Complexity Results for Equistable Graphs and Related Classes

Martin Milanič [*][†] James Orlin[‡] Gábor Rudolf [§]

January 20, 2009

Abstract

We describe a simple pseudo-polynomial-time dynamic programming algorithm to solve the maximum weight stable set problem along with the weighted independent domination problem in some classes of graphs, including equistable graphs. These classes, not contained in any nontrivial hereditary class, are defined by the existence of a cost structure on the vertices where maximal stable sets are characterized by their costs. Our results are obtained within the wider context of Boolean optimization; corresponding hardness results are also provided.

1 Introduction

In this paper, we present an approach to solving the MAXIMUM WEIGHT STABLE SET PROBLEM, as well as the WEIGHTED INDEPENDENT DOMINATION PROBLEM in some graph classes for which these problems are NP-hard, including the well-known class of equistable graphs. A lot of recent work focuses on solving such problems on hereditary classes of graphs, typically using characterizations by forbidden induced subgraphs (see, for example, [1, 2, 5, 8, 12] and the references therein). In contrast, the graph classes in this paper, such as the class of equistable graphs, are not contained in any non-trivial hereditary class; therefore a different approach becomes necessary.

Our results are based on the more general framework of Boolean optimization. Let V be a finite set and $f : \mathcal{B}^V \to \mathcal{B}$ a Boolean function, where $\mathcal{B} = \{0, 1\}$. Denote the set of the *false points* of f by $\mathcal{F}(f) = \{\mathbf{x} \in \mathcal{B}^V : f(\mathbf{x}) = 0\}$. Now consider the following MAXIMUM WEIGHT FALSE POINT PROBLEM with objective coefficients (weights) $\mathbf{w} \in \mathbb{R}_+^V$:

$$\begin{array}{ll} \max & \mathbf{w}^\top \mathbf{x} \\ \text{s.t.} & \mathbf{x} \in \mathcal{F}(f)\,. \end{array} \tag{1}$$

[*]Universität Bielefeld, Technische Fakultät, 33501 Bielefeld, Germany. mmilanic@cebitec.uni-bielefeld.de

[†]This author gratefully acknowledges partial support by the group “Combinatorial Search Algorithms in Bioinformatics” funded by the Sofja Kovalevskaja Award 2004 of the Alexander von Humboldt Stiftung and the German Federal Ministry of Research and Education.

[‡]MIT Sloan School of Management, Cambridge, MA. 02139, USA. jorlin@mit.edu

[§]RUTCOR, Rutgers University, Piscataway, NJ. 08854, USA. grudolf@rutcor.rutgers.edu

The connection between Problem (1) and the MAXIMUM WEIGHT STABLE SET PROBLEM is provided by the following definition:

The *maximal stability function* $f : \mathcal{B}^V \to \mathcal{B}$ of a graph $G = (V, E)$ takes the value $f(\mathbf{x}) = 0$ if and only if $\mathbf{x}$ is the characteristic vector of a maximal stable set of G, and takes the value $f(\mathbf{x}) = 1$ otherwise. Notice that for such a function f, (1) becomes the well-known MAXIMUM WEIGHT STABLE SET PROBLEM for G.

Similarly to (1) one can define the MINIMUM WEIGHT FALSE POINT PROBLEM. When the function f is the maximal stability function of some graph, this problem becomes the WEIGHTED INDEPENDENT DOMINATION PROBLEM.

The key feature of our approach is to represent, when possible, the feasible set of (1) as the set of solutions where, given a cost function $\mathbf{c} : V \to \mathbb{N}$ on the variables, the total cost of variables taking value 1 lies in some set $T \subset \mathbb{R}_+$:

$$\mathcal{F}(f) = \{\mathbf{x} \in \mathcal{B}^V \ : \ \mathbf{c}^\top \mathbf{x} \in T\}. \tag{2}$$

In particular, we are interested in the following special cases:

Case 1. T consists of a single value: $T = \{t\}$.

Case 2. T is an interval: $T = [a, b]$.

Case 3. The set T is given by a membership oracle, along with an upper bound $M \in \mathbb{N}$ satisfying $T \subset [0, M]$.

Let us now recall the original definition of *equistable graphs* by Payan in 1980 [13]: A graph $G = (V, E)$ is called equistable if and only if there exists a positive integer t and a cost function $\mathbf{c} : V \to \mathbb{N}$ on the vertices of G such that a subset $S \subset V$ is a maximal stable set of G if and only if $\sum_{v \in S} c(v) = t$. In this case $\mathbf{c}$ is called an *equistable cost function*, while the pair $(\mathbf{c}, t)$ is called an *equistable cost structure*.

In the recent years, equistable graphs have been receiving an increasing amount of attention (see for example Chapter 14 in [9] and the papers [6, 7, 10, 14]). We remark that in the literature the costs $\mathbf{c}$ are usually called weights; in order to avoid confusion with the weights related to the MAXIMUM WEIGHT STABLE SET PROBLEM our paper does not follow this convention.

It is easy to observe that a graph is equistable if and only if its maximal stability function is of the type described in **Case 1** above. Similarly, one can consider the graph class corresponding to **Case 2**:

Definition 1. *A graph $G = (V, E)$ is called* interstable *if and only if there exists an interval $[a, b] \subset \mathbb{R}_+$ and a cost function $\mathbf{c} : V \to \mathbb{N}$ on the vertices of G such that a subset $S \subset V$ is a maximal stable set of G if and only if $\sum_{v \in S} c(v) \in [a, b]$. In this case $\mathbf{c}$ is called an* interstable cost function*, while the pair $(\mathbf{c}, [a, b])$ is called an* interstable cost structure.

Interstable graphs are a natural generalization of equistable graphs. These classes have many interesting structural properties of independent interest; for an overview including some recent results see [11].

We remark that allowing non-integer costs (i.e., considering cost functions of the form $\mathbf{c} : V \to \mathbb{R}_+$ instead of $\mathbf{c} : V \to \mathbb{N}$) does not change the set of representable functions and graphs. However, the complexity considerations in the remainder of this paper are only applicable to the integer case or to cases in which there is a specified common denominator Q.

The rest of the paper is structured as follows: we first introduce some necessary definitions and conventions. Then in Section 2, we provide hardness results for the problems under consideration and examine the relationship between equistable graphs and hereditary graph classes. In Section 3, we provide a pseudo-polynomial-time algorithm based on dynamic programming that solves (1) in a general setting, and examine the implications for the MAXIMUM WEIGHT STABLE SET PROBLEM in graphs. A variant of the method provides a solution to the WEIGHTED INDEPENDENT DOMINATION PROBLEM in the graph classes under consideration. In these results, we assume that the input graphs are given together with an equistable or interstable cost structure. Finally, Section 4 examines some recognition problems associated with equistable graphs.

Definitions and Notations

All graphs considered in this paper are finite, undirected, without loops or multiple edges. A class of graphs is *hereditary* if it is closed under deletion of vertices. For a graph G, we denote by $V(G)$ and $E(G)$ the vertex set and the edge set of G, respectively. As usual, P_n and K_n denote the chordless path and the complete graph on n vertices, respectively. The weight and cost of a subset $X \subseteq V$ are defined as $w(X) = \sum_{x\in X} w(x)$ and $c(X) = \sum_{x\in X} c(x)$, respectively. A *stable* (or *independent*) *set* in a graph is a set of pairwise non-adjacent vertices. The MAXIMUM STABLE SET PROBLEM is that of finding, in a given graph, a stable set of the maximum size. If each vertex of the graph is assigned a positive weight, the problem generalizes to the MAXIMUM WEIGHT STABLE SET PROBLEM, which asks for a stable set of the maximum total weight. A *dominating set* in a graph is a set $D \subseteq V(G)$ such that every vertex outside D is adjacent to some vertex in D. An *independent dominating set* is a set that is both independent and dominating. (Note that a set is an independent dominating set if and only if it is an (inclusion-wise) maximal stable set.) The WEIGHTED INDEPENDENT DOMINATION PROBLEM is that of finding, in a given vertex-weighted graph, an independent dominating set of minimum total weight.

We also use the following convention: for a function $v : V \to \mathbb{R}$ on a finite set V let $\mathbf{v}$ denote the corresponding vector with coordinates indexed by V.

2 Hardness Results

First, we observe that the MAXIMUM WEIGHT FALSE POINT PROBLEM is NP-hard as it generalizes the well-known *subset sum* problem [4], which asks whether, given positive integers $a_1, \ldots, a_n, b$, there is a subset $I \subseteq \{1, \ldots, n\}$ such that $\sum_{i\in I} a_i = b$.

Theorem 1. *The problem*

$$\begin{array}{ll} \max & \mathbf{w}^\top \mathbf{x} \\ \textit{s.t.} & \mathbf{x} \in \mathcal{B}^V\,,\ \mathbf{c}^\top \mathbf{x} \in T\,. \end{array}$$

is NP-hard, even when $T = \{t\}$ for some $t \in \mathbb{N}$.

In view of this negative result, it is natural to ask whether the problem becomes easier if the false points correspond to the maximal stable sets of a given graph. It turns out that this is not the case:

Theorem 2. *Finding a maximum weight stable set in an equistable graph is APX-hard, even if the graph is given together with an equistable cost structure. (This implies both inapproximability and strong NP-completeness.)*

Proof. We will carry out a transformation from the stable set problem in graphs, which is APX-hard.

Let $G = (V, E)$ be an undirected graph with vertices $\{1, 2, \dots, n\}$. We will create an equistable graph as follows.

Let $G' = (V', E')$ be a graph created as follows:

- $V' = \{v_1, \dots, v_n\} \cup \{w_1, \dots, w_n\} \cup \{u_e : e \in E\}$.
- For each $j = 1, \dots, n$, there is an edge $v_j w_j \in E'$.
- For each edge $e = ij \in E$, there are edges $v_i v_j$, $v_i u_e$, and $v_j u_e$ in E'.

Property 1. *A set $S \subseteq \{1, 2, \dots, n\}$ is a stable set in V (not necessarily maximal) if and only if the following set is a maximal stable set in V': $\{v_j : j \in S\} \cup \{w_j : j \notin S\} \cup \{u_{ij} : i \notin S, j \notin S\}$.*

By Property 1, there is a one-to-one correspondence between stable sets in V and maximal stable sets in V'.

We will next assign costs to each vertex of V' such that every maximal stable set of V' has the same cost t and every other subset of V' has a different cost.

Let $b_1, \dots, b_n$ be integers whose values will be assigned shortly.

Let $\{a_e : e \in E\}$ be a set of integers whose values will be assigned shortly.

The cost of vertex v_j is $b_j + 3\sum_{ij\in E} a_{ij}$. We refer to b_j as the V*-cost* of v_j, and we refer to $3\sum_{ij\in E} a_{ij}$ as the E*-cost* of v_j.

The cost of vertex w_j is $b_j + 2\sum_{ij\in E} a_{ij}$. We refer to b_j as the V*-cost* of w_j, and we refer to $2\sum_{ij\in E} a_{ij}$ as the E*-cost* of w_j.

The cost of vertex u_{ij} is a_{ij}, and we also call this value the E*-cost* of u_{ij}.

Finally, let $t = \sum_{i=1}^{n} b_i + 5\sum_{e\in E} a_e$.

Lemma 1. *Each maximal stable set in V' has cost t.*

Proof. Each maximal stable set S' has either vertex v_j or w_j, but not both. The sum of the V-costs of the vertices of S is thus $\sum_{i=1}^{n} b_i$. For each $ij \in E$, a maximal stable set S will contain exactly one of the following:

- v_i, w_j, or
- w_i, v_j, or
- w_i, w_j, u_{ij}.

All three stable sets contribute exactly $5a_{ij}$ to the E-cost of S. Thus the total E-cost of S' is $5\sum_{ij\in E} a_{ij}$, and the total cost of vertices of S' is t. □

We now assign values to the b's and a's.

There are $n+m$ different values we need to assign (where $m = |E|$). The first integer is 8, the second is 8^2, the third is 8^3, etc. After assigning these costs, the following lemma is true:

Lemma 2. *A subset in V' has cost t if and only if it is a maximal stable set.*

Proof. We only need to show that every subset with cost t must be a maximal stable set. We call a set $S \subseteq V(G')$ *vertex maximal* if for every j, S contains v_j or w_j but not both. We say that S is *edge maximal* if for every edge $ij \in E$, S contains (1) v_i and w_j or (2) w_i and v_j or (3) w_i, w_j and u_{ij}. A stable set S is a maximal stable set of G' if and only if it is vertex maximal and edge maximal.

For $i \in \{1,\dots,n\}$, let $b_i = 8^i$. Suppose a_j is the value associated with the j-th edge. Let $a_j = 8^{n+j}$. Suppose S has a cost of t. We will show that it is a maximal stable set. By considering values mod 8^{j+1}, one can show that S must contain vertex v_j or vertex w_j but not both; therefore, S is vertex maximal. Now consider edge $ij \in E$, and suppose it is the k-th edge. The contribution due to edge ij in any vertex maximal subset S is either $4a_{ij}, 5a_{ij}, 6a_{ij}$ or $7a_{ij}$. By considering values mod 8^{n+k+1}, one can show that the contribution of the edge ij must be $5a_{ij}$, and thus S has edge maximality with respect to the j-th edge, and so S is also edge maximal. Thus, S is a maximal stable set of G'. □

Therefore G' is equistable, and an equistable cost structure of G' is given by the costs defined above.

We are now ready to complete the proof that the maximum weight stable set problem on equistable graphs is NP-complete.

Consider the transformation given above, and let the weight of each vertex v_j be 1, and the weight of all other vertices is 0. Finding a maximum weight stable set in G' is equivalent to finding a maximum cardinality stable set in G, and this problem is APX-hard. □

Theorem 3. *Finding a maximum cardinality stable set in an equistable graph is APX-hard, even if the graph is given together with an equistable cost structure.*

Proof. Carry out the same transformation as in the proof of Theorem 2. However, in this case, replace each vertex v_j by Q identical copies of v_j, each with a cost of $\left(b_j + 3\sum_{ij\in E} a_{ij}\right)/Q$. For each $ij \in E$, a maximal stable set S' in the transformed graph G' will contain exactly one of the following:

- w_j plus all Q copies of v_i, or
- w_i plus all Q copies of v_j, or
- w_i, w_j, u_{ij}.

As before, every maximal stable set has the same cost t and every other subset has a different cost.

Moreover, any stable set S of cardinality K in G will induce a stable set S' in G' with

$$QK + n - K \le |S'| \le QK + n - K + m\,,$$

found as follows:

- For each $j \in S$, all Q copies of v_j are in S'.
- For each $j \in V \backslash S$, $w_j \in S'$.
- For all $ij \in E$ with $v_i \notin S$ and $v_j \notin S$, $u_{ij} \in S'$.

Suppose $Q = (m+n)/\epsilon$ for some fixed $\epsilon > 0$. Let S^* be a maximum stable set in G. Then G' contains a maximum stable set $\hat{S}$ such that $|\hat{S}| \geq Q|S^*|$. Suppose that one can guarantee a solution that is within a factor c from optimality for the stable set problem on equistable graphs $(c < 1)$. Then one could guarantee a factor $c - \epsilon$ from optimality for the stable set problem in general graphs. Suppose that we have a stable set S' in the transformed equistable graph G' such that $|S'| \geq c|\hat{S}|$. This set can be used to generate a stable set S in the original graph with

$$|S| \geq \frac{|S'| - (m+n)}{Q} \geq \frac{c|\hat{S}| - (m+n)}{Q} \geq c|S^*| - \epsilon|S^*| \,.$$

So, a c-approximation for the stable set problem in the equistable graphs yields a $(c - \epsilon)$-approximation for the stable set problem in the original graph. This shows that finding a maximum cardinality stable set in an equistable graph is APX-hard. □

The argument used to prove Theorem 1 also shows that the MINIMUM WEIGHT FALSE POINT PROBLEM is NP-hard. It turns out that the problem remains hard even for graphs with unit weights:

Theorem 4. *Finding a minimum independent dominating set in an equistable graph is APX-hard, even if the graph is given together with an equistable cost structure.*

Proof. One can do exactly the same transformation as above, this time with $Q = m^2$, except that one replaces each of the v_j vertices by Q copies of v_j and one replaces each of the w_j vertices by $2Q$ copies of w_j. Then any independent dominating set in the transformed graph will have all Q copies of v_j or it will have all $2Q$ copies of w_j.

Suppose that there is a maximum stable set S in G with K vertices. Then there is an independent dominating set S' in G' with

$$QK + 2Q(n - K) \leq |S'| \leq QK + 2Q(n - K) + m \,.$$

That is,

$$2Qn - QK \leq |S'| \leq 2Qn - QK + m \,.$$

The maximum stable set problem is APX-hard even if restricted to instances in which the maximum stable set size is strictly greater than $n/2$. We will show that any algorithm that guarantees a relative error of at most ϵ for the minimum independent domination problem for equistable graphs will induce a solution for the maximum stable set problem with a relative error of at most 3ϵ, restricted to instances with $m > 1/\epsilon$ and such that the maximum stable set size is strictly greater than $n/2$.

Consider such a graph G and let G' be the transformed equistable graph. Let S^* be a maximum stable set in G. Then G' contains an independent dominating set $\hat{S}$ such that $|\hat{S}| \leq 2Qn - Q|S^*| + m$. Suppose that we have an independent dominating set S' in G' such that $|S'| \leq (1 + \epsilon)|\hat{S}|$. This set can be used to generate a stable set S in the original graph with $|S| \geq 2n - |S'|/Q$. Since $|S'| \leq (1+\epsilon)|\hat{S}|$ and $|\hat{S}| \leq 2Qn - Q|S^*| + m$, it follows that $|S| \geq (1+\epsilon)|S^*| - 2\epsilon n - (1+\epsilon)m/Q$. Furthermore, as $(1+\epsilon)m/Q < 2\epsilon$, we obtain

$$|S| \geq (1+\epsilon)|S^*| - 2(n+1)\epsilon \geq (1+\epsilon)|S^*| - 4\epsilon|S^*| = (1-3\epsilon)|S^*|\,.$$

Thus, if one could approximate the minimum independent domination problem in equistable graphs by a factor better than $1+\epsilon$ in polynomial time, then one could approximate the maximum stable set problem by a factor better than $1-3\epsilon$. This proves that the minimum independent domination problem is APX-hard on equistable graphs. □

We conclude this section by examining the relationship between equistable graphs and hereditary graph classes. As already observed by Payan [13], equistable graphs do not form a hereditary class of graphs. For example, let A denote the graph obtained from a path P on four vertices by introducing a new vertex and joining it to the two middle vertices of P. The A graph is equistable and contains a non-equistable P_4 as an induced subgraph.

It is therefore natural to ask what is the largest hereditary class $[\mathcal{ES}]^-$ of graphs contained in the class of equistable graphs and, similarly, what is the smallest hereditary class $[\mathcal{ES}]^+$ of graphs that contains equistable graphs. Combining the above observations with some existing results from the literature, we can give a complete answer to these questions.

Proposition 1.
(i) $[\mathcal{ES}]^-$ *is the class of* P_4*-free graphs.*
(ii) $[\mathcal{ES}]^+$ *is the class of all graphs.*

Proof. The proof of (i) is straightforward. On one hand, since the graph P_4 is not equistable, the largest hereditary class of graphs contained in the class of equistable graphs must be a subclass of P_4-free graphs. On the other hand, P_4-free graphs are equistable [9]. Therefore, it follows that $[\mathcal{ES}]^- = \{P_4\text{-free graphs}\}$.

The reduction performed in the proof of Theorem 2 shows that every graph is an induced subgraph of an equistable graph. Therefore, the smallest hereditary class that contains equistable graphs is the class of all graphs. This establishes (ii). □

3 The Dynamic Programming Algorithm

In this section we present a dynamic programming solution for the MAXIMUM WEIGHT FALSE POINT PROBLEM (1). As special cases we obtain pseudo-polynomial-time algorithms for the MAXIMUM WEIGHT STABLE SET and the WEIGHTED INDEPENDENT DOMINATION PROBLEMS in equistable and interstable graphs (cf. Section 1), provided that the input graph is equipped with an equistable (resp. interstable) cost structure. Note that in the following analysis we adopt the simplifying assumption that arithmetic operations can be carried out in $O(1)$ time.

Let $V = \{v_1, \ldots, v_n\}$ be a finite set, $\mathbf{c} : V \to \mathbb{N}$ an integer-valued cost function and $w : V \to \mathbb{R}_+$ a set of weights. According to the framework outlined in the introduction we are going to represent the set of false points by requiring costs to fall within a prescribed subset of $\mathbb{R}_+$, see (2).

For a set $T \subset \mathbb{R}_+$ let $f_T : \mathcal{B}^V \to \mathcal{B}$ denote the function defined (via the set of false points) by $\mathcal{F}(f_T) = \{\mathbf{x} \in \mathcal{B}^V : \mathbf{c}^\top \mathbf{x} \in T\}$ and let $M \in \mathbb{N}$ be an integer satisfying $M \geq \sup(T)$. Let us also assume that there exists a membership oracle which for any given $k \in \mathbb{N}$ determines whether $k \in T$ holds.

Theorem 5. *Let V, $\mathbf{c}$, w, T and M as above. Then the* MAXIMUM WEIGHT FALSE POINT PROBLEM

$$\begin{array}{ll} \max & \mathbf{w}^\top \mathbf{x} \\ \textit{s.t.} & \mathbf{x} \in \mathcal{F}(f_T) \end{array} \tag{3}$$

can be solved in time $O(nM)$ using M calls to the membership oracle.

Proof. For each $i \in \{1, \ldots, n\}$ and $j \in \{0, 1, \ldots, M\}$, let us introduce the number $q_i(j)$ as the maximum possible weight of a subset of the first i elements of V whose total cost is j:

$$q_i(j) = \max \{ w(S) : S \subseteq \{v_1, \ldots, v_i\} \, , \ c(S) = j \} \, .$$

We can compute the values of $q_i(j)$ in a recursive manner. Starting with $i = 1$, we have, for each $j \in \{0, 1, \ldots, M\}$:

$$q_1(j) = \begin{cases} w(1), & \text{if } c(v_1) = j; \\ -\infty, & \text{otherwise.} \end{cases}$$

Now let $i > 1$ and assume that the values of $\{q_{i-1}(j) : j \in \{0, 1, \ldots, M\}\}$ have already been computed. If the cost of v_i exceeds j, then, since all the costs are positive, the element v_i cannot appear in any set that attains the maximum in $q_i(j)$; we thus have $q_i(j) = q_{i-1}(j)$ in this case. Otherwise, a subset of $\{v_1, \ldots, v_i\}$ achieving maximum weight can either contain v_i or not. Thus, for each $j \in \{0, 1, \ldots, M\}$:

$$q_i(j) = \begin{cases} \max \{ \ w(i) + q_{i-1}(j - c(v_i)) \ , \ q_{i-1}(j) \ \}, & \text{if } c(v_i) \leq j; \\ q_{i-1}(j), & \text{otherwise.} \end{cases}$$

Using the above recursive formula, we can compute all the $q_i(j)$ values in time $O(nM)$. The optimum of (3) is now given by $\max\{q_n(j) \mid j \in T\}$; since we already have the $q_n(j)$ values, we can easily find this value in time $O(M)$ using M calls to the membership oracle. □

Notice that by replacing "$\max$" with "$\min$" in the above algorithm, we can also solve the MINIMUM WEIGHT FALSE POINT PROBLEM. Thus Theorem 5 provides a solution to the MAXIMUM and MINIMUM WEIGHT FALSE POINT PROBLEMS for the generic Boolean framework outlined in **Case 3** (Section 1). We now specialize this result to **Cases 1** and **2**, which leads to solving the corresponding graph problems.

Corollary 1 (Equistable graphs)**.** *Let $G = (V, E)$ be a graph with an equistable cost structure $(\mathbf{c}, t)$. For any weight function $w : V \to \mathbb{R}_+$ the* MAXIMUM WEIGHT STABLE SET *and the* WEIGHTED INDEPENDENT DOMINATION PROBLEMS *for G can be solved in time $O(nt)$, where $n = |V|$.*

Proof. According to the definition of equistable graphs, for the set $T = \{t\}$ the function f_T is the maximal stability function of G. Let $M = t = \sup(T)$ and notice that V, $\mathbf{c}$, $\mathbf{w}$, T and M satisfy the conditions of Theorem 5. Since the membership oracle simply has to decide whether $k = t$ holds for a given integer k, the claim immediately follows. □

Corollary 2 (Interstable graphs)**.** *Let $G = (V, E)$ be a graph with an interstable cost structure $(\mathbf{c}, [a, b])$. For any weight function $w : V \to \mathbb{R}_+$ the* MAXIMUM WEIGHT STABLE SET *and the* WEIGHTED INDEPENDENT DOMINATION PROBLEMS *for G can be solved in time $O(nb)$, where $n = |V|$.*

Proof. Let $T = [a, b]$, and $M = b = \sup(T)$. The claim follows similarly to the proof of Corollary 1. □

4 Further Complexity Issues

For an equistable graph G, let us define

$$t(G) = \min\{t \in \mathbb{N} : \text{there is an equistable cost structure of } G \text{ with target } t\}\,.$$

In view of the above $O(nt)$ algorithm and the NP-hardness result of Section 2, it is natural to expect that there exist equistable graphs on n vertices such that $t(G)$ is not bounded by any polynomial. Indeed, it turns out that there are equistable graphs on n vertices for which $t(G) = \Omega\left(\frac{2^{n/2}}{\sqrt{n}}\right)$.

We start with two preliminary observations.

For a graph G, we denote by $\mathcal{S}(G)$ the set of all maximal stable sets of G, and by $\mathcal{T}(G)$ the set of all other nonempty subsets of $V(G)$.

Proposition 2. *Let G be a graph, and let $\mathbf{c} : V(G) \to \mathbb{R}_+$. Then, $\mathbf{c}$ is not an equistable cost function of G if and only if either $c(S_1) \neq c(S_2)$ for some $S_1, S_2 \in \mathcal{S}(G)$, or $c(S) = c(T)$ for some $S \in \mathcal{S}(G)$ and $T \in \mathcal{T}(G)$.*

Proof. Let $\mathbf{c} : V(G) \to \mathbb{R}_+$. Clearly, if not all maximal stable sets have the same cost, or if the cost of a non-maximal-stable set coincides with the cost of a maximal stable set, then $\mathbf{c}$ is not an equistable cost function.

Conversely, suppose that $c(S_1) = c(S_2)$ for all $S_1, S_2 \in \mathcal{S}(G)$. Then all maximal stable sets have the same cost, say t. If, in addition, $c(S) \neq c(T)$ holds for every $S \in \mathcal{S}(G)$ and $T \in \mathcal{T}(G)$, then the only sets of cost t are maximal stable sets, and the pair $(\mathbf{c}, t)$ is an equistable cost structure of G. □

We say that a finite set A of positive numbers has the *distinct-subset-sums* (DSS) property if and only if all the sums of the form $\sum_{a \in A'} a$, where A' ranges over all subsets $A' \subseteq A$, are distinct.

Let G_n denote a disjoint union of n copies of K_2. The graphs G_n are P_4-free, and thus equistable [9]. Also, we remark that the maximal stable sets of G_n are precisely the sets obtained by choosing one vertex from each copy of K_2.

Proposition 3. *Let G_n denote a disjoint union of n copies of K_2, and let $\mathbf{c} : V(G_n) \to \mathbb{R}_+$. Then, $\mathbf{c}$ is an equistable cost function of G_n if and only if the following two conditions are satisfied:*

(i) *For every $u, v \in V(G_n)$, $c(u) = c(v)$ if and only if $u = v$ or $u, v \in E(G_n)$.*

(ii) *The set of costs $\{c(v) : v \in V(G_n)\}$ has the distinct-subset-sums property.*

Proof. Let $V(G_n) = \{v_1, v_1', \ldots, v_n, v_n'\}$ so that $E(G_n) = \{v_1v_1', \ldots, v_nv_n'\}$.

First, we show necessity of the two conditions.

Consider an equistable cost structure $(\mathbf{c}, t)$ of G_n. Let uv be an edge of G_n, and let S be a maximal stable set in G_n such that $u \in S$. Then $v \notin S$, and the set S' obtained by replacing u by v in S is again maximally stable. Since all maximal stable sets have the same cost, we conclude that $c(u) = c(v)$.

Conversely, suppose that $c(u) = c(v)$ for two vertices u and v such that $u \neq v$ and $uv \notin E(G_n)$. Let S be a maximal stable set in G_n such that $u \in S$ and $v \notin S$. The set S' obtained by replacing u by v in S is of the same cost as S, and thus maximally stable. It follows that the unique neighbor v' of v in G_n does not belong to S'. But then $S \cap \{v, v'\} = \emptyset$, contradicting the fact that S is a maximal stable set. This settles (i).

For (ii), suppose that the set of costs $\{c(v) : v \in V(G_n)\}$ does not have the DSS property. Also, let c_i be the cost assigned to the vertices v_i and v_i', for $i \in \{1, \ldots, n\}$. Assume for contradiction that there exist two distinct nonempty subsets $I, J \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in I} c_i = \sum_{j \in J} c_j$ (without loss of generality, I and J can be assumed to be disjoint). Then, the set

$$U := \{v_i : i \in I\} \cup \{v_i' : i \in I\} \cup \{v_i : i \in \{1, \ldots, n\}, i \notin I \cup J\}$$

is a non-stable subset of $V(G_n)$ of total cost t, contradicting the fact that $\mathbf{c}$ is an equistable cost function of G_n with target t. This settles (ii) and with it the necessity of the two conditions.

Now, we show sufficiency. Suppose that $\mathbf{c} : V(G_n) \to \mathbb{R}_+$ satisfies the conditions (i) and (ii) but is not an equistable cost function. Since the maximal stable sets of G_n are precisely the sets obtained by choosing one vertex from each copy of K_2, condition (i) implies that they all have the same cost. By Proposition 2 we conclude that there exist $S \in \mathcal{S}(G)$ and $T \in \mathcal{T}(G)$ such that $c(S) = c(T)$. Clearly, we may assume that $S = \{v_1, \ldots, v_n\}$. Furthermore, we may assume by (i) that for every $i \in \{1, \ldots, n\}$, we have $v_i \in T$ whenever $v_i' \in T$ (since otherwise we can replace v_i' with v_i to obtain a set in $\mathcal{T}(G_n)$ of the same cost). Let $I = \{i : i \in \{1, \ldots, n\}, v_i, v_i' \in T\}$, and $J = \{j : j \in \{1, \ldots, n\}, v_j \in S \backslash T\}$. By definition, the sets I and J are disjoint. Moreover, since all the costs are positive, $c(S) = c(T)$ implies that neither of the sets S, T is contained in the other one, and thus I and J are non-empty. Finally, the condition $c(S) = c(T)$ implies that $\sum_{i \in I} c_i = c(T \backslash S) = c(S \backslash T) = \sum_{j \in J} c_j$. This contradicts the property (ii) and completes the proof of the proposition. □

Theorem 6. *Let G_n denote a disjoint union of n copies of K_2. Then, $t(G_n) = \Omega\left(\frac{2^n}{\sqrt{n}}\right)$.*

Proof. Consider an equistable cost structure $(\mathbf{c}, t)$ of G_n. By Proposition 3, the set of costs $\{c(v) : v \in V(G_n)\}$ has the DSS property. As shown by Erdős and Moser in [3], the maximum element of any n-element set of positive integers with the DSS property must be of order $\Omega\left(\frac{2^n}{\sqrt{n}}\right)$. Therefore, it follows that $t \geq \max\{c(v) : v \in V(G_n)\} = \Omega\left(\frac{2^n}{\sqrt{n}}\right)$ and the proof is complete. □

We conclude the paper with another hardness result. Whether equistable graphs can be recognized in polynomial time is an interesting, and to the best of our knowledge still open, question.[1] However, the theorem below seems to indicate that any potential polynomial recognition algorithm would have to rely on

[1]As mentioned in [7], referring to a remark by Igor Zverovich, there is an exponential-time algorithm to recognize an equistable graph.

the structural properties of equistable graphs, as even the 'correctness' of equistable cost functions is hard to verify.

Theorem 7. *Given a graph G and a cost function $\mathbf{c} : V(G) \to \mathbb{N}$, it is co-NP-complete to determine whether $\mathbf{c}$ is an equistable cost function of G.*

Proof. The problem is in co-NP, since by Proposition 2 we can exhibit a certificate (verifiable in polynomial time) which shows that $\mathbf{c}$ is not an equistable cost function.

To show NP-hardness, we use a reduction from the following NP-complete problem called *weak partition* [15, 16, 17]:

Instance: A finite set A and a size $s(a) \in \mathbb{N}$ for each $a \in A$.

Question: Are there disjoint non-empty subsets $A_1, A_2 \subseteq A$ such that $\sum_{a\in A_1} s(a) = \sum_{a\in A_2} s(a)$?

Consider an instance of the weak partition problem consisting of a set A and sizes $(s(a) : a \in A)$.

We may assume that all the sizes $s(a)$ are distinct (since otherwise the answer to the weak partition problem is *yes*). We construct a graph $G = (V, E)$ and a cost function $\mathbf{c} : V(G) \to \mathbb{N}$ as follows:

- $V = A \cup A'$ where $A' = \{a' : a \in A\}$ is a disjoint copy of A,
- $E = \{aa' : a \in A\}$,
- $c(a) = c(a') = s(a)$ for every $a \in A$.

Note that G is isomorphic to the graph G_n (with $n = |A|$) from Proposition 3. By Proposition 3, $\mathbf{c}$ is an equistable cost function of G if and only if the set $\{c(v) : v \in V(G_n)\}$ has the distinct-subset-sums property. Clearly, this is the case if and only if the answer to the weak partition problem is *no*, and any algorithm for determining whether a given cost function is an equistable cost function of a given graph can be used to solved the weak partition problem. This completes the proof. □

5 Conclusion

In this paper, we provided hardness results and simple pseudo-polynomial-time algorithms for the MAXIMUM WEIGHT STABLE SET and the WEIGHTED INDEPENDENT DOMINATION PROBLEMS in equistable graphs equipped with an equistable cost structure. The pseudo-polynomial algorithms are based on a dynamic programming approach and can be applied within the more general framework of Boolean optimization.

The problem of recognizing equistable graphs in polynomial time is still open. One of the results in this paper shows that verifying whether a given cost function on the vertices of a graph defines an equistable cost structure is a hard problem, indicating that any polynomial time recognition algorithm of equistable graphs would most probably have to rely on the structural properties of equistable graphs. This provides additional motivation for further investigation of the structural properties of equistable graphs, initiated for particular graph classes in [6, 7, 10, 14] and continued for general equistable graphs in [11].

Acknowledgments

The first and third author are indebted to Dr. Peter L. Hammer for introducing this topic to them, and for encouraging the early stages of research.

We would also like to thank Nir Halman for helpful discussions.

References

[1] A. BRANDSTÄDT, V.B. LE and S. MAHFUD, New applications of clique separator decomposition for the Maximum Weight Stable Set Problem. *Theor. Comput. Sci.* 70 (2007) 229–239.

[2] A. BRANDSTÄDT and S. MAHFUD, Maximum weight stable set on graphs without claw and co-claw (and similar graph classes) can be solved in linear time. *Inf. Process. Lett.* 84 (2002) 251–259.

[3] P. ERDŐS, Problems and results from additive number theory, *Colloq. Théorie des nombres*, Bruxells (1955) 127–137.

[4] M.R. GAREY and D.S. JOHNSON (1979). Computers and intractability. A guide to the theory of NP-completeness, *CA, Freeman*.

[5] V. GIAKOUMAKIS and I. RUSU, Weighted parameters in $(P_5, \overline{P}_5)$-free graphs, *Discrete Appl. Math.* 80 (1997) 255–261.

[6] E. KORACH and U. N. PELED, Equistable series-parallel graphs. *Stability in graphs and related topics. Discrete Appl. Math.*, 132, no. 1-3 (2003), 149–162.

[7] E. KORACH, U. N. PELED and U. ROTICS, Equistable distance-hereditary graphs, *Discrete Appl. Math.* 156 (2008) 462–477.

[8] V.V. LOZIN and M. MILANIČ, A polynomial algorithm to find an independent set of maximum weight in a fork-free graph, *Proceedings of the Seventeenth Annual ACM–SIAM Symposium on Discrete Algorithms (Miami, FL, 2006)*, 26–30, ACM, New York, 2006.

[9] N.V.R. MAHADEV and U.N. PELED, Threshold Graphs and Related Topics, *Annals of Discrete Mathematics*, 56 (1995) 1–543.

[10] N.V.R. MAHADEV, U.N. PELED and F. SUN, Equistable graphs, *J. Graph Theory*, 18 (1994) 281-299.

[11] M. MILANIČ and G. RUDOLF, Structural results for equistable graphs and related graph classes. In preparation.

[12] R. MOSCA, Stable sets of maximum weight in $(P_7, banner)$-free graphs, *Discrete Math.* 308 (2008) 20–33.

[13] C. PAYAN, A class of threshold and domishold graphs: equistable and equidominating graphs, *Discrete Math.*, 29-1 (1980) 47-52.

[14] U.N. PELED and U. ROTICS, Equistable chordal graphs. *Stability in Graphs and Related Topics. Discrete Appl. Math.* 132 (2003) 203–210.

[15] A. RUBIN, A note on sum-distinct sets and a problem of Erdős, *Abstracts of the AMS 2* (1981), 42.

[16] A. SHAMIR, On the cryptocomplexity of knapsack systems, *Proceedings 11th Ann. ACM Symp. on Theory of Computing*, pp. 118-129, Association for Computing Machinery, New York, 1979.

[17] P. VAN EMDE BOAS, Another NP-complete partition problem and the complexity of computing short vectors in a lattice, Report No. 81-04, Department of Mathematics, University of Amsterdam, 1981.