JOURNAL OF FUNDAMENTAL MATHEMATICS
AND APPLICATIONS (JFMA) VOL. 8 NO. 1 (2025)
Available online at www.jfma.math.fsm.undip.ac.id

Bounding Linear-width and Distance-width Using Feedback Vertex Set and MM-width for Graph

Takaaki Fujita

Gunma University, Japan
Email: t171d603@gunma-u.ac.jp

Abstract. Studying the upper and lower bounds of graph parameters is crucial for understanding the complexity and tractability of computational problems, optimizing algorithms, and revealing structural properties of various graph classes. In this brief paper, we explore the upper and lower bounds of graph parameters, including path-distance-width, MM-Width, Feedback Vertex Set, and linear-width. These bounds are crucial for understanding the complexity and structure of graphs.
Keywords: Path-distance-width, Linear-width, Maximum-matching-width, Feedback Vertex Set

I. Introduction

1.1. Graph Parameters

It is well-known that real-world concepts can be modeled using graph theory, and efficient algorithms have been developed for this purpose[1, 2]. When studying problems in graph algorithms and graph theory, it is common to focus on specific graph classes or parameters to find solutions[3, 4, 5, 6, 7]. Additionally, extensive research has been conducted to clarify the relationships between different graph classes and parameters [8, 9, 10, 11, 12].

The study of graph width parameters is a well-established area in graph theory, focusing on metrics that measure the structural complexity of graphs [13, 14, 15, 16]. Parameters such as tree-width [17, 18, 19, 20, 21], path-width [18, 22, 23], proper-path-width[24, 25], rank-width [26, 27], hypertree-width[28, 29, 30], superhypertree-width[31, 32, 33], and branch-width [34, 19, 35] are crucial for understanding the computational complexity of various problems.

Path-distance-width (PDW) [36, 37] is another graph parameter that extends path-width by incorporating a distance measure. It is known that determining whether a given graph has $PDW(T) \leq k$ when the input graph is restricted to trees is NP-complete [37]. Additionally, concepts of connected-path-distance-width and tree-distance-width have been introduced in the literature as related measures [37]. These parameters are also relevant in the context of graph isomorphism problems, where the structural complexity of graphs plays a crucial role in determining isomorphism efficiently (cf.[38, 39, 40]).

Linear-width [41, 42, 43, 44] is a graph parameter that measures the minimum width in a linear ordering of edges, ensuring that the number of vertices incident to edges across any cut in the order is minimized. These "linear" restrictions to underlying path structures are often beneficial in proving results for general parameters[45]. Moreover, these linear parameters offer valuable insights from a structural perspective, especially in the study of special graph classes. Linear-width is known to be extendable to matroids and connectivity systems[46, 47, 34].

Maximum-matching-width[48, 49, 50] is a graph parameter defined using a decomposition tree, where the width is determined by the size of the largest maximum matching in bipartite subgraphs induced by edge cuts. The Feedback Vertex Set[51, 52, 53] is the smallest set of vertices in a graph whose removal results in an acyclic graph, effectively eliminating all cycles and reducing the graph to a forest.

1.2. Our Contribution

In the aforementioned fields, the study of graph parameters themselves, as well as clarifying their relationships and upper and lower bounds, is considered one of the important research topics. In this brief paper, we explore the upper and lower bounds of graph parameters, including path-distance-width, Feedback Vertex Set, maximum-matching-width, and linear-width.

II. Preliminaries

This section presents the mathematical definitions of the relevant concepts.

2.1. Simple Notation in this paper

The following provides the fundamental definitions in graph theory. For additional fundamental graph-theoretic concepts, we refer the reader to [2, 54, 55, 56].

Definition 1 (Graph) *[2] A* graph *G is an ordered pair $G = (V, E)$, where:*

- *$V(G)$ is the set of vertices (or nodes),*
- *$E(G)$ is the set of edges, which are unordered pairs of distinct vertices.*

For simplicity, we often write $G = (V, E)$ when the context is clear. Throughout this paper, we consider only simple, undirected graphs, meaning there are no multiple edges or self-loops.

If X is a subset of $V(G)$ or $E(G)$, its complement is denoted by X^c, defined as:

$$X^c = V(G) \setminus X \quad \textit{or} \quad X^c = E(G) \setminus X,$$

depending on whether X refers to a vertex or edge subset.

Definition 2 (Subgraph) *[2] Given a graph $G = (V, E)$, a* subgraph *H of G is a graph $H = (V_H, E_H)$ such that:*

- *$V_H \subseteq V(G)$,*
- *$E_H \subseteq E(G)$, where each edge in E_H connects vertices in V_H.*

If $V_H = V(G)$, we say H is a spanning subgraph *of G.*

Definition 3 (Tree) *[2] A* tree *$T = (V, E)$ is a connected, acyclic graph, meaning that for every pair of vertices $u, v \in V$, there exists exactly one path between u and v. Equivalently, a tree is a graph with $|V| - 1$ edges and no cycles.*

Definition 4 (Cycle Graph) *(cf.[57]) A* cycle graph*, denoted by C_n, is a graph that consists of a single cycle containing n vertices. Formally,*

$$V(C_n) = \{v_1, v_2, \dots, v_n\}$$

and

$$E(C_n) = \{\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{n-1}, v_n\}, \{v_n, v_1\}\}.$$

Every vertex in C_n has degree 2.

Definition 5 (Path Graph) *(cf.[58]) A* path graph*, denoted by P_n, is a graph that consists of a single path with n vertices. That is,*

$$V(P_n) = \{v_1, v_2, \dots, v_n\}$$

and

$$E(P_n) = \{\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{n-1}, v_n\}\}.$$

In P_n, the two endpoints have degree 1 while all other vertices have degree 2.

Definition 6 (Distance in a Graph) *(cf.[2]) Let $G = (V, E)$ be a graph. The* distance *between two vertices $u, v \in V$, denoted by $d_G(u, v)$, is the length of the shortest path connecting u and v in G. If no such path exists, the distance is defined as ∞.*

Example 1 (Distance in a Path Graph) Consider the path graph P_5 with vertices

$$V(P_5) = \{v_1, v_2, v_3, v_4, v_5\}$$

and edges

$$E(P_5) = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_5\}\}.$$

The shortest path between v_1 and v_4 is (v_1, v_2, v_3, v_4), which has length 3. Thus, the distance between v_1 and v_4 is

$$d_{P_5}(v_1, v_4) = 3.$$

Definition 7 (Separator) *(cf.[59, 60]) Let $G = (V, E)$ be a graph. A* separator *in G is a subset $S \subseteq V$ such that removing the vertices in S (and the edges incident to them) disconnects G into two or more connected components. In particular, given two disjoint subsets $A, B \subseteq V$, a set S is called an (A, B)*-separator *if every path in G from a vertex in A to a vertex in B contains at least one vertex from S.*

Example 2 (Separator in a Path Graph) Consider the path graph P_5 with vertices

$$V(P_5) = \{v_1, v_2, v_3, v_4, v_5\}$$

and edges

$$E(P_5) = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_5\}\}.$$

If we choose $S = \{v_3\}$, then removing S disconnects P_5 into two components: one containing $\{v_1, v_2\}$ and the other containing $\{v_4, v_5\}$. Hence, $S = \{v_3\}$ is a separator in P_5, as it ensures that no path remains between the two disconnected parts.

Example 3 ((A,B)-Separator in a Cycle Graph) Consider the cycle graph C_6 with vertices

$$V(C_6) = \{v_1, v_2, v_3, v_4, v_5, v_6\}$$

and edges forming a cycle

$$E(C_6) = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_5\}, \{v_5, v_6\}, \{v_6, v_1\}\}.$$

Suppose we want to separate two disjoint subsets $A = \{v_1\}$ and $B = \{v_4\}$. The set $S = \{v_2, v_3\}$ is an (A, B)-separator because every path from v_1 to v_4 must pass through at least one vertex in S. Removing S disconnects A from B, making it a valid separator.

Definition 8 (Bipartite Graph) *(cf.[61, 62]) A graph $G = (V, E)$ is called a* bipartite graph *if its vertex set V can be partitioned into two disjoint subsets V_1 and V_2 such that every edge in E has one endpoint in V_1 and the other in V_2. Formally, there exists a partition $V = V_1 \cup V_2$ with $V_1 \cap V_2 = \emptyset$ such that:*

$$E \subseteq \{(u, v) \mid u \in V_1, v \in V_2\}.$$

If no such partition exists, the graph is said to be non-bipartite.

2.2. Graph Parameters and Graph Width Parameters

In this paper, we consider the following definitions.

Definition 9 (Branch Decomposition [19]) *Let $G = (V, E)$ be a graph. A* branch decomposition *of G is a pair (T, σ) satisfying:*

1. *T is a tree in which every vertex has degree at most 3.*
2. *σ is a bijection from the set of leaves of T onto the edge set E of G.*

For any edge $e \in E(T)$, removal of e disconnects T into two subtrees. The width *of e is defined as the number of vertices $v \in V$ for which there exist two leaves t_1 and t_2 located in different connected components of $T \setminus \{e\}$ such that the corresponding edges $\sigma(t_1)$ and $\sigma(t_2)$ are both incident with v in G. The width of the branch decomposition (T, σ) is the maximum width over all edges of T. Finally, the* branch width *of G, denoted bw(G), is the minimum width over all branch decompositions of G. By convention, if $|E(G)| \leq 1$, then bw$(G) = 0$.*

Example 4 (Branch-Width of a Cycle Graph) Consider the cycle graph C_4 with vertices

$$V(C_4) = \{v_1, v_2, v_3, v_4\}$$

and edges

$$E(C_4) = \{e_1 = \{v_1, v_2\},\ e_2 = \{v_2, v_3\},\ e_3 = \{v_3, v_4\},\ e_4 = \{v_4, v_1\}\}.$$

A branch decomposition (T, σ) of C_4 can be constructed as follows. Let T be the tree with four leaves arranged in a binary structure: for instance, let T have two internal nodes u and w such that u is adjacent to leaves t_1 and t_2, w is adjacent to leaves t_3 and t_4, and u and w are connected by an edge. Define the bijection σ by

$$\sigma(t_1) = e_1, \quad \sigma(t_2) = e_2, \quad \sigma(t_3) = e_3, \quad \sigma(t_4) = e_4.$$

For each edge in T, its removal splits T into two subtrees whose corresponding sets of edges in C_4 induce a vertex separator. One may verify that every such separator involves at most 2 vertices. Hence, the width of this branch decomposition is 2, and it can be shown that the branch-width $\mathrm{bw}(C_4) = 2$.

Definition 10 (Linear Decomposition (Linear Width)) *[42, 44] Let $G = (V, E)$ be a graph with $|E| = m$. A* linear decomposition *of G is an ordering of its edges, denoted by*

$$(e_1, e_2, \ldots, e_m).$$

For each index i with $1 \leq i < m$, define

$$S_i = \{\, v \in V : v \text{ is incident to an edge in } \{e_1, \ldots, e_i\} \text{ and also to an edge in } \{e_{i+1}, \ldots, e_m\}\,\}.$$

The width *of this linear decomposition is given by*

$$\max_{1 \leq i < m} |S_i|.$$

The linear width *of G, denoted $lw(G)$, is the smallest integer $k \geq 0$ for which there exists a linear decomposition of G with width k.*

Example 5 (Linear-Width of a Path Graph) Consider the path graph P_4 with vertices

$$V(P_4) = \{v_1, v_2, v_3, v_4\}$$

and edges

$$E(P_4) = \{e_1 = \{v_1, v_2\},\ e_2 = \{v_2, v_3\},\ e_3 = \{v_3, v_4\}\}.$$

Take the linear ordering of the edges as

$$(e_1, e_2, e_3).$$

For each index i with $1 \leq i < 3$, define the separator

$$S_i = \{\, v \in V(P_4) : v \text{ is incident to an edge in } \{e_1, \ldots, e_i\} \text{ and to an edge in } \{e_{i+1}, \ldots, e_3\}\,\}.$$

Then:

- For $i = 1$, the prefix is $\{e_1\}$ and the suffix is $\{e_2, e_3\}$. The common vertex is v_2, so $S_1 = \{v_2\}$ and $|S_1| = 1$.
- For $i = 2$, the prefix is $\{e_1, e_2\}$ and the suffix is $\{e_3\}$. The common vertex is v_3, so $S_2 = \{v_3\}$ and $|S_2| = 1$.

Thus, the linear-width of P_4 is

$$\mathrm{lw}(P_4) = \max\{|S_1|, |S_2|\} = 1.$$

Definition 11 (Tree Distance Decomposition and Tree Distance Width [36, 37]) *Let $G = (V, E)$ be a graph. A* tree distance decomposition *of G is a triple*

$$\Big(\{X_i \mid i \in I\},\ T = (I, F),\ r\Big),$$

which satisfies the following conditions:

1. ***Partition of vertices:*** *The collection $\{X_i\}_{i\in I}$ forms a partition of V; that is,*

$$\bigcup_{i\in I} X_i = V \quad \textit{and} \quad X_i \cap X_j = \emptyset \quad \textit{for every } i \neq j.$$

2. ***Tree structure:*** *$T = (I, F)$ is a tree with vertex set I and edge set F, and $r \in I$ is a designated root.*

3. ***Distance preservation:*** *For every vertex $v \in V$ with $v \in X_i$, the distance from v to the root set X_r in G is equal to the distance from r to i in T; that is,*

$$d_G(X_r, v) = d_T(r, i).$$

4. ***Edge connectivity:*** *For every edge $\{v, w\} \in E$, there exist indices $i, j \in I$ such that $v \in X_i$ and $w \in X_j$, and either $i = j$ or $\{i, j\} \in F$.*

The width *of a tree distance decomposition is defined as*

$$\max_{i\in I} |X_i|,$$

and the tree distance width *(TDW) of G is the minimum width over all such decompositions.*

A rooted tree distance decomposition *is a tree distance decomposition in which the root set X_r is a singleton (i.e., $|X_r| = 1$). The minimum width over all rooted tree distance decompositions defines the* rooted tree distance width *(RTDW) of G.*

Definition 12 (Path Distance Decomposition and Path Distance Width [36, 37]) *Let $G = (V, E)$ be a graph. A* path distance decomposition *of G is defined analogously to a tree distance decomposition, with the following modifications:*

1. *The tree T is required to be a* path*; that is, T is a tree in which every vertex has degree at most 2.*

2. *The designated root r in T is chosen such that it has degree one.*

In this setting, the decomposition can be represented as an ordered sequence of vertex subsets:

$$(X_1, X_2, \ldots, X_t),$$

where X_1 is the root set. The width *of a path distance decomposition is defined as*

$$\max_{1 \le i \le t} |X_i|,$$

and the path distance width *(PDW) of G is the minimum width over all such decompositions. Similarly, a rooted path distance decomposition is a path distance decomposition where the root set X_1 is a singleton (i.e., $|X_1| = 1$), and the corresponding graph parameter is denoted by RPDW.*

Example 6 (Path-Distance-Width of a Path Graph) Consider the path graph P_4 with vertices

$$V(P_4) = \{v_1, v_2, v_3, v_4\}$$

and edges

$$E(P_4) = \{\{v_1, v_2\},\ \{v_2, v_3\},\ \{v_3, v_4\}\}.$$

A natural rooted path distance decomposition of P_4 is given by the ordered sequence of vertex sets:

$$(X_1, X_2, X_3, X_4),$$

where we define

$$X_1 = \{v_1\}, \quad X_2 = \{v_2\}, \quad X_3 = \{v_3\}, \quad X_4 = \{v_4\}.$$

Here, X_1 is the root set (of size 1) and serves as the starting point. For each vertex $v \in X_i$, the distance from v to the root X_1 in P_4 is exactly $i - 1$, which matches the distance in the underlying path of the decomposition. The width of this decomposition is defined as

$$\max_{1 \le i \le 4} |X_i| = 1.$$

Thus, the path-distance-width of P_4 is $\mathrm{PDW}(P_4) = 1$.

Definition 13 (Feedback Vertex Set (FVS)) *[51, 52, 53] Let $G = (V, E)$ be a graph. A* feedback vertex set *(FVS) is a subset $S \subseteq V$ such that the graph obtained by removing all vertices in S (and their incident edges) is acyclic—that is, it becomes a forest. The size of the FVS is denoted by $|S|$, and in many applications one seeks an FVS of minimum size.*

Example 7 (Feedback Vertex Set (FVS)) Consider the cycle graph C_3 (a triangle) with vertices $\{v_1, v_2, v_3\}$ and edges $\{(v_1, v_2), (v_2, v_3), (v_3, v_1)\}$. Removing any single vertex (for instance, v_1) yields the graph with vertices $\{v_2, v_3\}$ and a single edge (v_2, v_3), which is acyclic. Thus, a minimal FVS for C_3 has size 1.

Definition 14 (Maximum Matching Decomposition) *[50] Let $G = (V, E)$ be a graph. A* maximum matching decomposition *of G is a pair (T, ∂) consisting of:*

1. *A tree T in which every vertex has degree at most 3.*
2. *A bijection $\partial : V \to L(T)$ that assigns each vertex of G to a unique leaf of T (here, $L(T)$ denotes the set of leaves of T).*

For each edge $e \in E(T)$, its removal splits T into two subtrees, say T_1 and T_2. These subtrees induce a partition of the vertices of G into

$$D_1 = \{\, v \in V : \partial(v) \in V(T_1) \,\} \quad \textit{and} \quad D_2 = \{\, v \in V : \partial(v) \in V(T_2) \,\}.$$

Let B_e be the bipartite subgraph of G consisting of all edges with one endpoint in D_1 and the other in D_2. The width *of the edge e is defined as the size of a maximum matching in B_e. The width of the decomposition (T, ∂) is the maximum width over all edges $e \in E(T)$. Finally, the* maximum matching-width *of G, denoted by MM-width(G), is the minimum width over all possible maximum matching decompositions of G.*

Example 8 (Maximum Matching Decomposition) Consider the path graph P_3 with vertices $\{v_1, v_2, v_3\}$ and edges $\{(v_1, v_2), (v_2, v_3)\}$. One possible maximum matching decomposition is as follows:

1. Choose T as a star tree with three leaves (one for each vertex) and a central internal node.
2. Define the bijection ∂ by mapping v_1, v_2, and v_3 to three distinct leaves of T.

Now, consider any edge e of T. Removing e partitions the leaves into two groups. For example, if the removal of e isolates the leaf corresponding to v_2 from the leaves corresponding to v_1 and v_3, then the induced partition of V is $D_1 = \{v_2\}$ and $D_2 = \{v_1, v_3\}$. The bipartite subgraph B_e then includes the edge (v_1, v_2) or (v_2, v_3) (depending on the precise structure of T). In either case, the maximum matching in B_e has size 1. As this holds for every edge in T, the width of this decomposition is 1, and hence MM-width$(P_3) = 1$.

III. Main result of this paper

We discuss about results of upper bounds and lower bounds.

3.1. Bounding Distance-width Using Feedback Vertex Set and MM-width

The following theorem presents bounds on Distance-width using the Feedback Vertex Set and MM-width.

Theorem 1 *Let G be a graph. Then:*

1. *Bounded path-distance-width does not necessarily imply that the size of a minimum feedback vertex set is bounded, i.e.,*

$$\textit{PDW}(G) \textit{ bounded} \not\Rightarrow \textit{FVS}(G) \textit{ bounded}.$$

2. *Conversely, if G has a bounded feedback vertex set, then its path-distance-width is bounded, i.e.,*

$$\textit{FVS}(G) \textit{ bounded} \Rightarrow \textit{PDW}(G) \textit{ bounded}.$$

Proof. We first address the claim that bounded path-distance-width does not imply a bounded feedback vertex set.

Counterexample: Consider a graph G that contains a large complete bipartite subgraph $K_{m,m}$ along with additional paths attached. It is possible to arrange the edges of G into a path-distance decomposition with small bags (i.e., with small PDW(G)) because the paths can be decomposed into vertex sets of bounded size. However, to eliminate all cycles within $K_{m,m}$, one must remove at least m vertices (for instance, by deleting an entire bipartition), implying that the minimum feedback vertex set has size at least m. By choosing m arbitrarily large, one obtains a graph with bounded path-distance-width but unbounded FVS.

Next, we prove that if G has a bounded feedback vertex set, then its path-distance-width is bounded.

Let S be a feedback vertex set of G with $|S| = k$, and let $G' = G - S$ be the acyclic graph (a forest) obtained by removing S. Since forests admit path-distance decompositions with small bags (in fact, each bag can be taken to have at most 2 vertices), we have

$$\text{PDW}(G') \leq 2.$$

Reintroducing the vertices in S into the decomposition increases the size of each bag by at most k. Hence, the path-distance-width of G satisfies

$$\text{PDW}(G) \leq \text{PDW}(G') + k \leq 2 + k.$$

Since k is fixed by the boundedness of the FVS, it follows that PDW(G) is bounded. □

Example 1 (Counterexample for Theorem 1 (Part 1)) Consider the graph G constructed as follows. Let G contain a complete bipartite subgraph $K_{m,m}$ with bipartition sets A and B (where $|A| = |B| = m$). To ensure connectivity, attach additional paths that join the vertices of $K_{m,m}$ into a single connected graph. One can design a path-distance decomposition for G with small bags (for instance, with width at most 2) by ordering the edges of the attached paths appropriately. However, because $K_{m,m}$ contains many cycles, any feedback vertex set must remove at least m vertices (for example, by deleting one of the bipartition sets entirely). Thus, even though PDW(G) is bounded by a constant, the size of the minimum feedback vertex set FVS(G) is at least m and can be made arbitrarily large by increasing m.

Example 2 (Example for Theorem 1 (Part 2)) Consider the graph G formed by taking a cycle C_5 and then removing one vertex to obtain a tree, and afterward reintroducing that vertex along with its incident edges to recreate the cycle. In this graph, the minimal feedback vertex set consists of a single vertex (removing that vertex breaks the cycle), so FVS$(G) = 1$. Since the acyclic graph $G - \{v\}$ (a tree) has a path-distance decomposition with width at most 2, reintroducing one vertex increases the bag sizes by at most 1. Hence, PDW$(G) \leq 2 + 1 = 3$. This demonstrates that if G has a bounded feedback vertex set, then its path-distance-width is also bounded.

Theorem 2 *If a graph G has bounded maximum matching-width, that is, if MM-width$(G) \leq k$ for some constant k, then the path-distance-width of G is also bounded.*

Proof. Assume that MM-width$(G) \leq k$. By definition, there exists a maximum matching decomposition (T, ∂) of G such that for every edge $e \in E(T)$, the corresponding bipartite subgraph B_e (induced by the partition of vertices defined by removing e from T) has a maximum matching of size at most k.

For each edge $e \in E(T)$, let D_1 and D_2 be the vertex sets corresponding to the two connected components of $T \setminus \{e\}$. The set of vertices incident to edges crossing between D_1 and D_2 forms a separator S_e in G. By Kőnig's theorem, the size of a minimum vertex cover of the bipartite graph B_e equals its maximum matching size, so there exists a vertex cover of B_e of size at most k. Even if S_e is not minimal, a straightforward argument shows that we can assume

$$|S_e| \leq 2k.$$

We now construct a path-distance decomposition of G by converting the tree T into a path (for example, by performing a depth-first traversal of T to obtain a linear ordering of the separators). In this decomposition, each bag X_i is associated with one of the separators S_e, and thus its size is bounded by $2k$. Consequently, the path-distance-width of G satisfies

$$\mathrm{PDW}(G) \leq 2k.$$

This completes the proof that bounded maximum matching-width implies bounded path-distance-width. □

Example 3 (Example for Theorem 2) Consider the path graph P_4 with vertex set

$$V(P_4) = \{v_1, v_2, v_3, v_4\}$$

and edge set

$$E(P_4) = \{e_1 = \{v_1, v_2\},\ e_2 = \{v_2, v_3\},\ e_3 = \{v_3, v_4\}\}.$$

A maximum matching decomposition of P_4 yields a maximum matching size of 1 for every separator, so MM-width$(P_4) = 1$. By performing a depth-first traversal of the corresponding decomposition tree, one obtains a path-distance decomposition in which each bag (associated with a separator) has size at most 2. Thus, PDW$(P_4) \leq 2$, confirming that bounded maximum matching-width implies bounded path-distance-width.

3.2. Bounding Linear-width Using Feedback Vertex Set and MM-width

The theorem concerning linear-width, bounded Feedback Vertex Set, and MM-width is as follows. It bears similarity to the proof of Path-distance-width discussed earlier.

Theorem 3 *Let G be a graph. Then:*

1. *There is no universal function f such that*

$$\mathit{FVS}(G) \leq f\big(\mathit{lw}(G)\big)$$

holds for every graph G; in other words, having bounded linear-width does not imply that the size of a minimum feedback vertex set (FVS) is bounded.

2. *Conversely, if G has a feedback vertex set S with $|S|$ bounded by a constant, then the linear-width $\mathit{lw}(G)$ is also bounded by a function of $|S|$.*

Proof. We prove each part in detail.

(1) Bounded lw(G) does not imply bounded FVS. We claim that there exists a family of graphs for which the linear-width is bounded by a constant while the size of the minimum feedback vertex set grows arbitrarily large.

Counterexample: For each integer $m \geq 1$, construct a graph G_m as follows. Start with m disjoint cycles, each being a triangle C_3. Note that each triangle admits a linear ordering of its edges with a small separator (for example, one may obtain $\text{lw}(C_3) \leq 2$). Now, connect these m triangles in a path-like manner by adding a single edge between one vertex of a triangle and one vertex of the next triangle. This joining can be done so that the overall linear-width of the connected graph G_m remains bounded by a constant (say, at most 3) independent of m.

However, in order to break all cycles in G_m, one must remove at least one vertex from each triangle. Consequently, the size of any feedback vertex set in G_m is at least m. Therefore, even though $\text{lw}(G_m)$ is bounded by a constant, the minimum FVS grows without bound as m increases. This shows that no function f exists which can bound FVS(G) solely in terms of $\text{lw}(G)$.

(2) Bounded FVS implies bounded lw(G). Assume that S is a feedback vertex set of G with $|S| = k$, and let $G' = G - S$ be the acyclic graph (a forest) obtained by removing S from G. Since G' is a forest, it can be decomposed linearly with very small separators. In fact, one may arrange the edges of G' in a linear order

$$(e'_1, e'_2, \ldots, e'_{m'})$$

so that for every index i, the set

$$S'_i = \{\, v \in V(G') \mid v \text{ is incident to an edge in both } (e'_1, \ldots, e'_i) \text{ and } (e'_{i+1}, \ldots, e'_{m'})\,\}$$

satisfies $|S'_i| \leq 1$. Hence,

$$\text{lw}(G') \leq 1.$$

Now, when the vertices in S are reintroduced to form G, each bag in the linear decomposition of G' may grow by at most k vertices. Therefore, we obtain

$$\text{lw}(G) \leq \text{lw}(G') + k \leq 1 + k.$$

Since $k = |S|$ is bounded by assumption, it follows that $\text{lw}(G)$ is bounded by a function of $|S|$.

Combining both parts, we conclude that while bounded linear-width does not guarantee a bound on the size of a feedback vertex set, a bounded feedback vertex set does imply bounded linear-width. □

Example 4 (Counterexample for Theorem 3 (Part 1)) For each integer $m \geq 1$, construct the graph G_m as follows. Start with m disjoint triangles (each triangle is a cycle C_3) and then connect these triangles sequentially by adding a single edge between one vertex of a triangle and one vertex of the next triangle. Each individual triangle admits a linear decomposition with a small separator (for example, $\text{lw}(C_3) \leq 2$), and with a careful ordering of the edges the overall graph G_m can be arranged to have a bounded linear-width (say, at most 3). However, since each triangle contains a cycle, any feedback vertex set must remove at least one vertex per triangle, which means $\text{FVS}(G_m) \geq m$. Thus, even though $\text{lw}(G_m)$ remains bounded by a constant, the size of the minimum feedback vertex set grows without bound as m increases.

Example 5 (Example for Theorem 3 (Part 2)) Consider the graph G obtained by taking a tree (which has linear-width 1) and adding one extra edge that creates a single cycle. In this graph, the minimal feedback vertex set has size 1 (removing that vertex breaks the cycle). Removing the vertex yields a forest with linear-width 1, and reintroducing the vertex increases the linear-width by at most 1. Therefore, $\text{lw}(G) \leq 1 + 1 = 2$, showing that a bounded feedback vertex set implies bounded linear-width.

Theorem 4 *If a graph G has bounded maximum matching-width, that is, if*

$$\textit{MM-width}(G) \leq k$$

for some constant k, then its linear-width is also bounded by a function of k. In particular,

$$\textit{lw}(G) \leq 2k.$$

Proof. Assume that G has a maximum matching decomposition (T, ∂) such that for every edge e in the tree T, the associated bipartite subgraph B_e (obtained by partitioning G according to the two connected components of $T \setminus \{e\}$) has a maximum matching of size at most k.

For each such edge e, let the removal of e partition T into two subtrees, which in turn correspond to two subsets D_1 and D_2 of vertices in G (as determined by the bijection ∂). By Kőnig's theorem, the bipartite graph B_e has a vertex cover of size at most k; however, to cover both endpoints of each edge in a matching, the separator induced by e in G involves at most $2k$ vertices.

We now construct a linear ordering of the edges of G based on a depth-first traversal of the tree T. This traversal yields a linear sequence

$$(e_1, e_2, \ldots, e_m)$$

of the edges of G (via the bijection ∂). For any index i with $1 \leq i < m$, let

$$S_i = \{\, v \in V(G) \mid v \text{ is incident to edges in both } (e_1, \ldots, e_i) \text{ and } (e_{i+1}, \ldots, e_m) \,\}$$

be the separator corresponding to the cut between the prefix and suffix of the ordering. By the construction, each such separator is associated with a cut in T whose size is bounded by $2k$. Hence,

$$|S_i| \leq 2k \quad \text{for all } i.$$

Taking the maximum over all i, we conclude that

$$\text{lw}(G) \leq 2k.$$

Thus, bounded maximum matching-width implies that the linear-width of G is also bounded by a function of k. $\square$

Example 6 (Example for Theorem 4) Again, consider the path graph P_4 with vertices

$$V(P_4) = \{v_1, v_2, v_3, v_4\}$$

and edges

$$E(P_4) = \{e_1, e_2, e_3\}.$$

A maximum matching decomposition of P_4 has MM-width$(P_4) = 1$, as every separator (obtained from the corresponding bipartite subgraphs) has a maximum matching of size at most 1. By converting the decomposition tree into a linear ordering via a depth-first traversal, we obtain a linear decomposition whose separators have size at most 2. Hence, $\mathrm{lw}(P_4) \leq 2$, which confirms that bounded maximum matching-width implies bounded linear-width.

IV. Conclusions and Future Research Directions

In this paper, we provided proofs for the upper and lower bounds of certain graph parameters. For future research, we aim to investigate the upper and lower bounds of other graph parameters, such as Tree-Partition-Width and Path-Partition-Width [63, 64, 65, 66].

Additionally, we plan to extend the graph parameters discussed in this paper to hypergraphs[67, 68, 69, 70, 71], Bidirected Graphs[72, 73], Mixed Graphs[74, 75], and superhypergraphs[76, 77, 78, 79, 31, 80, 81, 82], exploring their properties and implications in these broader settings.

Acknowledgments

We extend our sincere gratitude to everyone who provided insights, inspiration, and assistance throughout this research. We particularly thank our readers for their interest and acknowledge the authors of the cited works for laying the foundation that made our study possible. We also appreciate the support from individuals and institutions that provided the resources and infrastructure needed to produce and share this paper. Finally, we are grateful to all those who supported us in various ways during this project.

Data Availability

This research is purely theoretical, involving no data collection or analysis. We encourage future researchers to pursue empirical investigations to further develop and validate the concepts introduced here.

Conflicts of Interest

The authors confirm that there are no conflicts of interest related to the research or its publication.

References

[1] Jonathan L Gross, Jay Yellen, and Mark Anderson. *Graph theory and its applications*. Chapman and Hall/CRC, 2018.

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

[3] David R Karger and Nathan Srebro. "Learning Markov networks: maximum bounded tree-width graphs." In: *SODA*. 2001, pp. 392–401.

[4] Jinbo Xu, Feng Jiao, and Bonnie Berger. "A tree-decomposition approach to protein structure prediction". In: *2005 IEEE Computational Systems Bioinformatics Conference (CSB'05)*. IEEE. 2005, pp. 247–256.

[5] Borislav Slavchev, Evelina Masliankova, and Steven Kelk. "A machine learning approach to algorithm selection for exact computation of treewidth". In: *Algorithms* 12.10 (2019), p. 200.

[6] Nathan Srebro. "Maximum likelihood bounded tree-width Markov networks". In: *Artificial intelligence* 143.1 (2003), pp. 123–138.

[7] Gal Elidan and Stephen Gould. "Learning bounded treewidth Bayesian networks". In: *Advances in neural information processing systems* 21 (2008).

[8] Robert Sasak. "Comparing 17 graph parameters". MA thesis. The University of Bergen, 2010.

[9] Daniel J Harvey and David R Wood. "Parameters tied to treewidth". In: *Journal of Graph Theory* 84.4 (2017), pp. 364–385.

[10] Hans L Bodlaender. "A tourist guide through treewidth". In: *Acta cybernetica* 11.1-2 (1993), pp. 1–21.

[11] Hans L Bodlaender. "A partial k-arboretum of graphs with bounded treewidth". In: *Theoretical computer science* 209.1-2 (1998), pp. 1–45.

[12] Hans L Bodlaender and Arie MCA Koster. "Treewidth computations I. Upper bounds". In: *Information and Computation* 208.3 (2010), pp. 259–275.

[13] T. Johnson et al. "Directed tree-width". In: *Journal of Combinatorial Theory, Series B* 82.1 (2001), pp. 138–154.

[14] Takaaki Fujita. "Superhypertree-depth: A structural analysis within superhypergraphs". In: *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond* (), p. 11.

[15] William Pettersson and John Sylvester. "Bounds on the twin-width of product graphs". In: *Discrete Mathematics & Theoretical Computer Science* 25.Graph Theory (2023).

[16] Takaaki Fujita and Koichi Yamazaki. "Tangle and ultrafilter: Game theoretical interpretation". In: *Graphs and Combinatorics* 36.2 (2020), pp. 319–330.

[17] Neil Robertson and Paul D. Seymour. "Graph minors. III. Planar tree-width". In: *Journal of Combinatorial Theory, Series B* 36.1 (1984), pp. 49–64.

[18] Neil Robertson and Paul D. Seymour. "Graph minors. I. Excluding a forest". In: *Journal of Combinatorial Theory, Series B* 35.1 (1983), pp. 39–61.

[19] Neil Robertson and Paul D. Seymour. "Graph minors. X. Obstructions to tree-decomposition". In: *Journal of Combinatorial Theory, Series B* 52.2 (1991), pp. 153–190.

[20] Takaaki Fujita. "Ultrafilters and Their Dual Relationship to Tree-Width in Graph Theory". In: *Asian Research Journal of Mathematics* 21.1 (2025), pp. 98–114.

[21] Neil Robertson and Paul D Seymour. "Graph minors. IV. Tree-width and well-quasi-ordering". In: *Journal of Combinatorial Theory, Series B* 48.2 (1990), pp. 227–254.

[22] Haim Kaplan and Ron Shamir. "Pathwidth, bandwidth, and completion problems to proper interval graphs with small cliques". In: *SIAM Journal on Computing* 25.3 (1996), pp. 540–561.

[23] Dariusz Dereniowski. "From pathwidth to connected pathwidth". In: *SIAM Journal on Discrete Mathematics* 26.4 (2012), pp. 1709–1732.

[24] Atsushi TAKAHASHI, Shuichi UENO, and Yoji KAJITANI. "On the proper-path-decomposition of trees". In: *IEICE transactions on fundamentals of electronics, communications and computer sciences* 78.1 (1995), pp. 131–136.

[25] Atsushi Takahashi, Shuichi Ueno, and Yoji Kajitani. "Mixed searching and proper-path-width". In: *Theoretical Computer Science* 137.2 (1995), pp. 253–268.

[26] Sang-il Oum. "Rank-width: Algorithmic and structural results". In: *Discrete Applied Mathematics* 231 (2017), pp. 15–24.

[27] Sang-il Oum. "Rank-width and vertex-minors". In: *Journal of Combinatorial Theory, Series B* 95.1 (2005), pp. 79–100.

[28] Isolde Adler, Georg Gottlob, and Martin Grohe. "Hypertree width and related hypergraph invariants". In: *European Journal of Combinatorics* 28.8 (2007), pp. 2167–2181.

[29] Takaaki Fujita. "Obstruction for hypertree width and superhypertree width". In: *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond* (2025), p. 26.

[30] Georg Gottlob, Nicola Leone, and Francesco Scarcello. "Hypertree decompositions and tractable queries". In: *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. 1999, pp. 21–32.

[31] Takaaki Fujita and Smarandache Florentin. "Some Graph Parameters for Superhypertree-width and Neutrosophictree-width". In: *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024. ISBN: 978-1-59973-815-4.

[32] Takaaki Fujita. "Superhyperbranch-width and Superhypertree-width". In: *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond* (2025), p. 367.

[33] Takaaki Fujita. "Short note of supertree-width and n-Superhypertree-width". In: *Neutrosophic Sets and Systems* 77 (2024), pp. 54–78. URL: https://fs.unm.edu/nss8/index.php/111/article/view/5243.

[34] Jim Geelen, Bert Gerards, and Geoff Whittle. "Obstructions to branch-decomposition of matroids". In: *Journal of Combinatorial Theory, Series B* 96.4 (2006), pp. 560–570.

[35] James F Geelen, Albertus MH Gerards, and Geoff Whittle. "Branch-width and well-quasi-ordering in matroids and graphs". In: *Journal of Combinatorial Theory, Series B* 84.2 (2002), pp. 270–290.

[36] Koichi Yamazaki. "On approximation intractability of the path–distance–width problem". In: *Discrete applied mathematics* 110.2–3 (2001), pp. 317–325.

[37] Koichi Yamazaki et al. "Isomorphism for graphs of bounded distance width". In: *Algorithmica* 24 (1999), pp. 105–127.

[38] Jannis Bulian and Anuj Dawar. "Graph isomorphism parameterized by elimination distance to bounded degree". In: *Algorithmica* 75.2 (2016), pp. 363–382.

[39] Vikraman Arvind et al. "The isomorphism problem for k-trees is complete for logspace". In: *Information and Computation* 217 (2012), pp. 1–11.

[40] Bireswar Das, Jacobo Torán, and Fabian Wagner. "Restricted space algorithms for isomorphism on bounded treewidth graphs". In: *Information and Computation* 217 (2012), pp. 71–83.

[41] Dimitrios M Thilikos. "Algorithms and obstructions for linear-width and related search parameters". In: *Discrete Applied Mathematics* 105.1-3 (2000), pp. 239–271.

[42] Takaaki Fujita. "Bounded tree-depth, path-distance-width, and linear-width of graphs". In: *Journal of Fundamental Mathematics and Applications (JFMA)* 7.2 (2024), pp. 138–148.

[43] Fedor V Fomin and Dimitrios M Thilikos. "On the monotonicity of games generated by symmetric submodular functions". In: *Discrete Applied Mathematics* 131.2 (2003), pp. 323–335.

[44] Daniel Bienstock. "Graph searching, path-width, tree-width and related problems (a survey)." In: *Reliability of computer and communication networks* 5 (1989), pp. 33–50.

[45] Frank Gurski and Carolin Rehs. "Comparing linear width parameters for directed graphs". In: *Theory of Computing Systems* 63 (2019), pp. 1358–1387.

[46] Takaaki Fujita. "Matroid, Ideal, Ultrafilter, Tangle, and so on: Reconsideration of Obstruction to linear decomposition". In: *International Journal of Mathematics Trends and Technology (IJMTT)* 70 (2024), pp. 18–29.

[47] Takaaki Fujita. "The Interplay between Quasi-Matroid and Connectivity Systems". In: *Asian Research Journal of Mathematics* 21.1 (2025), pp. 115–129.

[48] Jisu Jeong, Seongmin Ok, and Geewon Suh. "Characterizing graphs of maximum matching width at most 2". In: *Discrete Applied Mathematics* 248 (2018), pp. 102–113.

[49] Kwangjun Ahn and Jisu Jeong. "Computing the maximum matching width is NP-hard". In: *arXiv preprint arXiv:1710.05117* (2017).

[50] Jisu Jeong, Sigve Hortemo Sæther, and Jan Arne Telle. "Maximum matching width: New characterizations and a fast algorithm for dominating set". In: *Discrete Applied Mathematics* 248 (2018), pp. 114–124.

[51] Yixin Cao, Jianer Chen, and Yang Liu. "On feedback vertex set: New measure and new structures". In: *Algorithmica* 73 (2015), pp. 63–86.

[52] Vineet Bafna, Piotr Berman, and Toshihiro Fujito. "A 2-approximation algorithm for the undirected feedback vertex set problem". In: *SIAM Journal on Discrete Mathematics* 12.3 (1999), pp. 289–297.

[53] Jianer Chen et al. "A fixed-parameter algorithm for the directed feedback vertex set problem". In: *Proceedings of the fortieth annual ACM symposium on Theory of computing*. 2008, pp. 177–186.

[54] Douglas Brent West et al. *Introduction to graph theory*. Vol. 2. Prentice hall Upper Saddle River, 2001.

[55] Reinhard Diestel. "Graduate texts in mathematics: Graph theory". In: ().

[56] Reinhard Diestel. "Graph theory 3rd ed". In: *Graduate texts in mathematics* 173.33 (2005), p. 12.

[57] Evelyn L Tan. "Some notes on cycle graphs". In: *Discrete mathematics* 105.1-3 (1992), pp. 221–226.

[58] Haitze J Broersma and Cornelis Hoede. "Path graphs". In: *Journal of graph theory* 13.4 (1989), pp. 427–444.

[59] Richard J Lipton and Robert Endre Tarjan. "A separator theorem for planar graphs". In: *SIAM Journal on Applied Mathematics* 36.2 (1979), pp. 177–189.

[60] Uriel Feige, MohammadTaghi Hajiaghayi, and James R Lee. "Improved approximation algorithms for minimum-weight vertex separators". In: *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. 2005, pp. 563–572.

[61] Armen S Asratian, Tristan MJ Denley, and Roland Häggkvist. *Bipartite graphs and their applications*. Vol. 131. Cambridge university press, 1998.

[62] Sarbari Mitra and Soumya Bhoumik. "Graceful labeling of triangular extension of complete bipartite graph". In: *Electron. J. Graph Theory Appl.* 7 (2019), pp. 11–30. URL: `https://api.semanticscholar.org/CorpusID:145850349`.

[63] David R. Wood. "On tree-partition-width". In: *European Journal of Combinatorics* 30.5 (2009), pp. 1245–1253.

[64] Hans L. Bodlaender, Carla Groenland, and Hugo Jacob. "On the parameterized complexity of computing tree-partitions". In: *arXiv preprint arXiv:2206.11832* (2022).

[65] Guoli Ding and Bogdan Oporowski. "On tree-partitions of graphs". In: *Discrete Mathematics* 149.1–3 (1996), pp. 45–58.

[66] Chun-Hung Liu and David R. Wood. "Quasi-tree-partitions of graphs with an excluded subgraph". In: *arXiv preprint arXiv:2408.00983* (2024).

[67] Claude Berge. *Hypergraphs: combinatorics of finite sets*. Vol. 45. Elsevier, 1984.

[68] Alain Bretto. "Hypergraph theory". In: *An introduction. Mathematical Engineering. Cham: Springer* 1 (2013).

[69] Yue Gao et al. "Hypergraph learning: Methods and practices". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.5 (2020), pp. 2548–2566.

[70] Yifan Feng et al. "Hypergraph neural networks". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 3558–3565.

[71] Song Feng et al. "Hypergraph models of biological networks to identify genes critical to pathogenic viral response". In: *BMC bioinformatics* 22.1 (2021), p. 287.

[72] Nanao Kita. "Bidirected Graphs I: Signed General Kotzig-Lovász Decomposition". In: *arXiv: Combinatorics* (2017). URL: `https://api.semanticscholar.org/CorpusID:119320661`.

[73] Jes'us Arturo Jim'enez Gonz'alez and Andrzej Mr'oz. "Bidirected graphs, integral quadratic forms and some Diophantine equations". In: 2023. URL: https://api.semanticscholar.org/CorpusID:258309617.

[74] Takaaki Fujita. "Extensions of MultiDirected Graphs: Fuzzy, Neutrosophic, Plithogenic, Soft, Hypergraph, and Superhypergraph Variants". In: *preprint* (2025).

[75] Takaaki Fujita and Florentin Smarandache. "Mixed graph in fuzzy, neutrosophic, and plithogenic graphs". In: *Neutrosophic Sets and Systems* 74 (2024), pp. 457–479.

[76] Takaaki Fujita and Florentin Smarandache. "A Concise Study of Some Superhypergraph Classes". In: *Neutrosophic Sets and Systems* 77 (2024), pp. 548–593. URL: https://fs.unm.edu/nss8/index.php/111/article/view/5416.

[77] Florentin Smarandache. "n-SuperHyperGraph and Plithogenic n-SuperHyperGraph". In: *Nidus Idearum* 7 (2019), pp. 107–113.

[78] Takaaki Fujita and Florentin Smarandache. *Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations*. Infinite Study, 2025.

[79] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025. ISBN: 978-1-59973-812-3.

[80] Takaaki Fujita. "Superhypertree-length and superhypertree-breadth in superhypergraphs". In: *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond* (2025), p. 41.

[81] Takaaki Fujita. "Exploration of Graph Classes and Concepts for SuperHypergraphs and n-th Power Mathematical Structures". In: *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond* 3.4 (), p. 512.

[82] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.