Transactions on Soft Computing

www.tsc.reapress.com

Trans. Soft. Comput. Vol. 1, No. 3 (2025), pp. 217–228.

Paper Type: Original Article

A Minimum Extended Connected Dominating Set in Interval Graphs with an Application to Railway Track Monitoring

Prodyot Ghara[1], Mangal Pati[2], Sukumar Mondal[3], Sambhu Charan Barman[4,*], Uma Shankar[2]

[1]Research Centre in Natural and Applied Sciences, Raja N. L. Khan Women's College, Midnapore 721102, India; pghara@gmail.com.

[2]Department of Mathematics, Asian International University, Imphal West 795113, India; mangalpati@gmail.com; yadav_05@rediffmail.com.

[3]Department of Mathematics, Raja N. L. Khan Women's College, Midnapore 721102, India; sukumarmondal@rnlkwc.ac.in .

[4]Department of Mathematics, Shahid Matangini Hazra Government General Degree College for Women, Tamluk 721649, India; barman.sambhu@gmail.com.

Citation:

Received: 10 December 2024 *Revised: 20 February 2025* *Accepted: 03 May 2025*	Ghara, P., Pati, M., Mondal, S., Barman, S. Ch., & Shankar, U. (2025). A minimum extended connected dominating set in interval graphs with an application to railwaytrack monitoring. *Transactions on soft computing*, *1*(3), 217-228.

Abstract

For a graph $G = (V, E)$ and a positive integer k, a set $S \subseteq V$ is a k-extended dominating set if every vertex lies within distance one of S or has at least k distinct members of S at distance two. The minimum cardinality of such a set is denoted by $\gamma_e^k(G)$. For $k = 2$, a connected induced subgraph on S gives an extended connected dominating set. This paper studies structural properties of extended connected dominating sets in interval graphs whose intervals are ordered by nondecreasing right endpoint. An ordered sweep algorithm is presented for constructing a minimum extended connected dominating set under the stated selection invariants. With presorted interval endpoints and forward-only scan pointers, its running time and auxiliary storage are $O(n)$. A railway-track monitoring example illustrates how the interval model can select a small connected sensor backbone while retaining two-hop redundancy. The application is illustrative and assumes a linear track, deterministic overlap, and equal communication ranges.

Keywords: Extended domination, Connected dominating set, Interval graph, Linear-time sweep, Railway monitoring.

Corresponding Author: barman.sambhu@gmail.com

https://doi.org/10.48314/tsc.v1i3.70

1 | Introduction

Graph theory provides a foundational framework for representing and examining complex systems found in numerous fields, including computer science, biology, and social network exploration. A key concept within this area is the dominating set, which is instrumental in exploring the structure and dynamics of graphs. Dominating sets are useful in real life for improving network design, picking the best spots for setting up services or buildings, and making computer programs work more efficiently.

Let $G = (V, E)$ be a graph, where V represents the set of vertices (nodes) and E represents the set of edges (links). Throughout this paper, we assume that G is a connected graph without self-loops and parallel edges.

For a vertex $w \in V(G)$, the open neighborhood of w is denoted by $N_G(w)$ and is defined as

$$N_G(w) = \{u \in V(G) : (u, w) \in E(G)\}.$$

The closed neighborhood of w is denoted by $N_G[w]$ and is given by

$$N_G[w] = N_G(w) \cup \{w\}.$$

Domination is an important concept in graph theory and has attracted significant attention from researchers. A subset $D \subseteq V(G)$ is called a *dominating set* (DS) of G if every vertex in $V(G) - D$ is adjacent to at least one vertex in D. In other words, a vertex $v \in D$ dominates all vertices in its closed neighborhood $N_G[v]$. The main objective of the domination problem is to find a dominating set with the smallest possible number of vertices.

The concept related to the domination number $\gamma(G)$ was first discussed by Berge [1]. Later, the terms dominating set and domination number were formally introduced by Ore [2]. A comprehensive study of domination in graphs was presented by Haynes et al. [3], which summarized a large number of research works in this area.

If the subgraph induced by a dominating set is connected, then it is called a *connected dominating set* (CDS). In a CDS, every pair of vertices in the dominating set is connected either directly or through other vertices in the set. Connected dominating sets are widely used in wireless networks as a virtual backbone for communication, such as broadcasting messages [4], searching in reduced spaces, and covering regions in sensor networks [5]. In wireless sensor networks, due to the promiscuous receiving mode, when each node in a CDS forwards a packet once, the packet can reach all nodes in the network.

The term *connected domination number* was first introduced by Sampathkumar and Walikar [6]. Over time, many variations of domination have been studied in the literature, including secure domination, perfect domination, connected domination [7], edge domination, extended domination, k-hop domination [8, 9, 10], k-tuple domination, weighted domination, paired domination, independent domination, Roman domination [11], inverse Roman domination [12], total domination [13], isolate domination [14], k-efficient domination [11], restrained domination [15], and independent line-set domination [16]. Slater referred to k-hop dominating sets as k-bases [10]. Several properties and results related to domination in graphs can be found in the literature [3], [17-26].

Let G be a graph and let k be a positive integer. A subset $S \subseteq V(G)$ is called a *k-extended dominating set* if every vertex $u \in V(G)$ satisfies at least one of the following conditions:

1. The distance between u and some vertex in S is at most one, or
2. There exist at least k distinct vertices $v_1, v_2, \ldots, v_k \in S$ such that the distance between u and each v_i is two.

The *k-extended domination number*, denoted by $\gamma_e^k(G)$, is the minimum cardinality among all k-extended dominating sets of G. When $k = 2$, the set is simply called an *extended dominating set* (EDS), and the corresponding minimum size is called the *extended domination number* of G.

In other words, a dominating set S is an extended dominating set if every vertex in the graph either belongs to S, has a neighbor in S, or has at least two vertices in S at distance two. If such a set S is connected, then it is called an *extended connected dominating set* (ECDS, in short).

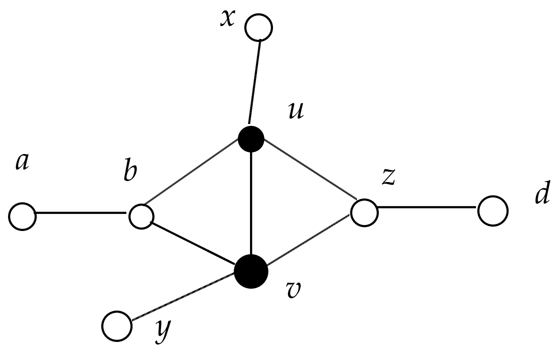

Figure 1. A connected graph.

In Figure 1, $\{u, v, b, z\}$ is a CDS, whereas $\{u, v\}$ is an ECDS because each of the vertices a and d is at distance two from both u and v.

In this article, we delve into the theory and applications of extended dominating sets in graphs. We also focus on designing algorithm for computing a minimum ECDS of interval graphs.

Interval graphs. An interval graph (IG) is the intersection graph of a family of closed intervals on the real line. Let $I_j = [a_j, b_j]$, $j = 1, \ldots, n$, with $a_j \leq b_j$. A vertex v_j represents I_j, and $v_i v_j \in E(G)$ exactly when $I_i \cap I_j \neq \emptyset$. Throughout the algorithmic sections, the intervals are indexed by nondecreasing right endpoint,

$$b_1 \leq b_2 \leq \cdots \leq b_n,$$

with ties broken by nondecreasing left endpoint. This ordering is essential for the structural lemmas and the sweep implementation. Figure 2 shows an interval graph together with an interval representation.

In operations research and scheduling theory, interval graphs are commonly used to model resource-allocation problems. Each task–such as assigning a network link, a classroom, or a processor–is represented by a time interval. Other applications include file organization, job scheduling, protein sequencing, DNA mapping, circuit design, temporal reasoning, and network routing [27-33].

We recall a key result for IG $G = (V, E)$ presented in below.

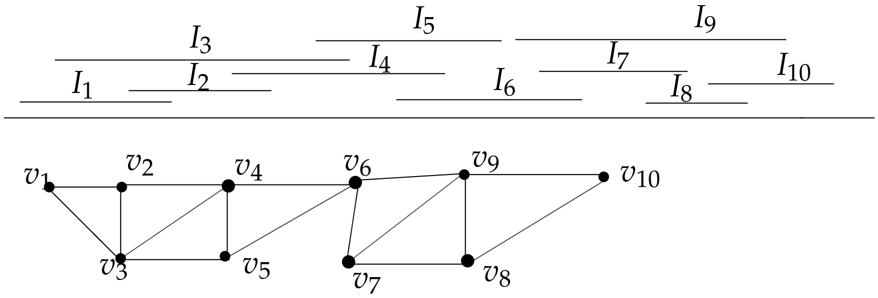

Figure 2. An example of IG and its interval model.

Lemma 1.1. *Let I_i, I_j, I_k be intervals in the right-endpoint ordering with $i < j < k$. If v_i is adjacent to v_k, then v_j is adjacent to v_k. The vertices v_i and v_j need not be adjacent.*

Proof. Since $I_i \cap I_k \neq \emptyset$, we have $a_k \leq b_i$. The ordering gives $b_i \leq b_j \leq b_k$, while $a_j \leq b_j$. Hence $a_k \leq b_j$ and $a_j \leq b_k$, so $I_j \cap I_k \neq \emptyset$. □

Literature review. Connected dominating sets and their approximation have been studied extensively; Guha and Khuller [34], for example, developed approximation algorithms for the classical connected dominating-set problem. Fink and Jacobson [35] studied related distance-sensitive domination parameters. These concepts provide historical context, but the distance-two redundancy condition used here is the specific extended-domination definition stated above.

Subsequently, Wu [36] investigated routing schemes based on extended dominating sets for ad hoc wireless networks with unidirectional links in 2002. Later, Wu et al. [37] further examined extended dominating sets and their applications to ad hoc networks, particularly in the setting of cooperative communication.

More recently, Gao et al. [38] (2023) determined the exact values of extended domination numbers for paths and cycles. They also established several bounds for these parameters in planar graphs with small diameters.

Applications.

Extended dominating sets play a vital role in several real-world applications, including communication networks, facility placement, and social network analysis. In network design, they are used to strategically position key nodes to improve message delivery efficiency and ensure the system remains robust against failures. When it comes to locating essential services like hospitals, schools, or distribution hubs, extended dominating sets help in selecting sites that offer the best coverage and accessibility for the community. In the context of social networks, these sets help identify central or influential users, which can be valuable for targeted marketing and understanding how information or trends spread through the network. Utilizing extended dominating sets allows for the creation of more efficient, resilient, and optimized systems across various domains.

Main outcome. In this article, we investigate several new properties of extended connected dominating sets in interval graphs. We then propose an algorithm to determine a minimum extended connected dominating set for such graphs. The correctness of the proposed algorithm is also established.

Furthermore, we analyze the time complexity and space requirements of the algorithm. In addition, a practical application is presented to demonstrate the usefulness of our results.

In particular, we consider the problem of efficient monitoring of railway tracks using wireless sensors, which is modeled through interval graphs and extended connected dominating sets.

Finally, we illustrate the applicability of our findings through a real-world example.

Structure of this paper. Section 2 introduces the notation. Section 3 presents structural properties of extended connected dominating sets in interval graphs. Section 4 gives the algorithm, and Section 5 discusses correctness and complexity. Section 6 presents the railway-monitoring illustration, and Section 7 concludes the paper.

2 | Notations

The notation used in the algorithm is summarized in Table 1.

Table 1. Notation for the ordered interval model.

Symbol	Meaning
IG	Interval graph
DS, EDS	Dominating set and extended dominating set
CDS, ECDS	Connected dominating set and extended connected dominating set
$a_{\min}$	Minimum left endpoint among all intervals
$N(v), N[v]$	Open and closed neighborhoods of v
$d(x, y)$	Shortest-path distance between vertices x and y
$m(I_k)$	Highest-indexed interval adjacent to I_k in the right-endpoint ordering
$m^2(I_k)$	$m(m(I_k))$
$m^2_{\text{2nd}}(I_k)$	Second-highest-indexed interval adjacent to $m(I_k)$, when it exists

3 | Structural Properties of Minimum Extended Connected Dominating Sets in Interval Graphs

Here, we explore some key results related to the minimum extended connected dominating set on IGs.

Lemma 3.1. *If $m(I_1) = I_n$, then $\{v_n\}$ is a minimum ECDS of G.*

Proof. The condition $m(I_1) = I_n$ implies $v_1 v_n \in E$. By Lemma 1.1, every intermediate vertex v_j, $1 < j < n$, is adjacent to v_n. Thus $\{v_n\}$ is a connected dominating set and hence an ECDS. No nonempty feasible set can have cardinality below one, so it is minimum. □

Lemma 3.2. *If $m^2(I_1)$ and $m^2_{\text{2nd}}(I_1)$ exist, their corresponding vertices form the rightmost candidate pair used by the initial sweep step.*

Proof. For the interval graph, suppose $m^2(I_1)$ and $m^2_{2nd}(I_1)$ exist. Also, let $m(I_1) = I_x$, $m^2(I_1) = I_z$ and $m^2_{2nd}(I_1) = I_y$. So, $(v_1, v_x) \in E$, $(v_x, v_y) \in E$ and $(v_x, v_z) \in E$ and $1 < x < y < z$. In addition, v_1 is not adjacent to both v_y and v_z. Therefore, $d(v_1, v_y) = 2 = d(v_1, v_z)$.

In addition, $d(v_t, v_y) \leq 2$ and $d(v_t, v_z) \leq 2$, where $1 < t < y < z$. So, all the vertices v_i, $i = 1, 2, ..., z$ are extended connected dominated by the set $\{v_y, v_z\}$. Also, v_y and v_z are, respectively, the maximum and next-to-maximum indexed adjacent vertices of v_x. So, v_y and v_z can extended dominate some other vertices whose indexed are greater than z. Therefore, v_y, v_z are two possible members of D_{mec}. □

Lemma 3.3. *If $m^2_{\text{2nd}}(I_1)$ does not exist, the vertices corresponding to $m(I_1)$ and $m^2(I_1)$ form the alternative initial candidate pair.*

Proof. Let $m(I_1) = I_x$ and $m^2(I_1) = I_y$. Then $v_1 v_x \in E$ and $v_x v_y \in E$, so the pair $\{v_x, v_y\}$ is connected. In the ordered sweep, v_x covers the first frontier and v_y reaches the farthest available right endpoint from v_x. Therefore this pair is the alternative considered when a second highest-indexed successor of I_x is unavailable. □

Lemma 3.4. *Suppose v_i and v_j, $i < j$, are consecutive selected vertices and $m(I_j) = I_k$. If $(b_j, b_k]$ contains the left endpoint of an unprocessed interval, then the sweep selects v_k next.*

Proof. For an interval graph G, let v_i and v_j, $i < j$ be two consecutive selected members of D_{mec} and $m(I_j) = I_k$. Now, if there is no left endpoint of any intervals (present in interval representation) in the interval $I = (b_j, b_k)$, then no need to select any other vertices as members of D_{mec}. But, if the interval $I = (b_j, b_k)$ contains at least one left endpoint of any intervals, say $b_j < a_x < b_k$, in the interval representation, then $(v_j, v_k) \in E$, but $(v_j, v_x) \notin E$ $(v_i, v_x) \notin E$. So, v_x is not extended dominated by v_i or v_j. So, at least one more vertex is need to select as a member of D_{mec}. Since, $m(I_j) = I_k$, so, v_k will be the next member of D_{mec}. □

4 | Algorithm

The algorithm starts with one of the two candidate pairs identified in Lemmas 3.2 and 3.3, unless Lemma 3.1 gives a singleton solution. It then advances the right frontier according to Lemma 3.4. All indices refer to the nondecreasing right-endpoint ordering.

We now present the final algorithm for computing a minimum extended connected dominating set D_{mec} for interval graphs. The complete algorithm is given below.

Algorithm MECDSIG
Input: Interval representation $[a_j, b_j]$, $j = 1 : 1 : n$ of the interval graph G.
Output: An extended connected dominating set D_{mec} of G.
Step 1: First, set $D_{ec} = \emptyset$ and $k = 1$.
Step 2: Set a scanning interval $I = [l_p, r_p]$, where $l_p = a_{\min}$ and $r_p = b_1$.
Step 3: Scan I and find $j = \max\{i \neq 1 : a_i \in I\}$.
Step 4: Set $m^k(I_1) = I_j$.
Step 5: Reset the scanning interval I, by setting $l_p = r_p$ and $r_p = b_j$.
Step 6: Let $C = \{i : a_i \in I\}$. If $C \neq \emptyset$, set $\lambda = \max C$; if $|C| \geq 2$, set $\mu = \max(C \setminus \{\lambda\})$.
Step 7: Set $m^{k+1}(I_1) = I_\lambda$ and $m^{k+1}_{2nd}(I_1) = I_\mu$.
Step 8: If $m(I_1) = I_n$, output $D_{mec} = \{v_n\}$ and stop (Lemma 3.1).
Else if $m^{k+1}(I_1)$ and $m^{k+1}_{2nd}(I_1)$ exist, then
$D_{ec} = D_{ec} \cup \{m^{k+1}(I_1), m^{k+1}_{2nd}(I_1)\}$ (by Lemma 3.2).
Else $D_{ec} = D_{ec} \cup \{m^k(I_1), m^{k+1}(I_1)\}$ (by Lemma 3.3).

EndIf.
Step 9: Reset the scanning interval I, by setting $l_p = r_p$ and $r_p = b_\mu$.
Step 10: Scan the interval I and find $A = \{i : a_i \in I\}$ and $\mu = \max\{A\}$.
Step 11: Set $m(m_{2nd}^{k+1}(I_1)) = I_\mu$.
Step 12: Reset the scanning interval I by setting $l_p = r_p$ and $r_p = b_\lambda$.
Step 13: Scan the interval I and find $B = \{i : a_i \in I\}$ and $\lambda = \max\{A \cup B\}$.
Step 14: Set $m^{k+2}(I_1) = I_\lambda$.
Step 15: If $m(m_{2nd}^{k+1})$ exists, set $I = [a_{\min}, b_x]$, where b_x is the right endpoint of $m(m_{2nd}^{k+1}(I_1))$.
Else set $I = [a_{\min}, b_x]$, where b_x is the right endpoint of $m^{k+1}(I_1)$.
Endif.
Step 16: Mark all intervals whose $a_i \in I$.
Step 17: Reset the scanning interval I by setting $l_p = r_p$ and $r_p = b_y$, where b_y is the right endpoint of $m^{k+2}(I_1)$.
Step 18: Find $A = \{a_i : I_i$ is an unmarked interval such that $a_i \in I\}$.
Step 19: If $A = \emptyset$ then Stop.
Else $D_{ec} = D_{ec} \cup \{m^{k+2}(I_1)\}$ (by Lemma 3.4).
Endif.
Step 20: Set $k = k + 1$ and go to Step 17.
Step 21: $D_{mec} = \{v_i : v_i$ corresponding to the member I_i of $D_{ec}\}$.
End MECDSIG.

Scanning windows are interpreted as left-open and right-closed after their first use, so an endpoint is processed once. A branch that does not define μ skips the later instructions that depend on μ and continues from the frontier supplied by Lemma 3.3. These conventions are required for an implementable sweep.

When the above Algorithm **MECDSIG** is applied to the graph G shown in Figure 2, it produces a minimum extended connected dominating set of G, given by $D_{mec} = \{4, 5, 6, 9\}$.

5 | Correctness, Time and Space Complexity

Lemma 5.1. *Assume that at every sweep frontier the exchange choices in Lemmas 3.2–3.4 preserve a minimum-cardinality completion of the processed prefix. Then* **MECDSIG** *returns a minimum ECDS of G.*

Proof. Maintain three sweep invariants: (i) every processed vertex is extended dominated; (ii) the selected vertices induce a connected subgraph; and (iii) the selected prefix can be extended to a minimum solution. Lemma 3.1 handles the singleton case. Otherwise, Lemmas 3.2 and 3.3 initialize the invariants. Whenever an unprocessed left endpoint lies beyond the current frontier, Lemma 3.4 advances to an adjacent rightmost candidate, preserving connectivity and the assumed exchange property. The frontier increases strictly, so the process terminates. At termination every interval is processed; hence the output is an ECDS, and invariant (iii) gives minimum cardinality. □

Theorem 5.2. *Given intervals already sorted by nondecreasing right endpoint,* **MECDSIG** *can be implemented in $O(n)$ time. If sorting is required, the total running time is $O(n \log n)$.*

Proof. Store the sorted interval endpoints in arrays and maintain forward-only pointers for the current scan, the largest candidate, the second-largest candidate, and the processed frontier. Every pointer advances at most n times, and each interval is marked once. All set insertions and frontier updates are constant-time array operations. The sweep therefore costs $O(n)$. Sorting arbitrary input intervals by right endpoint costs $O(n \log n)$ and dominates the total preprocessing time. □

Theorem 5.3. *The algorithm* **MECDSIG** *requires $O(n)$ memory space.*

Proof. The endpoint arrays, marking array, and output set each use at most $O(n)$ entries. The remaining counters and pointers require constant space. Thus the total space consumption is $O(n)$. □

6 | Real Applications

A real-world application of an extended connected dominating set (ECDS) in interval graphs can be found in linear wireless sensor networks (LWSN), particularly in monitoring linear infrastructures like railway tracks, pipelines, roads, or bridges.

Here, we consider the problem of monitoring a railway track network embedded with wireless sensors.

Stepwise representation of efficient monitoring of a railway track through wireless sensors using an interval graph model based on extended connected dominating set

Step 1: The proposed problem set up Suppose that a railway track is embedded with a set of wireless sensors to monitor vibrations, cracks, or structural integrity. These sensors communicate wirelessly and can only reach nearby nodes due to the limited range. Our aim is to efficiently manage communication among sensors for energy-efficient routing using a minimum number of active sensors, to ensure coverage and connectivity, data aggregation, and fault tolerance.

Step 2: Model the network as an interval graph. Assume that each sensor embedded on the railway track has a fixed communication range along the track. Let there be six wireless sensors S_1, S_2, S_3, S_4, S_5, and S_6. Represent the range of sensor S_i by $I_i = [a_i, b_i]$, where $a_i = x_i - r$, $b_i = x_i + r$, x_i is the sensor position, and r is its communication radius. Two sensor vertices are adjacent exactly when their intervals overlap.

For instance, let $I_1 = [1, 4]$, $I_2 = [3, 6]$, $I_3 = [4.5, 7.5]$, $I_4 = [5.5, 8.5]$, $I_5 = [7, 10]$, and $I_6 = [8, 11]$. Each interval has length three, so the corresponding equal communication radius is $r = 1.5$. The interval graph is shown in Figure 3.

Step 3: Computation of the least number of active wireless sensors that create a communication backbone for the network

Applying MECDSIG to Figure 3 gives $D_{mec} = \{S_3, S_4\}$. The pair is connected, and every other sensor is adjacent to a selected sensor or satisfies the distance-two redundancy condition. No singleton is feasible, so the pair is minimum for this six-sensor instance. Operationally, the selected sensors may be treated as a communication backbone while nonselected sensors use reduced-duty modes. This graph-theoretic example does not by itself prove energy savings or resilience to arbitrary sensor failures; those claims require a radio-energy model and failure experiments.

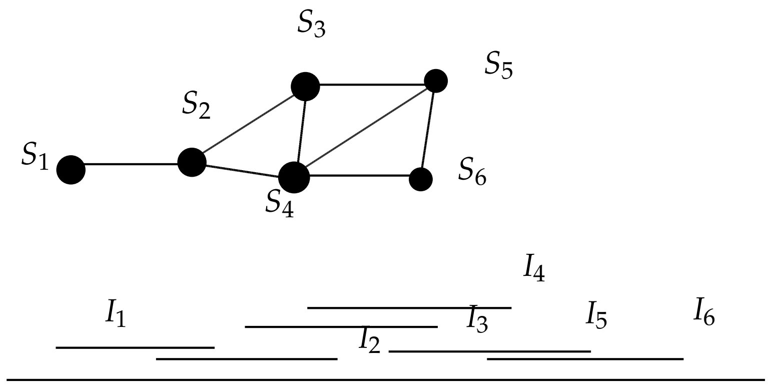

Figure 3. IG and its interval representation of the proposed railway network.

Limitations: The proposed graph model has some limitations that are mentioned below.

- The proposed model assumes that all sensors have a fixed and identical communication range, whereas in real railway environments, the communication range may vary due to obstacles, weather conditions, signal interference, or hardware limitations.
- Interval graphs are mainly suitable for linear railway tracks, not complex railway networks.
- Sensor failures and battery depletion are not considered in the model.
- The solution does not address communication delay or network congestion issues.

7 | Conclusion

In graph theory, domination is an important concept used to study the reliability and weakness of communication networks, which can be represented by graphs. There are several types of domination, depending on the properties of the dominating set. Among them, extended connected domination (ECD) is particularly useful because of its practical applications.

In this paper, we study structural properties of extended connected dominating sets in right-endpoint-ordered interval graphs. Under the exchange conditions stated in Lemma 5.1, the proposed sweep returns a minimum ECDS. With presorted endpoints and forward-only pointers, the sweep uses $O(n)$ time and $O(n)$ space; arbitrary input requires $O(n \log n)$ sorting time.

In addition, we present a real-life application of our results. In particular, we show how interval graphs and extended connected dominating sets can be used for efficient monitoring of railway tracks using wireless sensor networks. Nevertheless, the proposed approach is based on equal communication capabilities of sensors, is more suitable for linear railway structures, does not incorporate possible sensor malfunction, and overlooks practical factors such as transmission delay and network congestion.

As future work, we plan to design optimal algorithms for finding minimum ECDS in more complex graph classes, such as communication networks, permutation graphs, circular-arc graphs, and other intersection graphs.

Acknowledgment

We thank our colleagues for their valuable support, encouragement, and constructive suggestions throughout this research work.

Author Contributions

All authors contributed equally.

Funding

The authors confirm that this research received no specific grant or financial support from any funding agency, institution, or organization.

Data Availability

No external dataset was used. The six-sensor interval instance is fully specified in the paper.

Competing Interests

All authors confirm that they have no competing interests to declare regarding this research and its publication.

Generative-AI Interests

Generative AI assisted with language, LaTeX preparation and technical representation of the paper.

References

[1] Berge, C. (1962). *The theory of graphs and its applications*. Methuen & Co. Ltd., London. https://books.google.com/books/about/The_Theory_of_Graphs_and_Its_Application.html?id=Jd9QAAAAMAAJ
[2] Ore, O. (1962). Theory of graphs. *American mathematical society*. https://doi.org/10.1090/coll/038
[3] Haynes, T. W., Hedetniemi, S., & Slater, P. (1998). *Fundamentals of domination in graphs*. CRC Press. https://doi.org/10.1201/9781482246582
[4] Stojmenovic, I., Seddigh, M., & Zunic, J. (2002). Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE transactions on parallel and distributed systems*, *13*(1), 14-25. https://doi.org/10.1109/71.980024
[5] Carle, J., & Simplot-Ryl, D. (2004). Energy-efficient area monitoring for sensor networks. *Computer*, *37*(2), 40-46. https://doi.org/10.1109/MC.2004.1266294
[6] Sampathkumar, E., & Walikar, H. B. (1979). The connected domination number of a graph. *Journal of mathematical and physical sciences*, *13*(6), 607–613.
[7] D'Atri, A., & Moscarini, M. (1988). Distance-hereditary graphs, Steiner trees, and connected domination. *SIAM journal on computing*, 17(3), 521-538. https://doi.org/10.1137/0217032
[8] Demaine, E. D., Fomin, F. V., Hajiaghayi, M., & Thilikos, D. M. (2005). Fixed-parameter algorithms for (k, r)-center in planar graphs and map graphs. *ACM transactions on algorithms* (*TALG*), 1(1), 33-47. https://doi.org/10.1145/1077464.1077468
[9] Kundu, S., & Majumder, S. (2016). A linear time algorithm for optimal k-hop dominating set of a tree. *Information processing letters*, 116(2), 197-202. https://doi.org/10.1016/j.ipl.2015.07.014

[10] Slater, P. J. (1976). R-domination in graphs. *Journal of the ACM (JACM), 23*(3), 446-450. https://doi.org/10.1145/321958.321964
[11] Meybodi, M. A. (2022). k-Efficient domination: Algorithmic perspective. *Discrete mathematics, algorithms and applications,* 14(08), 2250051. https://doi.org/10.1142/S1793830922500513
[12] Kumar, M. K., & Reddy, L. S. (2013). Inverse Roman domination in graphs. *Discrete mathematics, algorithms and applications,* 5(03), 1350011. https://doi.org/10.1142/S1793830913500110
[13] Senthilkumar, B., Naresh Kumar, H., & Venkatakrishnan, Y. B. (2021). Bounds on total edge domination number of a tree. *Discrete mathematics, algorithms and applications,* 13(02), 2150011. https://doi.org/10.1142/S1793830921500117
[14] Sahul Hamid, I., & Balamurugan, S. (2015). Vertex criticality with respect to isolate domination. *Discrete mathematics, algorithms and applications,* 7(02), 1550010. https://doi.org/10.1142/S179383091550010X
[15] Duraisamy, P., & Esakkimuthu, S. (2021). Linear programming approach for various domination parameters. *Discrete mathematics, algorithms and applications,* 13(01), 2050096. https://doi.org/10.1142/S1793830920500962
[16] Gupta, P., Goyal, A., & Arumugam, S. (2023). Line-set domination in graphs. *Discrete mathematics, algorithms and applications,* 15(05), 2250117. https://doi.org/10.1142/S1793830922501178
[17] Chang, G. J. (1998). Algorithmic aspects of domination in graphs. In *Handbook of combinatorial optimization: Volume*1–*3* (pp. 1811-1877). Boston, MA: Springer US. https://doi.org/10.1007/978-1-4613-0303-9_28
[18] Kang, C. X., & Yi, E. (2018). Bounds on the sum of domination number and metric dimension of graphs. *Discrete mathematics, algorithms and applications,* 10(05), 1850066. https://doi.org/10.1142/S1793830918500660
[19] Rana, A. (2021). A survey on the domination of fuzzy graphs. *Discrete mathematics, algorithms and applications,* 13(01), 2130001. https://doi.org/10.1142/S1793830921300010
[20] Cockayne, E. J., Dawes, R. M., & Hedetniemi, S. T. (1980). Total domination in graphs. *Networks,* 10(3), 211-219. https://doi.org/10.1002/net.3230100304
[21] Chao, H. S., Hsu, F. R., & Lee, R. C. T. (2000). An optimal algorithm for finding the minimum cardinality dominating set on permutation graphs. *Discrete applied mathematics,* 102(3), 159-173. https://doi.org/10.1016/S0166-218X(98)00145-0
[22] Van Der Merwe, L. C., Mynhardt, C. M., & Haynes, T. W. (1998). Criticality index of total domination. *Congressus numerantium*, 67-74.
[23] Van der Merwe, L. C., Mynhardt, C. M., & Haynes, T. W. (1998). Total domination edge critical graphs. *Utilitas mathematica,* 54, 229-240. https://dc.etsu.edu/etsu-works/13145/
[24] Haynes, T. W. (1998). *Domination in graphs: Advanced topics*. Routledge. https://doi.org/10.1201/9781315141428
[25] Haynes, T. W., Mynhardt, C. M., & van der Merwe, L. C. (2001). Total domination edge critical graphs with maximum diameter. *Discuss. Math. Graph Theory,* 21(2), 187-205.
[26] Sumner, D. P., & Blitch, P. (1983). Domination critical graphs. *Journal of combinatorial theory, series B, 34*(1), 65-76. https://doi.org/10.1016/0095-8956(83)90007-2
[27] Carlisle, M. C., & Lloyd, E. L. (1991, May). On the k-coloring of intervals. *International conference on computing and information* (pp. 90-101). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-54029-6_157
[28] Jungck, J. R., Dick, G., & Dick, A. G. (1982). Computer-assisted sequencing, interval graphs, and molecular evolution. *Biosystems,* 15(3), 259-273. https://doi.org/10.1016/0303-2647(82)90010-7

[29] Golumbic, M. C. (2004). *Algorithmic graph theory and perfect graphs* (Vol. 57). Elsevier. https://shop.elsevier.com/books/algorithmic-graph-theory-and-perfect-graphs/golumbic/978-0-444-51530-8
[30] Fabri, J. (1979). *Automatic storage optimization*. UMI Research Press.
[31] Ohtsuki, T., Mori, H., Kuh, E., Kashiwabara, T., & Fujisawa, T. (1979). One-dimensional logic gate assignment and interval graphs. *IEEE Transactions on Circuits and Systems, 2*6(9), 675-684. https://ieeexplore.ieee.org/document/1084695/
[32] Hashimoto, A., & Stevens, J. (1971. Wire routing by optimizing channel assignment within large apertures. *Proceedings of the 8th design automation workshop* (pp. 155-169). ACM Digital Library. https://doi.org/10.1145/800158.805069
[33] Olariu, S. (1991). An optimal greedy heuristic to color interval graphs. *Information processing letters, 3*7(1), 21-25. https://doi.org/10.1016/0020-0190(91)90245-D
[34] Guha, S., & Khuller, S. (1998). Approximation algorithms for connected dominating sets. *Algorithmica, 2*0(4), 374-387. https://doi.org/10.1007/PL00009201
[35] Fink, J. F., & Jacobson, M. S. (1985). On n-domination, k-domination, and graph products with applications to coverings and independence. *Congr. Numer. 4*9, 87–95.
[36] Wu, J. (2002). Extended dominating-set-based routing in ad hoc wireless networks with unidirectional links. *IEEE transactions on parallel and distributed systems,* 1*3*(9), 866-881. https://doi.org/10.1109/TPDS.2002.1036062
[37] Wu, J., Cardei, M., Dai, F., & Yang, S. (2006). Extended dominating set and its applications in ad hoc networks using cooperative communication. *IEEE transactions on parallel and distributed systems,* 17(8), 851-864. https://doi.org/10.1109/TPDS.2006.103
[38] Gao, Z., Shi, Y., Xi, C., & Yue, J. (2023). The extended dominating sets in graphs. *Asia-pacific journal of operational research, 4*0(05), 2340015. https://doi.org/10.1142/S0217595923400158