The threshold dimension of a graph

Lucas Mol

THE UNIVERSITY OF WINNIPEG

Joint work with Matthew J. H. Murphy (University of Toronto)
and Ortrud R. Oellermann (The University of Winnipeg)

East Coast Combinatorics Conference 2019

PLAN

METRIC DIMENSION

THRESHOLD DIMENSION

BOUNDS

EMBEDDINGS

RESOLVING SETS

Let G be a graph.

RESOLVING SETS

Let G be a graph.

- A set $S \subseteq V(G)$ is a *resolving set* of G if every vertex of G is uniquely determined by its vector of distances to vertices in S.

RESOLVING SETS

Let G be a graph.

- A set $S \subseteq V(G)$ is a *resolving set* of G if every vertex of G is uniquely determined by its vector of distances to vertices in S.

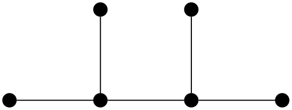
  Example:

Resolving Sets

Let G be a graph.

- A set $S \subseteq V(G)$ is a *resolving set* of G if every vertex of G is uniquely determined by its vector of distances to vertices in S.

Example:

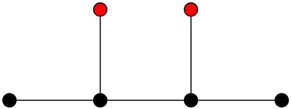


Resolving Sets

Let G be a graph.

- A set $S \subseteq V(G)$ is a *resolving set* of G if every vertex of G is uniquely determined by its vector of distances to vertices in S.

Example:

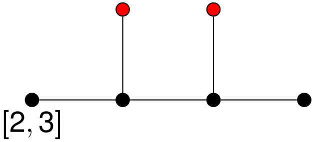


RESOLVING SETS

Let G be a graph.

- A set $S \subseteq V(G)$ is a *resolving set* of G if every vertex of G is uniquely determined by its vector of distances to vertices in S.

Example:

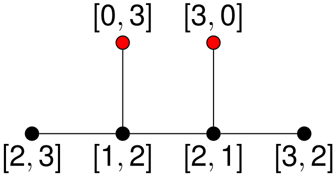


RESOLVING SETS

Let G be a graph.

- A set $S \subseteq V(G)$ is a *resolving set* of G if every vertex of G is uniquely determined by its vector of distances to vertices in S.

Example:



Motivation

MOTIVATION

- Think of the vertices in a resolving set as “landmark” vertices.

Motivation

- Think of the vertices in a resolving set as “landmark” vertices.
- Suppose that if an agent is sitting at some vertex of the graph, the landmark vertices can tell how far away the agent is.

MOTIVATION

- Think of the vertices in a resolving set as “landmark” vertices.
- Suppose that if an agent is sitting at some vertex of the graph, the landmark vertices can tell how far away the agent is.
- By pooling information from all landmark vertices, one can tell exactly where the agent is sitting!

Motivation

- Think of the vertices in a resolving set as “landmark” vertices.
- Suppose that if an agent is sitting at some vertex of the graph, the landmark vertices can tell how far away the agent is.
- By pooling information from all landmark vertices, one can tell exactly where the agent is sitting!
- Applications: locating an intruder, robot navigation, etc.

MOTIVATION

- Think of the vertices in a resolving set as “landmark” vertices.
- Suppose that if an agent is sitting at some vertex of the graph, the landmark vertices can tell how far away the agent is.
- By pooling information from all landmark vertices, one can tell exactly where the agent is sitting!
- Applications: locating an intruder, robot navigation, etc.
- If there is a cost associated with establishing or maintaining landmark vertices, then one would be interested in finding the smallest possible resolving set.

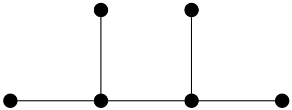
METRIC DIMENSION

- The minimum cardinality of a resolving set of G is called the *metric dimension* of G, denoted $\beta(G)$.

Metric Dimension

- The minimum cardinality of a resolving set of G is called the *metric dimension* of G, denoted $\beta(G)$.

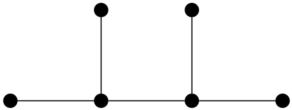
Example:



Metric Dimension

- The minimum cardinality of a resolving set of G is called the *metric dimension* of G, denoted $\beta(G)$.

Example:

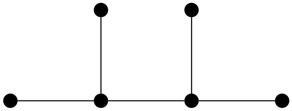


- We have already seen a resolving set of cardinality 2 in this graph.

METRIC DIMENSION

- The minimum cardinality of a resolving set of G is called the *metric dimension* of G, denoted $\beta(G)$.

Example:

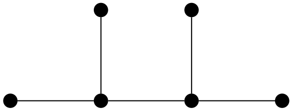


- We have already seen a resolving set of cardinality 2 in this graph.
- One checks that there is no resolving set of cardinality 1.

METRIC DIMENSION

- The minimum cardinality of a resolving set of G is called the *metric dimension* of G, denoted $\beta(G)$.

Example:



- We have already seen a resolving set of cardinality 2 in this graph.
- One checks that there is no resolving set of cardinality 1.
- Therefore, this graph has metric dimension 2.

More Examples

$\beta(P_n) = 1$

MORE EXAMPLES

$\beta(P_n) = 1$

- This is the unique connected graph of order n and metric dimension 1.

More Examples

$\beta(P_n) = 1$

- This is the unique connected graph of order n and metric dimension 1.

$\beta(K_n) = n - 1.$

MORE EXAMPLES

$\beta(P_n) = 1$

- This is the unique connected graph of order n and metric dimension 1.

$\beta(K_n) = n - 1.$

- This is the unique connected graph of order n and metric dimension $n - 1$.

More Examples

$\beta(P_n) = 1$

- This is the unique connected graph of order n and metric dimension 1.

$\beta(K_n) = n - 1$.

- This is the unique connected graph of order n and metric dimension $n - 1$.

Fact: If G has order n, then $1 \leq \beta(G) \leq n - 1$.

A Brief History

A BRIEF HISTORY

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.

A Brief History

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.
- Finding the metric dimension is NP-hard in general – deciding whether the metric dimension of a graph is at most a given integer is NP-complete (Garey & Johnson 1979).

A Brief History

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.
- Finding the metric dimension is NP-hard in general – deciding whether the metric dimension of a graph is at most a given integer is NP-complete (Garey & Johnson 1979).
- Polynomial time algorithms exist for several restricted classes of graphs:

A Brief History

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.
- Finding the metric dimension is NP-hard in general – deciding whether the metric dimension of a graph is at most a given integer is NP-complete (Garey & Johnson 1979).
- Polynomial time algorithms exist for several restricted classes of graphs:
  - outerplanar graphs (Díaz et al., 2012)

A BRIEF HISTORY

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.
- Finding the metric dimension is NP-hard in general – deciding whether the metric dimension of a graph is at most a given integer is NP-complete (Garey & Johnson 1979).
- Polynomial time algorithms exist for several restricted classes of graphs:
  - outerplanar graphs (Díaz et al., 2012)
  - graphs of bounded cyclomatic number (Epstein et al., 2012)

A Brief History

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.
- Finding the metric dimension is NP-hard in general – deciding whether the metric dimension of a graph is at most a given integer is NP-complete (Garey & Johnson 1979).
- Polynomial time algorithms exist for several restricted classes of graphs:
  - outerplanar graphs (Díaz et al., 2012)
  - graphs of bounded cyclomatic number (Epstein et al., 2012)
  - etc.

A Brief History

- Introduced independently by Slater (1975), and Harary and Melter (1976). Both gave a linear time algorithm for calculating the metric dimension of a tree.
- Finding the metric dimension is NP-hard in general – deciding whether the metric dimension of a graph is at most a given integer is NP-complete (Garey & Johnson 1979).
- Polynomial time algorithms exist for several restricted classes of graphs:
  - outerplanar graphs (Díaz et al., 2012)
  - graphs of bounded cyclomatic number (Epstein et al., 2012)
  - etc.
- Upper bounds in terms of diameter (Khuller et al., 1996, sharpened by Hernando et al., 2010).

PLAN

METRIC DIMENSION

THRESHOLD DIMENSION

BOUNDS

EMBEDDINGS

Threshold Dimension

THRESHOLD DIMENSION

- If landmark vertices are expensive, we want to find a smallest possible resolving set.

Threshold Dimension

- If landmark vertices are expensive, we want to find a smallest possible resolving set.
- Imagine that we can add edges to a graph cheaply (relative to landmark vertices).

Threshold Dimension

- If landmark vertices are expensive, we want to find a smallest possible resolving set.
- Imagine that we can add edges to a graph cheaply (relative to landmark vertices).
- Then we would want to find the smallest resolving set across all graphs H that can be obtained from G by adding edges.

Threshold Dimension

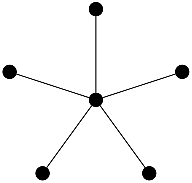
- If landmark vertices are expensive, we want to find a smallest possible resolving set.
- Imagine that we can add edges to a graph cheaply (relative to landmark vertices).
- Then we would want to find the smallest resolving set across all graphs H that can be obtained from G by adding edges.

- The *threshold dimension* of G, denoted $\tau(G)$, is the size of such a smallest resolving set:

$$\tau(G) = \min\{\beta(H) : H \text{ contains } G \text{ as a spanning subgraph}\}$$

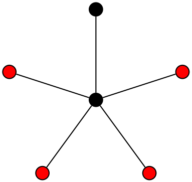
AN EXAMPLE

Consider the graph $G = K_{1,5}$.



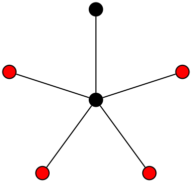
AN EXAMPLE

Consider the graph $G = K_{1,5}$.



An Example

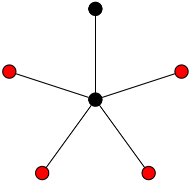
Consider the graph $G = K_{1,5}$.



$\beta(G) = 4$

AN EXAMPLE

Consider the graph $G = K_{1,5}$.

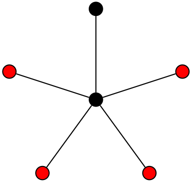


$\beta(G) = 4$

Now add a couple of carefully chosen edges:

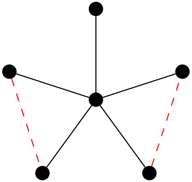
AN EXAMPLE

Consider the graph $G = K_{1,5}$.



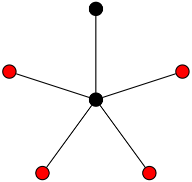
$\beta(G) = 4$

Now add a couple of carefully chosen edges:



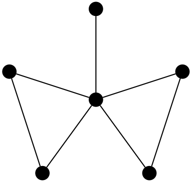
An Example

Consider the graph $G = K_{1,5}$.



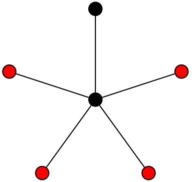
$\beta(G) = 4$

Now add a couple of carefully chosen edges:



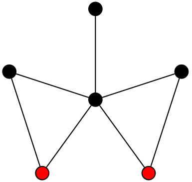
AN EXAMPLE

Consider the graph $G = K_{1,5}$.



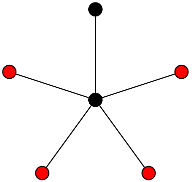
$\beta(G) = 4$

Now add a couple of carefully chosen edges:



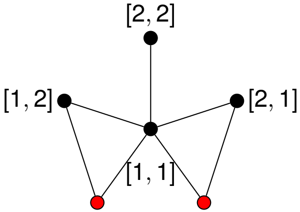
AN EXAMPLE

Consider the graph $G = K_{1,5}$.



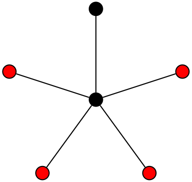
$\beta(G) = 4$

Now add a couple of carefully chosen edges:



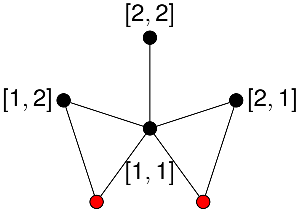
AN EXAMPLE

Consider the graph $G = K_{1,5}$.



$\beta(G) = 4$

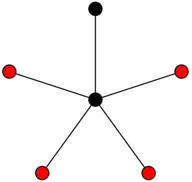
Now add a couple of carefully chosen edges:



$\tau(G) \leq 2$

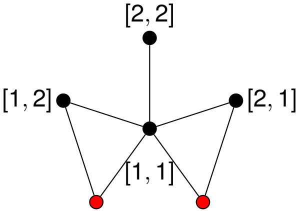
An Example

Consider the graph $G = K_{1,5}$.



$\beta(G) = 4$

Now add a couple of carefully chosen edges:



$\tau(G) = 2$

Plan

Metric Dimension

Threshold Dimension

Bounds

Embeddings

TREES

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.
Sketch of Proof:

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.
Sketch of Proof:

- If $\beta(T) \leq d_n$, then we are done.

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.
Sketch of Proof:

- If $\beta(T) \leq d_n$, then we are done.
- Otherwise, it must be the case that T has at least d_n leaves.

Trees

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.
Sketch of Proof:

- If $\beta(T) \leq d_n$, then we are done.
- Otherwise, it must be the case that T has at least d_n leaves.
- Take any set W of d_n leaves – this is going to be our resolving set.

Trees

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.
Sketch of Proof:

- If $\beta(T) \leq d_n$, then we are done.
- Otherwise, it must be the case that T has at least d_n leaves.
- Take any set W of d_n leaves – this is going to be our resolving set.
- Since $n \leq 2^{d_n} + d_n$, there are at most 2^{d_n} vertices outside of W.

TREES

For every positive real number x, let d_x denote the smallest positive integer such that $x \leq 2^{d_x} + d_x$.

- Note that $d_x < \log_2(x)$.

Theorem (MMO 2019+): Let T be a tree of order n. Then $\tau(T) \leq d_n$. Moreover, this bound is sharp.
Sketch of Proof:

- If $\beta(T) \leq d_n$, then we are done.
- Otherwise, it must be the case that T has at least d_n leaves.
- Take any set W of d_n leaves – this is going to be our resolving set.
- Since $n \leq 2^{d_n} + d_n$, there are at most 2^{d_n} vertices outside of W.
- Attach each vertex not in W to a unique subset of W.

A Bound in terms of the Chromatic Number

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

Sketch of proof:

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

Sketch of proof:

- The vertices of G can be partitioned into k independent sets $V_1, \ldots V_k$, say of orders $n_1, \ldots, n_k$, respectively.

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

Sketch of proof:

- The vertices of G can be partitioned into k independent sets $V_1, \dots V_k$, say of orders $n_1, \dots, n_k$, respectively.
- Add all edges between these independent sets – we are now looking at $K_{n_1, n_2, \dots, n_k}$.

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

Sketch of proof:

- The vertices of G can be partitioned into k independent sets $V_1, \ldots V_k$, say of orders $n_1, \ldots, n_k$, respectively.
- Add all edges between these independent sets – we are now looking at $K_{n_1, n_2, \ldots, n_k}$.
- For every i, take d_{n_i} vertices from V_i – together, these will form a resolving set.

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

Sketch of proof:

- The vertices of G can be partitioned into k independent sets $V_1, \ldots V_k$, say of orders $n_1, \ldots, n_k$, respectively.
- Add all edges between these independent sets – we are now looking at $K_{n_1, n_2, \ldots, n_k}$.
- For every i, take d_{n_i} vertices from V_i – together, these will form a resolving set.
- Use ideas like we did for trees.

A Bound in terms of the Chromatic Number

Theorem (MMO 2019+): Let G be a graph of order n with chromatic number k. Then

$$\tau(G) < k(d_{n/k} + 2) < k(\log_2(n/k) + 2).$$

Sketch of proof:

- The vertices of G can be partitioned into k independent sets $V_1, \dots V_k$, say of orders $n_1, \dots, n_k$, respectively.
- Add all edges between these independent sets – we are now looking at $K_{n_1, n_2, \dots, n_k}$.
- For every i, take d_{n_i} vertices from V_i – together, these will form a resolving set.
- Use ideas like we did for trees.
- Finally, show that the worst case is when the n_i's are approximately equal.

Plan

Metric Dimension

Threshold Dimension

Bounds

Embeddings

EMBEDDINGS AND STRONG PRODUCTS

An *embedding* of G in H is an injective function $\phi : V(G) \to V(H)$ satisfying

$$xy \in E(G) \Rightarrow \phi(x)\phi(y) \in E(H).$$

EMBEDDINGS AND STRONG PRODUCTS

An *embedding* of G in H is an injective function $\phi : V(G) \to V(H)$ satisfying

$$xy \in E(G) \Rightarrow \phi(x)\phi(y) \in E(H).$$

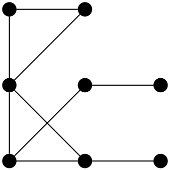
In other words, an embedding of G in H is an injective homomorphism from G to H.

Embeddings and Strong Products

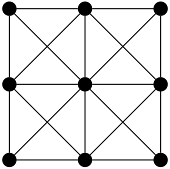
An *embedding* of G in H is an injective function $\phi : V(G) \to V(H)$ satisfying

$$xy \in E(G) \Rightarrow \phi(x)\phi(y) \in E(H).$$

In other words, an embedding of G in H is an injective homomorphism from G to H.



G



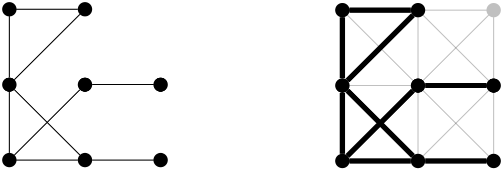
H

Embeddings and Strong Products

An *embedding* of G in H is an injective function $\phi : V(G) \to V(H)$ satisfying

$$xy \in E(G) \Rightarrow \phi(x)\phi(y) \in E(H).$$

In other words, an embedding of G in H is an injective homomorphism from G to H.



G H

STRONG PRODUCTS

STRONG PRODUCTS

We will be concerned with embeddings of graphs in the strong product of a number of paths.

STRONG PRODUCTS

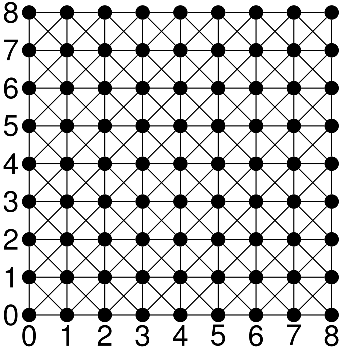
We will be concerned with embeddings of graphs in the strong product of a number of paths.

- The strong product of 2 paths is a 2-dimensional grid with diagonal edges in addition to horizontal and vertical ones.

STRONG PRODUCTS

We will be concerned with embeddings of graphs in the strong product of a number of paths.

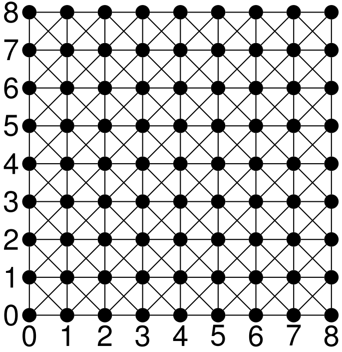
- The strong product of 2 paths is a 2-dimensional grid with diagonal edges in addition to horizontal and vertical ones.



STRONG PRODUCTS

We will be concerned with embeddings of graphs in the strong product of a number of paths.

- The strong product of 2 paths is a 2-dimensional grid with diagonal edges in addition to horizontal and vertical ones.



- The strong product of *b* paths is an analogous *b*-dimensional grid.

Lemma (MMO 2019+): If $\beta(G) = b$, then G can be embedded in the strong product of b paths.

Lemma (MMO 2019+): If $\beta(G) = b$, then G can be embedded in the strong product of b paths.

Idea of proof:

Lemma (MMO 2019+): If $\beta(G) = b$, then G can be embedded in the strong product of b paths.

Idea of proof:

- Let $W = \{w_1, \ldots, w_b\}$ be a resolving set of G.

Lemma (MMO 2019+): If $\beta(G) = b$, then G can be embedded in the strong product of b paths.

Idea of proof:

- Let $W = \{w_1, \ldots, w_b\}$ be a resolving set of G.
- Define ϕ by

$$\phi(x) = [d(x, w_1), d(x, w_2), \ldots, d(x, w_k)].$$

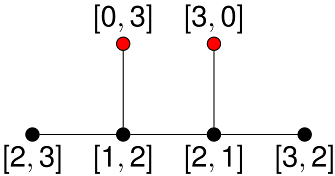
Lemma (MMO 2019+): If $\beta(G) = b$, then G can be embedded in the strong product of b paths.

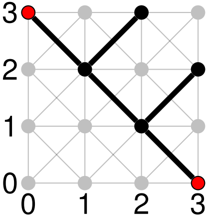
Idea of proof:

- Let $W = \{w_1, \dots, w_b\}$ be a resolving set of G.
- Define ϕ by

$$\phi(x) = [d(x, w_1), d(x, w_2), \dots, d(x, w_k)].$$

Example:



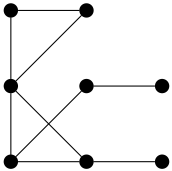


Notation

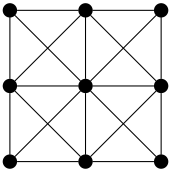
- For an embedding ϕ of G in H, let $\phi(G)$ denote the subgraph of H induced by the set $\phi(V(G))$.

NOTATION

- For an embedding ϕ of G in H, let $\phi(G)$ denote the subgraph of H induced by the set $\phi(V(G))$.



G



H

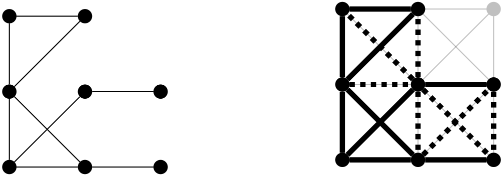
NOTATION

- For an embedding ϕ of G in H, let $\phi(G)$ denote the subgraph of H induced by the set $\phi(V(G))$.

G H

NOTATION

- For an embedding ϕ of G in H, let $\phi(G)$ denote the subgraph of H induced by the set $\phi(V(G))$.



G $\qquad$ $\phi(G)$

GOOD EMBEDDINGS

Good Embeddings

- Call an embedding ϕ of G in the strong product of b paths *good* if there is a set of vertices $W = \{w_1, \ldots, w_b\}$ such that for all vertices x of G, we have

$$\phi(x) = [d_{\phi(G)}(\phi(x), \phi(w_1)), \ldots, d_{\phi(G)}(\phi(x), \phi(w_b))].$$

GOOD EMBEDDINGS

- Call an embedding ϕ of G in the strong product of b paths *good* if there is a set of vertices $W = \{w_1, \dots, w_b\}$ such that for all vertices x of G, we have

$$\phi(x) = [d_{\phi(G)}(\phi(x), \phi(w_1)), \dots, d_{\phi(G)}(\phi(x), \phi(w_b))].$$

- Essentially, the coordinates of $\phi(x)$ are the distances from $\phi(x)$ to the vertices in the set $\phi(W)$ **in the subgraph** $\phi(G)$**.**

GOOD EMBEDDINGS

- Call an embedding ϕ of G in the strong product of b paths *good* if there is a set of vertices $W = \{w_1, \ldots, w_b\}$ such that for all vertices x of G, we have

$$\phi(x) = [d_{\phi(G)}(\phi(x), \phi(w_1)), \ldots, d_{\phi(G)}(\phi(x), \phi(w_b))].$$

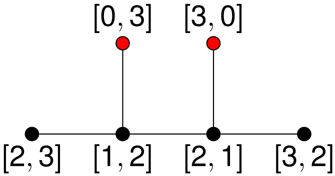
- Essentially, the coordinates of $\phi(x)$ are the distances from $\phi(x)$ to the vertices in the set $\phi(W)$ **in the subgraph** $\phi(G)$**.**
- The embeddings constructed in the previous lemma are the prototypical good embeddings.

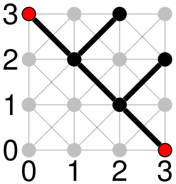
GOOD EMBEDDINGS

- Call an embedding ϕ of G in the strong product of b paths *good* if there is a set of vertices $W = \{w_1, \ldots, w_b\}$ such that for all vertices x of G, we have

$$\phi(x) = [d_{\phi(G)}(\phi(x), \phi(w_1)), \ldots, d_{\phi(G)}(\phi(x), \phi(w_b))].$$

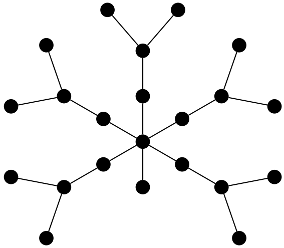
- Essentially, the coordinates of $\phi(x)$ are the distances from $\phi(x)$ to the vertices in the set $\phi(W)$ **in the subgraph** $\phi(G)$**.**
- The embeddings constructed in the previous lemma are the prototypical good embeddings.

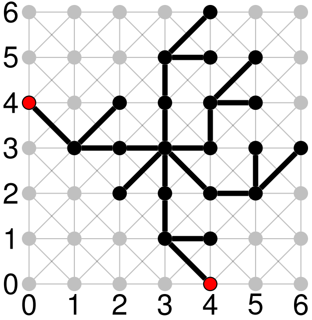




More Good Embeddings

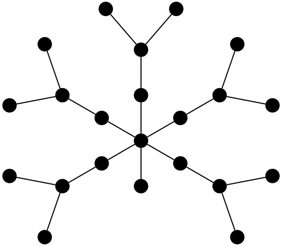
This tree has metric dimension 5, but has a good embedding in the strong product of only 2 paths.

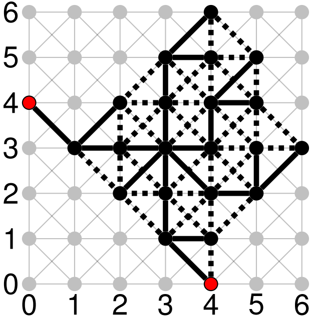




More Good Embeddings

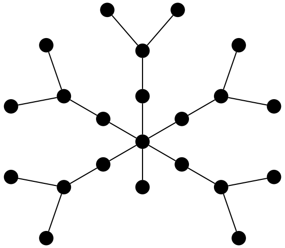
This tree has metric dimension 5, but has a good embedding in the strong product of only 2 paths.

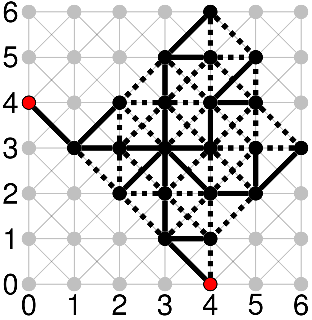




More Good Embeddings

This tree has metric dimension 5, but has a good embedding in the strong product of only 2 paths.





This tree has threshold dimension 2.

A GEOMETRICAL CHARACTERIZATION

Theorem (MMO 2019+): Let G be a graph. Then $\tau(G) = b$ if and only if b is the smallest number such that there is a good embedding of G in the strong product of b paths.

A Geometrical Characterization

Theorem (MMO 2019+): Let G be a graph. Then $\tau(G) = b$ if and only if b is the smallest number such that there is a good embedding of G in the strong product of b paths.

- So the notion of threshold dimension corresponds in some way to our usual geometrical notion of dimension.

A GEOMETRICAL CHARACTERIZATION

Theorem (MMO 2019+): Let G be a graph. Then $\tau(G) = b$ if and only if b is the smallest number such that there is a good embedding of G in the strong product of b paths.

- So the notion of threshold dimension corresponds in some way to our usual geometrical notion of dimension.
- We thought this was cool!

A GEOMETRICAL CHARACTERIZATION

Theorem (MMO 2019+): Let *G* be a graph. Then $\tau(G) = b$ if and only if *b* is the smallest number such that there is a good embedding of *G* in the strong product of *b* paths.

- So the notion of threshold dimension corresponds in some way to our usual geometrical notion of dimension.
- We thought this was cool!
- Is it useful?

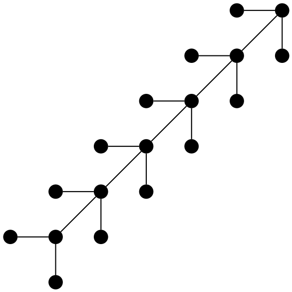
Question: Are there trees of arbitrarily large metric dimension whose threshold dimension is 2?

Question: Are there trees of arbitrarily large metric dimension whose threshold dimension is 2?

Answer: Yes.

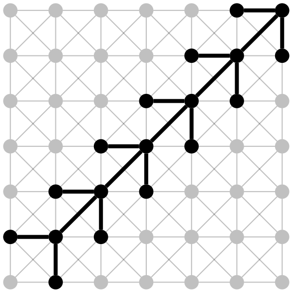
Question: Are there trees of arbitrarily large metric dimension whose threshold dimension is 2?

Answer: Yes.



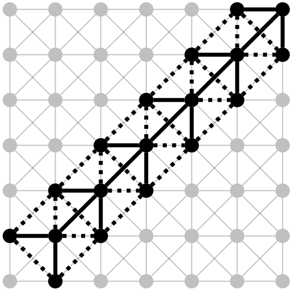
Question: Are there trees of arbitrarily large metric dimension whose threshold dimension is 2?

Answer: Yes.



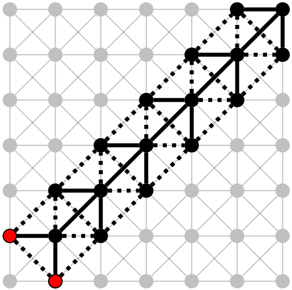
Question: Are there trees of arbitrarily large metric dimension whose threshold dimension is 2?

Answer: Yes.



Question: Are there trees of arbitrarily large metric dimension whose threshold dimension is 2?

Answer: Yes.



Future Prospects

Future Prospects

- Complexity

Future Prospects

- Complexity
  - Is the problem of determining the threshold dimension NP-hard?

FUTURE PROSPECTS

- Complexity
  - Is the problem of determining the threshold dimension NP-hard?
  - Is there a polynomial time algorithm for determining the threshold dimension of a tree?

Future Prospects

- Complexity
  - Is the problem of determining the threshold dimension NP-hard?
  - Is there a polynomial time algorithm for determining the threshold dimension of a tree?
- How much difference can a single edge make?

Future Prospects

- Complexity
  - Is the problem of determining the threshold dimension NP-hard?
  - Is there a polynomial time algorithm for determining the threshold dimension of a tree?
- How much difference can a single edge make?
  - Theorem (Chartrand et al., 2000): If H is obtained from a tree T by adding a single edge, then

$$\beta(T) - 2 \leq \beta(H) \leq \beta(T) + 1.$$

Future Prospects

- Complexity
  - Is the problem of determining the threshold dimension NP-hard?
  - Is there a polynomial time algorithm for determining the threshold dimension of a tree?
- How much difference can a single edge make?
  - Theorem (Chartrand et al., 2000): If H is obtained from a tree T by adding a single edge, then

$$\beta(T) - 2 \leq \beta(H) \leq \beta(T) + 1.$$

  - The proof relies heavily on properties of trees. What can be said for general graphs?

Thank you!