Robust Location Detection in Emergency Sensor Networks

S. Ray, R. Ungrangsi, F. D. Pellegrini, A. Trachtenberg, and D. Starobinski. Robust location detection in emergency sensor networks. In Proceedings of IEEE INFOCOM 2003.

Presented by Matthew Tan Creti and DongHoon Shin

Goals

- Create an emergency response system by covering an area with sensor nodes
- During an emergency sensors should be able to provide real time information about location, size, and extent of the disaster area
- The location and identities of first responders in the disaster area should be known

Challenges

- Indoor localization is much more difficult than in open areas
- Need to robustly deal with changing structures and node failures

Options for Emergency Location Detection

- GPS is based on triliteration of satellite signals and works well for most outdoor applications but performs very poorly indoors
- Infrared; can easily be blocked
- Ultrasound; requires line of sight
- Radio; not robust to changes in the environment

Basic Idea of the Paper

- Allow sensor coverage areas to overlap
- Ensure that each position that needs to be resolvable is covered by a unique set of nodes; this set is called the positions signature
- Framework based on *identifying code* theory (uniquely identify every point)
- Has a finer resolution than networks where coverage areas are not allowed to overlap
- Can provide robustness to node failures

tradeoff

resolution ⟷ robustness

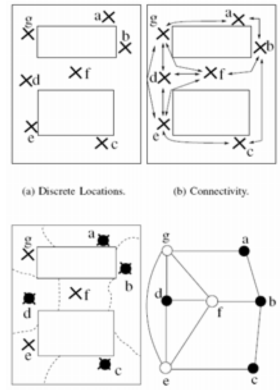
- Use identifying code theory to provide a given amount of robustness with a minimum number of active (not sleeping) nodes

The Model

- Divide the continuous coverage area of the network into a finite set of regions
- Construct a graph where each region is a single vertex and edges between vertices represent the ability for regions to directly communicate with each other
- The problem to be solved is on which vertices to place code words such that every vertex is covered by a unique set of sensors nodes
- We can also solve the problem for r-robust identifying codes; meaning each vertex is uniquely identifiable so long as fewer then r sensor nodes have failed
- Larger values of r require more active nodes

System Overview

- In localization there is a trade off between correctness and resolution; for emergency systems correctness is more important
- Location service or location tracking
- First a set of points are identified for an area; then sensor nodes are placed on a subset of these points
- Each point should be covered by a unique set of nodes
- An observer is then able to identify its location based on the unique collection of ID packets received in a region



(a) Discrete Locations. (b) Connectivity.

v :	a	b	c	d	e	f	g
$ID(v)$:	$\{a,b\}$	$\{a,b,c\}$	$\{b,c\}$	$\{d\}$	$\{c,d\}$	$\{b,d\}$	$\{a,d\}$

Algorithms

- The problem of finding *optimal* identifying codes for arbitrary graphs is NP-complete
- So the authors propose a greedy algorithm that constructs an *irreducible* identifying code
- Irreducible means that if any codeword is removed the code will no longer be an identifying code
- This converges to a local minimum solution and turns out to be very near the optimal solution in most cases
- It should also be noted that the optimal identifying code is an element of the set of all possible irreducible identifying codes (so if we removed codewords in the right order we could produce to optimal identifying code!)

Problem Definition

$\rho(u, v)$ minimum hop count from u to v

$B(v) = \{w \in V : \rho(w, v) \le 1\};$ the ball of v

$\mathbb{C} \subseteq V$ is a code whose elements are called *codewords*

$I_{\mathbb{C}}(v) = B(v) \cap \mathbb{C}$ the identifying set of v

If for every $u, v \in V$ $I_{\mathbb{C}}(u) \neq I_{\mathbb{C}}(v)$ Then C is an identifying code

C is *irreducible* if deletion of any codeword from C results in a code that is no longer an identifying code

A graph is *distinguishable* if it permits an identifying code

The Optimal Problem (NP-complete)

Given a distinguishable graph G=(V,E), determine a subset C of V of minimum cardinality that is an identifying code

The Greedy Problem

Given a distinguishable graph $G = (V, E)$, compute a subset $\mathbb{C}$ of V such that $\mathbb{C}$ is an irreducible identifying code for G.

Distinguishability

- It is important that we are able to determine if a graph is distinguishable.
- If it is not, we can not find an identifying code.

Lemma 1 *For a given graph $G = (V, E)$, if $\mathbb{C}$ is an identifying code, then every $\mathbb{D} \supseteq \mathbb{C}$ is also an identifying code.*

Proof: Assume that there exists $\mathbb{D} \supseteq \mathbb{C}$ that is not an identifying code. Then, by definition, there exist $u, v \in V$ such that

$$\begin{aligned} I_{\mathbb{D}}(u) &= I_{\mathbb{D}}(v) \\ \mathbb{D} \cap B(u) &= \mathbb{D} \cap B(v) \\ \mathbb{C} \cap \mathbb{D} \cap B(u) &= \mathbb{C} \cap \mathbb{D} \cap B(v) \\ \mathbb{C} \cap B(u) &= \mathbb{C} \cap B(v), \quad \text{since } \mathbb{C} \subseteq \mathbb{D} \\ I_{\mathbb{C}}(u) &= I_{\mathbb{C}}(v) \end{aligned}$$

Corollary 1 *V is an identifying code for any distinguishable graph $G = (V, E)$.*

Lemma 2 *A graph $G = (V, E)$ is distinguishable if and only if*

$$\forall\ u, v \in V,\ B(u) \neq B(v).$$

- When we combine corollary 1 and lemma 2 we are able to draw the conclusion that to check if a graph is distinguishable we only need to check that no two vertices have the same ball.
- In practice graphs that have very high or low average degrees are likely to be indistinguishable.

The Algorithm

- The goal of ID-CODE is to find an irreducible code if an identifying code exists.
- We can prove its correctness by contradiction.

```
ID-CODE(G, a)      O(|V|^2 log |V|)
ℂ = V
if ℂ is not an identifying code
  do EXIT
for each vertex x ∈ a, taken in order
  do D = ℂ \ {x}
     if ∃ u, v ∈ V such that I_D(u) = I_D(v)
          ℂ = ℂ
     else ℂ = D
return ℂ
```

Theorem 1 *The code $\mathbb{C}$ returned by ID-CODE is irreducible.*

Proof: Assume, for sake of contradiction, that $\mathbb{C} \setminus X$ is an identifying code for some $X \neq \emptyset$. Choose any codeword $x \in X$ and let i be the iteration in which the ID-CODE considers codeword x and let $\mathbb{C}_i \supset \mathbb{C}$ be the resultant code at the beginning of this iteration. It must be that the set $D \triangleq \mathbb{C}_i \setminus \{x\}$ is not an identifying code, or else ID-CODE would have removed x from $\mathbb{C}$. Moreover, $D \supseteq \mathbb{C} \setminus \{x\} \supseteq \mathbb{C} \setminus X$. However, since $\mathbb{C} \setminus X$ is an identifying code, Lemma 1 implies that D is an identifying code as well, which completes the contradiction. ■

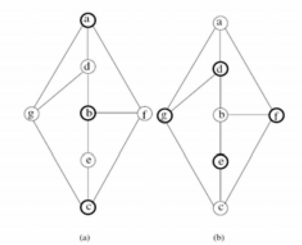


Fig. 3. Different irreducible identifying codes for different sequences: (a) If the sequence of the vertices visited by ID-CODE is $\mathbf{a} = \{f, g, d, e, a, b, c\}$, then the resultant code is $\mathbb{C} = \{a, b, c\}$; (b) on the other hand, with the input sequence $\mathbf{a} = \{a, b, c, d, e, f, g\}$, the resultant code is $\mathbb{C} = \{d, e, f, g\}$.

R-Robust

- In realistic sensor networks radio range is not highly predictable, so for a practical implementation we must be able to accept some errors

$$A \oplus B = (A \setminus B) \cup (B \setminus A)$$

Definition 1 *An identifying code $\mathbb{C}$ over a given graph $G = (V, E)$ is said to be r-robust if*

$$I_{\mathbb{C}}(u) \oplus A \neq I_{\mathbb{C}}(v) \oplus B$$

for all $u, v \in V$ and $A, B \subseteq V$ with $|A|, |B| \leq r$.

- So up to r node insertions or deletions of any identifying set does not prevent unique location identification
- R-robustness can be determined by the minimum symmetric difference

$$d_{\min}(\mathbb{C}) \triangleq \min_{u,v \in V} |I_{\mathbb{C}}(u) \oplus I_{\mathbb{C}}(v)|$$

Theorem 3 *A code $\mathbb{C}$ is r-robust if and only if*

$$d_{min}(\mathbb{C}) \geq 2r + 1.$$

```
r-ID-CODE(G, a, r)
ℂ = V
if d_min(ℂ) ≤ 2r
  do EXIT
for each vertex x ∈ a
  do D = ℂ \ {x}
     if d_min(D) ≤ 2r
          ℂ = ℂ
     else ℂ = D
return ℂ
```

Evaluation of Ordering Methods

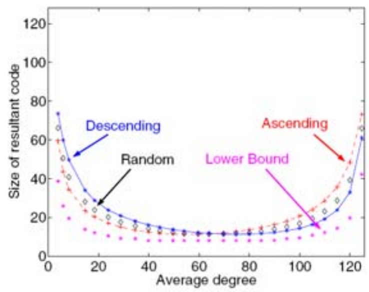


Fig. 6. Performance of various heuristics for $|V| = 128$ vertices graphs.

- Lower Bound – the optimal ordering (NP-complete)
- Random – order vertices randomly
- Descending – sort vertices in decreasing order
- Ascending – sort vertices in increasing order

- The algorithm is more likely to remove nodes that are visited early on
- To get a better result we should visit the "good" vertices last
- When average degree is low, good vertices will likely have a high degree, because this will minimize the number of codewords to cover the graph
- When average degree is high, good vertices will likely have a low degree, because high degree vertices will have similar balls

Results

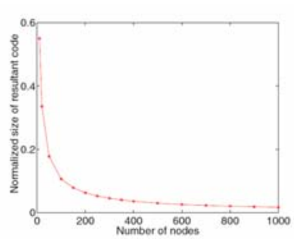


Fig. 7. Scalability of the resultant identifying code C. The normalized size $|C|/|V|$ is plotted against the number of vertices $|V|$ in the graph.

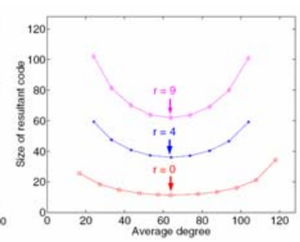


Fig. 8. Behavior of r-robust codes with 128 vertices.

- The smallest resultant code is when average degree is V/2
- Ratio of codewords to graph vertices scales well (in proximity-based systems it would remain at 1)

Extensions

- Can we find disjoint irreducible identifying codes? This would allow the network to evenly distribute the energy cost of being an active node.
- Use SNR rather than range