Sequential Metric Dimension

Julien Bensmail\textsuperscript{1}, Dorian Mazauric\textsuperscript{2}, \textbf{Fionn Mc Inerney}\textsuperscript{1}, Nicolas Nisse\textsuperscript{1}, Stéphane Pérennes\textsuperscript{1}

\textsuperscript{1}Université Côte d’Azur, Inria, CNRS, I3S, France
\textsuperscript{2}Université Côte d’Azur, Inria, France

May 10, 2019, Lyon, France

Séminaire Graphes@Lyon
A facility, city or some other location is modelled by a graph $G$.

A target is hidden at a vertex of $G$ (i.e., a building).

Detector at each vertex; when “probed”, returns its distance to the target.

### Metric Dimension of $G$

**Min. # of vertices** needed to be probed all at once to **locate the target in $G$**.
Some known results on the Metric Dimension

- Decision problem is NP-C [Garey and Johnson, 1979].
  - NP-C in planar graphs [Díaz et al., 2017].
  - NP-C in diameter 2 graphs [Foucaud et al., 2017].
- W[2]-hard [Hartung and Nichterlein, 2013].
- FPT in graphs with bounded tree-length [Belmonte et al., 2017].
- Bounds in interval and permutation graphs [Foucaud et al., 2017].
- Easy in trees (contract all degree 2 vertices, now take all leaves but one in each star that remains) [Slater, 1975, Harary and Melter, 1976].
Sequential Locating Game on Graphs [Seager, 2013] & Game of Guess Who?

$SL(G)$: Sequential location number of $G$

Min. # of turns of probing one vertex per turn, to locate a target hidden in $G$.

- Solved for subdivisions of caterpillars [Seager, 2013].
Sequential Locating Game on Graphs \cite{Seager2013} & Game of Guess Who?

\textbf{SL}(G) : Sequential location number of $G$

Min. \# of turns of probing one vertex per turn, to locate a target hidden in $G$.

- Solved for subdivisions of caterpillars \cite{Seager2013}. 

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{Sequential_Locating_Game_on_Graphs.png}
\caption{Sequential Locating Game on Graphs and Game of Guess Who.}
\end{figure}
Sequential Locating Game on Graphs [Seager, 2013] & Game of Guess Who?

Femme

Alfred
Alex
Anita
Anne
Bernard
Charles
Bill
Claire
David
Eric
George
Frans
Herman
Joe
Maria
Max
Paul
Philip
Peter
Richard
Robert
Sam
Susan

Moustache

Chapeau

Blond

Roux

Lunettes

Couettes

Femme

Ch. blanc

Gros nez

Barbu

Moustache

Cheveux Bruns/Noirs

Chauve

Boucles d'oreilles

Chapeau

Yeux bleus/ clairs

Sequential Metric Dimension
Sequential Locating Game on Graphs [Seager, 2013] & Game of Guess Who?
Sequential Locating Game on Graphs [Seager, 2013] & Game of Guess Who?

Bensmail, Mazauric, Mc Inerney, Nisse, Pérennes

Sequential Metric Dimension
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al., 2017]; $k = 1$ [Seager, 2012].

- NP-hard to decide if probing $k$ vertices per turn guarantees locating moving target [Bosek et al., 2017].
- Probing at most 3 vertices per turn guarantees locating in outerplanar graphs [Bosek et al., 2017].
- Probing at most 1 vertex per turn guarantees locating non-backtracking target in trees [Seager, 2012].
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$

turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.


$\left\lceil \frac{MD(G)}{k} \right\rceil$ turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
Sequential Metric Dimension

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

Related: Locate moving target with \( k \)? [Bosek et al, 2017]; \( k = 1 \) [Seager, 2012].

\[
\left\lceil \frac{MD(G)}{k} \right\rceil \text{ turns suffice to locate target. But it can be located faster.}
\]

\( \text{Blue vertex probed} \)

\( \text{Answer: } d = 1 \)

\( MD(G) = 19 \) and for \( k = 4 \), the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al., 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

Blue vertex probed

Answer: $d = 3$

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
Sequential Metric Dimension

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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$\left\lceil \frac{MD(G)}{k} \right\rceil$ turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

\[ \lceil \frac{MD(G)}{k} \rceil \] turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

\[
\left\lceil \frac{MD(G)}{k} \right\rceil
\]
turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

\[
\left\lceil \frac{MD(G)}{k} \right\rceil
\]

turns suffice to locate target. But it can be located faster.

$MD(G) = 19$

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
Sequential Metric Dimension

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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

MD($G$) = 19 and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
Sequential Metric Dimension

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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

MD($G$) = 19 and for $k = 4$, the target can be located in 2 turns.
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 $k$ vertices each turn.

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

\[
\left\lceil \frac{MD(G)}{k} \right\rceil
\]

turns suffice to locate target. But it can be located faster.

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
Sequential Metric Dimension

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

Related: Locate moving target with $k$? [Bosek et al, 2017]; $k = 1$ [Seager, 2012].

$$\left\lceil \frac{MD(G)}{k} \right\rceil$$ turns suffice to locate target. But it can be located faster.

Now sequential probing

All vertices in blue probed

Locate in 2 turns (SECOND turn)

$MD(G) = 19$ and for $k = 4$, the target can be located in 2 turns.
Sequential Metric Dimension - General Complexity

\[ \lambda_k(G) : \text{min. number of turns to locate target in } G, \text{ probing } k \text{ vertices per turn.} \]

**Theorem**

Deciding whether \( \lambda_k(G) \leq \ell \) is polynomial-time solvable (in time \( n^{O(k \ell)} \)) when both \( k \), the number of vertices to be probed, and \( \ell \), the number of turns, are fixed.

**Theorem**

NP-complete to decide whether \( \lambda_k(G) \leq \ell \) when either \( k \), the number of vertices to be probed, or \( \ell \), the number of turns, is fixed.
Trees

NP-complete in trees.

Difficulty only comes from first turn.

polynomial-time \((+1)\)-approximation algorithm for trees.
More precisely in trees ...

$\lambda_k(T)$

min. # turns to locate target in $T$, probing $k$ vertices per turn.

$(+1)$-approximation algorithm

- computes strategy that locates target in $T$ in at most $\lambda_k(T) + 1$ steps.
- Time complexity : $O(n \log n)$.

Exact algorithm

- computes strategy that locates target in $T$ in at most $\lambda_k(T)$ steps.
- Time complexity : $O(n^{k+2} \log n)$.
Theorem

Determining $\lambda_k(T)$ is \textit{NP-hard} in trees $T$.

\textbf{Proof (sketch)}. Reduction from \textsc{Hitting Set} (given a set $B := \{b_1, \ldots, b_n\}$ and a set $S := \{S_1, \ldots, S_m\}$ of subsets of $B$, find a smallest subset of $B$ hitting all $S_i$'s).

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{diagram.png}
\end{figure}
Theorem

Determining $\lambda_k(T)$ is \emph{NP-hard} in trees $T$.

\textbf{Proof (sketch).} Reduction from \textsc{Hitting Set} (given a set $B := \{b_1, ..., b_n\}$ and a set $S := \{S_1, ..., S_m\}$ of subsets of $B$, find a smallest subset of $B$ hitting all $S_i$'s).

Main ideas:
- Many big stars in the tree, so target hides in one.
- Find the hosting big star, and then "peel" its leaves.
- $\Rightarrow$ Identifying the big star early $\iff$ \textsc{Hitting set}.
NP-hardness in trees

Theorem
Determining $\lambda_k(T)$ is NP-hard in trees $T$.

Proof (sketch). Reduction from Hitting Set (given a set $B := \{b_1, ..., b_n\}$ and a set $S := \{S_1, ..., S_m\}$ of subsets of $B$, find a smallest subset of $B$ hitting all $S_i$’s).

Main ideas:
- Many big stars in the tree, so target hides in one.
- Find the hosting big star, and then “peel” its leaves.
Theorem

Determining $\lambda_k(T)$ is NP-hard in trees $T$.

Proof (sketch). Reduction from Hitting Set (given a set $B := \{b_1, ..., b_n\}$ and a set $S := \{S_1, ..., S_m\}$ of subsets of $B$, find a smallest subset of $B$ hitting all $S_i$'s).

Main ideas:

- Many big stars in the tree, so target hides in one.
- Find the hosting big star, and then “peel” its leaves.

$\Rightarrow$ Identifying the big star early $\iff$ Hitting set.
Reduction, illustrated

More identification turns, but one eliminated star.
Reduction, illustrated

\[ r \]

↓  \[ b_1 \]

↓  \[ b_i \]

↓  \[ b_{i'} \]

↓  \[ b_{i''} \]

↓  \[ b_n \]

More identification turns, but one eliminated star.

Probing

More identification turns, with no eliminated star.
Reduction, illustrated
Reduction, illustrated

Probing

Immediate star identification.

More identification turns, with no eliminated star.

Bensmail, Mazauric, Mc Inerney, Nisse, Perennes

Sequential Metric Dimension
Reduction, illustrated

Probing

Immediate star identification.

More identification turns, but one eliminated star.

Probing

Probing

More identification turns, with no eliminated star.
Reduction, illustrated

Immediate star identification.

Probing
Probing

Reduction, illustrated
Reduction, illustrated

Probing

More identification turns, with no eliminated star.

Probing

Probing

Probing

Probing

Probing

Probing

Probing
Reduction, illustrated

More identification turns, but one eliminated star.
Reduction, illustrated

Probing

Probing

Sequential Metric Dimension
Reduction, illustrated

More identification turns, with no eliminated star.

Probing

Probing
Trees: why problem is “easy” after 1\textsuperscript{st} turn

Probe any 1 vertex on 1\textsuperscript{st} turn.
Probe any 1 vertex on 1st turn.
Trees: why problem is "easy" after 1\textsuperscript{st} turn

Probe any 1 vertex on 1\textsuperscript{st} turn.

Result: Tree $T'$ rooted in $r$ where all leaves are the same distance from $r$ and the target is known to occupy a leaf.
Probe any 1 vertex on 1st turn.

Result: Tree $T'$ rooted in $r$ where all leaves are the same distance from $r$ and the target is known to occupy a leaf.
Second key argument for “easiness”

$T_v$ : subtree rooted in $v$ of $T'$ rooted in $r$, $v$ is a child of $r$.

Probing 1 vertex in $T_v$ allows to know if target is in $T_v$ or in $T' \setminus T_v$. 

![Diagram](image-url)
Two parameters needed for algorithm

Assume target is known to occupy a leaf of $T_i$. Then,

$\lambda_k(T_i)$ : min. # of turns of probings to locate target in $T_i$.

$\pi_k(T_i)$ : min. # of vertices to probe in $T_i$ during first turn of probing vertices in $T_i$ to locate target in $\lambda_k(T_i)$ turns.

$k = 3$ \hspace{1cm} $\lambda_3(T) = 3$ \hspace{1cm} $\pi_3(T) = 2$
Need for tradeoff between probing 1 or $\pi(T_{v_i})$ vertices in $T_{v_i}$

$k = 3$

$\lambda(T) = 5$

$(\lambda_3(T_{v_i}), \pi_3(T_{v_i}))$
Need for tradeoff between probing 1 or $\pi(T_{v_i})$ vertices in $T_{v_i}$

$$k = 3 \quad \lambda(T) = 5$$

$$\lambda_3(T_{v_i}), \pi_3(T_{v_i})$$
Need for tradeoff between probing 1 or $\pi(T_{v_i})$ vertices in $T_{v_i}$

$k = 3$  \hspace{1cm} $\lambda(T) = 5$

$\lambda_3(T_{v_i}), \pi_3(T_{v_i})$
Example of algorithm in trees $(k = 5)$ $(\lambda_5(T_{v_i}), \pi_5(T_{v_i}))$

Current $\lambda_5(T) : 0$
Current leftover vertices that can be probed : 0
Example of algorithm in trees \((k = 5) (\lambda_5(T_{v_i}), \pi_5(T_{v_i}))\)

Current \(\lambda_5(T) : 1\)
Vertices to probe in \(T_{v_5} : 2\)
Current leftover vertices that can be probed : 3
Example of algorithm in trees \((k = 5)\) \((\lambda_5(T_{v_i}), \pi_5(T_{v_i}))\)

Current \(\lambda_5(T) : 2\)

Vertices to probe in \(T_{v_4} : 4\)

Current leftover vertices that can be probed : 1
Example of algorithm in trees \((k = 5)\) \((\lambda_5(T_{v_i}), \pi_5(T_{v_i}))\)

Current \(\lambda_5(T)\) : 3

Vertices to probe in \(T_{v_3}\) : 1

Current leftover vertices that can be probed : 4
Example of algorithm in trees \((k = 5)\) \((\lambda_5(T_{v_i}), \pi_5(T_{v_i}))\)

Current \(\lambda_5(T) : 5\)  
Vertices to probe in \(T_{v_2} : 3\)  
Current leftover vertices that can be probed : 2  

5\(^{th}\) last turn : 3 in \(T_{v_2}\)  
3\(^{rd}\) last turn : 1 in \(T_{v_3}\)  
2\(^{nd}\) last turn : 4 in \(T_{v_4}\)  
Last turn : 2 in \(T_{v_5}\)
Example of algorithm in trees \((k = 5)\) \((\lambda_5(T_{v_i}), \pi_5(T_{v_i}))\)

\[
\begin{align*}
\text{6}\text{th last turn} &: 1 \text{ in } T_{v_1} \\
\text{5}\text{th last turn} &: 3 \text{ in } T_{v_2} \\
\text{3}\text{rd last turn} &: 1 \text{ in } T_{v_3} \\
\text{2}\text{nd last turn} &: 4 \text{ in } T_{v_4} \\
\text{Last turn} &: 2 \text{ in } T_{v_5}
\end{align*}
\]

\[(\lambda_5(T), \pi_5(T)) = (6, 1)\]

Current \(\lambda_5(T)\) : 6

Vertices to probe in \(T_{v_1}\) : 1

Current leftover vertices that can be probed : 4
Variant with relative distances

Relative distances returned instead of exact distances.

Target

Relative dist. vector from probing blue vertices: \((v_7 < v_4 = v_8 < v_1)\).

Complexity

**NP-complete** when either number of vertices to be probed or number of turns is fixed.
Future work

- Study the problem with exact distances in other graph classes such as planar and interval graphs.
- Try to get exact values for the problem with relative distances in paths [Foucaud et al, 2014].
- Study the problem with relative distances in trees.
Thanks!