JOURNAL OF FUNDAMENTAL MATHEMATICS
AND APPLICATIONS (JFMA) VOL. 7 NO. 2 (NOV 2024)
Available online at www.jfma.math.fsm.undip.ac.id

BOUNDED TREE-DEPTH, PATH-DISTANCE-WIDTH, AND LINEAR-WIDTH OF GRAPHS

Takaaki Fujita[1*]

[1]*Independence Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.*
Email: [1]t171d603@gunma-u.ac.jp
*Corresponding author

Abstract. The study of width parameters and related graph parameters is an active area of research in graph theory. In this brief paper, we explore the upper and lower bounds of graph parameters, including path-distance-width, tree-distance-width, tree-depth, and linear-width. These bounds are crucial for understanding the complexity and structure of graphs.

Keywords: Path-distance-width, Linear-width, Tree-depth, Graph Width

I. INTRODUCTION

Graph theory studies networks of vertices (nodes) and edges, modeling relationships and structures in both mathematics and real-world systems [1, 2]. In graph theory, graph parameters quantify structural properties, such as width, depth, and connectivity, which significantly influence computational complexity and algorithmic performance [3]. The study of graph width parameters is a well-established area in graph theory, focusing on metrics that measure the structural complexity of graphs. Parameters such as tree-width [4], path-width [5], and branch-width [6, 7] are essential for understanding the computational complexity of various problems [8, 9, 10, 11]. These parameters, as discussed later, have been extensively studied in terms of their bounds for reasons outlined in subsequent sections.

In this brief paper, we explore the relationships and establish upper and lower bounds on graph parameters, including path-distance-width, tree-distance-width, tree-depth, and linear-width. Path-distance-width measures the size of vertex sets in a decomposition along a path, ensuring graph edge coverage with minimal overlap [12, 13]. Tree-depth represents the minimum height of a rooted forest whose closure contains the graph as a subgraph, reflecting graph hierarchy[14, 15]. Linear-width calculates the maximum number of vertices shared between consecutive edge partitions in a linear edge ordering of the graph[7]. Understanding these bounds is essential for comprehending the complexity and structural properties of graphs (cf. [16, 17]).

Our contributions are as follows:

1. **Establishing Relationships Between Graph Parameters:**
   - We prove that if a graph G has bounded path-distance-width k, then it also has bounded linear-width $lw(G) \leq k$ (Theorem 1).
   - We demonstrate that if a graph G has bounded tree-depth h, then it has bounded path-distance-width $\text{PDW}(G) \leq h$, linear-width $lw(G) \leq h$, and tree-distance-width $\text{TDW}(G) \leq h$ (Theorems 2, 4, and corresponding results).

2. **Implications for Algorithmic Complexity and Graph Decomposition:**
    - Our results have significant implications for algorithm design and graph decomposition strategies. Specifically, they indicate that algorithms optimized for graphs with bounded tree-depth or path-distance-width can be applied more broadly under certain conditions.

Note that the importance of establishing these bounds lies in several key areas:

1. **Algorithmic Complexity:**
    - **Upper Bounds** help determine whether problems can be solved efficiently. If a graph's width parameter has a low upper bound, algorithms that are generally difficult on arbitrary graphs may become more tractable.
    - **Lower Bounds** indicate the intrinsic difficulty of problems. A high lower bound suggests that certain computational problems will remain challenging, regardless of algorithmic optimizations.
2. **Graph Decomposition:**
    - **Upper Bounds** enable efficient graph decomposition into simpler structures like trees or paths, which are easier to manage computationally.
    - **Lower Bounds** highlight the minimum complexity required for any decomposition, reflecting inherent structural challenges within the graph.
3. **Graph Class Comparison:**
    - Comparing upper and lower bounds across different graph classes reveals structural differences. This can guide the development of specialized algorithms tailored to specific classes of graphs.
4. **Optimization and Approximation:**
    - **Upper Bounds** can lead to the development of efficient approximation algorithms by limiting the scope of computationally intensive tasks.
    - **Lower Bounds** demonstrate the difficulty of achieving near-optimal solutions, informing researchers about the potential need for heuristic or approximate methods.

II. PRELIMINARIES AND DEFINITIONS

This section presents an overview of the fundamental definitions and notations used throughout the paper. Additionally, several examples are provided.

2.1. Simple Notation in this paper

Consider a simple undirected graph G, where the vertex set is denoted by $V(G)$ and the edge set by $E(G)$. For simplicity, we will often write $G = (V, E)$, with V representing the vertices and E the edges. If X is a subset of the vertices $V(G)$ (or the edges $E(G)$), then X^c represents the complement set $V(G) \setminus X$ (or $E(G) \setminus X$, respectively), which includes all elements not in X. For other basic concepts in graph theory, please refer to [18].

2.2. Graph width parameter

In this paper, we consider about follwoing definitions of graph parameters.

Definition 1 *[7] A branch decomposition of a graph $G = (V, E)$ is a pair (T, σ), where T is a tree with vertices of degree at most 3, and σ is a bijection from the set of leaves of T to E. The width of an edge e in T is the number of vertices v in V such that there exist leaves t_1 and t_2 in T that are in different components of $T[E(T) \setminus \{e\}]$ with $\sigma(t_1)$ and $\sigma(t_2)$ both incident to v. The width of (T, σ) is the maximum width over all edges of T. The branch width, bw(G), of a graph G is the minimum width over all its branch decompositions. If $|E(G)| \leq 1$, the branch width of G is zero by definition.*

To define linear width, let $G = (V, E)$ be a graph with $|E| = m$. The linear width, lw(G), of G is defined as the smallest integer $k \geq 0$ such that the edges of G can be arranged in a linear ordering $(e_1, \ldots, e_m)$ in such a way that for every $i = 1, \ldots, m-1$, there are at most k vertices incident to edges that belong both to $(e_1, \ldots, e_i)$ and to $(e_{i+1}, \ldots, e_m)$. Linear orders over the edges of a graph and branch decompositions have a relationship that resembles the one between tree decompositions and path decompositions.

Example 1 Consider the cycle graph $G = (V, E)$ with vertices $V = \{v_1, v_2, v_3, v_4\}$ and edges $E = \{e_1, e_2, e_3, e_4\}$, where:

- $e_1 = \{v_1, v_2\}$
- $e_2 = \{v_2, v_3\}$
- $e_3 = \{v_3, v_4\}$
- $e_4 = \{v_4, v_1\}$

We construct a branch decomposition (T, σ):

- Tree T is a binary tree with four leaves corresponding to the edges of G.
- Bijection σ maps leaves to edges:
  - $\sigma(t_1) = e_1$
  - $\sigma(t_2) = e_2$
  - $\sigma(t_3) = e_3$
  - $\sigma(t_4) = e_4$

The width of each edge in T is the number of vertices shared between the two parts when the edge is removed. In this example, the maximum width is 2, so the branch width of G is 2.

Definition 2 *[12, 13] A tree distance decomposition of a graph $G = (V, E)$ is a triple $(\{X_i \mid i \in I\}, T = (I, F), r)$, where*

- *$\bigcup_{i \in I} X_i = V(G)$ and, for all $i \neq j$, $X_i \cap X_j = \emptyset$,*

- *For each* $v \in V$*, if* $v \in X_i$*, then* $d_G(X_r, v) = d_T(r, i)$*,*
- *For each edge* $\{v, w\} \in E$*, there exist* $i, j \in I$ *such that* $v \in X_i$*,* $w \in X_j$*, and either* $i = j$ *or* $\{i, j\} \in F$*,*
- $r \in I$.

The node r *is called the root of the tree* T*, and* X_r *is called the root set of the tree distance decomposition. The width of a tree distance decomposition* $(\{X_i \mid i \in I\}, T, r)$ *is equal to* $\max_{i \in I} |X_i|$*. The tree distance width of a graph G is the minimum width over all possible tree distance decompositions of G. The corresponding graph parameter is denoted by TDW.*

A rooted tree distance decomposition of a graph $G = (V, E)$ *is a tree distance decomposition* $(\{X_i \mid i \in I\}, T = (I, F), r)$ *of G in which* $|X_r| = 1$*. The rooted tree distance width of a graph G is the minimum width over all rooted tree distance decompositions. The corresponding graph parameter is denoted by RTDW.*

The (rooted) path distance decomposition and the parameter of (rooted) path distance width of a graph $G = (V, E)$ *are defined similarly to the (rooted) tree distance decomposition and (rooted) tree distance width, but now the tree* T *is required to be a path, and the root has degree one in* T*. For simplicity, we denote a (rooted) path distance decomposition as* $(X_1, X_2, \ldots, X_t)$*, where* X_1 *is the root set of the decomposition. We denote the corresponding graph parameters by PDW and RPDW, respectively.*

Example 2 Consider the path graph $G' = (V', E')$ with $V' = \{v_1, v_2, v_3, v_4\}$ and edges:

- $\{v_1, v_2\}$
- $\{v_2, v_3\}$
- $\{v_3, v_4\}$

We construct a tree-distance decomposition $(\{X_i\}, T, r)$:

- Tree T is a rooted tree with nodes $I = \{i_1, i_2, i_3, i_4\}$ connected linearly.
- Root $r = i_1$.
- Assign vertex sets:
  - $X_{i_1} = \{v_1\}$
  - $X_{i_2} = \{v_2\}$
  - $X_{i_3} = \{v_3\}$
  - $X_{i_4} = \{v_4\}$

This decomposition satisfies the conditions of a tree-distance decomposition with width 1.

Example 3 Using the same graph G', we construct a path-distance decomposition:

- Path T consists of nodes i_1, i_2, i_3, i_4 connected sequentially.
- Root $r = i_1$.
- Assign vertex sets:
  - $X_{i_1} = \{v_1\}$
  - $X_{i_2} = \{v_2\}$
  - $X_{i_3} = \{v_3\}$
  - $X_{i_4} = \{v_4\}$

This satisfies the conditions of a path-distance decomposition with width 1.

Definition 3 *[14, 15] A rooted forest is a disjoint union of rooted trees. The height of a vertex v in a rooted forest F is the number of vertices in the path from the root (of the tree to which v belongs) to v and is denoted by $h(v)$. The height of F is the maximum height of the vertices in F. Let u and v be vertices of F. The vertex u is an ancestor of v in F if u belongs to the path connecting v to the root of the tree in F to which v belongs. The closure $cl(F)$ of a rooted forest F is the graph with vertex set $V(F)$ and edge set $\{\{u, v\} \mid u$ is an ancestor of v in $F\}$. A rooted forest F defines a partial order on its set of vertices: $u \leq_F v$ if u is an ancestor of v in F. The comparability graph of this partial order is clearly $cl(F)$.*

The tree-depth td(G) of a graph G is the minimum height of a rooted forest F such that the closure $cl(F)$ contains G as a subgraph.

Example 4 Consider the graph $G = (V, E)$ where:

- $V = \{v_1, v_2, v_3\}$
- $E = \{\{v_1, v_2\}, \{v_2, v_3\}\}$

This is a path graph on three vertices.

Our goal is to find the minimum height of a rooted forest F such that the closure $cl(F)$ contains G as a subgraph.

Attempt with Height 1 At height 1, each vertex is a root of its own tree:

- Tree 1: root v_1
- Tree 2: root v_2
- Tree 3: root v_3

In this forest:

- There are no ancestor-descendant relationships.
- The closure $cl(F)$ has no edges.

Since $cl(F)$ has no edges, it does not contain G.

Attempt with Height 2 Construct a rooted forest F with height 2:

- Tree:
  - Root v_2
  - Children v_1 and v_3

In this forest:

- v_1 and v_3 are descendants of v_2.
- Ancestor-descendant pairs are (v_2, v_1) and (v_2, v_3).

The closure $cl(F)$ includes edges:

- $\{v_2, v_1\}$ and $\{v_2, v_3\}$ (since v_2 is an ancestor of both v_1 and v_3).

Edges in G:

- $\{v_1, v_2\}$ is in $cl(F)$.
- $\{v_2, v_3\}$ is in $cl(F)$.

Thus, $cl(F)$ contains G as a subgraph.

The minimal height achieved is 2. Therefore, the tree-depth of G is:

$$\text{td}(G) = 2$$

III. MAIN RESULT OF THIS PAPER

In this section, we discuss the relationships between various graph width parameters, specifically focusing on upper and lower bounds.

Theorem 1 *If a graph G has bounded path-distance-width, then it has bounded linear-width.*

Proof. Given that G has path-distance-width k, there exists a path-distance decomposition $(X_1, X_2, \ldots, X_t)$ where each $X_i \subseteq V(G)$, $X_i \cap X_j = \emptyset$ for $i \neq j$, and $|X_i| \leq k$ for all i.

We construct a linear ordering of the edges of G as follows:

- For each $i = 1$ to $t - 1$:
  - List all edges with both endpoints in X_i.
  - List all edges between X_i and X_{i+1}.
- Finally, list all edges with both endpoints in X_t.

Let $E(G) = \{e_1, e_2, \ldots, e_m\}$ be this ordered list of edges. For each $1 \leq i < m$, consider the partition of $E(G)$ into $P_i = \{e_1, e_2, \ldots, e_i\}$ and $S_i = \{e_{i+1}, e_{i+2}, \ldots, e_m\}$.

Define $V_{\text{prefix}} = \{v \in V(G) \mid v \text{ is incident to an edge in } P_i\}$ and $V_{\text{suffix}} = \{v \in V(G) \mid v \text{ is incident to an edge in } S_i\}$.

The *linear-width* $lw(G)$ is the maximum size of $V_{\text{prefix}} \cap V_{\text{suffix}}$ over all i:

$$lw(G) = \max_{1 \leq i < m} |V_{\text{prefix}} \cap V_{\text{suffix}}|.$$

At each partition point between edges in our ordering, the overlap $V_{\text{prefix}} \cap V_{\text{suffix}}$ consists of vertices that are in X_j where X_j is the boundary between the prefix and the suffix.

Since each X_j has at most k vertices, we have:

$$|V_{\text{prefix}} \cap V_{\text{suffix}}| \leq k.$$

Therefore, $lw(G) \leq k$, which completes the proof. □

Theorem 2 *If a graph G has bounded tree-depth h, then it has bounded path-distance-width.*

Proof. Given that G has tree-depth h, there exists a rooted forest F of height h such that the closure $cl(F)$ contains G as a subgraph.

We can construct a path-distance decomposition of G with width at most h as follows:

- Perform a depth-first traversal of the rooted forest F, listing the vertices in the order they are first visited. Let this ordering be $v_1, v_2, \ldots, v_n$.
- Partition the vertices into sets X_i, where each X_i consists of a single vertex v_i. Since each X_i has size $1 \leq h$, the width of the decomposition is at most h.
- The path T in the decomposition corresponds to the sequence $X_1, X_2, \ldots, X_n$.

For each edge $\{u, v\} \in E(G)$:

- Since $cl(F)$ contains G, u and v are related in F as ancestor and descendant or share a common ancestor.
- Therefore, the distance between u and v in G corresponds to their positions in the traversal, and they will either be in the same X_i or in consecutive ones.

This satisfies the conditions of a path-distance decomposition. Thus, PDW$(G) \leq h$. □

Example 1 (Example of Theorem 2) Consider the star graph S_n with center vertex v_0 connected to n leaves $v_1, v_2, \ldots, v_n$.

We can construct a rooted tree F with root v_0 and leaves $v_1, v_2, \ldots, v_n$. The height of F is $h = 2$, so td$(S_n) = 2$.

Perform a depth-first traversal of F, obtaining the ordering:

$$v_0, v_1, v_2, \ldots, v_n.$$

Create the path-distance decomposition:

$$X_1 = \{v_0\}, \quad X_2 = \{v_1\}, \quad X_3 = \{v_2\}, \quad \ldots, \quad X_{n+1} = \{v_n\}.$$

Each X_i has size $1 \leq h$.

Since $\text{PDW}(S_n) \leq h = 2$, this example demonstrates Theorem 2.

Theorem 3 *It is not possible to directly determine the tree-depth of a graph solely from its path-distance-width k.*

Proof. We provide two graphs with the same path-distance-width but different tree-depths.

Example 1: Path Graph P_n

- P_n is a path with n vertices.
- Path-distance-width $\text{PDW}(P_n) = 1$, since the graph itself is a path.
- Tree-depth $\text{td}(P_n) = n$, because any rooted tree containing P_n as a subgraph must have height at least n.

Example 2: Star Graph S_n

- S_n is a star with $n + 1$ vertices (one central vertex connected to n leaves).
- Path-distance-width $\text{PDW}(S_n) = 1$, since we can arrange the central vertex and leaves linearly.
- Tree-depth $\text{td}(S_n) = 2$, as the minimal rooted tree has height 2.

Both graphs have PDW $= 1$, but $\text{td}(P_n) = n$ and $\text{td}(S_n) = 2$. This shows that tree-depth cannot be directly determined from path-distance-width. □

Theorem 4 *If a graph G has bounded tree-depth, then it has bounded linear-width.*

Proof. From Theorems 1 and 2, we have:

- $\text{td}(G) = h \implies \text{PDW}(G) \leq h$ (from Theorem 2).
- $\text{PDW}(G) \leq h \implies lw(G) \leq h$ (from Theorem 1).

Therefore, $\text{td}(G) = h \implies lw(G) \leq h$. □

Example 2 (Example of theorem 4) Using the star graph S_n from the previous example:

Tree-Depth As before, $\text{td}(S_n) = 2$.

Linear Ordering of Edges We list the edges connecting the center to the leaves:

$$e_1 = \{v_0, v_1\}, \quad e_2 = \{v_0, v_2\}, \quad \ldots, \quad e_n = \{v_0, v_n\}.$$

Calculating Linear-Width At any partition between e_i and e_{i+1}:

- V_{prefix} includes v_0 and $v_1, \ldots, v_i$.
- V_{suffix} includes v_0 and $v_{i+1}, \ldots, v_n$.
- Intersection $V_{\text{prefix}} \cap V_{\text{suffix}} = \{v_0\}$.

Thus, $|V_{\text{prefix}} \cap V_{\text{suffix}}| = 1 \leq h$.

Since $lw(S_n) \leq \text{td}(S_n) = 2$, this example supports Theorem 4.

Theorem 5 *If a graph G has bounded tree-depth, then it has bounded tree-distance-width.*

Proof. Using the rooted forest F of height h associated with G's tree-depth, we construct a tree-distance decomposition.

- Let T be the tree structure of F.
- For each vertex $v \in V(G)$, assign it to the set X_i where i corresponds to its depth $d_T(r, v)$ in T.
- Since the maximum depth is h, the number of different sets X_i is at most h.

Edges in G are covered because $cl(F)$ contains G, and the distance conditions are satisfied by construction.

The width of this tree-distance decomposition is the maximum size of any X_i, which is at most h. Therefore, $\text{TDW}(G) \leq h$. □

Example 3 (Example of theorem 5) Again, consider the star graph S_n:

Tree-Depth $\text{td}(S_n) = 2$.

Tree-Distance Decomposition We construct a tree T identical to the rooted tree used for tree-depth, with root v_0 and children $v_1, \ldots, v_n$.

Define $X_1 = \{v_0\}$ and $X_2 = \{v_1, v_2, \ldots, v_n\}$.

Width Calculation The maximum size of X_i is:

$$|X_2| = n.$$

However, to ensure the width is bounded by h, we can adjust the decomposition:

Assign each leaf to its own set:

$$X_1 = \{v_0\}, \quad X_2 = \{v_1\}, \quad X_3 = \{v_2\}, \quad \ldots, \quad X_{n+1} = \{v_n\}.$$

The tree T now has a path structure.

Width Calculation Each X_i has size $1 \leq h$.

Thus, $\text{TDW}(S_n) \leq h = 2$, illustrating the theorem.

IV. CONCLUSIONS AND FUTURE RESEARCH DIRECTION

We showed the proof of the upper and lower bounds on graph parameters, including path-distance-width, tree-distance-width, tree-depth, and linear-width. In the future, we consider about upper bound and lower bound of other graph parameters such as Tree-partition-width and Path-Partition-width.

REFERENCES

[1] R. Diestel, *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[2] R. Hashemi, H. Darabi, M. Hashemi, and J. Wang, “Graph theory in ecological network analysis: A systematic review for connectivity assessment,” *Journal of Cleaner Production*, vol. 472, p. 143504, 2024.

[3] X. Yu, Y. Wu, F. Meng, X. Zhou, S. Liu, Y. Huang, and X. Wu, “A review of graph and complex network theory in water distribution networks: Mathematical foundation, application and prospects,” *Water Research*, p. 121238, 2024.

[4] N. Robertson and P. D. Seymour, “Graph minors. iii. planar tree-width,” *Journal of Combinatorial Theory, Series B*, vol. 36, no. 1, pp. 49–64, 1984.

[5] N. Robertson and P. D. Seymour, “Graph minors. i. excluding a forest,” *Journal of Combinatorial Theory, Series B*, vol. 35, no. 1, pp. 39–61, 1983.

[6] J. Geelen, B. Gerards, and G. Whittle, “Obstructions to branch-decomposition of matroids,” *Journal of Combinatorial Theory, Series B*, vol. 96, no. 4, pp. 560–570, 2006.

[7] N. Robertson and P. D. Seymour, “Graph minors. x. obstructions to tree-decomposition,” *Journal of Combinatorial Theory, Series B*, vol. 52, no. 2, pp. 153–190, 1991.

[8] F. V. Fomin, P. Fraigniaud, P. Montealegre, I. Rapaport, and I. Todinca, “Distributed model checking on graphs of bounded treedepth,” *arXiv preprint arXiv:2405.03321*, 2024.

[9] F. V. Fomin, P. Fraigniaud, P. Montealegre, I. Rapaport, and I. Todinca, "Brief announcement: Distributed model checking on graphs of bounded treedepth," in *Proceedings of the 43rd ACM Symposium on Principles of Distributed Computing*, pp. 205–208, 2024.

[10] A. Kaznatcheev, "Local search for valued constraint satisfaction parameterized by treedepth," *arXiv preprint arXiv:2405.12410*, 2024.

[11] M. Lampis and M. Vasilakis, "Structural parameterizations for two bounded degree problems revisited," *ACM Transactions on Computation Theory*, vol. 16, no. 3, pp. 1–51, 2024.

[12] K. Yamazaki, "On approximation intractability of the path–distance–width problem," *Discrete applied mathematics*, vol. 110, no. 2–3, pp. 317–325, 2001.

[13] Y. Otachi, "Isomorphism for graphs of bounded connected-path-distance-width," in *International Symposium on Algorithms and Computation*, (Berlin, Heidelberg), Springer Berlin Heidelberg, 2012.

[14] J. Nešetřil and P. O. De Mendez, "Tree-depth, subgraph coloring and homomorphism bounds," *European Journal of Combinatorics*, vol. 27, no. 6, pp. 1022–1041, 2006.

[15] M. DeVos, O.-j. Kwon, and S.-i. Oum, "Branch-depth: Generalizing tree-depth of graphs," *European Journal of Combinatorics*, vol. 90, p. 103186, 2020.

[16] R. Sasak, "Comparing 17 graph parameters," Master's thesis, The University of Bergen, 2010.

[17] D. J. Harvey and D. R. Wood, "Parameters tied to treewidth," *Journal of Graph Theory*, vol. 84, no. 4, pp. 364–385, 2017.

[18] D. B. West *et al.*, *Introduction to graph theory*, vol. 2. Prentice hall Upper Saddle River, 2001.