Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 28, no. 3, pp. 69–85 (2024)
DOI: 10.7155/jgaa.v28i3.2972

Some Algorithmic Results for Eternal Vertex Cover Problem in Graphs

Kaustav Paul[1] *Arti Pandey*[1]

[1]Department of Mathematics
Indian Institute of Technology Ropar
Nangal Road, Rupnagar, Punjab 140001, INDIA

Submitted: June 2023 Accepted: May 2024 Published: September 2024
Article type: Regular paper Communicated by: C.-C. Lin, B. M.-T. Lin, G. Liotta

Abstract. The eternal vertex cover problem is a variant of the vertex cover problem. It is a two-player (attacker and defender) game in which, given a graph $G = (V, E)$, the defender needs to allocate guards at some vertices so that the allocated vertices form a vertex cover. The attacker can attack one edge at a time, and the defender needs to move the guards along the edges such that at least one guard moves through the attacked edge and the new configuration still remains a vertex cover. The attacker wins if no such move exists for the defender. The defender wins if there exists a strategy to defend the graph against infinite sequence of attacks. The minimum number of guards with which the defender can form a winning strategy is called the *eternal vertex cover number* of G, and is denoted by $evc(G)$. Given a graph G, the problem of finding the eternal vertex cover number is NP-hard for general graphs and remains NP-hard even for bipartite graphs. We have given a polynomial time algorithm to find the Eternal vertex cover number in chain graphs and P_4-sparse graphs. We have also given a linear-time algorithm to find the eternal vertex cover number for split graphs, an important subclass of chordal graphs.

1 Introduction

Given a graph $G = (V, E)$, several problems have been formulated on the vertex attacking problem. Mobile robots are used to defend against those attacks. Several papers have explored different issues related to safeguarding the vertices of G against a series of attacks. A variant of this problem is the *eternal domination problem*, also known as the *eternal security problem* (refer to [10]). In this scenario, there is a restriction of having at most one guard positioned at each vertex. Guards are capable of safeguarding the vertex they occupy and can relocate to neighboring vertices to

E-mail addresses: kaustav.20maz0010@iitrpr.ac.in (Kaustav Paul) arti@iitrpr.ac.in (Arti Pandey)

defend against attacks. The sequence of attacks can extend infinitely, and it necessitates that the arrangement of guards forms a dominating set both before and after each attack is repelled. In 2009, Klostermeyer et al. redirected their attention towards infinite sequences of attacks on edges rather than the vertices [11]. To defend against an attack, a guard situated at an incident vertex traverses the attacked edge. Now, we define the problem formally.

Formally, the *Eternal vertex cover problem* is a two-player (attacker and defender) game such that given a graph $G = (V, E)$, the defender is permitted to allocate guards in some vertices of G so that the vertices, where guards are allocated form a vertex cover. The attacker can attack one edge at a time. Now, for each guard, the defender can either move the guard to one of its neighbours or can keep it untouched, such that at least one guard from any of the endpoints of the attacked edge moves through the edge to settle at the other endpoint. So, the new allocation should also remain a vertex cover to defend the next attack. If no such configuration exists, then the attacker wins. If the allocation can defend any infinite sequence of attacks, then the defender wins. The minimum number of guards with which a winning strategy for the defender can be formed is known as the *eternal vertex cover number* of G, and is denoted by $evc(G)$. In this paper, we are assuming that at most one guard can be allocated to each vertex. If C_i be the allocation of the guards before the i-th attack, then after defending the i-th attack by moving the guards to configuration C_{i+1}, C_{i+1} needs to be a vertex cover(for each $i \in \mathbb{N}$), to form a winning strategy for the defender. If it is not then the $(i+1)$-th attack will be on the edge which is not covered by C_{i+1} and the attacker will win. So after reconfiguring at each step, the vertices where the guards are allocated should form a vertex cover. The problem of finding the eternal vertex cover number is known as eternal vertex cover problem. A preliminary version of this article was published in the proceedings of the WALCOM 2023 conference [13].

1.1 Related Works

In 2009, Klostermeyer et al. introduced the eternal vertex cover problem. In the same paper, they showed that for any graph G, we have $mvc(G) \leq evc(G) \leq 2mvc(G)$ [11]. They gave a characterization of the graphs for which $evc(G) = 2mvc(G)$ is attained [11]. Babu et al. have given some special graph classes for which $evc(G)$ attains the lower bound, that is $evc(G) = mvc(G)$ [2].

The first NP-hard result for this problem was given by Fomin et al., where they showed that the problem is NP-hard for general graphs [7]. They also show that the problem is in PSPACE, though it is still unknown whether the problem belongs to the class NP or not [7]. In the same paper, they also proposed a 2-approximation algorithm for the problem [7]. Babu et al. proved that the problem remains NP-hard even for locally connected graphs, which includes the family of all the biconnected internally triangulated planar graphs [2]. They proposed polynomial-time algorithms for cactus graphs and chordal graphs [4, 5]. The most recent development on the NP-hardness result has also been produced by Babu et al., where they proved the NP-hardness of the problem for the class of bipartite graphs. Along with that, they also proved that the problem can also be solved in polynomial time for co-bipartite graphs [3]. Araki et al. computed the $evc(G)$ for generalized trees where each edge of the tree is replaced by some elementary bipartite graphs [1].

1.2 Our Results

In this paper, we further extend the algorithmic study of the problem by proposing polynomial-time algorithms for some special graph classes. The rest of the paper is organized as follows: In Section 2.1, all notations and definitions used in the paper are presented. In Section 2.2, some theorems from existing literature are stated, which are used in the proofs presented in this paper. In Section 2.3, eternal vertex cover number is provided for some special subclasses of bipartite graphs. In Section 3, a linear-time algorithm is given to compute $evc(G)$ in chain graphs. In Section 4, a linear-time algorithm to compute $evc(G)$ in split graphs is presented. In Section 5, a polynomial time algorithm to compute $evc(G)$ for P_4-sparse graphs is presented. Finally, Section 6 concludes the paper.

2 Preliminaries

2.1 Definitions and Notations

All graphs considered in this paper are finite, connected, undirected, and simple. Let $G = (V, E)$ be a graph. The set of neighbours of a vertex v in G is denoted by $N(v)$. A set $I \subseteq V$ is called an *independent set* of G if for all $u, v \in I$, $\{u, v\} \notin E$. *Degree* of a vertex $v \in V$ is the number of neighbours of v in G and it is denoted as $deg(v)$. Given a subset V' of V, the number of neighbours of v in V' is denoted by $deg_{V'}(v)$. A vertex $v \in V$ is said to be a *cut vertex* if $G[V \setminus \{v\}]$ is not connected. The *join* of two graphs H_1 and H_2 is a graph formed from disjoint copies of H_1 and H_2 by connecting each vertex of $V(H_1)$ to each vertex of $V(H_2)$.

A *vertex cover* S of $G = (V, E)$ is a subset of V, which contains at least one endpoint from each edge in E. A vertex cover of minimum cardinality is called a *minimum vertex cover*. The cardinality of a minimum vertex cover is denoted as *minimum vertex cover number* or $mvc(G)$. Given $B \subseteq V$, the cardinality of the minimum vertex cover containing B is denoted as $mvc_B(G)$. If the graph induced on S, i.e. $G[S]$ is connected, then S is called a *connected vertex cover*. The cardinality of minimum connected vertex cover is denoted as $cvc(G)$. The independent set of maximum cardinality is called *maximum independent set* of G and its cardinality is denoted as $mis(G)$.

Consider a graph $G = (V, E)$ with $|V| = n$ and $|E| = m$. A *hamiltonian cycle* of a graph $G = (V, E)$ is a cycle in G, that visits each $v \in V$ exactly once. A graph possessing a hamiltonian cycle is known as *hamiltonian graph*. A graph $G = (V, E)$ is said to be *k-regular* if $deg(v) = k$, for each $v \in V$.

Let $G = (X \cup Y, E)$ be a bipartite graph. G is said to be a *chain graph* if vertices in X can be ordered $\{x_1, x_2, \ldots, x_{|X|}\}$, such that $N(x_1) \subseteq N(x_2) \subseteq \ldots \subseteq N(x_{|X|})$. Similarly vertices of Y can be ordered $\{y_1, y_2, \ldots, y_{|Y|}\}$, such that $N(y_1) \supseteq N(y_2) \supseteq \ldots \supseteq N(y_{|Y|})$. Throughout this manuscript, we assume all the chain graphs considered are connected. The cardinality of X and Y are denoted by p and q, respectively, in this paper. An example of a chain graph is given below in Figure 1.

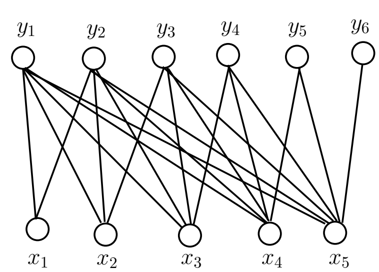

Figure 1: An example of chain graph $G = (X \cup Y, E)$

A graph $G = (V, E)$ is called a *split graph* if V can be partitioned in K and I, such that K is clique and I is an independent set. An example of a chain graph is given below in Figure 2. The class of split graphs is an important subclass of the class of chordal graphs.

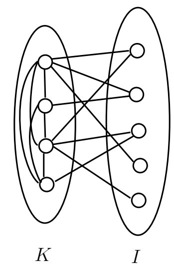

Figure 2: An example of split graph $G = (K \cup I, E)$

Cographs are P_4-free graphs. A graph $G = (V, E)$ is said to be P_4-*sparse graph* if a subgraph induced on any 5 vertices of G contains at most one P_4. A *spider* is a graph $G = (V, E)$, where V admits a partition in three subsets S, C and R such that

- $C = \{c_1, \ldots, c_l\}$ $(l \geq 2)$ is a clique.
- $S = \{s_1, \ldots, s_l\}$ is an independent set.
- Every vertex in R is adjacent to every vertex in C and nonadjacent to all vertex of S.

A spider $G(S, C, R)$ is said to be a *thin spider* if for every $i \in [l]$, $N(s_i) = \{c_i\}$ and it is called a *thick spider* if for every $i \in [l]$, $N(s_i) = C \setminus \{c_i\}$ (refer to Figure 3). This definition of spider can be found in [9].

(a) Thin spider (b) Thick spider

Figure 3: Examples of spiders with spider partition (S, C, R)

2.2 Existing Results

For the sake of convenience, we are stating some important theorems which will be used in the proofs presented in our paper.

Theorem 1 *[2] Let $G = (V, E)$ be a graph with no isolated vertex and every minimum vertex cover of G is connected. If for every vertex $v \in V$, there exists a minimum vertex cover S_v of G such that $v \in S_v$, then $evc(G) = mvc(G)$. Otherwise, $evc(G) = mvc(G) + 1$.*

Theorem 2 *[11] Let $G = (V, E)$ be a nontrivial, connected graph and let D be a minimum connected vertex cover of G. Then $evc(G) \leq |D| + 1$ and $D \cup \{v\}$ forms an initial configuration of eternal vertex cover, where $v \in V \setminus D$.*

Theorem 3 *[2] Let $G = (V, E)$ be a graph with at least 2 vertices and X be the set of cut vertices of G. If every minimum vertex cover S of G with $X \subseteq S$ is connected, then the following characterization holds: $evc(G) = mvc(G)$ if and only if for every vertex $v \in V \setminus X$, there exists a minimum vertex cover S_v of G such that $X \cup \{v\} \subseteq S_v$.*

Theorem 4 *[2] Let $G = (V, E)$ be a graph with no isolated vertices. If $evc(G) = mvc(G)$, then for every vertex $v \in V$, there is some minimum vertex cover of G containing v.*

2.3 Eternal Vertex Cover Number for Some Subclasses of Bipartite Graph

For a k-regular bipartite graph, the following observation can be made.

Observation 1 *Given a k-regular bipartite graph $G = (X \cup Y, E)$, for each $e \in E$, there exists a perfect matching that contains e.*

Note that, if the initial guard allocation is X (or Y), then attack on any edge e can be defended by moving the guards to Y (or X) through the perfect matching that contains e. So, from the Observation 1 it can be concluded that for a k-regular bipartite graph G, $evc(G) = mvc(G) = |X| = |Y|$.

For a hamiltonian bipartite graph $G = (X \cup Y, E)$ (with $|X| = |Y| = n$), suppose a hamiltonian cycle of G is $v_1 v_2 \cdots v_{2n} v_1$, where $X = \{v_1, v_3, \ldots, v_{2n-1}\}$ and $Y = \{v_2, v_4, \ldots, v_{2n}\}$. Then, we have the following observation.

Observation 2 *Given a hamiltonian bipartite graph $G = (X \cup Y, E)$ and a hamiltonian cycle $C = v_1 v_2 \cdots v_{2n} v_1$ of G; for every edge $e \in E$, there exists a perfect matching of G that contains e.*

Proof: Note that $X = \{v_1, v_3, \ldots, v_{2n-1}\}$ and $Y = \{v_2, v_4, \ldots, v_{2n}\}$ and G has at least two perfect matchings U_1 and U_2, where $U_1 = \{v_1 v_2, v_3 v_4, \ldots, v_{2n-1} v_{2n}\}$ and $U_2 = \{v_2 v_3, v_4 v_5, \ldots, v_{2n} v_1\}$. Now consider any edge $e = v_i v_j$, where without loss of generality, let us assume $v_i \in X$ and $v_j \in Y$. If v_i and v_j are consecutive vertices in the cycle, then e is an edge of the cycle C. Hence e is contained in either U_1 or U_2. Now if v_i and v_j are not consecutive in C, then note that the edge e splits the cycle C in two cycles C_1 and C_2 (refer to Figure 4).

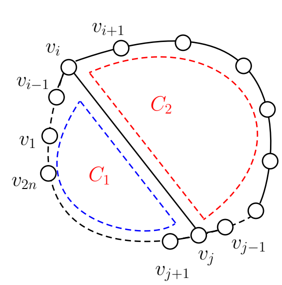

Figure 4: A Hamiltonian Bipartite graph

Note that C_1 and C_2 are also even cycles, as G is a bipartite graph. Hence either $U_1 \cap C_2$ contains both the edges v_iv_{i+1} and $v_{j-1}v_j$ or $U_2 \cap C_2$ contains both the edges. Without loss of generality, let $U_2 \cap C_2$ contain both the two edges. Hence U_1 does not contain the edges v_iv_{i+1} and $v_{j-1}v_j$ but contains the edges $v_{i-1}v_i$ and v_jv_{j+1}. But since $U_1 \cap U_2 = \emptyset$, then U_2 does not contain the edges $v_{i-1}v_i$ and v_jv_{j+1}. Hence, $(U_1 \cap C_2) \cup (U_2 \cap C_1) \cup \{v_iv_j\}$ forms a perfect matching of G. Hence, for every edge $e \in E$, there exists a perfect matching of G that contains e. □

From Observation 2, it can be concluded that $evc(G) = mvc(G) = |X| = |Y|$, where X (or Y) is the initial configuration of guards.

3 A Polynomial Time Algorithm for Chain Graphs

In this section, we present a linear-time algorithm to compute the $evc(G)$ of a given chain graph G. We also show that for a chain graph G, $evc(G) \in \{mvc(G), mvc(G)+1, mvc(G)+2\}$.

For a chain graph $G = (X \cup Y, E)$, we assume that it is connected and $|X| \leq |Y|$. The eternal vertex cover problem in the class of chain graphs is studied in 2 exhaustive cases: (i) chain graphs having pendant vertices only in Y, and (ii) chain graphs having pendant vertices both in X and Y or only in X.

3.1 For chain graphs where only Y can have pendant vertices

In this section we will assume that either there exists no pendant vertex in the graph or only Y contains pendant vertices. Note that a minimum vertex cover of a chain graph can be computed in linear time [12]. Let S be a minimum vertex cover G. If $|S| < min\{|X|, |Y|\}$, then $X \cap S \neq \emptyset$ and $Y \cap S \neq \emptyset$. First, we state the following observation.

Observation 3 *Given a chain graph $G = (X \cup Y, E)$ and a minimum vertex cover S of G; if $x_i \in S$, then $x_j \in S$, for each $i \leq j \leq p$ and if $y_i \in S$, then $y_j \in S$, for each $1 \leq j \leq i$.*

Proof: For the sake of contradiction, let $x_i \in S$, but $x_j \notin S$, for some $i < j$. Since $x_j \notin S$, then $N(x_j) \subseteq S$. This implies $N(x_i) \subseteq N(x_j) \subseteq S$, hence $S \setminus \{x_i\}$ is also a vertex cover, which contradicts the assumption that S is a minimum vertex cover. Hence $x_j \in S$. Similarly, the proof can be done for vertices of Y. □

Lemma 1 *For a chain graph $G = (X \cup Y, E)$, if $mvc(G) < min\{|X|, |Y|\}$, then $evc(G) = mvc(G) + 1$.*

Proof: Let S be a minimum vertex cover. Note that, if $|S| < min\{|X|, |Y|\}$, then S has nonempty intersection with both X and Y. Hence, by Observation 3, S contains both x_p and y_1. This implies that S is a connected vertex cover, and hence $cvc(G) = mvc(G)$. Hence, every vertex cover of size $mvc(G)$ is connected (as an example, refer to Figure 5, where the set of red vertices, that is $\{y_1, y_2.x_4, x_5\}$, forms a minimum vertex cover which is also connected).

But there does not exist any minimum vertex cover S' that contains x_1 (If $x_1 \in S'$, then by Observation 3, $X \subseteq S'$, which implies that $mvc(G) \geq |X| > |S|$, a contradiction). So, by Theorem 1, if for a chain graph G, $mvc(G) < min\{|X|, |Y|\}$, then $evc(G) = mvc(G) + 1$ and by Theorem 2, the initial configuration of guards is $\{u\} \cup S$, where $u \in V(G) \setminus S$. □

Now we consider the case when $mvc(G) = min\{|X|, |Y|\}$. Again two cases may arise, one is $|X| < |Y|$ and the another is $|X| = |Y|$.

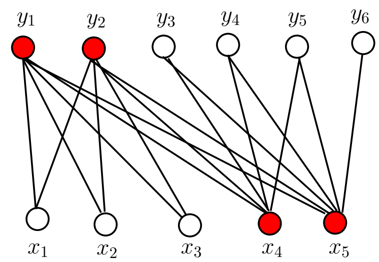

Figure 5: A chain graph G with $mvc(G) < min\{|X|, |Y|\}$

Claim 1 *For a chain graph $G = (X \cup Y, E)$, if $|X| < |Y|$ and $mvc(G) = min\{|X|, |Y|\}$, then $evc(G) \neq mvc(G)$.*

Proof: Let $evc(G) = mvc(G)$, then $x_p \in S$, for any minimum vertex cover S of G (by Observation 3). If the attacker attacks $\{x_p, y_q\}$, then the guard at x_p moves to y_q and rest of the guards are adjusted so that the new configuration remains a vertex cover. Since in the new configuration, $y_q \in S'$, (where S' is a minimum vertex cover), by Observation 3, $Y \subseteq S'$. Which leads to a contradiction since $mvc(G) < |Y|$. Hence $mvc(G) \neq evc(G)$. □

Lemma 2 *For a chain graph $G = (X \cup Y, E)$, if $|X| < |Y|$, $mvc(G) = min\{|X|, |Y|\}$, and there exists a minimum vertex cover containing x_p, y_1, then $mvc(G) = evc(G) + 1$.*

Proof: If for a given chain graph G, there exists a minimum vertex cover that contains x_p, y_1, then $cvc(G) = mvc(G)$. Since $evc(G) \neq mvc(G)$ and by Theorem 2, $evc(G) \leq cvc(G) + 1$, we may conclude that $evc(G) = mvc(G) + 1$. □

Now let us consider the case when there does not exist any minimum vertex cover that contains x_p, y_1, $mvc(G) = min\{|X|, |Y|\}$ and $|X| < |Y|$. In this case, X is the only minimum vertex cover.

Lemma 3 *For a given chain graph $G = (X \cup Y, E)$, if $mvc(G) = min\{|X|, |Y|\}$ and $|Y| = |X|+1$, and X is the only minimum vertex cover of G, then $evc(G) = mvc(G) + 1$.*

Proof: Let $|N(x_1)| > 2$ or $y_{q-1} \notin N(x_1)$. If the initial configuration is $\{x_1, x_2, \ldots, x_p, y_q\}$, attack any edge $\{x_i, y_j\}(y_j \neq y_q)$; by Hall's Theorem there exists a perfect matching from $X \setminus \{x_i\}$ to $Y \setminus \{y_j, y_q\}$, since $| \cup_{j=1}^{k} N(x_j) | \geq k + 1$, for each $k \in [p]$ (Refer to Figure 6). So all the guards can be moved from $X \cup \{y_q\}$ to Y.

Now if Y is the guard allocation configuration and $\{y_j, x_i\}(y_j \neq y_q)$ is attacked then the next configuration will be $X \cup \{y_q\}$. If $y_j = y_q$ then the configuration will be $X \cup \{y_{q-1}\}$. Thus any infinite sequence of attack can be defended using $mvc(G) + 1$ guards. So $evc(G) = mvc(G) + 1$. If $|N(x_1)| \leq 2$ and $y_{q-1} \in N(x_1)$, then $|Y| \leq 3$ and $|X| \leq 2$. For this case, it is easy to observe $evc(G) = mvc(G) + 1$. □

Observation 4 *Let $G = (X \cup Y, E)$ is a chain graph with $|Y| > |X| + 1$ for which the only minimum vertex cover is X and S be a vertex cover of size $mvc(G)+1$. If $| S \cap Y | \geq 2$ and $y_i \in S$, then $y_j \in S$, for each $j \in [i]$. We may also conclude that there exists two kinds of vertex covers of size $mvc(G) + 1$*

i. $X \cup \{y_i\}$; $i \in [q]$.

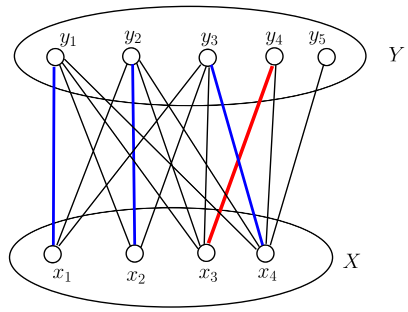

Figure 6: The attacked edge is x_3y_4, highlighted in red. The perfect matching from $X \setminus \{x_i\}$ to $Y \setminus \{y_j, y_q\}$ is highlighted in blue.

ii. $\{y_1, \ldots, y_{i+1}, x_{i+1}, \ldots, x_p\}$; $i \in [p-2]$.

Let $k = min\{i \mid \{x_i, y_q\} \in E\}$.

Lemma 4 *For a given chain graph $G = (X \cup Y, E)$ with $mvc(G) = min\{|X|, |Y|\}$ and $|Y| > |X| + 1$, if X is the only minimum vertex cover of G and $| \cup_{j=1}^{k-1} N(x_j) | = k$, then $evc(G) = mvc(G) + 1$.*

Proof: By above definition $k = min\{i \mid \{x_i, y_q\} \in E\}$, if $| \cup_{j=1}^{k-1} N(x_j)| = k$. Then any attack can be defended by moving the guards from the configuration $X \cup \{y_q\}$ to configuration $\{y_1, \ldots, y_k, x_k, \ldots, x_p\}$ (or from $\{y_1, \ldots, y_k, x_k, \ldots, x_p\}$ to $X \cup \{y_q\}$). So, in this case $evc(G) = mvc(G) + 1$. □

Let $V' = \{i \mid | \cup_{j=1}^{i} N(x_j)| = i + 1\}$.

Lemma 5 *For a given chain graph $G = (X \cup Y, E)$ with $mvc(G) = min\{|X|, |Y|\}$ and $|Y| > |X| + 1$, if X is the only minimum vertex cover of G and $| \cup_{j=1}^{k-1} N(x_j)| > k$, then $evc(G) = mvc(G) + 2$.*

Proof: $mvc(G) = |X|$, and $X \cup \{y_1\}$ forms a connected vertex cover. Hence by Theorem 2, $evc(G) \leq |X \cup \{y_1\}| + 1 = mvc(G) + 2$. If $| \cup_{j=1}^{k-1} N(x_j)| > k$ and $V' \neq \emptyset$, then let $l = max\{i \mid i \in V'\}$. If the initial configuration is of type-ii, then attack $\{x_p, y_q\}$ and make the configuration $X \cup \{y_q\}$, if possible. Then attack $\{x_{l+1}, y_{l+1}\}$, the guard at x_{l+1} moves to y_{l+1} and since $\{y_q, x_{l+1}\} \notin E$, so there does not exist any guard which can move to x_{l+1}, hence no defending move exists, hence $evc(G) = mvc(G) + 2$.

If the set $V' = \emptyset$, then $| \cup_{j=1}^{i} N(x_j) | > i + 2$, which implies all vertex covers of size $mvc(G) + 1$ are of type-i. Now whatever the initial configuration be attack $\{x_p, y_q\}$. The configuration after defending this should be $X \cup \{y_q\}$. Now attack $\{x_{k-1}, y_{k-1}\}$, the guard at x_{k-1} moves to y_{k-1} now there is no guard which can move to x_{k-1} and form a vertex cover. So $evc(G) = mvc(G) + 2$. □

Now, consider the case when $|X| = |Y|$.

Lemma 6 *For a given chain graph $G = (X \cup Y, E)$, if $mvc(G) = min\{|X|, |Y|\}$, $|X| = |Y|$ and there exists a minimum vertex cover containing y_1 and x_p, then $evc(G) = mvc(G) + 1$.*

Proof: There exists a minimum vertex cover of G that contains both y_1 and x_p. This implies there exists $i \in [p]$, such that $\cup_{j=1}^{i} N(x_j) = \cup_{j=1}^{i} \{y_j\}$ and $evc(G) \in \{mvc(G), mvc(G) + 1\}$.

If $evc(G) = mvc(G)$, then the initial configuration can be of 3 types: (i) X, (ii) Y and (iii) $\{y_1, \ldots, y_i, x_{i+1}, \ldots, x_p\}$, $i \in [p]$.

If the initial configuration is of type-iii, then attack $\{x_1, y_1\}$ and change it to X if possible. Then attack $\{y_i, x_{i+1}\}$, so the guard at x_{i+1} moves to y_i and i guards at $x_1, x_2, \ldots, x_i$ have $i-1$ places, i.e. $y_1, y_2, \ldots, y_{i-1}$ to move. Hence no new configuration can be made which will form a vertex cover.

If the initial configuration is Y, then attack $\{y_i, x_{i+1}\}$. The guard at y_i moves to x_{i+1} and $p-i$ guards at $y_{i+1}, \ldots, y_p$ have $p-i-1$ places, i.e. $x_{i+2}, \ldots, x_p$ to move. Hence no new configuration can be made which will form a vertex cover.

This implies G can not be defended with $mvc(G)$ guards. So, $evc(G) = mvc(G) + 1$. □

Lemma 7 *For a given chain graph $G = (X \cup Y, E)$, with $mvc(G) = min\{|X|, |Y|\}$ and $|Y| = |X|$, if the only minimum vertex covers are X and Y, then $evc(G) = mvc(G)$.*

Proof: The only type of minimum vertex covers are X and Y. This implies $| \cup_{j=1}^{l} N(x_j) | \geq l+1$, for all $l \in [p-1]$. Now if the initial configuration is X, then attack on any edge $\{x_i, y_j\}$ can be defended by moving all the guards to Y, this can be done since by Hall's Theorem there exists a perfect matching in $(X \setminus \{x_i\}, Y \setminus \{y_j\})$. Similarly, if the initial configuration is Y, then attack on any edge $\{x_i, y_j\}$ can be defended by moving all the guards to X, this can also be done since by Hall's Theorem there exists a perfect matching in $(X \setminus \{x_i\}, Y \setminus \{y_j\})$. So, $evc(G) = mvc(G)$. □

Hence, from the above lemmas and observations, the following theorem can be concluded.

Theorem 5 *Given a chain graph $G = (X \cup Y, E)$, where only Y can contain pendant vertices, $evc(G)$ can be computed in time linear time.*

3.2 For chain graphs with pendant vertices in X or in X, Y both

If y_1 and x_p both have pendant vertices attached (consider that the graph is not K_2; for K_2, $evc(G) = mvc(G) = 1$), then there exists a minimum vertex cover that contains x_p and y_1, which implies $evc(G) \in \{mvc(G), mvc(G)+1\}$. Now if $evc(G) = mvc(G)$, then there exists a configuration such that a guard is allocated at the pendant vertex x_1 (if not then we can attack the edge $\{y_1, x_1\}$ and shift the guard at y_1 to x_1). This implies that there is no guard in y_1. Now attack $\{x_p, y_1\}$, then the guard at x_p moves to y_1 and the guard at x_1 has to stay at x_1. So in this new configuration, x_1 and y_1 both have guards allocated, a contradiction since no minimum vertex cover can contain the pendant vertex and its respective stem. So, $evc(G) = mvc(G) + 1$.

Now consider the case when only X has pendant vertices, that is, only y_1 is the stem. If $mvc(G) < min\{|X|, |Y|\}$, then $evc(G) = mvc(G) + 1$. If $mvc(G) = |X|$, then y_1 has only one pendant neighbour (otherwise $mvc(G) < |X|$, leading to a contradiction). Since $\{y_1, x_2, \ldots, x_p\}$ forms a minimum vertex cover and it is connected, $mvc(G) = cvc(G)$. This implies that $evc(G) \in \{mvc(G), mvc(G)+1\}$ Further, two cases may arise.

Case 1: $|X| < |Y|$

If $evc(G) = mvc(G)$, the initial guard allocation can be of 2 types; X and $\{y_1, \ldots, y_i, x_{i+1}, \ldots, x_p\}$.

If the initial configuration is X, then if $\{x_p, y_1\}$ is attacked then the guard at x_1 can not move anywhere, failing to produce a valid defending move.

If the initial configuration is $\{y_1, \ldots, y_i, x_{i+1}, \ldots, x_p\}$ then attack $\{x_1, y_1\}$, the only configuration it can form is X. But then, attacking $\{x_p, y_1\}$ will lead to a win for the attacker.

So, $evc(G) \neq mvc(G)$. This implies $evc(G) = mvc(G) + 1$.

Case 2: $|X| = |Y|$
If the initial guard allocation is X or Y, then attacking $\{x_p, y_1\}$ will lead to a win for the attacker.

If the initial configuration is $\{y_1, \ldots, y_i, x_{i+1}, \ldots, x_p\}$ then attack $\{x_1, y_1\}$, the only configuration it can form is X. But then, attacking $\{x_p, y_1\}$ will lead to a win for the attacker.

So, $evc(G) \neq mvc(G)$. This implies that $evc(G) = mvc(G) + 1$.

The above characterization is done by observing a property that for a given chain graph $G = (X \cup Y, E)$, whether there exists a minimum vertex cover S that contains both x_p and y_1 or not. This property can be checked in polynomial time for a given chain graph. Before starting the process of the algorithm, by preprocessing, an array $A[1, 2, \ldots, p]$ can be formed, where i^{th} cell contains the degree of x_i. If there exists a $j \in [p-1]$, such that $A[j] \leq j$, then there exists a minimum vertex cover of G that contains both x_p and y_1. If there does not exist such j, then the only vertex covers are of the form X or Y.

From the above lemmas and Theorem 5, we can conclude the following theorem and Algorithm 1.

Algorithm 1: An algorithm to compute the $evc(G)$ for connected chain graphs

Input: A connected chain graph $G = (X \cup Y, E)$, where $|X| \leq |Y|$.
Output: $evc(G)$.
Compute $mvc(G)$;
if *G is isomorphic to some complete bipartite graph* $K_{f,g}$ **then**
    if $f = g$ **then**
        $evc(G) = mvc(G)$;
    else
        $evc(G) = mvc(G) + 1$;
else
    if *There exists a minimum vertex cover that contains both* y_1 & x_p **then**
        $evc(G) = mvc(G) + 1$;
    else
        if *G does not has any pendant vertex or only Y has pendant vertex* **then**
            if $|Y| = |X| + 1$ **then**
                $evc(G) = mvc(G)$;
            else if $|Y| > |X| + 1$ **then**
                $k = min\{i | \{x_i, y_q\} \in E\}$;
                if $|\cup_{j=1}^{k-1} N(x_j)| = k$ **then**
                    $evc(G) = mvc(G) + 1$;
                else
                    $evc(G) = mvc(G) + 2$;
            else
                $evc(G) = mvc(G)$;
        else
            $evc(G) = mvc(G) + 1$;
return $evc(G)$;

Theorem 6 *Given a connected chain graph* $G = (V, E)$, $evc(G)$ *can be computed in* $O(n + m)$ *time.*

4 A Linear Time Algorithm for Split Graphs

In this section, we present a linear-time algorithm to compute the eternal vertex cover number for split graphs. Note that, there already exists a quadratic time algorithm to compute $evc(G)$ for chordal graphs. Since the class of split graphs is a subclass of chordal graphs, we also have a quadratic time algorithm to compute $evc(G)$ for split graphs. But, in this section, we present a linear-time algorithm to compute $evc(G)$ for any split graph G.

The following result is already known regarding the eternal vertex cover number of chordal graphs.

Theorem 7 *[4] Given a connected chordal graph $G = (V, E)$ and the set of all cut vertices X of G, $evc(G) = mvc_X(G)$ if and only if for every vertex $v \in V(G) \setminus X$, we have $mvc_{X\cup\{v\}}(G) = mvc_X(G)$; otherwise $evc(G) = mvc_X(G) + 1$.*

Since split graphs are chordal graphs, for any split graph G we have $evc(G) \in \{mvc_X(G), mvc_X(G) + 1\}$.

Let $G = (K \cup I, E)$ be a connected split graph, where K is a clique and I is an independent set. Without loss of generality, we may assume that K is a maximal clique of G. Let X denote the set of cut vertices of G. Now, we first prove the following lemmas.

Lemma 8 *If for each $x \in K$, $|N(x)| > |K| - 1$, then $mvc(G) = mvc_X(G) = |K|$. Otherwise $mvc(G) = mvc_X(G) = |K| - 1$.*

Proof: If for each $x \in K$, $|N(x)| > |K| - 1$, then each $x \in K$ has at least one neighbour in I. Note that any minimum vertex cover must contain at least $|K| - 1$ vertices from K. If there exists a minimum vertex cover S that contains only $|K| - 1$ vertices from K, then there exists a vertex $v \in K$, which does not belong to S. So, S must contain all neighbours of v from I, implying that $|S| \geq |K|$. Since K is itself a vertex cover of size $|K|$, if v has more than one neighbour in I, then $|S| > |K|$, a contradiction. So, K always form a minimum vertex cover in this case. Since $X \subseteq K$, it can be concluded that $mvc(G) = mvc_X(G) = |K|$.

Now if there exists $x \in K$, such that $|N(x)| = |K| - 1$, then $K \setminus \{x\}$ forms a minimum vertex cover of cardinality $|K| - 1$. Note that x cannot be a cut vertex (as it has no neighbour in I). So, $X \subseteq K \setminus \{x\}$ and $K \setminus \{x\}$ forms a minimum vertex cover, implying that $mvc(G) = mvc_X(G) = |K| - 1$. □

Lemma 9 *$evc(G) \in \{mvc(G), mvc(G) + 1\}$.*

Proof: The proof follows from the fact that $evc(G) \in \{mvc_X(G), mvc_X(G) + 1\}$ and $mvc(G) = mvc_X(G)$. □

Lemma 10 *Let $mvc(G) = |K| - 1$. Then $evc(G) = mvc(G) + 1$ if $I \neq \emptyset$ and $evc(G) = mvc(G)$ if $I = \emptyset$.*

Proof: If $I \neq \emptyset$, then consider a vertex $y \in I$. By Theorem 4, if $evc(G) = mvc(G) = |K| - 1$, then there exists a minimum vertex cover S that contains y, which implies $|S \cap K| \leq |K| - 2$, leading to a contradiction. Hence $evc(G) = mvc(G) + 1$. If $I = \emptyset$, then G is a complete graph, implying $evc(G) = mvc(G)$. □

Lemma 11 *Let $mvc(G) = |K|$ and there exists at least one pendant vertex $y_i \in I$, then $evc(G) = mvc(G) + 1$.*

Algorithm 2: An algorithm to compute the $evc(G)$ for connected split graphs

```
Input: A connected split graph G = (K ∪ I, E), K(≠ ∅) and I(≠ ∅) are clique and
  independent set respectively.
Output: evc(G).
Find mvc(G) in linear time for G;
if mvc(G) = |K| − 1 then
    evc(G) = mvc(G) + 1;
    return evc(G);
else
    if there exists at least one pendant vertex in G then
        evc(G) = mvc(G) + 1;
        return evc(G);
    else
        if deg(x) ≥ |K| + 1, for all x ∈ K then
            evc(G) = mvc(G) + 1;
            return evc(G);
        Derive X1; X1= Set of all vertices in K with degree |K| + 1

        if N_I(X1) = I then
            evc(G) = mvc(G);
        else
            evc(G) = mvc(G) + 1;
return evc(G);
```

Proof: Let x_j be the only neighbour of the pendant vertex y_i, then $x_j \in X$. On contrary assume that $evc(G) = mvc(G)$, then by Theorem 7 there exists a minimum vertex cover S that contains both X and y_i. Hence $x_j \in S$ as $y_i \in S$. Then there must be a vertex $x_k \in K$, which does not belong to S. Since $mvc(G) = |K|$, by Lemma 8, $N(x) \cap I \neq \emptyset$ for all $x \in K$. Hence, if x_k is not in S then all of its neighbours should be in S. Since y_i is not a neighbour of x_k, no neighbour of x_k in I belongs to S. Hence contradiction arises. So, $evc(G) = mvc(G) + 1$. □

Lemma 12 *Let $mvc(G) = |K|$, G has no pendant vertices and for each $x \in K$, $deg(x) \geq |K| + 1$. Then, $evc(G) = mvc(G) + 1$.*

Proof: Note that $mvc_X(G) = mvc(G)$. On contrary assume that $evc(G) = mvc(G)$. Then, by Theorem 4, for any $y_i \in I$, there exists a minimum vertex cover S that contains y_i. Then $|K \cap S| = |K| - 1$. Let $x_j \in K$ be the vertex which is not in S. Since $|N_I(x_j)| \geq 2$, S contains at least 2 vertices from I. But, then $|S| \geq |K|+1$, a contradiction arises. Hence, $evc(G) = mvc(G)+1$. □

Lemma 13 *Let G does not has any pendant vertex with $mvc(G) = |K|$ and $X_1 = \{x \in K : deg_I(x) = 1\}$. If $N(X_1) \cap I = I$, then $evc(G) = mvc(G)$, otherwise if $N(X_1) \cap I$ is properly contained in I, then $evc(G) = mvc(G) + 1$.*

Proof: Since G is a connected split graph with no pendant vertex, then any minimum vertex cover is connected. So, by Theorem 1, if for each $v \in V$, there is a minimum vertex cover S_v that contains v, then $evc(G) = mvc(G)$. Now K is a minimum vertex cover; so, there exists a minimum vertex cover which contains all the vertices of K. Since $N(X_1) \cap I = I$, then for each $y_i \in I$, there exists $x_j \in X_1$, such that $\{x_j, y_i\} \in E$. Which implies $S = \{y_i\} \cup (X \setminus \{x_j\})$ is a minimum vertex cover. So, $evc(G) = mvc(G)$.

If $N(X_1) \cap I \subset I$, then there exists $y_i \in I$ such that there does not exist any $x \in X_1$ so that $\{x, y_i\} \in E$. Let there exists a minimum vertex cover S' that contains y_i. So, there exists $x_j \in K$ such that $x_j \notin S'$. Take any $x_l \in N(y_i)$; $deg_I(x_l) \geq 2$, which implies $x_l \in S'$. If $deg_I(x_j) \geq 2$, then $|S' \cap I| \geq 2$, contradiction arises. So, $deg_I(x_j) = 1$, and let $N_I(x_j) = \{y_k\}$ where $(k \neq i)$. Clearly $y_k \notin S'$ as $S' \cap I = \{y_i\}$. Which implies $x_j \in S'$, a contradiction. So there does not exists any minimum vertex cover that contains y_i. So by Theorem 4, $evc(G) = mvc(G) + 1$. □

Theorem 8 *For a connected split graph $G(K \cup I, E)$, $evc(G)$ can be computed in time $O(n + m)$.*

Proof: The proof of the theorem is straightforward from the above lemmas and Algorithm 2. Before starting the algorithm, by preprocessing, an array $A[1, 2, \ldots, n]$ can be formed, such that $A[i]$ stores the degree of the vertex v_i. With the help of this array, the algorithm can run in $O(n + m)$ time. □

5 Algorithm for P_4-sparse Graphs

In [13], we propose a polynomial time algorithm to compute $evc(G)$ for cographs. In this section, we present a polynomial-time algorithm to solve the eternal vertex cover problem for P_4-sparse graphs. The class of P_4-sparse graphs is a superclass of the class of cographs. Below, we present a characterization theorem for P_4-sparse graphs.

Theorem 9 *[6] A graph G is said to be P_4-sparse if and only if one of the following conditions hold*

- *G is a single vertex graph.*
- *$G = G_1 \cup G_2$, where G_1 and G_2 are P_4-sparse graphs.*
- *$G = G_1 \oplus G_2$, where G_1 and G_2 are P_4-sparse graphs.*
- *G is a spider which admits a spider partition (S, C, R) where either $G[R]$ is a P_4-sparse graph or $R = \phi$.*

Hence, by Theorem 9, a connected graph that is P_4-sparse and contains at least two vertices can be classified as either a join of two P_4-sparse graphs or a particular type of spider (thick or thin). Note that if G is a spider such that $|R| \leq 1$; then G is a split graph, and $evc(G)$ can be computed in linear time by Algorithm 2.

Now let us consider the case when $G = (S, C, R)$ is a spider with $|R| \geq 2$.

Lemma 14 *Let $G = (V, E)$ be a thin spider with the property: $|R| \geq 2$ in the spider partition (S, C, R); then $evc(G) = mvc(G) + 1$.*

Proof: Let D be a minimum vertex cover of G. Then, it is easy to observe that $D = C \cup D'$, where D' is a minimum vertex cover of $G[R]$. Note that $G[D]$ is also connected. This implies that D is a minimum connected vertex cover as well. Hence $evc(G) \leq |D| + 1$.

Let s_i be a vertex in S. For the sake of contradiction, let $evc(G) = |D|$. Hence, there exists a minimum vertex cover D of G that contains s_i, which is a contradiction since all the minimum vertex covers of G are of the form $D = C \cup D'$, where D' is a minimum vertex cover of $G[R]$. Hence, $evc(G) = mvc(G) + 1$. □

Lemma 15 *Let $G = (V, E)$ be a thick spider with the property: $|R| \geq 2$ in the spider partition (S, C, R); then $evc(G) = mvc(G) + 1$.*

Proof: The proof is exactly similar to the proof of previous lemma. □

Now the case that is yet to be seen is when G is a join of two P_4-sparse graphs G_1 and G_2.

By Theorem 1, given a connected P_4-sparse graph $G = (V, E)$, for which each minimum vertex cover is connected, $evc(G)$ can be calculated by checking $mvc_v(G) = mvc(G)$ for each $v \in V$. To check this condition for any $v \in V$, a new graph $G' = (V', E')$ can be formed from G, where $V' = V \cup \{u\}$, $E' = E \cup \{uv\}$; then we can check whether $mvc(G) = mvc(G')$. The class of P_4-sparse graphs is not closed under pendant vertex addition. But P_4-sparse graphs are also weakly chordal graphs, which are closed under pendant vertex addition. So, we are proposing a polynomial time algorithm $EVC\_CHECK(G)$ for connected P_4-sparse graph G, for which every minimum vertex cover is connected, to compute $evc(G)$. We are using the algorithm given in [8] to compute minimum vertex cover for weakly chordal graphs.

Algorithm 3: EVC_CHECK(G)

Input : A connected P_4-sparse graph $G = (V, E)$, for which every minimum vertex cover is connected.
Output : $evc(G)$.

```
Compute mvc(G);
count = 0;
for each u ∈ V do
    Add a pendant vertex v to u in G, the new graph is G_v;
    Compute mvc(G_v) from the algorithm in [8];
    if mvc(G) = mvc(G_v) then
        count = count + 1;
if count = |V| then
    evc(G) = mvc(G);
else
    evc(G) = mvc(G) + 1;
return evc(G);
```

Note that to calculate $mvc(G)$ in a weakly chordal graph G, the time complexity of the algorithm mentioned in [8], is $O(nm)$. Hence the time complexity of $EVC\_CHECK(G)$ is $O(n^2m)$.

Now let us consider the graph G is join of two P_4-sparse graphs G_1 and G_2. Here we are assuming $|V(G_1)| \leq |V(G_2)|$ and both $V(G_1)$ and $V(G_2)$ are non-empty. If $mis(G) > min\{|V(G_1)|, |V(G_2)|\}$, then a maximum independent set I of G is a subset of G_2. If $I \subset G_2$, then every minimum vertex cover D is connected, since $G_2 \cap D \neq \emptyset$ and $V(G_1) \subset D$. So $evc(G)$ can be computed by $EVC\_CHECK(G)$. If $I = V(G_2)$, then $V(G_1)$ is the only minimum vertex cover and there does not exist any minimum vertex cover D that contains any vertex of G_2, so by Theorem 4, $evc(G) \neq mvc(G)$. In this case, $V(G_1) \cup \{u\}$, such that $u \in V(G_2)$, forms an initial configuration of guards, as G_2 is independent, implying $evc(G) = mvc(G) + 1$.

So, the case that remains to be observed is, when $mis(G) \leq min\{|G_1|, |G_2|\}$. If $mis(G) < min\{|G_1|, |G_2|\}$, then any minimum vertex cover D is connected, since $G_2 \cap D \neq \emptyset$ and $G_1 \cap D \neq \emptyset$. So, $evc(G)$ can be computed by $EVC\_CHECK(G)$.

Now for the case when $mis(G) = min\{|V(G_1)|, |V(G_2)|\}$. If $|G_1| = |G_2|$, then at least one among the sets $V(G_1)$ and $V(G_2)$ is an independent set.

If both are independent then G is $K_{|V(G_1)|,|V(G_1)|}$, and $evc(G) = mvc(G)$.

If G_2 is not independent, then G_1 is independent. This implies $V(G_2)$ is the only minimum vertex cover of G. Hence, no minimum vertex cover contains any vertex from G_1. So, by Theorem 4, $evc(G) \neq mvc(G)$. So, $evc(G) = mvc(G) + 1$ and $G_2 \cup \{u\}$ where $u \in G_1$, forms an initial guard allocation configuration.

Lemma 16 *Given a connected P_4-sparse graph $G = (V, E)$ which is a join of two P_4-sparse graphs G_1 and G_2. If $mis(G) = min\{|V(G_1)|, |V(G_2)|\}$ and $|V(G_1)| < |V(G_2)|$, then $evc(G)$ can be calculated in polynomial time.*

Proof: Since $|V(G_1)| < |V(G_2)|$, $V(G_2)$ is not an independent set. If $V(G_1)$ is not an independent set, then $mis(G_2) = mis(G)$. So, each minimum vertex cover is connected. So, $evc(G) = mvc(G)$ or $evc(G) = mvc(G) + 1$ can be decided in polynomial time by $EVC\_CHECK(G)$. Now if G_1 is independent, then two cases may arise.

Case 1: $mis(G_2) = |V(G_1)|$. Let I_2 be a maximum independent set of G_2. So, we have at least 2 minimum vertex covers; $V(G_2)$ and $V(G_1) \cup (V(G_2) \setminus I_2)$. If at some point of time, the guards are allocated at G_2 then consider any attack on $uv \in E(G)$; where $u \in V(G_1)$ and $v \in V(G_2)$. If $v \in I_2$ then move all guards of I_2 to $V(G_1)$ and reach the configuration $V(G_1) \cup (V(G_2) \setminus I_2)$. If $v \in V(G_2) \setminus I_2$, then the guard at v moves to u. v has at least one neighbour in I_2(since I_2 is a maximum independent set of G_2), say w. The guard at w moves to v and all the guards at $I_2 \setminus \{w\}$ moves to $V(G_1) \setminus \{u\}$ to reach the configuration $V(G_1) \cup (V(G_2) \setminus I_2)$. So, when the guards are allocated at configuration $V(G_2)$, any attack(on edges for which at least one end point does not has any guard) can be defended by changing the configuration to $V(G_1) \cup (V(G_2) \setminus I_2)$.

If the guards are allocated in $V(G_1) \cup (V(G_2) \setminus I_2)$. Attack an edge $uv \in E(G)$. If $u \in I_2$ and $v \in V(G_2) \setminus I_2$, then move the guard at v to u and all the guards at $V(G_1)$ to $(I_2 \setminus \{u\}) \cup \{v\}$ reaching the new configuration G_2. Now if $u \in I_2$ and $v \in V(G_1)$, then move all guards at $v(G_1)$ to I_2 to reach the configuration $V(G_2)$. So, any attack can be defended by placing $mvc(G)$ number of guards in configuration $V(G_2)$ or $V(G_1) \cup (G_2 \setminus I_2)$. So, $evc(G) = mvc(G)$.

Case 2: $mis(G_2) < |V(G_1)|$. Then $V(G_2)$ is the only possible minimum vertex cover of G. So, any attack on edge $uv \in E(G)$, where $u \in G_1$ and $v \in V(G_2)$ can not be defended because there does not exist any minimum vertex cover which contains any vertex from G_1, implying $evc(G) \neq mvc(G)$, by Theorem 4. So, $evc(G) = mvc(G) + 1$ and $V(G_2) \cup \{u\}$, such that $u \in V(G_1)$, forms an initial configuration of guard allocation. □

Hence, by the above discussion, we can conclude the following theorem.

Theorem 10 *Given a connected P_4-sparse graph G, $evc(G)$ can be computed in time $O(n^2m)$.*

6 Conclusion and Future Aspects

In this paper, we have given polynomial time algorithms for three restricted subclasses of perfect graphs, i.e., chain graphs, split graphs, and P_4-sparse graphs. For split graphs, the running time of our algorithm is linear. The class of split graphs is an important subclass of chordal graphs, for which a quadratic time algorithm was already known in the literature. It will also be interesting to look for linear-time algorithms for eternal vertex cover problem for chordal graphs, or some other important subclasses of chordal graphs. The eternal vertex cover problem is NP-hard for bipartite graphs and the class of chain graphs is the largest class of bipartite graphs for which linear time algorithm has been found. The complexity status of the eternal vertex cover problem is still unknown for other important subclasses of bipartite graphs. Here, we have solved the complexity status of the eternal vertex cover problem for P_4-sparse graphs. It will be interesting to check the complexity status of the problem for other well-known graph classes like distance-hereditary graphs and AT-free graphs.

References

[1] H. Araki, T. Fujito, and S. Inoue. On the eternal vertex cover numbers of generalized trees. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, 98-A(6):1153–1160, 2015. doi:10.1587/transfun.E98.A.1153.

[2] J. Babu, L. S. Chandran, M. C. Francis, V. Prabhakaran, D. Rajendraprasad, and J. N. Warrier. On graphs whose eternal vertex cover number and vertex cover number coincide. *Discret. Appl. Math.*, 319:171–182, 2022. doi:10.1016/j.dam.2021.02.004.

[3] J. Babu, N. Misra, and S. G. Nanoti. Eternal vertex cover on bipartite graphs. In *Computer Science – Theory and Applications: 17th International Computer Science Symposium in Russia, CSR 2022, Virtual Event, June 29 – July 1, 2022, Proceedings*, page 64–76, Berlin, Heidelberg, 2022. Springer-Verlag. doi:10.1007/978-3-031-09574-0_5.

[4] J. Babu and V. Prabhakaran. A new lower bound for the eternal vertex cover number of graphs. *Journal of Combinatorial Optimization*, 06 2021. doi:10.1007/s10878-021-00764-8.

[5] J. Babu, V. Prabhakaran, and A. Sharma. A substructure based lower bound for eternal vertex cover number. *Theor. Comput. Sci.*, 890:87–104, 2021. doi:10.1016/j.tcs.2021.08.018.

[6] G. Bagan, H. B. Merouane, M. Haddad, and H. Kheddouci. On some domination colorings of graphs. *Discret. Appl. Math.*, 230:34–50, 2017. URL: https://doi.org/10.1016/j.dam.2017.06.013, doi:10.1016/J.DAM.2017.06.013.

[7] F. V. Fomin, S. Gaspers, P. A. Golovach, D. Kratsch, and S. Saurabh. Parameterized algorithm for eternal vertex cover. *Inf. Process. Lett.*, 110(16):702–706, 2010. doi:10.1016/j.ipl.2010.05.029.

[8] R. Hayward, J. P. Spinrad, and R. Sritharan. Weakly chordal graph algorithms via handles. In *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, January 9-11, 2000, San Francisco, CA, USA*, pages 42–49. ACM/SIAM, 2000. URL: http://dl.acm.org/citation.cfm?id=338219.338231, doi:10.1145/338219.338231.

[9] B. Jamison and S. Olariu. Recognizing p_4 sparse graphs in linear time. *SIAM J. Comput.*, 21(2):381–406, 1992. doi:10.1137/0221027.

[10] W. Klostermeyer and G. MacGillivray. Eternal security in graphs of fixed independence number. *JCMCC. The Journal of Combinatorial Mathematics and Combinatorial Computing*, 63, 11 2007.

[11] W. F. Klostermeyer and C. M. Mynhardt. Edge protection in graphs. *Australas. J Comb.*, 45:235–250, 2009. URL: http://ajc.maths.uq.edu.au/pdf/45/ajc_v45_p235.pdf.

[12] R. M. McConnell and J. P. Spinrad. Modular decomposition and transitive orientation. *Discret. Math.*, 201(1-3):189–241, 1999. doi:10.1016/S0012-365X(98)00319-7.

[13] K. Paul and A. Pandey. Some algorithmic results for eternal vertex cover problem in graphs. In C. Lin, B. M. T. Lin, and G. Liotta, editors, *WALCOM: Algorithms and Computation - 17th International Conference and Workshops, WALCOM 2023, Hsinchu, Taiwan, March 22-24, 2023, Proceedings*, volume 13973 of *Lecture Notes in Computer Science*, pages 242–253. Springer, 2023. doi:10.1007/978-3-031-27051-2_21.