Distributed-Infrastructure Generation of Hybrid Homological Patrol Strategies

Peter Kingston and Magnus Egerstedt

Abstract We develop distributed algorithms that mobile agents, by coordinating with static infrastructure like wireless base stations or air traffic control towers, use to execute patrol behaviors reflecting topological properties of their environment. Infrastructure nodes communicate locally with one another using linear dynamics, to asymptotically synthesize hybrid controllers that are supplied to mobile agents, and that correspond to navigating loops in the environment.

1 Introduction

When multi-robot systems are deployed in developed areas, static infrastructure – like wireless routers, cell phone base stations, or air traffic control towers – will typically be used for communication between the agents. In this paper, we describe methods by which the existence of infrastructure can also be exploited to create motion coordination algorithms. The idea, illustrated by Figure 1, is that static infrastructure nodes are connected in a communication network and that, by talking only to their neighbors, these nodes can synthesize local controllers for the mobile agents that, when combined, satisfy desirable global properties.

In particular, we consider a scenario in which mobile agents are intended to patrol an environment. The controllers synthesized by the static infrastructure will guide the mobile agents to loop around various obstacles in the environment, while, simultaneously avoiding concentrations in any one region.

Work on robotic control laws for patrol applications includes [1] and [2], which ensure that constraints are satisfied on the frequency with which agents visit the locations to be monitored; [3], [4], and [5], which study the problem from a game-theoretic perspective and develop stochastic policies to avoid predictability that can

Peter Kingston e-mail: `kingston@gatech.edu` · Magnus Egerstedt e-mail: `magnus@ece.gatech.edu`
Georgia Institute of Technology, Atlanta, GA 30332, USA

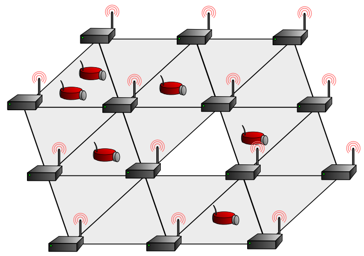

Fig. 1 Multiple mobile robots (red) execute distributed patrol strategies in a triangulated environment with the help of wireless base stations (dark gray).

be exploited by adversaries; and [6], which suggests that chaotic systems be used to generate patrol trajectories.

This paper takes a different, algebraic-topological approach, in which we will aim to produce interesting patrol strategies using *linear* protocols, whose characteristics come inherently from the underlying topology of the environment. At the technical heart of our work lies the *Helmholtz-Hodge decomposition*, primarily of *chains* on simplicial complexes (as in [7] or [8]), but also of a vector field on a smooth manifold (see e.g. [9]). Analogies between these discrete and continuous objects is, implicitly, central to *discrete exterior calculus* (discussed in [10]), which has found application in a number of areas (e.g. [11], [12], [13], [14]). Our approach is closest to [13] and [14], in that representatives of the first homology group are found, but, differs fundamentally in that *controllers over the continuous space are synthesized*, which will drive the development of modified Laplacians, whose null spaces characterize the controllers of interest.

We will proceed by introducing some mathematical preliminaries; then describing how, by using existing algorithms, one can *almost* satisfy a particular collection of necessary requirements; and finally presenting new, unified algorithms that, in addition to being considerably simpler than the former, satisfy *all* of the needed requirements.

2 Background

The environment to be patrolled by the mobile agents will be modeled as a simplicial complex – specifically, as an *abstract simplicial complex* (which captures the topology of the static infrastructure), together with a *realization* for that abstract simplicial complex (which describes where the infrastructure is in space). The following paragraphs briefly give formal definitions for these and related objects, in order to introduce notation prior to the problem statement in Section 3. For more background, the interested reader may wish to refer to [7], [8], and the introductions to [12] and [14], all of which use similar definitions.

Given a finite set $V(K)$ of *vertices* – each of which will, in our case, happen to represent a static infrastructure node – an *oriented simplex* $\Delta \subset V(K)$ is an ordered subset of $V(K)$ (modulo even permutations, and with a formal sign[1]); a simplex with $k+1$ vertices has *order* k. The i-th oriented face of an oriented k-simplex $\Delta = [v_0, \cdots, v_k]$ is, the $k-1$-simplex

$$F_i(\Delta) = (-1)^i [v_0, \cdots, v_{i-1}, v_{i+1}, \cdots, v_k] = (-1)^i \Delta / v_i$$

and a *simplicial complex* K is a finite set of simplices that is closed with respect to taking faces; i.e., if $\Delta \in K$ and σ is a face of Δ, then $\sigma \in K$. A simplicial k-complex K is said to be *pure* if all simplices whose order is less than k are faces of higher-order simplices; and *consistently oriented* if, for every pair of lower-adjacent k-simplices Δ_1, Δ_2 sharing a face σ, Δ_1 and Δ_2 induce opposite orientations on σ. We denote the k-simplices of K by $\Sigma_k(K)$. A simplex $\Delta \in K$ is a *coface* of $\sigma \in K$ if σ is a face of Δ. Two simplices σ_1, σ_2 are *lower-adjacent* (denoted $\sigma_1 \smile \sigma_2$) if they share a face, and *upper-adjacent* (denoted $\sigma_1 \frown \sigma_2$) if they share a coface.

We will model flows of agents throughout the environment using *chains* on the complex, which assign numbers to oriented simplices. Formally, a *k-chain* $c \in C_k(K)$ over an oriented simplicial complex K is a formal sum of elements from $\Sigma_k(K)$ taking coefficients from some commutative ring – the reals, $\mathbb{R}$, in our case. For instance, the formal sum $1.2v_0 + 2.6v_1 - 0.5v_4$ is a 0-chain over an appropriate simplicial complex. Formal sums can be added and multiplied by scalars in the natural way, so $C_k(K)$ forms a finite-dimensional real vector space. Additionally, we equip $C_k(K)$ with an inner product, $\langle \cdot, \cdot \rangle$, defined by

$$\left\langle \sum_{i=0}^{N} a_i \sigma_i, \sum_{i=0}^{N} b_i \sigma_i \right\rangle = \sum_{i=0}^{N} a_i b_i \tag{1}$$

where $\Sigma_k(K) = \{\sigma_0, \cdots, \sigma_N\}$, and $a_i, b_i \in \mathbb{R} \; \forall i$ are the chain coefficients.

We will work with a number of linear operators and quadratic forms throughout this paper, almost all of which are built from *boundary operators*. The k-th boundary operator $\delta_k(K) : C_k(K) \to C_{k-1}(K)$ on the oriented simplicial complex K is defined,

$$\delta_k(K) \left(\sum_{i=0}^{N} a_i \sigma_i \right) = \sum_{i=0}^{N} a_i \sum_{j=0}^{k} F_j(\sigma_i) \; ; \tag{2}$$

by convention, $\delta_0(K) = 0$. The null space of $\delta_k(K)$ is called the *k-cycles* of K and denoted $Z_k(K)$; these represent loops, closed surfaces, and closed hypersurfaces. The image of $\delta_{k+1}(K)$ is called the *k-boundaries* and denoted $B_k(K)$; its elements are the faces of individual higher-dimensional simplices. The k-th homology group – which is the vector space that our controllers will represent – is the quotient space $H_k(K) = Z_k(K)/B_k(K)$; in words, it the space one obtains by identifying cycles that can "continuously deformed" (i.e., obtained by adding and subtracting simplex

[1] The formal sign is necessary only to allow 0-simplices to have two orientations.

boundaries) into one another. Finally, the dimension of $H_k(K)$ is called the *k-th Betti Number* of K. E.g., the first Betti number of a simplicial 3-complex is its number of connected components; the second, the number of "tunnels" through it; and the third, the number of "voids" contained within in.

It will also be useful to define the *boundary subcomplex* $B(K)$ of K, which consists of those faces that agents cannot cross; specifically, the *boundary subcomplex* $B(K)$ of a pure simplicial n-complex K is the $n-1$-subcomplex,

$$B(K) = \mathrm{Cl}\left\{\sigma \in \Sigma_{n-1}(K) \,\middle|\, \begin{array}{l}\sigma \text{ has fewer than} \\ \text{two cofaces.}\end{array}\right\} \tag{3}$$

where Cl denotes simplicial closure.[2]

Finally, a *realization* of a simplicial complex K is an isomorphic complex K' whose vertex set $V(K')$ is a finite subset of $\mathbb{R}^n$ for some $n \in \mathbb{N}$, and the corresponding *Rips Shadow* $\mathscr{R}(K) \subset \mathbb{R}^n$ is the union of the convex hulls of K''s simplices' vertex sets.

With mathematical preliminaries addressed, we now turn to the problem at hand.

3 Problem formulation

We consider an environment in which stationary base stations, like wireless access points or cellular towers, communicate with one another to generate control laws that they supply to mobile agents, like autonomous robots or aerial vehicles, who navigate this environment with their assistance.

At the heart of our problem lies a finite, pure, oriented, abstract simplicial n-complex K, which, together with realization K', both serves as a model for the environment, and represents the information topology of the static infrastructure. The idea is that each vertex $v \in V(K)$ of the complex represents a base station, and that the control laws executed by mobile agents in the Rips Shadow $\mathscr{R}(\Delta) \subset \mathbb{R}^n$ of a given simplex are determined by values communicated by the static agents represented by the vertices $V(\Delta)$ of Δ.

The instrumented environment described by K is inhabited by N mobile agents with time-varying positions $x_1(t), \cdots, x_N(t) \in \mathbb{R}^n$, who perform the patrol tasks with the help of the infrastructure just described. Although the controllers we will develop for these agents are deterministic, it will be helpful to work in a stochastic framework, in which agents are distributed throughout the environment according to a probability measure P defined over $\mathscr{R}(K)$; for our purposes, this distribution can be treated as a smooth, time-varying function $m : \mathscr{R}(K) \times [0,\infty) \to [0,1]$.[3]

[2] I.e., we add whatever simplices are needed to ensure that the complex is closed under taking faces.

[3] This is as opposed to defining P more generally as a tempered distribution, which will not be necessary.

The goal is for the static agents $\Sigma_n(K)$, in a distributed fashion, to produce a family of vector fields $\{f_i : \mathscr{R}(K) \to \mathbb{R}^n\}_{i=1}^M$ having certain desired properties, by which the mobile agents can circulate throughout the environment, as required for robot patrol or aircraft holding-pattern applications.

In particular, we look for any vector field v in this family to satisfy three properties:

P1.1 The first of these, *uniform coverage*, insists that if the initial probability density of the agents is the uniform distribution $m(x,0) = 1/\operatorname{Area}(\mathscr{R}(K))$ over the environment, that this condition persist; i.e., that $m(x,t) = 1/\operatorname{Area}\mathscr{R}(K)$ for all positions x and positive times t.

P1.2 The second, *no local cycles*, encourages that efficient paths without unnecessary loops be traced out by the agents following v, and is expressed by the requirement that no closed integral curve of f be contractible in $\mathscr{R}(K)$ to a point.

P1.3 The third, *zero boundary flux*, requires that agents not leave the environment. Formally, this means that f must have no component orthogonal to the *boundary* of the complex.

The requirements are illustrated by Figure 2.

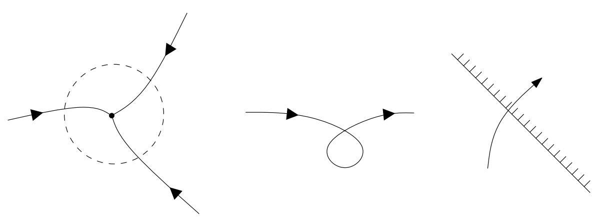

Fig. 2 The vector fields produced avoid the concentration of agents within any one control volume (dashed circle) (left), paths with local loops (center), and collisions with the boundary of the complex (right).

This problem will be addressed in three parts. First, it is shown how existing distributed protocols can be used to produce face fluxes satisfying a *discrete* version of most (but not all) of the requirements P1. Next, symplectic vector fields are generated over the *continuous* space that are consistent with given flows, also in a distributed fashion. Finally, a simple, unified algorithm is presented that solves the continuous and discrete parts of the problem simultaneously, while also satisfying the remaining requirements of P1.

4 Distributed Computation of Homological Streamfunctions

In the next subsections, we will describe two classes of distributed methods for computing vector fields satisfying P1. The first, which serves to motivate the second, employs a method described in [13] to compute face fluxes within K, and then, as a separate step, adapts another distributed algorithm, described in [15], to compute vector fields that are consistent with those fluxes, via streamfunctions. The second, which is one of the contributions of this paper, consists of considerably simplified, unified algorithms that compute the discrete flows and continuous controllers simultaneously, and that also satisfy additional requirements neglected by the first method.

We will use $n-1$-chains on K (i.e., elements of $C_{n-1}(K)$) to represent flows of mobile agents across faces. The idea is that, if a simplex $\sigma \in \Sigma_{n-1}(K)$ appears in a chain with positive coefficient $c \in \mathbb{R}$, then agents are flowing across σ *from* the coface of σ with dissimilar orientation, *to* the coface with similar orientation.

In the subsequent sections, we will be interested in computing face fluxes in $C_{n-1}(K)$ that serve as representatives of the homology group $H_{n-1}(K)$. Later in the paper, continuous control laws will be produced that achieve these fluxes. In order to do this, we will first need to introduce a few Laplacian operators, which we do in the next section.

4.1 Laplacian Operators and Energy Functions

A convenient way to introduce the symmetric Laplacian operators is by means of particular scalar-valued functions, which we will refer to as *energy functions*:

Definition 1. For an abstract simplicial complex K, the k-th *energy function* $\mathscr{E}_k(K)$: $C_k(K) \to \mathbb{R}$ is defined by,

$$\begin{aligned}\mathscr{E}_k(K)(x) = \quad & \frac{1}{2}\sum_{\Delta\in\Sigma_{k+1}(K)} \langle \delta_{k+1}(\Delta), x\rangle^2 \\ & + \frac{1}{2}\sum_{\sigma\in\Sigma_{k-1}(K)} \langle \delta_k^*(\sigma), x\rangle^2 \ ,\end{aligned} \tag{4}$$

with the convention that $\Sigma_i(K) = \emptyset$ for all $i < 0$, and that summations over the empty set evaluate to zero.

By way of these energy functions, the (standard) combinatorial Laplacian can then be defined simply:

Definition 2. For an abstract simplicial complex K, the k-th *combinatorial Laplacian* $\mathscr{L}_k(K) : C_k(K) \to C_k(K)$ is the Hessian of the k-th energy function, $\mathscr{E}_k(K)$.

It will also be useful to define generalizations of the energy functions that omit certain terms from the summations of Definition 1, as well as corresponding Laplacian operators:

Definition 3. Let K be an abstract simplicial complex and $L \subset K$ be a subcomplex of K. Then the k-th *restricted energy function* $\mathscr{E}_k(K,L)$ is defined by,

$$\begin{aligned}\mathscr{E}_k(K,L)(x) = \quad & \frac{1}{2}\sum_{\Delta\in\Sigma_{k+1}(K)/\Sigma_{k+1}(L)} \langle\delta_{k+1}(\Delta),x\rangle^2 \\ & + \frac{1}{2}\sum_{\sigma\in\Sigma_{k-1}(K)/\Sigma_{k-1}(L)} \langle\delta_k^*(\sigma),x\rangle^2 \\ & + \frac{1}{2}\sum_{\sigma\in\Sigma_k(L)} \langle\sigma,x\rangle^2 \end{aligned} \tag{5}$$

The restricted Laplacian is then defined in much the same way as before:

Definition 4. For an abstract simplicial complex K, and subcomplex $B \subset K$, the k-th *restricted combinatorial Laplacian* $\mathscr{L}_k(K,B) : C_k(K) \to C_k(K)$ is the Hessian of the k-th restricted energy function, $\mathscr{E}_k(K,B)$.

Moreover, we will refer to the special case of $L(K,B(K))$ as the *boundary-restricted Laplacian* corresponding to K.

The boundary-restricted Laplacian is particularly useful because it guarantees zero flow out of the complex, while still characterizing the homology group of the complex, as described by the next theorem:

Theorem 1. *Let $x \in C_{k-1}(K)$ be a $(k-1)$-chain on a pure k-complex K. If x is zero on $B(K)$, then x is in the null space of the boundary-restricted Laplacian $\mathscr{L}_{k-1}(K,B(K))$ if and only if it is in the null space of the standard Laplacian $\mathscr{L}_{k-1}(K)$.*

Proof. Since x is zero on $B(K)$, all terms of the third summation of (5) are zero; consequently, we need only consider the first two summations. If $x \in \text{null}\,\mathscr{L}(K)$, it then follows that $x \in \text{null}\,\mathscr{L}(K,B(K))$, since the terms of the first two summations in (5) are a subset of those in (4), and each term is positive. To show the converse, we note that the only terms that appear in (4) but not in the first two summations of (5) correspond to faces in $B(K)$, and, since x is zero on $B(K)$, these terms are zero. ■

Finally, we define a directed zeroeth Laplacian operator:

Definition 5. Let $G = (V,E)$ be a directed graph. Then the *directed graph Laplacian* $L(G) : C_0(V) \to C_0(V)$ is defined by,

$$[L(G)(x)]_i = \sum_{j|(v_j,v_i)\in E} (x_i - x_j) \ . \tag{6}$$

Note that, since G in Definition 5 is directed, it may be that $(v_i,v_j) \in E$ while $(v_j,v_i) \notin E$, in which case $L(G)$ is not symmetric.

With the necessary Laplacian operators thus defined, we now consider the computation of face fluxes that serve as representatives of the first homology group.

4.2 Projection onto the first Homology Group

In this subsection, we describe a distributed algorithm for generating elements of the homology group $H_k(K)$ without global knowledge of the graph topology. These 1-chains will neglect to satisfy P1.3, but serve to motivate subsequent sections.

Recalling that $H_k(K) = Z_k(K)/B_k(K)$, what we will do is produce unique representatives of elements of $H_k(K)$, that have a component in $Z_k(K)$ but not in $B_k(K)$. We are able to do this because a natural isomorphism exists between $H_k(K)$ and $\operatorname{null}\mathscr{L}_k(K)$. Since $\operatorname{null}\mathscr{L}_k(K) = Z_k(K) \cap B_k(K)^{\perp}$, an element $v \in \operatorname{null}\mathscr{L}_k(K)$ is the unique representative of an equivalence class in $H_k(K) = Z_k(K)/B_k(K)$ whose component in $B_k(K)$ is zero.

Since the restricted energy function $\mathscr{E}_1(K, B(K))$ is convex, gradient descent from any point converges asymptotically to $\operatorname{null}\mathscr{L}_k(K, B(K))$. Indeed, since $\mathscr{E}_1(K, B(K))$ has a quadratic but not a linear term, those gradient dynamics are simply,

$$\dot{x} = -\mathscr{L}_k(K, B(K))x \tag{7}$$

which, so long as k is not too large, constitute a distributed method for asymptotically computing the projection of a given k-chain $x(0)$ onto $\operatorname{null}\mathscr{L}_k = H_k(K)$. To elaborate, the sparsity pattern of $\mathscr{L}_k$ implies that this process requires $(k+1)$-hop communication in each round.

In [13], this property of the dynamics (7) was used to project a random 1-chain onto the homology group $H_1(K)$ in order to determine, with unit probability, whether it is trivial (i.e., has dimension zero). In this way, a sensor network could determine whether it contained any holes.

For our purposes, what matters is that this is an algorithm for producing unique representatives of elements of $H_1(K)$. With such a method thus in hand, we now turn our attention to the generation of continuous control laws from these edge flows. When the environment is two-dimensional (i.e., $n = 2$, and K is a pure 2-complex), these can be produced via *streamfunctions*, which we describe in the next subsection.

4.3 Hybrid Streamfunctions

A *streamfunction* $\phi : \mathscr{R}(K) \to \mathbb{R}$ is a scalar-valued function defined throughout the environment $\mathscr{R}(K) \subset \mathbb{R}^2$, whose purpose is to induce the Hamiltonian vector field $f : \mathscr{R}(K) \to \mathbb{R}^2$, defined by

$$f(x) = J \operatorname{grad} \phi(x) \ , \tag{8}$$

where the skew-symmetric matrix $J \triangleq \left[\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right]$ has the interpretation both of being a 90-degree rotation matrix, and of being the matrix representation of the symplectic form $(a,b) \mapsto \det([a,b])$. Note that the (images of the) integral curves of f are exactly the level sets of ϕ.

We will compute streamfunctions over $\mathscr{R}(K)$ that induce vector fields that agree with the flows computed in the previous section in the following sense:

Definition 6. Given an oriented simplicial k-complex K with realization $r : V(K) \to \mathbb{R}^n$, a vector field $f : \mathscr{R}(K) \to \mathbb{R}^n$ *agrees with* a $(k-1)$-chain ν if, for each simplex $\Delta \in \Sigma_{k-1}(K)$, the flux of f across $\mathscr{R}(\Delta)$ equals $\langle \nu, \Delta \rangle$.

It is sufficient to consider piecewise linear streamfunctions that are defined by barycentric interpolation of 0-chains across simplices. This is done in the following way: Letting $\bar{K}'$ be the canonical realization of K, a unique affine map $\beta : \mathscr{R}(K) \to \mathscr{R}(\bar{K})$ exists, under the assumption that the interiors of the Rips Shadows of all the simplices are disjoint, that takes a point $x \in \mathscr{R}(K) \subset \mathbb{R}^2$ to canonical coordinates in $\mathbb{R}^{|V(K)|}$.[4] With this defined, the streamfunction $\phi_c : \mathscr{R}(K) \to \mathbb{R}$ corresponding to a 0-chain $c = c_1 v_1 + \cdots + c_N v_N$ is defined simply by,

$$\phi_c(x) = \sum_{i=0}^{N} c_i \beta_i(x) \ . \tag{9}$$

Since the only nonzero terms of this sum correspond to the vertices of the simplex containing the given point x, this function can be computed at any time using only local information.

Our first link between the continuous and the discrete is then provided by the following simple lemma, which we arrive at by combining (9), (8), and Definition 6:

Lemma 1. *Let K be a finite pure abstract simplicial 2-complex with realization K', $\phi_c : \mathscr{R}(K) \to \mathbb{R}$ as defined by (9), $f : \mathscr{R}(K) \to \mathbb{R}^2$ as defined by (8), and $v = \delta_1^*(c)$. Then f* agrees with *v (in the sense of Definition 6).*

In short, it is precisely the coboundary operator $\delta_1^* : C_0(K) \to C_1(K)$ that maps from the 0-chains that represent streamfunctions to the corresponding 1-chains that represent face fluxes.

In [15], it was shown as a corollary that, within a 2-simplex $\Delta = (v_1, v_2, v_3) \in \Sigma_2(K)$, the 0-chain corresponding to a discrete incompressible flow $v \in C_1(K)$ can be computed as,

$$c = [\delta_1^*(\Delta)]^\dagger v + \mathbf{1}s \tag{10}$$

where $\mathbf{1} \in C_0(\Delta)$ is the 0-chain that assigns a 1 to each vertex and s is a separate 2-chain computed by another Laplacian-based distributed algorithm (described in [15]). In this manner, a continuous streamfunction is produced that agrees with a given divergence-free 1-chain on G.

[4] For each point $x \in \mathscr{R}(K)$, this is a sparse vector, whose only nonzero elements correspond to the vertices of the simplex containing x; these take the values of the barycentric coordinates of x in that simplex.

Now that we have both (1) a method for computing 1-chains in $H_1(K)$ and (2) a method for computing streamfunctions that agree with them, in principle the two algorithms can simply be composed to produce homological streamfunctions satisfying P1.1 and P1.2. However, it is possible to perform a similar computation in a more unified manner, while additionally satisfying P1.3, as described in the next section.

5 Unified algorithms

The key idea of this section is that the two algorithms just described can be unified by considering as a whole the properties that the 0-chain defining the streamfunction must satisfy; in the process, we will enforce the additional constraints imposed by P1.3. We will first consider a undirected, 2-hop algorithm that follows directly from our definitions, before introducing a directed, 1-hop algorithm that, remarkably, converges to the same set.

5.1 An undirected, 2-hop algorithm

The most straightforward approach, which results in an at-most-2–hop algorithm, results from composing the boundary and energy operators, as follows:

Theorem 2. *The dynamics,*

$$\dot{c}(t) = \delta_1(K)\mathscr{L}_1(K,B(K))\delta_1^*(K)c(t)\ \forall t > 0 \tag{11}$$

converge asymptotically from any initial condition $c(0) \in C_0(K)$ to a value $c(\infty) \in C_0(K)$ such that, if $x = \delta_1^*(K)c$, then $x \in \operatorname{null}\mathscr{L}(K,B(K))$.

Proof: The restricted energy function $\mathscr{E}(K,B(K))$ can be written,

$$\mathscr{E}(K,B(K))(x) = \frac{1}{2}x^*\mathscr{L}_1(K,B(K))x\,; \tag{12}$$

consequently

$$\mathscr{E}(K,B(K))(\delta_1^*(K)c) = \frac{1}{2}c^*(t)\delta_1(K)\mathscr{L}_1(K,B(K))\delta_1^*(K)c(t) \tag{13}$$

whose gradient-descent dynamics with respect to c are (11). ■

What this means is that, by running the simple, linear, 2-hop protocol (11), we can asymptotically compute a 0-chain, that induces a streamfunction, that induces a vector field satisfying P1. However, in the next section, we will see that this is also achievable with a directed, 1-hop algorithm.

5.2 A directed, 1-hop algorithm

Ultimately, we will produce a 0-chain satisfying the desired properties (i.e., that induces a vector field satisfying P1) by running Laplacian dynamics on a *directed* graph built from the complex K, which allows information to flow bidirectionally within the interior of the complex, as well as bidirectionally within the boundary, but only in a single direction between the two. This directed graph, which we will refer to as the *insulated 1-skeleton*, $G(K)$, only allows a 1-way flow of information from the boundary to the interior of the complex.

Definition 7. The *insulated 1-skeleton* $G(K) = (V, E)$ of a pure simplicial n-complex K is the graph with vertex set $V = \Sigma_0(K)$, in which

P2.1 For all $a, b \in \Sigma_0(K/B(K))$, we have $(a,b),(b,a) \in E$ if and only if $(a,b) \in \Sigma_1(K)$ or $(b,a) \in \Sigma_1(K)$.
P2.2 For all $a, b \in \Sigma_0(B(K))$, we have $(a,b),(b,a) \in E$ if and only if $(a,b) \in \Sigma_1(B(K))$ or $(b,a) \in \Sigma_1(B(K))$
P2.3 For all $a \in \Sigma_0(B(K)), b \in \Sigma_0(K/B(K))$, we have $(a,b) \in E$ if and only if $(a,b) \in \Sigma_1(K)$ or $(b,a) \in \Sigma_1(K)$.
P2.4 For all $a \in \Sigma_0(K/B(K)), b \in \Sigma_0(B(K))$, we have $(a,b) \notin E$.

Definition 7 is illustrated by Figure 5.2. In essence, one undirected graph links the vertices in the interior of the complex; another links those in its boundary; and edges between the two are directed from the boundary to the interior.

The insulated 1-skeleton $G(K)$ produced through definition 7 is illustrated by Figure 5.2. The next theorem explains how consensus dynamics on $G(K)$ can then be used to compute representatives of the homology group and streamfunctions together in a unified way.

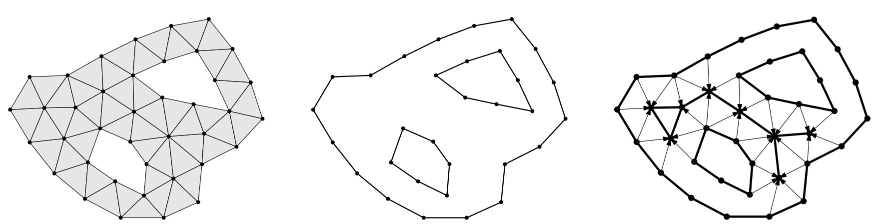

Fig. 3 A pure simplicial 2-complex K (left), its *boundary subcomplex* $B(K)$ (center), and the corresponding *insulated 1-skeleton* $G(K)$ (right)

Theorem 3. *Let K be a pure simplicial n-complex, $G(K)$ its insulated 1-skeleton, and L the directed Laplacian corresponding to $G(K)$. For any 0-chain $c_0 \in C_0(G(K))$, the directed Laplacian dynamics,*

$$\begin{aligned} \dot{c}(t) &= -Lc(t) \quad \forall t > 0 \\ c(0) &= c_0 \end{aligned} \tag{14}$$

converge asymptotically to a 0*-chain* c_∞ *such that* $\delta_1^*(c_\infty) \in \operatorname{null}\mathscr{L}_1(K,B(K))$.

Proof : We must demonstrate that the system is stable, and that the linear subspace $\{c \mid \delta_1^* c \in \operatorname{null}\mathscr{L}_1(K,B(K))\}$ is its equilibrium set. First, we address stability. Without loss of generality, the dynamics (14) can be block-decomposed as,

$$\begin{bmatrix} \dot{x} \\ \dot{b} \end{bmatrix} = \begin{bmatrix} -(L_x + D_{xb}) & C \\ \mathbf{0} & -L_b \end{bmatrix} \begin{bmatrix} x \\ b \end{bmatrix} \tag{15}$$

where L_x is the (undirected) Laplacian for the undirected graph $G(K)/B(K)$, D_{xb} is a diagonal, positive-semidefinite matrix (representing the extra degree due to edges connecting vertices in $G(K)/B(K)$ to $B(K)$), L_b is the (undirected) Laplacian for B, and C_{xb} is a coupling matrix representing edges from $G(K)/B(K)$ to $B(K)$.

To demonstrate stability, we must show that none of the Jordan blocks for the system's zero eigenvalues are larger than 1x1, and that all eigenvalues are nonpositive. First, note that, although the lower diagonal block, $-L_b$, does have a nontrivial nullspace, it, by the Spectral Theorem, is diagonalizable, so *none* of its Jordan Blocks are larger than 1x1. Next, consider the upper diagonal block $-(L_x + D_{xb})$: Since there is at least one edge from B to each connected component of $G(K)/B(K)$, no zero-chain in the null space of L_x (i.e., that is constant on each connected component of $G(K)/B(K)$) can also be in the null space of D_{xb}, and consequently $-(L_x + D_{xb})$ is negative *definite*. Since the upper diagonal block has strictly negative eigenvalues, and, although the lower block does have zero eigenvalues, they correspond to Jordan blocks of size 1x1, the system is stable.

Next, we demonstrate that the equilibrium set is precisely the subspace $\{c \mid \delta_1^* c \in \operatorname{null}\mathscr{L}_1(K,B(K))\}$, by considering in turn each of the requirements that a vector in $\operatorname{null}\mathscr{L}_1(K,B(K))$ must satisfy, with reference to (5):

- *No circulation:* The first sum of (5) is zero if and only if, for each simplex $\Delta \in \Sigma_2(K)/\Sigma_2(B(K))$, we have $\langle \delta_2(\Delta), \delta_1^*(c_\infty)\rangle = 0$, or equivalently (by definition of an adjoint operator), if $\langle \delta_1\delta_2(\Delta), c_\infty\rangle = 0$. Since $\delta_k\delta_{k+1} \equiv 0$ for all $k \in \mathbb{N}$, this is automatically true.
- *No divergence in interior:* The second sum of (5) is zero if and only if, for each $\sigma \in \Sigma_0(K)/\Sigma_2(B(K))$, we have $\langle \delta_1^*(\sigma), \delta_1^*(c_\infty)\rangle = 0$, or, equivalently, if $\langle \sigma, \delta_1\delta_1^* c_\infty\rangle = \langle \sigma, \mathscr{L}_0(c_\infty)\rangle = 0$. This is precisely the condition, specified by the upper diagonal block of the system matrix, for vertices in $\sigma \in \Sigma_0(K)/\Sigma_2(B(K))$ to be at equilibrium.
- *Boundary condition:* The third sum of (5) is zero if and only if, for each $\sigma \in \Sigma_1(B(K))$, we have $\langle \sigma, \delta_1^*(c_\infty)\rangle = 0$, or, equivalently, iff c_∞ is constant on each connected component of $B(K)$. The lower diagonal block of the system matrix is the standard zeroeth combinatorial Laplacian for $B(K)$, and its null space is exactly such zero-chains.

With stability guaranteed and the equilibrium set characterized, we conclude the proof. ∎

6 From the discrete to the continuous

We next show that the 0-chains produced by, e.g., (14), in fact do induce vector fields that satisfy the properties P1 for which we have aimed. This will be stated by Theorem 4, which will follow the next, prerequisite lemma:

Lemma 2. *Let K be a finite pure abstract simplicial 2-complex with realization K', let $c \in \operatorname{null}\mathscr{L}_1(K,B(K))$, $\phi_c : \mathscr{R}(K) \to \mathbb{R}$ according to (9), and $x \in \operatorname{int}\mathscr{R}(K)$. Then $\phi(x) \geq \min\phi$.*

Proof : A point $x \in \mathscr{R}(K)$ lies in exactly one simplex of the realization K' of K, of some order $m \leq 2$, and with vertices $v_1, \cdots, v_m$. First, consider when $m > 0$. Letting $j = \operatorname{argmax}_{k=1,\cdots,m}\{\phi(v_k)\}$, we define $\gamma : [0,1] \to \mathscr{R}(K)$ by $\gamma(t) = (1-t)x + tv_j$. Then, since $\phi(x)$ is a convex combination of $\phi(v_1), \cdots, \phi(v_m)$ (by (9)), we note that $\phi \circ \gamma$ is a decreasing function, so x is not a strict local minimum. Next, consider the case when $m = 0$, and let $u_1, \cdots, u_p$ be those vertices edge-adjacent to $v_1 = x$. We then define $j = \operatorname{argmax}_{k=1,\cdots,m}\{\phi(v_k)\}$ and $\gamma(t) = (1-t)x + tu_j$. Since $c \in \operatorname{null}\mathscr{L}_1(K,B(K))$ and $v_1 \notin B(K)$, we have $\langle c, v_1\rangle = \frac{1}{p}(\langle c, u_1\rangle + \cdots + \langle c, u_p\rangle) \geq u_j$, so $\phi \circ \gamma$ is again decreasing and x is again not a strict local minimum. ∎

Lemma 2 can be seen as an analogue to the Maximum Principle for harmonic functions, which states that those functions take their extreme values on the boundary of their domain. Lemma 2 states the same property for the piecewise-linear functions that constitute our streamfunctions. This property will be useful in the proof of the next theorem, which shows that the vector fields that we generate indeed satisfy our desired requirements P1.

Theorem 4. *Let K be a finite pure abstract simplicial 2-complex with realization K', $c \in \operatorname{null}\mathscr{L}_1(K,B(K))$, $v = \delta_1^*(c)$, and define $\phi : \mathscr{R}(K) \to \mathbb{R}$ and $f : \mathscr{R}(K) \to \mathbb{R}^2$ by (9) and (8) respectively. Then f satisfies P1.*

Proof : We will address each property in turn:

1. Since the vector field f is Hamiltonian, it satisfies P1.1 by Liouville's theorem.
2. Suppose $\gamma[0,1] \to \operatorname{int}\mathscr{R}(K)$ is a closed integral curve of (8) that is contractible to a point $p \in \operatorname{int}\mathscr{R}(K)$. Since γ is closed, by the Jordan curve theorem it divides $\mathbb{R}^2/\gamma([0,1])$ and consequently $\mathscr{R}(K)/\gamma([0,1])$ into two disjoint sets, A and B. Let A be the set that contains p. Since γ is an integral curve of (8), $(\phi \circ \gamma)(t) \equiv k$ for some $k \in \mathbb{R}$, and, since $\operatorname{grad}\phi(\gamma(t)) \not\equiv 0$, there exists a point $q \in A$ such that $\phi(q) \neq k$ (e.g., arbitrarily close to $\gamma([0,1])$); without loss of generality, assume $\phi(q) > k$. Since $A \cup \gamma([0,1])$ is compact, it contains a maximizer $s \in A \cup \gamma([0,1])$ to ϕ, and, since $\phi(s) \geq \phi(q) > k$, $s \notin \gamma([0,1])$; i.e., γ encloses a strict local maximum to ϕ. However, by Lemma 2, such a point cannot exist in $\operatorname{int}\mathscr{R}(K)$, so we have a contradiction, and P1.2 is satisfied.
3. Since $c \in \operatorname{null}\mathscr{L}_1(K,B(K))$, then for any vertices v_1, v_2 in the same connected component of $S(v_1,v_2) \subset B(K)$, $\langle c, v_1\rangle = \langle c, v_2\rangle$. Consequently, ϕ_c is constant on $\mathscr{R}(S(v_1,v_2))$; by the chain rule, $\operatorname{grad}\phi_c \perp S(v_1,v_2)$; and by (8), $f \parallel S(v_1,v_2)$; i.e., f satisfies P1.3. ∎

With it now demonstrated that the vector fields generated satisfy P1, we describe simple algorithms that take advantage of this fact to produce linearly-independent families of vector fields satisfying those properties.

6.1 Infrastructure-Assisted Behavior Generation

In an environment with h holes,[5] the distributed homological-streamfunction generation algorithm of the previous section can be employed in a straightforward way to generate, with probability 1, an h-dimensional vector space of patrol behaviors. With the "heavy lifting" done by the distributed projection algorithm, the remainder of the behavior generation algorithm is exceedingly simple: So long as $q \geq h$, the

Input: Oriented simplicial complex K; $q \in \mathbb{N}$
Output: $c_{\infty,1}, \cdots, c_{\infty,q}$
Algorithm:

- For $i = 1$ to q
  - Generate random 0-chain $c_0 \in C_0(K)$, according to any probability distribution p on $C_0(K)$, the span of whose support is $C_0(K)$.
  - Run dynamics (14) or (11) from initial condition c_0 until convergence; store result as $c_{\infty,i}$.

Fig. 4 Distributed algorithm for computing homological patrol strategies.

resulting family of vector fields induced by the zero-chains $c_{\infty,1}), \cdots, c_{\infty,q}$ will span the space of vector fields satisfying P1.

A possible objection can be raised, which is that, if q is chosen according to a conservative upper bound on h, then the set of behaviors obtained will not be linearly independent; i.e., vector fields will be generated that are redundant in the sense that they lie in the span of the others. In general, two ways to deal with this situation exist:

The first is to perform distributed orthonormalization. This is the approach taken in e.g. [16], [17], [18], [19], which perform distributed Arnoldi-like iterations to compute Laplacian spectra. The unavoidable disadvantage of approaches of this type is that the computation of inner products inherently requires sums across all of the agents; hence each outer iteration of these algorithms involves an entire consensus problem to compute inner products, and the algorithms are consequently (and necessarily) quite slow.

The second is to accept redundancy. So long as our objective is to generate behaviors satisfying the properties P1, there is no particular reason to believe that a *minimal* set of such behaviors is required. Indeed, it is precisely by relaxing the orthonormality requirement found in spectral algorithms that the algorithms we present obtain their speed.

[5] i.e., in a 2-complex whose first Betti number is h

7 Implementation

An experimental framework was developed in which the distributed algorithm (14) can be used to project either random flows as in Figure 6.1, or human-generated inputs – obtained from motion-capture – onto constraint subspaces like that specified by P1. Figure 7 illustrates the environment.

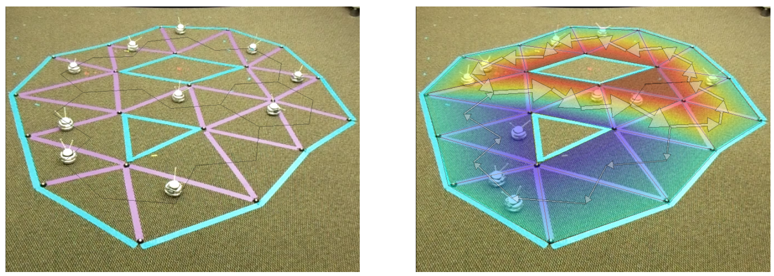

Fig. 5 Khepera III mobile robots in a simplicial complex (left) (internal edges are shown in purple and boundary edges in blue), and robots moving in the same complex according to a streamfunction, overlaid (right).

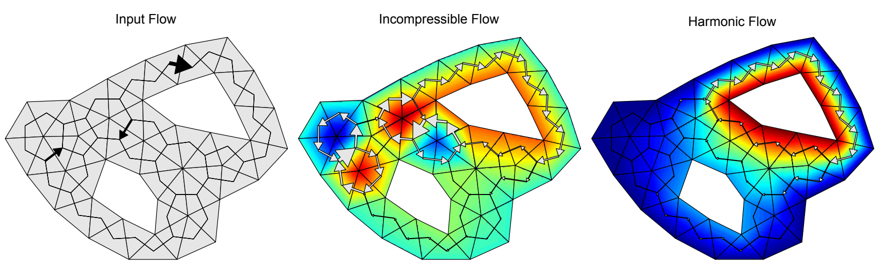

Fig. 6 Depicted are a specified flow (left), and its projections onto the incompressible (center) and harmonic (right) subspaces. The harmonic flow described in this paper (right) differs from the incompressible flow described in [15] (center), in that the former avoids the local vortices visible in the latter. In both cases, it is a collection of hybrid, piecewise-linear controllers that realize the flows. These controllers are produced as the Hamiltonian vector field corresponding to a piecewise-linear streamfunction (color gradients).

8 Conclusion

We have developed a collection of distributed, consensus-like algorithms by which static infrastructure nodes can synthesize controllers for mobile agents that cause them to circulate throughout an environment without either concentrating their mass in any location, or following paths with contractible loops. This is done first by combining existing algorithms for computing flows and synthesizing controllers that agree with the flows; then by, in a unified fashion, computing controllers and flows together in a symmetric, 2-hop algorithm; and finally by an equivalent 1-hop algorithm that, remarkably, arises from a *directed* Laplacian. The result is a family of linear protocols that converge exponentially to hybrid controllers representing the topology of the environment.

Acknowledgements This work was supported in part by the Office of Naval Research through MURI-HUNT.

References

1. Y. Elmaliach, N. Agmon, and G.A. Kaminka, "Multi-robot area patrol under frequency constraints," in *Robotics and Automation, 2007 IEEE International Conference on*, april 2007, pp. 385 –390.
2. F. Sempe and A. Drogoul, "Adaptive patrol for a group of robots," in *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, October 2003, vol. 3, pp. 2865 – 2869 vol.3.
3. N. Agmon, S. Kraus, and G.A. Kaminka, "Multi-robot perimeter patrol in adversarial settings," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, may 2008, pp. 2339 –2345.
4. Noa Agmon, Sarit Kraus, Gal A. Kaminka, and Vladimir Sadov, "Adversarial uncertainty in multi-robot patrol," in *Proc IJCAI 2009*, 2009.
5. F. Amigoni, N. Gatti, and A. Ippedico, "A game-theoretic approach to determining efficient patrolling strategies for mobile robots," 2008.
6. Luiz S. Martins-Filho and Elbert E. N. Macau, "Patrol mobile robots and chaotic trajectories," .
7. A. Hatcher, *Algebraic Topology*, Cambridge University Press, 2002, `http://www.math.cornell.edu/~hatcher/AT/ATpage.html`.
8. James R. Munkres, *Elements of Algebraic Topology*, Addison-Wesley, 1984.
9. John M. Lee, *Introduction to Smooth Manifolds*, Springer, 2000.
10. M. Desbrun, A. N. Hirani, M. Leok, and J. E. Marsden, "Discrete exterior calculus," (preprint, `http://arxiv.org/abs/math/0508341`), 2003.
11. Yiying Tong, Santiago Lombeyda, Anil N. Hirani, and Mathieu Desbrun, "Discrete multiscale vector field decomposition," in *Proc. SIGGRAPH*, 2003.
12. X. Jiang, L.-H. Lim, Y. Yao, and Y. Ye, "Statistical ranking and combinatorial hodge theory," *Mathematical Programming*, vol. 127, pp. 203–244, 2011.
13. Abubakr Muhammad and Ali Jadbabaie, "Decentralized computation of homology groups in networks by gossip," in *Proc. American Control Conference*, 2007.
14. Alireza Tahbaz-Salehi and Ali Jadbabaie, "Distributed coverage verification algorithms in sensor networks without location information," *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1837–1849, August 2010.
15. P. Kingston and M. Egerstedt, "Distributed-infrastructure multi-robot routing using a helmholtz-hodge decomposition," in *Proc. Conference on Decision and Control*, 2012.
16. Leonardo Borges and Suely Oliveira, "A parallel davidson-type algorithm for several eigenvalues," vol. 144, pp. 727–748, 1998.
17. S. Korada, A. Montanari, and S. Oh, "Gossip pca," 2011.
18. A. Srinivasan and M. Mascagni, *Monte Carlo Techniques for Estimating the Fiedler Vector in Graph Applications*, 2002.
19. Michael Holzrichter and Suely Oliveira, "A graph based method for generating the fiedler vector of irregular problems," in *In Lecture Notes in Computer Science*. 1999, pp. 978–985, Lecture.