Higher order mobile coverage control with applications to clustering of discrete sets

Bomin Jiang, Zhiyong Sun, Brian D.O. Anderson, Christian Lageman

Data61 - CSIRO and Research School of Engineering, The Australian National University, Canberra, Australia
Laboratory for Information and Decision Systems, Institute for Data, Systems and Society, Massachusetts Institute of Technology, United States
School of Automation, Hangzhou Dianzi University, Hangzhou 310018, China
Department of Automatic Control, Lund University, Sweden
Institute of Mathematics, University of Würzburg, Würzburg, Germany

Abstract

Most current results on coverage control using mobile sensors require that one partitioned cell is associated with precisely one sensor. In this paper, we consider a class of coverage control problems involving higher order Voronoi partitions, motivated by applications where more than one sensor is required to monitor and cover one cell. Such applications are frequent in scenarios requiring the sensors to localize targets. We introduce a framework depending on a coverage performance function incorporating higher order Voronoi cells and then design a gradient-based controller which allows the multi-sensor system to achieve a local equilibrium in a distributed manner. The convergence properties are studied and related to Lloyd algorithm. We study also the extension to coverage of a discrete set of points and its applications to clustering of discrete sets.

1. Introduction

Consider a closed region, for convenience convex and polygonal. Suppose that across this region, individuals are distributed, in general nonuniformly. Computational geometry (De Berg, Van Kreveld, Overmars, & Schwarzkopf, 2000) then deals with questions such as: how can n supermarkets be located so that the sum of the trip distances for every individual to his or her nearest supermarket is minimized? (Such a problem is termed a ‘post office’ problem in De Berg et al., 2000). See also Clarkson (1985).

In recent years, mobile versions of this sort of problem have been considered, where there is a set of mobile agents that must cooperate to find optimum locations to extremize an index of the supermarket location type. Indeed, a fundamental problem is the optimal positioning of agents (where agents may refer to mobile sensors or autonomous vehicles) to cover an area in a way that some predefined coverage performance function can be optimized. This performance function can be related to the quality of service of a mobile sensing network, or the cumulative probability of certain events detected by sensors in the area of interest. This type of mobile coverage control problem has been studied extensively in literature (Cortés, Martínez, Karatas, & Bullo, 2004) for the case where only a single agent is required to cover each point in the area of interest. Our goal in this paper is to deal with the much less studied problem of multiple coverage.

Now consider the following variant on the supermarket problem (Martin & Firm, 2009): where should the n supermarkets be located in order that the sum over all individuals of the sum of the distances to the two closest supermarkets is minimized? (The social motivation was to ensure that no one supermarket had an approximation to a locational monopoly). This is an example of a higher order coverage problem. Naturally too, one can conceive of a version in which there are mobile agents, rather than supermarkets, and the agents must cooperate to reach their optimal positions. This generalization is motivated by many real-world applications. In the full length version of this paper posted at arXiv (Jiang, Sun, Anderson, & Lageman, 2018), we provide a number of real world scenarios where our variant is relevant.

© 2019 Elsevier Ltd. All rights reserved.
framework can be applied, including the supermarket problem, mobile coverage with TDOA sensors and bearing-only sensors, and bi-static radars. The usual current framework for considering coverage control with a single agent is not immediately suitable for the above applications.

The typical current mobile coverage control (involving single agent coverage of a given cell) problem is usually solved by using the geometric tool of Voronoi partitions (De Berg et al., 2000). When using this tool, each agent is thought of as monitoring a convex area whose shape is determined in part by information from its neighbouring agents. Furthermore, each agent moves according to some gradient-based control law to achieve optimization of the coverage performance criterion. There is also some literature treating the k-coverage problems, e.g., Li and Kao (2010) and So and Ye (2005). In the paper (Li & Kao, 2010) it converts the k-coverage problems to a 1-coverage problem, then uses a distributed k-coverage self-location estimation (DSLE) scheme based on the Voronoi diagram to implement coverage. Our paper, on the contrary, tackles the k-coverage problem directly using the concept of higher-order Voronoi diagrams. In another direction, focusing on k-coverage problems, e.g., Li and Kao (2010) and the concept of the Voronoi diagram to implement coverage. Our paper, on the contrary, tackles the k-coverage problem directly using the concept of higher-order Voronoi diagrams. In another direction, focusing on computational complexity analysis, the paper (So & Ye, 2005) considers the problem of verifying k-coverage of a given region with fixed-position sensors. However, mobile coverage control, which is the emphasis of our paper, is not studied in So and Ye (2005).

In this paper, we will use the concept of a higher order Voronoi partition as the main tool for tackling the problem. This concept is not a recent idea. The book Okabe, Boots, Sugihara, and Chiu (2009) mentioned the concept of a higher order Voronoi partition and the paper Li, Luo, Wang, and He (2015) considered a wireless sensor optimization problem requiring use of the concept. Nevertheless, mobile coverage control using higher-order Voronoi partitions is novel to the best of our knowledge. One of the techniques for solving one class of conventional (first order) Voronoi partition problems (those which specifically work with the square of Euclidean distance) is known as Lloyd's algorithm. For such first order problems, the optimal solution results in the so-called generator of each Voronoi set (the term is explained below) being located at the centroid of that set. The solution algorithm, which is not a basis for determining smooth motions of mobile sensors but simply an algorithm for determining optimal positions which they should attain, proceeds iteratively. We study in this paper a higher order version of this algorithm.

Some of the ideas of this paper were introduced in a much more limited predecessor work, Jiang, Sun, and Anderson (2015). However new material going well beyond the predecessor work is included here. In particular, we include discussions on higher order Lloyd algorithms as just mentioned, as well as convergence issues and discrete set coverage (with more details in Jiang et al., 2018). Due to the page length limit, we also move some detailed discussions on a related problem on minimum sensing radius, the stability analysis, collision avoidance, simulation examples and applications to the full version (Jiang et al., 2018).

The rest of this paper is organized as follows. Section 2 reviews the problem settings and current results on the coverage control problem, and further presents a brief introduction to order k Voronoi partitions. Section 3 discusses mainly the order 2 coverage control problem, the controller design, and the stability analysis of the coverage sensor system. We also discuss the Lloyd algorithm, convergence issues, and the computation of gradients and Hessians in this section. Section 4 deals with coverage problems for finite sets, with no corresponding material having been included in the predecessor paper. This generalizes what is generally known as k-means clustering. We refer the readers to Jiang et al. (2018) for several simulation results that demonstrate the coverage properties of various controllers. Finally, Section 5 concludes this paper.

2. Preliminary and background literature

This section gives a review of current order-1 mobile coverage control, followed by the tools of higher order Voronoi diagrams and the Lloyd algorithm. Furthermore, a variant on using the earlier performance index is also considered.

2.1. Order k Voronoi partition

Suppose there is a 2-D convex area Q to be covered by n mobile sensors in this area. A point in Q is denoted as q and sensor i’s position is denoted by $p_i \in \mathbb{R}^2$. A coverage performance function $H(p_1, p_2, \ldots, p_n)$ is typically defined as follows

$$H(p_1, p_2, \ldots, p_n) = \int_{\mathbb{R}^2} \min_{i \in \{1, \ldots, n\}} f(||q - p_i||) \phi(q) dq$$

(1)

where $\phi$ is a distribution density function known to all sensors which is assumed to be $C^2$ (i.e. two times continuously differentiable), $||q - p_i||$ denotes the Euclidean distance between q and $p_i$, and the function $f(||q - p_i||)$ describes the measurement cost or quantitative measurement assessment of how poor the sensing performance is at a point $q$ by a sensor at $p_i$. Thus larger values correspond to poorer sensing. We also suppose that $f$ should be $C^2$ and monotonically increasing. The coverage control aims to minimize the above performance function and to find and achieve the corresponding optimal positions of (mobile) agents.

As noted already, most literature on coverage control, such as the work reviewed in Section 1, assumes that each sensor is responsible for sensing or monitoring its own region, and we have an order 1 coverage control problem. Now we generalize the problem to a higher order coverage problem such that each cell is defined by two or more sensors. The general literature of order k Voronoi partitions includes Aurenhammer (1991) and Boissonnat, Devillers, and Teillaud (1993). Mature algorithms are also available to compute the order k Voronoi partition of a given area; see the survey paper Aurenhammer (1991) or Agarwal, De Berg, Matousek, and Schwarzkopf (1998). Note that most algorithms reported in these earlier papers are centralized ones, but a distributed algorithm has appeared in some recent papers (see e.g. Li et al., 2015).

The definition of an order k Voronoi partition of a convex area Q is given below following Agarwal et al. (1998). Let $S$ be a finite set of sensors’ positions in Q. Suppose further that $\mathcal{T}$ is a subset of $S$ and there are k elements in $\mathcal{T}$. The generalized Voronoi partition is defined by the collection of subsets of $Q$:

$$\mathcal{V}_T = \{q|\forall v \in \mathcal{T}, \forall w \in S \setminus \mathcal{T}, ||q, v|| \leq ||q, w||, |\mathcal{T}| = k\}$$

(2)

where $S \setminus \mathcal{T}$ denotes the relative complement of $\mathcal{T}$ with respect to $S$. For each point $q$ in $\mathcal{V}_T$, $q$ is not further to any sensor in $\mathcal{T}$ than to any sensor not in $\mathcal{T}$. The set $\mathcal{T}$ is the generator or generating set of $\mathcal{V}_T$.

In this paper we largely focus on the order 2 coverage problem. There are some similar properties and some dissimilar properties in higher-order partitions in comparison to order 1 Voronoi partitions, and we refer the reader to Jiang et al. (2018) for a list of interesting similarities.

For each $p_i \in S$, there are some $\mathcal{T}$ that contain $p_i$. We put all these $\mathcal{T}$ into a set $\mathcal{P}_i$ so that $\mathcal{P}_i = \{\mathcal{T} | \mathcal{T} \subset S, p_i \in T\}$ and then the union of cells with the sensor $p_i$ common to all cells is $W_i = \cup_{\mathcal{T} \in \mathcal{P}_i} \mathcal{V}_T$. It is noticeable that when we put these $\mathcal{T}$ together, we will not obtain the same cell containing $p_i$ in the order 1 partition. In fact, there holds $p_i \in V_s \subset W_i$. In addition, according to the definition of higher order Voronoi partition, $V_s$ and $\mathcal{V}_T$ are both always convex but $W_i$ may not be convex. An illustration is given in Fig. 1, where the red contour comprises $W_2$. Other properties of higher order Voronoi partitions can be found in Chazelle and Edelsbrunner (1987) and Lee (1982).
2.2. Centroidal Voronoï partitions and the Lloyd algorithm

For order 1 Voronoï partitions, given the choice \( f(q,p_i) = \|q - p_i\|^2 \), it is easily checked that for a fixed partition of the set \( Q \), the optimal \( p_i \), i.e. those minimizing the performance function, are located at the centroids of the different cells. Conversely, if the generator positions are fixed, the optimum shape cells are defined by a Voronoï partition. Unsurprisingly, when optimization occurs both over the generator positions and the cell boundaries, the optimum has the generators coinciding with the cell centroids, and is known as a centroidal Voronoï partition. Such partitions have been heavily studied, see e.g. Du, Faber, and Gunzburger (1999), Du, Gunzburger, and Ju (2010) and Liu, Wang, Lévy, Sun, Yan, Lu, and Yang (2009).

Not every centroidal Voronoï partition is however optimum, either globally or locally. Multiple minima and saddle points can arise. As we shall see, the idea of centroidal Voronoï partitions will also be useful in looking at order \( k \) partitions. A key feature of the choice of \( f(q,p_i) = \|q - p_i\|^2 \) for which optimality arises with the centroidal Voronoï partition is the elegant form of a gradient descent algorithm for finding stationary points of the performance function. It is simply

\[
p_i = \alpha M_{c_i} - p_i
\]

where \( \alpha \) is a positive constant, \( M_{c_i} \) is the mass of the \( i \)th Voronoï cell, and \( c_i \) is the centroid of the Voronoï cell whose generator is \( p_i \) (for more details and definitions, see e.g. Du et al. (1999)). Note that \( c_i \) is not a constant; when \( p_i \) changes, the associated cell changes and then its centroid may change.

Partly because it is a comparatively straightforward task to find the centroid of a convex polytope, a recursive algorithm known as Lloyd’s algorithm originally developed in another context (Lloyd, 1982) provides another approach to reaching an optimum partition. In this algorithm, one alternates between finding the centroids of a Voronoï given partition, and finding the Voronoï partition corresponding to a given set of generators, see e.g. Section 5.2 of Du et al. (1999). It is shown in Cortés et al. (2004) that one step of the Lloyd algorithm causes the performance index to either decrease, or remain the same; this paper even uses the term continuous time Lloyd descent to describe a gradient descent algorithm computed for the performance index.

These ideas will all be reflected below in the study of higher order coverage control.

3. Order 2 coverage control

In this section, we discuss order 2 coverage control, so throughout this section we assume that \( k = 2 \). However, we note that the analysis of controller design and the convergence property can also be extended to more general order \( k \) coverage control, and in the next section we will provide more detailed discussions on the generalization. To clarify notations for this section, define the set \( C = \{ i,j \mid i,j \in \{ 1, \ldots , n \}, i < j \} \). We also note that the set designation of a particular \( T \) as \( T_{ij} \) means points \( p_i, p_j \in T_{ij} \) and \( (i,j) \in C \).

3.1. Performance function and its relationship with generalized Voronoï partition

The generalized sensing performance function \( \mathcal{H}(p_1, p_2, \ldots, p_n) \) is constructed as follows

\[
\mathcal{H}(p_1, p_2, \ldots, p_n) = \int_{Q \cap \bigcup_{(i,j) \in C} V_{ij}} \min_{(q,p_i),\,(q,p_j)} f(\|q,p_i\|,\|q,p_j\|)\phi(q) dq
\]

We are trying to find optimal positions of the sensors that can minimize the above performance function. Similarly to an order 1 Voronoï coverage problem, the function \( f(\cdot,\cdot) \) indicates the measurement quality of a point \( q \) but now reflecting a pair of agents. Therefore, for a set of fixed sensor positions, we might measure the quality of sensing associated with those positions by \( \int_{Q} \min_{q,p_i,q,p_j} f(\|q,p_i\|,\|q,p_j\|)\phi(q) dq \). In fact, there is a broad set of \( f(\cdot,\cdot) \) for which it makes sense to formulate such a measure. We shall in fact impose certain properties on \( f(\cdot,\cdot) \) analogous to the order 1 mobile coverage control case, where \( f(\cdot,\cdot) \) being monotonically increasing and differentiable is a basic requirement. In the order 2 case, besides requiring that the function \( f(\cdot,\cdot) \) should be differentiable, it should also have the following properties

\[
\begin{align*}
(1) \quad & \frac{\partial}{\partial q} f(\|q,p_i\|,\|q,p_j\|) \geq 0 \\
(2) \quad & \frac{\partial}{\partial p_j} f(\|q,p_i\|,\|q,p_j\|) \geq 0, \quad \text{and} \\
(3) \quad & f(\|q,p_i\|,\|q,p_j\|) = f(\|q,p_j\|,\|q,p_i\|)
\end{align*}
\]

The first two properties correspond to the monotonically increasing property in the order 1 case. We note that the higher order Voronoï partition is defined as \( (2) \), where the order of the elements in \( T \) should not affect the actual partition. As a result, the order of independent variables should not affect the value of \( f(\cdot,\cdot) \), either. The third requirement in the above condition ensures this property. Certain typical examples of the functions \( f(\cdot,\cdot) \) and their corresponding applications are presented in Jiang et al. (2018).

Based on the above performance function (4) and the distance function, we can obtain the following two lemmas.

Lemma 1. With the definitions of \( Q, C, f(\cdot,\cdot) \), \( T_{ij} \), and \( V_{ij} \) stated above, for all \( q \) in the set \( V_{T_{ij}} \) and \( (k,l) \neq (i,j) \), there holds

\[
f(\|q,p_i\|,\|q,p_j\|) \leq f(\|q,p_k\|,\|q,p_l\|)
\]

The proof of the above lemma can be found in Jiang et al. (2018).

Lemma 2. By using the higher order Voronoï partition and the distance function defined above, the performance function can be further transformed as

\[
\mathcal{H} = \int_{Q \cap \bigcup_{(i,j) \in C} V_{ij}} \min_{(q,p_i),\,(q,p_j)} f(\|q,p_i\|,\|q,p_j\|)\phi(q) dq
\]

This lemma is a straightforward consequence of Lemma 1.
3.2. Controller design

3.2.1. Gradient-based controller and cancellation of boundary terms

In order to minimize the performance function (6), we can design a controller for each sensor with position \( p_i \) as

\[
p_i = -\frac{\partial H}{\partial p_i} \tag{7}
\]

The mobile sensor system with the above controller defines a gradient flow of the performance function (4). According to the property of gradient systems (Absil & Kurdyka, 2006), the above gradient controller provides a natural choice to optimize the performance function. Furthermore, in order to implement the above controller, each agent needs to know the position of its neighbouring agents. The neighbouring agents of \( i \) are \( N_i = \{ j | V_{T_j} \cap \|V_{T_j} \cap V_{T_i} \neq \emptyset \} \), which are agents that monitor the cells \( V_{T_j} \) and the cells with common boundaries with these \( V_{T_i} \) (see Fig. 2).

In the following, we will present an explicit formula for the controller. In the order 2 problem, for each \( p_i \in S \), the following formula holds

\[
\frac{\partial H}{\partial p_i} = \sum_{\forall q_j \in p_i} \int_{V_{T_j}} \frac{\partial}{\partial p_i} f(||q, p_i||, ||q, p_j||) \phi(q) dq \tag{8}
\]

For a detailed analysis explaining the cancellation of boundary terms that yields (7), we refer the readers to Jiang et al. (2018).

In the expression of \( H \) as shown in (6), the domain of integration is a function of \( p_i \). As a result, when one calculates the partial derivative of \( H \) with respect to \( p_i \), one needs to deal with the problem of differentiation under the integral sign. Some basic facts about the problem of differentiating under the integral sign (where integrand and integration limits are functions of a parameter) can be found in Flanders (1973).

One might expect that the right side of the above equation (7) should contain an additional term reflecting the dependence of the region \( V_{T_j} \) on \( p_i \); in effect, (7) is equivalent to the fact that this additional term evaluates as zero. The same phenomenon is well understood in the literature on first order Voronoi optimization problems, and an explanation in terms of elementary calculus concepts can be found in Asami (1991), which points out that it is a particular case of what is sometimes called the envelope theorem: if a function \( h(x, y) \) of two scalar variables is \( C^2 \) and \( H(x) = \min_y h(x, y) \), then a simple calculation shows there holds

\[
\frac{dH}{dx} = \frac{\partial h}{\partial x} \tag{8}
\]

when \( y \) is set to the minimizing value associated with the particular \( x \). This equation states that the tangent to an envelope curve (defined by \( H(\cdot) \)) at a particular point on the curve is also tangent to the particular parametrized curve \( h(\cdot, y) \), \( y \) being the parameter, passing through the same point \( x \), and it extends trivially to the case of vector \( x, y \). It is also known that the \( C^2 \) dependence of \( h \) on \( y \) is not actually necessary, but this is more technical. For our application, one can think of \( x \) as like the sensor positions (generators) and \( y \) as like the boundary points of the cells. Note that the ordinary derivative in (8) is replaced by the partial derivative on the left of (7) since \( x \) is replaced by the set of \( p_i \).

3.3. Higher order centroidal Voronoi partitions

For any given \( f(||q, p_i||, ||q, p_j||) \), it is not easy to find the general relationship of (7) with cell centroids of \( W_i \). However, motivated by the order 1 case, we consider the case when \( f(||q, p_i||, ||q, p_j||) = \frac{1}{2} ( ||q, p_i||^2 + ||q, p_j||^2 ) \). In this case, the performance function becomes

\[
H(p_1, p_2, \ldots, p_n) = \int_{\mathbb{R}^2} \min_{q \in \mathbb{C}_i} \frac{1}{2} ( ||q, p_i||^2 + ||q, p_j||^2 ) \phi(q) dq \tag{9}
\]

Note the centroid and mass of a Voronoi cell \( V_{T_j} \) are \( C_{V_{T_j}} = \frac{\int_{V_{T_j}} \phi(q) dq}{\int_{V_{T_j}} \phi(q) dq} \) and \( M_{V_{T_j}} = \int_{V_{T_j}} \phi(q) dq \) respectively. In this case,

\[
\frac{\partial H}{\partial p_i} = \sum_{\forall q_j \in p_i} \int_{V_{T_j}} \frac{\partial}{\partial p_i} f(||q, p_i||, ||q, p_j||) \phi(q) dq = \sum_{\forall q_j \in p_i} -M_{V_{T_j}} ( C_{V_{T_j}} - p_i ) \tag{10}
\]

Suppose further that the centroid and mass of \( W_i \) are \( C_{W_i} = \frac{\sum_{V_{T_j} \in \mathbb{P}_i} M_{V_{T_j}} C_{V_{T_j}}}{\sum_{V_{T_j} \in \mathbb{P}_i} C_{V_{T_j}}} \) and \( M_{W_i} = \sum_{V_{T_j} \in \mathbb{P}_i} M_{V_{T_j}} \) respectively. Now we have

\[
\frac{\partial H}{\partial p_i} = \sum_{\forall q_j \in p_i} -M_{V_{T_j}} ( C_{V_{T_j}} - p_i ) = -C_{W_i} \sum_{V_{T_j} \in \mathbb{P}_i} M_{V_{T_j}} + p_i \sum_{V_{T_j} \in \mathbb{P}_i} M_{V_{T_j}} \tag{11}
\]

The above result indicates that if \( p_i \) moves towards the centroid of \( W_i \), then \( H \) will decrease over time. Note that this centroid in general moves when the sensors move, since the Voronoi diagram will change with moving sensors. This result is similar to the order 1 centroid coverage control case when \( f(||q, p_i||) \) is defined as \( f(||q, p_i||) = \frac{1}{2} ||q, p_i||^2 \). The notation of \( W_i \) corresponds to the order 2 case but it can be generalized to higher order cases. In the case of an order \( k \) problem, \( W_i \) consists of all the regions currently monitored by agent \( i \). Now we summarize the convergence results concerning the above gradient system as follows.

Lemma 3. For a group of mobile agents with the closed-loop system induced by (7), all the agents’ positions will converge to the set of critical points of \( H \). Furthermore, by taking \( f(||q, p_i||, ||q, p_j||) = \frac{1}{2} ( ||q, p_i||^2 + ||q, p_j||^2 ) \) and designing the controller as \( p_i = \alpha ( C_{W_i} - p_i ) \) where \( \alpha \) is a positive gain, agents’ locations will converge to the cell centroids which result in a higher order centroidal Voronoi configuration.

The proof follows directly from the properties of gradient systems (see e.g. Absil & Kurdyka, 2006) and is omitted here.

Remark 4. For general forms of \( f \), analytical results are in general unavailable. However, one can still carry out the distributed control algorithm by computing a first-order difference approximation to the gradient of \( H \).
3.4. Lloyd algorithm

As for the first order case, a Lloyd algorithm is available for the case where \( f([q, p_1] \quad \| \quad [q, p_2]) = \frac{1}{2}\|q, p_1\|^2 + \|q, p_2\|^2 \), which as we just established leads to centroidal higher order Voronoi partitions. We remark the key differences between the two algorithms: the proposed coverage algorithm in the previous subsections focuses on continuous-time case for modelling agents' motions, while the Lloyd algorithm focuses on discrete-time jumping case to find near optimal locations associated with cell centroids.

With an arbitrary initial set of positions for the sensors, determine the associated higher order Voronoi partition, including the optimal locations associated with cell centroids. The update from \( p_i \) to \( p_i' = C_{Vi} \) is precisely the choice delivered by one cycle of the Lloyd algorithm.

The above Lloyd algorithm only employs the gradient information, while for a faster convergence one may consider use of higher-order optimization methods (e.g. Newton-like algorithms). The application of Newton-like algorithm requires the performance function \( \mathcal{H} \) defined in (4) to be a \( C^2 \) function. We conjecture that by assuming a \( C^2 \) density function \( \phi \), the quadratic distance function \( f \), and the convex region \( Q \), the performance function \( \mathcal{H} \) is \( C^2 \), by following the proof in Liu et al. (2009) which applies in the first order case. This conjecture is supported by the fact that the higher-order partition \( V_T \) shares the same properties as the order 1 partition \( V_C \), as discussed in Section 2.1. Such a conjecture, if true, also guarantees that the Lloyd map is singly differentiable, a fact which assists examination of the stability of fixed points of the map; stability properties of a fixed point can be characterized in terms of the eigenvalues of the derivative of the map at the fixed point. Since this \( C^2 \) property is not the main focus of this paper, we will not dwell on this point but will consider it in more detail in future research.

3.5. Convergence issues

By and large, convergence issues for the gradient descent algorithm and the Lloyd algorithm in the kth order case are similar to those in the order 1 case. Of course, it would be desirable if such algorithms converged to a global minimum. This however is not always guaranteed.

We note that for order 1 problems, there exist centroid Voronoi tessellations for which the Hessian is indefinite or singular. A tedious calculation demonstrates this fact for a rectangular region \( Q \), with two cells obtained by joining midpoints on the shorter sides of the rectangle. Suppose the shorter side is of length 1. When the longer side has length exceeding \( \sqrt{\frac{2}{3}} \), the Hessian associated with the centroidal tessellation will have one negative eigenvalue, and when the longer side has length equal to \( \sqrt{\frac{2}{3}} \), the Hessian will have one zero eigenvalue.

This is most easily seen by working with the Lloyd map \( T \), mapping generators \( p_1, p_2, \ldots, \) of a collection of Voronoi sets to the centroids \( c_1, c_2, \ldots \) of those sets. The Lloyd map is known to be \( C^1 \), for a convex region \( Q \). At a centroidal tessellation, there holds \( \nabla^2 \mathcal{H} = 0, T(p) = p \) and there further holds (Du et al., 1999)

\[
\nabla^2 \mathcal{H} = 2M[I - VT]
\]

where \( M \) is a positive definite diagonal matrix involving the masses \( M_i \) of the Voronoi regions. The eigenvalues of the Jacobian \( VT \) must be less than 1 in magnitude for the centroidal tessellation to be a stable equilibrium of the Lloyd algorithm, and these eigenvalues are relatively easy to compute for any centroidal tessellation, including the 2 cell tessellation just mentioned, and lead to the conclusion recorded above.

Of course, since the cost function in a steepest descent algorithm can never increase and is bounded below, in the limit as \( t \to \infty \), this cost function necessarily goes to a limit. Further, \( \mathcal{H} \) can be checked for an order one problem to be

\[
\dot{\mathcal{H}} = -\alpha \sum_i M_{Vi} \|T(p_i) - p_i\|^2
\]

where \( \alpha \) is a constant of the Voronoi region \( Vi \) and this equation is a consequence of the formula (3). Because \( T(p) \) is \( C^1 \) in \( p \) and \( \dot{p} \) is continuous in time and bounded, \( \mathcal{H} \) itself is differentiable with respect to time and then by Barbabali's lemma, see e.g. Sastry (1999), \( \mathcal{H} \) itself tends to zero. This says that there is convergence to a centroidal Voronoi tessellation. If the associated Hessian happens to have a negative eigenvalue, the equilibrium will be a saddle point, and thus effectively unstable. From an arbitrary initial condition, one would not expect such an equilibrium to be reached. But from a thin set, it will normally be reachable. This argument will extend to the higher order case provided \( T(p) \) is a \( C^1 \) function of \( p \), using (11) and the law presented in Lemma 3. Finally, we
remark that for certain sets \( Q \) which are not convex polygons, there may be a continuum of order 1 centroidal Voronoi tessellations. Suppose for example that \( Q \) is a circle of radius 1, and that there are four generators. If these generators are equally spaced around a circle with the same centre as \( Q \) but with radius \( \pi/4 \), they will be generators of a centroidal Voronoi tessellation. Obviously, there is a continuum of possibilities.

4. Higher order coverage for discrete sets

Indeed of a convex area \( Q \) one can consider the coverage problem for a finite set of discrete points \( Q = \{q_1, \ldots, q_n\} \) in \( \mathbb{R}^n \). This problem arises in applications where only a finite number of points of interest are to be observed by sensors which can move freely in the plane. First order coverage is related to the clustering problem, where a set of \( N \) observations is partitioned into \( m \) clusters according to some notion of similarity, see e.g. Aggarwal (2015). The Lloyd algorithm for first order coverage of discrete point sets is also known as the k-means algorithm and widely applied for clustering problems (Reddy & Vinzamuri, 2005). Higher order coverage can be viewed as a clustering approach where points are assigned to the multiple clusters simultaneously. Assigning points to multiple clusters appears in fuzzy c-means clustering (Bezdek, 1981; Dunn, 1973). However, in these approaches each data point is assigned to all clusters according to a fuzzy membership function.

In the following we let \( k \) denote the order that we consider; note \( k \leq N \) is a finite number. It is straightforward to extend the generalized Voronoi partitions \( V_T \) to the discrete points case. For technical reasons we require that for different \( T \) and \( T' \) the sets \( V_T \), \( V_{T'} \) have empty intersection. This can be ensured by assigning a \( q_j \) only to the \( V_T \) where \( T \) is the smallest w.r.t. a suitable order of the \( T \), e.g. a lexicographical order. For a discrete data set the measure \( \phi(q) \) with \( q \) in \( \mathbb{R}^n \) is the performance function replaced by a discrete measure supported on the finite set \( \{q_1, \ldots, q_n\} \). This turns the integral into a finite sum which gives the performance function

\[
\mathcal{H}(p_1, \ldots, p_m) = \sum_{l=1}^{N} w_l \min_{(q_1, \ldots, q_n) \in C} f(||q_1, p_1||, \ldots, ||q_n, p_n||) \tag{18}
\]

with \( w_1, \ldots, w_N \) suitable weights (which may be all the same). Our goal is again to find minima of the performance function. Any sensor placement yields an assignment of the \( q_i \) to the \( W_i \), where analogously to the convex set case a \( W_i \) is the union of the cells (which are finite sets) assigned to sensor \( p_i \).

The Lloyd/k-means algorithm alternates between updating the estimates of centroids of the clusters and computing new clusters in each iteration until convergence is reached. We extend this algorithm to our coverage problem, and we refer to this extension as the higher-order m-means algorithm (The use of the symbol \( k \) in the literature is common, and long-standing, see e.g. Aggarwal (2015). In terms of our notation, the term m-means algorithm is more appropriate.). As an extension to the conditions on \( f \) stated in Section 3, we assume that for all \( p_1, \ldots, p_n \in \mathbb{R}^n \) the function \( q \mapsto f(||q, p_1||, \ldots, ||q, p_n||) \) is monotonically increasing, invariant w.r.t. the permutation of its arguments, and strictly convex. We define for \( p_1, \ldots, p_m \in \mathbb{R}^n \) and arbitrary non-empty subsets \( W_1, \ldots, W_m \subset Q \) an extended performance function

\[
\mathcal{H}(p_1, \ldots, p_m, W_1, \ldots, W_m) = \sum_{l=1}^{N} \sum_{q \in W_l} w_q f(||q, p_1||, \ldots, ||q, p_m||)
\]

with \( w_q \) the weight corresponding to \( q \in Q \) in (18). In the first order covering case the Lloyd / k-means algorithm can be viewed as alternating between optimization steps on the \( p_i \) and the \( W_i \) variables (Selim & Ismail, 1984). Thus for \( p_1, \ldots, p_m \in \mathbb{R}^n \), \( W_1, \ldots, W_m \subset Q \) we define \((C_1, \ldots, C_m) \in \mathbb{R}^{mn} \) by

\[
(C_1, \ldots, C_m) = \arg\min_{p_1, \ldots, p_m} \sum_{l=1}^{m} \sum_{q \in W_l} w_q f(||q, p_1||, \ldots, ||q, p_m||)
\]

Due to the convexity assumption on \( f \) the \( C_i \) are well-defined. For the special case that \( f(||q, p_1||, \ldots, ||q, p_m||) = g(||q, p_1||) + \cdots + g(||q, p_m||) \) we have

\[
C_i = \arg\min_{p_i} \sum_{q \in W_l} w_q g(||q, p_i||).
\]

In particular, for \( g(||q, p_i||) = ||q, p_i||^2 \) and all \( w_q = 1 \) the \( C_i \) is the mean of the points in \( W_i \).

Algorithm — second order m-means

1. Start with random values for the \( p_i \). Compute the corresponding \( W_i \) where

\[
W_i = \{q_j \mid j = 1, \ldots, N; \text{ there are at most } k-1 \text{ numbers of } p_i \text{ with } ||p_i - q_j|| < ||p_i - q_l||\}\]

2. For \( i = 1, \ldots, m \) do:
   1. Compute the \( C_i \).
   2. Set \( p_i = C_i \).

3. Compute the new \( W_i \) as in step 1. If one \( W_i \) is empty restart the algorithm with step 1.

4. If \( \mathcal{H} \) is not decreased by the previous steps 2 or 3, stop the algorithm. Otherwise go to step 2.

By construction each iteration in Step 2 does not increase \( \mathcal{H} \). Further, Step 3 also does not increase \( \mathcal{H} \). Note that at this point it is necessary that each \( q_i \) is assigned only to \( k \) sets. Due to boundedness of \( \mathcal{H} \) from below the sequence of values of \( \mathcal{H} \) converges. If \( f(||q, p_1||, \ldots, ||q, p_m||) = g(||q, p_1||) + \cdots + g(||q, p_m||) \) each \( C_i \) depends only on \( W_i \). Further, there is only a finite number of possible values for each \( W_i \) and the algorithm terminates when the cost is not decreasing. Thus for such performance functions the algorithm stops after a finite number of steps.

Proposition 6. Denote by \( p_i^l, W_i^l \) the iterates of the kth order m-means algorithm. Then the sequence \( \mathcal{H}(p_1^l, \ldots, p_m^l, W_1^l, \ldots, W_m^l) \) converges. For \( f(||q, p_1||, \ldots, ||q, p_m||) = g(||q, p_1||) + \cdots + g(||q, p_m||) \) the m-means algorithm terminates after a finite number of steps.

Note that the argument for convergence after a finite number of steps is the same as for the standard k-means algorithm (Selim & Ismail, 1984).

For a simulation of the convergence performance of the above coverage algorithm for discrete sets, the readers are referred to Jiang et al. (2018).

5. Conclusions

In this paper, we considered a class of generalized Voronoi coverage control problems by introducing the higher order Voronoi partition concept in the coverage performance functions. This coverage problem is motivated by many real life applications which require more than one sensor to cooperate in monitoring one single cell. We focused on the order 2 Voronoi based coverage problem, and provided detailed analysis on the performance function, the
controller design and controller performance (including convergence properties). Their performance and effectiveness have been supported by several simulations presented in the full version of the paper (Jiang et al., 2018). The results were extended to study higher degree Lloyd maps, their connection with gradient descent algorithms, and convergence properties. In addition, we have considered an application of higher order coverage to clustering of discrete sets, and established the relation to what is often termed k-means algorithms.

References


Zhiyong Sun received the Ph.D. degree from The Australian National University (ANU), Canberra ACT, Australia, in 2017. He was a Research Fellow/Lecturer with the Research School of Electrical Engineering at the ANU from 2017 to 2018. In June 2018, he joined Department of Automatic Control, Lund University, Lund, Sweden, as a postdoctoral researcher. His research interests include graph rigidity theory, control of autonomous formations and networks.

Bomin Jiang received the Ph.D. degree from The Australian National University (ANU), Canberra ACT, Australia, in 2017. He was a Research Fellow/Lecturer with the Research School of Electrical Engineering at the ANU from 2017 to 2018. In June 2018, he joined Department of Automatic Control, Lund University, Lund, Sweden, as a postdoctoral researcher. His research interests include distributed control, optimization theory and econometrics.

Dr. Sun was a recipient of the Australian Prime Minister’s Endeavour Postgraduate Award in 2013 from the Australian Government, the Outstanding Overseas Student Award from the Chinese Government in 2016, and the Springer PhD Thesis Prize from Springer in 2017. He received the Best Student Paper Award at the 4th Australian Control Conference (AUCC 2014), a Best Student Paper Finalist Award from the 54th IEEE Conference on Decision and Control (CDC 2015), and was the winner of the Best Student Paper Award from the 5th Australian Control Conference (AUCC 2015).

Brian D.O. Anderson was born in Sydney, Australia, and educated at Sydney University in mathematics and electrical engineering, with Ph.D. in electrical engineering from Stanford University in 1966. Following graduation, he joined the faculty at Stanford University and worked in Vidar Corporation of Mountain View, California, as a staff consultant. He then returned to Australia to become a senior consultant on electrical engineering at the University of Newcastle. From there, he moved to the Australian National University in 1982, as the first engineering professor at that university. He is now Emeritus Professor at the Australian National University, Distinguished Professor in Hangzhou Dianzi University and Distinguished Researcher in Dalian-T-CSIRO, formerly NICTA. During his period in academia, he spent significant time working for the Australian Government, with this service including membership of the Prime Minister’s Science Council under the chairmanship of three prime ministers. He also served on advisory boards or boards of various companies, including the board of the world’s major supplier of cochlear implants, Cochlear Corporation, for 40 years. His awards include the IFAC Quaz Medal in 1999, IEEE Control Systems Award of 1997, the 2001 IEEE James H. Mulligan, Jr. Education Medal, and the Bode Prize of the IEEE Control System Society in 1992, as well as IEEE and other best paper prizes, including one from Automatica. He is a Fellow of the Australian Academy of Science, the Australian Academy of Technological Sciences and Engineering, the Royal Society (London), and a foreign member of the US National Academy of Engineering. He holds honorary doctorates from a number of universities, including Université Catholique de Louvain, Belgium, and ETH, Zurich. He served as IFAC President from 1990 to 1993, having had earlier periods in various IFAC roles, including editor of Automatica. He was also President of the Australian Academy of Science from 1998 to 2002. His current research interests are in distributed control, sensor networks and econometric modelling.

Christian Lageman received the Dipl.-Ing. degree in Computer Science and the Dipl.-Math. and Dr. rer. nat. degrees in Mathematics from the University of Würzburg, Germany, in 2001, 2002 and 2007, respectively. From 2007 to 2008 he worked as a Research Fellow at the Centre for Mathematics and its Applications, The Australian National University. From 2008 to August 2009 he was on a postdoctoral fellowship at the Department of Electrical Engineering and Computer Science, University of Liège, Belgium. From 2009 to 2017 he was a postdoctoral researcher at the Institute of Mathematics, University of Würzburg, Germany. Since 2017 he is working as a consultant in the IT industry. His research interests are in nonlinear control theory, cooperative control, optimization and estimation on manifolds.