Complexity of Eternal Security

William F. Klostermeyer
Dept. of Computer and Information Sciences
University of North Florida
Jacksonville, FL 32224-2669
E-mail: wkloster@unf.edu

Abstract

We show that deciding if a set of vertices is an eternal 1-secure set is complete for co$-NP^{NP}$, solving a problem stated by Goddard, Hedetniemi, and Hedetniemi [*JCMCC*, vol. 52, pp. 160-180].

1 Introduction

Let $G = (V, E)$ be a simple graph. Let $N(v)$ denote the open neighborhood of vertex v and $N[v] = N(v) \cup \{v\}$ the closed neighborhood of v. A dominating set of G is a set $D \subseteq V$ such that, for all v, $N[v] \cap D \neq \emptyset$.

Considerable recent interest has been given to problems concerned with protecting the vertices in a graph from a series of one or more attacks, see for example [1, 2, 5]. In this scenario, guards are located at vertices and can protect the vertices at which they are located and can move to a neighboring vertex to defend an attack there. Under this simple set of rules, a dominating set suffices to defend a graph against a single attack. Several variations of this problem have been proposed including Roman Domination [3], Weak Roman Domination [4] and k-secure sets/eternal secure sets [1, 2, 5].

Let R denote a sequence of vertices. If sequence R is of length k, we will sometimes denote it as R_k. Let $R(i)$ denote the i^{th} vertex in R, with $R(1)$ being the first vertex in R. R is interpreted as the location of a sequence of attacks at vertices, each of which must be defended by a guard. At most one guard is allowed to move to defend each attack.

Let D_0 be the initial location of the guards and D_i the location of the guards after $R(i)$ is defended. We allow at most one guard per vertex. It must be that $R(i) \in D_i$. Then if $R(i) \notin D_{i-1}$, $D_i = D_{i-1} \setminus \{v\} \cup R(i)$, where $v \in D_{i-1}$ and $R(i) \in N(v)$, and we say a guard *moved* to $R(i)$.

Note that in order to defend any first attack, D_0 must be a dominating set. We are now ready to define two types of guard sets.

(1) D is a *k-secure dominating set* if, for all R_k, there exists a sequence $D_0, D_1, \ldots, D_k$ such that $D_i = D_{i-1} \setminus \{v\} \cup R(i)$ (possibly $v = R(i)$), $R(i) \in N[v]$, and each D_i is a dominating set. The size of a smallest k-secure set in G is denoted $\gamma_k(G)$ [1].

(2) D is an *eternal secure set* if, for all R, there exists a sequence $D_0, D_1, \ldots,$ such that $D_i = D_{i-1} \setminus \{v\} \cup R(i)$ (possibly $v = R(i)$), $R(i) \in N[v]$, and each D_i is a dominating set. This is also called an eternal 1-secure set [5]. The size of a smallest eternal set in G is denoted $\gamma_\infty(G)$ [2].

Goddard et al. conjectured that deciding if a set of vertices if an eternal secure set lies in the first few levels of the polynomial time hierarchy [5]. In this paper, we resolve the complexity of this problem and consider the complexity of two related problems.

2 NP-completeness

The following result shows that the problem of deciding which guards to move to defend a sequence of attacks is difficult, even if one knows the sequence of attacks in advance.

Theorem 1 *It is NP-complete to decide if, given a sequence R_k as part of the input, a set of vertices D is a k-secure set in graph $G = (V, E)$.*

Proof: The problem is obviously in NP, as one can guess a guard to move to $R(i)$ for each i and then verify whether or not each D_i is a dominating set.

We now show it is NP-hard via a reduction from 3-SAT. Let F be an instance of 3-SAT with clause set $C = c_1, c_2, \ldots, c_r$ and variable set $U = u_1, u_2, \ldots u_q$. Denote the three literals (i.e., variables or negations of variables) in clause c_i as u_1^i, u_2^i, u_3^i. Assume without loss of generality that no clause contains both a variable and its negation. Variable u is said to be the *underling variable* of literals u and $\neg u$.

Construct a graph as follows. For each clause c_i create a *clause vertex* c_i. For each of the three literals in clause c_i, create *literal vertices* u_1^i, u_2^i, u_3^i corresponding to the literals in clause c_i and connect each to c_i. Then for each pair of literal vertices u_k^i, u_p^j such that u_k^i corresponds to the negation of u_p^j, create a vertex $w_{i,j}^{k,p}$ (called a w-type vertex) and add edges $u_k^i w_{i,j}^{k,p}$, $u_p^j w_{i,j}^{k,p}$. Note that we create exactly one w-type vertex for each pair of complementary literals. The w-type vertices will serve to ensure a consistent truth assignment.

Now consider the sequence of vertices $R = c_1, c_2, \ldots, c_r$ and the set of vertices $D = \{u_i^j | 1 \leq i \leq 3, 1 \leq j \leq r\}$. We claim that D is an eternal secure set for G and R if and only if F is satisfiable. Note that the graph, D, and R can be constructed in polynomial time. We interpret the vertices in D as guards stationed at those vertices.

First suppose that F is satisfiable. Then each clause c_i has at least one "witness," which is a literal that evaluates to "true" (i.e., a non-negated literal whose underlying variable is assigned the value "true" or vice versa). Move a guard stationed at one such true literal vertex to the clause vertex for that clause. Do this for all clauses in the order specified by R. Note that the resulting configuration of guards is a dominating set, since the truth assignment is consistent, and thus each $w_{i,j}^{k,p}$ is dominated by the resulting configuration of guards.

Next suppose that D is a k-secure set for G and R. We show F is satisfiable. To satisfy a request c_i in R, a guard at one literal vertex must be moved to c_i. This indicates which literal satisfies clause c_i. To see that this truth assignment is consistent, observe that if a variable and its negation are witnesses for distinct clauses, then some $w_{i,j}^{k,p}$ vertex will not be dominated by the final configuration of guards. □

The graph constructed by this reduction can be seen to be bipartite.

3 Eternal Security

The *clique cover number* of G, $\theta(G)$, is the minimum number of cliques required to cover the vertices of G (which is exactly the chromatic number of the complement of G). It is easy to prove that $\gamma_\infty(G) \leq \theta(G)$ [2, 5]. Goddard et al. demonstrate some graphs for which $\gamma_\infty(G) + c < \theta(G)$ for any constant c [5]. Let $\beta(G)$ denote the independence number of G. It is easy to see that $\gamma_\infty(G) \geq \beta(G)$ for all G [5].

In [5] the complexity of deciding if a set is an eternal secure set was left open. We now solve this problem.

Theorem 2 *Deciding if a set D is an eternal set is in co-NP^{NP}.*

Proof: Consider the complementary problem of deciding if a set D is not an eternal set for G. We show this problem is in NP^{NP}. Assume without loss of generality that G is connected, else consider each component separately.

Let D_1, D_2 be subsets of V of equal cardinality. Define distance(D_1, D_2) as the minimum number of moves required to change D_1 into D_2, where a move replaces a vertex v in D_1 with a neighbor of v. Obviously, for all D_1, D_2 in G, distance$(D_1, D_2) < |V|^2$.

Observe that if D is not an eternal set for G, there exists a finite sequence R_k such that D_k (recall that D_k is the configuration of guard after defending the sequence of attacks in R_k) is not a dominating set. In fact, since the distance between any two sets is less than $|V|^2$, there exists such a sequence having length less than $|V|^2$. So if such a sequence exists, a non-deterministic polynomial time Turing machine can guess it.

Given such a sequence R_k, for each vertex $R(i)$ in the sequence, there may be more than one guard that is adjacent to that vertex (and thus there can be more than one possible way to move a guard to $R(i)$). Hence for the sequence R_k, there may be (exponentially) many ways of defending the sequence of attacks. Using an oracle for NP, we can decide if none of those ways is such that D_i is a dominating set for all i, $0 \leq i \leq k$. Hence the result. □

Note that the same complexity applies the problem of testing of a set is a k-secure set, if k is sufficiently large.

Theorem 3 *Deciding if a set D is an eternal set is hard for co-NP^{NP}.*

Proof: Let F^* be a quantified 3-SAT instance of the form

$$\forall u_1, u_2, \ldots, u_m \exists u_{m+1}, u_{m+2}, \ldots u_q \; F$$

where F is a 3-SAT instance with clause set $C = c_1, c_2, \ldots, c_r$ and variable set $U = u_1, u_2, \ldots u_q$. Assume without loss of generality that no clause contains both a variable and its negation.

We will start with the same reduction used Theorem 1 and make the following additions. First, connect any $w_{i,j}^{k,p}$ vertices that correspond to the same underlying variable together into a clique. For each universally quantified variable u_i, create two new literal vertices $u_i, \neg u_i$ and three other new vertices b_i, d_i, e_i and edges $u_i b_i, b_i \neg u_i, \neg u_i d_i, d_i e_i, u_i e_i, u_i \neg u_i$. Call the subgraph induced by these five vertices a *quantifier gadget.* Create additional w-type vertices for each pair of literal vertices $u_i(\neg u_i), u_k^j$ that correspond to negations of one another, as before (but no w-type vertex is created for the specific pair $u_i, \neg u_i$). Of course, these additional w-type vertices will become part of the various cliques of w-type vertices.

One can observe that $\beta(G) = 3r + 2m$ (take all the u_i^j, b_i, e_i vertices, for example). Let D be the set of literal vertices, keeping in mind that D contains the u_i^j vertices and the subset $\{u_i, \neg u_i : 1 \leq i \leq m\}$. Therefore $|D| = 3r + 2m$. We claim D is an eternal set if and only if F^* is satisfiable. Suppose D is an eternal set. Then the following 2^m different sequences of attacks can be defended with the final configuration of guards being a dominating set:

$$d_1(e_1), d_2(e_2), \ldots, d_m(e_m), c_1, c_2, \ldots, c_r$$

By the notation $d_i(e_i)$ in this sequence, we mean that you can choose either of those two vertices to be attacked. In terms of F^*, choosing e_i to be attacked means that we are forcing the variable u_i to be true (since this will force some literal vertex corresponding to the negation of u_i, to defend a w-type vertex that is a neighbor of $\neg u_i$, and this literal will not be able to serve as a witness for any clause); choosing d_i to be attacked means we are forcing the variable u_i to be false. If each such sequence can be defended with the ending configuration of guards being a dominating set, then F^* is satisfiable, using the same logic as in Theorem 1.

Now suppose F^* is satisfiable; we must show D is an eternal set. By the same logic used before, we can defend any sequence of attacks of the form

$$d_1(e_1), d_2(e_2), \ldots, d_m(e_m), c_1, c_2, \ldots, c_r$$

But of course there are other sequences of attacks to consider. The subgraph induced by a quantifier gadget has clique cover number equal to two and each clique of w-type vertices has clique cover number equal to one. So if F^* is satisfiable, any sequence of attacks can be defended by the following strategy: (1) each c_i vertex is defended by a guard at the witness literal vertex for the clause (as in Theorem 1 and as above), (2) any w-type vertex is defended by a guard at a neighboring literal vertex that either "false" (i.e., a non-negated literal whose underlying variable is assigned the value "false" or vice versa) or is not the witness chosen to defend that particular clause vertex (note that at some point in time, it is possible that this guard is residing at another w-type vertex within the same clique) (3) each quantifier gadget is defended by the two guards that initially reside within the gadget (4) any attack at an unoccupied literal vertex can be defended by a guard at a neighboring w-type vertex. Hence the proof. □

Of course, the graphs constructed by this reduction are, in general, not bipartite; thus the following partial result regarding bipartite graphs.

Fact 4 *Deciding if a set D is an eternal secure set for a bipartite graph G is in P^{NP}.*

Proof: We show that D is an eternal secure set if and only if $|D| \geq \beta(G)$ and D is a dominating set. This implies the result, as one must be able to compute the exact value of $\beta(G)$ (which is in P^{NP}, is NP-hard, but not necessarily in NP.) One direction is trivial. For the other, suppose $|D| \geq \beta(G)$ and D is a dominating set. Then D is a clique cover and thus forms an eternal secure set. □

We leave open the question of determining the exact complexity of the last problem.

References

[1] A. Burger, E. Cockayne, W. Gründlingh, C. Mynhardt, J. van Vuuren, and W. Winterbach (2004), Finite Order Domination in Graphs (2004), *J. Comb. Math. Comb. Comput.*, vol. 49, pp. 159-176

[2] A. Burger, E. Cockayne, W. Gründlingh, C. Mynhardt, J. van Vuuren, and W. Winterbach (2004), Infinite Order Domination in Graphs (2004), *J. Comb. Math. Comb. Comput.*, vol. 50, pp. 179-194

[3] E. Cockayne, P. Dreyer, S. M. Hedetniemi, and S. T. Hedetniemi (2004), Roman domination in graphs *Discrete Math.*, vol. 278, pp. 11-22

[4] E. Cockayne, O. Favaron, and C. Mynhardt (2003), Secure domination, weak roman domination and forbidden subgraphs, *Bull. Inst. Combin. Appl.*, vol. 39, pp.

[5] W. Goddard, S. M. Hedetnieni, and S. T. Hedetniemi (2005), Eternal Security in Graphs, *J. Comb. Math. Comb. Comput.*, vol. 52, pp. 160-180