A Robust Boundary Detection Algorithm Based on Connectivity Only for 3D Wireless Sensor Networks

University of Louisiana at Lafayette

Hongyu Zhou, Hongyi Wu and Miao Jin

LOUISIANA
Lafayette

OUTLINE

- **Introduction**
  - **Background**
  - **Related work**
  - **Motivation**
- Coconut algorithm
- Simulation
- Conclusion

INTRODUCTION

- Background
  - Boundary is a key attribute that characterizes a sensor network for geographic exploration and monitoring tasks
  - 3D sensor network has attracted increasing interests recently, e.g. underwater exploration and atmospheric monitoring

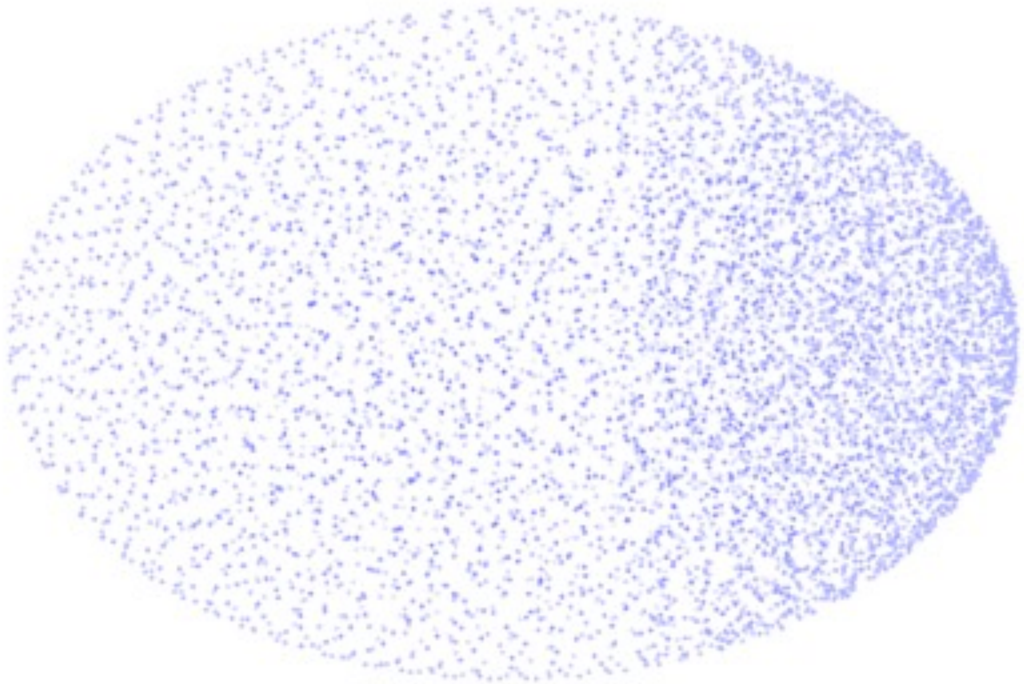
INTRODUCTION

- Related work
  - 2D boundary detection algorithms can't be applied in 3D wireless sensor networks
  - In 3D WSN:
    - Geometry-based approaches (Fit-ball, UNFOLD)
    - Topology-based schemes (CABET)

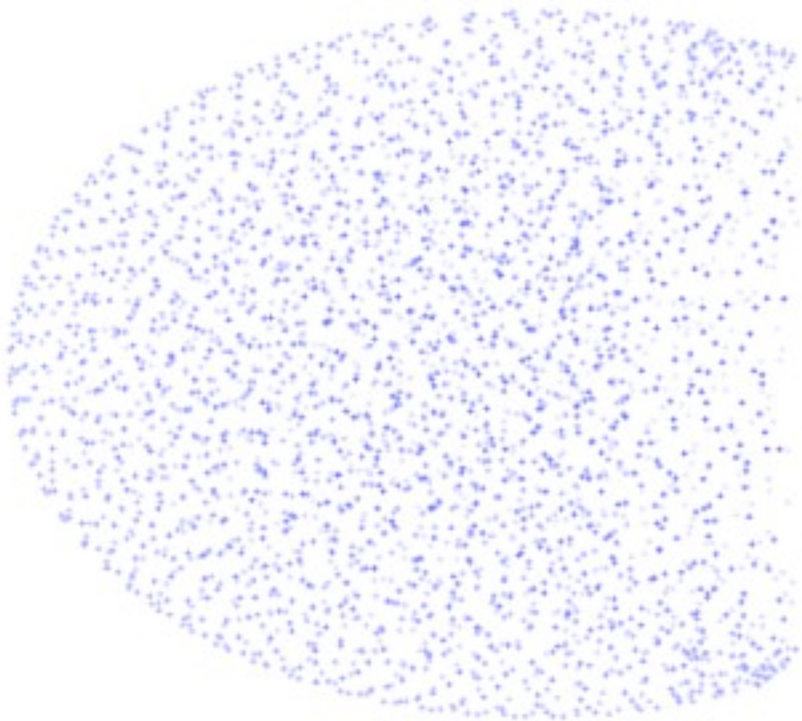
LOUISIANA Lafayette

INTRODUCTION

- Motivation:
  - Require connection only and robust with node density



Topology (non-uniform)



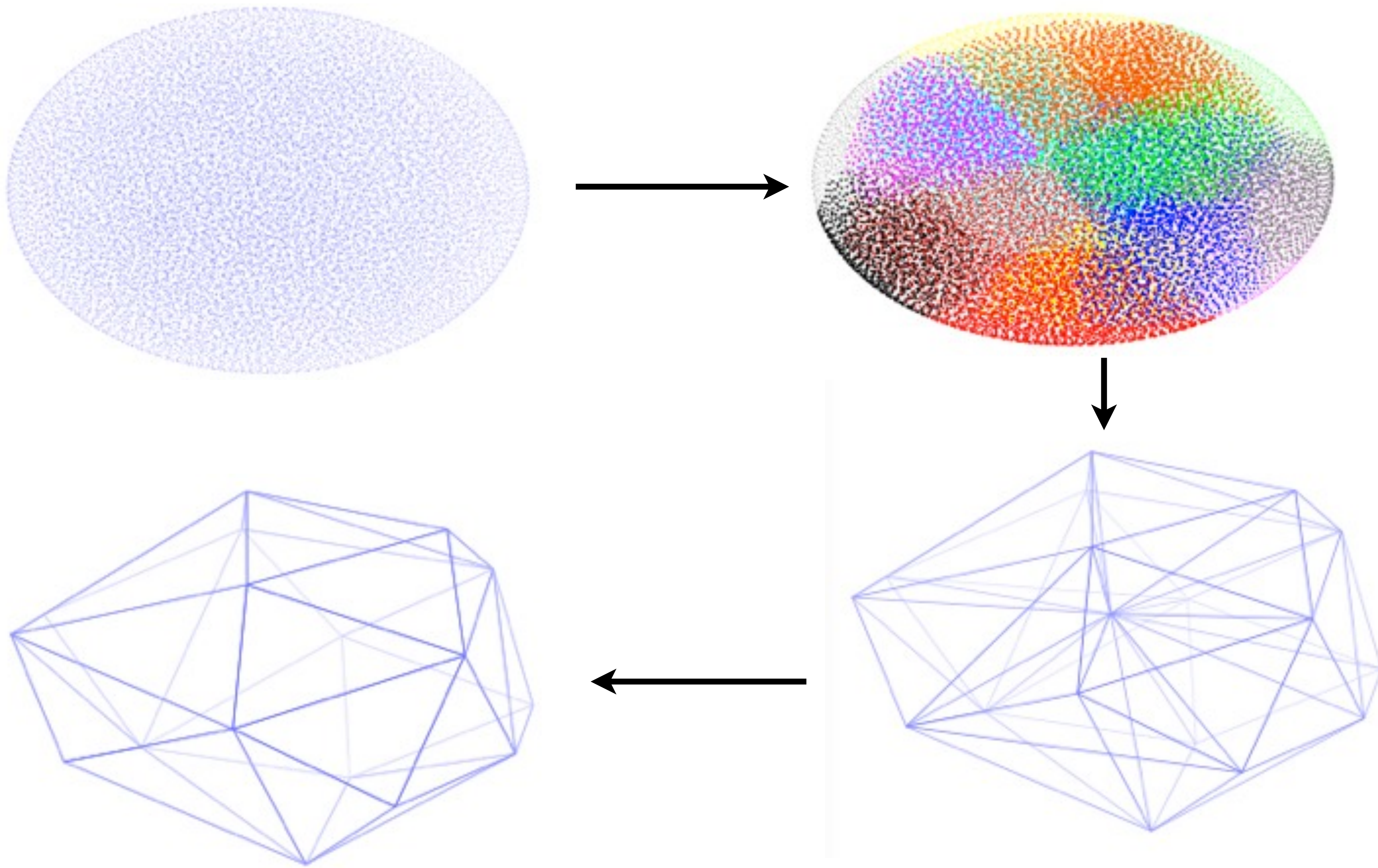
Boundary nodes by CABET

OUTLINE

- Introduction
- **Coconut algorithm**
  - **Coarse Boundary Surface Construction**
  - **Surface Sealing and Internal Hollowing**
  - **Boundary Refinement**
- Simulation
- Conclusion

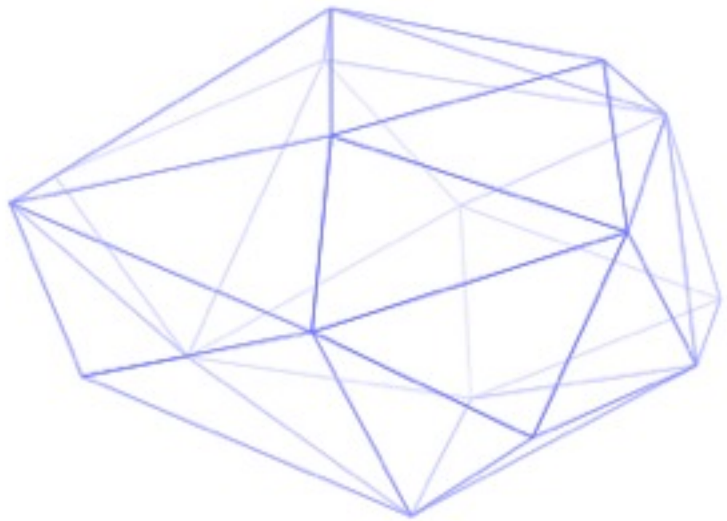
COCONUT ALGORITHM

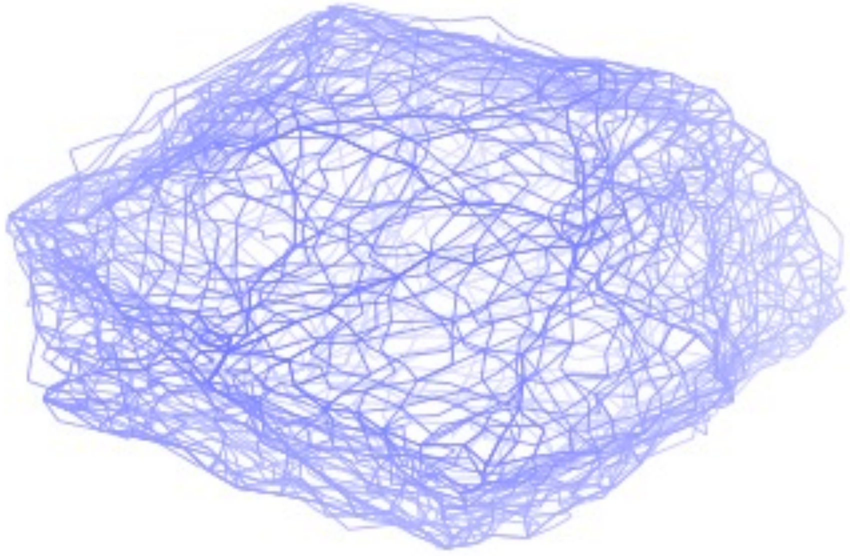
- Coarse Boundary Surface Construction

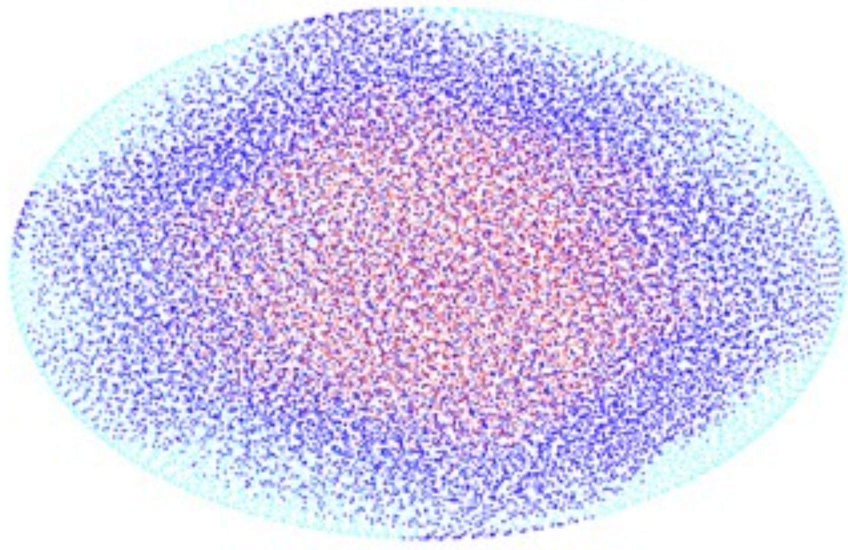


COCONUT ALGORITHM

- Surface Sealing and Internal Hollowing

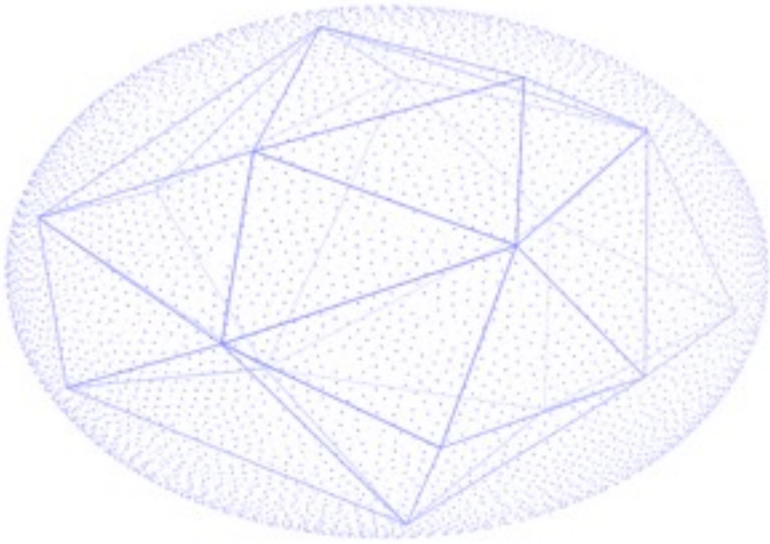


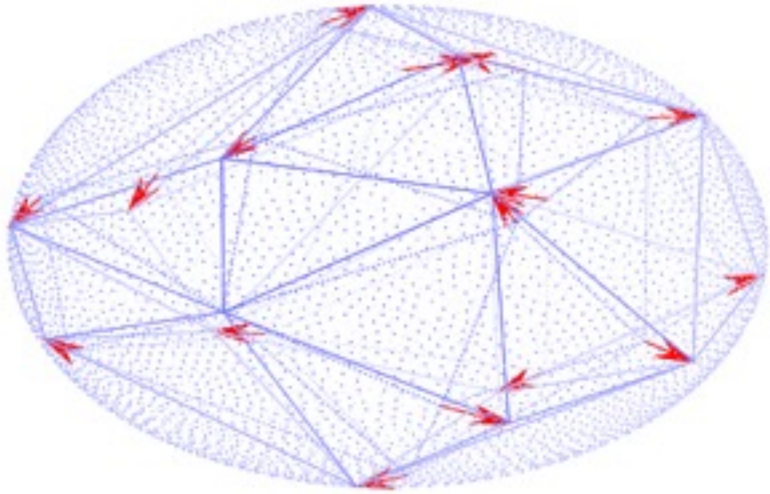


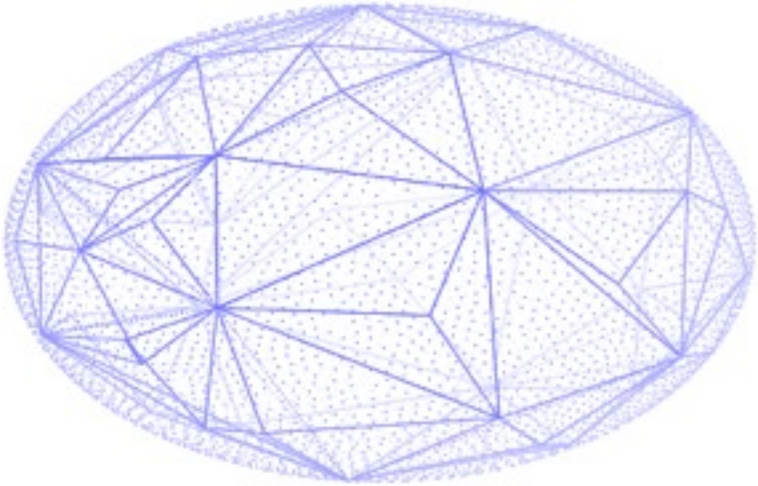


COCONUT ALGORITHM

- Boundary Refinement







LOUISIANA
Lafayette

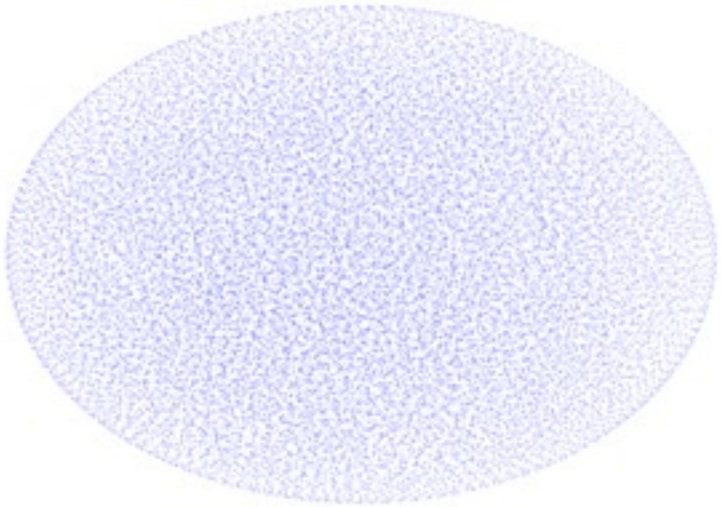
Coconut Algorithm

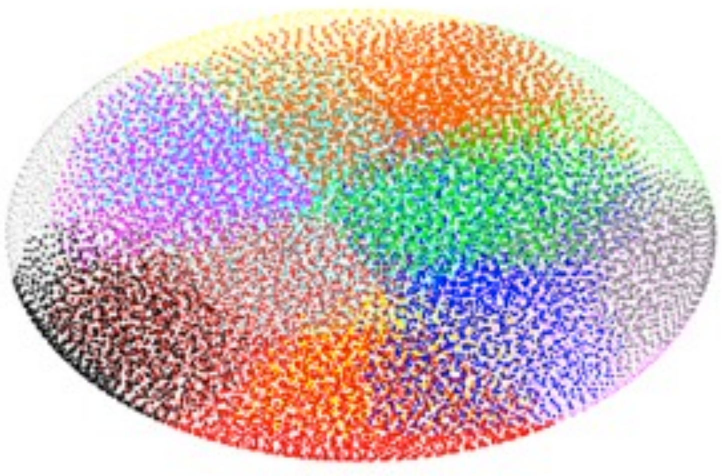
- **Coarse Boundary Surface Construction**
  - **Voronoi cell construction**
  - **Tetrahedron construction**
  - **Triangular surface construction**
- Surface Sealing and Internal Hollowing
- Boundary Refinement

LOUISIANA
Lafayette

COCONUT ALGORITHM

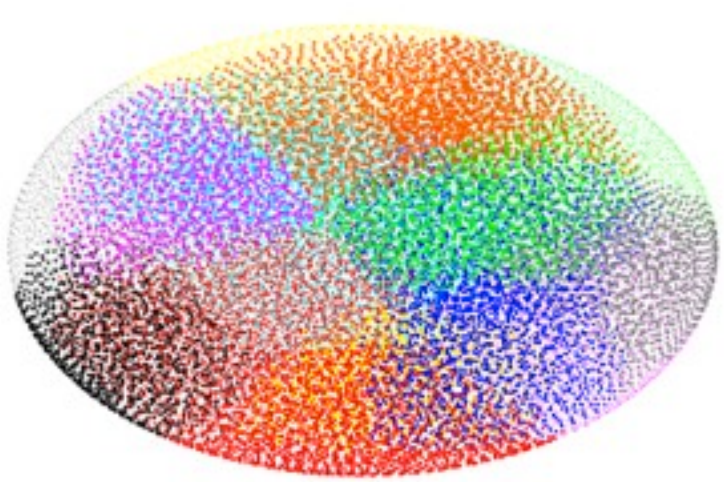
- Coarse Boundary Surface Construction
  - **Voronoi cell construction**
    - Random select landmarks, but every two of them are at least K hops away (k = 6~8)
    - Every other nodes will be associated with the closest landmark to it
    - Every landmark and its associated nodes form a voronoi cell

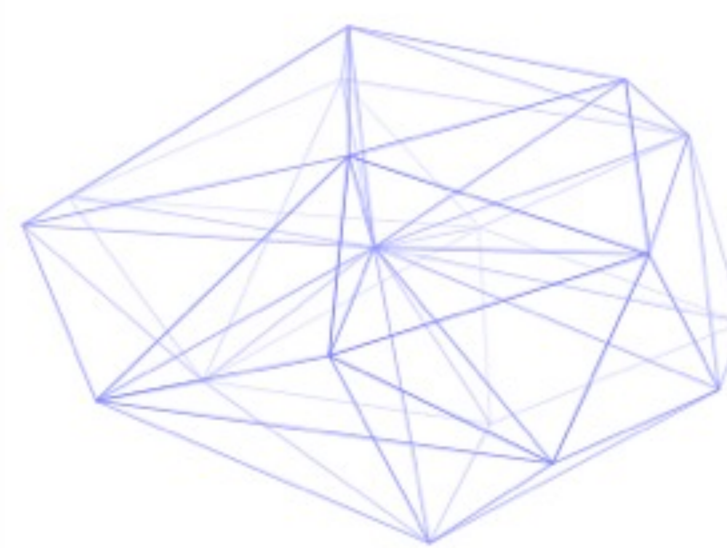




COCONUT ALGORITHM

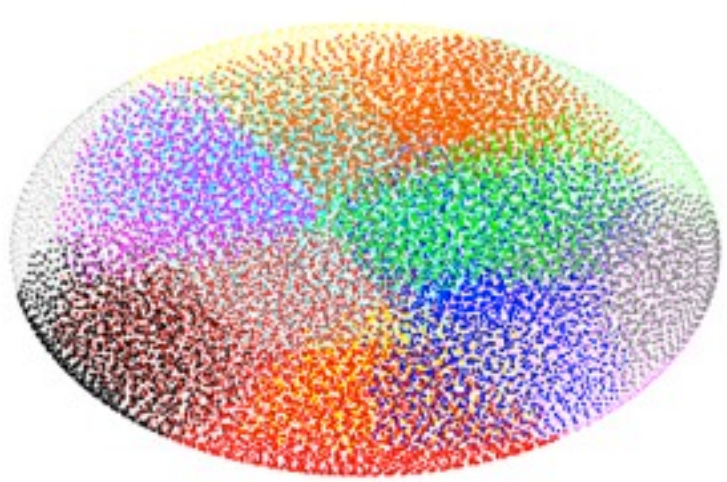
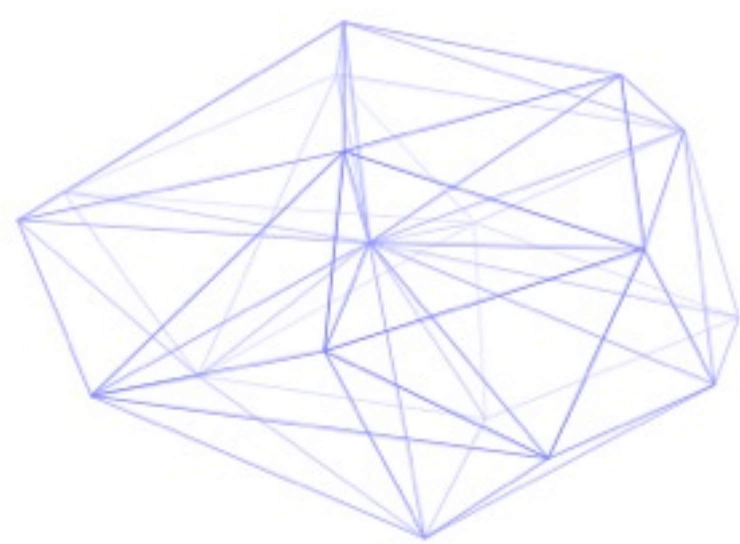
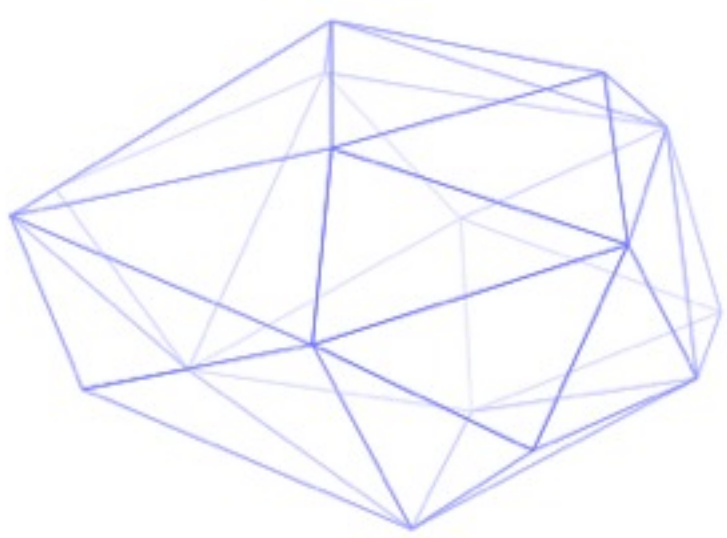
- Coarse Boundary Surface Construction
  - Voronoi cell construction
  - **Tetrahedron construction**
    - If there is a shortest path between two landmarks and all the nodes on the path are belong to these two landmarks, these two landmarks are connected





COCONUT ALGORITHM

- Coarse Boundary Surface Construction
  - Voronoi cell construction
  - Tetrahedron construction
  - **Triangular surface construction**
    - Surface triangle is shared by one tetrahedron
    - Inner triangle is shared by two tetrahedra
    - Landmarks are divided into two kinds: inner and surface landmarks

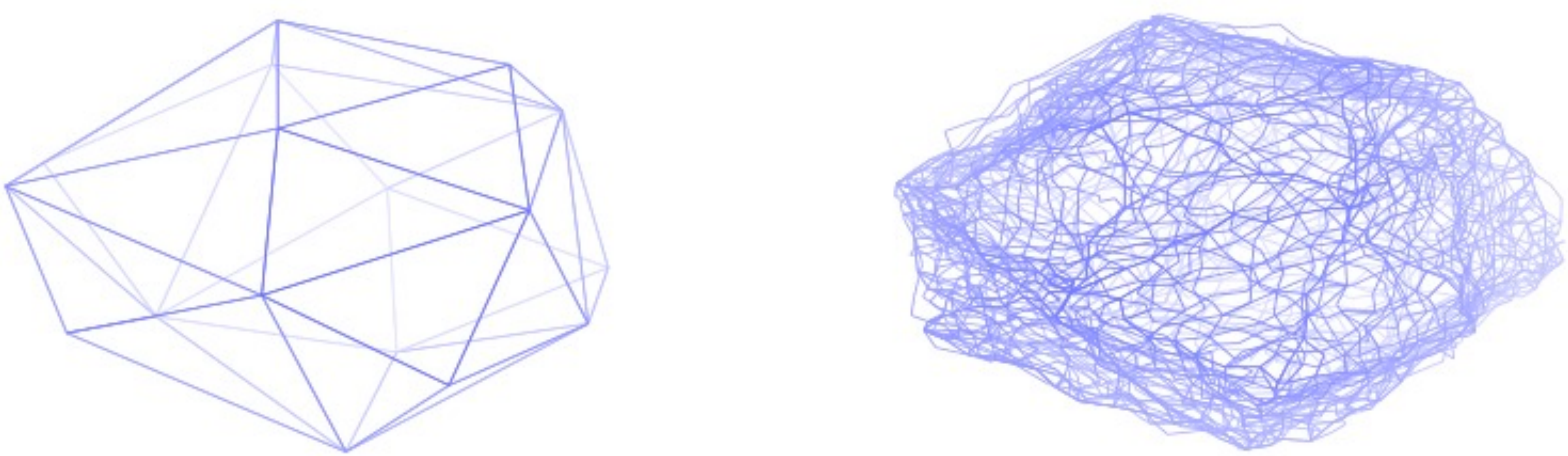




COCONUT ALGORITHM

- Coarse Boundary Surface Construction
- **Surface Sealing and Internal Hollowing**
  - **Surface Sealing**
  - **Internal Hollowing**
- Boundary Refinement

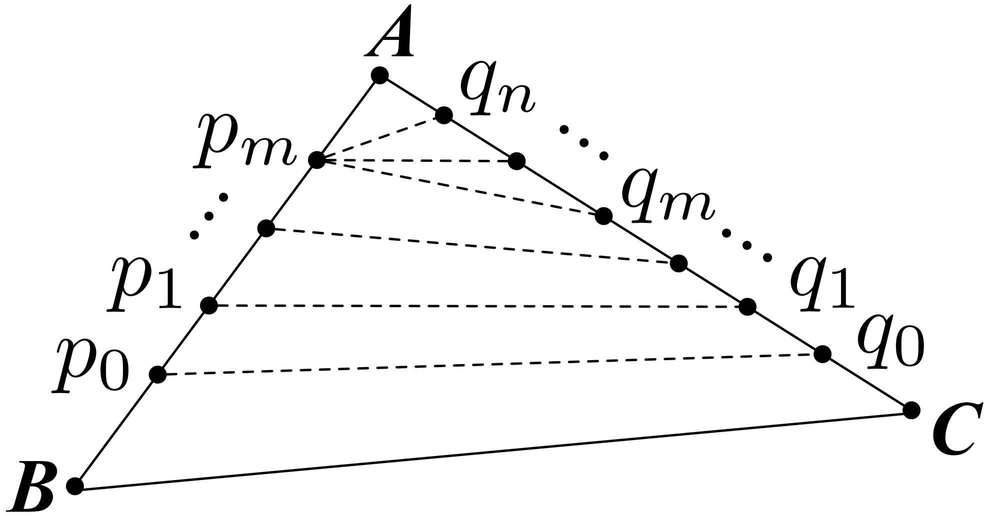
COCONUT ALGORITHM

- **Surface Sealing** and Internal Hollowing
  - Surface sealing: differentiate the inner nodes and outer nodes by surface
  - How can we build a skeleton of surface?



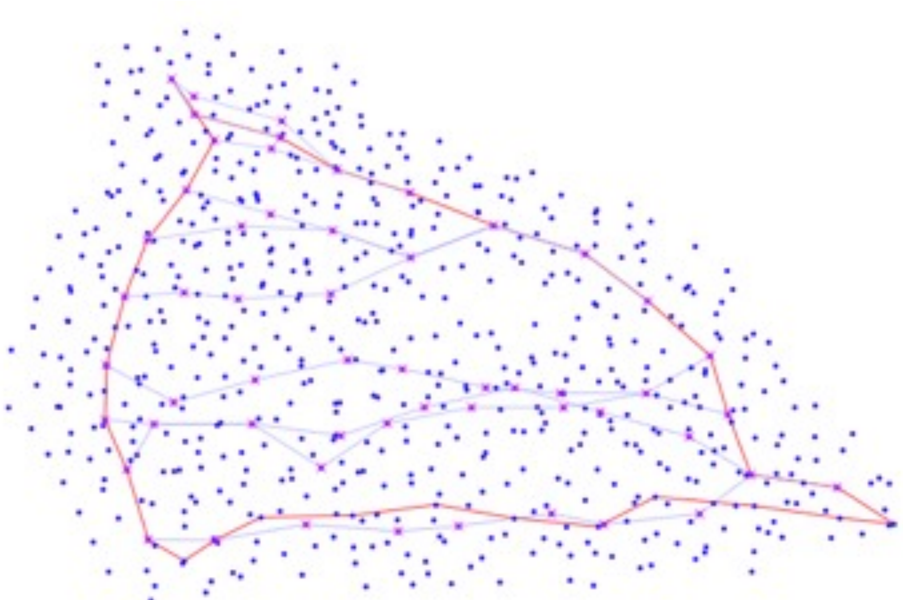
COCONUT ALGORITHM

- **Surface Sealing** and Internal Hollowing
  - Build shortest paths between two nodes on edges of surface triangle, ***Γ* (*p*, *q*)**
  - There are multiple shortest paths between given two nodes

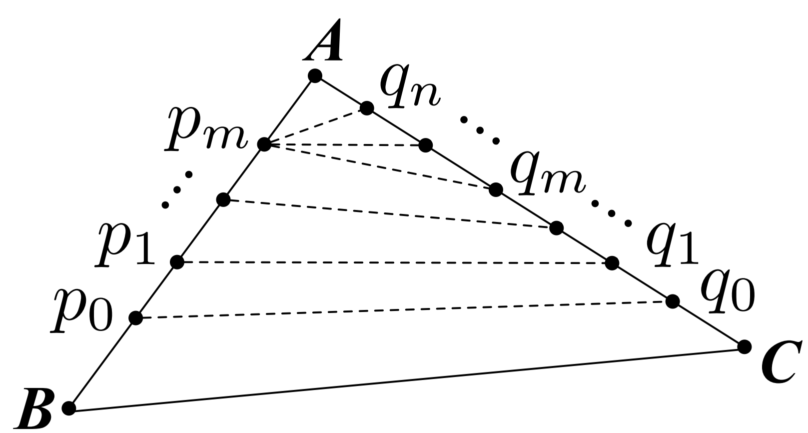


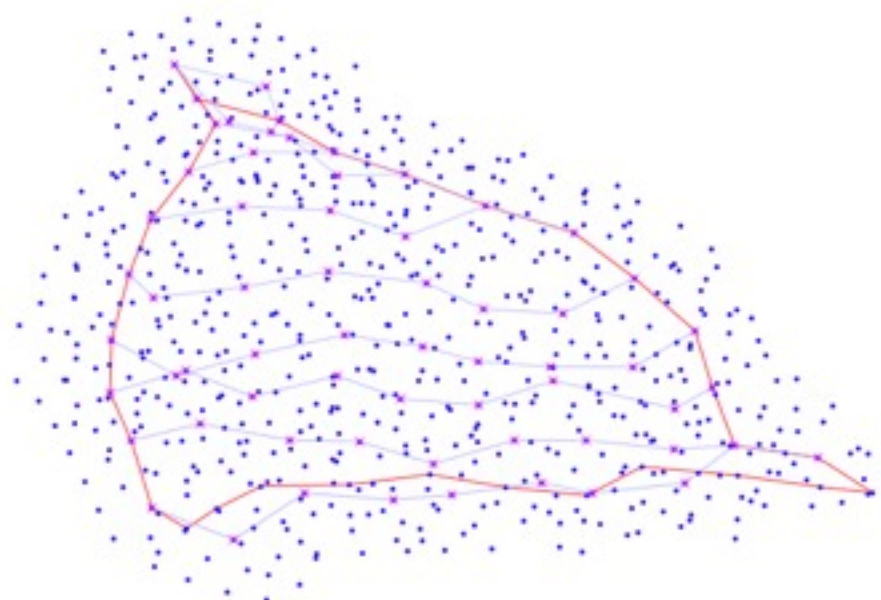
COCONUT ALGORITHM

- **Surface Sealing** and Internal Hollowing
  - How to choose the right shortest path?
  - Start from ***Γ*** **(*B,C*)**, make sure nodes in ***Γ*** **(*p_0*, *q_0*)** are the ***1-hop*** neighbor of ***Γ*** **(*B,C*)**



Random selection

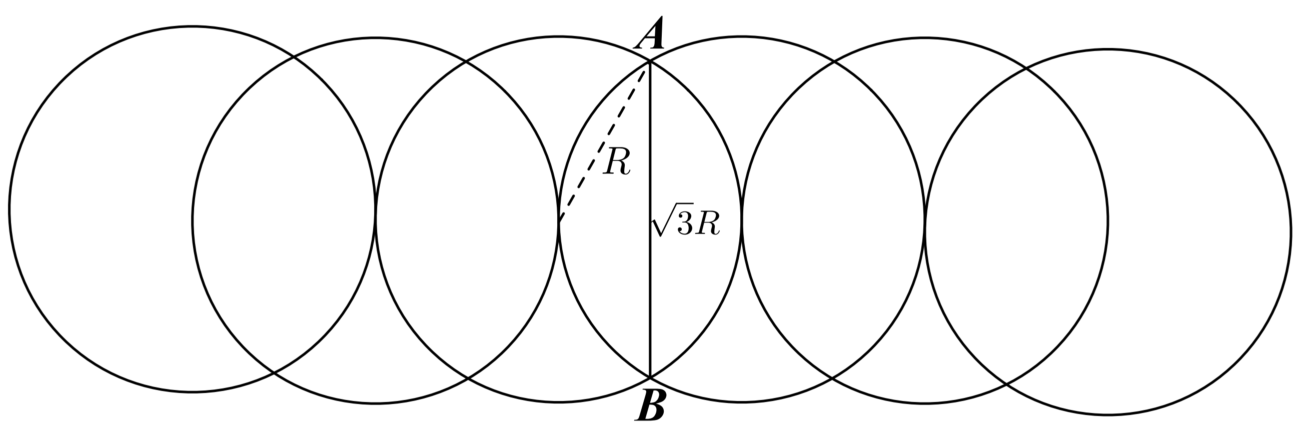




1-hop neighbor selection

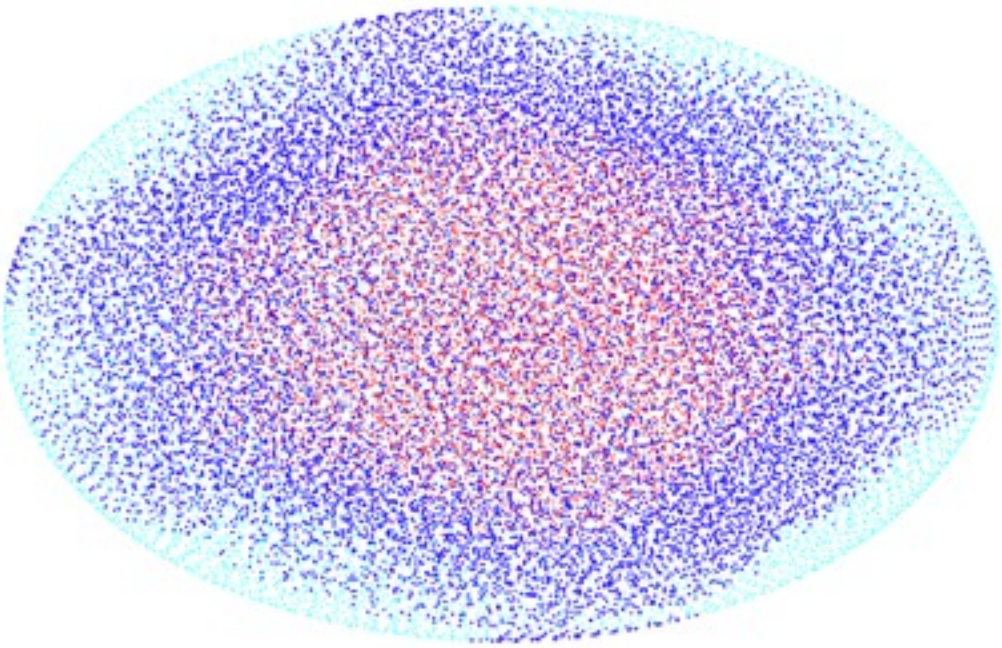
COCONUT ALGORITHM

- **Surface Sealing** and Internal Hollowing
  - All the shortest paths (***Γ-nodes***) are not enough
  - The ***Γ-nodes*** and their one-hop neighbors form the sealed boundary face (***S-nodes)***.



COCONUT ALGORITHM

- Surface Sealing and **Internal Hollowing**
  - After Surface Sealing, sensor nodes will be grouped into three types: ***internal nodes*** (or ***I-nodes***), ***surface nodes*** (or ***S-nodes***), and ***outside nodes***(or ***O-nodes***)
  - ***I-nodes*** are not boundary nodes that can be removed



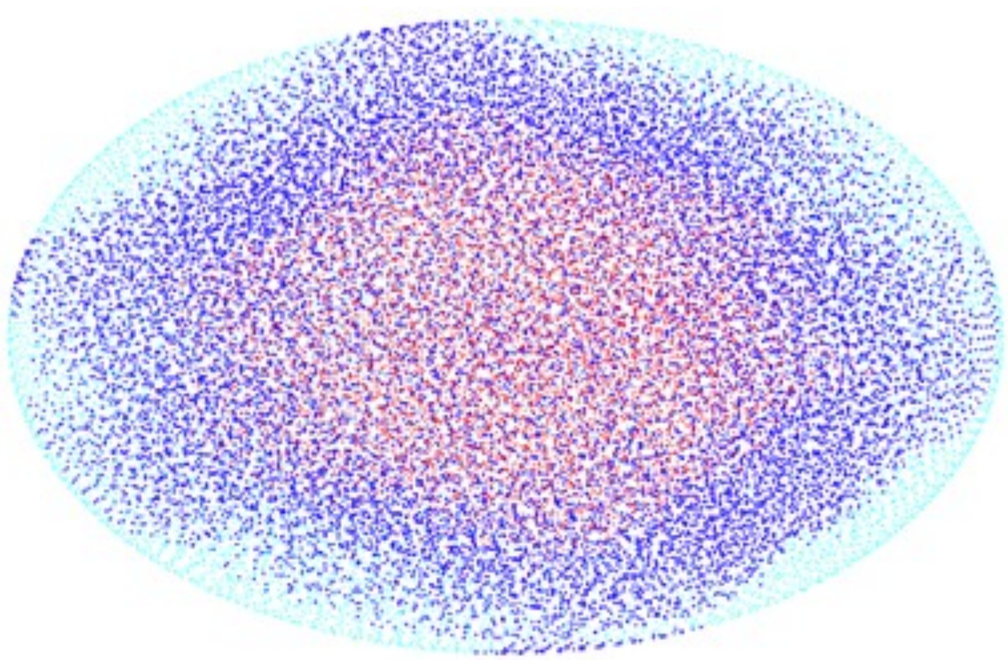
red: ***I-nodes;*** blue: ***S-nodes;*** Aqua: ***O-nodes***

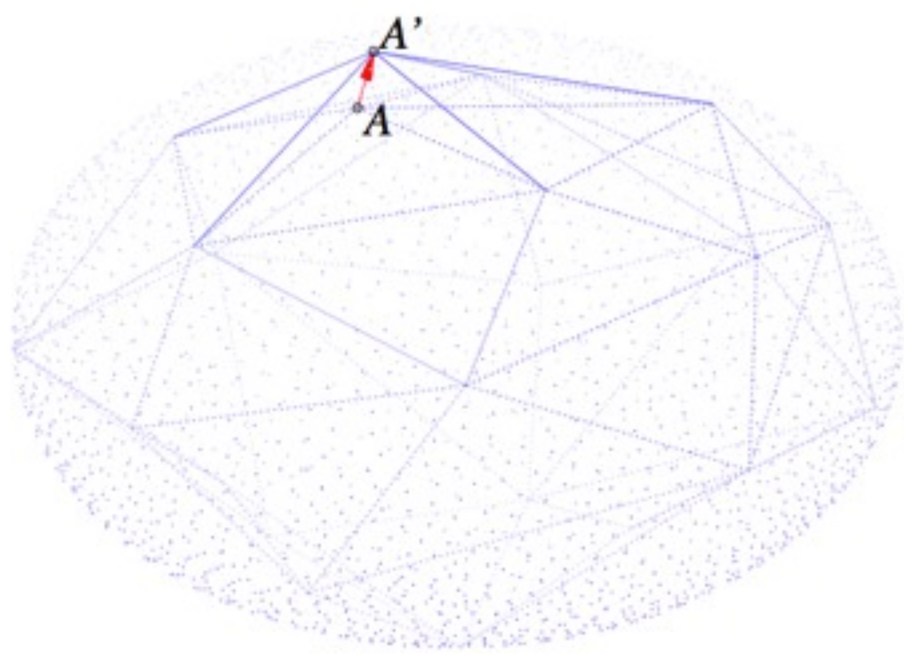
COCONUT ALGORITHM

- Coarse Boundary Surface Construction
- Surface Sealing and Internal Hollowing
- **Boundary Refinement**
  - **Boundary Landmark Expansion**
  - **Boundary Face Splitting**
  - **Boundary Surface Thinning**
- Algorithm Complexity

COCONUT ALGORITHM

- Boundary Refinement
  - **Boundary Landmark Expansion:** push boundary landmark to the surface of network
  - ***A'*** and ***A*** are in the same cell and ***A'*** *is* a ***O-nodes***

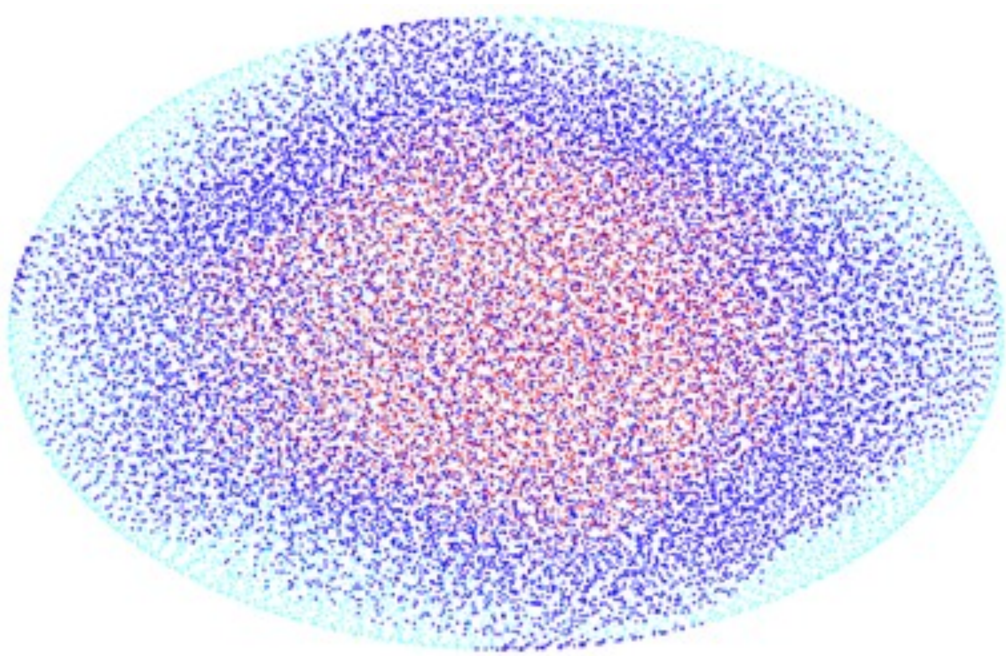


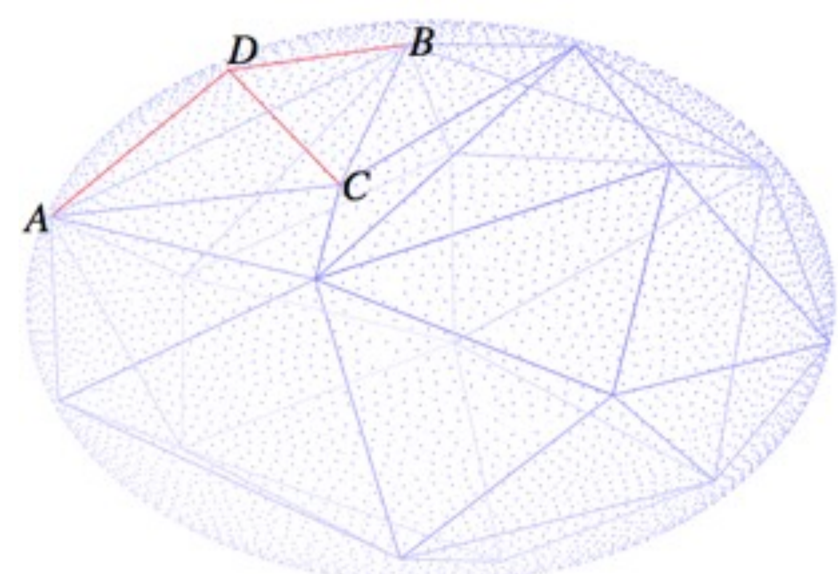


red: ***I-nodes;*** blue: ***S-nodes;*** Aqua: ***O-nodes***

COCONUT ALGORITHM

- Boundary Refinement
  - **Boundary Face Splitting**: push boundary faces to the surface of the network
  - If there exists an ***O-node D*** that has equal hop distance (or differed by one) to three landmarks ***A, B, C*** of a triangular boundary face, $\triangle$***ABC*** is thus replaced by $\triangle$***ACD***, $\triangle$***ABD*** and $\triangle$***BCD***

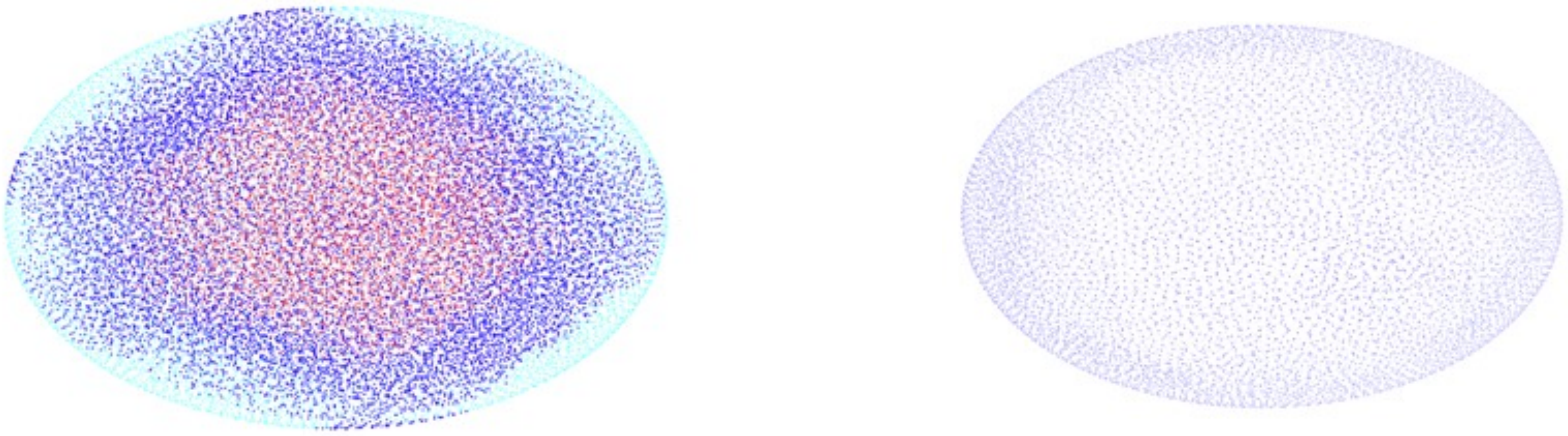




red: ***I-nodes;*** blue: ***S-nodes;*** Aqua: ***O-nodes***

COCONUT ALGORITHM

- Boundary Refinement
  - After Landmark Expansion and Boundary Face Splitting, there are only few or no ***O-nodes*** left
  - **Boundary Surface Thinning**: remove the extra nodes in the boundary surface nodes (***S-nodes+O-nodes***) with at most two-hop thickness



red: ***I-nodes;*** blue: ***S-nodes;*** Aqua: ***O-nodes***

COCONUT ALGORITHM

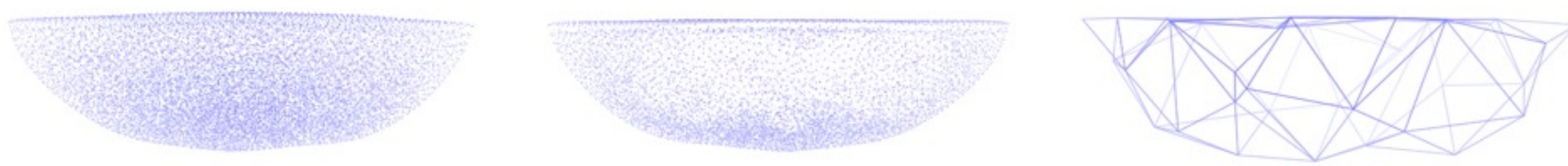
- Computational and Communication Complexity

	Computational	Communication
Surface Construction	$O(n)$	$O(n)$
Surface Sealing & Internal Hollowing	$O(m^2)$	$O(m^2)$
Boundary Surface Thinning	$O(k^2)$	$O(k^2)$
$k \ll m \ll n$		
$\boldsymbol{k}$: # of landmark nodes; $\boldsymbol{m}$: # of Γ- nodes of one face; $\boldsymbol{n}$: # of nodes in the network		

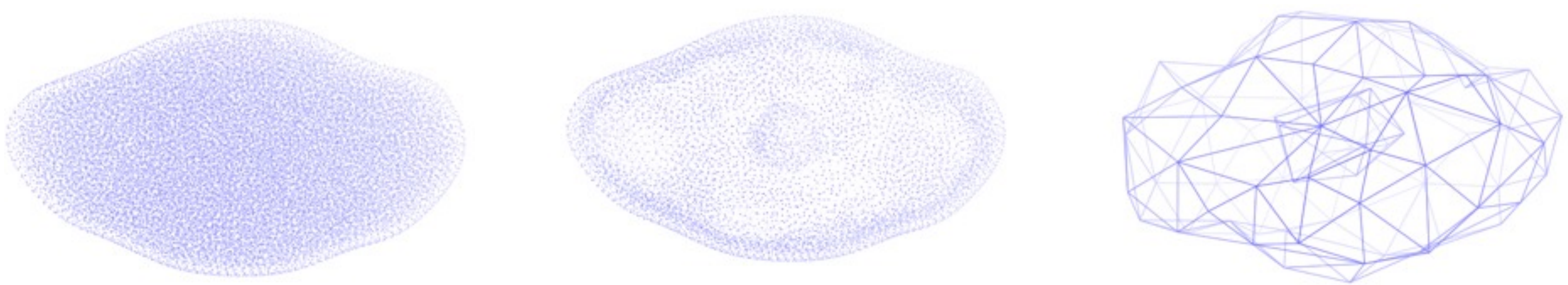
The over all computational complexity and communication cost is dominant by $O(n)$

SIMULATION

- Lake Model:



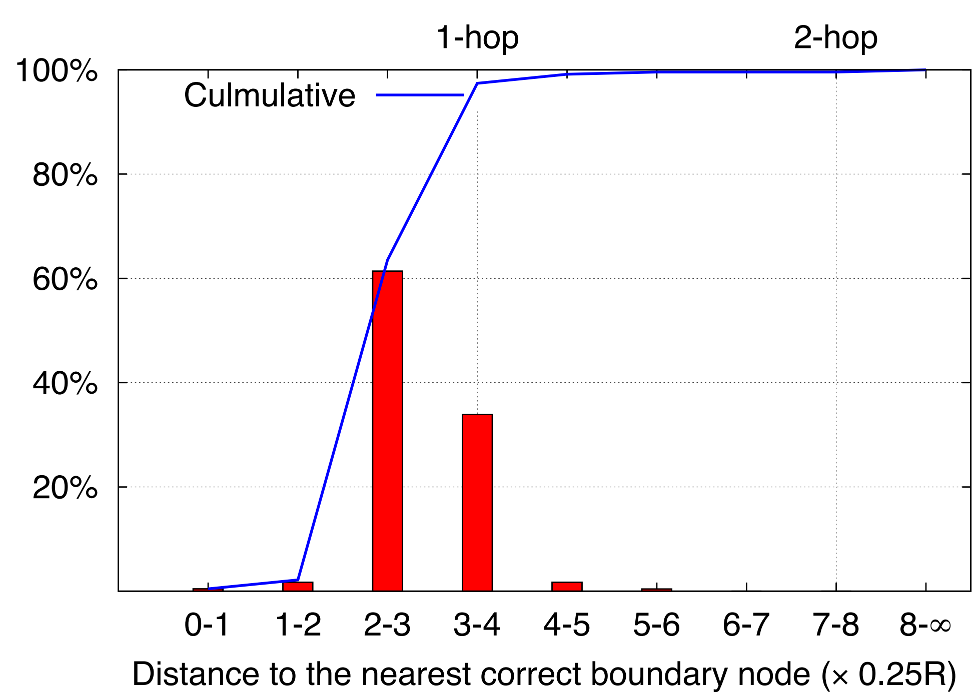
- Cloud Model with a hole:



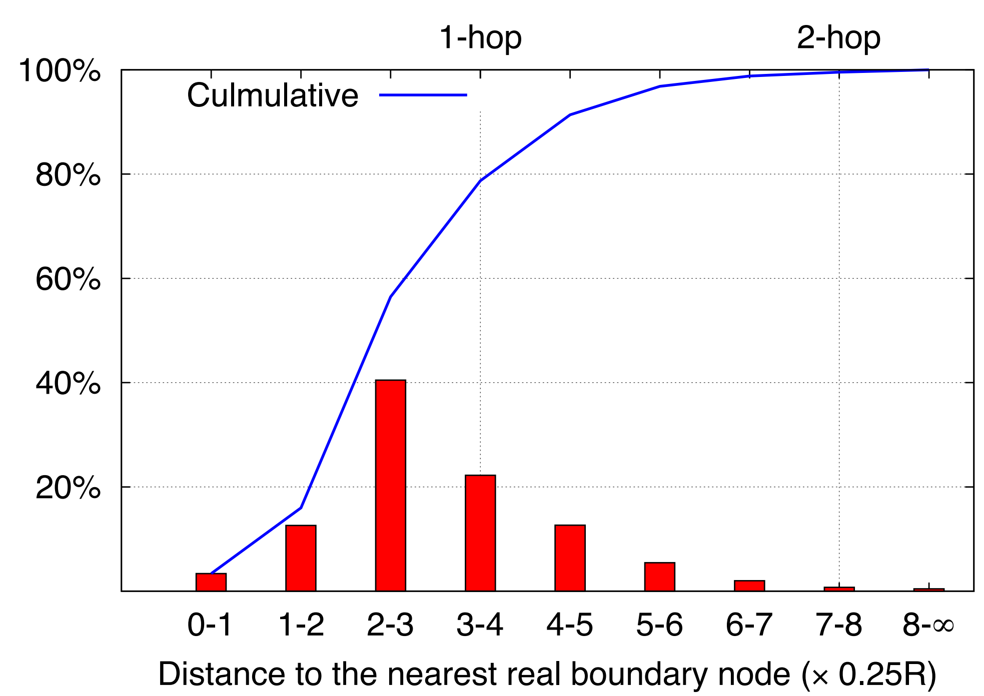
SIMULATION

TABLE II
BOUNDARY DETECTION RESULTS.

Model	Correct	Missing	Mistaken
Model 1	99.9%	0.1%	47.9%
Model 2	99.9%	0.1%	147.1%
Model 3	98.1%	1.9%	174.9%
Model 4	97.0%	3.0%	54.1%



(a) Missing node distribution.



(b) Mistaken node distribution

CONCLUSION

- We have proposed a distributed boundary detection algorithm, dubbed Coconut, for 3D wireless sensor networks
- The proposed Coconut algorithm is a connectivity-based approach
- It has no constraint on communication models, and it's robust about the node density

LOUISIANA Lafayette