Multi-robot Coverage and Exploration on Riemannian Manifolds with Boundary

Subhrajit Bhattacharya Robert Ghrist Vijay Kumar

Abstract

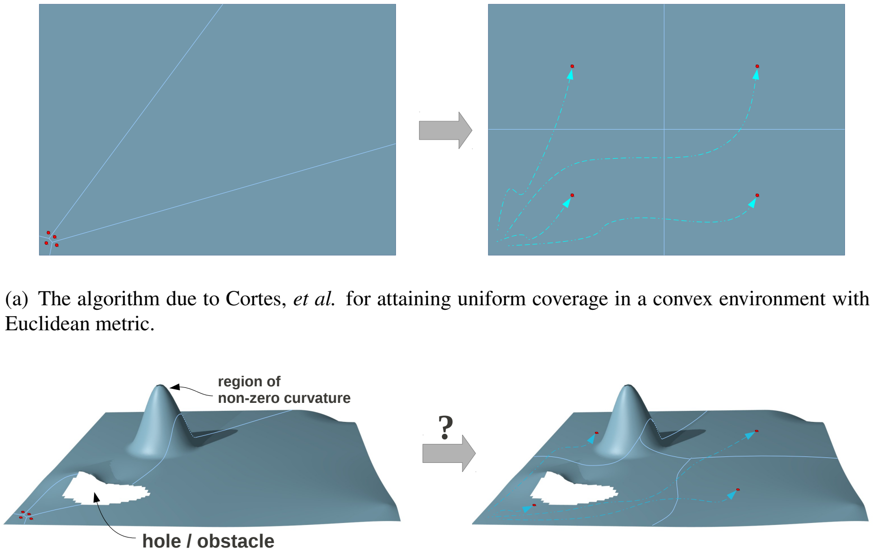
Multi-robot coverage and exploration are fundamental problems in robotics. A widely-used, efficient and distributable algorithm for achieving coverage of a convex environment with Euclidean metric is that proposed by Cortes, *et al.*, which is based on the discrete-time Lloyd's algorithm. This algorithm is not directly applicable to general Riemannian manifolds with boundary that are non-convex and are intrinsically non-Euclidean. In this paper we generalize the control law based on minimization of the *coverage functional* to such non-Euclidean spaces punctured by obstacles. We also propose a practical discrete implementation based on standard graph search-based algorithms. We demonstrate the applicability of the proposed algorithm by solving efficient coverage problems on a sphere and a torus with obstacles, and exploration problems in non-convex indoor environments. [†]

1 Introduction

The geometry underlying configuration spaces of multiple robots is a critical feature implicit in several important challenges in planning and coordination. Metric considerations are fundamental to problems of coverage [Lloyd 82, Cortez 05, Cortes 04, Bullo 09], exploration [Thrun 05, Stachniss 05, Stachniss 06], and more. A well-known approach to solving coverage problems with n robots involve partitioning the appropriate configuration space into n-*tessellation* (a partition of the configuration space into simply-connected domains) [Lloyd 82, Cortez 05]. In particular, this method requires a Voronoi tessellation on the configuration space geometry. While such a tessellation is easy to achieve in a convex environment with Euclidean metric, it becomes increasingly difficult in environments with obstacles and non-Euclidean metrics. The presence of obstacles makes it a geodesically non-convex manifold with boundary. Non-Euclidean metrics can arise in the geometry of a configuration space as inherited from the structure of the underlying domain (*e.g.*, from irregular terrain), or via direct manipulation of the configuration space geometry for problem goals (*e.g.*, in multi-robot cooperative exploration problems [Bhattacharya 10]).

The problem of attaining balanced coverage of an environment is fundamental to many practical multi-robot problems. One common coverage control approach — efficient and distributable — is through the definition of feedback control laws defined with respect to the centroids of Voronoi cells resulting from the Voronoi tessellation of the domain. Lloyd's algorithm [Lloyd 82] is a discrete-time algorithm that minimizes a *coverage functional*. A continuous-time version of the algorithm is described in [Cortes 04], where the authors propose gradient descent-based individual robot control laws that guarantee optimal coverage of a convex environment given a density function which

[†] Parts of this work have been presented at the 10^{th} International Workshop on the Algorithmic Foundations of Robotics (WAFR), 2012.



(a) The algorithm due to Cortes, *et al.* for attaining uniform coverage in a convex environment with Euclidean metric.

(b) We would like to solve the problem for a geodesically non-convex Riemannian manifold with boundary.

Figure 1: An overview of the main contribution of the paper.

represents the desired coverage distribution. To remove the limiting assumption of a convex environment, the authors of [Pimenta 08] propose the use of *geodesic Voronoi tessellation* determined by the geodesic distance rather than the Euclidean distance. However such a method both involves computationally-difficult geometric computations and is still limited to Euclidean environments with polygonal obstacles. Recent work [Bhattacharya 10] has used a graph search-based approach to develop tools for solving the coverage problem in non-convex environments with a non-Euclidean metric. However, in order to explicitly compute an analog of a *generalized centroid* in non-convex tessella, an approximate method involving *centroid projection* was used. Such a method is, admittedly ad hoc, gives weak guarantees, and is difficult to implement when the configuration space is not topologically simple (equivalent to a punctured simply-connected domain). There exist search-based discrete-time algorithms that explicitly search every vertex in a tessellation to find the best position for the robot in every time-step (as in [Durham 12]). Although such a controller can solve the problem of multi-robot decentralized coverage on arbitrary metric graphs, the high computational complexity of this approach makes it impractical for fine discretization or large graphs.

Contributions and Organization:

In this paper we generalize the method for computing the control law that is described in [Pimenta 08, Lloyd 82] and adapt it to non-Euclidean metric spaces with obstacles that are not necessarily polygonal (Figure 1). In [Bhattacharya 12] we proposed the mathematical formulae and the basic algorithm for attaining coverage in such spaces. The present work focuses on making the mathematical framework more rigorous by formalizing details like differentiability and convergence of the control laws.

The principal theoretical tools are developed in Section 2.6 along with an explicit formula ready for computation. They relate geodesics, distance derivatives, and the metric tensor. However, the computability relies on the existence of minimal paths and differentiability of the distance function. These are formally discussed in details in Section 2.3. This inspires the design of the control law for coverage by multiple robots on a manifold with boundary. This control law does not suffer from the issues of robots getting *stuck* at the boundaries of obstacles because of the fact that we use the *length metric* as our distance function. We realize the control law using a graph search-based method to achieve an efficient discrete implementation. We illustrate our methodology by showing how to solve Coverage problems on non-Euclidean Riemannian manifolds with boundary, multi-robot cooperative exploration, and cooperative human-robot exploration.

In Section 2 of the paper we primarily discuss some mathematical tools related to Riemannian manifolds with boundary. The main contribution of the paper appears in Section 3 where we develop and prove stability of the proposed control law for coverage on Riemannian manifolds with boundary. We present the graph search-based discrete implementation and simulation results in Section 4. For better readability, we have placed the proofs of the lemmas, the corollary and the proposition in the paper in the appendix at the end of the paper.

2 Background – Manifolds with Boundary

In this section we discuss and build some of the theoretical tools that will be essential in designing and proving the stability of the control law (which we will do in Section 3) for coverage in general Riemannian manifolds with boundary. Unless otherwise mentioned, we will assume all manifolds mentioned in this paper to be path connected.

2.1 Preliminaries: Riemannian Geometry

We assume that the reader is familiar with the mathematical concepts of *manifolds*, *tangent bundles*, *cotangent bundles*, *coordinate atlas*, *Riemannian metric* and *geodesics*. For detailed discussion on these topics please refer to Appendix A of [Murray 94] or any standard text book on Manifolds and Riemannian geometry [Jost 97, Petersen 06]. As a quick reference, we provide below some of the most frequently used concepts from Riemannian geometry.

Definition D1 (Metric or Distance function)**.** A *metric* on a topological space [Munkres 99], X, is a function $d : X \times X \to \mathbb{R}$ such that, for any $p, q \in X$, the following conditions are satisfied,

i. $d(p, q) \geq 0$,

ii. $d(p, q) = 0$ if and only if $p = q$,

iii. $d(p, q) = d(q, p)$, and,

iv. $d(p, q) \leq d(p, r) + d(r, q)$ for all $p, q, r, \in X$ (this is called the *triangle inequality*).

Riemannian geometry is concerned with a specific class of topological spaces, namely *manifolds*.

Definition D2 (Manifold)**.** A manifold is a topological space that locally looks like an Euclidean space everywhere. That is, if M is a topological space, and $p \in M$ is a point in it, then there exists a open neighborhood U of p (*i.e.* an open set U with $p \in U$), such that one can construct homeomorphisms $\psi : U \to \mathbb{R}^D$ for some nonnegative integer D. The minimum value of D for which it is possible to construct such homeomorphisms is called dimension of the manifold.

Definition D3 (Coordinate Chart)**.** Given an open subset [Munkres 99] U of a D-dimensional manifold M, and a continuous injective function $\phi : U \to \mathbb{R}^D$, we say $C = (U, \phi)$ is a coordinate chart on U. ϕ in fact needs to be a homeomorphism as well (*i.e.* have a continuous inverse over its image).

A couple of related concepts are those of *atlas* and *chart transition*, for which we will refer the reader to the standard texts.

Definition D4 (Tangent Space)**.** Given a coordinate chart (U, ϕ) on a smooth manifold M, the tangent space at a point p on it represented by the coordinate variable $\mathbf{x} \in \Omega = \mathit{Img}(\phi) \subset \mathbb{R}^D$ is a D-dimensional (abstract) vector space, $T_pM = T_{\mathbf{x}}\Omega$, spanned by the basis $\frac{\partial}{\partial x^1}, \frac{\partial}{\partial x^2}, \cdots, \frac{\partial}{\partial x^D}$.

The dual of the tangent space (*i.e.*, the space of all linear maps $T_p \to \mathbb{R}$) at p is known as the *cotangent space*, and is represented by T_p^*.

Definition D5 (Riemannian Metric Tensor)**.** A Riemannian metric tensor (or simply, a *Riemannian metric*) on a differentiable manifold, M, is a nondegenerate, symmetric bilinear scalar product on each tangent space, T_pM, for every $p \in M$, such that it varies smoothly with p. That is, it is a bilinear function (for a given $p \in M$) $\eta : T_pM \times T_pM \to \mathbb{R}$, that is symmetric in its two parameters, is zero only if one of its parameters is the zero vector, and varies smoothly with p.

As a consequence, in a particular coordinate chart, η has a matrix representation. The components of this matrix are written as η_{ij}. We will use the notation $\eta_{\bullet\bullet}$ to denote the matrix itself (in the specific coordinate chart).

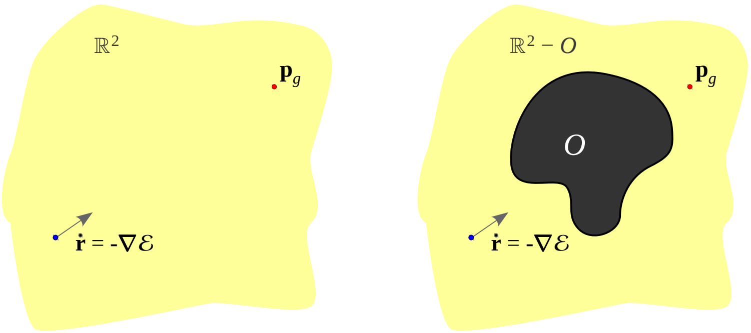
A manifold equipped with a Riemannian metric at every point is called a Riemannian manifold. The metric tensor, η, gives a notion of distance-minimizing curves locally at each point, that satisfy the *geodesic equation* (see [Petersen 06]). Integral curves of the geodesic equation are known as *geodesics*. If all geodesics can be extended indefinitely on the Riemannian manifold (*i.e.*, all integral curves of the geodesic equation exists in the manifold), it is called a *geodesically complete* Riemannian manifold.

2.2 Motivation: Stability and Convergence of Control Laws

If the configuration space of a robot (or a system of robots) can be described by a *finite dimensional* Banach space (*i.e.* a vector space with a norm, $\|\cdot\|$), one can easily specify convergence and stability of a proposed control law. In particular, Lyapunov stability, asymptotic stability and exponential stability [Sastry 99] are some of the conditions that are often desired. These stability analyses often boil down to finding an energy-like function (often called the *Lyapunov function*), and showing that the proposed control law can be expressed as the gradient of the energy-like function. Conditions on the function and its gradients on the entire space translate into stability conditions.

However, very often the configuration space under consideration is a general Riemannian manifold. If it is a complete manifold (*i.e.* every geodesic curve on it can be extended indefinitely in both directions), one approach is to consider the tangent space at a point [Petersen 06], which is a Hilbert space (and hence a Banach space), and study the local stability of the control laws in the tangent space via the exponential map. Since the exponential map is often not bijective, especially for compact manifolds, the study of convergence in terms of Lyapunov function makes sense only locally at each point.

However the scenario is most difficult when the manifold is not complete. These are typically manifolds obtained by removing closed sets from other complete manifolds. In such cases the exponential map is not even defined completely on the entire tangent space at a point. Geodesic emanating from a point can hit a hole/puncture on the manifold. We often include the boundary near the holes/punctures to make it a *manifold with boundary*. That makes it complete as a metric space, but it no longer remains a manifold since the points at the boundary do not locally resemble Euclidean space. Thus we lose the notion of a tangent bundle on this space with the tangent space being fibers that are identical at every point.


$\mathbb{R}^2$
$\mathbf{p}_g$
$\dot{\mathbf{r}} = -\nabla\mathcal{E}$
$\mathbb{R}^2 - O$
$\mathbf{p}_g$
O
$\dot{\mathbf{r}} = -\nabla\mathcal{E}$

(a) Point robot at $\mathbf{r}$ navigating towards $\mathbf{p}_g$ in the Banach space, $\mathbb{R}^2$, by descending gradient of $\mathcal{E}$, will reach its goal.

(b) $\mathbb{R}^2 - O$ is not a Banach space. On this space the integral curve of $\dot{\mathbf{r}} = -\nabla\mathcal{E}$ does not exist.

Figure 2: In presence of obstacles, a robot can get stuck at '*local minima*' at the boundary of the obstacles. This is because of the fact that the control vector at a point on the boundary may not exist inside the *tangent cone* at the point, and hence tend to 'push' the robot into obstacles.

Unfortunately it is the last kind of spaces that one encounters most often in problems of robot navigation [Rimon 91, LaValle 06]. The holes or punctures arise due to presence of obstacles. Supposing one can embed the manifold with boundary in an Banach space, stability of control laws on the embedding space does not correspond to stability on the manifold with punctures.

The simplest example is that of a point robot navigating on $\mathbb{R}^2$, and the objective being reaching a goal point, $\mathbf{p}_g \in \mathbb{R}^2$. One can simply construct a smooth energy function, $\mathcal{E} : \mathbb{R}^2 \to \mathbb{R}$, with an unique global minima at $\mathbf{p}_g$, and non-zero gradient everywhere else (*e.g.* $\mathcal{E}(\mathbf{r}) = \|(\mathbf{r} - \mathbf{p}_g)\|^2$). The robot following the negative of the gradient of this function would reach the goal (*i.e.* the integral curve of $\dot{\mathbf{r}} = -2(\mathbf{r} - \mathbf{p}_g)$, exists in $\mathbb{R}^2$, and converges to $\mathbf{p}_g$ as $t \to \infty$, the convergence being exponential in this particular example – Figure 2(a)). However, if we remove an open subset O from $\mathbb{R}^2$ and make that the configuration space of the robot, the same control law will end up making the robot hit the boundary of O and 'get stuck' there, since the control action will tend to make the robot move into O (Figure 2(b)). These are the *local minima* at the boundaries where the gradient of the energy function does not vanish (also see Figure 10). The more fundamental reason that these kinds of minima occur is that the control vector may not always belong to the *tangent cone* (an analog of the tangent space: see Section 2.4) at the point on the boundary.

To solve this problem, one may attempt to generate a smooth energy function in $\mathbb{R}^2$ such that, starting at any point in $\mathbb{R}^2 - O$, the integral curves of the gradient of the function will remain completely inside $\mathbb{R}^2 - O$. This was the approach adopted in [Rimon 92], which involved the construction of specially crafted *navigation functions*. However, the construction of such energy functions is, in general, difficult, computationally expensive and works only for special classes of obstacles, O. The other, more direct approach in $\mathbb{R}^2$ involves the construction of visibility graphs [Lozano-Pérez 79]. In this approach the robot essentially follows the shortest path that lies entirely in the free configuration space. This, in effect, generates a vector field with the property of integral curves lying entirely in $\mathbb{R}^2 - O$ as desired.

While the direct construction of this vector field is simple in case of the problem of goal-directed navigation for a single robot, very often one is faced with an objective that is more nat-

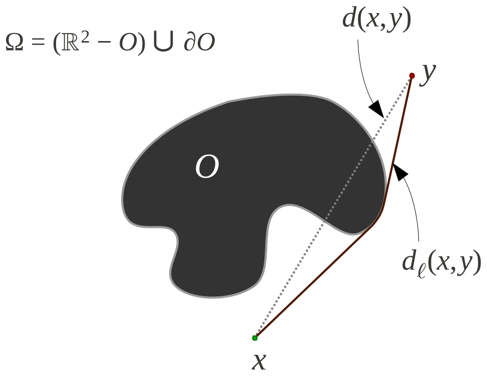


Figure 3: In this figure O represents a compact obstacle with smooth boundary in $\mathbb{R}^2$. $\Omega = (\mathbb{R}^2 - O) \cup \partial O$ is a manifold with smooth boundary that is complete as a metric space (note that we included the boundary of O to make the space compact). The metric induced by $\mathbb{R}^2$ is not a path metric since a path of length $d(x, y)$ does not exist in Ω. But d_ℓ is a path metric.

urally described in terms of minimization of an energy function. The problem of robot coverage [Lloyd 82, Cortez 05, Cortes 04, Bullo 09] is one such problem we will consider extensively later in this paper. In such problems it is natural to compute the control commands from the gradient of the energy-like functions. So the more general question that arises is:

> What class of energy-like functions defined on a manifold, Ω, possibly with boundary, will guarantee that the integral curves of their gradients, starting from a point in Ω, will completely lie inside Ω? What can we say about the convergence/stability of the system?

While in this paper we will mostly focus on the problem of multi-robot coverage, we hope that the analysis that we will present in the following sections will help in establishing a much broader control paradigm on manifolds with boundary.

2.3 Length Metric

We consider a smooth manifold, Ω, possibly with boundary. Such spaces are well-studied in mathematics [Wolter 85, Alexander 81]. In robotics they are of great interest since they represent configuration spaces of many robots and linkages with no “immaterial edge” assumptions. For most of the analysis in this paper we will assume that Ω is embedded in some Euclidean space equipped with its standard Euclidean metric $(\mathbb{R}^D, d)$. The topology of Ω is assumed to be the *subspace topology* [Munkres 99] derived from $\mathbb{R}^D$ (so that the boundary of Ω is part of the same topological space). The metric induced by the ambient Euclidean space is, in general, not a *path metric* unless Ω is convex (see p.10 of [Gromov 99]). That means, if $i : \Omega \hookrightarrow \mathbb{R}^D$ is the embedding, for any points $p, q \in \Omega$, there may not exist a path $[0, 1] \to \Omega$ of length $d(i(p), i(q))$.

However, if Ω is path connected, one can define a different metric structure on Ω that makes it a path metric space, (Ω, d_ℓ). The new metric, d_ℓ, called the *length metric*, can be defined such that $d_\ell(p, q)$ is the infimum of the length of *rectifiable* curves or paths (computed using the *length structure* [Gromov 99] induced by the embedding) joining p and q, but lying entirely in Ω. It is not difficult to see that (Ω, d_ℓ) is a complete metric space (though not geodesically complete). Thus, due to the Hopf-Rinow theorem [Gromov 99] there exists a minimizing path (not necessarily unique) joining every pair of points. Although this last observation may appear somewhat trivial, it is easy to construct cases of non-compact manifolds where such minimizing geodesics may not exist (*e.g.*, in Figure 3, consider $\mathbb{R}^D - O$, without the boundary of O – a path of length equal to the infimum does not exist in that space for the shown points x, y). Thus we cannot simply remove the boundaries from the space of interest.

2.4 Generalization of Tangent Space

By retaining the boundary, Ω no more remains a manifold in the strict sense. Neighborhoods of points on the boundary of Ω (*i.e.*, points on $\partial\Omega$) resemble an Euclidean half space. However, $\Omega - \partial\Omega$ is indeed a manifold, although not complete. Thus, given $p \in (\Omega - \partial\Omega)$, one can define the tangent space $T_p(\Omega - \partial\Omega)$ in its usual sense. However, we do not have the usual notion of a tangent space at points on $\partial\Omega$.

Let us now consider a point, $p \in \Omega$. The tangent space of the Euclidean space in which Ω is embedded is thus $T_{i(p)}\mathbb{R}^D$. Now we consider the subset of this tangent space that consists of the vectors (and their non-negative scalings) that are tangents to curves emanating from p and lying entirely in Ω. That is, we define the cone,

$$T_p\Omega = \bigcup_{v \in \Lambda_p\Omega} \{\alpha v \mid \alpha \geq 0\}$$

where,

$$\Lambda_p\Omega = \left\{ \left(\frac{d\gamma^i}{dt}\Big|_{t=0} \right) \frac{\partial}{\partial x^i} \;\middle|\; \begin{array}{l} \gamma : [0, a] \rightarrow \mathbb{R}^D, a > 0 \text{ are } C^1 \text{ paths parametrized by their lengths,} \\ \text{with } \gamma(0) = p,\ \mathit{Img}(\gamma) \subseteq \Omega \end{array} \right\}$$

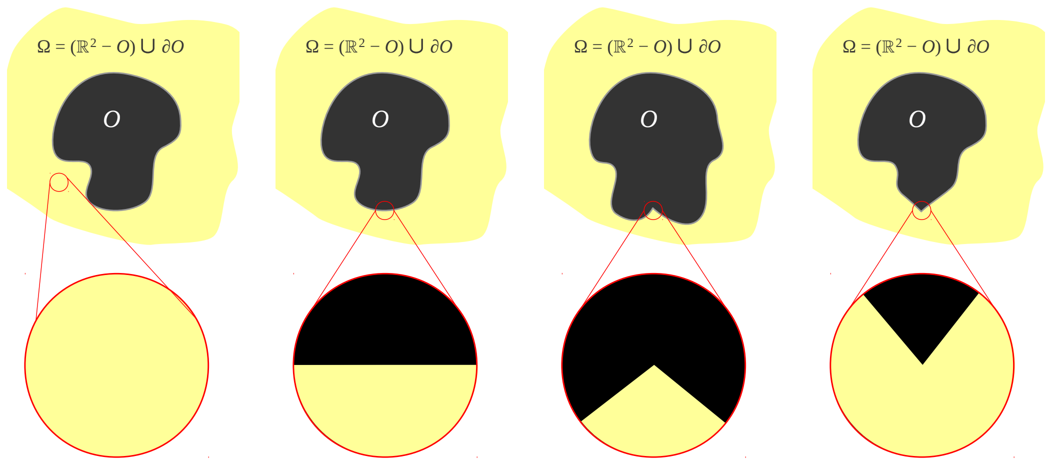
Here trajectory, γ, has been represented using a coordinate system of choice on $\mathbb{R}^D$ (thus γ^i is the i^{th} component of the point's coordinate). We call $T_p\Omega$ the *tangent cone* at p in Ω.

Lemma L1. *If Ω is a smooth manifold with smooth boundary embedded in $\mathbb{R}^D$, $T_p\Omega$ is a half space in $T_{i(p)}\mathbb{R}^D$ for $p \in \partial\Omega$, and a full space for $p \in (\Omega - \partial\Omega)$ (see Figure 4).*

Remark R1. The above lemma can be generalized to assert that $T_p\Omega$ will be a convex cone for points on a wider class of manifolds with boundary with corners (*i.e.* boundary that is not necessarily smooth). As long as the space is locally convex (*i.e.* for every point $p \in \Omega$, there exists a open neighborhood, U, in Ω, such that the metric induced by d makes U a path metric space), $T_p\Omega$ will be a convex cone (Figure 4(c)).

Remark R2. We will allow $\partial\Omega$ to be not smooth (C^1) at finite number of distinct points, where they may not be locally convex either. In our coverage problem such a point can at the worst be an unstable 'local minima', in which case we will rely on presence of small noise that would 'push' the robot out that isolated point (see Section 3). Alternatively we can assume smoothness of the boundary to be a generic condition, wherein a boundary with finite and distinct set of non-smooth points can be *smoothed* locally (using a *mollifier* [Hormander 90]) to obtain a C^1 boundary. In which case the result of Lemma L1 will hold for every point.

Remark R3. We define the *cotangent cone* at p as the dual of the cone $T_p\Omega$, and represent it as $T_p^*\Omega$. If $T_p\Omega$ is convex, so is $T_p^*\Omega$, and vise-versa. Consider a real-valued smooth function, $f : \Omega \rightarrow \mathbb{R}$.


$\Omega = (\mathbb{R}^2 - O) \cup \partial O$
O
$\Omega = (\mathbb{R}^2 - O) \cup \partial O$
O
$\Omega = (\mathbb{R}^2 - O) \cup \partial O$
O
$\Omega = (\mathbb{R}^2 - O) \cup \partial O$
O

(a) In the interior of Ω, neighborhoods are Euclidean space, which is convex.

(b) On smooth boundary of Ω, neighborhoods are Euclidean half space, which is convex.

(c) Although this is a point on the boundary where it is not C^1, $T_p\Omega$ is still convex.

(d) $T_p\Omega$ is not convex at this point.

Figure 4: Points on Ω and the shape of $T_p\Omega$ (the lighter region inside the circle). (a), (b) show Ω with smooth boundary. (c), (d) show Ω with non-smooth boundary.

A differential of f is defined as an element of $T_p^*\Omega$ in usual way at points that lie in $(\Omega - \partial\Omega)$. To define a differential at a point p that lies on $\partial\Omega$, we take a Cauchy sequence of points in $\Omega - \partial\Omega$ that converges to p (recall that Ω is a complete metric space). The limit of the differentials of f in the cotangent cones of these points gives us the required differential at p. This limit, however, may or may not exists in $T_p^*\Omega$, and that will depend on the nature of f. The gradient of f is the corresponding dual of the differential in $T_p\Omega$.

2.5 Cut Locus

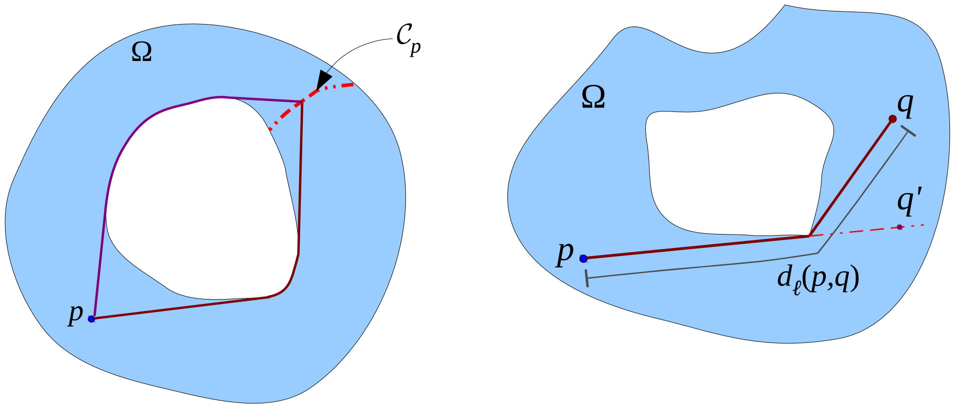
Definition D6 (Cut locus on manifolds with boundary – See Definition 3.4.II of [Wolter 85])**.** Let $p \in \Omega$. A point, $q \in (\Omega - \partial\Omega)$, is called a *pica* relative to p if p and q can be joined with two or more distinct minimal paths (*i.e.* paths of length $d_\ell(p,q)$) with distinct tangents at q. The closure of the set of all picas relative to p is called the *cut locus* of p and is denoted by $\mathcal{C}_p$ (Figure 5(a)).

Note that by using the term 'distinct minimal paths' we mean to impose the condition that the image of any two distinct paths are not the same. This excludes the multiplicity of paths due to mere re-parameterization. The definition of cut locus in terms of 'pica' (equivalently, *non-extenders*) is necessary for retaining certain properties of the standard cut locus on manifolds without boundary.

Lemma L2 (Wolter [Wolter 85])**.** *If Ω is a smooth manifold (with boundary), which is complete as a metric space, and can be defined as a subset of a smooth, complete manifold of same dimension, then for every $p \in \Omega$,*

i. $\mathcal{C}_p$ is a closed set of measure zero in Ω,

ii. the function $g_p := d_\ell(p,\cdot)$ is of class C^1 in $\Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$, and



(a) Points on $\mathcal{C}_p$ can be joined with p using 2 minimal paths.

(b) On points on the dotted line in this figure (*e.g.* the point q'), g_p is C^1, but not C^2.

Figure 5: The cut locus of p and the smoothness of the function g_p.

iii. the gradient of g_p is bounded in $\Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$.

Lemma L3. *For almost every point $p \in \Omega$: $q \notin \mathcal{C}_p \implies p \notin \mathcal{C}_q$.*

2.6 Gradient of Distance Function

Proposition P1. *Let $p \in \Omega$. The negative of the gradient of $g_p := d_\ell(p, \cdot)$ exists in $T_q\Omega$ (equivalently, the negative of the differential of g_p exists in $T_q^*\Omega$) at all points $q \in \Omega - (p \cup \mathcal{C}_p)$ that are equipped with a Riemannian metric in their neighborhoods. The negative of the gradient is equal to a normalized vector at q along the tangent to the minimal path connecting q to p (equivalently, the differential is the dual of the unit tangent vector along the minimal path).*

Remark on existence: If q is a point in $\Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$, the gradient (and its negative) obviously exist in $T_q\Omega$ (part 'ii.' of Lemma L2). We need to check the existence for points on the boundary $\partial\Omega$. Recall that for points, $q \in \partial\Omega$, we defined the gradient of a function as the limit of the gradients over a Cauchy sequence of points, $\{q_n\}$, in $\Omega - (p \cup \partial\Omega \cup \mathcal{C}_p)$ converging to q. We can prove the later half of the statement of the proposition for each of these q_n (*i.e.* that the negative of the gradient is equal to a normalized vector at q_n along the tangent to the minimal path connecting q_n to p). Then, the existence of the negative of the gradient at q is implied by the existence of the minimal path. Since the tangent to the minimal path connecting q to p exists in $T_q\Omega$ (by definition of the tangent cone), the negative of the gradient will also exist (see Figure 6).

Thus, the proof of the above proposition relies on being able to compute the gradient of g_p at $q \in \Omega - (p \cup \partial\Omega \cup \mathcal{C}_p)$ in terms of the tangent to the minimal path. Using the lemma and corollary that follows next, we will establish an explicit relationship between the gradient of g_p

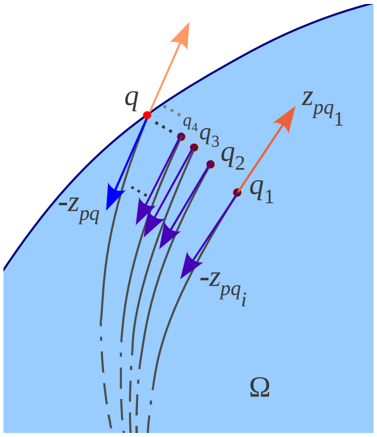


Figure 6: Illustration for Proposition P1. The negative of the gradient of g_p at $q \in \partial\Omega$ is the limit of $-z_{pq_i}$ as $i \to \infty$, where $q_1, q_2, \cdots$ is a Cauchy sequence converging to q.

and tangents to minimal paths using a coordinate representation of the metric tensor, and using these will be able to prove the Proposition P1. We will assume summation over repeated indices following the Einstein summation convention, and coordinatize tangent spaces and the cotangent spaces by $\{\frac{\partial}{\partial x^i}\}_{i=1,2,\cdots,N}$ and $\{\, \mathrm{d}x^i\}_{i=1,2,\cdots,N}$ respectively [Jost 97, Petersen 06].

Notations: We will use regular italic letters to denote points and vectors (*e.g* $p \in \Omega$ and $u \in T_p\Omega$), and boldface to denote the coordinate representation of points (*e.g.*, $\mathbf{p} \in \mathbb{R}^N$) and coefficient vectors (*e.g.* $\mathbf{u} \in \mathbb{R}^N$) in a particular coordinate chart. Later in Section 4, we will use regular font to represent vertices in a discrete graph. We will typically use 'η' to denote the Riamannian metric tensor. Also, given a metric, d_*, on a manifold, M, and a coordinate chart $C = (U, \phi)$ on an open subset $U \subseteq M$, we define,

i. $d_*^C : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$, the metric restricted to U and described in terms of the coordinate chart C, as $d_*^C(\mathbf{a}, \mathbf{b}) = d_*(\phi^{-1}(\mathbf{a}), \phi^{-1}(\mathbf{b})), \ \ \forall\, \mathbf{a}, \mathbf{b} \in Img(\phi)$, and,

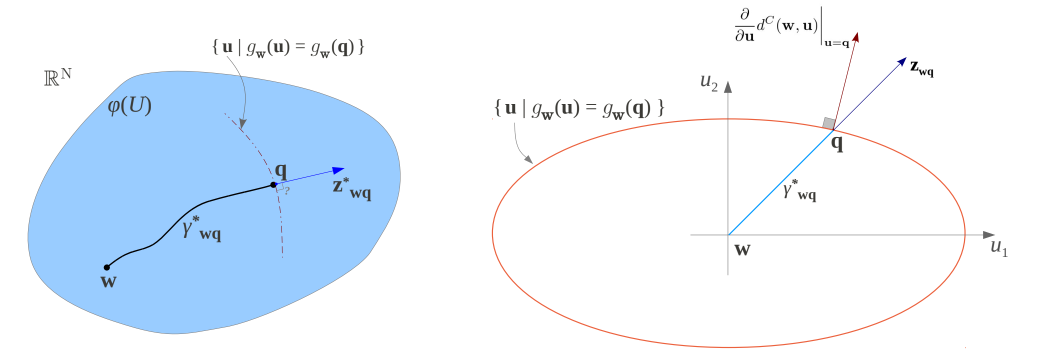
ii. $d_*^{\tilde{C}} : M \times \mathbb{R}^N \to \mathbb{R}$, the metric with the second parameter restricted to U and described in terms of the coordinate chart C, as $d_*^{\tilde{C}}(a, \mathbf{b}) = d_*(a, \phi^{-1}(\mathbf{b})), \ \ \forall\, \mathbf{b} \in Img(\phi)$.

Lemma L4. *Let U be a geodesically convex open set in a N-dimensional Riemannian manifold (possibly with boundary) Ω, equipped with a Riemannian metric tensor, η, at every point. We also assume that for all $p \in U$, $\mathcal{C}_p$ is empty inside U. Let d be the metric on U, induced by the Riemannian metric tensor, η (which agrees with the length metric, d_ℓ, since U is geodesically convex).*

Then for every coordinate chart, $C = (U, \phi)$, defined on U (with coordinate variables $u^1, u^2, \cdots, u^N$), and every $w, q \in U$, with $\mathbf{w} = \phi(w), \mathbf{q} = \phi(q)$, the following is true (Figure 7(a)),

$$\left[\frac{\partial}{\partial \mathbf{u}} d^C(\mathbf{w}, \mathbf{u})\bigg|_{\mathbf{u}=\mathbf{q}}\right]_i \equiv \frac{\partial}{\partial u^i} d^C(\mathbf{w}, \mathbf{u})\bigg|_{\mathbf{u}=\mathbf{q}} = \frac{\eta_{ij}(\mathbf{q})\; z^j_{\mathbf{wq}}}{\sqrt{\eta_{mn}(\mathbf{q})\; z^m_{\mathbf{wq}} z^n_{\mathbf{wq}}}}$$

where, $\mathbf{z}_{\mathbf{wq}} = [z^1_{\mathbf{wq}}, z^2_{\mathbf{wq}}, \cdots, z^N_{\mathbf{wq}}]^T$ is the coefficient vector (in coordinate chart C) of the tangent vector at q to the shortest geodesic connecting w to q, and by $\left[\frac{\partial f}{\partial \mathbf{u}}\right]_i$ we mean the i^{th} component of $\left[\frac{\partial f}{\partial u^1}, \frac{\partial f}{\partial u^2}, \cdots, \frac{\partial f}{\partial u^N}\right]$.


$\mathbb{R}^N$
$\varphi(U)$
$\{ \mathbf{u} \mid g_{\mathbf{w}}(\mathbf{u}) = g_{\mathbf{w}}(\mathbf{q}) \}$
$\mathbf{q}$
$\mathbf{z}^*_{\mathbf{wq}}$
$\gamma^*_{\mathbf{wq}}$
$\mathbf{w}$
$\frac{\partial}{\partial \mathbf{u}} d^C(\mathbf{w},\mathbf{u})\Big|_{\mathbf{u}=\mathbf{q}}$
$\mathbf{z}_{\mathbf{wq}}$
u_2
$\{ \mathbf{u} \mid g_{\mathbf{w}}(\mathbf{u}) = g_{\mathbf{w}}(\mathbf{q}) \}$
$\mathbf{q}$
$\gamma^*_{\mathbf{wq}}$
u_1
$\mathbf{w}$

(a) Illustration for Lemma L4 showing the relationship between the tangent to the geodesic $\gamma^*_{\mathbf{wq}}$ at $\mathbf{q}$, and the normal to the surface $\{\mathbf{u}|g(\mathbf{u}) = g(\mathbf{q})\}$ at $\mathbf{q}$.

(b) Illustration in a simple non-Euclidean, anisotropic metric. Note that the normal to the ellipse is not parallel to the tangent to the geodesic, $\mathbf{z}_{\mathbf{wq}}$. It is however parallel to the cotangent, $\mathbf{z}^*_{\mathbf{wq}}$, with coefficients $z^*_{i,\mathbf{wq}} = \eta_{ij}(\mathbf{q})\, z^j_{\mathbf{wq}}$.

Figure 7: Relationship between tangent to a geodesic and the derivative of the distance function.

If we define $g_{\mathbf{w}}(\mathbf{u}) := d^C(\mathbf{w},\mathbf{u})$, $\forall \mathbf{u} \in \mathit{Img}(\phi)$ (*i.e.*, $g_{\mathbf{w}}(\mathbf{q})$ is the length of the shortest geodesic connecting w to q), the statement of the proposition essentially implies that the normals to the constant $g_{\mathbf{w}}$ surfaces are parallel to the dual of the tangents (cotangents) to the geodesics. This is illustrated in Figure 7(a). The statement of the proposition essentially expresses the gradient of the distance function d (with respect to one of its arguments) in terms of the tangent to the geodesic connecting two points.

Examples:

1. We note that when the metric tensor is Euclidean in the given chart (*i.e.* $\eta_{ij} = \delta_{ij}$ everywhere as was the case in [Pimenta 08]), the result of the proposition simply reduces to $\frac{\partial}{\partial u^i} d(\mathbf{w},\mathbf{u})\big|_{\mathbf{u}=\mathbf{q}} = z^i_{\mathbf{wq}}$. This is no surprise since we know that the vector $\frac{\partial}{\partial \mathbf{u}} d(\mathbf{w},\mathbf{u})\big|_{\mathbf{u}=\mathbf{q}}$ is essentially an unit normal to the sphere with center $\mathbf{w}$ (which is the surface of constant $d(\mathbf{w},\mathbf{u})$) at the point $\mathbf{u} = \mathbf{q}$, which is well-known to be parallel to the straight line connecting $\mathbf{w}$ to $\mathbf{q}$ (a radial line of the sphere).

2. If the metric is locally isotropic in the given chart (*i.e.* if the matrix representation of the metric is a multiple of the identity matrix at every point), and can be written as $\eta_{ij}(\mathbf{w}) = \zeta(\mathbf{w})\delta_{ij}$ for some $\zeta : \mathbb{R}^N \to \mathbb{R}$, then the result of the proposition reduces to $\frac{\partial}{\partial \mathbf{u}} d^C(\mathbf{w},\mathbf{u})\big|_{\mathbf{u}=\mathbf{q}} = \sqrt{\zeta(\mathbf{q})}\; \mathbf{z}^T_{\mathbf{wq}}$ (where, $\mathbf{z}^T_{\mathbf{wq}} = [z^1_{\mathbf{wq}}, z^2_{\mathbf{wq}}, \cdots, z^N_{\mathbf{wq}}]$ is the transpose of the coefficient vector, $\mathbf{z}_{\mathbf{wq}}$, of the tangent to the geodesic).

3. Finally, we consider a simple, yet nontrivial, example of a non-Euclidean, anisotropic metric. Consider this matrix representation of the metric tensor: $\eta_{\bullet\bullet} = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$ (in coordinate chart C). Since the Christoffel symbols vanish in this coordinate chart, one can infer from the geodesic equation that the geodesics are essentially represented by straight lines when plotted with u^i as orthogonal axes (Figure 7(b)). However, the curves of constant distance from $\mathbf{w}$ become ellipses centered at $\mathbf{w}$ and with aspect ratio of 2. Now consider the point $\mathbf{q} = \mathbf{w} + [1,1]^T$. A direct computation of the normal at this point to the ellipse, $(u^1 -$

$w^1)^2/4 + (u^2 - w^2)^2 = c$, passing through this point, reveals the coefficient co-vector of $\frac{\partial}{\partial \mathbf{u}} d^C(\mathbf{w}, \mathbf{u})\big|_{\mathbf{u}=\mathbf{q}}$ to be parallel to $[\frac{1}{2}, 2]$. However, the coefficient vector of the tangent to the geodesic is $\mathbf{z_{wq}} = [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}]^T$. This gives the following: $\sqrt{\eta_{mn}(\mathbf{q})\, z^m_{\mathbf{wq}} z^n_{\mathbf{wq}}} = \sqrt{\frac{5}{2}}$, $z_{1,\mathbf{wq}} = \sum_j \eta_{1j}\, z^j_{\mathbf{wq}} = \frac{1}{\sqrt{2}}$, $z_{2,\mathbf{wq}} = \sum_j \eta_{2j}\, z^j_{\mathbf{wq}} = 2\sqrt{2}$. Thus, the coefficient co-vector of $\mathbf{z}^*_{\mathbf{wq}}$ is parallel to $[\frac{1}{\sqrt{2}}, 2\sqrt{2}]$. This indeed is parallel to $[\frac{1}{2}, 2]$. The exact computation of the scalar multiple will require a more careful computation of $\frac{\partial}{\partial \mathbf{u}} d^C(\mathbf{w}, \mathbf{u})$.

In the following corollary we relax some conditions imposed on U in the Lemma L4 so that we can generalize the result to manifolds that are not necessarily geodesically convex.

Corollary C1. *Let Ω be a N-dimensional manifold possibly with boundary, which is complete as a metric space, with length metric d_ℓ. Let $p \in \Omega$ and $q \in \Omega - (p \cup \partial\Omega \cup \mathcal{C}_p)$. Let q be such that the metric in its neighborhood is induced by a Riemannian metric tensor, η. Let d_ℓ be the length metric on Ω.*

Then the following holds for every coordinate chart, $D = (V, \psi)$, defined on open set $V \ni q$,

$$\left[\frac{\partial}{\partial \mathbf{u}} d_\ell^{\tilde{D}}(p, \mathbf{u})\bigg|_{\mathbf{u}=\mathbf{q}} \right]_i \equiv \frac{\partial}{\partial u^i} d_\ell^{\tilde{D}}(p, \mathbf{u})\bigg|_{\mathbf{u}=\mathbf{q}} = \frac{\eta_{ij}(\mathbf{q})\; z^j_{\mathbf{pq}}}{\sqrt{\eta_{mn}(\mathbf{q})\; z^m_{\mathbf{pq}} z^n_{\mathbf{pq}}}}$$

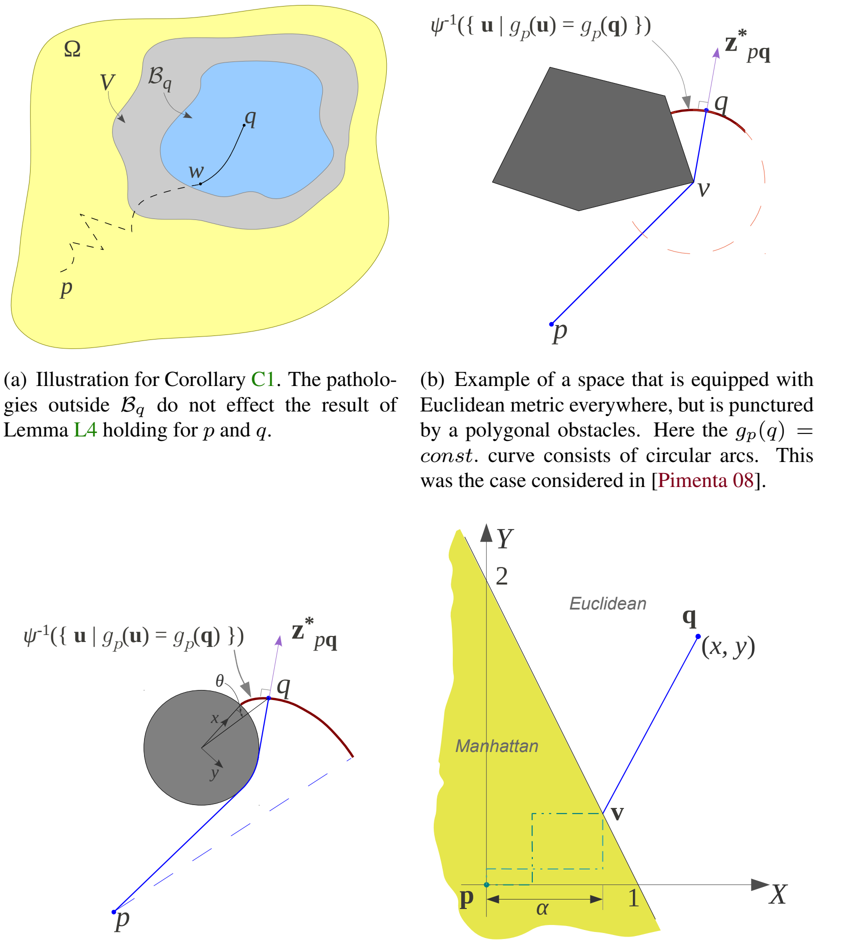
where, $\mathbf{q} = \psi(q)$, and $\mathbf{z_{pq}} = [z^1_{\mathbf{pq}}, z^2_{\mathbf{pq}}, \cdots, z^N_{\mathbf{pq}}]^T$ is the coefficient vector (in coordinate chart D) of the tangent vector at q to the shortest geodesic connecting $p \in \Omega$ to $q \in V$. By $\left[\frac{\partial f}{\partial \mathbf{u}}\right]_i$ we mean the i^{th} component of $\left[\frac{\partial f}{\partial u^1}, \frac{\partial f}{\partial u^2}, \cdots, \frac{\partial f}{\partial u^N}\right]$.

*Remark R4***.** This Corollary is applicable to a wider class of metric spaces than Lemma L4. Here we only need to assume a Riemannian metric in the neighborhood of q (Figure 8(a)). This will enable us to use the result for *locally* Riemannian manifolds allowing for pathologies outside local neighborhoods (*e.g.* boundaries/holes/punctures/obstacles – the kind of spaces we are most interested in), as well as opens up possibilities for more general metric spaces that may not be Riemannian outside a small ball neighborhood of q, $\mathcal{B}_q$ (e.g. Manhattan metric in $\mathcal{M} \subset \Omega$, Riemannian metric elsewhere). However, in this paper we will not consider the later kind of cases.

Examples:

1. The simplest example of a space that is only locally Euclidean is one that is $\mathbb{R}^2$ punctured by polygonal obstacles (Figure 8(b)). Due to the 'pointedness' of the obstacles, the constant-g_p manifolds are essentially circular arcs centered at p or a vertex v of a polygon. Thus, as illustrated by Figure 8(b), the normals to the arcs are parallel to the tangent to the segment joining v to q.

2. A less trivial case is seen in the example where the obstacles have curved boundaries. Then the corollary essentially reduces to the assertion that the normal at any point on an involute [Cundy 89] is parallel to the 'taut string', the end of which traces the involute – and this is true irrespective of the curve used to generate the involute. While the statement has an obvious intuitive explanation by considering the possible directions of motion of the end of the taut string, we provide an explicit computation for an involute created using a circle (Figure 8(c)). Consider a taut string unwrapping off a circle of radius r (starting from $\theta = 0$ when


Ω
V
$\mathcal{B}_q$
q
w
p
$\psi^{-1}(\{\ \mathbf{u} \mid g_p(\mathbf{u}) = g_p(\mathbf{q})\ \})$
$\mathbf{z}^*_{\mathbf{pq}}$
q
v
p
$\psi^{-1}(\{\ \mathbf{u} \mid g_p(\mathbf{u}) = g_p(\mathbf{q})\ \})$
$\mathbf{z}^*_{\mathbf{pq}}$
θ
q
x
y
p
Y
2
Euclidean
q
(x, y)
Manhattan
v
p
α
1
X

(a) Illustration for Corollary C1. The pathologies outside $\mathcal{B}_q$ do not effect the result of Lemma L4 holding for p and q.

(b) Example of a space that is equipped with Euclidean metric everywhere, but is punctured by a polygonal obstacles. Here the $g_p(q) = const.$ curve consists of circular arcs. This was the case considered in [Pimenta 08].

(c) An example with a circular obstacle where the $g_p(q) = const.$ curve consists of an involute and circular arc.

(d) An example involving Manhattan and Euclidean metrics in two different regions. The distance function between points lying in the two different regions is given by $d(\mathbf{p}, \mathbf{q}) = \min_{\mathbf{v}} (d_{man}(\mathbf{p}, \mathbf{v}) + d_{eu}(\mathbf{v}, \mathbf{q}))$, where $\mathbf{v}$ is a point lying on the boundary of the two regions.

Figure 8: Corollary C1 and illustrative examples.

it is completely wrapped). Thus, when the string has unwrapped by an angle θ, the string points at a direction $[\sin(\theta), -\cos(\theta)]^T$. Now, it is easy to verify that the involute is described by the parametric curve $x = r(\cos(\theta) + \theta\sin(\theta)), y = r(\sin(\theta) - \theta\cos(\theta))$. Thus we have $\frac{\mathrm{d}x}{\mathrm{d}\theta} = \theta\cos(\theta)$, $\frac{\mathrm{d}y}{\mathrm{d}\theta} = \theta\sin(\theta)$. Thus the normal to the involute pointing in the direction $[\frac{\mathrm{d}y}{\mathrm{d}\theta}, -\frac{\mathrm{d}x}{\mathrm{d}\theta}]$ is indeed parallel to the direction in which the string points.

3. The last example that we will illustrate involves a mixture of Manhattan distance (in a particular given coordinate chart) and Euclidean metric. Consider the case in Figure 8(d), where in the given coordinate chart $\mathbf{p}$ is the origin, $(0, 0)$. For any two points inside the half-plane $\mathcal{M} = \{(x, y) | 2x + y < 2\}$, the distance function is the Manhattan distance. Outside $\mathcal{M}$ it is induced by Euclidean metric. It is to be noted that although the distance function is defined in $\mathcal{M}$, geodesics are not uniquely defined. Let us consider the point $\mathbf{q} = (x, y)$ outside $\mathcal{M}$ (so that there exists a $\mathcal{B}_\mathbf{q}$ as required by Corollary C1). The distance is given by $d(\mathbf{p}, \mathbf{q}) = \min_{\alpha\in\mathbb{R}} \left(\alpha + 2(1-\alpha) + \sqrt{(x-\alpha)^2 + (y - 2(1-\alpha))^2}\right)$. Denoting the quantity inside 'min' by $f(\alpha)$, and by solving $\frac{\partial f}{\partial \alpha} = 0$, one obtains the unique solution $\alpha = (4x - 3y + 6)/10$. This gives $d(\mathbf{p}, \mathbf{q}) = (3x + 4y + 12)/5$. Thus, the normals to the constant-d surfaces are parallel to $[3/5, 4/5]$. Again, the segments $\overline{\mathbf{vq}}$ have tangent pointing in the direction $[x - \alpha, y - 2(1-\alpha)]^T = [(6x + 3y - 6)/10, (8x + 4y - 8)/10]^T$. Thus we see that they are indeed parallel.

2.7 Minimization of the Second Moment Function

Definition D7 (Second moment about a point)**.** Let Λ be a N-dimensional manifold (possibly with boundary), $w : \Lambda \to \mathbb{R}_+$ a continuous (and bounded) *weight function* on it, and $f(x) = x^2 + c,\ x \in \mathbb{R}_+$. If d_* is a metric on Λ, we define the 2^{nd} *moment* of Λ about a point $p \in \Lambda$ as

$$\mathcal{M}(p) = \int_\Lambda f\left(d_*(p, q)\right)\ w(q)\ \ \mathrm{d}q$$

where $\mathrm{d}q$ is a volume N-form on Λ. We call $\mathcal{M} : \Lambda \to \mathbb{R}$ the *second moment function* on Λ (with d_* being the metric of choice). The definition, more generally, is valid for any measurable metric space.

The minima of $\mathcal{M}$ are of interest to us, which we will study in this section. We first describe the standard notion of a *generalized centroid*. We will illustrate how, for many manifolds with boundary, the set of generalized centroids form degenerate sets and/or are not unique. However, we will also discuss its uniqueness for some classes of metric spaces.

The *generalized centroid* of Λ is defined as $p^* = \underset{p\in\Lambda}{\operatorname{argmin}}\ \mathcal{M}(p)$. In cases where the minimizing point is not unique, we define the 'set of generalized centroids' as

$$P^* = \left\{ p \in \Lambda \;\middle|\; \mathcal{M}(p) = \min_{r\in\Lambda} \mathcal{M}(r) \right\}$$

Note that the the constant c in the definition of f is redundant in the definition of p^* since the presence of it simply adds a constant term to $\mathcal{M}$ (the objective of the minimization problem).

> *Degeneracy of generalized centroid in spaces that are not simply-connected:* We consider a couple of very simple example for $N = 1$ to illustrate how the generalized centroid is degenerate in many manifolds that are not simply-connected. Consider a

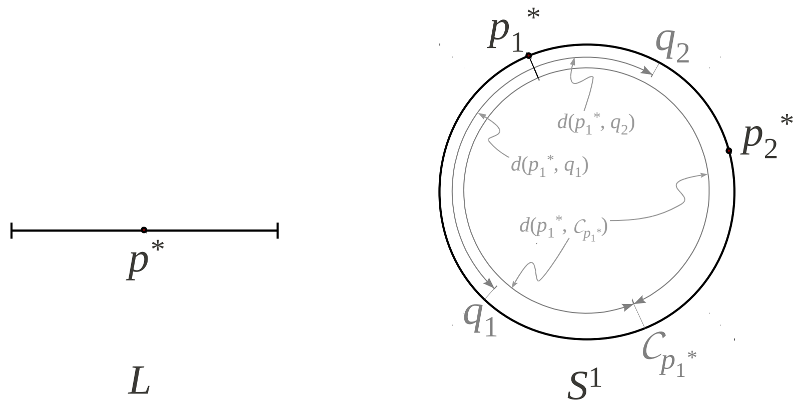


Figure 9: Degeneracy of *generalized centroid* in spaces that are not simply-connected. For a line segment, L, there is an unique point, p^* such that $\mathcal{M}(p^*) = \min_{p \in L} \mathcal{M}(p)$. But for the circle, S^1, $\mathcal{M}(p_1^*) = \mathcal{M}(p_2^*) = \min_{p \in S^1} \mathcal{M}(p) \ = \mathcal{M}(p'), \forall p' \in S^1$.

line segment, L, with its usual length metric d_ℓ and constant weight $w(q) = 1,\ \forall q \in L$. Clearly the *generalized centroid* of L is its usual 'center' and is unique (Figure 9). However, now let's consider a circle, S^1, once again with usual length metric and constant weight. Note that for any $p \in S^1$, the diametrically opposite point constitutes the cut locus, $\mathcal{C}_p$. It is obvious (due to symmetry) that the value of $\mathcal{M}(p)$ will be same for all $p \in S^1$, and hence the entire space is P^*. Likewise, for a flat annular region one can easily see that the generalized centroid is degenerate and hence not unique.

We now consider the problem of minimizing $\mathcal{M}$ using the method of gradient decent.

Lemma L5 ($\mathcal{M}$ is arbitrarily close to a C^1 function)**.** *Let Λ be a smooth, compact Riemannian manifold (possibly with smooth boundary) with the length metric d_ℓ induced by the Riemannian metric tensor. Then $\mathcal{M}$ is arbitrarily close to a C^1 function on Λ (i.e., C^1 everywhere except possibly a set of measure zero). Moreover, $\mathcal{M}$ is guaranteed to be C^1 if $\mathcal{C}_{p'} = \emptyset,\ \forall p' \in \Lambda$ (i.e., Λ does not contain any cut loci).*

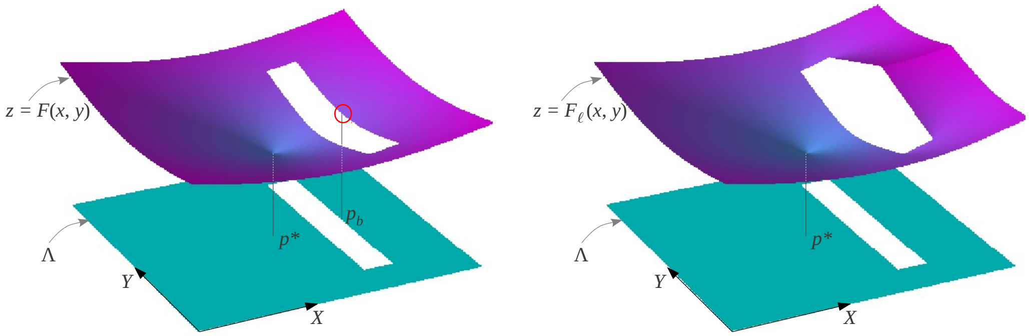
As a consequence of the above lemma, for all problems involving minimization of $\mathcal{M}$, we can instead minimize the corresponding *smoothed* function, $\overline{\mathcal{M}}$, and obtain results that are arbitrarily close to the un-smoothed problem. In the discussions that follow, we will use $\mathcal{M}$ to refer to the *smoothed* version of the second moment function so as to avoid a plethora of notations. Thus we will be able to compute gradients of $\mathcal{M}$.

Lemma L6. *[Local minima of $\mathcal{M}$ lie in the interior of Λ] Let Λ be as before. If the length metric d_ℓ is used to compute $\mathcal{M}$, there does not exist a local minimum of $\mathcal{M}$ on $\partial\Lambda$ (i.e., all local minima lie in the interior of Λ).*

Corollary C2. *Suppose the length metric, d_ℓ, is the metric of choice for computing $\mathcal{M}$. Then some immediate consequence of the above lemma are*

i. A gradient decent of $\mathcal{M}$ in Λ (with small noise so as to avoid saddle points) will converge to a point in $\Lambda - \partial\Lambda$ (interior of Λ).

ii. Generalized centroids of Λ lie in $\Lambda - \partial\Lambda$.



(a) The function $F : \Lambda \to \mathbb{R}$ has a local minimum at $p_b \in \partial\Lambda$. This minimum appears because of the boundary created by the rectangular obstacle/puncture in Λ.

(b) The function $F_\ell : \Lambda \to \mathbb{R}$ does not have a local minimum on the boundary even though there is a obstacle/puncture in Λ. For this function, p^* is the only local (and global) minimum.

Figure 10: Functions with and without local minima on the boundary. Using the length metric gets rid of local minima on the boundary.

The implication of Corollary C2.i. is that a gradient decent of $\mathcal{M}$ does not get *stuck at a boundary point* (equivalently, all integral curves of $-\nabla\mathcal{M}$, starting at $p \in \Lambda$ with direction $v \in T_p\Lambda$, exists in Λ — see Section 2.2 for detailed discussion on this phenomenon). This is illustrated in Figure 10.

Notes on uniqueness: As discussed earlier, the minima of $\mathcal{M}$ can be degenerate. In such cases (like the example of a circle, S^1, as discussed) the notion of a *generalized centroid* as an unique point becomes ambiguous. If however the generalized centroid is unique (*i.e.*, P^* is a singleton), it is possible that there can be local minima of $\mathcal{M}$ other than the generalized centroid (all of which lie in $\Lambda - \partial\Lambda$ due to the aforesaid corollary and lemma).

However, there are large classes of metric spaces for which the generalized centroid is not only unique, it is also the only local minimum of $\mathcal{M}$. We now list a few of those known metric spaces along with relevant references

i. It is well-known and easy to prove that the generalized centroid of a convex subset of Euclidean space (with its usual Euclidean metric) is unique and the only local minimum of $\mathcal{M}$.

ii. If Λ_R is a geodesically convex Riemannian manifold with its usual distance function, d (which agrees with d_ℓ in this case), and is free of cut loci, it was proved by Eugenio Calabi that the the generalized centroid is an unique point and that it is the only local minimum of $\mathcal{M}$ (*i.e.*, unique point where the gradient of $\mathcal{M}$ vanishes). See Section 6.1.5 of [Berger 03].

iii. Certain generalizations of the above was achieved in [Corcuera 98] for some special cases by considering an additional weight function (modeled as probability distribution).

iv. C_k domains (with $k \geq 0$) of a general metric space, (Λ, d_*), [Ohta 07, Palfia 11] satisfy the condition that for any point z in the domain and a minimal path, $\gamma : [0, 1] \to \Lambda$, connecting points x and y in the domain, the following holds: $d_*(z, \gamma(t))^2 \leq (1 - \tau)d_*(x, z)^2 + \tau d_*(z, y)^2$ for some $\tau \in (0, 1)$. For such domains it is easy to show that the generalized centroid is indeed unique and is the only local minimum of $\mathcal{M}$.

3 The coverage problem in robotics

In this section we discuss the derivation of the control laws in the continuous-time version of the Lloyd's algorithm. We draw parallels with the classical case for convex environments with Euclidean metric [Lloyd 82] to motivate our generalization to general manifolds with boundary and general Riemannian metric.

Background: Let Ω be a path-connected metric space that represents the environment, equipped with a metric, $d_* : \Omega \times \Omega \to \mathbb{R}_+$. In [Lloyd 82, Cortes 04, Pimenta 08] Ω is assumed to be a convex subset of $\mathbb{R}^N$ and is equipped with the Euclidean metric tensor at every point. However, in the present scenario we relax d_* to a more general metric on a metric space, Ω, that need not necessarily be a manifold. We will eventually consider the length metric, d_ℓ, on manifolds with boundary, induced by a general Riemannian metric. We will introduce those restrictions gradually as they are required.

Coverage functional: Suppose there are n mobile robots in Ω. The position of the k^{th} robot is represented by $p_k \in \Omega$. By definition, a *tessellation* [Lloyd 82, Cortes 04, Pimenta 08] of the environment is a partition of Ω, written as $\{W_k\}_{k=1,2,\cdots,n}$, such that each W_k (called a *tessellum* or a *cell*) is path connected, $p_k \in W_k$, $\mathit{Int}(W_k) \cap \mathit{Int}(W_l) = \emptyset, \forall k \neq l$ (where $\mathit{Int}(\Lambda) = \Lambda - \partial\Lambda$ denotes the *interior* of Λ), and $\cup_{k=1}^{n} W_k = \Omega$. The *tessellum* associated with the k^{th} robot is $W_k, \forall k = 1, 2, \ldots, n$. For a given set of robot positions $P = \{p_1, p_2, \ldots, p_n\}$ and tessellation $W = \{W_1, W_2, \ldots, W_n\}$, the *coverage functional* is defined as:

$$\mathcal{H}(P, W) = \sum_{k=1}^{n} \mathcal{H}(p_k, W_k) = \sum_{k=1}^{n} \int_{W_k} f_k(d_*(q, p_k))\, w(q)\, \mathrm{d}q \tag{1}$$

where $f_k : \mathbb{R}_+ \to \mathbb{R}$ are smooth and strictly increasing functions, $w : \Omega \to \mathbb{R}_+$ is a weight or density function, and $\mathrm{d}q$ represents an infinitesimal volume element (more formally, it is a top-dimensional *measure* of an an infinitesimal element, associated with the metric d_* [Gromov 99]).

The name "*coverage functional*" is indicative of the fact that $\mathcal{H}$ measures how *bad* the coverage is — *i.e.*, the more well-distributed the robots are throughout the environment, the lower is the value of $\mathcal{H}$. In fact, for a given set of initial robot positions, P, we devise a control law that minimizes the function $\tilde{\mathcal{H}}(P) := \min_W \mathcal{H}(P, W)$ (*i.e.* the best value of $\mathcal{H}(P, W)$ for a given P). It is easy to show [Lloyd 82, Cortes 04] that $\tilde{\mathcal{H}}(P) = \mathcal{H}(P, V)$, where $V = \{V_1, V_2, \cdots, V_n\}$ is the Voronoi tessellation given by

$$V_k = \{q \in \Omega \mid f_k(d_*(q, p_k)) \leq f_l(d_*(q, p_l)), \forall l \neq k\} \tag{2}$$

Requiring Ω to be a manifold with boundary: The control law for minimizing $\tilde{\mathcal{H}}(P) = \sum_{k=1}^{n} \int_{V_k} f_k(d_*(q, p_k)) w(q)\, \mathrm{d}q$ can be reduced to the problem of moving along the direction of its steepest descent. So far in this section we have not required that Ω be a manifold. However, we would like to be able to compute the gradient of $\tilde{\mathcal{H}}(P)$ so that the negative of it is the direction of steepest descent. It suffices that the configuration space be a manifold so that each point on it has a tangent space in which the gradient of a function (or its negative) will reside. However, based on our discussions in Section 2, we are now capable of defining and computing gradients on manifolds with boundary (see Remark *R3*). However, we need to check the existence of the negative of the gradient inside the *tangent cone* at a point on the boundary, so that the robots can move along that direction. In the following discussion we use explicit coordinate charts, $C_k = (U_k, \phi_k)$, with $p_k \in U_k$ — the position of the k^{th} robot, at which we compute the gradients. Using the notation introduced in Corollary C1, we write $d_*^{\tilde{C}_k}(a, \mathbf{b}) := d_*(a, \phi_k^{-1}(\mathbf{b}))$ for $\mathbf{b} \in \mathit{Img}(\phi_k)$.

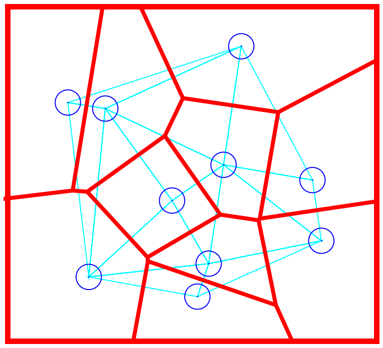


Figure 11: Voronoi tessellation of a convex Ω (rectangular region) with $n = 10$ robots (marked by small circles). The bold line segments show the boundary of the Voronoi cells. Note how a boundary segment is the perpendicular bisector of the thin line joining the robots sharing the boundary segment.

The domains of integration, V_k are functions of P, and hence the gradient of $\tilde{\mathcal{H}}(P)$ would, in general, involve boundary terms, ∂V_k. However, it can be shown using methods of differentiation under integration [Cortes 04, Pimenta 08] that the boundary terms vanish. So the final formula for the partial derivatives of $\tilde{\mathcal{H}}(P)$ with respect to the position of the robots become,

$$\frac{\partial \tilde{\mathcal{H}}(P)}{\partial \mathbf{p}_k} = \int_{V_k} \frac{\partial}{\partial \mathbf{p}_k} f_k(d_*^{\tilde{C}_k}(q, \mathbf{p}_k)) \; w(q) \; \mathrm{d}q \tag{3}$$

In practice, it is adequate to choose $f_k(x) = x^2$ for most implementations [Cortes 04]. However, a variation of the problem for taking into account finite sensor footprint of the robots, constructs a *power Voronoi tessellation* [Brown 79], in which one chooses $f_k(x) = x^2 - R_k^2$, where R_k can represent, for example, the radius of the sensor footprint of the k^{th} robot. In this paper we will be working with the following form for f_k

$$f_k(x) = x^2 + c_k \tag{4}$$

Requiring d_* to be a metric induced by Riemannian metric tensor: Until now we haven't made any major assumption on the distance function d_* besides the fact that it is a metric on a manifold with boundary that can be differentiated at the points p_k. We next impose the condition that the metric is locally represented by a Riemannian metric tensor, η.

Remark R5. If the space Ω is convex, and we can construct a single coordinate chart, $B = \{\Omega, \psi\}$, over entire Ω such that the matrix representation of the metric tensor is Euclidean everywhere, then d_* is the Euclidean distance given by $d_*^B(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$ (where, $d_*^B(\mathbf{a}, \mathbf{b}) = d_*(\psi^{-1}(\mathbf{a}), \psi^{-1}(\mathbf{b}))$ – refer to notation introduced in Lemma L4). This was the case considered in [Lloyd 82] (see Figure 11). Under this assumption, and using the form of f_k in (4), the formula of (3) can be simplified to obtain

$$\frac{\partial \tilde{\mathcal{H}}(P)}{\partial \mathbf{p}_k} = 2A_k(\mathbf{p}_k - \mathbf{p}_k^*) \tag{5}$$

where, $A_k = \int_{V_k} w(q) \; \mathrm{d}q$ is the weighted *volume* of V_k, and $\mathbf{p}_k^* = \frac{\int_{V_k} \mathbf{q} \; w(q) \; \mathrm{d}q}{A_k}$ (with $\mathbf{q} = \psi(q)$, the coordinate representation of q) is the *weighted centroid* of V_k. Moreover,

> the Euclidean distance function makes computation of the Voronoi tessellation very easy: V, due to Equation (2), can be constructed from the *perpendicular bisectors* of the line segments $\overline{\mathbf{p}_k\mathbf{p}_l}$, $\forall k \neq l$ in the coordinate chart B, thus making each V_k a convex polygon, which are also simply connected (Figure 11). This also enable closed-form computation of the volume, A_k, and the centroid, $\mathbf{p}_k^*$ when the weight function, w, is uniform. Equation (5) yields the simple control law (in coordinate chart B) in continuous-time Lloyd's algorithm: $\mathbf{u}_k = -\kappa A_k(\mathbf{p}_k - \mathbf{p}_k^*)$, with some positive *gain*, κ. Lloyd's algorithm [Lloyd 82] and its continuous-time asynchronous implementations [Cortes 04] are distributed algorithms for minimizing $\mathcal{H}(P, W)$ with guarantees on completeness and asymptotic convergence to a local optimum, when Ω is convex Euclidean. However, this simplification does not work when there does not exist a coordinate chart as B in which the metric can be expressed as the Euclidean norm of difference for every pair of points — an inherent characteristic of spaces with holes/obstacles or with metric that is intrinsically non-Euclidean.

Requiring d_* to be the length metric: We also impose the condition that the metric be the length metric, *i.e.* $d_* = d_\ell$, that was introduced in Section 2.3. In the following discussion we will hence use results derived in Section 2 to make assertions about the existence of the gradient, and hence the convergence.

Now, with $f_k(x) = x^2 + c_k$, we have $\frac{\partial}{\partial \mathbf{p}_k} f_k(d_\ell^{\tilde{C}_k}(q, \mathbf{p}_k)) = 2\ d_\ell(q, p_k)\ \frac{\partial}{\partial \mathbf{p}_k} d_\ell^{\tilde{C}_k}(q, \mathbf{p}_k)$ where, $\mathbf{p}_k = \phi_k(p_k)$. Substituting this into Equation (3) and using Proposition P1 (with the explicit formula from Corollary C1) we obtain,

$$\begin{aligned}\left[\frac{\partial \tilde{\mathcal{H}}(P)}{\partial \mathbf{p}_k}\right]_i &= 2 \int_{V_k} d_\ell(q, p_k)\ \frac{\partial}{\partial \mathbf{p}_k} d_\ell^{\tilde{C}_k}(q, \mathbf{p}_k)\ \ w(q)\ \ \mathrm{d}q \\ &= 2 \int_{V_k - (p \cup \mathcal{C}_{p_k})} d_\ell(q, p_k)\ \frac{\eta_{ij}(\mathbf{p}_k)\ z^j_{q\mathbf{p}_k}}{\sqrt{\eta_{mn}(\mathbf{p}_k)\ z^m_{q\mathbf{p}_k} z^n_{q\mathbf{p}_k}}}\ \ w(q)\ \ \mathrm{d}q \qquad (6)\end{aligned}$$

where $z^j_{q\mathbf{p}_k}$ is the j^{th} component of the coefficient vector (in coordinate chart C_k) of the tangent at p_k to the minimal path connecting q and p_k.

That we can write the second equality is due to the fact that $p_k \cup \mathcal{C}_{p_k}$ is a set of measure zero in V_k (a consequence of Lemma L2), outside which $\frac{\partial}{\partial \mathbf{p}_k} d_\ell^{\tilde{C}_k}(\cdot, \mathbf{p}_k)$ exists (a consequence of Proposition P1 and and Lemma L3): The domain of integration in the second integral implies $q \notin \mathcal{C}_{p_k}$. This, due to Lemma L3, implies that $p_k \notin \mathcal{C}_q$ almost always (*i.e.* for all p_k, except possibly for a set a measure zero in Ω). Also, $p_k \neq q$. This, due to Proposition P1, implies the gradient of $d_\ell(q, \cdot)$ exists at p_k. Thus we could write the quantity inside the second integral. Moreover, due to part 'i' of Lemma L2, $p_k \cup \mathcal{C}_{p_k}$ is a set of measure zero in V_k (since V_k is not a set of measure zero in Ω). Also, due to part 'iii.' of the same lemma, the gradient at the points in the neighborhoods of $p_k \cup \mathcal{C}_{p_k}$ is finite. Thus, removing $p_k \cup \mathcal{C}_{p_k}$ from the domain of integration does not change the value of the integral, but lets us compute the integral.

We consider the negative of the quantity in (6). The $\left[-\frac{\partial \tilde{\mathcal{H}}(P)}{\partial \mathbf{p}_k}\right]_i$ are coefficients (in chart C_k) of a covector that resides in $T^*_{p_k}\Omega$. The corresponding vector (the dual) will thus have coefficients $\left[-\frac{\partial \tilde{\mathcal{H}}(P)}{\partial \mathbf{p}_k}\right]_i \eta^{il}(\mathbf{p}_k)$ (where $\eta^{\bullet\bullet}$ is matrix inverse of the matrix representation of the metric tensor, $\eta_{\bullet\bullet}$ [Jost 97, Petersen 06]). This is the vector along which the k^{th} robot needs to be moved for reducing the value of $\tilde{\mathcal{H}}$ the most. Using a scalar gain of κ and noting that $\eta_{ij}\eta^{il} = \delta^l_j$, the l^{th}

coefficients of the control vector for the k^{th} robot will be

$$\begin{aligned}
[\mathbf{u}_k]^l \ =: \ u_k^l &= \kappa \left[-\frac{\partial \tilde{\mathcal{H}}(P)}{\partial \mathbf{p}_k} \right]_i \eta^{il}(\mathbf{p}_k) \\
&= 2\,\kappa \int_{V_k - (p_k \cup \mathcal{C}_{p_k})} d_\ell(q, p_k) \, \frac{-z^l_{q\mathbf{p}_k}}{\sqrt{\eta_{mn}(\mathbf{p}_k)\, z^m_{q\mathbf{p}_k} z^n_{q\mathbf{p}_k}}} \, w(q) \;\; \mathrm{d}q \qquad (7)
\end{aligned}$$

We now test if this vector actually exists in $T_{p_k}\Omega$ for all p_k (including ones on $\partial\Omega$). Clearly $-z^j_{q\mathbf{p}_k} \frac{\partial}{\partial x^j}$ exists in $T_{p_k}\Omega$, since the minimal path connecting p_k and q exists (see proof of Proposition P1). Now we recall that $T_{p_k}\Omega$ is a convex cone almost everywhere in Ω due to Lemma L1 and Remark *R1* (except for possibly isolated points on $\partial\Omega$, in which case we either smoothen the boundary near that point or rely on presence of small noise that would 'push' the robot out that isolated point – see Remark *R2*). Thus a positive linear combination of vectors in $T_{p_k}\Omega$ (which the quantity in (7) is) will also be in $T_{p_k}\Omega$. This essentially implies that if the k^{th} robot follows this control law, it won't get 'stuck' at a point on the boundary. It will stop only when the gradient of $\tilde{\mathcal{H}}$ becomes zero at a point in the interior of V_k, making the control vector zero (see the illustration in Figure 10).

The aforesaid convergence to a non-boundary point is also obvious from Corollary C2.i. by noting that the control velocity of the k^{th} robot is nothing but the gradient of the second moment function of the Voronoi cell, V_k, computed using the length metric d_ℓ (note that a point in the interior of V_k will also be in the interior of Ω). As a consequence of the discussions above, we state the final result on convergence guarantee formally in the proposition below:

Proposition P2. *Robots navigating on a Riemannian manifold with boundary, using the control law of Equation* (7)*, will asymptotically converge to points in the the interior of Ω (i.e., $\Omega - \partial\Omega$), where the gradients of the second moment functions on their respective Voronoi cells (computed using the length metric on Ω) vanish.*

Although there can be local minima of the second moment function on the interior of V_k which are different from a generalized centroid (global minima), as discussed in Section 2.7 there exists many metric spaces and/or domains in metric spaces where there is an unique local minimum of the second moment function (namely the global minimum). It should however be noted that even in such cases the global minimum of $\tilde{\mathcal{H}}$ is not guaranteed to be attained. For example, even when Ω is a convex subset of $\mathbb{R}^N$, the original Lloyd's algorithm or Cortes' algorithm only guarantee convergence to a local minimum of $\tilde{\mathcal{H}}$.

Assuming uniform weight function, a few examples of manifolds with boundary on which the robots' Voronoi cells have an unique local minimum (namely the generalized centroid) are:

i. Convex subsets of Euclidean spaces.

ii. Whenever (Λ, d_ℓ) is $CAT(0)$ as a metric space (since in that case it is also a C_k space – see [Ohta 07] or [Palfia 11]). This include spaces with non-positive curvature everywhere.

iii. A complete sphere with number of robots, $n \geq 2$. In that case each Voronoi cell is a hemisphere or smaller, and a geodesically convex subset of a Riemannian manifold not containing any cut locus, as demonstrated in Figure 15 (see Section 6.1.5 of [Berger 03]).

iv. A $CAT(1)$ space with diameter less than $\pi/2$, which is a C_k space. Hence any domain in it will be C_k as well (see Proposition 3.1 of [Ohta 07]).

A more general classification of manifolds with boundary with the aforesaid property is within the scope of future work.

Formula in terms of a single coordinate chart: As discussed earlier, a single coordinate chart over entire Ω may not exists. However, for many computational purposes, one can use a single coordinate chart, $B = (\Omega - \mathcal{S}, \psi)$, that describes almost the whole of Ω, except for a set of measure zero, $\mathcal{S}$ (*e.g.* the spherical coordinate on a sphere describes everything except one longitudinal line and the poles). Thus, it is worth re-writing equation (7) entirely in terms of a coordinate chart. Besides changing q to $\mathbf{q} = \psi(q)$ and p_k to $\mathbf{p}_k = \psi(p_k)$, we need to compute the volume of the infinitesimal element $\mathrm{d}q$ in the particular coordinate chart. Assuming $\mathrm{d}\mathbf{q}$ to be an infinitesimal element in a particular coordinate chart, this is given by $\mathrm{d}q = \sqrt{\det(\eta_{\bullet\bullet}(\mathbf{q}))}\, \mathrm{d}\mathbf{q}$. Thus we have,

$$u_k^l \;=\; 2\kappa \int_{V_k - (p_k \cup \mathcal{C}_{p_k} \cup \mathcal{S})} d_\ell^B(\mathbf{q}, \mathbf{p}_k) \, \frac{-z^l_{\mathbf{q}\mathbf{p}_k}}{\sqrt{\eta_{mn}(\mathbf{p}_k)\, z^m_{\mathbf{q}\mathbf{p}_k} z^n_{\mathbf{q}\mathbf{p}_k}}} \; w^B(\mathbf{q}) \; \sqrt{\det(\eta_{\bullet\bullet}(\mathbf{q}))}\, \mathrm{d}\mathbf{q} \qquad (8)$$

where d_ℓ^B is the length metric expressed in terms of coordinate chart B, *i.e.*, $d_\ell^B(\mathbf{a}, \mathbf{b}) = d_\ell(\psi^{-1}(\mathbf{a}), \psi^{-1}(\mathbf{b}))$. Likewise, $w^B : \mathbb{R}^N \to \mathbb{R}_+$ such that $w^B(\mathbf{a}) = w(\psi^{-1}(\mathbf{a}))$. $\mathbf{z}_{\mathbf{q}\mathbf{p}_k} = [z^1_{\mathbf{q}\mathbf{p}_k}, z^2_{\mathbf{q}\mathbf{p}_k}, \cdots, z^N_{\mathbf{q}\mathbf{p}_k}]^T$ is the coefficient vector (in chart B) of the tangent vector at q to the minimal path joining p_k and q. We emphasize on the fact that $(p_k \cup \mathcal{C}_{p_k} \cup \mathcal{S})$ is a set of measure zero in Ω, and does not concern us as far as computation is concerned.

The formula in (8), as we will see in the next section, gives an algorithm for approximately computing the gradient of $\tilde{\mathcal{H}}$. This gives a generalized Lloyd's algorithm with guarantee of asymptotic stability (since we showed that the control vector will always exist in the tangent cones at points in Ω, and hence the only way the robots can stop is when the control becomes zero, *i.e.* the gradient of $\tilde{\mathcal{H}}$ vanishes). In the final converged solution, each robot will be at a point in the interior of its respective Voronoi cell where the gradient of the second moment function of the cell vanishes.

4 Graph Search-based Implementation

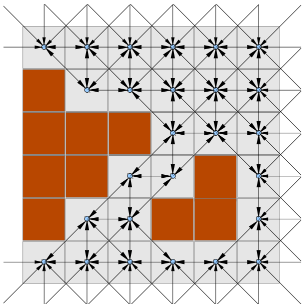


Figure 12: Graph construction: An 8-connected grid graph created from a uniformly discretized coordinate space. The brown cells represent obstacles.

In order to develop a version of the generalized continuous-time Lloyd's algorithm for a general distance function, we first need to be able to compute the general Voronoi tessellation of Equation (2) for the metric, d_ℓ. We adopt a discrete graph-search based approach for achieving that, not unlike the approach taken in previous work [Bhattacharya 10]. While discretization invariably implies certain

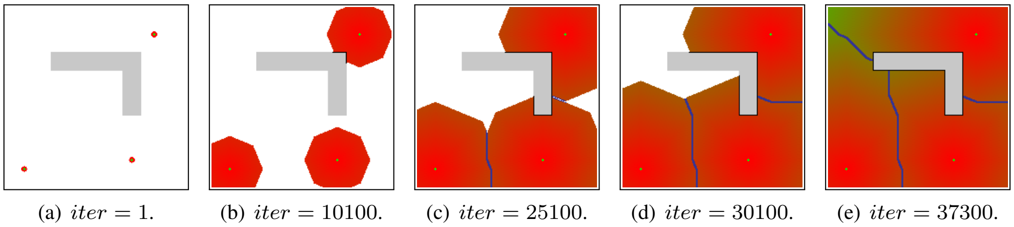

(a) $iter = 1$. (b) $iter = 10100$. (c) $iter = 25100$. (d) $iter = 30100$. (e) $iter = 37300$.

Figure 13: Progress of the algorithm for tessellation and control computation in an environment with a L-shaped obstacle. The graph is constructed by 200×200 uniform square discretization of the environment (see [Bhattacharya 10]). Tessellation is created starting from three points (the location of the agents) to the complete diagram after expansion of about 37300 vertices. The filled area indicates the set of *expanded* vertices. The boundaries of the tessellation are visible in blue.

level of approximation and deviation from the original metric space, in order to make any continuous problem computationally feasible it is an indispensable trade-off.

We consider a uniform square tiling (Figure 12) of the space of coordinate variables (in a particular coordinate chart) and create a graph G out of it (with vertex set $\mathcal{V}(G)$, edge set $\mathcal{E}(G) \subseteq \mathcal{V}(G) \times \mathcal{V}(G)$ and cost function $\mathcal{C}_G : \mathcal{E}(G) \rightarrow \mathbb{R}_+$). The costs/weights of the edges of the graph are the metric lengths of the edges. It is to be noted that in doing so we end up restricting the metric of the original space to the discrete graph. Because of this, as well as due to the discrete computation of the integrations (as discussed later), this discrete graph-search based approach is inherently an approximate method, where we trade off the accuracy and elegance of a continuous space for efficiency and computability with an arbitrary metric.

The key idea is to make a basic modification to Dijkstra's algorithm [Dijkstra 59, Cormen 01] that enables us in creating a geodesic Voronoi tessellation. For creating Voronoi tessellation we initiate the *open set* with multiple start vertices from which we start propagation of the wavefronts. Thus the wavefronts emanate from multiple sources. The places where the wavefronts collide will hence represent the boundaries of the Voronoi tessellation (see [Adamatzky 96, Velic 09, Durham 12] for similar wavefront-based Voronoi partitioning methods). In addition, we can conveniently alter the distance function, the level-set of which represents the boundaries of the Voronoi cells. This enables us to even create *geodesic power Voronoi tessellation*. Figure 13 illustrates the progress of the algorithm in creation of the tessellation.

In order to compute the control command for the robots (*i.e.* the action of the robot in the next time step), we use the formula in Equation (8). In a discretized setup, the position of the k^{th} robot corresponds to a vertex $\mathrm{p}_k \in \mathcal{V}(G)$ (note that we use a vertical, regular-weight font to distinguish a vertex from the point $p_k \in \Omega$ or its coordinate representation $\mathbf{p}_k$). The coefficient vector $\mathbf{z}_{\mathbf{q}\mathbf{p}_k}$ of the tangent vector is approximated by the coefficient vectors along edges of the form $[\mathrm{p}'_k, \mathrm{p}_k] \in \mathcal{E}(G)$ for some $\mathrm{p}'_k \in \mathcal{N}_G(\mathrm{p}_k)$ (the set of neighbors of p_k) such that the shortest path in the graph connecting p_k and $\mathrm{q} \in \mathcal{V}(G)$ passes through p'_k. For a given q, we know the index of the robot, $\tau(\mathrm{q})$, whose tessella it belongs to, and can compute the shortest path in the graph joining the vertices $\mathrm{p}_{\tau(\mathrm{q})}$ and q. The neighbor of $\mathrm{p}_{\tau(\mathrm{q})}$, through which the shortest path passes, is the desired $\mathrm{p}'_{\tau(\mathrm{q})}$, and it is maintained in a variable $\eta(\mathrm{q})$ in an efficient way. Thus, we can also compute the integration of (8) on the fly (by approximating it as a summation over the discrete cells) as we compute the tessellation. The complete pseudo-code of the algorithm is given below.

```
$\{\tau, \{\mathrm{p}'_k\}\}$ = **Tessellation_and_Control_Computation** $(G, \{\mathrm{p}_k\}, \{R_k\}, \overline{w})$
  Inputs:   a. Graph $G$ (with vertex set $\mathcal{V}(G)$, edge set $\mathcal{E}(G) \subseteq \mathcal{V}(G) \times \mathcal{V}(G)$,
                     and cost function $\mathcal{C}_G : \mathcal{E}(G) \to \mathbb{R}_+$)
            b. Agent locations $\mathrm{p}_k \in \mathcal{V}(G),\ k = 1, 2, \cdots, n$
            c. Agent weight $R_k \in \mathbb{R}^+,\ k = 1, 2, \cdots, N$
            d. Discretized weight/density function $\overline{w} : \mathcal{V}(G) \to \mathbb{R}$
            e. The matrix representation of metric tensor, $\eta_{\cdot\cdot} : Img(\psi) \to \mathbb{R}^{D \times D}$, in coordinate chart $B$
  Outputs:  a. The tessellation map $\tau : \mathcal{V}(G) \to \{1, 2, \cdots, n\}$
            b. The next position of each robot, $\mathrm{p}'_k \in \mathcal{N}_G(\mathrm{p}_k),\ k = 1, 2, \cdots, n$
1   Initiate $g$: Set $g(\mathrm{v}) := \infty$, for all $\mathrm{v} \in \mathcal{V}(G)$   // Shortest distances
2   Initiate $\rho$: Set $\rho(\mathrm{v}) := \infty,\ \forall \mathrm{v} \in \mathcal{V}(G)$   // Power distances
3   Initiate $\tau$: Set $\tau(\mathrm{v}) := -1,\ \forall \mathrm{v} \in \mathcal{V}(G)$   // Tessellation
4   Initiate $\nu$: Set $\nu(\mathrm{v}) := \emptyset,\ \ \forall \mathrm{v} \in \mathcal{V}(G)$ // Pointer to robot neighbor. $\nu : \mathcal{V}(G) \to \mathcal{V}(G)$
5   **for each** $(\{k \in \{1, 2, \cdots, n\}\})$
6       Set $g(\mathrm{p}_k) = 0$
7       Set $\rho(\mathrm{p}_k) = -R_k^2$
8       Set $\tau(\mathrm{p}_k) = k$
9       Set $\mathbf{I}_k := \mathbf{0}$   // The negative of differential of $\tilde{\mathcal{H}}$ in coordinate chart $B$ (a covector).
10      **for each** $(\{\mathrm{q} \in \mathcal{N}_G(\mathrm{p}_k)\})$   // For each neighbor of $\mathrm{p}_k$
11          Set $\nu(\mathrm{q}) = \mathrm{q}$
12  Set $Q := \mathcal{V}(G)$   // Set of un-expanded vertices
13  **while** $(Q \neq \emptyset)$
14      $\mathrm{q} := \mathrm{argmin}_{\mathrm{q}' \in Q}\ \ \rho(\mathrm{q}')$   // Maintained by a heap data-structure.
15      **if** $(g(\mathrm{q}) == \infty)$
16          **break**
17      Set $Q = Q - \mathrm{q}$   // Remove q from $Q$
18      Set $l := \tau(\mathrm{q})$
19      Set $\mathrm{s} := \nu(\mathrm{q})$
20      **if** $(\mathrm{s}\ != \emptyset)$   // Equivalently, $\mathrm{q} \notin \{\mathrm{p}_k\}_{k=1,2,\ldots,n}$
21          Set $\mathbf{z} = \mathbf{P}(\mathrm{s}) - \mathbf{P}(\mathrm{p}_l)$   // Negative of tangent vector
21          Set $\mathbf{M} = \eta_{\cdot\cdot}(\mathbf{P}(\mathrm{p}_l))$   // Metric tensor at $p_l$ as a matrix in coordinate representation of $B$.
21          Set $\mathbf{I}_l$ += $g(\mathrm{q}) \times \frac{\mathbf{Mz}}{\sqrt{\mathbf{z}^T\mathbf{Mz}}} \times \overline{w}(\mathrm{q}) \times \sqrt{\det \mathbf{M}}$   // Negative of integral in gradient of $\tilde{\mathcal{H}}$
22      **for each** $(\{\mathrm{w} \in \mathcal{N}_G(\mathrm{q})\})$   // For each neighbor of q
23          Set $g' := g(\mathrm{q}) + \mathcal{C}_G([\mathrm{q}, \mathrm{w}])$
24          Set $\rho' := \mathbf{PowerDist}(g', R_l)$
25          **if** $(\rho' < \rho(\mathrm{w}))$
26              Set $g(\mathrm{w}) = g'$
27              Set $\rho(\mathrm{w}) = \rho'$
28              Set $\tau(\mathrm{w}) = l$
29              **if** $(\mathrm{s}\ != \emptyset)$   // Equivalently, $\mathrm{q} \notin \{\mathrm{p}_k\}_{k=1,2,\cdots,n}$
30                  Set $\nu(\mathrm{w}) = \mathrm{s}$
31  **for each** $(\{k \in \{1, 2, \cdots, n\}\})$
32      Set $\mathrm{p}'_k := \mathrm{argmax}_{\mathrm{u} \in \mathcal{N}_G(\mathrm{p}_k)}\ \ (\mathbf{P}(\mathrm{u}) - \mathbf{P}(\mathrm{p}_k)) \cdot \mathbf{I}_k$   // Choose action best aligned along $\mathbf{I}_k$.
33  **return** $\{\tau, \{\mathrm{p}'_k\}\}$
```

where, we used the coordinate chart $B = (\Omega - \mathcal{S}, \psi)$ as discussed earlier. The function $\mathbf{P} : \mathcal{V}(G) \to \Omega$ is such that $\mathbf{P}(\mathrm{q})$ gives the coordinate of the vertex q. Thus, $\mathbf{P}(\mathrm{p}_k) = \mathbf{p}_k$ and $\mathbf{P}(\mathrm{q}) = \mathbf{q}$ in relation to Equation (8). In an uniform discretization setting we take $\overline{w}(\mathrm{q}) = \alpha\ w(\psi^{-1}(\mathbf{P}(\mathrm{q})))$ for an arbitrary positive constant α representing the 'area' of each discretized cell. Note that $\mathbf{I}_k$ is the coefficient vector of the differential of $\tilde{\mathcal{H}}$ (a covector), while $\mathbf{u}_k$ is its dual (a vector – see Equation (7)). Thus, in line 32 of the algorithm, the 'dot product' is an element-wise product followed by a summation and not the inner product (*i.e.* for $a \in T_p^*\Omega$ and $b, c \in T_p\Omega$, we define $a \cdot b = a_i b^i$, but $<b, c>= \eta_{ij} b^i c^j$). The function $\mathbf{PowerDist}(x, R_k) \equiv f_k(x) = x^2 - R_k^2$ computes the *power distance* for *power Voronoi tessellation*.

Complexity: The complexity of the algorithm is the same as the standard Dijkstra's algorithm, which for a constant degree graph is $O(V_G \log(V_G))$ (where $V_G = |\mathcal{V}(G)|$ is the number of vertices in the graph). This is in sharp contrast to the complexity of search for optimal location as in [Durham 12], where explicit computation of the second moment is performed with respect to every vertex in a tessellation (each of which require a Dijkstra's-like algorithm to start from the chosen vertex and expand every other vertex of the Voronoi cell and compute the distances to those). Although the approach in [Durham 12] finds the global minimum of $\mathcal{M}(V_k)$ at every iteration, the complexity of the algorithm is $O(V_G^2 \log(V_G))$, and becomes prohibitively expensive when the environment is finely discretized.

Distributed and Decentralized Implementation: The algorithm described above lends itself to distributed implementation quite naturally. The distributability of the original Lloyd's algorithm is well-known in literature [Pavone 11, Bullo 09]. The key idea behind a distributed implementation is to have minimal information available and the minimal computation performed on the i^{th} processor for computing the control commands for only the i^{th} robot. Then, multiple processors can compute the control commands for the different robots parallely. Also, as described next, besides the minimal communication of positions, there is no need for a centralized processor to consolidate the results obtained from the parallel computations, thus even allowing decentralization of the control computation.

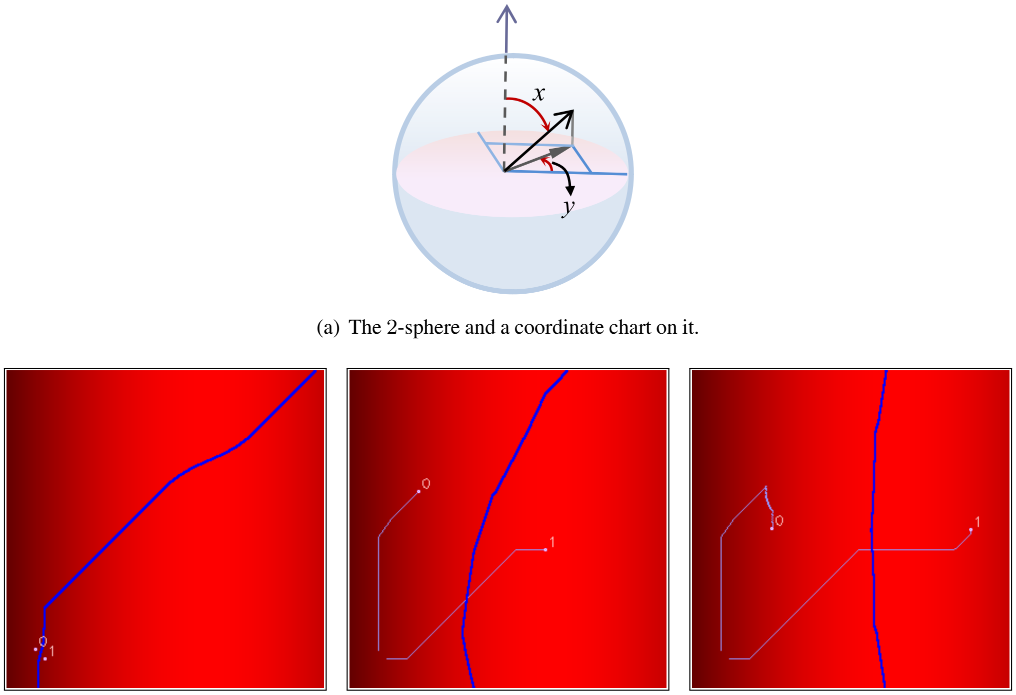
For adapting the algorithm **Tessellation_and_Control_Computation** for a distributed implementation, we note that the control velocity of the i^{th} robot depends only on the shape of its own tessellum, V_i (see (8)). So on the i^{th} processor we need to do enough to compute only the i^{th} tessellum. For this, the i^{th} processor (assume it is stationed on board the i^{th} robot) needs to know at least the position of the *neighbors* of the i^{th} robot – robots with which it shares a tessellum boundary (One can use a heuristic to determine which robots are such *potential boundary-sharing neighbors*. But if that is not possible, considering all robots in the environment as potential neighbors will still allow the decentralized implementation). Say that set of robot indices is $\mathcal{N}_i = \{j_1, j_2, \cdots, j_{n_i}\}$ (n_i being the number of potential boundary-sharing neighbors of the i^{th} robot). Thus the i^{th} robot will need to communicate with these neighbors to obtain their positions (as vertices in the graph G), $\{\mathrm{p}_i, \mathrm{p}_{j_1}, \mathrm{p}_{j_2}, \cdots, \mathrm{p}_{j_{n_i}}\} =: P^i$ (note we included the robot's own position as the first element in the set). Likewise we construct the set of agent weights $R^i = \{\mathrm{R}_i, \mathrm{R}_{j_1}, \mathrm{R}_{j_2}, \cdots, \mathrm{R}_{j_{n_i}}\}$.

Then, all that the i^{th} processor will need to do is run the **Tessellation_and_Control_Computation** algorithm with only the neighbors and itself as inputs. That is, it needs to run

$$\{\tau,\ P'\} \ = \ \textbf{Tessellation\_and\_Control\_Computation}(G,\ P^i,\ R^i,\ \overline{w})$$

with the '**for each**' statements in lines 5 being run over the set of indices $k \in \{i, j_1, j_2, \cdots, j_{n_i}\}$ only, and the control computation in line 31-32 being done only for $k = i$. Moreover, the main loop (line 13–30) can be exited as soon as the i^{th} tessellum has been completely computed (This becomes true when the *open set*, *i.e.*, the set of un-expanded vertices with finite power distance, does not contain a vertex that is also in the tessellum of robot i. That is, when $\{\mathrm{q} \mid \mathrm{q} \in Q, \rho(\mathrm{q}) \neq \infty, \tau(\mathrm{q}) = i\} = \emptyset$).

Thus the first element of the set of controls returned, P', is the required p_i' — the next position for the i^{th} robot in the graph.



(a) The 2-sphere and a coordinate chart on it.

(b) $t = 1$.

(c) $t = 150$.

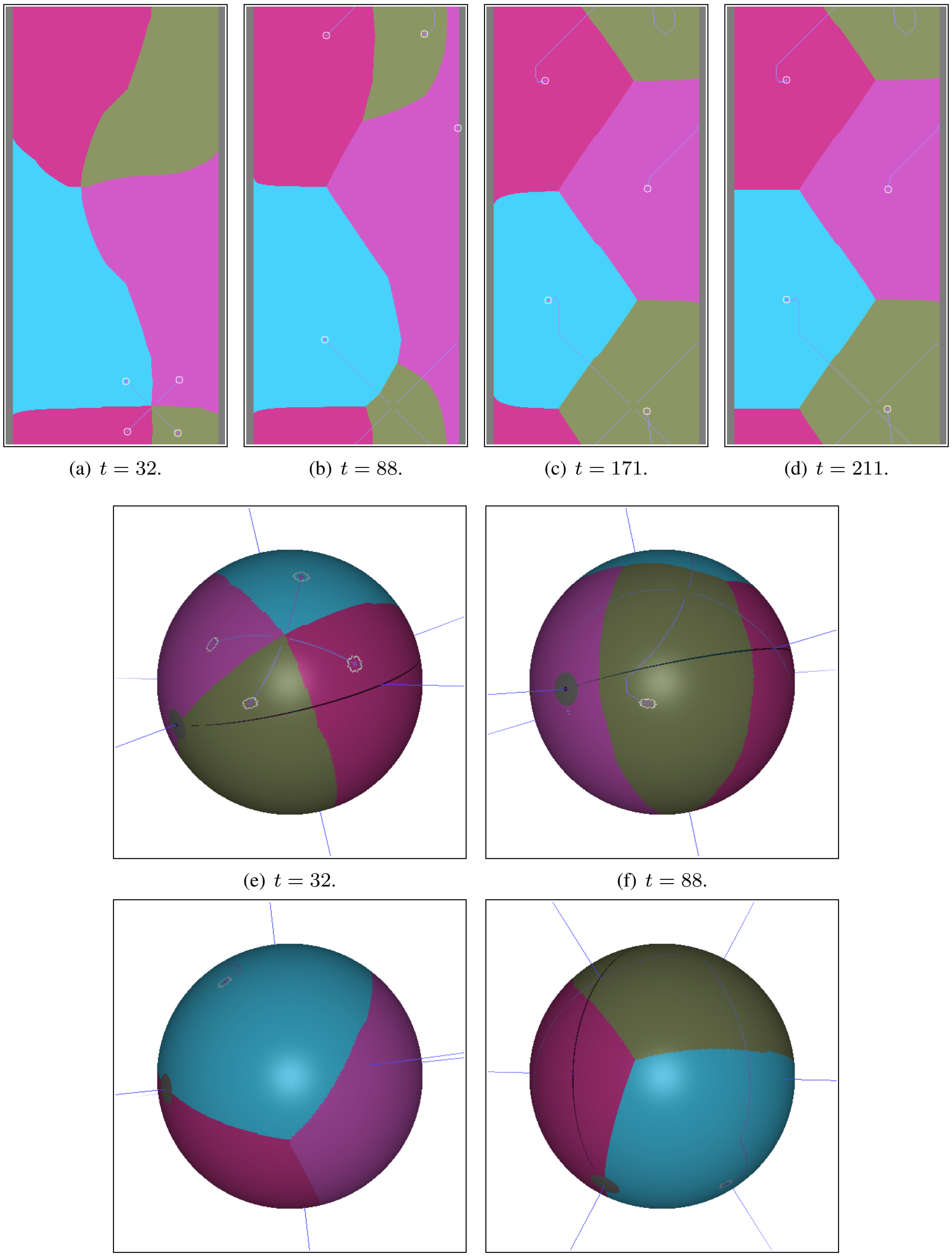
(d) $t = 250$. Converged solution.

Figure 14: Coverage using a discrete implementation of generalized continuous-time Lloyd's algorithm on a part of the 2-sphere. The chosen coordinate variables, x and y, are the latitudinal and longitudinal angles respectively, and the domain shown in figures (b)-(d) represent the region on the sphere where $x \in [\pi/16, 3\pi/4], y \in [\pi/16, 3\pi/4]$. x is plotted along horizontal axis and y along vertical axis on linear scales. The intensity indicates the determinant of the metric, the thick curves are the tessellation boundaries, and the thin curves are the robot trajectories.

4.1 Application to Coverage on Non-Euclidean Metric Spaces

In this section we will illustrate examples of coverage using the generalized continuous-time Lloyd's algorithm on a 2-sphere. We use a coordinate chart with coordinate variables $x \in (0, \pi)$, the latitudinal angle, and $y \in [0, 2\pi)$, the longitudinal angle (Figure 14(a)). The matrix representation of the metric on the sphere using this coordinate chart is $\eta_{\bullet\bullet} = \begin{bmatrix} 1 & 0 \\ 0 & \sin^2(x) \end{bmatrix}$. As usual, we use a uniform square discretization of the coordinate space to create an 8-connected grid graph [Bhattacharya 10]. However, in order to model the complete sphere (in the example of Figure 15), we need to establish appropriate edges between vertices at the extreme values of y, *i.e.* the ones near $y = 0$ and $y = 2\pi$. Similarly, we use an additional vertex for each pole to which the vertices corresponding to the extreme values of x connect.

Figure 14(b)-(d) shows two robots pursuing the coverage control law on a subset of the 2-sphere by following the control command computed using the algorithm of Section 4 at every time-step. The region of the sphere that we restrict to is that of latitudinal angle $x \in [\pi/16, 3\pi/4]$, and longitudinal angle $y \in [\pi/16, 3\pi/4]$. The robots start off from the bottom left of the environment near the

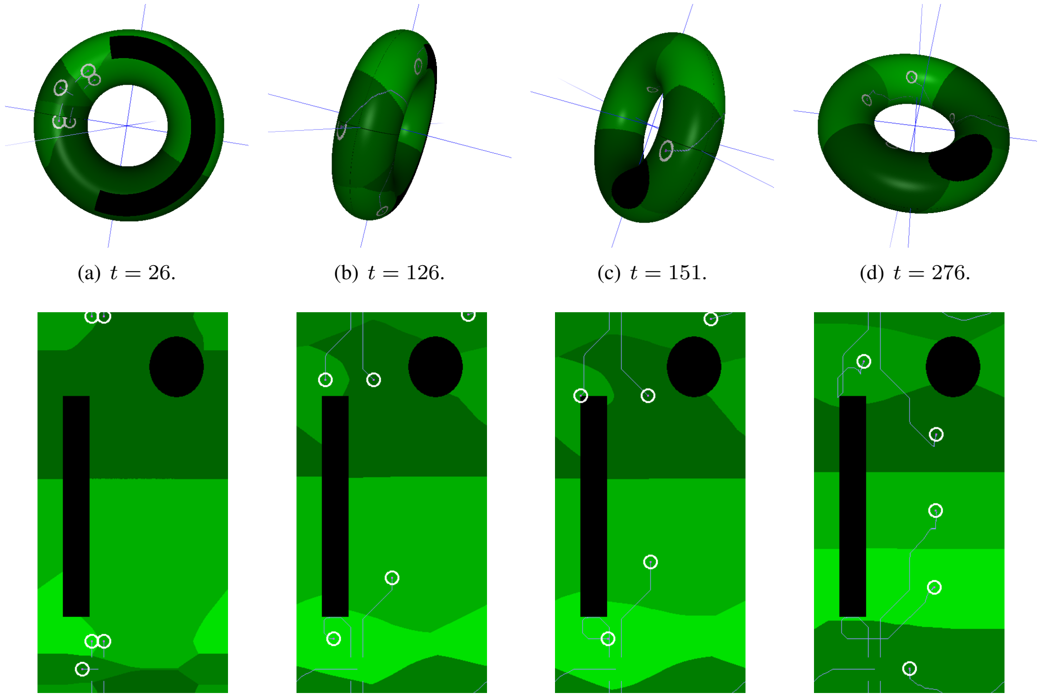


(a) $t = 32$. (b) $t = 88$. (c) $t = 171$. (d) $t = 211$.

(e) $t = 32$. (f) $t = 88$.

(g) $t = 171$. (h) $t = 211$.

Figure 15: Coverage on a complete sphere. In Figures (a)-(d), $x \in (0, \pi)$ (latitudinal angle) is plotted along horizontal axis and $y \in [0, 2\pi)$ (longitudinal angle) along vertical axis on linear scales. Figures (e)-(h) show the same plot mapped on the 2-sphere. The colors are used to indicate the tessella of the robots. Note that in (e)-(h) different viewing angles are used to facilitate visualization. (d) and (h) are the converged solutions.



(a) $t = 26$. (b) $t = 126$. (c) $t = 151$. (d) $t = 276$.

(e) $t = 26$. (f) $t = 126$. (g) $t = 151$. (h) $t = 276$.

Figure 16: Coverage control on a 2-torus with two obstacles on it (marked in black) with 5 robots. (a)-(d): The torus is rotated to view from different angles as the coverage algorithm progresses. (e)-(h): The plot on the coordinate chart.

point $[0.42, 0.45]$, and follow the control law of Equation (8) until convergence is achieved. Note that the tessellation has a curved boundary in Figures 14(c) and 14(d) because it has to be a segment of the great circle on the sphere (note that the jaggedness is due to the fact that the curve actually resides on the discrete graph rather than the original metric space of the sphere). In the converged solution of Figure 14(d), note how the robots get placed such that the tessellation splits up the area on the sphere equally rather than splitting up the area of the non-isometric embedding in $\mathbb{R}^2$ that depends on the chosen coordinate chart. The weight function is chosen to be constant, $w(q) = 1$. Note that in the figures t denote the number of times the control commands were computed (a single increment of t thus means a complete execution of algorithm '*Tessellation_and_Control_Computation*'). For this example, the program ran at a rate of about 4 Hz (*i.e.*, 4 passes of the algorithm per second) on a machine with 2.1 GHz processor and 3 GB memory.

A more complete example is shown in Figure 15 in which 4 robots attain coverage on a complete 2-sphere. The robots start off close to each other on the sphere, and follow the control law of Equation (8), until they converge attaining good and uniform coverage of the sphere. In order to avoid numerical problems near the poles, we cordon off small disks near the poles (marked by the gray regions), and establish 'invisible' edges across those disks connecting the vertices on their diametrically opposite points. Figures 15(a)-(d) show the tessellation in the coordinate chart with x plotted along the horizontal axis and y along the vertical axis (300×600 uniformly discretized).

Figures 15(e)-(h) show the same tessellation mapped on the sphere. The weight function, once again is chosen to be $w(q) = 1$. For this example, the program ran (control computation as well as plotting of the graphics) at a rate of about 1 Hz on a machine with 2.1 GHz processor and 3 Gb memory.

Figure 16 shows a similar example of coverage on a 2-torus with 2 obstacles by 5 robots. The matrix representation of the metric tensor on the torus is given by $\eta_{\bullet\bullet} = \begin{bmatrix} r^2 & 0 \\ 0 & (R + r\cos x)^2 \end{bmatrix}$, where R is the radius of the axial circle, r the radius of the *tube* of the torus, x is the *latitudinal* angle, and y is the *longitudinal* angle. Note how the Voronoi tessella (distinguished by the shades of green) in this case are not simply connected.

4.2 Application to Cooperative Exploration and Coverage Problem

Next we apply the tools developed to the problem of cooperative exploration and simultaneous coverage. In context of an exploration problem, one efficient way of representing uncertainty in the environment (knowledge about occupancy or a lack of it) is to use the *Shannon entropy* [Ihara 93]. Each point in the environment can be assigned a numerical value (the Shannon entropy) – higher the value, more uncertain we are about the accessibility/inaccessibility of the point, while lower value implies greater certainty about occupancy. Thus, in an exploration problem, greater attention needs to be given to regions with high entropy where exploration is yet to be performed. If we are using a Voronoi partitioning and a Lloyd's-like algorithm for deployment of robots and exploration, we thus need to produce an effect such that the unknown/unexplored regions get "stretched out" and get greater attention than the explored regions. This stretching out effect is naturally induced by using the entropy as a weighing factor in the Riemannian metric of the space for computing the tessellation and controls for the robots. The apparatus developed so far thus comes to use.

In previous work [Bhattacharya 10] we had used an approximate and ad hoc "projection of centroid" method in order to compute an analog of *generalized centroid* and device a control law that was essentially to follow the approximate generalized centroid. However such a method has very weak guarantee on convergence and could very well make robots to 'get stuck' at boundaries of obstacles or even oscillate. Equipped with the control law of Equation (8), we now perform the same exploration task in a more systematic way with guarantees on convergence.

The problem involves n mobile robots that are equipped with on-board range sensors and can localize themselves in a global coordinate frame (for example, using on-board GPS devices). Each mobile robot maintains and communicates a probability map for the discretized environment, such that $p(\mathrm{q})$ is the probability that the vertex q is inaccessible (*i.e.* occupied or represents an obstacle) for all $\mathrm{q} \in \mathcal{V}(G)$. Since the initial environment is unknown, the robots start off with

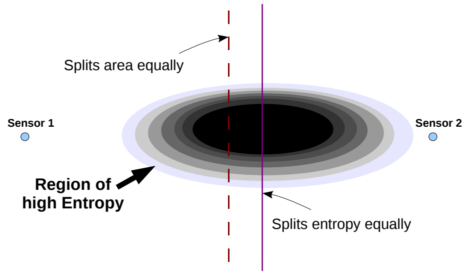


Figure 17: Entropy-weighted metric for voronoi tessellation in exploration problem.

$p(q) = 0.5,\ \forall \mathrm{q} \in \mathcal{V}(G)$. This probability map evolves as the robots obtain sensor data and communicate with each other (see [Bhattacharya 10] for a communication protocol in a decentralized implementation). A threshold on the value of probability determines whether a particular vertex in $\mathcal{V}(G)$ is occupied/inaccessible for computation of the Geodesic Voronoi tessellations as well as control.

As detailed in [Bhattacharya 10], we choose the Shannon entropy for constructing the density function as well as to weigh the metric (Figure 17). Thus, the Shannon entropy is given by $e(\mathrm{q}) = -\left(p(\mathrm{q})\ln(p(\mathrm{q})) + (1-p(\mathrm{q}))\ln(1-p(\mathrm{q}))\right)$. We use this for modeling the discretized version of the weight function, $\overline{w}$, and an isotropic metric, ζI (where I is the identity matrix). Noting that in an exploration problem, the occupancy probability, p, and hence the entropy e, will be functions of time as well, we use the following formulae for $\overline{w}$ and ζ,

$$\overline{w}(\mathrm{q},t) = \left\{ \begin{array}{l} \epsilon_{\overline{w}}, \text{ if } e(\mathrm{q},t) < \tau \\ e(\mathrm{q},t), \text{ otherwise.} \end{array} \right. , \qquad \zeta(\mathrm{q},t) = \left\{ \begin{array}{l} \epsilon_{\zeta}, \text{ if } e(\mathrm{q},t) < \tau \\ e(\mathrm{q},t), \text{ otherwise.} \end{array} \right. \tag{9}$$

for some small $\epsilon_{\overline{w}}$ and ϵ_ζ representing zero (for numerical stability).

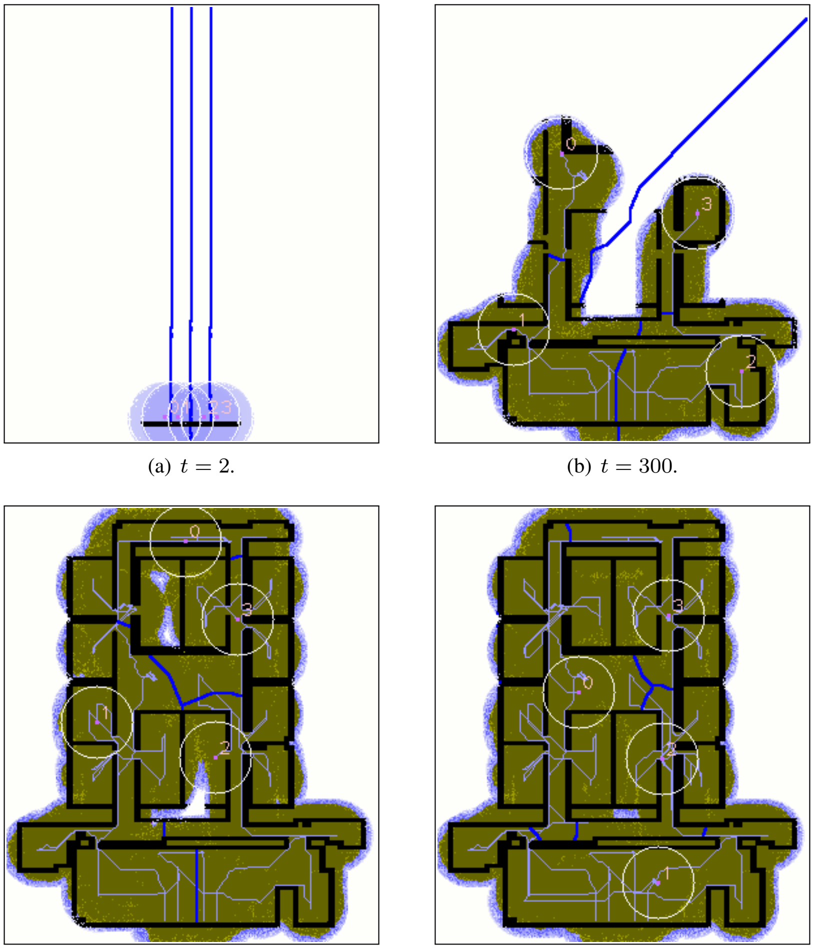
Each mobile robot maintains, updates and communicates a probability map for the discretized environment and updates its entropy map. We use a sensor model similar to that described in [Bhattacharya 10], and 'freeze' a vertex to prevent any change to its probability value when its entropy drops below some $\tau'\ (< \tau)$.

In addition, to avoid situations where a robot gets stuck at a minima inside its tessella even when there are vertices with entropy greater than τ in the tessella (this can happen when there are multiple high entropy regions in the tessella that exert equal and opposite pull on the robot so that the net velocity becomes zero), we perform a check on the value of the integral of the weight function, $\overline{w}$, within the tessella of the k^{th} robot when its control velocity vanishes. If the integral is above the value of $\int_{V_k} \epsilon_{\overline{w}}\, \mathrm{dq}$, we switch to a greedy exploration mode where the k^{th} robot essential head directly towards the closest point that has entropy greater than the value of τ. This ensures exploration of the entire environment (*i.e.* the entropy value for every accessible vertex drops below τ). And once that is achieved, both $\overline{w}$ and ζ become independent of time. Thus convergence is guaranteed.

Figure 18 shows screenshots of a team of 4 robots exploring a part of the 4^{th} floor of the Levine building at the University of Pennsylvania. The intensity of white represents the value of entropy. Thus in Figure 18(a) the robots start with absolutely no knowledge of the environment (maximum entropy), explore the environment, and finally converge to a configuration attaining coverage and minimum entropy (Figure 18(d)).

Figure 19 shows a similar scenario. However, in this case a human operator chooses to take control of one of the robots (Robot 0) soon after they start cooperative exploration of the environment. That robot is forced to stay inside the larger room at the bottom of the environment. Moreover, in this case we use a team of heterogeneous robots (robots with different sensor footprint radii), thus requiring a *power voronoi tessellation*. This simple example illustrates the flexibility of our framework with respect to incorporating human inputs to guide exploration.

For either of the above examples, with the environment was discretized into a 284×333 grid, the program ran at a rate of about $3-4$ Hz on a machine with 2.1 GHz processor and 3 Gb memory.



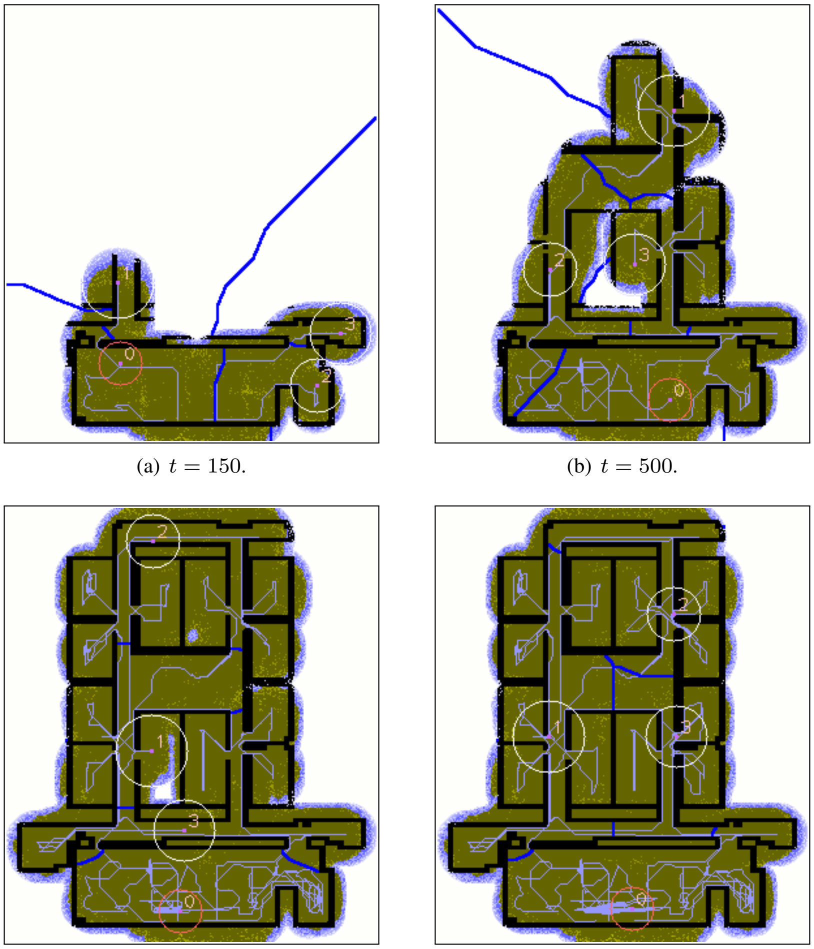
(a) $t = 2$.

(b) $t = 300$.

(c) $t = 900$.

(d) $t = 1550$. Converged solution.

Figure 18: Exploration and coverage of an office environment by a team of 4 robots. Thick curves indicate boundaries of tessellation, intensity of white indicates the value of entropy.



(a) $t = 150$.

(b) $t = 500$.

(c) $t = 1450$.

(d) $t = 2100$. Converged solution.

Figure 19: Exploration and coverage of an office environment discretized into a 284×333 grid by a team of 4 robots, with one of the robots (robot ‘0’) being controlled by a human user.

5 Conclusion

In this paper we have extended the coverage control algorithm proposed by *Cortes, et al.* [Cortes 04], to non-Euclidean configuration spaces that are, in general, non-convex. The key idea is the transformation of the problem of computing gradients of distance functions to one of computing tangents to geodesics. We have shown that this simplification allows us to implement our coverage control algorithm in any space after reducing it to a discrete graph. We have illustrated the algorithm by considering multiple robots achieving uniform coverage on a 2-sphere and an indoor environment with walls and obstacles. We have also shown the application of the basic ideas to the problem of multi-robot cooperative exploration of unknown or partially known environments.

Acknowledgements

We gratefully acknowledge the support of the Office of Naval Research [grant numbers N00014-07-1-0829 and N00014-09-1-1031], the Army Research Laboratory [grant number W911NF-10-2-0016] and the Air Force Office of Scientific Research [grant number FA9550-10-1-0567].

We would like to thank Dr. Luciano Pimenta, Professor of the Electronic Engineering Department, Universidade Federal de Minas Gerais, for his valuable insights and discussions during the course of writing this paper. We also thank the reviewers for their valuable feedback that greatly improved the quality of this paper.

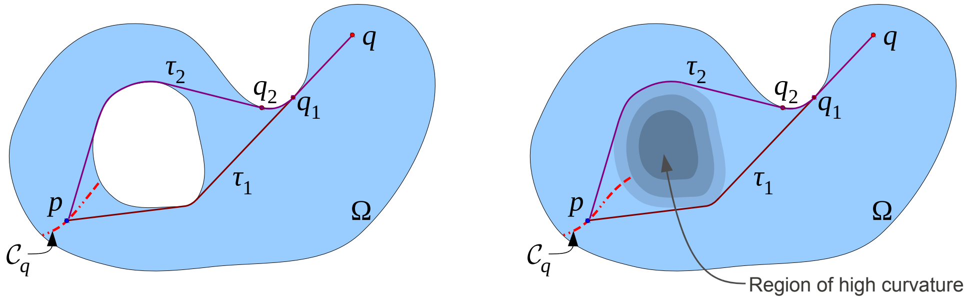
Appendix: Proofs

Sketch of proof for Lemma L1:

> Since Ω is a manifold with C^1 boundary, the 'neighborhood' of every $p \in \Omega$ will either resemble a complete Euclidean space (Figure 4(a)) or an Euclidean half space (Figure 4(b)). More technically, a λ-scaling of Ω at p will converge to $\mathbb{R}^N$ or $\mathbb{H}^N$ as $\lambda \to \infty$, where N is the topological dimension of Ω (see Proposition 3.15 of [Gromov 99]). This indeed is a convex cone in $T_{i(p)}\mathbb{R}^D$ (which, by definition, is λ-scaling of $\mathbb{R}^D$ at $i(p)$). To prove that this space is $T_p\Omega$, we just invoke the local connectedness property of $\mathbb{R}^N$ or $\mathbb{H}^N$, and hence existence of paths emanating from p that 'fill' the space. ■

Proof of Lemma L2:

> An Ω that can be represented as a subset of a smooth, complete manifold of same dimension, along with the described path metric d_ℓ, is sometimes referred to as a *metric space of type* (Δ) (see Theorem 2.1 of [Wolter 85]).
>
> i. The first result follows from the construction of the cut locus (see discussion in p. 42 of [Wolter 85]).
>
> ii. The fact that g_p is C^1 in $\Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$ follows from Theorem 3.1.b of [Wolter 85] (see Figure 5(a)). In fact a stronger condition holds: The gradient of g_p is locally Lipschitz continuous in $\Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$ (by Theorem 5.1 of [Wolter 85]).
>
> iii. If we choose $q, q' \in \Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$ at a distance of ϵ from each other, then due to the triangle inequality for the metric d_ℓ, we have $|g_p(q) - g_p(q')| \leq \epsilon$. Thus it follows that the gradient of g_p is bounded.
>
> ■



(a) Illustration for Lemma L3, adapted from Figure 3.1 of [Wolter 85].

(b) The same phenomenon can be created due to locally non-Euclidean metric on manifold with boundary.

Figure 20: Examples where $q \notin \mathcal{C}_p$, but $p \in \mathcal{C}_q$. This is because the minimal paths, τ_1 and τ_2, have parallel tangents at q.

Sketch of proof for Lemma L3:

> First we note that, unlike the case in complete manifolds (Lemma 2 of [Klingenberg 59]), the statement is not true in general for every p on a manifold with boundary. For an example refer to Figure 3.1 of [Wolter 85] or Figure 20. Thus, the main objective is to prove that the set of points for which the statement is not true is a set of measure zero in Ω. For the purpose of this sketch we assume that the boundary of Ω is smooth. However, this can be relaxed without too much difficulty.
>
> We consider the set of points, $\mathcal{D}_\Omega \subset \Omega$, for which the statement is not true. Let $p \in \mathcal{D}_\Omega$ and $q \notin \mathcal{C}_p$ but $p \in \mathcal{C}_q$. Thus the two minimal paths (say τ_1 and τ_2) connecting p and q should have parallel tangents at q but distinct tangents at p. For that being possible, one of the minimal paths (say τ_1) needs to be tangent at some point $q_1 \in \partial\Omega$ to $\partial\Omega$, where the two distinct tangents emanating from p meet parallely for the first time (and hence continue together up to q). For the two paths to be distinct before q_1, the other minimal path (τ_2) needs to meet the boundary earlier than q_1, and 'graze' the boundary up to q_1 – say it meets at $q_2 \in \partial\Omega$ (see Figure 20). Clearly, the part of τ_2 connecting q_1 and q_2 has to be a geodesic segment on $\partial\Omega$.
>
> Let Ω be of dimension N. A choice of the point p and two directions for the paths emanating from p is equivalent to choosing a point on the double sphere bundle $\mathcal{S}\Omega := \Omega \times \mathbb{S}^{N-1} \times \mathbb{S}^{N-1}$. Let the point p, along with the tangents to τ_1 and τ_2 at p (which we represent as t_1 and t_2 respectively), be represented on $\mathcal{S}\Omega$ by the triple (p, t_1, t_2). Now, we look at the tangent space of $\mathcal{S}\Omega$ at (p, t_1, t_2) and determine the dimension of the subspace along which we can perturb the position of the point p and the directions of the emanating minimal paths (*i.e.* t_1 and t_2) such that the aforesaid conditions are satisfied. We write $T_{(p,t_1,t_2)}\mathcal{S}\Omega$ for that tangent space, which is of dimension $N + 2(N-1) = 3N - 2$, with $T_p\Omega$ regarded as a N-dimensional subspace.
>
> We consider the case where τ_j $(j = 1, 2)$ do not touch $\partial\Omega$ between p and q_j (like the scenario in Figure 20(b), but unlike the scenario in Figure 20(a)). It can be easily verified that the situations in which they do, result in additional constraints that further reduces the number of orthogonal directions in which (p, t_1, t_2) can be perturbed. The case that we consider here can arise, for example, when the central hole/obstacle in Figure 20(a) is removed, but is replaced by a region of high curvature as in Figure 20(b).
>
> First, in order to maintain tangency of τ_1 at q_1 with $\partial\Omega$, the perturbations will need to satisfy 1 constraint. That gives us a $(3N-2) - 1 = 3N - 3$ dimensional subspace of $T_{(p,t_1,t_2)}\mathcal{S}\Omega$ in which we can test our further perturbations. Now consider the geodesic curve on $\partial\Omega$ emanating from q_1 and along the direction of the tangent of τ_1 at q_1. For τ_2 to intersect this curve, we have another $N-2$ constraints (there would be 0 constraints to intersect any arbitrary point on the $N-1$ dimensional $\partial\Omega$. Thus to intersect a 1 dimensional submanifold of it, we need to impose $(N-1) - 1$ constraints). This point of intersection is q_2. Further, in order to ensure that the tangent to τ_2 at q_2 is parallel to the geodesic segment connecting q_1 and

> q_2 at q_2, we will need to impose another $N-1$ constraints (since the space of possible directions in which it is possible to pass through q_2 is $\mathbb{S}^{N-1}$, and we need τ_2 to be tangent to a specific direction at q_2). Thus, so far, we have $1+(N-2)+(N-1) = 2N-2$ constraints, that leaves us with a $(3N-2)-(2N-2) = N$ dimensional subspace of $T_{(p,t_1,t_2)}\mathcal{S}\Omega$ along which we can choose a direction to move (p, t_1, t_2) ans still satisfy the required conditions. Finally, we need to impose the constraint that the lengths of the paths τ_1 and τ_2 need to be equal. This imposes 1 additional constraint. Thus, the dimensionality of the subspace of $T_{(p,t_1,t_2)}\mathcal{S}\Omega$ along which we can perturb the point p and the emanating directions of τ_1 and τ_2 is $N-1$. Even if this subspace lies entirely in $T_p\Omega \subset T_{(p,t_1,t_2)}\mathcal{S}\Omega$, it is one dimension less than the dimensionality of $T_p\Omega$. Thus the dimensionality of $\mathcal{D}_\Omega$ can at most be $N-1$. ■

As discussed earlier, we first prove Lemma L4 and Corollary C1, using which we will prove the Proposition P1.

It is possible to prove Lemma L4 using more direct arguments that establish the direction of steepest descent of a function in the tangent space as the dual of the differential of the function (*i.e.* gradient of a function). However, for completeness, we provide an explicit proof.

Proof of Lemma L4:

> For notational convenience, let us define $g_{\mathbf{w}}(\mathbf{u}) := d^C(\mathbf{w}, \mathbf{u}),\ \forall \mathbf{u} \in \mathit{Img}(\phi)$. Recall that geodesically convex implies that any two points in U can be connected using a smooth geodesic segment (a curve satisfying the geodesic equation at every point) lying entirely in U.
>
> Consider $g_{\mathbf{w}}$ as a function from $\mathbb{R}^N$ to $\mathbb{R}_+$ with an unique minima at $\mathbf{w}$. Let $\gamma_{\mathbf{wq}}$ represents any arbitrary curve in $\mathit{Img}(\phi) \subseteq \mathbb{R}^N$ connecting $\mathbf{w}$ to $\mathbf{q}$. By the *fundamental theorem of calculus* and using the fact that $g_{\mathbf{w}}(\mathbf{w}) = 0$, we have,
>
> $$I(\gamma_{\mathbf{wq}}) \ := \ g_{\mathbf{w}}(\mathbf{q}) \ = \ \int_{\gamma_{\mathbf{wq}}} \frac{\partial}{\partial \mathbf{u}} g_{\mathbf{w}}(\mathbf{u}) \cdot \mathrm{d}\mathbf{u} \ \equiv \ \int_{\gamma_{\mathbf{wq}}} \frac{\partial g_{\mathbf{w}}(\mathbf{u})}{\partial u^i}\, \mathrm{d}u^i \tag{10}$$
>
> where, $[\,\mathrm{d}u^1, \mathrm{d}u^2, \cdots, \mathrm{d}u^N]$ is the coefficient vector (in chart C) of an infinitesimal element along the tangent to the curve.
>
> Now, the length of the curve $\gamma_{\mathbf{wq}}$ is given by
>
> $$L(\gamma_{\mathbf{wq}}) \ := \ \int_{\gamma_{\mathbf{wq}}} \sqrt{\eta_{ij}(\mathbf{u})\ \mathrm{d}u^i\, \mathrm{d}u^j} \tag{11}$$
>
> By definition, the value of $L(\gamma_{\mathbf{wq}})$ is minimum when $\gamma_{\mathbf{wq}}$ is the shortest geodesic (which, due to the hypothesis that the cut locus of any point in U is empty, is unique – call it $\gamma^*_{\mathbf{wq}}$) connecting $\mathbf{w}$ and $\mathbf{q}$, and the minimum value is clearly $g_{\mathbf{w}}(\mathbf{q})$ (by definition of $g_{\mathbf{w}}$). Thus,
>
> $$\begin{array}{c} L(\gamma_{\mathbf{wq}}) \ \geq \ I(\gamma_{\mathbf{wq}}) \ \ [= g_{\mathbf{w}}(\mathbf{q}),\ \text{a const. independent of } \gamma_{\mathbf{wq}}]\ , \\ \text{equality holds when } \gamma_{\mathbf{wq}} = \gamma^*_{\mathbf{wq}} \end{array} \tag{12}$$
>
> Now, consider a family of infinitesimal elements of $\mathbb{R}^N$ represented by the coefficient vector $\mathrm{d}\mathbf{u} = [\,\mathrm{d}u^1, \mathrm{d}u^2, \cdots, \mathrm{d}u^N]^T$ located at an arbitrary point $\mathbf{u} \in \mathit{Img}(\phi)$ such that $\mathbf{u} + \mathrm{d}\mathbf{u}$ lies inside $\mathit{Img}(\phi)$. From the triangle inequality of d^C (since it is induced by a Riemannian metric) we have,
>
> $$\begin{aligned} & d^C(\mathbf{w}, \mathbf{u}+\mathrm{d}\mathbf{u}) \ \leq \ d^C(\mathbf{w}, \mathbf{u}) + d^C(\mathbf{u}, \mathbf{u}+\mathrm{d}\mathbf{u}) \\ \Longrightarrow \ & d^C(\mathbf{w}, \mathbf{u}+\mathrm{d}\mathbf{u}) - d^C(\mathbf{w}, \mathbf{u}) \ \leq \ d^C(\mathbf{u}, \mathbf{u}+\mathrm{d}\mathbf{u}) \\ \Longrightarrow \ & \left.\frac{\partial}{\partial \mathbf{u}} g_{\mathbf{w}}(\mathbf{u})\right|_{\mathbf{u}} \cdot \mathrm{d}\mathbf{u} \ \leq \ \sqrt{\eta_{ij}(\mathbf{u})\ \mathrm{d}u^i\, \mathrm{d}u^j} \qquad (13) \\ \Longrightarrow \ & \frac{\partial g_{\mathbf{w}}(\mathbf{u})}{\partial u_i}\, \mathrm{d}u^i \ \leq \ \sqrt{\eta_{ij}(\mathbf{u})\ \mathrm{d}u^i\, \mathrm{d}u^j} \ = \ \frac{\eta_{ij}(\mathbf{u})\, \mathrm{d}u^i\, \mathrm{d}u^j}{\sqrt{\eta_{mn}(\mathbf{u})\ \mathrm{d}u^m\, \mathrm{d}u^n}} \end{aligned}$$

Equality of the triangle inequality in (13) of course holds when $\mathbf{u}$ lies on the geodesic connecting $\mathbf{w}$ and $\mathbf{u} + \mathrm{d}\mathbf{u}$.

Now consider a curve $\gamma'_{\mathbf{wq}}$ (connecting $\mathbf{w}$ and $\mathbf{q}$) with infinitesimal elements $\mathrm{d}\mathbf{u}$ along the tangents to the curve. If there exists a set of points of non-zero measure (1-dimensional measure), along that curve on which the inequality in (13) is not an equality, then the integrals of the quantities on the two sides of the inequality will not be equal. That is, for such a curve we will have $I(\gamma'_{\mathbf{wq}}) < L(\gamma'_{\mathbf{wq}})$.

But we know that there does exist a curve, $\gamma^*_{\mathbf{wq}}$, such that $I(\gamma^*_{\mathbf{wq}}) = L(\gamma^*_{\mathbf{wq}})$ does hold. Thus, for that curve it should be true that the equality of (13) holds true everywhere except possibly a set of measure zero. However, since the curves under consideration are continuous, this is possible only when the curves are the same. Thus we have essentially shown that for the geodesic, $\gamma^*_{\mathbf{wq}}$, at each and every point of the curve the following holds

$$\frac{\partial g_{\mathbf{w}}(\mathbf{u})}{\partial u_i}\,\mathrm{d}u^i \;=\; \frac{\eta_{ij}(\mathbf{u})\,\mathrm{d}u^i\,\mathrm{d}u^j}{\sqrt{\eta_{mn}(\mathbf{u})\;\mathrm{d}u^m\,\mathrm{d}u^n}}$$

where $\mathrm{d}\mathbf{u}$ are of course infinitesimal elements at $\mathbf{u}$ along the tangent to $\gamma^*_{\mathbf{wq}}$.

One can normalize by dividing by $\|\,\mathrm{d}\mathbf{u}\|_2$ to obtain

$$\frac{\partial g_{\mathbf{w}}(\mathbf{u})}{\partial u_i} z^i_{\mathbf{qu}} \;=\; \frac{\eta_{ij}(\mathbf{u}) z^i_{\mathbf{qu}} z^j_{\mathbf{qu}}}{\sqrt{\eta_{mn}(\mathbf{u})\; z^m_{\mathbf{qu}} z^n_{\mathbf{qu}}}} \tag{14}$$

where $z^i_{\mathbf{qu}}$ is the i^{th} component of the tangent vector at $\mathbf{u}$ to the geodesic connecting $\mathbf{w}$ to $\mathbf{u}$, which due to our assumption is unique. We note that the right-hand-side of the above equation represents a scalar field (call it S). Also, $z^i_{\mathbf{qu}}$ (which are functions of $\mathbf{u}$) represent the coefficients of a contravariant vector field in $(U - \mathbf{w})$. Thus, writing X_i for $\frac{\partial g_{\mathbf{w}}(\mathbf{u})}{\partial u_i}$, one can rewrite Equation (14) as

$$X_i z^i_{\mathbf{qu}} = S(\mathbf{u}) \tag{15}$$

where we need to solve for the coefficients $X_i(\mathbf{u}) := \frac{\partial g(\mathbf{u})}{\partial u_i}$. Of course a particular solution is

$$X_{0,i}(\mathbf{u}) = \frac{\eta_{ij}(\mathbf{u}) z^j_{\mathbf{qu}}}{\sqrt{\eta_{mn}(\mathbf{u})\; z^m_{\mathbf{qu}} z^n_{\mathbf{qu}}}} \tag{16}$$

These coefficients clearly transform as coefficients of a covariant vector field. Moreover, the contravariant vector field corresponding to this covariant field (*i.e.* the vectors with coefficients $X^{0,i} = \eta^{ij} X_{0,j}$) is parallel to $z^i_{\mathbf{qu}}$. From this we can infer that the solution mentioned in (16) is the only solution of (15) that transforms as coefficients of a covariant vector field (This is because of the following: Every covariant transformation of X_i corresponds to an unique contravariant transformation of X^i. Again, the general solutions of X^i need to be such that $\eta_{ij} z^i_{\mathbf{qu}} X^j = \eta_{ij} z^i_{\mathbf{qu}} X^{0,j} = \beta \eta_{ij} z^i_{\mathbf{qu}} z^j_{\mathbf{qu}}$ for some scalar field β. This needs to be true in every coordinate chart. Due to positive definiteness of $\eta_{\bullet\bullet}$, this is possible only with X^j parallel to $z^j_{\mathbf{qu}}$, which also fixes the scalar multiple β since we have the known scalar field, S.).

Thus we have

$$\frac{\partial g_{\mathbf{w}}(\mathbf{u})}{\partial u_i} = \frac{\eta_{ij}(\mathbf{u}) z^j_{\mathbf{qu}}}{\sqrt{\eta_{mn}(\mathbf{u})\; z^m_{\mathbf{qu}} z^n_{\mathbf{qu}}}} \tag{17}$$

Thus, by specializing for $\mathbf{u} = \mathbf{q}$, we obtain the proposed result. ■

Proof of Corollary C1:

We first note that due to Lemma L2, we can always choose an open neighborhood of q, $\mathcal{B}_q \subseteq V$, which is geodesically convex, on which the function $g_p := d_\ell(p, \cdot)$ is of class C^1, and the cut locus of every point in which is empty in $\mathcal{B}_q$ (since $q \notin \mathcal{C}_q$).

Consider a minimal path γ^*_{pq} connecting p and q. Let $w (\neq q)$ be a point on this path (between p and q) that lies inside $\mathcal{B}_q$ (which we can always find since $\mathcal{B}_q$ is open) – see Figure 8(a).

From the definition of minimal path we have $\gamma^*_{pq} = \gamma^*_{pw} \cup \gamma^*_{wq}$, for a minimal path, γ^*_{pw}, connecting p and w, and the minimal geodesic, γ^*_{wq}, connecting w and q (which is unique since $q \notin \mathcal{C}_p$). Thus it follows that,

$$\mathbf{z}_{p\mathbf{q}} = \mathbf{z}_{w\mathbf{q}} \tag{18}$$

Again, by triangle inequality, for any $\mathbf{u} \in \psi(\mathcal{B}_q)$

$$\begin{aligned} & d_\ell^{\tilde{D}}(p, \mathbf{u}) \ \leq \ d_\ell(p, w) + d_\ell^{\tilde{D}}(w, \mathbf{u}) \\ \Rightarrow \quad & g_p(\mathbf{u}) \ \leq \ h(\mathbf{u}) \end{aligned} \tag{19}$$

where $h(\mathbf{u}) := d_\ell(p, w) + d_\ell^{\tilde{D}}(w, \mathbf{u})$ and $g_p(\mathbf{u}) := d_\ell^{\tilde{D}}(p, \mathbf{u})$.
However, equality does hold when p, w and $\psi^{-1}(\mathbf{u})$ lie on the same shortest path. This, in particular, is true when $\mathbf{u} = \mathbf{q}$ (due to our choice of w). Now, since $q \notin \mathcal{C}_p$ and by our choice of $\mathcal{B}_q$, both g_p and h are of class C^1 at $\mathbf{q}$. Thus we have $g_p(\mathbf{u}) \leq h(\mathbf{u})$ for $\mathbf{u} \in \psi(\mathcal{B}_\mathbf{q})$, and at $\mathbf{u} = \mathbf{q}$ they satisfy equality and are differentiable. This implies the differentials of the functions at $\mathbf{q}$ should be same,

$$\begin{aligned} & \frac{\partial}{\partial \mathbf{u}} g_p(\mathbf{u}) \Big|_{\mathbf{u}=\mathbf{q}} \quad = \quad \frac{\partial}{\partial \mathbf{u}} h(\mathbf{u}) \Big|_{\mathbf{u}=\mathbf{q}} \\ \Rightarrow \quad & \frac{\partial}{\partial \mathbf{u}} d_\ell^{\tilde{D}}(p, \mathbf{u}) \Big|_{\mathbf{u}=\mathbf{q}} \quad = \quad \frac{\partial}{\partial \mathbf{u}} d_\ell^{\tilde{D}}(w, \mathbf{u}) \Big|_{\mathbf{u}=\mathbf{q}} \end{aligned} \tag{20}$$

Now, $\mathcal{B}_q$ satisfies the conditions for U in Lemma L4, and w and q are points inside it. Thus by Lemma L4,

$$\frac{\partial}{\partial u^i} d_\ell^{\tilde{D}}(w, \mathbf{u}) \Big|_{\mathbf{u}=\mathbf{q}} \quad \equiv \quad \frac{\partial}{\partial u^i} d_\ell^{D}(\mathbf{w}, \mathbf{u}) \Big|_{\mathbf{u}=\mathbf{q}} \quad = \quad \frac{\eta_{ij}(\mathbf{q}) \ z^j_{w\mathbf{q}}}{\sqrt{\eta_{mn}(\mathbf{q}) \ z^m_{w\mathbf{q}} z^n_{w\mathbf{q}}}} \tag{21}$$

where $\mathbf{w} = \psi(w)$.
Substituting from (18) and (20) into (21) we obtain the proposed result. ■

Proof of Proposition P1:

Equipped with Corollary C1 we can now conclude the proof of Proposition P1. The gradient of a function is, by definition, the dual of the differential, *i.e.* $\text{grad}(f) = \left(\frac{\partial f}{\partial x^i} \, \mathrm{d}x^i\right)^* = \eta^{ij} \frac{\partial f}{\partial x^j} \frac{\partial}{\partial x^i}$. Let q be a point on $\partial\Omega$ and $\{q_i\}$ be a Cauchy sequence of points converging to q. Due to Corollary C1, at every point $q_i \in \Omega - (\partial\Omega \cup p \cup \mathcal{C}_p)$, the negative of the gradient of the distance from p is parallel to $-z_{pq_i}$, the tangent at q_i to the minimal path connecting q_i to p (Figure 6). Due to our definition of gradient at boundary points, the gradient of the distance function at q will then be $\lim_{i \to \infty} -z_{pq_i}$. Since d_ℓ is a path metric, the minimal path connecting each q_i to p exists in Ω, as does the path connecting q to p. Thus, the limit exists in $T_q\Omega$. ■

Proof of Lemma L5:

We note that $\mathcal{M} : \Lambda \to \mathbb{R}$ is a continuous function (since d_ℓ is continuous, being a metric on a topological space). In fact, upon fixing p, the only points where $g_p(\cdot) := d_\ell(p, \cdot)$ is not differentiable, due to Proposition P1, is $p \cup \mathcal{C}_p$. This, due to Lemma L2.i., is a set of measure zero in Λ, and due to Lemma L2.iii., the value of the gradient of g_p at these points is bounded. Thus, we can re-write the 2^{nd} moment about p as $\mathcal{M}(p) = \int_{\Lambda - (p \cup \mathcal{C}_p)} f\left(d_\ell(p, q)\right) \ w(q) \ \ \mathrm{d}q$ (since the integral of a bounded function over $p \cup \mathcal{C}_p$, a set of measure zero, is zero). Since the points, q, over which the integral is evaluated are such that $q \notin \mathcal{C}_p$, due to Lemma L3 we have $p \notin \mathcal{C}_q$ for almost every $p \in \Lambda$. Thus $\mathcal{M}$ is a C^1 function almost everywhere in Λ (except for possibly a set of measure zero, $\mathcal{D}_\Lambda$ – see notation introduced in the proof of

Lemma L3). That is, it is *arbitrarily close* [Milnor 63] to a C^1 function on Λ. Let such a C^1 function be $\overline{\mathcal{M}}$. The fact that $\mathcal{M}$ will itself be C^1 when Λ does not contain any cut loci is clear from the preceding discussion. ■

Proof of Lemma L6:

As before, we use $\mathcal{M}$ to refer to the *smoothed* version of the second moment function (which is arbitrarily close to a C^1 function).

The fact that a point $p \in \partial\Lambda$ cannot be a local minimum is easily seen by explicitly writing down the negative of the gradient of $\mathcal{M}$ at p:

$$- \nabla\mathcal{M}|_p \quad = \quad \int_{\Lambda - (p \cup \mathcal{C}_p)} 2\, d_\ell(p,q)\, (-\hat{z}_{qp}) \;\; w(q) \;\; \mathrm{d}q \tag{22}$$

where, $-\hat{z}_{qp} \in T_p\Lambda$ is the negative of the normalized tangent at p to the geodesic connecting q to p, and due to Proposition P1 it exists in the tangent cone $T_p\Lambda$ (in explicit coordinate representation, due to Lemma L4, the l^{th} component of this vector is $\hat{z}^l_{qp} = \frac{z^l_{qp}}{\sqrt{\eta_{mn}(p)\; z^m_{qp} z^n_{qp}}}$, where z_{qp} is an un-normalized tangent). Thus the integral in the expression for $-\nabla\mathcal{M}|_p$ is a sum of vectors in $T_p\Lambda$, a convex cone (due to Lemma L1), with positive coefficients. Thus $-\nabla\mathcal{M}|_p$ exists in the tangent cone at p, $T_p\Lambda$, and is not zero. Thus, from p, following the integral curve of $-\nabla\mathcal{M}$, we can reach a point $p' \in (\Lambda - \partial\Lambda)$ in the neighborhood of p such that $\mathcal{M}(p') < \mathcal{M}(p)$. ■

Proof of Proposition P2:

Recall from (7) (and using (22)) that the control law for the k^{th} robot (using coordinates C_k) is

$$\begin{aligned} u^l_k \quad &= \quad 2\,\kappa\, \textstyle\int_{V_k - (p_k \cup \mathcal{C}_{p_k})} d_\ell(q, p_k)\, \frac{-z^l_{q\mathbf{p}_k}}{\sqrt{\eta_{mn}(\mathbf{p}_k)\; z^m_{q\mathbf{p}_k}\, z^n_{q\mathbf{p}_k}}} \;\; w(q) \;\; \mathrm{d}q \\ &= \quad -\,\kappa \frac{\mathcal{M}^k(\mathbf{p}_k)}{p^l_k} \end{aligned}$$

where $\mathcal{M}^k$ is the second moment function on V_k in terms of the coordinates of C_k.

Due to Lemma L6 $\mathcal{M}^k$ does not have local minimum on ∂C_k. Thus it can converge only to points in the interior of C_k, where, due to the fact that $\mathcal{M}^k$ is arbitrarily close to a C^1 function (Lemma L5), the control velocities will be zero. The fact that the convergence is asymptotic (and there is no oscillation/orbits, for example) follows simply by considering $\widetilde{\mathcal{H}}$ in (7) a Lyapunov function, the value of which decreases steadily with time (by construction). ■

References

[Adamatzky 96] A.I. Adamatzky. *Voronoi-like partition of lattice in cellular automata*. Mathematical and Computer Modelling, vol. 23, no. 4, pages 51 – 66, 1996.

[Alexander 81] S. Alexander. *Distance geometry in Riemannian manifolds-with-boundary*. vol. 838, pages 12–18, 1981.

[Berger 03] M. Berger. A panoramic view of riemannian geometry. Springer-Verlag GmbH, 2003.

[Bhattacharya 10] S. Bhattacharya, N. Michael & V. Kumar. *Distributed coverage and exploration in unknown nonconvex environments*. In Proceedings of the 10th International Symposium on Distributed Autonomous Robotics Systems, pages 1–14. Springer, 2010.

[Bhattacharya 12] S. Bhattacharya, R. Ghrist & V. Kumar. *Multi-Robot Coverage and Exploration in Non-Euclidean Metric Spaces*. In Proceedings of The Tenth International Workshop on theAlgorithmic Foundations of Robotics, 13-15 June 2012.

[Brown 79] K. Q. Brown. *Voronoi diagrams from convex hulls*. Information Processing Letters, vol. 9, no. 5, pages 223 – 228, 1979.

[Bullo 09] F. Bullo, J. Cortés & S. Martínez. Distributed control of robotic networks: A mathematical approach to motion coordination algorithms. Applied Mathematics Series. Princeton University Press, 2009.

[Corcuera 98] J. M. Corcuera, W. S. Kendall & J m. Corcuera. *Riemannian Barycentres and Geodesic Convexity*. Math. Proc. Cambridge Philos. Soc, vol. 127, 1998.

[Cormen 01] T. H. Cormen, C. Stein, R. L. Rivest & C. E. Leiserson. Introduction to algorithms. McGraw-Hill Higher Education, 2001.

[Cortes 04] J. Cortes, S. Martinez, T. Karatas & F. Bullo. *Coverage control for mobile sensing networks*. IEEE Trans. Robot. Autom., vol. 20, no. 2, pages 243–255, April 2004.

[Cortez 05] J. Cortez, S. Martinez & F. Bullo. *Spatially-distributed coverage optimization and control with limited-range interactions*. ESIAM: Control, Optimisation and Calculus of Variations, vol. 11, pages 691–719, 2005.

[Cundy 89] H. Cundy & A. Rollett. Mathematical models. Tarquin Pub., 3rd edition, 1989.

[Dijkstra 59] Edsger W. Dijkstra. *A note on two problems in connexion with graphs*. Numerische Mathematik, vol. 1, pages 269–271, 1959.

[Durham 12] J.W. Durham, R. Carli, P. Frasca & F. Bullo. *Discrete Partitioning and Coverage Control for Gossiping Robots*. IEEE Transactions on Robotics, vol. 28, no. 2, pages 364–378, 2012.

[Gromov 99] M. Gromov, J. Lafontaine & P. Pansu. Metric structures for riemannian and non-riemannian spaces. Progress in mathematics. Birkhäuser, 1999.

[Hormander 90] Lars Hormander. The Analysis of Linear Partial Differential Operators I: Distribution Theory and Fourier Analysis. Classics in Mathematics. Springer, 2nd edition, 1990.

[Ihara 93] S. Ihara. Information theory for continuous systems. Series on Probability and Statistics. World Scientific, 1993.

[Jost 97] J. Jost. Compact riemann surfaces. Springer-Verlag, 1997.

[Klingenberg 59] W. Klingenberg. *Contributions to Riemannian Geometry in the Large*. Annals of Mathematics, vol. 69, no. 3, pages 654–666, May 1959.

[LaValle 06] S. M. LaValle. Planning algorithms. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.

[Lloyd 82] S. P. Lloyd. *Least squares quantization in PCM*. IEEE Trans. Inf. Theory, vol. 28, pages 129–137, 1982.

[Lozano-Pérez 79] T. Lozano-Pérez & M. A. Wesley. *An algorithm for planning collision-free paths among polyhedral obstacles*. Commun. ACM, vol. 22, no. 10, pages 560–570, October 1979.

[Milnor 63] J.W. Milnor. Morse theory. Annals of Mathematics Studies : Vol 51. University Press, 1963.

[Munkres 99] J. Munkres. Topology. Prentice Hall, 1999.

[Murray 94] R.M. Murray, Z. Li, S.S. Sastry & S.S. Sastry. A mathematical introduction to robotic manipulation. CRC PressINC, 1994.

[Ohta 07] S. Ohta. *Convexities of metric spaces*. Geometriae Dedicata, vol. 125, no. 1, pages 225–250, March 2007.

[Palfia 11] M. Palfia. *Means in metric spaces and the center of mass*. Journal of Mathematical Analysis and Applications, vol. 381, no. 1, pages 383–391, September 2011.

[Pavone 11] M. Pavone, A. Arsie, E. Frazzoli & F. Bullo. *Distributed Algorithms for Environment Partitioning in Mobile Robotic Networks*. Automatic Control, IEEE Transactions on, vol. 56, no. 8, pages 1834–1848, 2011.

[Petersen 06] P. Petersen. Riemannian geometry. Graduate Texts in Mathematics. Springer, 2006.

[Pimenta 08] L. C. A. Pimenta, V. Kumar, R. C. Mesquita & G. A. S. Pereira. *Sensing and Coverage for a Network of Heterogeneous Robots*. In Proc. of the IEEE Conf. on Decision and Control, pages 3947–3952, Cancun, Mexico, December 2008.

[Rimon 91] E. Rimon & D.E. Koditschek. *The construction of analytic diffeomorphisms for exact robot navigation on star worlds*. Trans. of the American Mathematical Society, vol. 327, no. 1, Sept. 1991.

[Rimon 92] E. Rimon & D.E. Koditschek. *Exact Robot Navigation Using Artificial Potential Fields*. IEEE Transactions on Robotics and Automation, vol. 8, no. 5, pages 501–518, 1992.

[Sastry 99] S. Sastry. Nonlinear systems: Analysis, stability and control. Interdisciplinary applied mathematics: Systems and control. Springer, 1999.

[Stachniss 05] C. Stachniss, G. Grisetti & W. Burgard. *Information Gain-based Exploration Using Rao-Blackwellized Particle Filters*. In Proc. of Robot.: Sci. and Syst., pages 65–72, Cambridge, MA, June 2005.

[Stachniss 06] C. Stachniss. *Exploration and Mapping with Mobile Robots*. PhD thesis, University of Freiburg, Freiburg, Germany, April 2006.

[Thrun 05] S. Thrun, W. B. & D. Fox. Probabilistic robotics (intelligent robotics and autonomous agents). The MIT Press, 2005.

[Velic 09] Mirko Velic, Dave May & Louis Moresi. *A Fast Robust Algorithm for Computing Discrete Voronoi Diagrams*. Journal of Mathematical Modelling and Algorithms, vol. 8, no. 3, pages 343–355, 2009.

[Wolter 85] F. E. Wolter. *Cut loci in bordered and unbordered Riemannian manfolds*. PhD thesis, Department of Mathematics, Technical University of Berlin, 1985.