Localization in graphs and sequential metric dimension

Julien Bensmail, Dorian Mazauric, Fionn Mc Inerney, Nicolas Nisse and Stéphane Pérennes

Université Côte d'Azur, Inria, CNRS, I3S, France

Xidian University, Xi’an, September 5th, 2019
Outline

1. Metric dimension

2. Sequential localization of an **immobile** target

3. Metric dimension in oriented graphs
Precisely locate using few information

Fix any three points $A$, $B$ and $C$ in the plane. For any point $v$, $(\text{dist}(A, v), \text{dist}(B, v), \text{dist}(C, v))$ is sufficient to locate $v$ !

How to generalize to graph metric?
Metric Dimension of graphs

A target is hidden at some (unknown) vertex $t$ of a graph $G = (V, E)$.
Probing a vertex $v \in V(G) \Rightarrow$ the distance $dist_G(t, v)$ between $v$ and $t$.

Resolving set : set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \cdots, v_i\} \subseteq V$ s.t. $(dist_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$.

example : for any tree $T$, $\text{MD}(T) = \#\text{leaves} - \#\text{branching nodes}$

Computing $\text{MD}(G)$ [Harary, Melter 76, Slater 75]
is NP-c in planar graphs [Díaz et al. 17], W[2]-hard [Hartung, Nichterlein 13], FPT in tree-length [Belmonte et al. 17]...
A target is hidden at some (unknown) vertex $t$ of a graph $G = (V, E)$
Probing a vertex $v \in V(G) \Rightarrow$ the distance $dist_G(t, v)$ between $v$ and $t$.

**Resolving set**: set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \ldots, v_i\} \subseteq V$ s.t. $(dist_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$. 

Conflicts

Metric Dimension $\text{MD}(G)$: min. size of a resolving set in $G$.

Example: for any tree $T$, $\text{MD}(T) = \#\text{leaves} - \#\text{branching nodes}$.

Computing $\text{MD}(G)$ [Harary, Melter 76, Slater 75] is NP-c in planar graphs [Díaz et al. 17], $W[2]$-hard [Hartung, Nichterlein 13], FPT in tree-length [Belmonte et al. 17]...
Metric Dimension of graphs

A target is **hidden** at some (unknown) vertex $t$ of a graph $G = (V, E)$
Probing a vertex $v \in V(G) \Rightarrow$ the distance $dist_G(t, v)$ between $v$ and $t$.

**Resolving set** : set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \ldots, v_i\} \subseteq V$ s.t. $(dist_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$.

---

**Example**: for any tree $T$, $MD(T) = \#$leaves $- \#$branching nodes

Computing $MD(G)$ [Harary, Melter 76, Slater 75] is NP-c in planar graphs [D ´ıaz et al. 17], $W[2]$-hard [Hartung, Nichterlein 13], FPT in tree-length [Belmonte et al. 17]...

---

Bensmail, Mazaric, Mc Inerney, Nisse, Pérennes

Localization in graphs and sequential metric dimension.
Metric Dimension of graphs

A target is hidden at some (unknown) vertex $t$ of a graph $G = (V, E)$.

Probing a vertex $v \in V(G) \Rightarrow$ the distance $dist_G(t, v)$ between $v$ and $t$.

**Resolving set**: set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \cdots, v_i\} \subseteq V$ s.t. $(dist_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$.

---

Conflicts

Bensmail, Mazaureic, Mc Inerney, Nisse, Pérennes
Localization in graphs and sequential metric dimension.
A target is **hidden** at some (unknown) vertex $t$ of a graph $G = (V, E)$

Probing a vertex $v \in V(G)$ \(\Rightarrow\) the distance $dist_G(t, v)$ between $v$ and $t$.

**Resolving set**: set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \ldots, v_i\} \subseteq V$ s.t. $(dist_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$. 

---

Metric Dimension of graphs

**Example**: for any tree $T$, $\text{MD}(T) = \#\text{leaves} - \#\text{branching nodes}$

Computing $\text{MD}(G)$ \cite{Harary, Melter76, Slater75} is NP-complete in planar graphs \cite{Diaz17}, $W[2]$-hard \cite{Hartung, Nichterlein13}, FPT in tree-length \cite{Belmonte17}...
Metric Dimension of graphs

A target is hidden at some (unknown) vertex \( t \) of a graph \( G = (V, E) \). Probing a vertex \( v \in V(G) \) \( \Rightarrow \) the distance \( \text{dist}_G(t, v) \) between \( v \) and \( t \).

Resolving set: set of vertices to probe s.t. the target is uniquely located

Set \( R = \{v_1, \ldots, v_i\} \subseteq V \) s.t. \((\text{dist}_G(v, v_i))_{j \leq i}\) pairwise distinct \( \forall v \in V \).
A target is hidden at some (unknown) vertex $t$ of a graph $G = (V, E)$. Probing a vertex $v \in V(G)$ $\Rightarrow$ the distance $\text{dist}_G(t, v)$ between $v$ and $t$.

Resolving set: set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \cdots, v_i\} \subseteq V$ s.t. $(\text{dist}_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$. 

Metric Dimension of graphs

Example: for any tree $T$, $\text{MD}(T) = \#$ leaves $- \#$ branching nodes

Computing $\text{MD}(G)$ [Harary, Melter 76, Slater 75] is NP-c in planar graphs [Díaz et al. 17], W[2]-hard [Hartung, Nichterlein 13], FPT in tree-length [Belmonte et al. 17]...
**Metric Dimension of graphs**

A target is **hidden** at some (unknown) vertex $t$ of a graph $G = (V, E)$.

Probing a vertex $v \in V(G)$ ⇒ the distance $\text{dist}_G(t, v)$ between $v$ and $t$.

**Resolving set** : set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \cdots, v_i\} \subseteq V$ s.t. $(\text{dist}_G(v, v_i))_{j \leq i}$ pairwise distinct $\forall v \in V$.

**Metric Dimension** $MD(G)$ : min. size of a resolving set in $G$. ($MD(G) \leq |V|$)

Example : for any tree $T$, $MD(T) = \# \text{leaves} - \#\text{"branching nodes"}$

Computing $MD(G)$ [Harary, Melter 76, Slater 75] is NP-c in planar graphs [Díaz et al. 17], $W[2]$-hard [Hartung,Nichterlein 13], FPT in tree-length [Belmonte et al. 17]...
Outline

1. Metric dimension

2. Sequential localization of an **immobile** target

3. Metric dimension in oriented graphs
**Sequentiel variant** : Seager (2013) : Probe only **ONE** vertex per turn.
**Sequential Metric Dimension**

**Sequentiel variant**: Seager (2013): Probe only **ONE** vertex per turn.
Sequentiel variant: Seager (2013): Probe only ONE vertex per turn.

Each turn brings some new information
**Sequentiel variant** : Seager (2013) : Probe only **ONE** vertex per turn.
**Sequentiel variant**: Seager (2013): Probe only **ONE** vertex per turn.
**Sequential variant** : Seager (2013) : Probe only **ONE** vertex per turn.

Target found in $< n$ turns in any $n$-node graph :
Probe each vertex (but one) one by one
Sequential Metric Dimension

**Sequentiel variant** : Seager (2013) : Probe only **ONE** vertex per turn.

Target found in $< n$ turns in any $n$-node graph :
Probe each vertex (but one) one by one

**Goal** : Minimize # of turns to locate an **immobile** target hidden in $G$. 
Sequential Metric Dimension & Game of Guess Who?

Localization in graphs and sequential metric dimension.
Sequential Metric Dimension & Game of Guess Who?

Note: One universal vertex is not depicted on the figure.
Note: One universal vertex is not depicted on the figure.
Note: One universal vertex is not depicted on the figure.
Sequential Metric Dimension & Game of Guess Who?

Note: One universal vertex is not depicted on the figure.
What if more than one vertex can be probed per turn?

Sequential Metric Dimension of $G$

Given $k, \ell, G$, is it possible to locate the immobile target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G) : \text{min. \# turns to locate an immobile target, probing } k \text{ vertices per turn.}$
What if more than one vertex can be probed per turn?

**Sequential Metric Dimension of $G$**

Given $k, \ell, G$, is it possible to locate the **immobile** target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G)$: min. $\#$ turns to locate an immobile target, probing $k$ vertices per turn.

**Remark**: for any $G, k \geq 1$, $\lambda_k(G) \leq \lceil \frac{MD(G)}{k} \rceil$

(at each turn, probe $k$ vertices of an optimal resolving set)
What if more than one vertex can be probed per turn?

**Sequential Metric Dimension of $G$**

Given $k, \ell, G$, is it possible to locate the **immobile** target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G) : \text{min. \# turns to locate an immobile target, probing } k \text{ vertices per turn.}$

**Remark** : for any $G$, $k \geq 1$, $\lambda_k(G) \leq \left\lceil \frac{MD(G)}{k} \right\rceil$

Metric Dimension $MD(G) = ?$
What if more than one vertex can be probed per turn?

Sequential Metric Dimension of $G$

Given $k, \ell, G$, is it possible to locate the immobile target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G)$: min. # turns to locate an immobile target, probing $k$ vertices per turn.

Remark: for any $G$, $k \geq 1$, $\lambda_k(G) \leq \left\lceil \frac{MD(G)}{k} \right\rceil$

Metric Dimension $MD(G) = 19$
What if more than one vertex can be probed per turn?

**Sequential Metric Dimension of $G$**

Given $k, \ell, G$, is it possible to locate the immobile target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G) : \text{min. } \# \text{ turns to locate an immobile target, probing } k \text{ vertices per turn.}$

**Remark** : for any $G$, $k \geq 1$, $\lambda_k(G) \leq \left\lceil \frac{MD(G)}{k} \right\rceil$

---

Metric Dimension $MD(G) = 19$

$\lambda_4(G) \leq \left\lceil \frac{19}{4} \right\rceil = 5.$

But...
Sequential Metric Dimension

What if more than one vertex can be probed per turn?

**Sequential Metric Dimension of $G$**

Given $k, \ell, G$, is it possible to locate the **immobile** target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G) : \text{min. \# turns to locate an immobile target, probing } k \text{ vertices per turn.}$

**Remark** : for any $G$, $k \geq 1$, $\lambda_k(G) \leq \lceil \frac{MD(G)}{k} \rceil$

---

**Localisation in graphs and sequential metric dimension.**
What if more than one vertex can be probed per turn?

Sequential Metric Dimension of $G$

Given $k, \ell, G$, is it possible to locate the **immobile** target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G) : \text{min. \# turns to locate an immobile target, probing } k \text{ vertices per turn.}$

**Remark** : for any $G, k \geq 1$, $\lambda_k(G) \leq \left\lceil \frac{MD(G)}{k} \right\rceil$

Metric Dimension $MD(G) = 19$

$\lambda_4(G) \leq \left\lceil \frac{19}{4} \right\rceil = 5.$

But...
In one turn, only five locations remain possible.
What if more than one vertex can be probed per turn?

**Sequential Metric Dimension of** \( G \)

Given \( k, \ell, G \), is it possible to locate the **immobile** target in \( G \) in at most \( \ell \) turns by probing at most \( k \geq 1 \) vertices each turn.

\[ \lambda_k(G) : \text{min. \# turns to locate an immobile target, probing } k \text{ vertices per turn.} \]

**Remark** : for any \( G, k \geq 1 \), \( \lambda_k(G) \leq \left\lceil \frac{MD(G)}{k} \right\rceil \)

**Metric Dimension** \( MD(G) = 19 \)

\[ \lambda_4(G) \leq \left\lceil \frac{19}{4} \right\rceil = 5. \]

**But…**

\[ \lambda_4(G) = 2 < \left\lfloor \frac{19}{4} \right\rfloor. \]
What if more than one vertex can be probed per turn?

**Sequential Metric Dimension of $G$**

Given $k, \ell, G$, is it possible to locate the **immobile** target in $G$ in at most $\ell$ turns by probing at most $k \geq 1$ vertices each turn.

$\lambda_k(G) : \min. \# \text{ turns to locate an immobile target, probing } k \text{ vertices per turn.}$

**Remark :** for any $G, k \geq 1, \lambda_k(G) \leq \lceil \frac{MD(G)}{k} \rceil$

Metric Dimension $MD(G) = 19$

$\lambda_4(G) = 2 < \lceil \frac{19}{4} \rceil$.

**Lemma :** for any $k \geq 1$

$\lambda_k(G)$ may be arbitrary smaller than $\lceil \frac{MD(G)}{k} \rceil$.
\( \lambda_k(G) \): min. \# turns to locate an immobile target, probing \( k \) vertices per turn.

**Our contribution in general graphs:**

**Computational complexity**

- Let \( k \geq 1 \) be a fixed integer. The problem that takes any graph \( G \) with diameter 2 and an integer \( \ell \geq 1 \) as inputs and decides if \( \lambda_k(G) \leq \ell \) is NP-complete.

- Let \( \ell \geq 1 \) be a fixed integer. The problem that takes any graph \( G \) with diameter 2 and an integer \( k \geq 1 \) as inputs and decides if \( \lambda_k(G) \leq \ell \) is NP-complete.

**Polynomial-time algorithm**

- Let \( k, \ell \geq 1 \) be two fixed integers. The problem of deciding if \( \lambda_k(G) \leq \ell \), for any \( n \)-node graph \( G \), can be solved in time \( n^{O(k\ell)} \).
\( \lambda_k(G) \): min. \# turns to locate an immobile target, probing \( k \) vertices per turn.

**Our contribution in TREES:**

<table>
<thead>
<tr>
<th>Computational complexity</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The problem that takes any tree ( T ) and two integers ( k, \ell \geq 1 ) as inputs and decides if ( \lambda_k(T) \leq \ell ) is NP-complete.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Polynomial-time algorithms</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>There exists a polynomial-time +1-approximation to compute ( \lambda_k(T) ).</td>
<td></td>
</tr>
<tr>
<td><strong>Precisely</strong>, there exists an algorithm that computes, in time ( O(n \log n) ), a localization strategy using ( \ell ) turns, probing ( k ) vertices per turn in any ( n )-node tree ( T ), with ( \lambda_k(T) \leq \ell \leq \lambda_k(T) + 1 ).</td>
<td></td>
</tr>
<tr>
<td>Let ( k \geq 1 ) be a fixed integer. The problem of deciding if ( \lambda_k(T) \leq \ell ), for any ( n )-node tree ( T ) and any ( \ell \geq 1 ), can be solved in time ( O(n^{k+2} \log n) ).</td>
<td></td>
</tr>
</tbody>
</table>
Deciding $\lambda_k(T)$ NP-hard in trees [Bensmail, Mazauric, Mc Inerney, N., Pérennes, 2018]

**Sketch**: Reduction from **Hitting Set**: Given $k \geq 1$, a set $B = \{b_1, \cdots, b_n\}$ of elements and a set of subsets $S = \{S_1, \cdots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?
Deciding $\lambda_k(T)$ NP-hard in trees [Bensmail, Mzaouric, Mc Inerney, N., Pérennes, 2018]

**Sketch : Reduction from Hitting Set :** Given $k \geq 1$, a set $B = \{b_1, \ldots, b_n\}$ of elements and a set of subsets $S = \{S_1, \ldots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

Tree $T_\gamma$ ($\gamma$ depends on $n, k$) s.t. $\lambda_k(T_\gamma) \leq 1 + \lceil \sigma^{-1} \rceil + \lceil \gamma^{-1} \rceil$ iff such a set $F$ exists
Deciding $\lambda_k(T)$ NP-hard in trees [Bensmail, Mazauric, Mc Inerney, N., Pérennes, 2018]

**Sketch : Reduction from Hitting Set :** Given $k \geq 1$, a set $B = \{b_1, \ldots, b_n\}$ of elements and a set of subsets $S = \{S_1, \ldots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?
Sketch: Reduction from Hitting Set: Given $k \geq 1$, a set $B = \{b_1, \ldots, b_n\}$ of elements and a set of subsets $S = \{S_1, \ldots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

**Diagram Description:**
- The diagram illustrates a tree structure with nodes labeled $b_i$, $b''_i$, and $b''''_i$.
- Each node represents an element from the set $B$.
- The tree is rooted at node $r$.
- The $i^{th}$ branch is labeled with element $b_i$.
- "Level" $2j$ nodes $b_{2j}$ are associated with set $S_j$.
- $b_i \in S_j$ if $b_i$ is a center of the star (7 leaves).
- Example: $b_i, b''_i \in S_j, b_1, b''_i, b_n \notin S_j$.

**Diagram Details:**
- Elements $b_i$ are placed at various levels of the tree, with each level $2j$ indicating membership in set $S_j$.
- The tree branches out with nodes $b_{2m+1}$ at the bottom level.
Deciding $\lambda_k(T)$ NP-hard in trees [Bensmail, Mazauric, Mc Inerney, N., Pérennes, 2018]

**Sketch : Reduction from Hitting Set :** Given $k \geq 1$, a set $B = \{b_1, \cdots, b_n\}$ of elements and a set of subsets $S = \{S_1, \cdots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

Recall that locating in “big” stars is “long”
Sketch : Reduction from Hitting Set : Given $k \geq 1$, a set $B = \{b_1, \cdots, b_n\}$ of elements and a set of subsets $S = \{S_1, \cdots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

If $b_i \notin S_j$, the target may still be in any leaf of the $\sigma$ stars of level $j$. 

"level" $2j \sim$ set $S_j$

$b_i \in S_j \iff b_{2j}^i$, center of star ($\gamma$ leaves)

$\Rightarrow \sigma$ stars/level

Probing (the last vertex of) some Branch $i$ identifies (in worst case) some level $j$. 

$i^{th}$ branch $\sim$ element $b_i$
Deciding $\lambda_k(T)$ NP-hard in trees [Bensmail, Mazaucric, Mc Inerney, N., Pérennes, 2018]

Sketch: Reduction from Hitting Set: Given $k \geq 1$, a set $B = \{b_1, \ldots, b_n\}$ of elements and a set of subsets $S = \{S_1, \ldots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

If $b_i \notin S_j$, the target may still be in any leaf of the $\sigma$ stars of level $j$.

If $b_i \in S_j$, one star of level $j$ is removed from possible locations.
Deciding $\lambda_k(T)$ NP-hard in trees [Bensmail, Mazaurs, Mc Inerney, N., Pérennes, 2018]

**Sketch : Reduction from Hitting Set :** Given $k \geq 1$, a set $B = \{b_1, \cdots, b_n\}$ of elements and a set of subsets $S = \{S_1, \cdots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

If $b_i \notin S_j$, the target may still be in any leaf of the $\sigma$ stars of level $j$.
If $b_i \in S_j$, one star of level $j$ is removed from possible locations.

Probing a Hitting set at first turn $\Rightarrow$ remove one star in each level
**Sketch:** Reduction from **Hitting Set**: Given $k \geq 1$, a set $B = \{b_1, \cdots, b_n\}$ of elements and a set of subsets $S = \{S_1, \cdots, S_m\} \subseteq 2^B$ with $|S_i| = \sigma$ for any $1 \leq i \leq m$, is there a set $F \subseteq B$ with $F \cap S_i$ for all $i \leq m$ and $|F| \leq k$?

The difficulty only comes from the **FIRST** turn!
After the first turn, it becomes “easy”.
Indeed: probe any vertex
After the first turn, it becomes “easy”.
Indeed: probe any vertex
After the first turn, it becomes “easy”.
Indeed: probe any vertex

So, after this first turn, the instance becomes:
A rooted tree, with all leaves at same distance from the root, and the target on some leaf.
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

(1) Find in which subtree the target hide (2) recursively search in this subtree.
**Theorem**: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$

1. Find in which subtree the target hide
2. Recursively search in this subtree.

**Key point**: Probing a single vertex in a subtree says if the target is in this subtree or not.
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1. Find in which subtree the target hide
2. Recursively search in this subtree.

Key point: Probing a single vertex in a subtree says if the target is in this subtree or not.

1st approach: Probe one vertex per subtree until finding the “good” one.
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$

(1) Find in which subtree the target hide 
(2) recursively search in this subtree.

Key point: Probing a single vertex in a subtree says if the target is in this subtree or not.

1st approach: Probe one vertex per subtree until finding the “good” one.
1-approximation in trees

[Bensmail, Mazauric, Mc Inerney, N., Pérennes, 2018]

**Theorem:** In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$

1. Find in which subtree the target hide
2. Recursively search in this subtree.

**Key point:** Probing a single vertex in a subtree says if the target is in this subtree or not.

1° **Approach:** Probe one vertex per subtree until finding the “good” one.
**Theorem:** In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$

1. Find in which subtree the target hide
2. Recursively search in this subtree.

**Key point:** Probing a single vertex in a subtree says if the target is in this subtree or not.

1st **approach:** Probe one vertex per subtree until finding the “good” one.

**Question:** In which order to probe the subtrees?
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1. Find in which subtree the target hide.
2. Recursively search in this subtree.

Key point: Probing a single vertex in a subtree says if the target is in this subtree or not.

1st approach: Probe one vertex per subtree until finding the “good” one.

Question: In which order to probe the subtrees?
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1. Find in which subtree the target hide
2. Recursively search in this subtree.

Key point: Probing a single vertex in a subtree says if the target is in this subtree or not.

1st approach: Probe one vertex per subtree until finding the “good” one.

Question: In which order to probe the subtrees? ⇒ by non-increasing order of their $\lambda_k^*$: probe first the subtrees that are long to search.
**Theorem**: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1. Find in which subtree the target hide
2. Recursively search in this subtree.

1<sup>st</sup> **approach**: Probe one vertex per subtree (in non-increasing order of their $\lambda_k^*$) until finding the “good” one.
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$

1. Find in which subtree the target hide
2. Recursively search in this subtree.

$1^{th}$ approach: Probe one vertex per subtree (in non-increasing order of their $\lambda^*_k$) until finding the “good” one. ⇒ Not optimal :(
Theorem: In a rooted tree with all leaves at the same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1. Find in which subtree the target hide
2. Recursively search in this subtree.

1st approach: Probe one vertex per subtree (in non-increasing order of their $\lambda_k^*$) until finding the “good” one. ⇒ Not optimal :

Subtlety: what is the difference between these two graphs?

possible locations are the leaves
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1) Find in which subtree the target hide (2) recursively search in this subtree.

1st approach: Probe one vertex per subtree (in non-increasing order of their $\lambda^*_k$) until finding the “good” one. ⇒ Not optimal :(

Subtlety: what is the difference between these two graphs?

Possible locations are the leaves

\begin{align*}
\lambda^*_3(S_5) &= 2 \\
\lambda^*_3(S_6) &= 2
\end{align*}

with a single vertex probed at the first turn

but at least two vertices must be probed at the first turn
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$

1. Find in which subtree the target hide
2. Recursively search in this subtree.

1st approach: Probe one vertex per subtree (in non-increasing order of their $\lambda^*_k$) until finding the “good” one. ⇒ Not optimal :(

Subtlety: what is the difference between these two graphs?

$\lambda^*_3(S_5) = 2$ with a single vertex probed at the first turn

$\lambda^*_3(S_6) = 2$ but at least two vertices must be probed at the first turn

⇒ During Phase (1): probing more than one vertex in some subtrees may “accelerate” the search in these subtrees.
Theorem: In a rooted tree with all leaves at same distance from the root and the target at some leaf, an optimal strategy can be computed in time $O(n \log n)$.

1. Find in which subtree the target hide
2. Recursively search in this subtree.

Optimal approach: Probe one or some vertices per subtree (in non-increasing order of their $\lambda_k^*$) until finding the “good” one.

The number of vertices to probe in each subtree in Phase (1) is decided by the (a bit) technical part of our Dynamic Programming algorithm.
Further work on sequential metric dimension

Open questions

- Can we compute $\lambda_k(T)$ in FPT time in trees (i.e., in time $f(k) \cdot \text{poly}(n)$)?
- Complexity of deciding if $\lambda_k(G) \leq \ell$ in other graph classes (interval graphs, chordal graphs, bounded treewidth, etc.)

Other variant: when the target can move

At each turn, after $k$ vertices have been probed, the target may move to a neighbor of its current position

- Already many results on this variant
- but also many open questions
...sorry, no time to go further on it
Outline

1. Metric dimension
2. Sequential localization of an \textit{immobile} target
3. Metric dimension in oriented graphs
**Orientation of** $G$: each edge $\{u, v\}$ becomes exactly one arc among $uv$ or $vu$.

Probing a vertex $v \in V(G) \Rightarrow$ the distance $\text{dist}_G(v, t)$ FROM $v$ TO $t$.

**Resolving set**: set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \ldots, v_i\} \subseteq V$ s.t. $(\text{dist}_D(v_i, v))_{j \leq i}$ pairwise distinct $\forall v \in V$.

**Remark**: *A priori*, $\text{dist}_D(v, t)$ may be $\infty$.

$\rightarrow$ ONLY strongly connected orientations.
Metric Dimension in Oriented Graphs

**Orientation of** $G$ : each edge $\{u, v\}$ becomes exactly one arc among $uv$ or $vu$.

**Probing** a vertex $v \in V(G) \Rightarrow$ the distance $dist_G(v, t)$ FROM $v$ TO $t$.

**Resolving set** : set of vertices to probe s.t. the target is uniquely located

Set $R = \{v_1, \cdots, v_i\} \subseteq V$ s.t. $(dist_D(v_i, v))_{j \leq i}$ pairwise distinct $\forall v \in V$.

**Remark** : *A priori*, $dist_D(v, t)$ may be $\infty$.

$\Rightarrow$ **ONLY** strongly connected orientations.

**MD**($D$) : min. size of a resolving set in a **strong** oriented graph $D$.

**Few related work**

- upper bounds [Chartrand et al. 00]
- NP-complete in strong oriented graphs [Rajan et al. 14]
- complete graphs [Lozano 13], Cayley digraphs [Fehr et al. 06], de Bruijn and Kautz [Rajan et al. 14]
Given a class $\mathcal{G}$ of undirected $n$-node graphs,

- $WOMD(\mathcal{G}) = \sup_{D \text{ strong orientation of } G \in \mathcal{G}} \frac{MD(D)}{n}$
- $BOMD(\mathcal{G}) = \inf_{D \text{ strong orientation of } G \in \mathcal{G}} \frac{MD(D)}{n}$
Given a class $\mathcal{G}$ of undirected $n$-node graphs,

- $\text{WOMD}(\mathcal{G}) = \sup_{D \text{ strong orientation of } G \in \mathcal{G}} \frac{\text{MD}(D)}{n}$
- $\text{BOMD}(\mathcal{G}) = \inf_{D \text{ strong orientation of } G \in \mathcal{G}} \frac{\text{MD}(D)}{n}$

Very few previous work

- Tournaments: $\text{WOMD}(K_n) = 1/2$
- $\mathcal{H}_n$, class of Hamiltonian $n$-node graphs: $\text{BOMD}(\mathcal{H}_n) = 1/n$.

Every Hamiltonian graph has an orientation $D$ with $\text{MD}(D) = 1$. 

Localization in graphs and sequential metric dimension.
Our contributions

Focus on Worst Orientations (WOMD) for various graph classes.

\(G_{\Delta}\) : class of graphs with maximum degree \(\leq \Delta\).

- \(\frac{2}{5} \leq WOMD(G_3) \leq \frac{1}{2}\)
- \(\frac{1}{2} \leq WOMD(G_4) \leq \frac{6}{7}\)
- \(\lim_{\Delta \to \infty} WOMD(G_{\Delta}) = 1\)

**Grids** : class of cartesian grids.

- \(\frac{1}{2} \leq WOMD(Grids) \leq \frac{2}{3}\)

\(WOMD^*\) defined as \(WOMD\) but over Eulerian orientations (in-degree=out-degree).

**Tori** : class of cartesian tori.

- \(WOMD^*(Tori) = \frac{1}{2}\)
Easy lemmas but very useful

**Lower bound**

- *S* set of vertices with exactly same in-neighbors
  - \( \Rightarrow \) Every resolving set contains \( \geq |S| - 1 \) vertices in *S*.
Easy lemmas but very useful

**Lower bound**

$S$ set of vertices with exactly **same in-neighbors**

$\Rightarrow$ Every resolving set contains $\geq |S| - 1$ vertices in $S$.

**Upper bound**

$D = (V, A)$ be any strong oriented graph

$G_{aux}$ undirected graph with vertex-set $V$

$\{u, v\} \in E(G_{aux}) \iff N_D^-(u) \cap N_D^-(v) \neq \emptyset$ (intersecting in-neighborhoods).

**Lemma** : Every (non-empty) vertex cover of $G_{aux}$ is a resolving set for $D$
Easy lemmas but very useful

**Lower bound**

$S$ set of vertices with exactly **same in-neighbors**

$\Rightarrow$ Every resolving set contains $\geq |S| - 1$ vertices in $S$.

**Upper bound**

$D = (V, A)$ be any strong oriented graph

$G_{aux}$ undirected graph with vertex-set $V$

$\{u, v\} \in E(G_{aux}) \iff N_D^-(u) \cap N_D^-(v) \neq \emptyset$ (intersecting in-neighborhoods).

**Lemma** : Every (non-empty) vertex cover of $G_{aux}$ is a resolving set for $D$

**Application** : If $G_{aux}$ has max. degree $\Delta'$, then $\chi(G_{aux}) \leq \Delta' + 1$ (chromatic number), so $\alpha(G_{aux}) \leq \frac{\Delta'}{\Delta' + 1} n$, and so $MD(D) \leq \frac{\Delta'}{\Delta' + 1} n$ for every strong orientation $D$ of $G$. 
Lower bound: \( S \) set of vertices with exactly same in-neighbors
\[ \Rightarrow \text{Every resolving set contains } \geq |S| - 1 \text{ vertices in } S. \]

Maximum degree \( \Delta = d + 1 \geq 3 \)
- “Complete” \( d \)-ary tree depth \( k \)
  (Force a “large” resolving set)
Lower Bound for $G_{\Delta}$

**Lower bound**: $S$ set of vertices with exactly same in-neighbors

$\Rightarrow$ Every resolving set contains $\geq |S| - 1$ vertices in $S$.

Maximum degree $\Delta = d + 1 \geq 3$

- “Complete” $d$-ary tree depth $k$ (Force a “large” resolving set)
- Add a “reversed” complete $d$-ary tree depth $k - 1$ (ensure strong connectedness)
Lower Bound for $\mathcal{G}_\Delta$

**Lower bound**: $S$ set of vertices with exactly same in-neighbors

$\Rightarrow$ Every resolving set contains $\geq |S| - 1$ vertices in $S$.

Maximum degree $\Delta = d + 1 \geq 3$

- “Complete” $d$-ary tree depth $k$
  (Force a “large” resolving set)

- Add a “reversed” complete $d$-ary tree depth $k - 1$
  (ensure strong connectedness)

- Contract green edges
  (reduce the size, preserving resolving set)
Lower Bound for $G_\Delta$

**Lower bound**: $S$ set of vertices with exactly same in-neighbors

$\Rightarrow$ Every resolving set contains $\geq |S| - 1$ vertices in $S$.

Maximum degree $\Delta = d + 1 \geq 3$

- “Complete” $d$-ary tree depth $k$
  (Force a “large” resolving set)

- Add a “reversed” complete $d$-ary tree depth $k - 1$
  (ensure strong connectedness)

- Contract green edges
  (reduce the size, preserving resolving set)

**Do the maths**:

\[
\lim_{k \to \infty} \frac{MD}{|V|} \geq \frac{2}{5} \quad \text{for } \Delta = 3
\]

\[
\lim_{k \to \infty} \frac{MD}{|V|} \geq \frac{1}{2} \quad \text{for } \Delta = 4
\]

\[
\lim_{\Delta \to \infty} \frac{MD}{|V|} = 1
\]
Lower Bound: $S$ set of vertices with exactly same in-neighbors

$\Rightarrow$ Every resolving set contains $\geq |S| - 1$ vertices in $S$.

Lemma: For $\mathcal{G} \in \{\text{Grids, Tori}\}$, $\text{WOMD}(\mathcal{G}) \geq \frac{1}{2}$
Thm: Eulerian orientation \( \vec{T} \) of the torus \( (d^+ = d^- = 2) \Rightarrow MD(\vec{T}) \leq n/2 \)

Start with a MIS \( X \) (in black), local modifications till resolving set of same size.

While \( X \) is not a resolving set, problems in vertex-disjoint “bad squares”

\( u \) and \( v \) have same in-neighbours.
**Thm**: Eulerian orientation $\vec{T}$ of the torus ($d^+ = d^- = 2$) $\Rightarrow$ $MD(\vec{T}) \leq n/2$

Start with a MIS $X$ (in **black**), local modifications till resolving set of same size.

While $X$ is not a resolving set, problems in vertex-disjoint “bad squares” $u$ and $v$ have same in-neighbours.

Replace $n_v$ by $u$ in $X$, sequentially in all “bad squares”, makes $X$ a resolving set (proof by case analysis).
Thm: Any orientation $\tilde{G}$ of a grid $\Rightarrow MD(\tilde{G}) \leq 2n/3$

Start with a set $X$ (in black), local modifications till resolving set of same size.

Again, proof by case analysis...
Further work on Directed Metric Dimension

This is a preliminary work.

- Many bounds to be tightened (Grids, subcubic graphs...)
- Improve tools for upper bounds
- Generalize tools and results to planar graphs.
Further work on Directed Metric Dimension

This is a preliminary work.

- Many bounds to be tightened (Grids, subcubic graphs...)
- Improve tools for upper bounds
- Generalize tools and results to planar graphs.

Why focusing on strong orientations?

- Not strong orientations seem to require different approaches
- Allowing infinite vertex (source not in resolving set) seems to change many things (ongoing work in trees with Julien and UFC)
- Link with MIS?
Further work on Directed Metric Dimension

This is a preliminary work.

- Many bounds to be tightened (Grids, subcubic graphs...)
- Improve tools for upper bounds
- Generalize tools and results to planar graphs.

Why focusing on strong orientations?

- Not strong orientations seem to require different approaches
- Allowing infinite vertex (source not in resolving set) seems to change many things (ongoing work in trees with Julien and UFC)
- Link with MIS?

Thank you, 谢谢, Merci!