In particular, if $\bar{\boldsymbol{\xi}}_i \in \mathrm{int}(\mathfrak{R}_i)$, then $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) = \bar{\boldsymbol{\xi}}_i$ and we proceed as in Section 3.2. If, however, $\bar{\boldsymbol{\xi}}_i \notin \mathrm{int}(\mathfrak{R}_i)$, then, in view of Proposition 7, it is possible that both $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ and $\boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ belong to $\mathrm{bd}(\mathfrak{R}_i)$, for some $\boldsymbol{e} \in \mathbb{S}^1$. Specifically, in order to compute $\boldsymbol{x}^{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$, one can apply the Iterative Process 1 described in Section 3.2; the same process can be used for the computation of $\boldsymbol{x}_{\star}(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$, after the necessary modifications have

been carried out. In particular, the new objective is to find the closest point to $\bar{\boldsymbol{\xi}}_i$ that belongs to $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$, for which $\rho(\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i); \bar{\boldsymbol{\xi}}_i, \mu_i) \leq \rho(\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i); \bar{\boldsymbol{\xi}}_j, \mu_j)$, for all $j \in \mathcal{I}_n \backslash \{i\}$. The main steps of this iterative process are given next.

Step 0: Set $\boldsymbol{\mathcal{E}}_0 \leftarrow \boldsymbol{\mathcal{E}}^0$.

Step 1: For all $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}^0$, compute $\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ using the algorithm presented in Section 3.2. Then, if Assumption 1 is verified, set $\boldsymbol{\mathcal{E}}_0 \leftarrow \boldsymbol{\mathcal{E}}_0 \backslash \{\boldsymbol{e}\}$ and $\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \leftarrow \bar{\boldsymbol{\xi}}_i$; otherwise, set $\boldsymbol{x}_{[0]}(\boldsymbol{e}) \leftarrow \boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$. Note that because $\bar{\boldsymbol{\xi}}_i$ is not necessarily an interior point of $\boldsymbol{\mathfrak{R}}_i$, it is possible that the ray $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ does not intersect at all with the cell $\boldsymbol{\mathfrak{R}}_i$ for some $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}^0$, in which case we say that $\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ is a null vector. In this case, $\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ is also a null vector and we set $\boldsymbol{\mathcal{E}}_0 \leftarrow \boldsymbol{\mathcal{E}}_0 \backslash \{\boldsymbol{e}\}$.

Step 2: Pick an error tolerance $\varepsilon > 0$ and apply, for all $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}_0$, a similar process with the Iterative Process 1 after making the following substitutions and modifications: 1) Set $\boldsymbol{x}_{[k]}(\boldsymbol{e}) \leftarrow \boldsymbol{x}^{[k]}(\boldsymbol{e})$, $\Delta\boldsymbol{x}_{[k]}(\boldsymbol{e}) \leftarrow \Delta\boldsymbol{x}^{[k]}(\boldsymbol{e})$, $\rho_{[k]}(\boldsymbol{e}) \leftarrow \rho^{[k]}(\boldsymbol{e})$, $\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i) \leftarrow \boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ and 2) flip the signs before $\Delta\boldsymbol{x}_{[k]}(\boldsymbol{e})$ in the two expressions for $\boldsymbol{x}_{[k]}(\boldsymbol{e})$ in the **if-else-end** statement. At the end of this process, the finite sequence of points $\boldsymbol{x}_{[k]}(\boldsymbol{e})$ will converge to (an approximation) of $\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$.

Remark 2 The performance of the previous baseline algorithm can be improved by a number of modifications that are aimed at reducing the number of line searches in the case when $\bar{\boldsymbol{\xi}}_i$ is not an interior point of $\boldsymbol{\mathfrak{R}}_i$. For example, when there are two consecutive unit vectors $\boldsymbol{e}_k$ and $\boldsymbol{e}_{k+1} \in \boldsymbol{\mathcal{E}}_0$ such that both $\boldsymbol{x}^\star(\boldsymbol{e}_k, \bar{\boldsymbol{\xi}}_i)$ and $\boldsymbol{x}^\star(\boldsymbol{e}_{k+1}, \bar{\boldsymbol{\xi}}_i)$ are null vectors, then one can find a unit vector $\boldsymbol{e}'$ that is parallel to a convex combination of $\boldsymbol{e}_k$ and $\boldsymbol{e}_{k+1}$ and determines a line passing through $\bar{\boldsymbol{\xi}}_i$ separating $\mathbb{R}^2$ into two half-spaces such that the cell $\boldsymbol{\mathfrak{R}}_i$ is completely contained in one of them exclusively. In this case, each vector $\boldsymbol{e} \in \boldsymbol{\mathcal{E}}_0$ for which the ray $\Gamma(\bar{\boldsymbol{\xi}}_i, \boldsymbol{e})$ lies in the half-space that does not contain the cell $\boldsymbol{\mathfrak{R}}_i$ should not be considered at all for the iterative process given that both $\boldsymbol{x}^\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ and $\boldsymbol{x}_\star(\boldsymbol{e}, \bar{\boldsymbol{\xi}}_i)$ will be null vectors. Alternatively, one can utilize techniques aiming at identifying a point $\boldsymbol{\chi}_i$ different than $\bar{\boldsymbol{\xi}}_i$ that belongs to the interior of $\boldsymbol{\mathfrak{R}}_i$, and then utilize a similar algorithm to that described in Section 3.2. Due to space limitations, we will not present in detail all these possible modifications.

4 Numerical Simulations

In this section, we present numerical simulations that illustrate the theoretical developments presented so far. For our simulations, we consider a team of twenty mobile agents with double integrator (open-loop) kinematics. This choice of kinematics would allow us to compare the results of this work and our previous work on partitioning problems with teams of agents with linear dynamics [1]. For our simulations we have taken $\gamma = 1.4$, $\rho = 0.6$, $\mathbf{K}_1 = 2\mathbf{I}_2$, $\mathbf{K}_2 = 3\mathbf{I}_2$, $\boldsymbol{\Lambda} = \mathbf{P} = \mathbf{I}_2$, whereas the locations of the agents are selected randomly.

Figure 1 illustrates the affine Voronoi diagram that solves Problem 2 in vector geometry representation, which was computed, for verifications purposes, via a standard centralized, "exact," algorithm. The "cross" markers in Fig. 1 denote the points $\bar{\boldsymbol{\xi}}_i \in \overline{\boldsymbol{\Xi}}$, whereas the initial positions $\bar{\boldsymbol{x}}_i \in \overline{\boldsymbol{\mathcal{X}}}$ are denoted by the "plus sign" markers. The initial velocities of the agents are denoted by black arrows. We observe that all the points $\bar{\boldsymbol{\xi}}_i \in \overline{\boldsymbol{\Xi}}$ belong to the interior of their associated cells; something that is not true for all the points $\bar{\boldsymbol{x}}_i \in \overline{\boldsymbol{\mathcal{X}}}$. Note that the partition illustrated in Fig. 1, which is an affine diagram, enjoys similar structural properties with the one that would have been obtained if T were taken to be uniformly constant throughout $\mathcal{S}$ and the proximity metric were taken to be the minimum control effort, as in [1]. Note that the terminal time herein is not prescribed a priori and it actually turns out to be state-dependent.

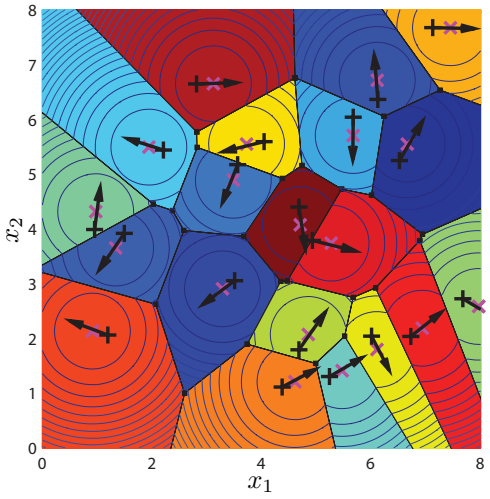

Fig. 1. The partition that solves Problem 2 for $n = 20$ generators.

Next, we compute the affine diagram $\boldsymbol{\mathfrak{R}}$ associated with the solution to the Problem 2, by utilizing the decentralized partitioning algorithm presented in Section 3.2. In particular, Fig. 2 illustrates the approximations of two of the cells that comprise the affine diagram $\boldsymbol{\mathfrak{R}}$ computed independently via the proposed decentralized algorithm. For the computation of these cells, we have discretized $\mathbb{S}^1$ into a uniform grid graph, $\boldsymbol{\mathcal{E}}^0$, consisting of 80 nodes and we have taken $\varepsilon = 0.01$. We observe that the approximations of the two cells obtained with the decentralized algorithm are very close to the cells obtained with the exact (centralized) partitioning algorithm, despite the fact that the grid $\boldsymbol{\mathcal{E}}^0$ is rather coarse. An important observation here is that the points $\bar{\boldsymbol{\xi}}_8$ and $\bar{\boldsymbol{\xi}}_{18}$ belong to the interior of the cells $\boldsymbol{\mathfrak{R}}_8$ and $\boldsymbol{\mathfrak{R}}_{18}$, respectively; something that allows one to directly apply the proposed decentralized partitioning algorithm; by contrast, $\bar{\boldsymbol{x}}_8$ is not an interior point of $\boldsymbol{\mathfrak{R}}_8$, and thus application of the algorithm proposed in [2] to directly compute the cells of the partition $\boldsymbol{\mathfrak{V}}$ of $\boldsymbol{\mathcal{Z}}_0$ would not be possible.

Figure 3 illustrates the performance of the proposed decentralized partitioning algorithm, in terms of the execu-

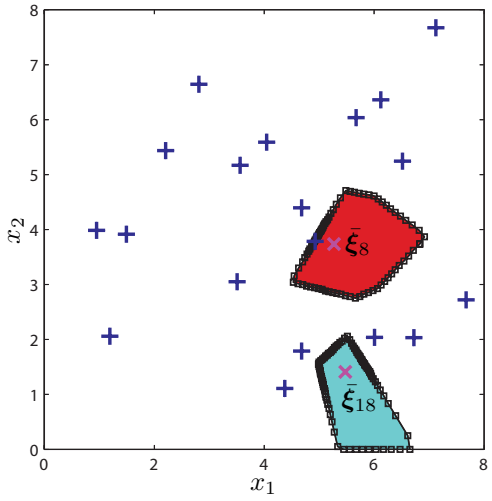

Fig. 2. Discrete approximations of $\mathfrak{R}_8$ and $\mathfrak{R}_{18}$ computed independently by their corresponding agents via the proposed decentralized partitioning algorithm using a grid $\boldsymbol{\mathcal{E}}^0$ with $|\boldsymbol{\mathcal{E}}^0| = 80$.

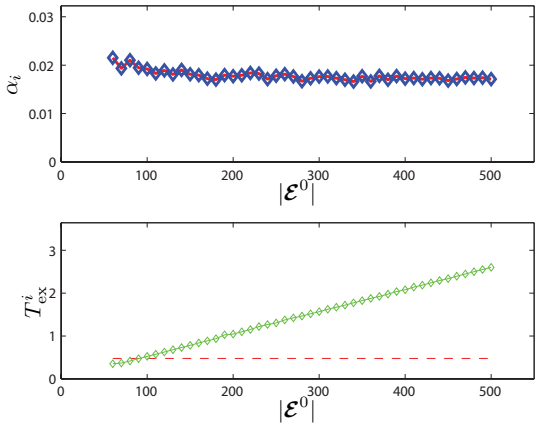

Fig. 3. Performance of the decentralized algorithm measured in terms of the approximation error and the actual computation time for different sizes of the grid $\boldsymbol{\mathcal{E}}^0$.

tion time T^i_{ex} and the convergence error $\alpha_i := |\mathcal{A}(\mathfrak{R}^\star_i) - \mathcal{A}(\mathfrak{R}_i)|/\mathcal{A}(\mathfrak{R}_i)$. We observe that the execution time of the decentralized algorithm T^i_{ex} depends linearly on the size of the grid. The dashed line in the same figure corresponds to the running time of the centralized algorithm that computes the cells of the partition represented in vector geometry, which is illustrated in Fig. 1. We observe that even with a relative coarse grid $\boldsymbol{\mathcal{E}}^0$, we can satisfactorily approximate the cell $\mathfrak{R}_i$ of $\mathfrak{R}$ with a discrete set $\mathfrak{R}^\star_i$, which can be computed via the proposed decentralized algorithm with an execution time that is comparable to that of the exact centralized algorithm.

5 Conclusion

In this paper, we have addressed a spatial partitioning problem involving a team of mobile agents with nonlinear second order dynamics. The proximity metric of the proposed spatial partition was taken to be the reduction of a generalized energy metric that takes place during the transfer of each agent to an arbitrary destination with zero terminal velocity (soft landing) in finite time. In particular, the arrival time is a state-dependent quantity whose functional description is not known a priori in contrast with our previous work on partitioning problems for multi-agent systems with linear dynamics, in which it was a prescribed constant throughout the partition space. We have shown that the solution to the partitioning problem considered herein is associated with a class of affine diagrams, which can be computed by means of simple partitioning algorithms that allow the agents to compute their own cells independently from their teammates (decentralized partitioning algorithms) provided that their sensing radii are sufficiently large.

References

[1] E. Bakolas. Optimal partitioning for multi-vehicle systems using quadratic performance criteria. *Automatica*, 49(11):3377–3383, 2013.

[2] E. Bakolas. Decentralized spatial partitioning for multi-vehicle systems in spatiotemporal flow-field. *Automatica*, 50(9):2389–2396, 2014.

[3] E. Bakolas and P. Tsiotras. The Zermelo-Voronoi diagram: a dynamic partition problem. *Automatica*, 46(12):2059–2067, 2010.

[4] E. Bakolas and P. Tsiotras. Optimal partitioning for spatiotemporal coverage in a drift field. *Automatica*, 49(7):2064–2073, 2013.

[5] S. Bhat and D. Bernstein. Finite-time stability of continuous autonomous systems. *SIAM Journal on Control and Optimization*, 38(3):751–766, 2000.

[6] J-D. Boissonnat and M. Yvinec. *Algorithmic Geometry*. Cambridge University Press, Cambridge, United Kingdom, 1998.

[7] J. Cortés, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004.

[8] Y. Feng, X. Yu, and Z. Man. Non-singular terminal sliding mode control of rigid manipulators. *Automatica*, 38(12):2159–2167, 2002.

[9] W. M. Haddad and V. Chellaboina. *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton Univ. Press, Princeton, NJ, USA, 2008.

[10] Y. Hong, Y. Xu, and J. Huang. Finite-time control for robot manipulators. *Systems & Control Letters*, 46(4):243–253, 2002.

[11] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. John Wiley and Sons Ltd, West Sussex, England, second edition, 2000.

[12] D. Reem. An algorithm for computing Voronoi diagrams of general generators in general normed spaces. In *ISVD '09*, pages 144 –152, June 2009.

[13] Y. Ru and S. Martinez. Coverage control in constant flow environments based on a mixed energy-time metric. *Automatica*, 49(9):2632–2640, 2013.

[14] J-J. E. Slotine and W. Li. On the adaptive control of robot manipulators. *International Journal of Robotics Research*, 6(3):49–59, 1987.

[15] M. W. Spong and M. Vidyasagar. *Robot Dynamics and Control*. John Wiley and Sons, NJ, USA, 1989.

[16] S. Yu, X. Yu, B. Shirinzadeh, and Z. Man. Continuous finite-time control for robotic manipulators with terminal sliding mode. *Automatica*, 41(11):1957–1964, 2005.