Length-Based Anchor-Free Localization in a Fully Covered Sensor Network

Buddhadeb Sau[1] Krishnendu Mukhopadhyaya[2]

[1]Department of Mathematics
Jadavpur University, Kolkata, India
bsau@math.jdvu.ac.in

[2]Advanced Computing and Microelectronics Unit
Indian Statistical Institute, Kolkata, India
krishnendu@isical.ac.in

First International Conference on COMmunication Systems and NETworkS (COMSNETS), January 5-10, 2009, Bangalore, India

Outline

1. **Introduction**
2. **Basic Model and Problem Statement**
3. **Localization under the Basic Model**
4. **Restricted Model**
   - Localization under the Restricted Model
   - Correctness and Complexity
   - Simulation Studies
5. **Conclusion**

Outline

Introduction

Applications of WSN

WSN has wide-ranging applications

- habitat monitoring
- battlefield serveillance
- monitoring forest-fire
- disaster management etc.

Introduction
Localization

Why is the localization a major issue in WSN?

- Sensors may be deployed randomly,
- positions of the sensors are completely unknown,
- positions are extremely important in routing, security etc.

Introduction
Localization

Why is the localization a major issue in WSN?

- Sensors may be deployed randomly,
- positions of the sensors are completely unknown,
- positions are extremely important in routing, security etc.

Definition

Finding the positions of sensors is known as localization in WSN.

Outline

Basic Model

- Field of interest is convex.
- Sensors are identical.
- Sensing range $\rightarrow s$. Communication range $\rightarrow r$.
- No barrier present in the WSN field.
- Sensors are in general position.
- Distances are measured accurately.
- For (s_i, s_j),
  if $dist(s_i, s_j) \leq r$, the value of $dist(s_i, s_j)$ is known.
  If $dist(s_i, s_j) > r$, the value of $dist(s_i, s_j)$ is unknown.

Basic Model

Graph Model of WSN

Let WSN contain n sensors $s_1, s_2, \cdots, s_n$. We construct an undirected edge-weighted graph $G = (V, E, w)$ as follows :

- Each vertex in V represents a sensor.
- $(s_i, s_j) \in E$ iff the distance between s_i and s_j is known.
- $\forall (s_i, s_j) \in E, \quad w(s_i, s_j) = dist(s_i, s_j)$.

Basic Model

Graph Model of WSN

Let WSN contain n sensors $s_1, s_2, \cdots, s_n$. We construct an undirected edge-weighted graph $G = (V, E, w)$ as follows :

- Each vertex in V represents a sensor.
- $(s_i, s_j) \in E$ iff the distance between s_i and s_j is known.
- $\forall (s_i, s_j) \in E, \quad w(s_i, s_j) = dist(s_i, s_j)$.

Goal

To localize the sensors without using any anchor.

Basic Model

Graph Model of WSN

Let WSN contain n sensors $s_1, s_2, \cdots, s_n$. We construct an undirected edge-weighted graph $G = (V, E, w)$ as follows :

- Each vertex in V represents a sensor.
- $(s_i, s_j) \in E$ iff the distance between s_i and s_j is known.
- $\forall (s_i, s_j) \in E, \quad w(s_i, s_j) = dist(s_i, s_j)$.

Goal

To localize the sensors without using any anchor.

Definition

A realization of $G = (V, E, w)$ in $\mathbb{R}^m$ (Euclidean space of dimension m) is an assignment of coordinates $(x_1, \ldots, x_m)$ to the vertices so that weight of an edge represents the Euclidean distance between the points incident on the edge.

Problem Statement

We are interested only in $\mathbb{R}^2$.

Problem

Given an edge-weighted undirected graph $G = (V, E, w)$, our objective is to find all possible realizations of the graph G in $\mathbb{R}^2$ such that,

$$dist(s_i, s_j) \quad > \quad r, \;\; \forall \, (s_i, s_j) \notin E.$$

Problem Statement

We are interested only in $\mathbb{R}^2$.

Problem

Given an edge-weighted undirected graph $G = (V, E, w)$, our objective is to find all possible realizations of the graph G in $\mathbb{R}^2$ such that,

$$dist(s_i, s_j) \quad > \quad r, \ \ \forall\, (s_i, s_j) \notin E.$$

Sensor realization

A solution to this problem will be referred to as a sensor realization of G.
The sensor realization of the graph G may be denoted by $\mathcal{R} = (G, \Phi)$, where $\Phi : V \rightarrow \mathbb{R}^2$ and $\Phi(v)$ represents the corresponding realization of the vertex, $v \in V$.

Problem Statement

Is there any Sensor realization of the WSN at all?

- At least one sensor realization exists, since the distance information is coming from an actual deployment of sensors

Outline

Localization under the Basic Model

In general

- WSN may have multiple Sensor realizations.
- Actually, any sensor realization may be flipped, rotated and/or translated (shifting origin), (like any rigid body), to get another sensor realization.
- We are interested only in realizations which are structurally different.

Localization under the Basic Model

In general

- WSN may have multiple Sensor realizations.
- Actually, any sensor realization may be flipped, rotated and/or translated (shifting origin), (like any rigid body), to get another sensor realization.
- We are interested only in realizations which are structurally different.

Our main goal is to find only:

- A set of sensor realizations of a graph from which any other sensor realization may be obtained with the operation shifting, flipping and/or rotation suitably.

Localization under the Basic Model

Rigidity, Flipping, Rotation and Shifting

- From here onwards flip, rotation and shift in a sensor realization of a WSN graph, we mean a part of the sensor realization is flipped, rotated or shifted, giving us a new sensor realization while rest of the realization remains fixed.

Localization under the Basic Model

Rigidity, Flipping, Rotation and Shifting

Definition

An edge-weighted graph $G = (V, E, w)$ is called rigid, if the distance between any pair of sensors remains same in every sensor realization.

Localization under the Basic Model
Rigidity, Flipping, Rotation and Shifting

Definition

An edge-weighted graph $G = (V, E, w)$ is called rigid, if the distance between any pair of sensors remains same in every sensor realization.

Theorem

If two rigid bodies B_1 and B_2, in a sensor realization of a graph $G = (V, E, w)$ share three or more common vertices, $B_1 \cup B_2$ (alongwith all edges between them) forms a rigid body.

Localization under the Basic Model

Rigidity, Flipping, Rotation and Shifting

Definition

An edge-weighted graph $G = (V, E, w)$ is called rigid, if the distance between any pair of sensors remains same in every sensor realization.

Theorem

If two rigid bodies B_1 and B_2, in a sensor realization of a graph $G = (V, E, w)$ share three or more common vertices, $B_1 \cup B_2$ (alongwith all edges between them) forms a rigid body.

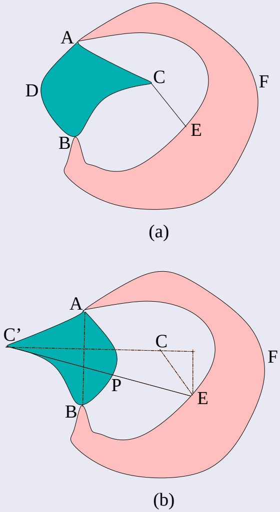
Definition

If two rigid graphs share exactly one vertex, then one of them may be rotated, around the common vertex, keeping the other fixed. Such a vertex will be called a joint.

Localization under the Basic Model
Rigidity, Flipping, Rotation and Shifting

Definition

If two rigid graphs share exactly two vertices, rotation about these vertices is no longer possible, but one may be flipped, about the line joining the two common vertices, keeping the other fixed. These two vertices will be called a flip joint or simply flip and each vertex as a flip vertex.



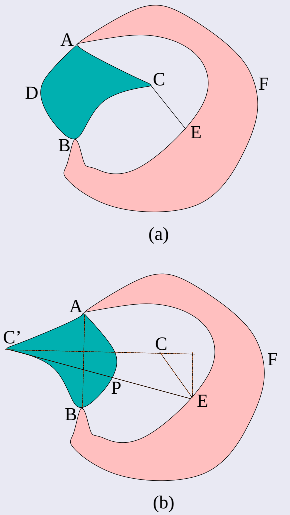
Localization under the Basic Model
Rigidity, Flipping, Rotation and Shifting

Definition

If two rigid graphs share exactly two vertices, rotation about these vertices is no longer possible, but one may be flipped, about the line joining the two common vertices, keeping the other fixed. These two vertices will be called a flip joint or simply flip and each vertex as a flip vertex.

Theorem

If two rigid subgraphs, B_1 and B_2, of a graph G share two common vertices and there is an edge connecting a vertex in B_1 to another vertex in B_2, $B_1 \cup B_2$ is rigid.



Outline

Restricted Model

Characteristics

1. The field $\mathcal{F}$ is sensing covered i.e., every point in $\mathcal{F}$ is in the sensing zone of at least one sensor.

Restricted Model

Characteristics

1. The field $\mathcal{F}$ is sensing covered i.e., every point in $\mathcal{F}$ is in the sensing zone of at least one sensor.

In VigilNet surveillance system

He et al. (*ACM Transactions on Sensor Networks*, 2006) uses full coverage of the field in VigilNet surveillance system. This may be required by many other application to guarantee a quality of service.

Restricted Model

Characteristics

1. The field $\mathcal{F}$ is sensing covered i.e., every point in $\mathcal{F}$ is in the sensing zone of at least one sensor.
2. $s \leq \frac{r}{2}$.

Restricted Model

Characteristics

1. The field $\mathcal{F}$ is sensing covered i.e., every point in $\mathcal{F}$ is in the sensing zone of at least one sensor.
2. $s \leq \frac{r}{2}$.

Is $s \leq \frac{r}{2}$ practical ?

Restricted Model

Characteristics

1. The field $\mathcal{F}$ is sensing covered i.e., every point in $\mathcal{F}$ is in the sensing zone of at least one sensor.
2. $s \leq \frac{r}{2}$.

Is $s \leq \frac{r}{2}$ practical ?

Sensor mote	Commu range, r	Sensing range, s	Range ratio, $\frac{s}{r}$
Mica2 (in VigilNet surveillance system)	$10m$	$3m$	$\frac{1}{3}$
Sensoria sGate (WSN platform by Sensoria Corporation)	$500m$	$< 100m$	$< \frac{1}{5}$
with Seismic sensors	$\geq 100m$	$50m$	$\leq \frac{1}{2}$

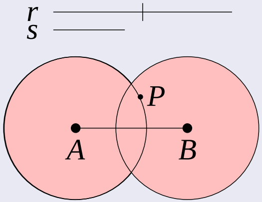
Outline

Localization under the Restricted Model

Full Coverage verification and Unique solution

Theorem

A WSN under the restricted model is fully covered if and only if for every sensor, any point on the boundary of its sensing zone is covered by at least another sensor.

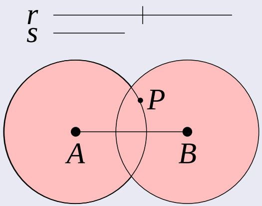


Localization under the Restricted Model

Full Coverage verification and Unique solution

Theorem

A WSN under the restricted model is fully covered if and only if for every sensor, any point on the boundary of its sensing zone is covered by at least another sensor.



Theorem

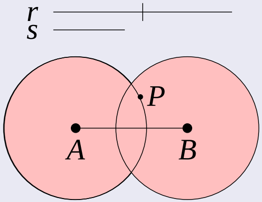
The edge-weighted graph $G = (V, E, w)$ under the above restricted model is rigid.

Localization under the Restricted Model

Full Coverage verification and Unique solution

Theorem

A WSN under the restricted model is fully covered if and only if for every sensor, any point on the boundary of its sensing zone is covered by at least another sensor.



Theorem

The edge-weighted graph $G = (V, E, w)$ under the above restricted model is rigid.
If $s > \frac{r}{2}$, then the graph may not be rigid even if all points are covered by some sensors.

Localization under the Restricted Model
Localization Algorithm

We propose a centralized localization technique.

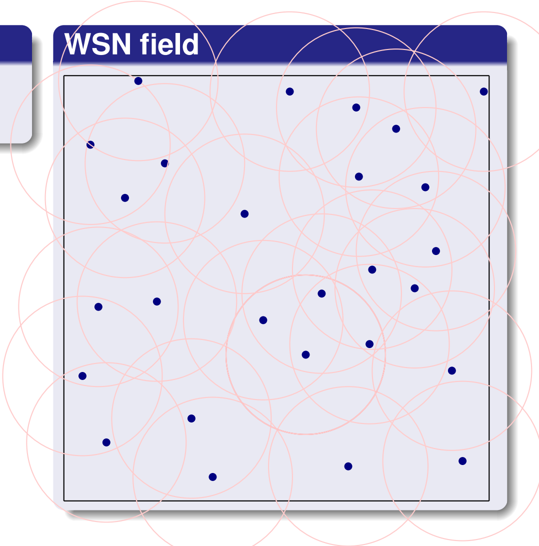
- Its worst case complexity time is $O(|E|)$, E is the edge set of G.
- Under random uniform deployment of nodes, the expected value of $|E|$ may be represented as $O(\frac{n^2}{R})$, where R is the field area.
- The expected time complexity of the proposed algorithm is significantly lower than $O(n^2)$, in particular for large scale WSN.

Localization under the Restricted Model
Localization Algorithm

Data structures:

- `located`, `locateIn2s` : stack;

WSN field



Localization under the Restricted Model
Localization Algorithm

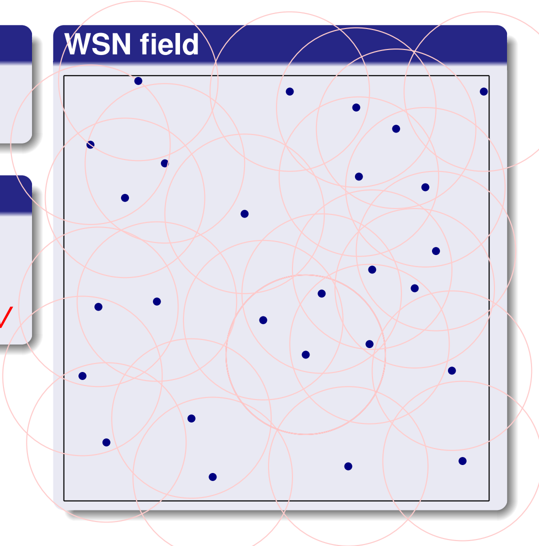
Data structures:

- `located`, `locateIn2s` : stack;

Initialization

- Set `located` $\leftarrow \emptyset$, `locateIn2s` $\leftarrow \emptyset$
- Set $\Phi(v) \leftarrow$ `null`, $\forall v \in V$

WSN field

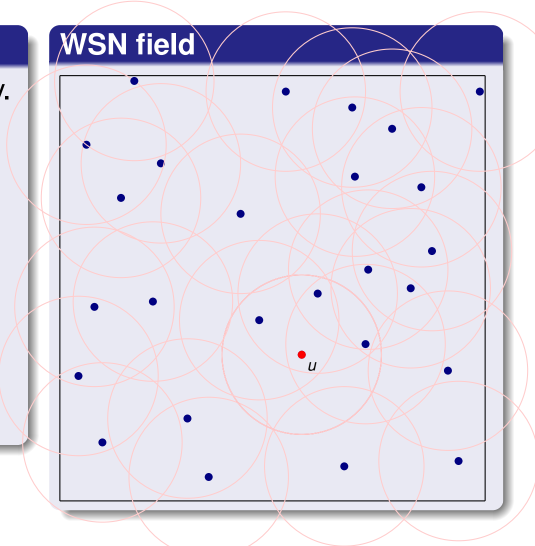


Localization under the Restricted Model
Localization Algorithm

Fixing coordinate system

- **Consider** $u \in V$ randomly.
- **Fix** $\Phi(u) \leftarrow (0, 0)$

WSN field

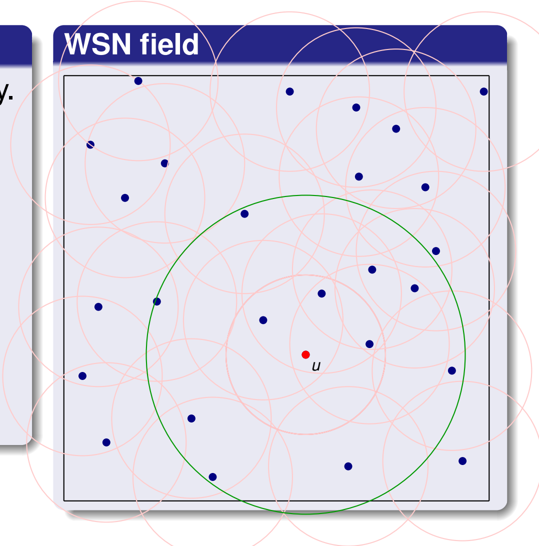


Localization under the Restricted Model
Localization Algorithm

Fixing coordinate system

- **Consider** $u \in V$ randomly.
- **Fix** $\Phi(u) \leftarrow (0,0)$
- **Let** $v \in C(u, 2s) - \{u\}$

WSN field

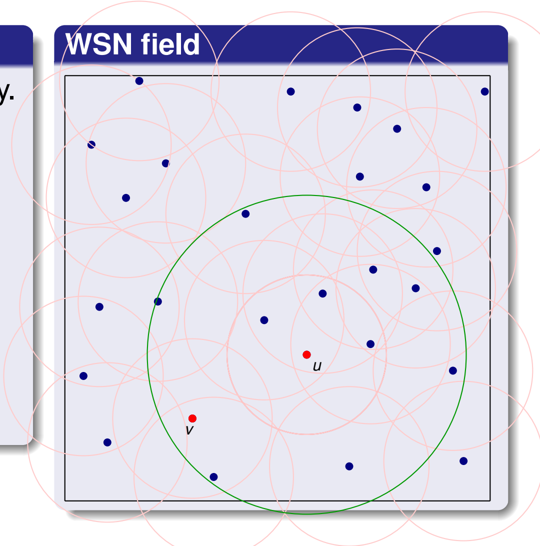


Localization under the Restricted Model
Localization Algorithm

Fixing coordinate system

- **Consider** $u \in V$ randomly.
- **Fix** $\Phi(u) \leftarrow (0, 0)$
- **Let** $v \in C(u, 2s) - \{u\}$

WSN field

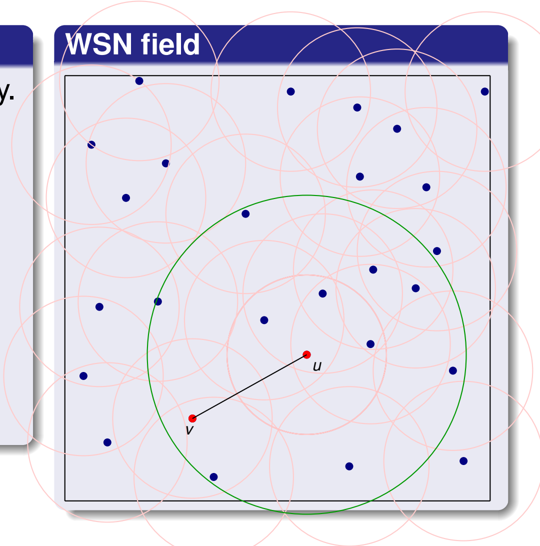


Localization under the Restricted Model
Localization Algorithm

Fixing coordinate system

- **Consider** $u \in V$ randomly.
- **Fix** $\Phi(u) \leftarrow (0, 0)$
- **Let** $v \in C(u, 2s) - \{u\}$
- **Fix** $\Phi(v) \leftarrow (w(u, v), 0)$

WSN field

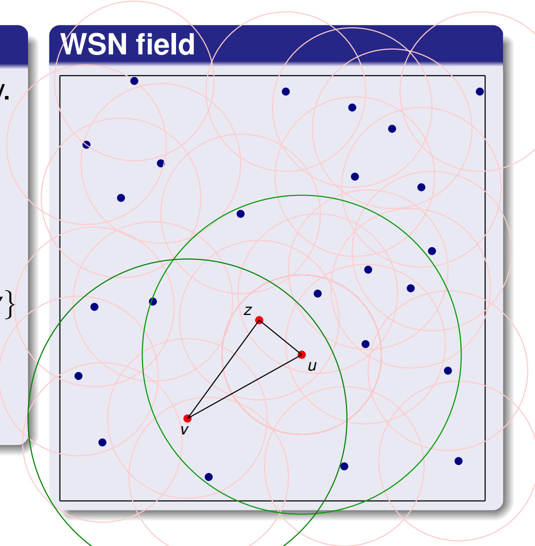


Localization under the Restricted Model
Localization Algorithm

Fixing coordinate system

- **Consider** $u \in V$ randomly.
- **Fix** $\Phi(u) \leftarrow (0, 0)$
- **Let** $v \in C(u, 2s) - \{u\}$
- **Fix** $\Phi(v) \leftarrow (w(u, v), 0)$
- **Let** $z \in C(u, 2s) \cap C(v, 2s) - \{u, v\}$
- **Fix** $\Phi(z)$ satisfying $dist(u, z) = w(u, z)$ and $dist(v, z) = w(v, z)$.

WSN field



Localization under the Restricted Model
Localization Algorithm

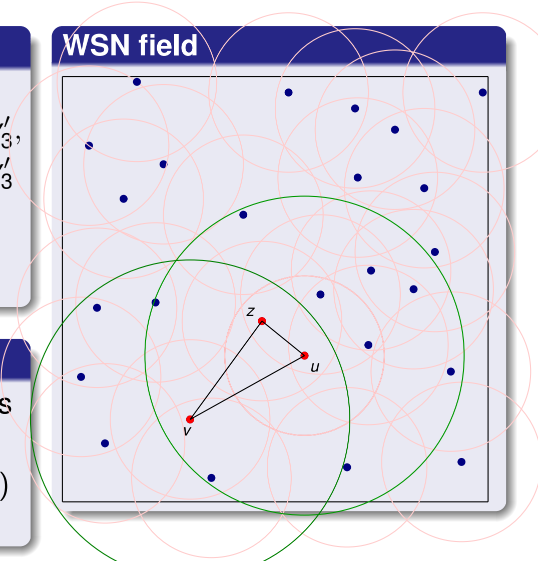
Find $z(x_3, y_3)$ using :

$$
\begin{aligned}
x_3 &= x_1 + \frac{x_2 - x_1}{d_3} x_3' - \frac{y_2 - y_1}{d_3} y_3', \\
y_3 &= y_1 + \frac{y_2 - y_1}{d_3} x_3' + \frac{x_2 - x_1}{d_3} y_3' \\
x_3' &= \frac{d_3^2 + d_2^2 - d_1^2}{2d_3}, \\
y_3' &= \pm\sqrt{d_2^2 - x_3'^2}.
\end{aligned}
$$

assuming

- the positions of u and v as (x_1, y_1) and (x_2, y_2) and
- $d_1 = w(v, z)$, $d_2 = w(z, u)$ and $d_3 = w(u, v)$.

WSN field

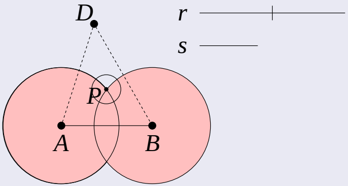


Localization under the Restricted Model
Localization Algorithm

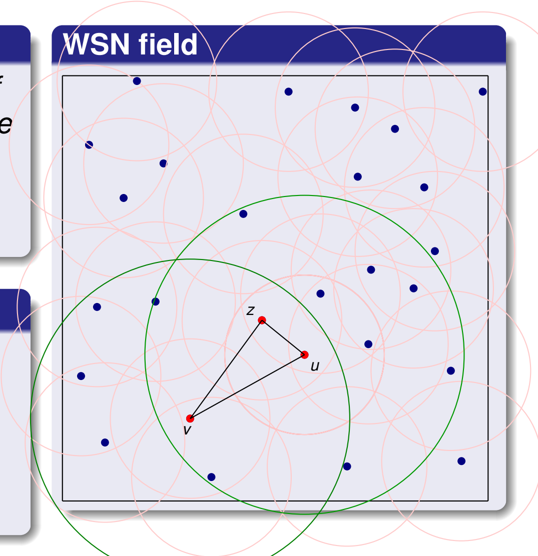
Theorem

Let $\Phi(u) = A$ and $\Phi(v) = B$. If P is an intersection point of the boundaries of $C(A, s)$ and $C(B, s)$, then P is covered by another sensor, say, D.

Existence of $D = \Phi(z)$



WSN field



Localization under the Restricted Model

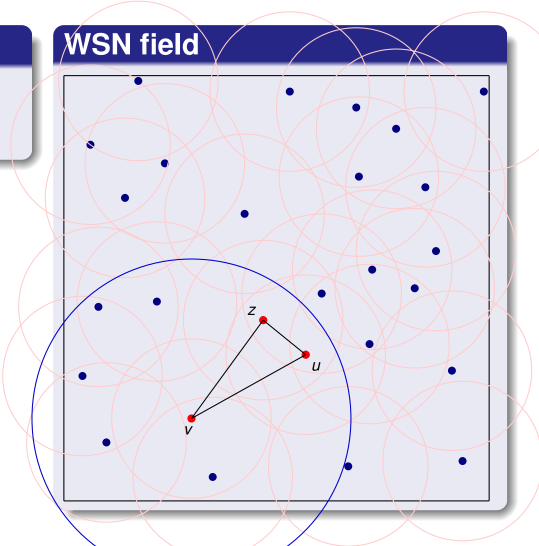
Localization Algorithm

Finding Positions

Push u and z to `located`

Set $s_1 \leftarrow v$

WSN field

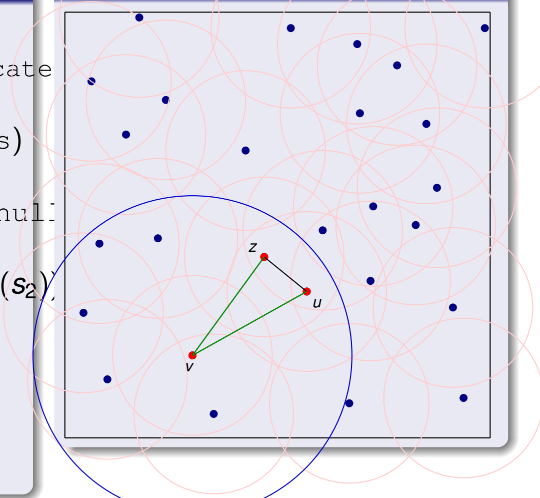


Localization under the Restricted Model
Localization Algorithm

Finding Positions

while($\mathtt{located} \neq \emptyset$)
  Push *Top*($\mathtt{located}$) to $\mathtt{locate}$
  while($\mathtt{locateIn2s} \neq \emptyset$)
    $s_2 \leftarrow Pop(\mathtt{locateIn2s})$
    forall($s_3 \in C(s_1, 2s) \cap$
    $C(s_2, 2s)$ and $\Phi(s_3) = \mathtt{nul}$
      $\Phi(s_3) \leftarrow$
      $\textsc{GetPos}(s_3, s_1, s_2, \Phi(s_2))$
      Push s_3 to $\mathtt{located}$
      and to $\mathtt{locateIn2s}$
    end for
  end while
  $s_1 \leftarrow Pop(\mathtt{located})$

WSN field



Localization under the Restricted Model

Localization Algorithm

func GetPos(s_3, s_1, s_2, p_2)

/*$\Phi(s_1)$ is fixed and s_2 is assigned a position, p_2. If possible return correct position of s_3. Otherwise, return `null`.*/

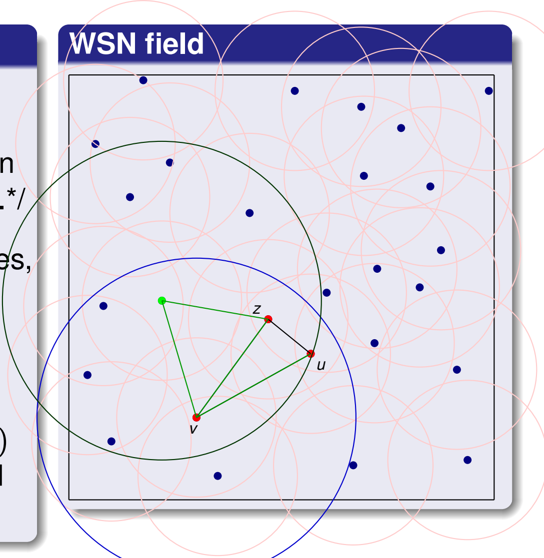
Mark s_2 /* The mark denotes, s_2 is visited but not fixed.*/

Set $p_3 \leftarrow$ `null`

$p, q \leftarrow$ two positions of s_3 from $\triangle s_1 s_2 s_3$.

Say $s_4 \in C(s_1, 2s) \cap C(s_3, 2s) - \{s_1, s_2, s_3\}$ is unmarked

WSN field

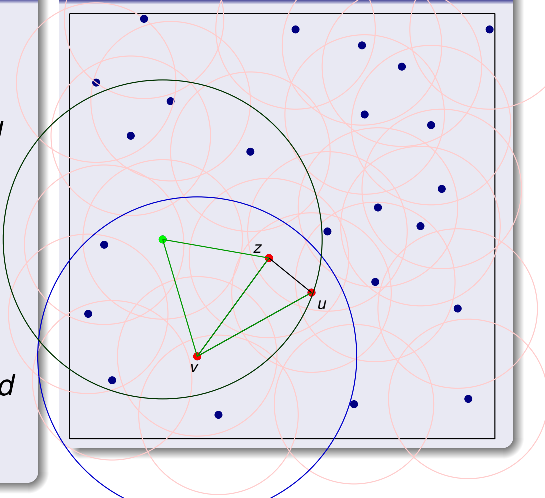


Localization under the Restricted Model
Localization Algorithm

Theorem

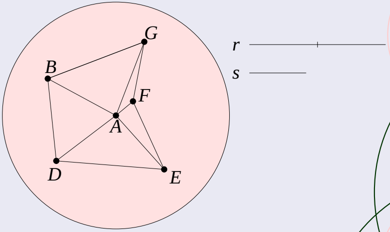
*Any sensor at A, has at least three neighbours within $C(A, 2s)$, such that the neighbours can be connected to form a simple polygon around v_1, with each side of length no more than $2s$. It is also possible to choose the neighbours such that the polygon encloses A. The polygon along with v_1 is a rigid subgraph (**rigid wheel** or simply **wheel**).*

WSN field

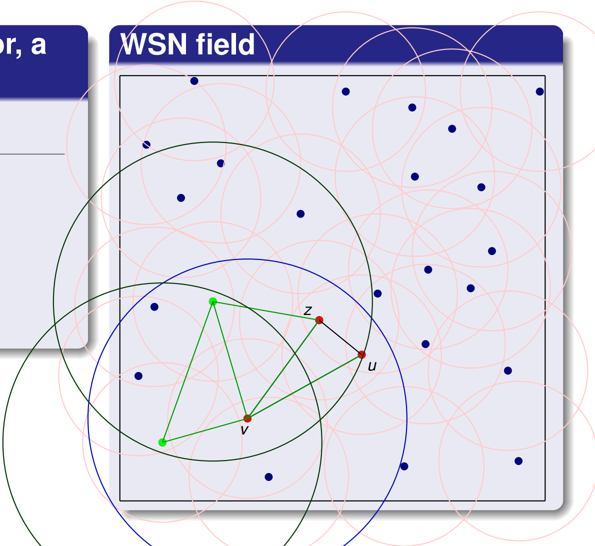


Localization under the Restricted Model
Localization Algorithm

Polygon around a sensor, a wheel.



WSN field



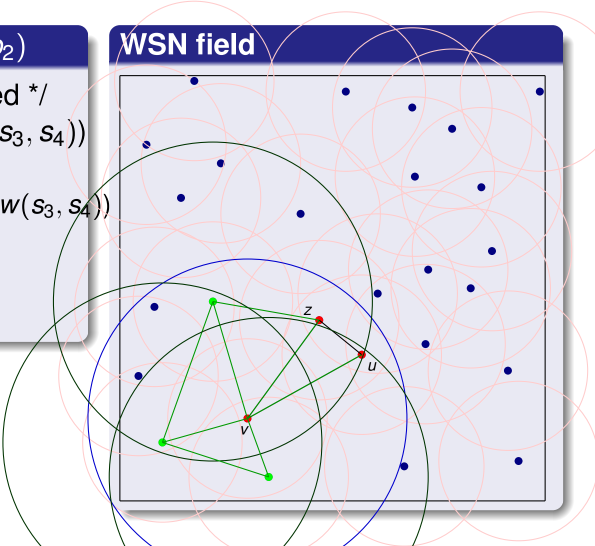
Localization under the Restricted Model

Localization Algorithm

func GETPOS(s_3, s_1, s_2, p_2)

if($\Phi(s_4)$) /* $\Phi(s_4)$ is fixed */
    if($dist(p, \Phi(s_4)) = w(s_3, s_4)$)
        Set $p_3 \leftarrow p$
    else if($dist(q, \Phi(s_4)) = w(s_3, s_4)$)
        Set $p_3 \leftarrow q$
    end if

WSN field



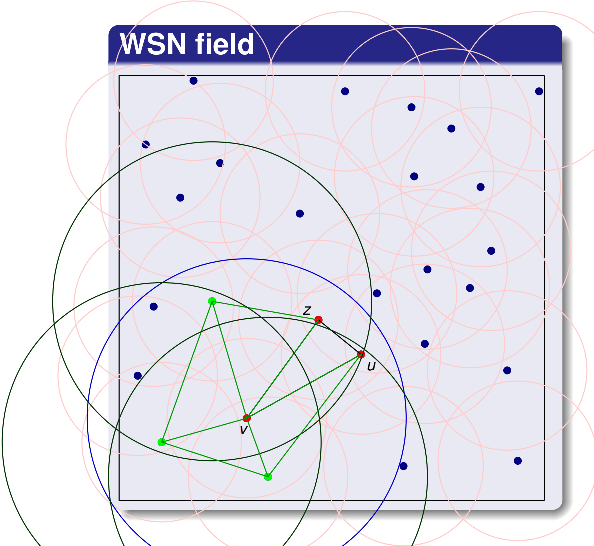
Localization under the Restricted Model
Localization Algorithm

func GetPos(s_3, s_1, s_2, p_2)

```
else   /* Φ(s_4) is not fixed */
    if(t ← GetPos(s_4, s_1, s_3, p))
        Set Φ(s_4) ← t
        Push s_4 to located and locateIn2s
        Set p_3 ← p
    else if(t ← GetPos(s_4, s_1, s_3, q))
        Set Φ(s_4) ← t
        Push s_4 to located and locateIn2s
        Set p_3 ← q
    end if
end if
Unmark s_2
return p_3
```

Localization under the Restricted Model

Localization Algorithm

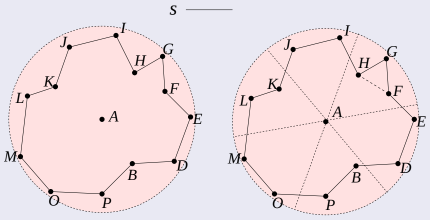


Outline

Restricted Model
Correctness and progress

Theorem

Under the restricted model, each sensor will be surrounded by at least one rigid wheel consisting of no more than twelve vertices in the boundary.



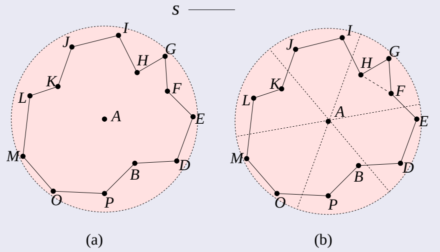
(a) (b)

Restricted Model

Correctness and progress

Theorem

Under the restricted model, each sensor will be surrounded by at least one rigid wheel consisting of no more than twelve vertices in the boundary.



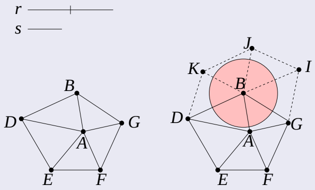
Termination

- The recursion GetPos(s_3, s_1, s_2, p_2) treminates after 12 calls at most.

Restricted Model
Correctness and progress

Theorem

At any point of time, if the network is not completely localized, there is at least one sensor which can be localized.

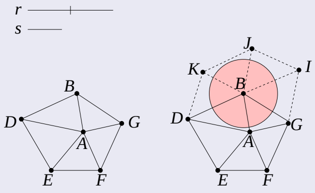


(a) (b)

Restricted Model
Correctness and progress

Theorem

At any point of time, if the network is not completely localized, there is at least one sensor which can be localized.



(a) (b)

Progress

- at least one sensor with unknown position will be localized in each twelve cycles

Restricted Model
Complexity of the Algorithm

Finding Positions

while($\texttt{located} \neq \emptyset$)
    Push *Top*($\texttt{located}$) to $\texttt{locateIn2s}$
    while($\texttt{locateIn2s} \neq \emptyset$)
        $s_2 \leftarrow Pop(\texttt{locateIn2s})$
        for all ($s_3 \in C(s_1, 2s) \cap C(s_2, 2s)$
        and $\Phi(s_3) = \texttt{null}$)
            $\Phi(s_3) \leftarrow \textsc{GetPos}(s_3, s_1, s_2, \Phi(s_2))$
            Push s_3 to $\texttt{located}$
            and to $\texttt{locateIn2s}$
        end for
    end while
    $s_1 \leftarrow Pop(\texttt{located})$
end while

1. The **for all** loop block will be executed in total (considering all repeatations of the earlier while loops) $|V|$ times, once for each vertex.
2. Checking in **for all** loop may have to be carried out is $O(|E|)$.

Restricted Model
Complexity of the Algorithm

Over all run time

Hence the worst-case time complexity of the algorithm is $O(|E|)$.

Outline

Restricted Model
Simulation Studies

1. the worst case complexity, i.e., $O(n^2)$ may be achieved if the WSN has $O(n)$ sensors each with $O(n)$ neighbours.
2. Under random uniform deployment of nodes, the expected value of $|E|$ will be $O(n^2 \frac{r^2}{R})$, R is the field area.
3. For fixed r, the expected value of $|E|$ much lower than $O(n^2)$, in large scale WSN.

Restricted Model
Simulation Studies

Simulation Environment

1. We deploy sensors randomly with uniform distribution till the field is fully covered.
2. We repeat the experiment 30,000 times.
3. We carried out simulation studies (using a C++ program).

Simulation parameters

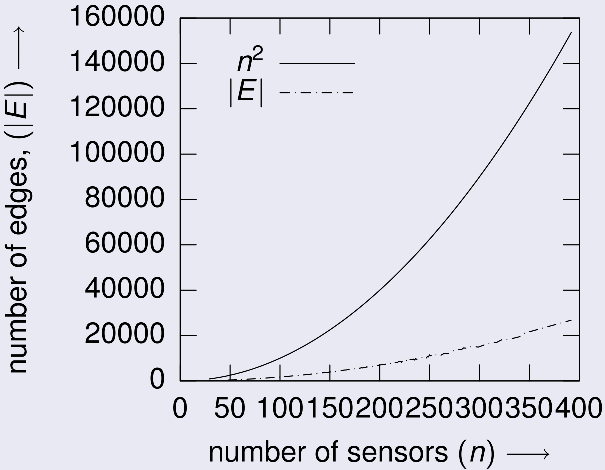
With the parameters :

Parameter	Value
WSN Field dimension	50×50 $unit^2$
Communication range of a sensor, r	24 $unit$
Sensing range of a sensor, s	10 $unit$
Range ratio, $\frac{r}{s}$	2.4

Restricted Model

Simulation Studies

Number of sensors vs. number of edges with length $\leq 2s$, where $s = 10$ *unit*, $R = 2500$ *unit*^{2}



Restricted Model
Simulation Studies

Observation

$|E|$ is much lower than n^2.

Outline

Conclusion

Current Model

- The WSN is fully covered.
- Distances are measured exactly.

Conclusion

Current Model

- The WSN is fully covered.
- Distances are measured exactly.

Future Work

- The WSN has small holes.
- Distances have errors.

Thank you !