Distributed Topology using Harmonics

Harish Chintakunta
SAMSI Workshop on Topological Data Analysis

February 6, 2014
Why do we need Distributed Algorithms?
Massively parallel architectures and supercomputers.
High Performance Computing

- Massively parallel architectures and supercomputers.
- Exploit the decoupled natured of processing requirements to speed up computations.
High Performance Computing

- Massively parallel architectures and supercomputers.
- Exploit the decoupled natured of processing requirements to speed up computations.
- Essential for coping with the large size of data.
Sensor Networks

- Will be ubiquitous in the future.
- Distributed algorithms ensure
  - In-network processing.
  - Low cost.
  - Robust to failures.
  - Fast response time
What are Harmonics?
Harmonics

Definition

- For a given complex $K$, and the boundary operators
  \[ \partial_{k+1} : C_{k+1} \to C_k, \partial_k : C_k \to C_{k-1} \]

- Consider the $k^{th}$ combinatorial Laplacian
  \[ L_k = \partial_{k+1} \partial_{k+1}^T + \partial_k^T \partial_k \]

- The elements in the null space of the $k^{th}$ Laplacian, $\ker(L_k)$, are denoted as $k$-harmonics.
Given a cycle $c$, is there a computationally efficient method to test for contractibility?
Given a cycle \( c \), is there a computationally efficient method to test for contractibility?
Given a cycle $c$, is there a computationally efficient method to test for contractibility?

The simplest method would be to construct the matrix $[\partial_{k+1} c]$ and reduce.
Given a cycle \( c \), is there a computationally efficient method to test for contractibility?

The simplest method would be to construct the matrix \( [\partial_{k+1} c] \) and reduce.

Can we do better?
Let $Z_k \subseteq C_k$ denote the space of all $k$-cycles, and $B_k = \text{Img}(\partial_{k+1})$ denote the space of $k$-boundaries.

- $L_k = \partial_{k+1} \partial^T_{k+1} + \partial_T^k \partial_k$
- For any $y \in \ker(L_k)$, $y \in Z_k$, and $y \perp B_k$, i.e., $\langle y, c \rangle = 0$, $\forall c \in B_k$.

**Theorem**

Let $c \in Z$ be a cycle in $Z$, and let $y$ be a generic element in $\ker(L_1)$. Then $c \in \text{img}(\partial_2) \iff \langle y, c \rangle = 0$ with probability 1.
Harmonics

Persistence

\[ K_1 \xrightarrow{i_1} K_2 \xrightarrow{i_2} K_3 \xrightarrow{i_3} K_4 \]
Let \([c] \in H_k(K_1)\). We want to check if \([c] \neq 0\), and if \(i_2^*(\langle c \rangle) \neq 0\).
Let \([c] \in H_k(K_1)\). We want to check if \([c] \neq 0\), and if \(i_2^*[c] \neq 0\).

Compute \(y_1 \in \ker(L_k(K_1))\), and \(y_3 \in \ker(L_k(K_3))\). If \(\langle y_1, c \rangle \neq 0\), and \(\langle y_3, c \rangle \neq 0\), then \([c]\) persists from \(K_1\) through \(K_3\).
$y \in \text{ker}(L_k)$ can be computed using the iteration

$$y^{k+1} = y^k - \delta L_k y^k$$

starting with a random vector $y^0$. 
Harmonics
Computation

$y \in \ker(L_k)$ can be computed using the iteration

$$y^{k+1} = y^k - \delta L_k y^k$$

starting with a random vector $y^0$.

**Theorem**

Let $K$ be the matrix with column space equal to the null space of $L_k$. Then the iteration converges to $y^\infty = KK^T y^0$ if and only if $\delta$ satisfies the following inequality:

$$0 < \delta < 2/\lambda_i, \forall i$$

where $\lambda_1 > \lambda_2 > \cdots$ are the eigenvalues of $L_k$. 
Distributed algorithms

Multiplication by a Matrix

Consider the operation of

\[ Y = LX \]
Consider the operation of

\[ Y = LX \]

- the value at the \( i^{th} \) row is

\[ y_i = \sum_j L_{ij}x_j = \sum_{j \in \mathcal{N}_i} x_j \]

where \( \mathcal{N}_i \) is the set of neighbors of node \( i \)
Distributed algorithms

Multiplication by a Matrix

Consider the operation of

\[ Y = LX \]

- The value at the \( i^{th} \) row is

\[ y_i = \sum_j L_{ij} x_j = \sum_{j \in \mathcal{N}_i} x_j \]

where \( \mathcal{N}_i \) is the set of neighbors of node \( i \)

- This can be accomplished by each node broadcasting its value to its neighbors.
Application Coverage in Sensor Networks
Coverage area
Coverage area

Balls of radius $r_c$
Coverage Problem in literature
Algebraic topological approaches

Coverage Problem in literature
Algebraic topological approaches


Problem Statement

- Detection: is $H_1(R_c) \neq 0$
Problem Statement

- Detection: is $H_1(R_c) \neq 0$

- Localization: compute sparse generators for $H_1(R_c)$
Problem Statement

- Detection: is $H_1(R_c) \neq 0$

- Localization: compute sparse generators for $H_1(R_c)$

- Sparse generators also have many other applications †

---

A direct method: Čech complex

- Let $V$ be the set of points in $\mathbb{R}^2$ (the nodes)
A direct method: Čech complex

- Let $V$ be the set of points in $\mathbb{R}^2$ (the nodes)

- The Čech complex $\check{C}(V, r_c)$ has the homotopy type of the union of the balls $B(V, r_c)$
A direct method: Čech complex

- Let $V$ be the set of points in $\mathbb{R}^2$ (the nodes).

- The Čech complex $\check{C}(V, r_c)$ has the homotopy type of the union of the balls $B(V, r_c)$.

- The detection problem is then: is $H_1(\check{C}(V, r_c)) = 0$?
Let

\[ E = \{ e = (v_i, v_j), \|v_i - v_j\| \leq 1 \} \]
Let 
\[ E = \{ e = (v_i, v_j), \|v_i - v_j\| \leq 1 \} \]
the graph \( G_1 = (V, E) \) is called the Unit Disk Graph (UDG) and its flag complex is the Rips complex \( K_{G_1} \)
Let
\[ E = \{ e = (v_i, v_j), \|v_i - v_j\| \leq 1 \} \]
the graph \( G_1 = (V, E) \) is called the Unit Disk Graph (UDG) and its flag complex is the Rips complex \( K_{G_1} \)
for a simplex \( \sigma \in K_{G_1} \), let \( Conv(\sigma) \) denote the convex hull of its vertices in the plane. The Rips shadow \( R_s \) is then the union \( \bigcup_{\sigma} Conv(\sigma) \).
Let 
\[ E = \{e = (v_i, v_j), \|v_i - v_j\| \leq 1\} \]
the graph \( G_1 = (V, E) \) is called the Unit Disk Graph (UDG) and its flag complex is the Rips complex \( K_{G_1} \)

for a simplex \( \sigma \in K_{G_1} \), let \( \text{Conv}(\sigma) \) denote the convex hull of its vertices in the plane. The Rips shadow \( R_s \) is then the union \( \bigcup_{\sigma} \text{Conv}(\sigma) \).

When \( r_c = 1/2 \), Rips shadow is a good approximation to the coverage area.
Theorem (Chambers et al, 2007)

For any set of points in $\mathbb{R}^2$, $\pi_1(p) : \pi_1(K_{G_1}) \rightarrow \pi_1(R_s)$ is an isomorphism, where $p$ is the projection map.
Theorem (Chambers et al., 2007)

For any set of points in $\mathbb{R}^2$, $\pi_1(p) : \pi_1(K_{G_1}) \rightarrow \pi_1(R_s)$ is an isomorphism, where $p$ is the projection map.

...this justifies the use of Rips complex to analyze the coverage area.
The knowledge of $G_1$ requires perfect length information for $G_1 = (V, E)$,

$$(v_1, v_2) \in E \iff \|v_1 - v_2\| \leq 1$$
The knowledge of $G_1$ requires perfect length information

for $G_1 = (V, E)$,

$$(v_1, v_2) \in E \iff \|v_1 - v_2\| \leq 1$$

In general, this is not true
The knowledge of $G_1$ requires perfect length information for $G_1 = (V, E)$,

$$(v_1, v_2) \in E \iff \|v_1 - v_2\| \leq 1$$

In general, this is not true.
Quasi unit disk graphs

\[ G_1^\epsilon = (V, E) \]

\[ \|v_1 - v_2\| \leq 1 - \epsilon \Rightarrow (v_1, v_2) \in E \]

\[ 1 - \epsilon < \|v_1 - v_2\| \leq 1 \]

\[ \Rightarrow (v_1, v_2) \in E \text{ w.p } 0.5 \]
Quasi unit disk graphs

- $G_1^\epsilon = (V, E)$
  
  \[ \|v_1 - v_2\| \leq 1 - \epsilon \Rightarrow (v_1, v_2) \in E \]
  
  \[ 1 - \epsilon < \|v_1 - v_2\| \leq 1 \]
  
  \[ \Rightarrow (v_1, v_2) \in E \text{ w.p } 0.5 \]

- $K_{G_1^\epsilon}$ does not have the same homology as its shadow.
It gets worse!!

Theorem (Chambers et al, 2007)

Given any value $\epsilon$, and any finitely presented group $G$, there exists a quasi-Rips complex $K_{G^\epsilon_i}$ with $\pi_1(K_{G^\epsilon_i}) \cong G \ast F$, where $F$ is a free group.
It gets worse!!

**Theorem (Chambers et al, 2007)**

*Given any value $\epsilon$, and any finitely presented group $G$, there exists a quasi-Rips complex $K_{G_\epsilon}$ with $\pi_1(K_{G_\epsilon}) \cong G \ast F$, where $F$ is a free group.*

...good news is coming...
Persistence to the rescue

- We can get a good estimate of $\epsilon$
Persistence to the rescue

- We can get a good estimate of $\epsilon$
- Consider the following filtration:

$$K_{G_1^\epsilon} \xrightarrow{i} K_{G_1} \xrightarrow{i} K_{G_1^{\epsilon_1+\epsilon}}$$
Persistence to the rescue

- We can get a good estimate of $\epsilon$
- Consider the following filtration:

$$K_{G_1^\epsilon} \xrightarrow{i} K_{G_1} \xrightarrow{i} K_{G_{1+\epsilon}}$$

- If $[c] \neq 0$, $[c] \in K_{G_1^\epsilon}$ and $i^2([c]) \neq 0$, then $i_*([c])$ is non-contractible in $K_{G_1}$
Detection

Given the complex $K_{G_i}$

1. Compute a basis $Z$ for $\ker(\partial_1)$
Detection

Given the complex $K_{G_i}$

1. Compute a basis $Z$ for $\ker(\partial_1)$

2. $H_1(K_{G_i}) \neq 0$ if and only if $\exists$ a cycle $c \in Z$ such that $[c] \neq 0$ $(c \not\in B, B = \text{img}(\partial_2))$
1. compute a spanning tree $T$ on $G_1^\epsilon$
Computing cycle basis $Z$

1. compute a spanning tree $T$ on $G^c_1$
2. let $r$ be the root of $T$. Then for $e = (v_i, v_j)$, let $\gamma(T, e)$ be the algebraic representation of the cycle $(v_i - r - v_j - v_i)$
Computing cycle basis $Z$

1. compute a spanning tree $T$ on $G_1$

2. let $r$ be the root of $T$. Then for $e = (v_i, v_j)$, let $\gamma(T, e)$ be the algebraic representation of the cycle $(v_i - r - v_j - v_i)$

3. the set $Z = \{\gamma(T, e), e \notin T\}$ forms a basis for $\ker(\partial_1)$

Compute the integral function of a harmonic $y$ on the tree $T$. 

H Chintakunta
North Carolina State University
Distributed testing for contractibility
Computing dot products with harmonics

- Compute the integral function of a harmonic $y$ on the tree $T$.
Localization

“Divide and Conquer”

First, find a pair of nodes which are “far apart” from each other in the network.
Localization
“Divide and Conquer”

First, find a pair of nodes which are “far apart” from each other in the network.
Localization
“Divide and Conquer”

Find the set of nodes equidistant from the diameter nodes and connect the subgraph.
Summary

Distributed
- computing the cycle basis $Z$,
- detecting non-contractible cycles in $Z$,
- localizing by dividing network, and
- checking for persistence for existence guarantees.
Computation of a sparse basis for Homology
Select representatives for homologous classes

Sparsifying the complex

- Start with a non-contractible cycles in the basis $\mathbb{Z}$. 
Select representatives for homologous classes

Sparsifying the complex

- Start with a non-contractible cycles in the basis $\mathbb{Z}$.
- Select one cycle from each class of homologous cycles.
Select representatives for homologous classes

Sparsifying the complex

- Start with a non-contractible cycles in the basis $Z$.
- Select one cycle from each class of homologous cycles.
- Denote the above set of cycles by $\mathcal{P}$. 

H. Chintakunta
North Carolina State University
Reduction using harmonics to obtain a basis set

- Compute $|P|$ number of harmonics, $Y$. 

H Chintakunta
North Carolina State University
Reduction using harmonics to obtain a basis set

- Compute $|P|$ number of harmonics, $Y$.
- Compute the matrix $\Phi$ where $\Phi_{ij} = \langle Y_i, P_j \rangle$. 
Reduction using harmonics to obtain a basis set

- Compute $|P|$ number of harmonics, $Y$.
- Compute the matrix $\Phi$ where $\Phi_{ij} = \langle Y_i, P_j \rangle$.
- Column reduce $\Phi$. The non-zero columns correspond to the basis elements.
Generalizations

- Generalization to higher dimensions: key issue is efficient computation of a cycle basis.
- Generalization to persistence.
Other techniques for reducing a complex


References and Acknowledgements


- sponsored by Defense Threat Reduction Agency (DTRA).

- Many thanks to all the members of the VISSTA group at NCSU for productive discussions.
Thank you for your attention!!