A Refined Algorithm for Maximum Independent Set in Degree-4 Graphs

MINGYU XIAO

School of Computer Science and Engineering, University of Electronic Science and Technology of China, China, myxiao@gmail.com

HIROSHI NAGAMOCHI

Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Japan, nag@amp.i.kyoto-u.ac.jp

Abstract

The maximum independent set problem is one of the most important problems in worst-case analysis for exact algorithms. Improvement on this problem in low-degree graphs can be used to get improvement on the problem in general graphs. In this paper, we show that the maximum independent set problem in an n-vertex graph with degree bounded by 4 can be solved in $O^*(1.1376^n)$ time and polynomial space, improving all previous exact algorithms for this problem. As most fast exact algorithms, this algorithm is a branch-and-search algorithm and analyzed by using the measure and conquer method. To effectively analyze the running time bound, we use the idea of 'shift' to save some decreasing on the measure from some good branches to some bad branches. After treating cycles of length 3 and 4 in the graph, we check carefully what will happen after branching on a degree-4 vertex (without any local structure), and then we can get the claimed improvement.

Key words. Exact Algorithm, Independent Set, Measure and Conquer

1 Introduction

The famous *maximum independent set* problem (MIS), to find a vertex set of maximum cardinality in a graph such that the induced subgraph on the vertex set has no edge, is one of the most important problems in the line of research on worst-case analysis of exact algorithms for NP-hard problems. The trivial algorithm of checking all possible vertex subsets will get running time bound of $O^*(2^n)$. In 1977, Tarjan and Trojanowski [14] designed the first nontrivial algorithm with running time $O^*(2^{n/3})$. The bound of the running time to exactly solve the problem has been further improved for many times [10, 13, 6, 11, 2]. Currently, the fastest algorithms are Robson's $O^*(1.2109^n)$-time exponential-space algorithm [13] and Bourgeois *et al.*'s $O^*(1.2114^n)$-time polynomial-space algorithm [2].

Almost all previous algorithms for MIS take MIS in low-degree graphs as the most important subcases and analyze them carefully. In fact, MIS in degree-i graphs (graphs with maximum degree i) for small i will become one of the crucial bottlenecks for solving MIS in degree-$(i+1)$ graphs (and then in general graphs). With improved running time bounds on MIS3 (MIS in degree-3 graphs) and MIS4 (MIS in degree-4 graphs), Bourgeois *et al.* [2] get improvements on MIS5 (MIS in degree-5 graphs), MIS6 (MIS in degree-6 graphs) and MIS in general graphs by using a bottom-up method. Due to the importance of the problems in low-degree graphs, we can find a long list of contributions to fast exact algorithms for MIS in low-degree graphs in the literature [7, 9, 17, 12, 16, 1, 18] . Now MIS3 can be solved in $O^*(1.0836^n)$ time and polynomial space [18]. For MIS4, Kneis *et al.* [11] got a running time bound of $O^*(1.2132^n)$ by using a computer-aided method to check a huge number of cases, and finally MIS4 will not be the bottleneck case in their algorithm for MIS in general graph. Bourgeois *et al.* [2] carefully checked some local structures and then got a bound of $O^*(1.1571^n)$.

[1] Technical report 2013-002, April 8, 2013

We further improved the bound to $O^*(1.1447^n)$ in [19], which can be regarded as a previous version of this paper. Compared with [19], this paper carefully checks what will happen after branching on a degree-4 vertex by including it to the solution or not, which is the worst case in [19]. We find that the worst case will not always occur, and then we can save some decreasing of the measure from the following 'good' branches to this 'bad' branch. Therefore, we can further improve the running time bound to $O^*(1.1376^n)$. To get this improvement, we also need to refine the analysis of treating vertices of degree ≥ 5 vertices (it is possible to create vertices of degree ≥ 5 in our algorithm even when the degree of the initial graph is bounded by 4), cycles of length 3 and 4, and other local structures.

The rest of the paper is organized as follows. Section 2 gives the notations used in the paper. Sections 3 and 4 introduce reduction and branching rules, respectively, which will be used in our algorithm. Section 5 presents our algorithm, a framework of the analysis and the main result, while the detailed analysis and proofs are discussed in Section 6. Finally Section 7 makes some concluding remarks.

2 Notation System

Let $G = (V, E)$ stand for a simple undirected graph with a set V of vertices and a set E of edges. Let n denote the total number of vertices in a given graph G. For simplicity, we may denote a singleton set $\{v\}$ by v. For a vertex v in a graph, $N(v)$ denotes the set of all neighbors of v, $\delta(v)$ $(= |N(v)|)$ the degree of v, and $N_2(v)$ the set of vertices with distance exactly 2 from v. Denote $N[v] = N(v) \cup \{v\}$ and $N_2[v] = N_2(v) \cup N[v]$. Let X be a subset of vertices in G. We may also use $N(X)$ to denote the neighbors of a set X of vertices, i.e., $N(X) = \cup_{v \in X} N(v) - X$, and let $N[X] = X \cup N(X)$. Let $G - X$ be the graph obtained from G by removing the vertices in X together with any edges incident to a vertex in X, $G[X] = G - (V - X)$ be the graph induced from G by the vertices in X, and G/X denote the graph obtained from G by contracting X into a single vertex (removing self-loops and parallel edges). A *k-cycle* C is a simple cycle of length $k \geq 3$, which is denoted by a sequence of the k vertices $v_1, v_2, \ldots, v_k$ in C such that, for each $i = 1, 2, \ldots, k$, v_i and v_{i+1} are adjacent, where we interpret $v_{k+1} = v_1$. The *line graph* of a graph G is the graph whose vertices correspond to the edges of G, and two vertices are adjacent if and only if the corresponding two edges share a same endpoint in G. Throughout the paper we use a modified O notation that suppresses all polynomially bounded factors. For two functions f and g, we write $f(n) = O^*(g(n))$ if $f(n) = g(n)\text{poly}(n)$ holds for a polynomial $\text{poly}(n)$ in n.

3 Reduction Rules

When the instance has some special structures, we may apply our reduction rules to decrease the size of instance directly by finding a partial solution. Reduction operations will not exponentially increase the running time bound. After reducing some special structures by applying the reduction rules, we can effectively search the solution in the resulting graph by applying our branching rules. Next, we introduce our reduction rules, all of which are known in the literature.

Let $\alpha(G)$ denote the size of a maximum independent set of a graph G. Clearly if there are degree-0 vertices in a graph, then any maximum independent set of the graph includes all these vertices and we remove them from the graph. Thus for the set V_0 of degree-0 vertices in G, it holds $\alpha(G) = \alpha(G - V_0) + |V_0|$.

Dominance

We say that a vertex u dominates another vertex v if $N[u] \subseteq N[v]$. A vertex is called dominated *if it is dominated by some other vertex. If there is a dominated vertex v in a graph, then we remove v from the graph.*

The following lemma is folklore and has been used in most previous algorithms for MIS and related problems [3, 6].

Lemma 1 *For a dominated vertex v in graph G,*

$$\alpha(G) = \alpha(G - v).$$

In particular, folding a degree-1 vertex u means to remove the unique neighbor v of u. Next we assume that the above rule has been repeatedly applied until there is no dominated vertex. Note that if the neighbor set $N(u)$ of a vertex u induces a complete graph, then every neighbor v of u is dominated by u. From this observation we see that there exists no longer a degree-1 vertex or a degree-2 vertex with two adjacent neighbors when a graph has no dominated vertex.

Unconfined Vertices

A *satellite* of a vertex u is a vertex $s \in N_2(u)$ which has a neighbor $s' \in N(u) \cap N(s)$ such that $N[s'] - N[u] = \{s\}$, where s' is also called the *parent* of satellite s. We use S_u to denote the set of vertex u and all satellites of u. The concept of satellites was first introduced in [11] and has been extended to "confining sets" in [18]. We can easily observe that when there is no maximum independent set containing u (i.e., every maximum independent set contains u), every maximum independent set contains S_u [11]. A vertex u is called *unconfined* if some two satellites in $S_u - \{u\}$ are adjacent[1]. It is shown that an unconfined vertex u can be removed without losing a maximum independent set [18].

Lemma 2 [18] *For an unconfined v in graph G,*

$$\alpha(G) = \alpha(G - v).$$

A vertex v is called a roof, if it is in a 5-cycle $vu_1u_2u_3u_4$ such that u_1 and u_4 are degree-3 vertices adjacent each other. If there is a roof, we remove it from the graph. We see that roofs are unconfined vertices.

Folding degree-2 vertices

Folding a degree-2 vertex v (with two nonadjacent neighbors a and b) means contracting v, a and b into a single vertex s.

Fig. 1(a) illustrates the operation of folding a degree-2 vertex. Note that the operation of folding a degree-2 vertex may create a vertex of high degree. Therefore, even if the initial graph is of maximum degree 4, our algorithm may create vertices of degree ≥ 5 during an execution of it.

A vertex v together with its neighbors $N(v)$ is called a *funnel* (or a $\delta(v)$-funnel) if $N[v] - \{a\}$ induces a complete graph for some $a \in N(v)$, and is denoted by a-v-$(N(v) - \{a\})$. Note that v dominates a vertex in $N(a) \cap N(v)$ if any. We have that $N(a) \cap N(v) = \emptyset$, when there is no dominated vertex.

Folding funnels

Folding a 3-funnel a-v-$\{b, c\}$ meas that we add an edge between every non-adjacent pair (x, y) of vertices $x \in N(a) - \{v\}$ and $y \in N(v) - \{a\} = \{b, c\}$ and then remove vertices a and v.

Fig. 1(b) illustrates the operation of folding a 3-funnel. Let $G^\star(v)$ denote the graph after folding a degree-2 vertex v or a 3-funnel a-v-$\{b, c\}$ in G. Then we have the following lemma.

Lemma 3 *For a degree-2 vertex v or a funnel a-v-$\{b, c\}$ in graph G, we have $\alpha(G) = 1 + \alpha(G^\star(v))$.*

[1] Unconfined vertices in [11] are defined in a more general way.

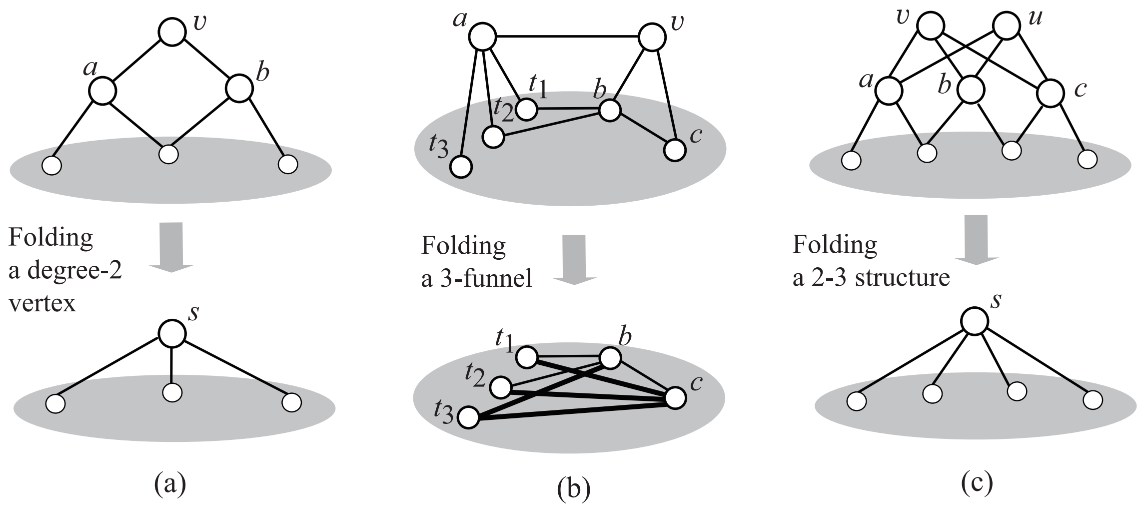

Figure 1: Illustrations of folding operations

The correctness of Lemma 3 has been discussed in many references [3, 6]. In fact, folding a funnel is also a special case of a reduction rule introduced in [6]. In general, folding a funnel may increase our measure (defined in Section 6), which is unexpected in our algorithm. We call a 3-funnel a-v-$\{b, c\}$ in a graph with minimum degree 3 a *short funnel* if $\delta(a) \leq 4$ and there are at least $\delta(a) - 2$ edges between $N(a) - \{v\}$ and $\{b, c\}$. In our algorithm, we will reduce short funnels only and leave some other funnels.

In our algorithm, we will also use the following reduction rules. A 2-3 *structure* is a subgraph induced by two independent degree-3 vertices v and u and their three common neighbors a, b and c (see Fig. 1(c)), and is denoted by $\{v, u\}$-$\{a, b, c\}$. Similarly a 3-4 *structure* is a subgraph induced by three independent vertices v, u and z of degree ≥ 3 and the union of their neighbors such that $|N(v) \cup N(u) \cup N(z)| = 4$, and is denoted by $\{v, u, z\}$-$N(v) \cup N(u) \cup N(z)$.

Folding 2-3 and 3-4 structures
Let A-B be a 2-3 structure or 3-4 structure in graph G. Folding A-B means
(a) removing $A \cup B$ from the graph, when B is not an independent set in G; or
(b) contracting $A \cup B$ into a singe vertex and deleting parallel edges and self-loops from the graph, when B is an independent set in G.

Lemma 4 *Let $G^\star$ be the graph obtained from a graph G by folding a k-$(k+1)$ structure ($k = 2$ or 3) in G. Then $\alpha(G) = k + \alpha(G^\star)$.*

The above reduction rule is a special case of the crown reduction introduced in [4]. The correctness of folding an A-B structure follows from the next observation: When B is not an independent set, there is a maximum independent set that contains A. When B is an independent set, there is a maximum independent set that contains either B or A.

Line graphs
If graph G is the line graph of a graph G', then we obtain a maximum independent set of G directly by finding a maximum matching M in G' and taking the corresponding vertex set V_M in G as a solution.

Not every graph is a line graph. There are several methods to check whether a graph is a line graph or not, which depend on characterizations of line graphs [15]. In this paper, we only need to check whether a graph is the line graph of a 3-regular graph, which can be easily done (note that a graph is the line graph of a 3-regular graph if and only if the graph has only degree-4 vertices and each of them is contained in two edge-disjoint triangles).

A set X of vertices is called a *reducible* vertex set if $|N(X)| \leq 2$, $2 \leq |X| \leq 26$ and X induces a connected subgraph from G. The following lemmas provide reduction rules to eliminate reducible vertex sets.

A partition (V_1, Z, V_2) of the vertex set of G is called a *separation* if $N(V_1) \subseteq Z \supseteq N(V_2)$. Let v be a vertex cut in a graph G, which gives a separation $(V_1, \{v\}, V_2)$. Let $G_i = G[V_i]$, $i = 1, 2$, and $V_1^v = V_1 - N(v)$. The induced graph $G[V_1^v]$ is denoted by G_1^v.

Lemma 5 *For subgraphs G_1 and G_1^v defined on a separation $(V_1, \{v\}, V_2)$ in a graph G, it holds*

$$\alpha(G) = \alpha(G_1) + \alpha(G^\star),$$

where $G^\star = G - V_1$ if $\alpha(G_1) = \alpha(G_1^v)$, and $G^\star = G_2$ otherwise. A maximum independent set in a graph G can be constructed from any maximum independent sets to G_1, G_1^v and $G^\star$.
(See Appendix for a proof.)

For a separation $(V_1, \{u, v\}, V_2)$ of a graph G, let $G_i = G[V_i]$ $(i = 1, 2)$, $V_1^v = V_1 - N(v)$, $V_1^u = V_1 - N(u)$ and $V_1^{uv} = V_i - N(\{u, v\})$, $i \in \{1, 2\}$. The induced graphs $G[V_1^v]$, $G[V_1^u]$ and $G[V_1^{uv}]$ are simply denoted by G_1^v, G_1^u and G_1^{uv} respectively. Let $\widetilde{G_2}$ denote the graph obtained from $G[V_2 \cup \{u, v\}]$ by adding an edge uv if v and u are not adjacent.

Lemma 6 *For subgraphs G_1, G_1^v, G_1^u and G_1^{uv} defined on a separation $(V_1, \{u, v\}, V_2)$ in a graph G, it holds*

$$\alpha(G) = \alpha(G_1) + \alpha(G^\star),$$

where

$$G^\star = \begin{cases} G[V_2 \cup \{u,v\}] & \text{if } \alpha(G_1^{uv}) = \alpha(G_1), \\ \widetilde{G_2} & \text{if } \alpha(G_1^{uv}) < \alpha(G_1^u) = \alpha(G_1^v) = \alpha(G_1), \\ G[V_2 \cup \{v\}] & \text{if } \alpha(G_1^u) < \alpha(G_1^v) = \alpha(G_1), \\ G[V_2 \cup \{u\}] & \text{if } \alpha(G_1^v) < \alpha(G_1^u) = \alpha(G_1), \\ G/(V_1 \cup \{u, v\}) & \text{if } \alpha(G_1^{uv}) + 1 = \alpha(G_1) \text{ and } \alpha(G_1^v) < \alpha(G_1), \\ G_2 & \text{otherwise } (\alpha(G_1^{uv}) + 2 \leq \alpha(G_1) \text{ and } \alpha(G_1^v) < \alpha(G_1)). \end{cases}$$

A maximum independent set in a graph G can be constructed from any maximum independent sets to G_1, G_1^v, G_1^u, G_1^{uv} and $G^\star$.
(See Appendix for a proof.)

Given a reducible vertex set X in a graph G, let $V_1 = X$ and $V_2 = V - N(V_1)$, where $Z = N(V_1)$ is a cut-vertex or a cut-pair. We convert G into $G^\star$ according to Lemma 5 or Lemma 6. Note that we can compute $\alpha(G_1)$ and $\alpha(G_1^v)$ in Lemma 5 or $\alpha(G_1)$, $\alpha(G_1^v)$, $\alpha(G_1^u)$ and $\alpha(G_1^{uv})$ in Lemma 6 in constant time since $|X|$ is bounded by a constant.

Definition 7 *A graph is called a* reduced graph*, if it contains none of dominated vertices, unconfined vertices, degree-2 vertices, short funnels,* 2-3 *structures,* 3-4 *structures, and reducible vertex sets, and has no connected component which is the line graph of a* 3*-regular graph, or a graph of at most* 23 *vertices.*

When a graph G is not a reduced graph, we use the algorithm RG(G, s) in Figure 2 to find a partial solution in polynomial time before a reduced graph is obtained. The algorithm in Figure 2 is a collection of all above reduction operations. Note that for the purpose of analysis, we apply the rule of folding degree-2 vertices in Step 2 before removing dominated vertices and unconfined vertices.

Input: A graph $G = (V, E)$ and the size s of the current partial solution (initially $s = 0$).
Output: A reduced graph $G' = (V', E')$ and the size s of a partial solution S' with $(S' \cup N(S')) \cap V' = \emptyset$ in G.

1. **If** {Graph G has a component P that has at most 23 vertices, or is the line graph of a 3-regular graph}, **return** $(G', s') := \mathrm{RG}(G - P, s + \alpha(P))$.
2. **Elseif** {There is a degree-2 vertex v whose neighbors are not adjacent}, **return** $(G', s') := \mathrm{RG}(G^\star(v), s + 1)$.
3. **Elseif** {There is a dominated or unconfined vertex $v \in V$}, **return** $(G', s') := \mathrm{RG}(G - v, s)$.
4. **Elseif** {There is a reducible vertex subset V_1}, **return** $(G', s') := \mathrm{RG}(G^\star, s{+}\alpha(G_1))$
5. **Elseif** {There is a k-$(k{+}1)$ structure ($k = 2$ or 3)}, **return** $(G', s') := \mathrm{RG}(G^\star, s{+}k)$.
6. **Elseif** {There is a short funnel a-v-$(N(v){-}\{a\})$}, **return** $(G', s') := \mathrm{RG}(G^\star(v), s+1)$.
7. **Else return** $(G', s') := (G, s)$.

Figure 2: The Algorithm $\mathrm{RG}(G, s)$

4 Branching Rules

If the current graph is a reduced graph, then we use the branch-and-search method to find a solution. In our algorithm, we use three kinds of branching rules. The simplest branch rule is to branch on a vertex by including it to the solution set or not. This simplest branch rule has been extended in [11, 18]. We can branch on a vertex v by excluding it from the solution set or including S_v to the solution set (S_v is defined in unconfined vertices). That is, in the first branch we will delete v from the graph and in the second branch we will delete $N[S_v]$ from the graph. This is our first branching rule. We will choose a vertex v to branch on such that a reducible subset is created by removal of v or v has the maximum degree. Besides this branching rule of branching on a single vertex, we also two other branching rules, *branching on a funnel* and *branching on a 4-cycle.*

We use reduction rules to short funnels only. For other kinds of funnels, we may use the following branching rules to treat them.

Lemma 8 [18] *Let a-v-$(N(v){-}\{a\})$ be a funnel in graph G. Then there is a maximum independent set S in G such that either $v \in S$ or $S_a \subseteq S$.*

Based on Lemma 8, we get the following branching rule.

Branching on a funnel
Branching on a funnel a-v-$(N(v) - \{a\})$ means branching by either including v in the solution set or including S_a in the solution set.

Lemma 9 [18] *Let $abcd$ be a 4-cycle in graph G. For any independent set S in G, either $a, c \notin S$ or $b, d \notin S$.*

Based on Lemma 9, we get the following branching rule.

Branching on a 4-cycle
Branching on a 4-cycle abcd means branching by either excluding a and c from the independent set or excluding b and d from the independent set.

5 The Algorithm, Basic Analysis and Results

The main idea of our algorithm is follows. While the current graph is not a reduced graph, we call RG(G, s) to reduce the graph and find a partial solution; otherwise we invoke our branching rules to search a solution. By branching on vertices of maximum degree, we can find a solution. However, this simple branching rule is not always good enough for achieving an improved time bound. We will apply the other two branching rules: branching on a funnel and branching on a 4-cycle, to eliminate certain types of triangles and all 4-cycles, which appear in the worst cases in the simple branching. Before giving the detailed algorithm, we introduce the idea to analyze the time bound of our algorithm.

5.1 Framework for Analysis

We use the measure and conquer method [6] to analyze the running time bound of our algorithm. By this method, we introduce a weight to each vertex in the graph according to the degree of the vertex, $w : \mathbb{Z}_+ \to \mathbb{R}_+$ (where $\mathbb{Z}_+$ and $\mathbb{R}_+$ denote the sets of nonnegative integers and nonnegative reals, respectively): we denote by w_i the weight $w(v)$ of a vertex v of degree $i \geq 0$. A vertex of higher degree may receive a larger weight. Then we adopt $w(G) = \sum_i w_i n_i$ as the measure of the graph G, where n_i denotes the number vertices of degree i in the graph. We will set vertex weight such that the measure $w = w(G)$ satisfies the *measure condition*: (i) when $w \leq 0$ the instance can be solved in polynomial time; (ii) the measure w will never increase after each application of the reduction/branching rules in the algorithm; and (iii) the measure w will decrease in each of the subinstances generated by applying a branching rule. This is important for us to construct recurrences for each branching operation in the algorithm.

For each branch operation, we will generate two subinstances G_1 and G_2 by deleting some vertices from the graph. After deleting some vertices, we can reduce the measure from two parts: the weight of the vertices being deleted and partial weight of the vertices adjacent to the deleted vertices since their degree will decrease. Let $t_{(i)}$ $(i = 1, 2)$ be a lower bound on the decrease of the measure in the subinstance (i.e., $w(G) - w(G_i) \geq t_{(i)}$). Then we get the recurrence

$$C(w) \leq C(w - t_{(1)}) + C(w - t_{(2)}),$$

where $C(w)$ is the worst size of the search tree when our algorithm runs on any graph with measure w. The unique positive real root of the function $f(x) = 1 - x^{-t_{(1)}} - x^{-t_{(2)}}$ is called the *branching factor* of the above recurrence. Let τ be the maximum branching factor among all branching factors in the search tree. Then $C(w) = \tau^w$. Readers are referred to the monograph on exact algorithms [8] for more details about the analysis of the size of the search tree.

To ease amortization on our analysis, we introduce "*shift*" σ for some recurrences which are not bottlenecks in our algorithm. Suppose that there are two branching operations A and B with recurrences $C(w) \leq C(w - t_{(A1)}) + C(w - t_{(A2)})$ and $C(w) \leq C(w - t_{(B1)}) + C(w - t_{(B2)})$, and that branching operation B is always applied to the subinstance G_1 generated by the first branch of A in the algorithm. The branching operation B may lead to a better recurrence (with smaller branching factor) than A does. To improve the recurrence for operation A, we will save some decreasing of the measure in operation B to A. Instead, we save $\sigma_B \geq 0$ from B by evaluating the branch rule B

with a worse recurrence

$$C(w) \leq C(w - (t_{(B1)} - \sigma_B)) + C(w - (t_{(B2)} - \sigma_B)).$$

The saved weight σ_B will be included to the recurrence for operation A to obtain

$$C(w) \leq C(w - (t_{(A1)} + \sigma_B)) + C(w - t_{(A2)}).$$

The amount of save is also called *shift*. In our algorithm, we introduce four shifts σ_i ($i = 1, 2, 3, 4$) in four recurrences: σ_1 for branching on good vertices or vertices with maximum degree $d \geq 6$ and σ_2 for branching on vertices with maximum degree $d = 5$ in Section 6.2.1; σ_3 for branching on a certain type of good funnels in Section 6.2.2; and σ_4 for branching on a certain type of optimal degree-4 vertices in Section 6.2.4.

Once the algorithm is designed, we can get a best value for each vertex weight by solving a quasiconvex program [5]. The quasiconvex program is generated by all the recurrences in the algorithm. In our algorithm, we will carefully select our branching rules to avoid some bad recurrences. This step is important to get further improvement. We need to check the graph and show that there is always a good structure, branching on which can get a good recurrence. We describe how to select the local structures and branching rules in the next subsection.

5.2 The Algorithm

We are ready to describe the detailed steps of our algorithm. Now we assume that the graph is a reduced graph and we begin to select our branching rules. Note that a reduce graph has minimum degree at least 3.

A subset $X \subseteq V$ is called *good* if $|N(X)| = 3$, $6 \leq |X| \leq 26$, X induces a connected subgraph and contains at least one vertex of degree ≥ 5, and X is maximal subject to these conditions. A vertex v is called *good* if it is a neighbor of a good subset $X \subseteq V$ (i.e., $v \in N(X)$). Our algorithm will first branch good vertices in the graph. We can see that when a good vertex is removed, we get some reducible vertex sets, which can be further reduced by applying reduction rules. This branching rule will be good enough in our analysis. After this, the algorithm will branch on a vertex v of degree ≥ 5 if it exits (recall that a high-degree vertex may be created by folding a degree-2 vertex and other reduction rules). Simply branching a vertex of maximum degree ≥ 5 can get a good recurrence since a vertex of high degree will have a large weight and a large number of neighbors. However, we assume that $N_2(v) \geq 4$ (since there is not good vertex now) and then get much better recurrences by deep analysis. In fact, we will save some shifts in this step. When the maximum degree of the graph is 3, we can also invoke previously known fast algorithms for MIS3 to solve the current instance. Thus we only need to consider the case where the maximum degree of the graph is 4.

Next, we assume the graph is a reduced graph with maximum degree 4. A triangle containing both degree-3 and degree-4 vertices is called *irregular*. When there is an irregular triangle (there is also a 3-funnel), we apply the rule of branching on a funnel to reduce them. The funnels to be selected first will be defined as "good funnels" (see the last paragraph of this subsection). Note that we do not deal with triangles containing only degree-4 vertices in a special way. When a degree-4 vertex v is contained in a 4-cycle, the algorithm will branch on an *optimal* 4-cycle (a 4-cycle containing a degree-4 vertex is called optimal if the number of degree-3 vertices in it is maximized among all such 4-cycles). When there is neither an irregular triangle nor a 4-cycle containing a degree-4 vertex, we have no good branch rules but simply branch on a degree-4 vertex. We will distinguish several cases by considering the degrees of the neighbors of the degree-4 vertex. It turns out that the worst branch is to branch on a degree-4 vertex with four degree-4 neighbors (not containing in any triangle or 4-cycle). We can select a degree-4 vertex with maximum number of degree-3 neighbors in the algorithm. But when the component of the graph is a 4-regular graph we may not be able to avoid this case. It can be regarded as one of the hardest cases, since there is no local structure. In this algorithm, we observe that after branching a degree-4 vertex, we may not get a 4-regular graph again (at least in the branch of removing this vertex). Then we can further branch with some good branches with shifts. This idea can help us to reduce this worst case and get further improvement. The degree-4 vertices to be selected first in the algorithm are called "optimal", which are defined as follows.

A *good funnel* is defined to be a funnel that satisfies one of the following: (i) a 4-funnel; (ii) a 3-funnel a-v-$\{b, c\}$ such that at least one of b and c is of degree 4 and $\delta(a) = 4$; (iii) when no funnel in (i) or (ii) exists, a 3-funnel a-v-$\{b, c\}$ such that both of b and c are of degree 4; and (iv) when no funnel in (i), (ii) or (iii) exists, a 3-funnel a-v-$\{b, c\}$ such that one of b and c is of degree 4.

Also a degree-4 vertex v is called *optimal* if v is not in two edge-disjoint triangles, and satisfies one of the following:
(a) v is adjacent to a degree-3 vertex in a triangle;
(b) no such vertices in (a) exist, and v is adjacent to a degree-3 vertex which has a degree-4 neighbor;
(c) no such vertices in (a) or (b) exist, v has a degree-4 neighbor u that is in a triangle disjoint with v but not in any other triangle, and the number of degree-3 neighbors of v is maximized; and
(d) no such vertices in (a), (b) or (c) exist, and the number of degree-3 neighbors of v is maximized.

Our algorithm for MIS4 is described in Figure 3.

Input: A graph G.
Output: The size of a maximum independent set in G.

1. Let $(G, s) := \mathrm{RG}(G, 0)$.
2. **If** $\{G$ has a vertex of degree $\geq 5\}$, pick up a good vertex v or a vertex v of maximum degree (when no good vertex exists), and **return** $s + \max\{\mathrm{MIS4}(G - v), |S_v| + \mathrm{MIS4}(G - N[S_v])\}$.
3. **Elseif** $\{G$ has a good funnel a-v-$(N(v) - \{a\})$, **return** $s + \max\{|S_a| + \mathrm{MIS4}(G - N[S_a]), 1 + \mathrm{MIS4}(G - N[v])\}$.
4. **Elseif**$\{G$ has a 4-cycle that contains a degree-4 vertex$\}$, select an optimal 4-cycle $abcd$, and **return** $s + \max\{\mathrm{MIS4}(G - \{a, c\}), \mathrm{MIS4}(G - \{b, d\})\}$.
5. **Elseif** $\{G$ has a degree-4 vertex$\}$, select an optimal degree-4 vertex v, and **return** $s + \max\{\mathrm{MIS4}(G - v), 1 + \mathrm{MIS4}(G - N[v])\}$.
6. **Else** $\{G$ is a 3-regular graph$\}$, use the algorithm for MIS3 in [18] to solve the problem, and **return** $s + \alpha(G)$.

Note: With a few modifications, the algorithm can deliver a maximum independent set.

Figure 3: Algorithm MIS4(G)

5.3 The Result

In our algorithm, we set vertex weight as follows

$$w_i = \begin{cases} 0 & \text{for } i = 0, 1 \text{ and } 2 \\ 0.6222 & \text{for } i = 3 \\ 1 & \text{for } i = 4 \\ 1.3937 & \text{for } i = 5 \\ 1.7715 & \text{for } i = 6 \\ w_6 + (i-6)(w_4 - w_3) & \text{for } i \geq 7. \end{cases}$$

Lemma 10 *With the above vertex weight setting, the recurrences generated by the algorithm in Figure 3 have a branching factor not greater than* 1.1376 *in average* (*with shifts*).

We derive a proof of this analytical lemma in Section 6. From the lemma we know that the size of the search tree generated by our algorithm is not greater than 1.1376^w, where w is not greater than the number n of vertices in the initial graph since it has maximum degree 4.

Theorem 11 *A maximum independent set in a degree-*4 *graph of* n *vertices can be found in* $O^*(1.1376^n)$ *time.*

6 Detailed Analysis

We still have two questions: why vertex weight is set as above and how Lemma 10 is proven. In fact, we predecide w_i for $i = 0, 1, 2$, and ≥ 7 as above, and determine the best values for the other weights w_3, w_4, w_5 and w_6, after generating all recurrences in the algorithm. When setting vertex weight, we also need to guarantee that the measure condition mentioned above will be satisfied.

6.1 Setting Weights

We set $w_i > 0$ for $i \geq 3$ and $w_0 = w_1 = w_2 = 0$. Then when measure w is 0, the problem can be solved in polynomial time, since the graph with $w = 0$ has only degree-0, degree-1 and degree-2 vertices and the maximum independent set problem can be solved in linear time. We also set

$$0 \leq w_3 \leq w_4 \leq 1 \tag{1}$$

so that a given degree-4 graph satisfies $0 \leq w \leq n$.

During an execution of our algorithm, a vertex with degree greater than 4 may be created by some reduction rules, but we set weights w_i, $i \geq 3$ so that the entire weight w never increases after any operation of reduction/branching rules. This is necessary to evaluate the time bound of our algorithm by analyzing how many instances will be generated until the measure becomes 0.

Let Δw_i denote $w_i - w_{i-1}$ for $i \geq 1$. We let $\Delta w_i \geq 0$ hold for each i, since the measure should not increase when the degree of a vertex decreases. To simplify some arguments for deriving all recurrences in the algorithm, we further assume that

$$0 \leq \Delta w_4 \leq \Delta w_i \leq \Delta w_3, \quad i \geq 5. \tag{2}$$

Many of the previous algorithms, such as the algorithm in [6], require that $\Delta w_i \leq \Delta w_j$ for $i \geq j$ to simply the argument and also to impose some upper/lower bounds on the vertex weight. In our algorithm, we rather require $\Delta w_4 \leq \Delta w_3$, which allows us to find a better vertex weight setting. With Assumption (2), we see that the measure will decrease by at least Δw_4 when the degree of a vertex v decreases by 1, where v is of degree at least 3.

We set the vertex weight of a vertex of degree ≥ 7 to be

$$w_i = w_6 + (i-6)\Delta w_4, \quad i \geq 7.$$

We only need to assign the value to w_3, w_4, w_5 and w_6, which will decide the value of w_i for all other i's. Next we will introduce several conditions on weights w_3, w_4, w_5 and w_6, where some conditions are necessary ones for analysis using measure while the others simplify our analysis.

One of the most important operations in the algorithm is folding a degree-2 vertex. When we fold a degree-2 vertex with two neighbors of degree i and j respectively, we delete three vertices with degree 2, i and j respectively and create a new vertex of degree $i+j-2$. So we need to have $w_2 + w_i + w_j \ (= w_i + w_j) \geq w_{i+j-2}$ for all $i, j \geq 1$. To get this, we assume that

$$w_i + w_j \geq w_{i+j-2}, \quad 3 \leq i, j \leq 5. \tag{3}$$

Under the assumption, we can prove

Lemma 12 *$w_i + w_j \geq w_{i+j-2}$ holds for all $i, j \geq 1$.*

Proof. If one of i and j, say i is at most 2, then $w_i + w_j = w_i \geq w_{i+j-2}$. Let $i, j \geq 3$. For $3 \leq i, j \leq 5$, we have $w_i + w_j \geq w_{i+j-2}$ by (3). Finally consider the case when at least one of i and j, say i, is greater than 5. Then we have that $w_{i+j-2} = w_i + (j-2)(w_4 - w_3)$ by the definition of w_i $(i \geq 7)$. Since $w_j \geq (j-2)(w_4 - w_3)$, this implies $w_{i+j-2} = w_i + (j-2)(w_4 - w_3) \leq w_i + w_j$. ∎

To simplify our analysis, we also assume that

$$w_4 + w_3 \geq 4 \max\{\Delta w_5, \Delta w_6\}. \tag{4}$$

This and $w_0 = w_1 = w_2 = 0$ imply

$$w_i \geq (i-2)(w_4 - w_3), \quad i \geq 0.$$

Now we can prove the following lemma, which implies that measure condition (ii) holds.

Lemma 13 *The measure never increases after applying the reduction operation in any step of RG. In particular, the following holds:*
(i) *Each application of the reduction operation in Step 1, 3, 4 or 5 of RG decreases the measure by at least $w_4 + 2\Delta w_4$ when the minimum degree of G is at least 3; and*
(ii) *Each application of the reduction operation in Step 6 of RG decreases the measure by at least $(\ell+1)\Delta w_4$ when the degree of ℓ vertices of maximum degree ≥ 5 decreases (possibly $\ell = 0$).*

Proof. Steps 1, 3 and 4: It is clear that the reduction operation in Steps 1, 3 and 4 in RG(G, s) removes some vertices without increase the degree of any other vertex, and this never increases the total weight w of the graph since weights are monotone increasing with respect to the degree. When the minimum degree of G is at least 3, each component contains at least four vertices, and removing a vertex v in Step 1 or 3 decreases the weight of v and its neighbors by at least $w_3 + 3\min_i \Delta w_i \geq w_4 + 2\Delta w_4$ by (2). In Step 4, any reducible vertex set V_1 with $|N(V_1)| = 1$ (resp., $|N(V_1)| = 2$) contains at least three (resp., two) vertices. If $|V_1| \geq 3$, then the measure decreases by $3w_3$, where $3w_3 \geq w_4 + 2\Delta w_4$ by (5). On the other hand if $|V_1| = 2$ then $|N(V_1)| = 2$ and each vertex $z \in N(V_1)$ is adjacent to both vertices in V_1, indicating that the degree of each vertex $z \in N(V_1)$ decreases by at least one in $G^\star$ and that the measure decreases by $2w_3 + 2\Delta w_4 = 2w_4$, where $2w_4 \geq w_4 + 2\Delta w_4$ by (2).

Step 2: Folding a degree-2 vertex in Step 2 of RG(G, s) is to delete a degree-2 vertex v and contract the two nonadjacent neighbors a and b of v into a new vertex s, i.e., it deletes two vertices of degree x and y from the graph and introduces a new vertex of degree at most $x+y-2$. By Lemma 12, we have $w_x + w_y \geq w_{x+y-2}$, and the measure does not increase.

Step 5: After Step 2 of $\mathrm{RG}(G, s)$, the minimum degree of G is at least 3. We consider folding a 2-3 structure $\{v, u\}$-$\{a, b, c\}$ with $\delta(v) = \delta(u) = 3$ in Step 5 of $\mathrm{RG}(G, s)$ (folding a 3-4 structure can be treated analogously). In Case (a), vertex v will be removed decreasing the measure by at least $w_4 + 2\Delta w_4$ as in Step 2. In Case (b), $\{v, u\}$ will be deleted and $\{a, b, c\}$ will be contracted. By noting that $w_{\delta(a)} + w_{\delta(b)} + w_{\delta(c)} \geq w_{\delta(a)+\delta(b)-2} + w_{\delta(c)} \geq w_{\delta(a)+\delta(b)+\delta(c)-4}$ by Lemma 12, we see that where contracting $\{a, b, c\}$ decreases the measure by at least $w_{\delta(a)+\delta(b)+\delta(c)-4} - w_{\delta(a)+\delta(b)+\delta(c)-6} \geq 2\min_i \Delta w_i \geq 2\Delta w_4$. In total the measure decreases by $2w_3 + 2\Delta w_4 \geq w_4 + 2\Delta w_4$ by (2).

Step 6: Folding a short funnel a-v-$\{b, c\}$ in Step 6 of $\mathrm{RG}(G, s)$ is to delete vertices $\{v, a\}$ with $\delta(v) = 3$ after letting every two vertices between $N(v)$ and $N(a)$ adjacent.

(1) First consider the case of $\delta(a) = 3$. Let $N(a) = \{v, t_1, t_2\}$, where t_1 is adjacent to b or c, say b. Then the weight of vertices v and a decreases by $2w_3$ in total, while the weight of vertices in $\{b, c, t_1, t_2\}$ increases by at most $\Delta w_{\delta(t_2)+1} + \Delta w_{\delta(c)+1} \leq 2\max\{\Delta w_5, \Delta w_6\}$ in total (note that $\delta(t_2), \delta(c) \geq 3$ since the minimum degree in G is at least 3). Hence the measure decreases by at least $2w_3 - 2\max\{\Delta w_4, \Delta w_5, \Delta w_6\} \geq \Delta w_4$ by (4). Now assume that the degree of ℓ vertices of maximum degree ≥ 5 decreases in the resulting graph $G^\star(v)$. Since $\delta(v) \leq \delta(a) < 5$, only vertices in $\{b, c, t_1, t_2\}$ can be the ℓ vertices of maximum degree ≥ 5, where the degree of each of them decreases by no more than one. Let $x \in \{b, c, t_1, t_2\}$ be one of the ℓ vertices of maximum degree ≥ 5. In the above analysis, the weight decrease from vertex x is not included. Hence the measure further decreases by at least $\ell \min_i \Delta w_i = \ell \Delta w_4$.

(2) Next let $\delta(a) = 4$ and $N(a) = \{v, t_1, t_2, t_3\}$, where there are at least two edges between $\{t_1, t_2, t_3\}$ and $\{b, c\}$. See Fig. 1(b). Without loss of generality that t_1 and b are adjacent, and t_2 is adjacent to b or c. We consider the case where t_2 is adjacent to c (the other case where t_2 is adjacent to b can be treated analogously). Then the weight of vertices v and a decreases by $w_4 + w_3$ in total, while the weight of vertices in $\{b, c, t_1, t_2\}$ increases by at most $\Delta w_{\delta(t_3)+1} + \Delta w_{\delta(b)+1} + \Delta w_{\delta(c)+1} \leq 3\max\{\Delta w_4, \Delta w_5, \Delta w_6\}$ in total. Hence the measure decreases by at least $w_4 + w_3 - 3\max\{\Delta w_4, \Delta w_5, \Delta w_6\} \geq \Delta w_6$ by (4). When the degree of ℓ vertices of maximum degree ≥ 5 decreases in $G^\star(v)$, we can show that the measure decreases by $(\ell+1)\Delta w_4$ in total. in the same manner of (1).

We have considered all cases finishing the proof. ∎

To evaluate the weight decrease in folding degree-2 vertices, we define

$$\beta = \min_{3 \leq i,j \leq 4} w_i + w_j - w_{i+j-2}.$$

Lemma 14 *Let v be a degree-2 vertex with two nonadjacent neighbors a and b such that $3 \leq \delta(a), \delta(b) \leq 4$. Then folding a degree-2 vertex v decreases the measure w by at least β.*

Proof. Contracting a and b into a new vertex decreases w by at least $w_{\delta(a)} + w_{\delta(b)} - w_{\delta(a)+\delta(b)-2} \geq \beta$. ∎

To simplify our analysis, we further assume that

$$1.5(w_4 - w_3) \leq w_3 \leq 2(w_4 - w_3). \tag{5}$$

Recall that $w_0 = w_1 = w_2 = 0$, $\Delta w_i \geq w_4 - w_3$, $3 \leq i \leq 4$, and $w_3 \leq 2(w_4 - w_3)$ by (5). Let X be a subset of vertices in a reduce graph $G = (V, E)$ and p be the number of edges between X and $V - X$. When we remove X, the total weight in the remaining set $V - X$ decreases by at least

$$kw_3 + (w_4 - w_3)\epsilon \tag{6}$$

for the integers k with $p = 3k + i$ ($i \in \{-1, 0, 1\}$) and ϵ such that $\epsilon = 1$ when $i = 1$, or $\epsilon = 0$ otherwise. In addition, if no degree-0 vertex is created in $G[V - X]$, then the total weight in the remaining set $V - X$ decreases by at least

$$k'w_3 + (w_4 - w_3)r \tag{7}$$

for the integers k' and $r \in \{0, 1\}$ such that $p = 2k' + r$. In our analysis, we also use the following properties on a degree-3 vertex v in a reduced graph: (i) Removing $N[v]$ creates no degree-0 vertex u (otherwise $\{v, u\}$-$N(v)$ would be a 2-3 structure); and (ii) If there is no edge between any two neighbors of degree-3 vertex v, then $|N_2(v)| \geq 4$ (otherwise $N(v)$-$N_2(v) \cup \{v\}$ would be a 3-4 structure).

The above analysis is frequently used in the next subsection to get a lower bound on the decrease of the measure or to eliminate some redundant case analysis. In the next subsection, we carefully check each branching step in the algorithm in Figure 3 and list out all possible recurrences.

6.2 Generating All Recurrences

6.2.1 Branches in Step 2

After Step 1, the graph is a reduced graph where the minimum degree is at least 3. In Step 2, the algorithm will branch on a vertex v of maximum degree by excluding it from the independent set or including it to the independent set. In the first branch, we will delete v from the graph. In the second branch, we will delete $N[S_v]$ from the graph. We use $\Delta_{out}(v)$ and $\Delta_{in}(v)$ to denote the decrease of the measure of w in the corresponding two branchings, respectively. Recall that $C(w)$ denotes the worst-case size of the search tree when the parameter of the graph is w. Then we obtain recurrence $C(w) = C(w - \Delta_{out}(v)) + C(w - \Delta_{in}(v))$ in this branch.

To analyze how much w decreases at least in each branch, we consider three cases. (i) v is a good vertex: Let $X \subseteq V$ be the subset with $v \in N(X)$ and $6 \leq |X| \leq 26$ such that X contains at least one vertex of degree ≥ 5, where if $|X| < 26$ then $|N(v) - X| \geq 2$ since X is maximal. Hence in the branch of deleting v, the measure decreases by at least $\Delta_{out}(v) \geq |X|w_3 + w_5 + |N(v) - X|\Delta w_4 \geq 6w_3 + w_5 + 2\Delta w_4$. In the other branch of deleting $N[S_v]$, the vertices in $X \cup N(v)$ will be removed and the measure decreases by at least $\Delta_{in}(v) \geq (|X \cup N(v)| - 1)w_3 + w_5 \geq 8w_3 + w_5$. Therefore we obtain recurrence

$$C(w) \leq C(w - (6w_3 + w_5 + 2\Delta w_4 - \sigma_1)) + C(w - (8w_3 + w_5 - \sigma_1)), \tag{8}$$

where we introduce shift σ_1 for branching on degree-4 vertices in Step 5.

We use the next lemma to obtain a lower bound on $\Delta_{in}(v)$ when a vertex v of maximum degree is picked up.

Lemma 15 *Let G be a reduced graph, and let v be a vertex of maximum degree $d \geq 5$. Then deleting $N[S_v]$ decreases the weight of vertices in $V - N[v]$ by at least $2w_4 + 2\Delta w_4$.*

Proof. Let p denote the number of edges between $N[v]$ and $N_2(v)$. Now $p \geq d = |N(v)|$ since each vertex $u \in N(v)$ is adjacent to a vertex in $N_2(v)$ (otherwise u would dominate v). If $p \geq 13$, then deleting $N[S_v]$ decreases the weight of vertices in $N_2(v)$ by at least $4w_3 + \Delta w_4$ by (6), where $4w_3 + \Delta w_4 \geq 2w_4 + 2\Delta w_4$ holds by (5).

Now assume that $p \leq 12$. Then $d \leq 12$. Let $N_2^+(v)$ be the set of vertices in $N_2(v)$ that have a neighbor in $V - N[v] - N_2(v)$. Now $|N_2^+(v)| \geq 4$ holds, since otherwise $X = N[v] \cup (N_2(v) - N_2^+(v))$ would be a reducible vertex set with $|N(X)| = |N_2^+(v)| = 3$ and $|X| \leq 1 + d + p \leq 25$ where $|N_2^+(v)| \geq 3$ (otherwise $X = N[v] \cup (N_2(v) - N_2^+(v))$ would be a reducible vertex set). Now G contains at least 24 vertices by definition of reduced graphs. Hence $N_2(v)$ contains at least

$|N_2^+(v)| \geq 4$ vertices that cannot be degree-0 vertices after removing edges between $N[v]$ and $N_2(v)$. Also if $|N_2^+(v)| = 4$, then for any two vertices $s, s' \in N_2^+(v)$, $N(\{s, s'\})$ contains at least two vertices in $V - (N[v] \cup N_2(v))$, since otherwise $X' = X \cup \{s, s'\}$ would be a reducible vertex set with $|X'| \leq 1 + d + p - 3 + 1 \leq 23$.

This means that if $p \geq 10$ then the weight of the vertices in $N_2(v)$ decreases by at least $2w_3 + 4\Delta w_4$ $(= 2w_4 + 2\Delta w_4)$. Assume that $p \leq 9$. Since $d \geq 5$, this implies that v has at least one satellite; i.e., $S_v - v \neq \emptyset$. Also if a vertex $t \in N_2^+(v)$ becomes a degree-1 vertex in $G - N[v]$, then we can fold the degree-1 vertex deleting a neighbor of t in $G - N[v] - N_2(v)$. In what follows, we mainly consider the case where no vertex $t \in N_2^+(v) \cap N[S_v - v]$ becomes a degree-1 vertex in $G - N[S_v]$; if there are p_1 edges between $N(v)$ and $N_2(v) - N[S_v - v]$ and there are p_2 vertices in $N_2^+(v) - -N[S_v - v]$, then deleting the p_1 edges decreases the weight of vertices in $N_2(v) - N[S_v - v]$ by at least $\lfloor (p_1 - p_2)/3 \rfloor w_3 + p_2 \Delta w_4$.

First consider the case where there are at least two vertices $s, s' \in N[S_v - v] \cap N_2(v)$. Let k be the number vertices in $\{s, s'\}$ that are contained in $N_2^+(v)$. We here claim that $N(\{s, s'\}) \cap N_2^+(v)$ and $N_2^+(v) - N[S_v - v]$ contain at least four vertices, say z_1, z_2, z_3 and z_4 in total. When $k = 0, 1$ or $|N_2^+(v)| \geq 5$, the claim holds since $|N_2^+(v)| \geq 4$ and any vertex in $N(\{s, s'\}) \cap N_2^+(v)$ has a neighbor in $V - N[v] - N_2(v)$. When $k = 2$ and $|N_2^+(v)| = 4$, as we have observed, $N(\{s, s'\})$ contains at least two vertices in $V - (N[v] \cup N_2(v))$, proving the claim. Hence removing $N[S_v]$ decreases the weight of the vertices in $\{s, s'\}$ and $\{z_1, z_2, z_3, z_4\}$ by at least $2w_3 + 4\Delta w_4$.

Now consider the other case; i.e., there is only one vertex $s \in N[S_v - v] \cap N_2(v)$; i.e., $S_v - v$ contains only one satellite s and no vertex in $N_2(v) - s$ is adjacent to s. Let h denote the number of edges between s and $N(v)$, where $1 \leq h \leq \delta(s) \leq d$. If $s \in N_2^+(v)$, then s is adjacent to a vertex $z \in V - N[v] - N_2(v)$ and let $S = \{s, z\}$; let $S = \{s\}$ otherwise. Then $S \cup N_2^+(v)$ contains at least four vertices other than s. We distinguish subcases according to h.
For $h \geq 5$, the weight of $S \cup N_2^+(v)$ decreases by at least $w_5 + 4\Delta w_4$, where $w_5 + 4\Delta w_4 \geq 2w_4 + 2\Delta w_4$ holds since $\Delta w_5 + \Delta w_4 \geq w_3$ by (2) and (5).
For $h = 4$, any vertex $u \in N(v) - N(s)$ has at least two neighbors in $N_2^+(v)$ (since v has no other satellite than s) and if $|N(v) - N(s)| = 1$ (resp., $|N(v) - N(s)| = 2$), then at least three (resp., two) vertices in $N_2(v) - \{s\}$ are adjacent to vertices in $N(v) \cap N(s)$ (otherwise $N[v] \cup \{s\}$ would contain a reducible vertex set). This indicates that there are at least five edges between $N(v)$ and $N_2(v) - \{s\}$, and the weight of vertices in $S \cup N_2^+(v)$ or in $V - N[v]$ decreases by at least $w_4 + 5\Delta w_4$ $(\geq 2w_4 + 2\Delta w_4)$ by (5) (recall that if a degree-1 vertex created in $N_2^+(v)$ then folding it delete a vertex in $N - N[v] - N_2(v)$ decreases the measure more).
For $h = 3$, any vertex $u \in N(v) - N(s)$ has at least two neighbors in $N_2^+(v)$, and $|N(v) - N(s)| \geq 2$, and if $|N(v) - N(s)| = 2$ then at least two vertices in $N_2(v) - \{s\}$ are adjacent to vertices in $N(v) \cap N(s)$, and there are at least six edges between $N(v)$ and $N_2(v) - \{s\}$. Then when $\delta(s) = 3$ (resp., $\delta(s) \geq 4$), the weight of vertices in $S \cup N_2^+(v)$ or in $V - N[v]$ decreases by at least $w_3 + w_3 + 4\Delta w_4$ $(\geq 2w_4 + 2\Delta w_4)$ by (5) (resp., $w_4 + w_3 + 3\Delta w_4$ $(= 2w_4 + 2\Delta w_4)$).
For $h = 1, 2$ ($s \in N^+(s)$ and $S = \{s, z\}$), each vertex $u \in N(v) - N(s)$ has at least two neighbors in $N_2^+(v)$, and $|N(v) - N(s)| \geq 3$, and and there are at least six edges between $N(v)$ and $N_2(v) - \{s\}$. Then the weight of vertices in $S \cup N_2^+(v)$ or in $V - N[v]$ decreases by at least $2w_3 + w_3 + 3\Delta w_4$ $(\geq 2w_4 + 2\Delta w_4)$ by (5). ∎

Let d_i denote the number of degree-i neighbors of v, where $d = \sum_{i=3}^{d} d_i$. For the first branch of deleting v from G, we have

$$\Delta_{out}(v) = w_d + \sum_{i=3}^{d} d_i \Delta w_i.$$

In the second branch where v is included to the independent set, we will remove all vertices in $N[v]$, which decreases the degree of the vertices in $N_2(v)$. By Lemma 15, deleting $N[v]$ decreases the weight of vertices in $V - N[v]$ is at least $2w_4 + \Delta w_4$, and we get

$$\Delta_{in}(v) \geq w_d + \sum_{i=3}^{d} d_i w_i + 2w_4 + \Delta w_4.$$

(ii) v is a vertex of maximum degree $d \geq 6$: Using $\Delta w_4 \leq \Delta w_i$ and $w_3 + \Delta w_3 \leq w_i + \Delta w_i$ ($i \geq 3$) by (2) and (5) and the property that $C(w-(a+b)) + C(w-(a+c)) \leq C(w-(a+b-\epsilon)) + C(w-(a+c+\epsilon))$ for $0 \leq a$, $0 \leq b \leq c$, and $0 \leq \epsilon \leq a+b$ (cf. [8]), we get the following recurrence

$$\begin{aligned}
C(w) &= C(w-\Delta_{out}(v)) + C(w-\Delta_{in}(v)) \\
&\leq C(w-(w_d + \textstyle\sum_{i=3}^{d} d_i \Delta w_i)) + C(w-(w_d + \sum_{i=3}^{d} d_i w_i + 2w_4 + 2\Delta w_4)) \\
&\leq C(w-(w_d + \textstyle\sum_{i=3}^{d} d_i \Delta w_4)) + C(w-(w_d + \sum_{i=3}^{d} d_i (w_i + \Delta w_i - \Delta w_4) + 2w_4 + 2\Delta w_4)) \\
&\leq C(w-(w_d + \textstyle\sum_{i=3}^{d} d_i \Delta w_4)) + C(w-(w_d + \sum_{i=3}^{d} d_i (2w_3 - \Delta w_4) + 2w_4 + 2\Delta w_4)) \\
&\leq C(w-(w_6 + 6\Delta w_4)) + C(w-(w_6 + 6(2w_3 - \Delta w_4) + 2w_4 + 2\Delta w_4)) \\
&= C(w-(w_6 + 6w_4 - 6w_3)) + C(w-(w_6 - 2w_4 + 16w_3)).
\end{aligned}$$

By introducing shift σ_1, we have

$$C(w) \leq C(w-(w_6 + 6w_4 - 6w_3 - \sigma_1)) + C(w-(w_6 - 2w_4 + 16w_3 - \sigma_1)). \tag{9}$$

(iii) v is a vertex of maximum degree $d = 5$: Analogously with (ii), we get the recurrence with shift σ_2 ($\leq \min\{\sigma_1, \Delta w_4\}$) for $d = 5$

$$\begin{aligned}
C(w) \leq\ & C(w-(w_5 + d_5(w_5 - w_4) + d_4(w_4 - w_3) + d_3 w_3 - \sigma_2)) \\
& + C(w-(w_5 + d_5 w_5 + d_4 w_4 + d_3 w_3 + 2w_4 + 2\Delta w_4 - \sigma_2)) \\
& \text{for } 0 \leq d_3, d_4, d_5 \leq 5 \text{ with } d_5 + d_4 + d_3 = 5.
\end{aligned} \tag{10}$$

6.2.2 Branches in Step 3

In this step, the graph is a reduced graph with maximum degree 4 where there is no short funnel and each vertex is of degree 3 or 4. If the graph contains an irregular triangle, then there exists a good funnel a-v-$(N(v) - \{a\})$, on which the algorithm will branch by removing either $N[S_a]$ or $N[v]$.

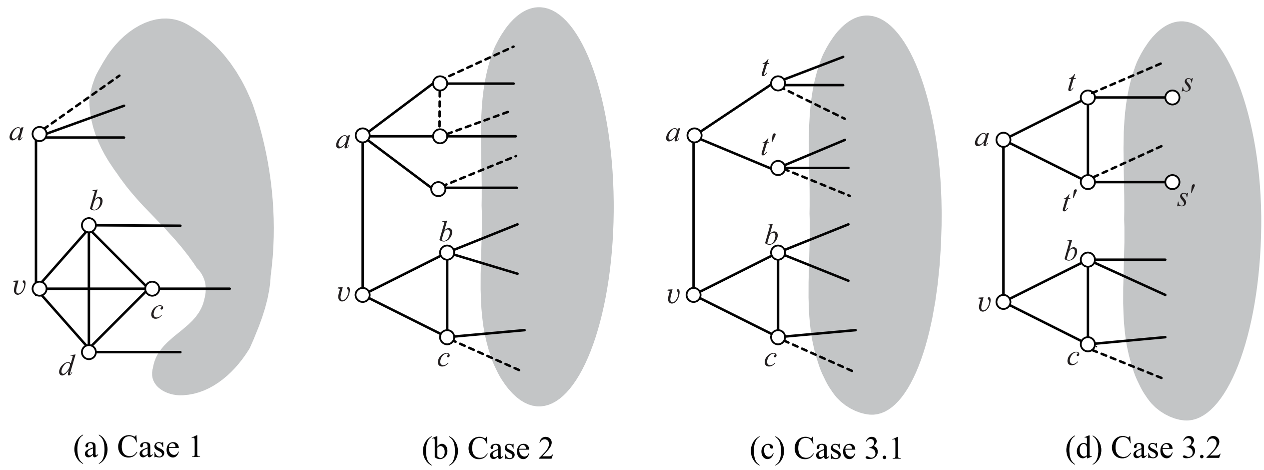

Figure 4: Illustrations of branching on good funnels

Case 1. The good funnel is a 4-funnel a-v-$\{b, c, d\}$ (see Fig. 4(a)): Vertex a is not adjacent to any vertex $t \in \{b, c, d\}$ since otherwise v would dominate t. We first consider the branch of removing

$N[v]$. Note that $N(\{b,c,d\})-\{v\}$ contains at least three vertices, since otherwise a vertex dominate would some other vertex in $\{b,c,d\}$. Hence there are at least five edges between $N[v]$ and $V-N[v]$, and no three of them meet at the same vertex since $|N(\{b,c,d\})-\{v\}|\geq 3$. This means that removing $N[v]$ does not create degree-0 vertices and decreases the total weight in $V-N[v]$ by at least $2w_3+\Delta w_4$ by (7), and the total weight in $N[v]$ by at least w_3+4w_4. Totally the measure w decreases by at least $5w_4+2w_3$. We next consider the other branch of removing $N[S_a]$. Note that $v\in N(a)$ and there are at least five edges between $N(a)$ and $N_2(a)$. Removing $N[a]$ decreases the total weight of vertices in $N[a]$ by at least $3w_3+w_4$, that in $\{b,c,d\}$ by $3(w_4-w_3)$ (at this point the degree of each vertex in $\{b,c,d\}$ is 3), and that in some vertices in $N_2(a)-\{b,c,d\}$ by at least $2(w_4-w_3)$. Note that $N[a]\subseteq N[S_a]$. In the branching of removing $N[S_a]$, the measure w decreases by at least $6w_4-2w_3$ in total. In Case 1, we get recurrence

$$C(w)\leq C(w-(5w_4+2w_3))+C(w-(6w_4-2w_3)). \tag{11}$$

In the rest of cases, let a-v-$\{b,c\}$ be a good 3-funnel. Note that a is not adjacent to any of b and c, otherwise v would dominate b or c. Assume without loss of generality that $\delta(b)\geq\delta(c)$. Then $\delta(b)\geq 4$. Let p_a (resp., p_v) be the number of edges between $N[a]$ and $N_2(a)$ (resp., $N[v]$ and $N_2(v)$). We distinguish three cases.

Case 2. $\delta(a)=4$ (see Fig. 4(b)): Note that b and c have no common neighbor in $N_2(v)$ (otherwise c would dominate b). Let k be the number of degree-3 vertices in $N(a)-\{v\}$. First we look at the branch where $N[S_a]$ is removed. In this branch at least the vertices in $N[a]$ ($\subseteq N[S_a]$) is removed. Each neighbor $t\in N(a)$ of vertex a is adjacent to a vertex in $N_2(a)$, otherwise t would dominate a. Hence $p_a\geq 5$ and $|N_2(a)|\geq 4$ (otherwise $N(a)-\{v\}$ would contain a dominated vertex or a-v-$\{b,c\}$ would be a short funnel). Removing $N[a]$ decreases the weights of vertices in $N[a]$ and $N_2(a)$ by at least $w_4+4w_3+(3-k)\Delta w_4=(1+3-k)w_4+(1+k)w_3$ and $w_3+3\Delta w_4$ (since $p_a\geq 5$ and $|N_2(a)|\geq 4$), respectively. Then this branch decreases w by at least $(4-k)w_4+(1+k)w_3+w_3+3\Delta w_4=7w_4-w_3-k(w_4-w_3)$. We also analyze a special case where a is not contained in any 4-cycle. Since a is not in a 4-cycle, there is at most one edge joining vertices $N(a)-\{v\}$ and there are at least four edges between $N(a)-\{v\}$ and $N_2(a)$. Also no two of the four edges meet at the same vertex in $N_2(a)$ or none of them is adjacen to b or c, indicating that $p_a\geq 6$ and $|N_2(a)|\geq 6$. Then this branch decreases w by at least $w_4+4w_3+(3-k)\Delta w_4+6\Delta w_4=10w_4-5w_3-k(w_4-w_3)$.

For the other branch where $N[v]$ is removed, we consider two cases: the degree of c is 3 or 4. Note that there is at most one edge between $\{b,c\}$ and $N(a)-\{v\}$ (otherwise a-v-$\{b,c\}$ would be a short funnel). When c is a degree-3 vertex, there are $p_v\geq 6$ edges between $N(v)$ and $N_2(v)$. Hence it is impossible to create a degree-0 vertex after removing $N[v]$. Then removing $N[v]$ decreases the weight of vertices in $N_2(v)$ by at least $kw_3+(3-k)\Delta w_4+2\Delta w_4=5(w_4-w_3)+k(2w_3-w_4)$. Then this branch decreases w by at least $2w_4+2w_3+5(w_4-w_3)+k(2w_3-w_4)=7w_4-3w_3+k(2w_3-w_4)$ in total. When c is a degree-4 vertex, there are $p_v\geq 7$ edges between $N(v)$ and $N_2(v)$. Note that b and c can have at most one common neighbor in $N_2(v)$, otherwise c would dominate b. Hence it is impossible to create two degree-0 vertices after removing $N[v]$. Then removing $N[v]$ decreases the total weight in $N_2(v)$ by at least $kw_3+(3-k)\Delta w_4+w_3+\Delta w_4=4w_4-3w_3+k(2w_3-w_4)$. Totally this branch decreases w by at least $3w_4+w_3+4w_4-3w_3+k(2w_3-w_4)=7w_4-2w_3+k(2w_3-w_4)>7w_4-3w_3+k(2w_3-w_4)$.

In Case 2, we can always branch with

$$\begin{aligned} C(w) &\leq C(w-(7w_4-w_3-k(w_4-w_3))) \\ &\quad +C(w-(7w_4-3w_3+k(2w_3-w_4)))\text{ for } k=0,1,2,3. \end{aligned} \tag{12}$$

Furthermore, if a is not contained in any 4-cycle and both of b and c are degree-4 vertices, we can branch with

$$\begin{aligned} C(w) &\leq C(w-(10w_4-5w_3-k(w_4-w_3)-\sigma_3)) \\ &\quad +C(w-(7w_4-2w_3+k(2w_3-w_4)-\sigma_3))\text{ for } k=0,1,2,3, \end{aligned} \tag{13}$$

where we introduce shift σ_3. The special case (13) is covered by (12). But it will be used in the analysis later.

Case 3. $\delta(a) = 3$ (see Fig. 4(c) and (d)): Let $N(a) = \{v, t, t'\}$, where $\delta(t) \le \delta(t')$ is assumed without loss of generality. Let $0 \le f_1 \le 2$ be the number of degree-4 neighbors of vertex a and $1 \le f_2 \le 2$ be the number of degree-4 vertices in $\{b, c\}$.

First, we look at the first branch where $N[v]$ is removed. This decreases the total weight of vertices in $\{a, v\}$ by $2w_3$ and that in $\{b, c\}$ by $f_2w_4 + (2 - f_2)w_3$. Next we analyze how much weight in $N_2(v)$ decreases after removing $N[v]$. There are $4 + f_2$ edges between $N[v]$ and $N_2(v)$. We consider two cases on the first branch: the degree of c is 3 or 4.

(1-i) c is a degree-3 vertex (now $f_2 = 1$): Let c' ($\neq v, b$) be the third neighbor of c. Then $c' \neq t, t'$ (otherwise b would be a roof) and the degree of c' is 3 (otherwise c'-c-$\{v, b\}$ would be a good funnel of Case (ii)). Let b' and b'' be the third and fourth neighbors of b. Note that $b', b'' \neq c'$ otherwise c would dominate b. Also $\{b', b'', c'\} \cap \{t, t'\} = \emptyset$, otherwise a-v-$\{b, c\}$ would be a short funnel with $\delta(a) = 3$. Then the weight of vertices in $N_2(v)$ decreases by at least $W_{N_2}(f_1) = f_1\Delta w_4 + (2 - f_1)w_3 + w_3 + 2\Delta w_4 = (2 + f_1)w_4 + (1 - 2f_1)w_3$ after removing $N[v]$. We show that when $f_1 = 0$ and a is not in a triangle, the weight in $V - N[v]$ further decreases by at least β ($\le w_3 - \Delta w_4$). If one of b' and b'' is a degree 3 vertex, then the weight of the vertex decreases by w_3 instead of Δw_4 in $W_{N_2}(f_1)$. On the other hand, if both b' and b'' are of degree 3, then the graph $G - N[v]$ has exactly three degree-2 vertices, t, t' and c', and one of t and t', say t is not adjacent to c', indicating that folding the degree-2 vertex t (or some other reduction) decreases the measure by at least $\min\{\beta, \Delta w_4\} = \beta$ by Lemma 13. Then in Case (1-i) removing $N[v]$ decreases the weight in $V - N[v]$ by at least $W_{N_2}(f_1) + \epsilon\beta$, where $\varepsilon = 1$ if $f_1 = 0$ and a is not in a triangle, and $\varepsilon = 0$ otherwise.

(1-ii) c is a degree-4 vertex (now $f_2 = 2$): Let c'' be the fourth neighbor of c. Note that $\{c', c''\} \neq \{b', b''\}$ since otherwise c would dominate b. Also $\{t, t'\} \cap \{c', c'', b', b''\} = \emptyset$, otherwise a-v-$\{b, c\}$ would be a short funnel with $\delta(a) = 3$. Then the weight of vertices in $\{t, t'\}$ decreases by at least $f_1(w_4 - w_3) + (2 - f_1)w_3$ and that in $\{c', c''\} \cap \{b', b''\}$ by at least $2(w_4 - w_3) + w_3$. Then removing $N[v]$ decreases the total weight of vertices in $N_2(v)$ by at least $W_{N_2}(f_1)$.

Therefore, the first branch decreases the measure w by at least

$$\begin{aligned} W_1(f_1, f_2) &= 2w_3 + f_2w_4 + (2 - f_2)w_3 + W_{N_2}(f_1) \\ &= (2 + f_1 + f_2)w_4 + (5 - 2f_1 - f_2)w_3 + \varepsilon\beta, \end{aligned}$$

where $\varepsilon = 1$ if $f_1 = 0$ and a is not in a triangle, and $\varepsilon = 0$ otherwise.

For the second branch where $N[S_a]$ is removed, we consider two subcases.

Case 3.1. Vertex a is not in a triangle (see Fig. 4(c)): For this case, we analyze how much measure w will decrease by removing only $N[a]$ ($\subseteq N[S_a]$). Removing $N[a]$ decreases the weight of vertices in $\{a, v\}$ by $2w_3$ and that in $\{t, t'\}$ by $f_1w_4 + (2 - f_1)w_3$. We consider the weight decrease of vertices in $N_2(a)$. There are $6 + f_1$ edges between $N[a]$ and $N_2(a)$.

(2-i) No degree-1 vertex is created after removing $N[a]$: Then the weight of vertices in $N_2(a)$ decreases by at least $W' = (6 + f_1 - (2 - f_2))\Delta w_4 + (2 - f_2)w_3$ (note that there are at least $2 - f_2$ degree-3 vertices in $N_2(a)$). We show that when $f_2 = 1$, the weight of vertices in $V - N[a]$ further decreases by at least β ($\le w_3 - \Delta w_4$). By $f_2 = 1$, the degree of c' is 3 (otherwise c'-c-$\{v, b\}$ would be a good funnel of Case (ii)). If one of t and t', say t is adjacent to a degree-3 vertex in $N_2(a)$, then the weight of a neighbor of t decreases by w_3 instead of Δw_4 in W'. On the other hand, none of t and t' is adjacent to any degree-3 vertex in $N_2(a)$, then the graph $G - N[a]$ has exactly one degree-2 vertex c, and folding the degree-2 vertex c (or some other reduction) decreases the measure by at least $\min\{\beta, \Delta w_4\} = \beta$ by Lemma 13.

(2-ii) A degree-1 vertex v' is created after removing $N[a]$: Consider the case of $v' = c'$. In this case, c' is a degree-3 vertex adjacent to both t and t'. The weight decrease of neighbors in $N_2(a)$ of t and t' in W' is $2\Delta w_4 + w_3$ instead of $4\Delta w_4$. However, folding the degree-1 c, we can further decrease the measure by at least w_3 of the degree-3 vertex b. In total, we can further decrease the weight in $V - N[a]$ by $w_3 - 2\Delta w_4 + w_3$ ($\geq 2\beta$). Assume that $v' \neq c'$. Then the neighbor v'' of v' is a vertex of degree ≥ 3 in $G' = G - N[a]$, since otherwise v', v'' and a vertex in $\{t, t'\}$ (say t) will form a triangle and then t'-v'-$\{v'', t\}$ would be a short funnel in G. In this case, we can further decrease w by at least w_3 ($\geq \beta$) by removing the dominated vertex v''.

From (2-i) and (2-ii), the weight of vertices in $V - N[a]$ decreases by at least $W'_{N_2}(f_1, f_2) = W' + (2 - f_2)\beta = (4 + f_1 + f_2)w_4 - (2 + f_1 + 2f_2)w_3 + (2 - f_2)\beta$. Hence the second branch decreases w by at least

$$\begin{aligned} W_2(f_1, f_2) &= 2w_3 + f_1 w_4 + (2 - f_1)w_3 + W'_{N_2}(f_1, f_2) \\ &= (4 + 2f_1 + f_2)w_4 + (2 - 2f_1 - 2f_2)w_3 + (2 - f_2)\beta. \end{aligned}$$

We get recurrences:

$$C(w) \leq C(w - W_1(f_1, f_2)) + C(w - W_2(f_1, f_2)), \tag{14}$$

where $f_1 \in \{0, 1, 2\}$ and $f_2 \in \{1, 2\}$.

Case 3.2. Vertex a is in a triangle (see Fig. 4(d)): Note that now v-a-$\{t, t'\}$ is also a 3-funnel. By choice of good funnels, $f_2 \geq f_1$ holds. After removing $N[S_a] \supseteq N[a]$, by the above analysis (1-i) and (1-ii) of the branch where $N[v]$ is removed, we know that w will decrease by at least

$$W'_2(f_1, f_2) = W_1(f_2, f_1) = (2 + f_1 + f_2)w_4 + (5 - f_1 - 2f_2)w_3.$$

For $f_1 = 2$, $f_2 = 2$ holds and we get recurrence

$$C(w) \leq C(w - W_1(2, 2)) + C(w - W'_2(2, 2)). \tag{15}$$

However, when $f_1 \leq 1$ this is not good enough for our analysis. In fact, for this case vertex a has $2 - f_1$ satellites and we can show that the measure decreases by at least $2w_4 + 6w_3$. First consider the case of $f_1 = 0$ (i.e., $\delta(t) = \delta(t') = 3$). Let s and s' be the third neighbor of t and t', respectively. Note that s and s' are not adjacent otherwise vertex a would be a roof. There are at least four edges between $\{s, s'\}$ and $N(\{s, s'\}) - \{t, t'\}$. It is impossible to have $N(\{s, s'\}) - \{t, t'\} = \{b, c\}$ otherwise the component would contain only eight vertices. Let $x \in N(\{s, s'\})$ be a vertex different from t, t', b and c. If one of b and c (say b) is in $N(\{s, s'\})$, then after removing $N[S_a]$, the degree of c will decrease by at least 2. In this case, removing $N[S_a]$ decreases the weight of vertices in $\{v, a, t, t', s, s', x, b, c\}$ by at least $7w_3 + w_4 + w_3 = w_4 + 8w_3$ ($\geq 2w_4 + 6w_3$). If neither of b and c is in $N(\{s, s'\})$, then $N(\{s, s'\}) - \{t, t'\}$ contains at least two vertices x and x' different from t, t', b and c, and removing $N[S_a]$ decreases the weight of vertices in $\{v, a, t, t', s, s', x, x'\}$ by at least $8w_3$ and that in $\{b, c\}$ by $2(w_4 - w_3)$ in total w decreases by at least $8w_3 + 2(w_4 - w_3) = 2w_4 + 6w_3$. Hence for $f_1 = 0$, the second branch decreases w by at least $2w_4 + 6w_3$.

Next consider the case of $f_1 = 1$, where one of t and t', say t is a degree-3 vertex, whose neighbor $s \in N(t) - N(a)$ is satellite, where s and t' are not adjacent (otherwise t would dominate t'). If $\delta(s) = 4$, then $N[a] \cup N[s]$ contains eight vertices two of which are of degree 4, and the second branch decreases w by at least $2w_4 + 6w_3$. Let $\delta(s) = 3$, where s and t' have no common neighbor (otherwise s-t-$\{a, t''\}$ would be a short funnel). If $N(s) \cap \{b, c\} = \emptyset$, then the weight of the four vertices in $\{b, c\}$ and $N(t'') - \{a, t\}$ decreases by at least $4\Delta w_4$, implying that removing $N[a] \cup N[s]$ decreases the measure by at least $w_4 + 6w_3 + 4\Delta w_4 = 5w_4 + 2w_3$ ($\geq 2w_4 + 6w_3$). If $|N(s) \cap \{b, c\}| = 1$, then removing $N[a] \cup N[s]$ decreases the weight of b and c by at least $w_4 + w_3$, implying that removing $N[a] \cup N[s]$ decreases the measure by at least $w_4 + 5w_3 + w_4 + w_3 = 2w_4 + 6w_3$. Finally let $\{b, c\} \subseteq N(s)$, where both of b and c are degree-4 vertices (otherwise c would dominate b). There are four edges between $N[a] \cup N[s]$ and $V - N[a] \cup N[s]$. Then removing $N[a] \cup N[s]$ decreases the measure by at least $3w_4 + 4w_3 + w_3 + \Delta w_4 = 4w_4 + 4w_3$ ($\geq 2w_4 + 6w_3$).

In Case 3.2, we get recurrences:

$$C(w) \le C(w - W_1(f_1, f_2)) + C(w - (2w_4 + 6w_3)), \tag{16}$$

where $f_1 \in \{0, 1\}$ and $f_2 \in \{1, 2\}$.

Note that after Step 6, the graph has no irregular triangle.

6.2.3 Branches in Step 4

In this step, we will branch on optimal 4-cycles that contain at least one degree-4 vertex. Without loss of generality, we assume that the algorithm will branch on an optimal 4-cycle $abcd$, where a is a degree-4 vertex. Note that if there is a degree-3 vertex in the cycle, a and c (b and d) are not adjacent, otherwise there would be an irregular triangle. According to the branching rule, our algorithm will branch by removing either $\{a, c\}$ or $\{b, d\}$ from the graph. We distinguish the following five cases.

Case 1. The other vertices than a in the 4-cycle are of degree 3: We assume that a' and a'' are the third and fourth neighbors of a, b' is the third neighbor of b, c' is the third neighbor of c, and d' is the third neighbor of d, where possibly $c' \in \{a', a''\}$ (see Fig. 5(a) for an illustration). Note that $b' \neq d'$, otherwise $\{b, d\}$-$\{a, c, b' = d'\}$ would be a 2-3 structure. Also $\{a', a'', c'\} \cap \{b', d'\} = \emptyset$, otherwise there would be an irregular triangle or a roof. Let $\varepsilon = 1$ if at least one of b' and d' is of degree 4, and $\varepsilon = 0$ otherwise.

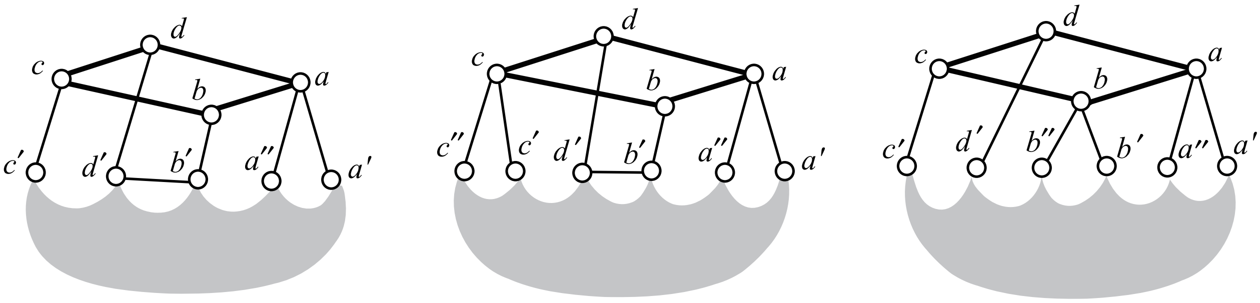

Figure 5: Branching on 4-cycles

In the branch where $\{a, c\}$ is removed, b and d will become degree-1 vertices. The algorithm will apply the reduction rules to eliminate degree-1 vertices immediately. Then b' and d' will be removed. Totally, at least six vertices a, b, c, d, b' and d' are removed from the graph. There are also at least five edges between $V' = \{a, b, c, d, b', d'\}$ and $V - V'$ (there may not be seven edges when b' and d' are adjacent). We consider how much weight of vertices in $V - V'$ decreases after removing V'. If $|N(V')| \geq 3$, then the weight in $V - V'$ decreases by at least $w_3 + 2(w_4 - w_3) = 2w_4 - w_3$. If $|N(V')| = 2$ (i.e., $N(V') = \{a', a''\}$), then each of a' and a'' is adjacent to a vertex in $\{c, b', d'\}$ (otherwise $b'd'a'$ or $b'd'a''$ would be an irregular triangle), and removing V' decrease the weight in $V - V'$ either by $w_{\delta(a')} + w_{\delta(a'')} \geq w_4 + w_3$, where one of a' and a'' is of degree 4 (otherwise V' would be contained in a component with size at most 14). Then in the branch of removing $\{a, c\}$ the measure w decreases by at least

$$w_4 + 5w_3 + \varepsilon \Delta w_4 + 2w_4 - w_3 = 3w_4 + 4w_3 + \varepsilon(w_4 - w_3).$$

In the other branch where $\{b, d\}$ is removed, c will become a degree-1 vertex and we will further remove c' from the graph. Thus the branch will remove $N[c]$. Let us see how much weight in $V - N[c]$

will decrease by removing $N[c]$. There are at least six edges between $N[c]$ and $V-N[c]$. Note that $|N_2(c)|\geq 3$ since $\{a,d,d'\}\subseteq N_2(c)$. Note that no two neighbors of c are adjacent. If $|N_2(c)|=3$, then $N(c)$-$N_2(c)\cup\{c\}$ would be a 3-4 structure. We know that $|N_2(c)|\geq 4$. No degree-0 vertex u is created after removing $N[c]$ otherwise $\{c,u\}$-$N(c)$ would be a 2-3 structure. If $|N_2(c)|\geq 5$, then the measure w decreases by at least $w_4+4\Delta w_4$ (w_4 from vertex a and $4\Delta w_4$ from the other four vertices in $N_2(c)$). Now let $|N_2(c)|=4$. If no degree-1 vertex is created after removing $N[c]$, then the weight in $N_2(c)$ still decreases by at least $w_4+4\Delta w_4+2(1-\varepsilon)(w_3-\Delta w_4)$. If a degree-1 vertex u is created, then the weight in $N_2(c)$ may only decrease by $w_4+2\Delta w_4+w_3+(1-\varepsilon)(w_3-\Delta w_4)$. Note that u is the unique degree-1 vertex in the graph, and we can further decrease w by at least Δw_4 by removing the dominated vertices adjacent to degree-1 vertices. Then for any case, the weight in $V-N[c]$ decreases by at least $w_4+4\Delta w_4+2(1-\varepsilon)(w_3-\Delta w_4)$. Totally, in the branch of removing $\{b,d\}$ the measure w decreases by at least

$$4w_3+w_4+4(w_4-w_3)+2(1-\varepsilon)(2w_3-w_4)=5w_4+(1-\varepsilon)(4w_3-2w_4).$$

In Case 1, we can always branch with the following recurrence

$$\begin{aligned} C(w) \quad \leq \quad & C(w-(3w_4+4w_3+\varepsilon(w_4-w_3))) \\ & +C(w-(5w_4+(1-\varepsilon)(4w_3-2w_4))) \text{ for } \varepsilon\in\{0,1\}. \end{aligned} \tag{17}$$

Case 2. a and c are the two degree-4 vertices in the 4-cycle: Let b' and d' be the third neighbor of b and d, respectively. Note that $b'\neq d'$ holds, otherwise $\{b,d\}$-$\{a,c,b'=d'\}$ would be a 2-3 structure. Also b' (d') is not adjacent to a or c, otherwise there would be an irregular triangle. See Fig. 5(b) for an illustration of this case. Let $0\leq k\leq 2$ be the number of vertices of degree 4 in $\{b',d'\}$.

It is easy to see that in the branch where $\{b,d\}$ is removed, the weight of b and d decreases by $2w_3$, that of a and c by $2w_4$, and that of b' and d' by at least $k(w_4-w_3)+(2-k)w_3$. Since a and c become degree-2 vertices in $G-\{b,d\}$, the measure further decreases by 2β by folding them. In total, the measure w decreases by $2w_3+2w_4+k(w_4-w_3)+(2-k)w_3+2\beta=(2+k)w_4+(4-2k)w_3+2\beta$.

In the other branch where $\{a,c\}$ is removed, b and d become degree-1 vertices and we will also further remove b' and d'. Let $V'=\{a,b,c,d,b',d'\}$. Removing V' decreases the sum of weights of vertices in V' by $(2+k)w_4+(4-k)w_3$. We consider how much weight of vertices in $V-V'$ decreases after removing V'. Note that there are at least $6+k$ edges between V' and $V-V'$ (b' and d' may be adjacent) and $|N(V')|\geq 4$. Then the weight in $V-V'$ decreases by at least $w_3+(3+k)(w_4-w_3)$ for $k\in\{0,1\}$ and by $2w_3+2(w_4-w_3)=2w_4$ for $k=2$. In Case 2, we can branch with at least one of the following recurrences:

$$C(w)\leq C(w-(2+k)w_4-(4-2k)w_3+2\beta)+C(w-(5+2k)w_4-(2-2k)w_3) \tag{18}$$

for $k\in\{0,1\}$; and

$$C(w)\leq C(w-4w_4)+C(w-(6w_4+2w_3)+2\beta) \quad \text{for } k=2. \tag{19}$$

Case 3. a and b (or a and d) are the two degree-4 vertices in the 4-cycle: Assume without loss of generality that a and b are the degree-4 vertices in the cycle. Define a', a'', b', c' and d' as in Case 1, and let b'' be the fourth neighbor of b (see Fig. 5(c) for an illustration). Since the graph has no irregular triangle, vertex c' (d') is different from any of b' and b'' (a' and a'') whereas $d'\in\{b',b''\}$ and $\{a',a''\}\cap\{b',b'',c'\}\neq\emptyset$. Also $c'\neq d'$, otherwise 5-cycle $c'cabd$ would contain a roof c'. We look at the branch where $\{a,c\}$ is removed. Vertex d will become a degree-1 vertex and we will further remove the dominated vertex d' and the degree-1 vertex d. Thus in this branch we will remove $N[d]$. We consider how much weight of vertices in $V-N[d]$ decreases after removing $N[d]$. Note that no two vertices in $N(d)$ are adjacent, otherwise there would be an irregular triangle or a roof. It is impossible to create a degree-0 vertex v after removing $N[d]$, otherwise $v = c'$ holds and $\{d,v\}$-$\{a,c,d'\}$ would be a 2-3 stricture. Hence each vertex in $N_2(d)$ has a neighbor in $V - N(d)$. There are at least seven edges between $N(d)$ and $N_2(d)$. If there is a degree-i vertex $z \in N_2(d)$ which has $(i-1)$ neighbors in $N(d)$, then the vertex z becomes a degree-1 vertex in $G - N[d]$, and folding z removes a vertex $x \in V - N[d]$, which deletes at least two edges between x and $N(x) - \{z\}$. Since there are at least $7 + 2 - (i-1)$ edges between $X = N[d] \cup \{z, x\}$ and $V - X$, deleting these edges decrease the weight of vertices in $V - X$ by at least $W = 2w_3 + \Delta w_4$ for $i = 3$ and $W = 2w_3$ for $i = 4$. In total, the weight of vertices in $V - N[v]$ decreases by at least $w_i + w_3 + W = w_4 + 3w_3$. Consider the case where each degree-i vertex in $N_2(d)$ has exactly $(i-2)$ neighbors in $N(d)$. In this case, deleting $N[d]$ decrease the weight of each degree-i vertex in $N_2(d)$ by w_i. Since there are at least seven edges between $N(d)$ and $N_2(d)$, deleting $N[d]$ decrease the weight of vertices in $N_2(d)$ by at least $3w_4 + w_3$ $(\geq w_4 + 3w_3)$. Therefore, the branch of removing $\{a, c\}$ decreases the measure w by at least $w_4 + 3w_3 + w_4 + 3w_3 = 2w_4 + 6w_3$.

This also holds for the other branch where $\{b, d\}$ is removed. In Case 3, we can branch with recurrence

$$C(w) \leq 2C(w - (2w_4 + 6w_3)). \tag{20}$$

Case 4. There are exactly three degree-4 vertices in the 4-cycle: Without loss of generality, we assume that the three degree-4 vertices are a, b and c. Note that the third neighbor d' of d is not adjacent to a or c. Note that a and d has no common neighbor, and a and b has a common neighbor a' only when $\delta(a') = 4$ (otherwise there would be an irregular triangle). Similarly for common neighbors between c and $\{b, d\}$. Hence in the branch of removing $\{b, d\}$, vertices a and c will be degree-2 vertices adjacent to pairs of vertices of degree ≥ 3. In the branch of removing $\{b, d\}$, the measure w decreases by at least $3w_4 + w_3 + w_3 + (w_4 - w_3) = 4w_4 + w_3$. Folding the two degree-2 vertices further decreases the measure by at least 2β.

In the other branch where $\{a, c\}$ is removed, vertex d becomes a degree-1 vertex and we will further remove $\{d'\}$. This decreases the weight of vertices in $\{a, b, c\}$ by $3w_4$, that in $\{d, d'\}$ by at least $2w_3$, and that in $V - \{a, b, c, d, d'\}$ by at least $3w_3$ (note that there are at least six edges between $\{a, c, d, d'\}$ and $V - \{a, b, c, d, d'\}$ and no degree-0 vertices will be created after removing $N[d]$). Totally the measure w decreases by at least $3w_4 + 5w_3$. In Case 4, we get recurrence

$$C(w) \leq C(w - (4w_4 + w_3 + 2\beta)) + C(w - (3w_4 + 5w_3)). \tag{21}$$

Case 5. All the vertices in the 4-cycle are degree-4 vertices: Since there are no 4-funnels in a reduced graph, the vertices $abcd$ do not induce a clique of size 4. Assume without loss of generality a and c are not adjacent. Let $N(a) = \{b, d, a', a''\}$ and $N(c) = \{b, d, c', c''\}$. Note that a and each of b and d have a common neighbor a' only when $\delta(a') = 4$ (otherwise a and a' would be in an irregular triangle), where a cannot have the common neighbor a' both with b and d (otherwise b would dominate d). Similarly for common neighbors between c and $\{b, d\}$. Hence the branch of removing $\{a, c\}$ decreases the weight of vertices a, b, c and d by $4w_4$. When vertices a and c have the third common neighbor z, the degree of z is four (otherwise $abcz$ would has more degree-3 vertices than the optimal 4-cycle $abcd$). Hence removing $\{a, c\}$ decreases the weight of vertices in $V - \{a, b, c, d\}$ by at least $4\Delta w_4$ if $\{a', a''\} \cap \{c', c''\} = \emptyset$ and $\delta(a') = \delta(a'') = \delta(c') = \delta(c'') = 4$ or by $4\Delta w_4 + (w_3 - \Delta w_4)$ otherwise. In the other branch of removing $\{b, d\}$, vertices a and c will be degree-2 vertices adjacent to pairs of vertices of degree ≥ 3. The branch of removing $\{b, d\}$ decreases the weight of vertices in $\{a, b, c, d\}$ by $4w_4$. When there are only two edges between $\{b, d\}$ and $V - \{a, b, c, d\}$, these edges can meet at a vertex z only when $\delta(z) = 4$ (otherwise $zdab$ would has more degree-3 vertices than the optimal 4-cycle $abcd$). Hence removing $\{b, d\}$ decreases the weight of vertices in $V - \{a, b, c, d\}$ by at least $\min\{2(w_4 - w_3), w_4\} = 2(w_4 - w_3)$. Folding the two degree-2 vertices further decreases the measure by at least 2β. In total the measure decreases by at least $6w_4 - 2w_3 + 2\beta$. When $\{a', a''\} \cap \{c', c''\} = \emptyset$ and $\delta(a') = \delta(a'') = \delta(c') = \delta(c'') = 4$, a vertex of degree ≥ 5 will be created and remain in a reduced graph or the measure further decreases by at least Δw_4 until all degree ≥ 5 vertices are eliminated in a reduced graph. In any case, we can save $\min\{\sigma_2, \Delta w_4\} = \sigma_2$ in the second branch. In Case 5 we get recurrences

$$C(w) \leq C(w-(8w_4 - 4w_3 + (w_3-\Delta w_4))+C(w-(6w_4 - 2w_3 + 2\beta)); \tag{22}$$

$$C(w) \leq C(w-(8w_4 - 4w_3))+C(w-(6w_4 - 2w_3 + 2\beta + \sigma_2)). \tag{23}$$

Note that after Step 4, no degree-4 vertex is contained in a 4-cycle.

6.2.4 Branches in Step 5

In this step, the algorithm will branch on an optimal degree-4 vertex v by either deleting v from the graph or deleting $N[v]$ from the graph.

We observe that when G has a degree-4 vertex in a component H after Step 4, there always exists a degree-4 vertex that is not contained in two edge-disjoint triangles, since otherwise H would be the line graph of a 3-regular graph, which must have been eliminated by our reduction rules.

Let v be an optimal degree-4 vertex v in a component H of a reduced graph G, where H contains none of irregular triangles, 4-cycles containing degree-4 vertices and vertices of degree ≥ 5 after Step 4. Since there is no irregular triangle in H, every degree-3 neighbor of v has exactly two neighbors in $N_2(v)$ (see Fig. 6 for all possible subgraphs induced by the neighbors of an optimal vertex v).

Let d_3 be the number of degree-3 vertices in $N(v)$, and p be the number of edges between $N(v)$ and $N_2(v)$, where $p = |N_2(v)|$ since v is not contained in any 4-cycle. Let ℓ ($\leq d_3$) be the number of degree-3 neighbors that have at least one degree-4 neighbor in $N_2(v)$, and k_3 be the number of degree-3 vertices in $N_2(v)$, where $k_3 \geq 2(d_3 - \ell)$ always holds. In the branch of deleting v, the total weight of the vertices in $N[v]$ decreases by at least

$$w_4 + (4 - d_3)\Delta w_4 + d_3 w_3 = (5 - d_3)w_4 + (2d_3 - 4)w_3.$$

By applying RG to the resulting graph $G - v$, the measure decreases as follows.

Lemma 16 *Let $d_3 \geq 1$. Then $G - v$ has $d_3 \geq 1$ degree-2 vertices. Let u_i ($1 \leq i \leq d_3$) be the degree-3 neighbors of v in G where u_i has neighbors $x_i, y_i \in N_2(v)$. Then one of the following* (i) *and* (ii) *holds:*
(i) *Applying RG to $G - v$ decreases the measure by at least $2w_3$ before a reduced graph is obtained from $G - v$;*
(ii) *No degree-3 neighbor of v is in a triangle or a 4-cycle in G, and no 6-cycle contains two degree-3 neighbors of v and their four neighbors other than v. None of Steps 1, 3, 4 or 5 of RG will be executed before a reduced graph is obtained from $G-v$. Step 2 of RG is executed each degree-2 vertex to $G-v$ without creating a new degree-2 vertex to leave a graph G^* with no degree-2 vertices, where each pair x_i and y_i is contracted into a vertex of degree $\delta(x_i)+\delta(y_i)-2$ and $w(G-v)-w(G^*) \geq d_3\beta$ holds.*

Proof. Assume that applying RG to $G - v$ does not decrease the measure by $2w_3$ or more before a reduced graph is obtained from $G - v$. Since none one of Steps 1, 3, 4 or 5 of RG will be executed since otherwise the measure would decrease by at least $w_4 + 2\Delta w_4$ ($\geq 2w_3$) by Lemma 13. Hence no degree-3 neighbor u of v in G is in a triangle uxy (where x and y are two degree-3 vertices in $N_2(v)$ since G has no irregular triangles and v is not contained in 4-cycle in G), because otherwise the vertex u becomes a dominating vertex degree-2 in $G-v$, contradicting that Step 3 of RG will not be executed. Then after removing v from G, Step 2 of RG will be repeatedly executed until no degree-2 vertex exists. Note that no two degree-3 neighbors of v have a common neighbor other than v since no 4-cycle contains a degree-4 vertex. Since the measure does not decrease by $2w_3$ or more, we see that G has no degree-3 neighbor of v is in a triangle or a 4-cycle in G and no 6-cycle contains two degree-3 neighbors of v and their four neighbors other than v. If a degree-3 neighbor u of v is in a 4-cycle $xuyz$ (where each of x, y and z is of degree 3 since no 4-cycle contains a degree-4 vertex), then folding the degree-2 vertex u in $G - v$ contracts x and y into a degree-3 vertex which is now incident to degree-2 vertex z, indicating that the weight of decrease from vertices x, y and z is $2w_3$, a contradiction. Similarly if G has a 6-cycle $u_1x_1x_2u_2y_2y_1$ for two degree-3 neighbors u_1 and u_2 of v, then folding degree-2 vertices u_1 and u_2 in $G - v$ decreases the degree of their neighbors by six in total, indicating that the weight of decrease from vertices x_1, y_1, x_2 and y_2 is at least $6\Delta w_4$ ($\geq 2w_3$ by (5)), a contradiction.

We finally claim that the degree of a vertex $t \notin \{v, u_1, \ldots, u_{d_3}\}$ decreases during the execution only when G has a 4-cycle containing a degree-3 neighbor of v or a 6-cycle that contains two degree-3 neighbors of v and their four neighbors other than v (when two vertices t and t' are contracted into t'', we regard that the degree of each of t and t' becomes that of t''). Let $t \notin \{v, u_1, \ldots, u_{d_3}\}$ be the first vertex whose degree decreases during folding degree-2 vertices in $G - v$ (hence no new degree-2 vertex than $u_1, \ldots, u_{d_3}$ has not been created yet). Immediately before the degree of x decreases, there must be a 4-cycle $u_1x_1ty_1$ for some degree-2 vertex u_1. Since no degree-3 neighbor of v is in a 4-cycle, this means that one of vertices x_1, t and y_1 is a vertex created by folding another degree-2 vertex u_2. If one of x_1 and y_1 is such a vertex, then u_1 and u_2 would have a common vertex x, contradicting that no two degree-2 vertices in $\{u_1, \ldots, u_{d_3}\}$ have a common vertex (even if x is assumed to be created by folding another degree-2 vertex u_3, we cannot avoid such a common vertex between two vertices in $\{u_1, \ldots, u_{d_3}\}$). Hence only t is a vertex created by folding another degree-2 vertex u_2. Hence before folding the degree-2 vertex u_2, the 4-cycle was a 6-cycle $u_1x_1x_2u_2y_2y_1$. Analogously none of vertices x_1, x_2, y_2 and y_1 can be created by folding degree-2 vertces in $\{u_1, \ldots, u_{d_3}\} - \{u_1, u_2\}$ (otherwise some two of them would have a common neighbor). This contradicts that such a 6-cycle exists in G, proving the claim. The claim implies that no new degree-2 vertex will be created during folding the d_3 degree-2 vertice in $G - v$, and the measure decreases by at least β by Lemma 14 before the d_3 degree-2 vertices are folded. Also the degree of any other vertex never decreases, and each pair x_i and y_i of u_i will be contracted into a vertex of degree $\delta(x_i) + \delta(y_i) - 2$, which will not decrease while Step 2 of RG is applied. ∎

In the second branch of deleting $N[v]$, the vertex weight decreases by $w_4 + (4 - d_3)w_4 + d_3w_3 = (5 - d_3)w_4 + d_3w_3$ from $N[v]$ and $k_3w_3 + (p - k_3)\Delta w_4$ from $N_2(v)$. The total weight decrease by deleting $N[v]$ is at least

$$(5 - d_3)w_4 + d_3w_3 + k_3w_3 + (p - k_3)\Delta w_4 = (5 - d_3 + p - k_3)w_4 + (d_3 - p + 2k_3)w_3,$$

which is at least $(5 - d_3 + p)w_4 + (d_3 - p)w_3$.

We analyze recurrences in Step 5 distinguishing five cases according to the number d_3 of degree-3 neighbors of v.

Case 1. $d_3 = 4$ (see Fig. 6(a)): Then $p = |N_2(v)| = 8$. By Lemma 16, the measure in $G - v$ further decreases by at least $\min\{4\beta, 2w_3\} = 4\beta$ and we get recurrence

$$C(w) \leq C(w - (w_4 + 4w_3 + 4\beta)) + C(w - (9w_4 - 4w_3)). \tag{24}$$

Case 2. $d_3 = 3$ (see Fig. 6(b)): Then $p = |N_2(v)| = 9$. By Lemma 16, the measure in $G - v$ further decreases by at least $\min\{3\beta, 2w_3\} = 3\beta$ and we get recurrence

$$C(w) \leq C(w - (2w_4 + 2w_3 + 3\beta)) + C(w - (11w_4 - 6w_3)). \tag{25}$$

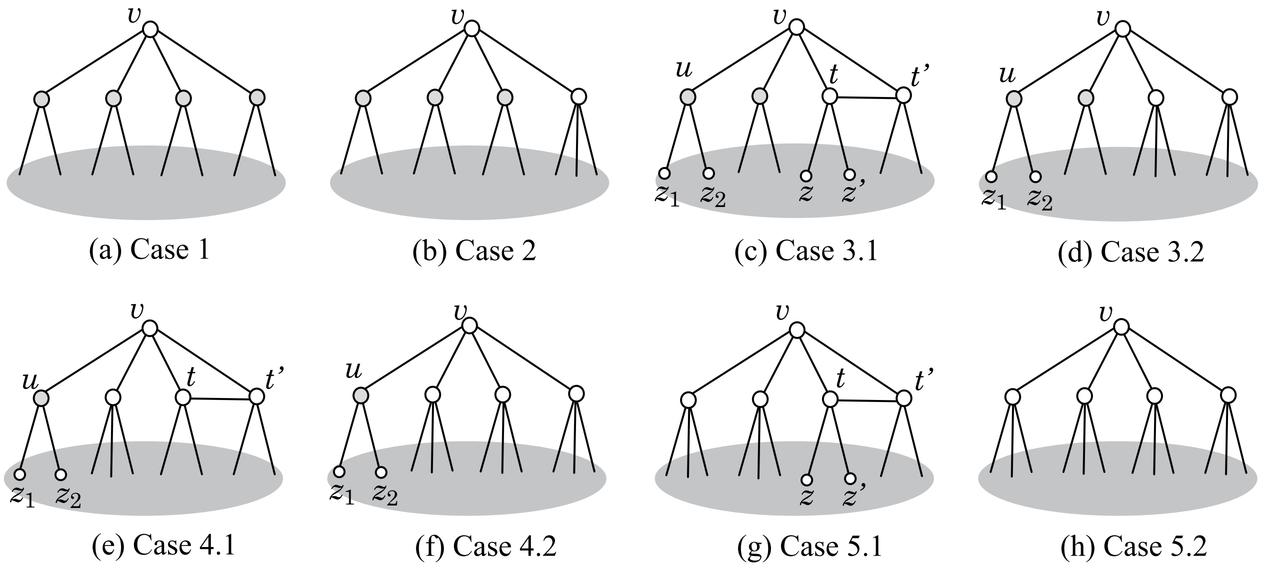

Figure 6: The structure of the neighbors of an optimal degree-4 vertex v in Step 5

Case 3. $d_3 = 2$: We distinguish two subcases on whether v is in a triangle or not.

Case 3.1 v is in a triangle vtt' (see Fig. 6(c)): Then $p = |N_2(v)| = 8$.

(i) Consider the case where the measure decreases by at least $2w_3$ before a reduced graph is obtained by applying RG to $G - v$ after removing v from G in the first branch. Then we get recurrence

$$C(w) \leq C(w-(3w_4 + 2w_3)) + C(w-(11w_4 - 6w_3)). \tag{26}$$

In what follows, we assume that the measure does not decrease by $2w_3$ or more before a reduced graph is obtained from $G-v$. Hence the conditions in Lemma 16(ii) hold. No degree-3 neighbor of v is in a triangle or a 4-cycle. In the first branch of removing v, we have ℓ vertices of degree ≥ 5 in the resulting graph G^* after folding all degree-2 vertices by Lemma 16(ii), where $w(G-v)-w(G^*) \geq 2\beta$.

(ii) Next assume that $\ell = 0$. Then $k_3 \geq 4$. This gives recurrence

$$C(w) \leq C(w-(3w_4+2\beta)) + C(w-(7w_4+2w_3)). \tag{27}$$

In the following we assume that $\ell \geq 1$.

(iii) There is a degree-6 vertex in a reduced graph obtained after applying RG to G^*. The algorithm branches on a vertex of degree ≥ 6 with recurrence (9) with shift σ_1, and we get recurrence

$$C(w) \leq C(w-(3w_4 + 2\beta + \sigma_1)) + C(w-(11w_4 - 6w_3)). \tag{28}$$

(iv) G^* has exactly h degree-6 vertices ($h = 1, 2$), and none of these degree-6 vertices remains in a reduced graph after applying RG to G^*. Then $k_3 \geq 2 - h$. By Lemma 13, we see that the measure decreases by at least $(h + 1)\Delta w_4$ before a reduced graph is obtained. Then we get recurrences

$$C(w) \leq C(w-(3w_4+2\beta+(h+1)\Delta w_4)) + C(w-((9+h)w_4-(2+2h)w_3)) \tag{29}$$

for $h = 1, 2$.

(v) G^* has no degree-6 vertex, and there is a degree-5 vertex in a reduced graph obtained after applying RG to G^*. Note that G^* has ℓ degree-5 vertices and $k_3 \geq 2$. In this case the algorithm branches on a vertex of degree 5 with recurrence (10) with shift σ_2, and we get recurrence

$$C(w) \leq C(w-(3w_4 + 2\beta + \sigma_2)) + C(w-(9w_4 - 2w_3)). \tag{30}$$

(vi) G^* has no degree-6 vertex, and no vertex of degree ≥ 5 exists in a reduced graph after applying RG to G^*. By Lemma 13, we see that the measure decreases by at least $(\ell+1)\Delta w_4$ before the *ell* degree-5 vertices are eliminated. By noting that $k_3 \geq 4-\ell$, we get recurrences

$$C(w) \leq C(w-(3w_4+2\beta+(\ell+1)\Delta w_4))+C(w-((7+\ell)w_4+(2-2\ell)w_3))$$

for $\ell = 1, 2$, which are covered by (29).

Case 3.2 v is not in any triangle (see Fig. 6(d)): Then $p = |N_2(v)| = 10$. We can derive recurrences analogously with Case 3.1, where the measure decrease in the second branch in Case 3.2 is larger by $2\Delta w_4$ than that in Case 3.1.

Case 4. $d_3 = 1$: Let u be the degree-3 neighbor of v, and $N(u) = \{v, z_1, z_2\}$. We distinguish two subcases on whether v is in a triangle or not.

Case 4.1 v is in a triangle vtt' (see Fig. 6(e)): Then $p = |N_2(v)| = 9$. We derive recurrences for $\ell \geq 1$ in a similar manner with Case 3.1.
(i) Consider the case where the measure decreases by at least $2w_3$ before a reduced graph is obtained by applying RG to $G-v$ after removing v from G in the first branch. Then we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+2w_3))+C(w-(13w_4-8w_3)). \tag{31}$$

In what follows, we assume that the measure does not decrease by $2w_3$ or more before a reduced graph is obtained from $G-v$. Hence the conditions in Lemma 16(ii) hold. No degree-3 neighbor of v is in a triangle or a 4-cycle. In the first branch of removing v, we have ℓ vertices of degree ≥ 5 in the resulting graph G^* after folding all degree-2 vertices by Lemma 16(ii), where $w(G-v)-w(G^*) \geq \beta$.
(ii) Assume that $\ell = 0$. Then the degree-3 neighbor u of v is adjacent to two degree-3 neighbors $z_1, z_2 \in N_2(v)$ which are not adjacent to each other. Hence $k_3 \geq 2$. We consider the second branch of removing $N[v]$. If $k_3 = 2$, then $G - N[v]$ has only two degree-2 vertices z_1 and z_2 (which are independent), and the measure decreases by at least $11w_4-4w_3+2\beta$ in the second branch since folding these degree-2 vertices in $G-N[v]$ further decreases the measure by 2β. If $k_3 = 3$, then $G-N[v]$ has exactly three degree-2 vertices and the measure decreases by at least $10w_4-2w_3+\beta$ in the second branch since folding these three degree-2 vertices in $G-N[v]$ finally contracts at least one pair of vertices of degree ≥ 3 even if some of these degree-2 vertices are adjacent. For $k_3 \geq 4$, the measure decreases by at least $9w_4$ in the second branch. Since $\min\{11w_4-4w_3+2\beta, 10w_4-2w_3+\beta, 9w_4\} = 11w_4-4w_3+2\beta$ by $\beta \leq 2w_3-w_4$, we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+\beta))+C(w-(11w_4-4w_3+2\beta)). \tag{32}$$

In the following assume that $\ell = 1$. Then the maximum degree of G is at least 5.
(iii) The maximum degree of G^* decreases before a reduced graph is obtained after applying RG to G^*. By Lemma 13, we see that the measure decreases by at least $2\Delta w_4$ before a reduced graph is obtained.

$$C(w) \leq C(w-(4w_4-2w_3+\beta+2\Delta w_4))+C(w-(13w_4-8w_3)). \tag{33}$$

In the following we assume that the maximum degree of G^* does not decrease before a reduced graph is obtained.
(iv) There is a degree-6 vertex in a reduced graph obtained after applying RG to G^*. The algorithm branches on a vertex of degree ≥ 6, and we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+\beta+\sigma_1))+C(w-(13w_4-8w_3)). \tag{34}$$

(v) There is a degree-5 vertex in a reduced graph obtained after applying RG to G^*. This means that G^* has a degree-5 vertex. Note that $k_3 \geq 1$. In this case the algorithm branches on a vertex of degree 5, and we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+\beta+\sigma_2))+C(w-(12w_4-6w_3)). \tag{35}$$

Case 4.2 v is not in any triangle (see Fig. 6(f)): Then $p = |N_2(v)| = 11$. For $\ell \geq 1$, we can derive recurrences analogously with Case 4.1, where the measure decrease in the second branch in Case 4.2 is larger by $2\Delta w_4$ than that in Case 4.1. However, we introduce shift σ_4 for each recurrence with $\ell \geq 1$.
(i) First consider the case where the measure decreases by at least $2w_3$ before a reduced graph is obtained by applying RG to $G - v$ after removing v from G in the first branch. Then we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+2w_3-\sigma_4))+C(w-(15w_4-10w_3-\sigma_4)). \tag{36}$$

In what follows, we assume that the measure does not decrease by $2w_3$ or more before a reduced graph is obtained from $G - v$. Hence the conditions in Lemma 16(ii) hold. No degree-3 neighbor of v is in a triangle or a 4-cycle. In the first branch of removing v, we have ℓ vertices of degree ≥ 5 in the resulting graph G^* after folding all degree-2 vertices by Lemma 16(ii).
(ii) Consider the case of $\ell = 0$. Then $k_3 \geq 2$ holds and we get recurrence

$$C(w) \leq C(w-(4w_4 - 2w_3 + \beta))+C(w-(13w_4 - 6w_3)). \tag{37}$$

In the following assume that $\ell = 1$. Then the maximum degree of G is at least 5.
(iii) The maximum degree of G^* decreases before a reduced graph is obtained after applying RG to G^*. By Lemma 13, we see that the measure decreases by at least $2\Delta w_4$ before a reduced graph is obtained. This gives recurrence

$$C(w) \leq C(w-(4w_4-2w_3+\beta+2\Delta w_4-\sigma_4))+C(w-(15w_4-10w_3-\sigma_4)). \tag{38}$$

(iv) There is a degree-6 vertex in a reduced graph obtained after applying RG to G^*. The algorithm branches on a vertex of degree ≥ 6, and we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+\beta+\sigma_1-\sigma_4))+C(w-(15w_4-10w_3-\sigma_4)). \tag{39}$$

(v) There is a degree-5 vertex in a reduced graph obtained after applying RG to G^*. This means that G^* has a degree-5 vertex. Note that $k_3 \geq 1$. In this case the algorithm branches on a vertex of degree 5, and we get recurrence

$$C(w) \leq C(w-(4w_4-2w_3+\beta+\sigma_2-\sigma_4))+C(w-(14w_4-8w_3-\sigma_4)). \tag{40}$$

Case 5. $d_3 = 0$: Then $p = |N_2(v)| \geq 10$. First consider the case where the measure decreases by at least $2w_3$ before a reduced graph is obtained by applying RG to $G - v$ after removing v from G in the first branch. Then we get recurrence

$$C(w) \leq C(w-(5w_4-4w_3+2w_3))+C(w-(15w_4-10w_3)). \tag{41}$$

In what follows, we assume that the measure does not decrease by $2w_3$ or more before a reduced graph is obtained from $G - v$. Hence the conditions in Lemma 16(ii) hold. Hence none of Steps 1, 3, 4 and 5 of RG is executed when RG is applied to $G - v$. We distinguish two subcases on whether v is in a triangle or not.

Case 5.1 v is in a triangle vtt' (see Fig. 6(g)): Then $p = |N_2(v)| = 10$. Let $N(v) = \{t, t', u_1, u_2\}$. Now v satisfies one of conditions (c) and (d) in the definition of optimal vertices. We show that v can only satisfy condition (c). For each $i = 1, 2$, degree-4 neighbor u_i of v not in triangle vtt' satisfies condition (c), and thereby the current vertex v also satisfies (c), i.e., one of u_1 or u_2, say u_1 is in a triangle $u_1y_1y_2$. Note that the last neighbor $y_3 \in N(u_1) - \{v, y_1, y_2\}$ is also a degree-4 vertex since otherwise u_1 should have been chosen as an optimal vertex satisfying (c) with a degree-3 neighbor.

Then y_3-u_1-$\{y_1, y_2\}$ will be a good funnel with $\delta(y_3) = 4$ after removing v in the first branch. We see that $G - v$ has no short funnel, since otherwise a 4-cycle and a triangle share an edge and this means that G would have a 4-cycle containing a degree-4 vertex or a roof. Hence not only Steps 1, 3, 4 and 5 but also Step 6 of RG will not be executed to obtain a reduced graph from $G - v$, and y_3-u-$\{y_1, y_2\}$ remains to be a good funnel in the reduced graph after applying RG. The algorithm then branches on the good funnel (or possibly the same type of good funnel generated at the other degree-3 neighbor u_2) with recurrence (13) with shift σ_3. Hence we get recurrence

$$C(w) \leq C(w-(5w_4-4w_3+\sigma_3))+C(w-(15w_4-10w_3)). \tag{42}$$

Case 5.2 v is not in any triangle (see Fig. 6(h)): Then $p = |N_2(v)| = 12$.
(i) $k_3 \geq 1$. In the first branch, the measure decreases by at least $5w_4 - 4w_3$. As observed in Case 4.1(v), the second branch decreases the measure by at least $15w_4 - 8w_3$ if $k_3 \geq 2$ and by at least $16w_4$–$10w_3$+β (by folding a degree-2 vertex) if $k_3 = 1$, where $\min\{15w_4–8w_3, 16w_4–10w_3+\beta\} = 16w_4 - 10w_3 + \beta$. Hence

$$C(w) \leq C(w-(5w_4-4w_3))+C(w-(16w_4-10w_3+\beta)). \tag{43}$$

(ii) In what follows, we assume that $k_3 = 0$.

(ii-1) Assume that a degree-4 neighbor u of v is in a triangle uy_1y_2 disjoint with v. Note that all neighbors in $N(u) = \{v, y_1, y_2, y_3\}$ are of degree 4 by $k_3 = 0$. Then after removing v in the first branch, y_3-u-$\{y_2, y_3\}$ will be a good funnel in $G - v$ such that $\delta(y_3) = 4$ and y_3 is not in a 4-cycle by the assumption on G. If $G - v$ is changed so that it has no good funnel by applying RG, then the measure further decreases by at least Δw_4 by Lemma 13. On the other hand, the algorithm branches on the good funnel with recurrence (13) with shift σ_3. Hence we get recurrence

$$C(w) \leq C(w-(5w_4-4w_3 + \min\{\Delta w_4, \sigma_3\}))+C(w-(17w_4-12w_3)). \tag{44}$$

(ii-2) Assume that no degree-4 neighbor of v is in a triangle disjoint with v. Hence there is no triangle containing a vertex in $N[v]$. Note that each degree-4 neighbor $u \in N(v)$ is adjacent to only degree-4 vertices in $N_2(v)$ (otherwise a neighbor $u \in N(v)$ with at least one degree-3 neighbor would be optimal instead of the current optimal vertex v). Hence the component H containing v is a 4-regular graph with no triangles. After removing v in the first branch, we see that Step 6 of RG will not be executed to $G - v$ (recall that none of Steps 1, 3, 4 and 5 of RG will be executed). When $G - v$ is a reduced graph, the algorithm chooses an optimal degree-4 vertex v' which is in $N_2(v)$ in G and now adjacent to a degree-3 vertex u in $G - v$. Hence v' satisfies the condition of Case 4.2(i) or Case 4.2(iii)-(v) since v is not in a 4-cycle in G the unique degree-3 neighbor u is adjacent to only degree-4 vertices in $G - v$. Then including the shift σ_4 saved from each recurrence in Case 4.2(i) or Case 4.2(iii)-(v), we get recurrence

$$C(w) \leq C(w-(5w_4-4w_3+\sigma_4)))+C(w-(17w_4-12w_3)). \tag{45}$$

6.2.5 Analysis for Step 6

It is easy to see that if none of the first six steps can be executed, the graph is a 3-regular graph. We use our $O^*(1.083506^n)$-time algorithm for MIS3 in [18] to solve it, and then we get

$$C(w) \leq O(1.083506^{\frac{w}{w_3}}), \tag{46}$$

which will generate the last constraint in our quasiconvex program.

6.3 Final Solution to Weights

Recurrences (9) to (46) generate the constraints in our quasiconvex program. By solving the quasiconvex program under conditions (1), (2), and (3) according to the method introduced in [5], we get a bound 1.13756673 of the branching factor for all recurrences by setting $w_3 = 0.6222440$, $w_4 = 1$, $w_5 = 1.3937424$ and $w_6 = 1.7714985$ (now $\beta = 0.22571$). This verifies Lemma 10.

Now the recurrence (23) and constraints $\Delta w_4 \leq \min\{w_3, \Delta w_5, \Delta w_6\}$ in (2), and $w_i + w_j \geq w_{i+j-2}$ with $3 \leq i \leq j \leq 4$ in (3) and (46) attain the tight branching factor 1.1376.

7 Concluding Remarks

After carefully checking the local structures and what will happen after branching on a degree-4 vertices, we improve the running time bound of MIS4 to $O^*(1.1376^n)$. Now (46) is the crucial bottleneck in the algorithm. If we remove these two constraints, many new bottlenecks will appear and the improvement is tiny. Maybe new technique and method are needed on this problem to get further significant improvement.

References

[1] Bourgeois, N., Escoffier, B., Paschos, V. T., van Rooij, J. M. M.: Maximum independent set in graphs of average degree at most three in $O(1.08537^n)$. In: TAMC, LNCS 6108 (2010) 373–384

[2] Bourgeois, N., Escoffier, B., Paschos, V. T., van Rooij, J. M. M., Fast algorithms for max independent set, Algorithmica 62(1-2), (2012) 382–415.

[3] Chen, J., Kanj, I. A., Xia, G.: Improved upper bounds for vertex cover. Theoretical Computer Science **411**(40-42) (2010) 3736–3756

[4] Chor, B., Fellows, M., Juedes, D. W.: Linear kernels in linear time, or how to save k colors in $O(n^2)$ steps. In: WG 2004. LNCS 3353, Springer (2004) 257–269

[5] Eppstein D.: Quasiconvex analysis of multivariate recurrence equations for backtracking algorithms. ACM Transactions on Algorithms **2**(4)(2006) 492-509

[6] Fomin, F. V., Grandoni, F., Kratsch, D.: A measure & conquer approach for the analysis of exact algorithms. J. ACM **56**(5) (2009) 1–32

[7] Fomin, F. V., Høie, K.: Pathwidth of cubic graphs and exact algorithms. Inf. Process. Lett. **97**(5) (2006) 191–196

[8] Fomin, F. V., Kratsch, D.: Exact Exponential Algorithms, Springer (2010)

[9] Fürer, M.: A faster algorithm for finding maximum independent sets in sparse graphs. In: LATIN 2006. LNCS 3887 , Springer (2006) 491–501

[10] Jian, T.: An $O(2^{0.304n})$ algorithm for solving maximum independent set problem. IEEE Transactions on Computers **35**(9) (1986) 847–851

[11] Kneis, J., Langer, A., Rossmanith, P.: A fine-grained analysis of a simple independent set algorithm. In Kannan, R., Kumar, K.N., eds.: FSTTCS 2009. V. 4 LIPIcs., Dagstuhl, Germany (2009) 287–298

[12] Razgon, I.: Faster computation of maximum independent set and parameterized vertex cover for graphs with maximum degree 3. J. of Discrete Algorithms **7**(2) (2009) 191–212

[13] Robson, J.: Algorithms for maximum independent sets. J. of Algorithms **7**(3) (1986) 425–440

[14] Tarjan, R., Trojanowski, A.: Finding a maximum independent set. SIAM J. on Computing **6**(3) (1977) 537–546

[15] West, D.: Introduction to Graph Theory. Prentice Hall. 1996

[16] Xiao, M.: A simple and fast algorithm for maximum independent set in 3-degree graphs. In: M. Rahman and S. Fujita: WALCOM 2010, LNCS 5942 (2010) 281–292

[17] Xiao, M., Chen, J., Han, X.: Improvement on vertex cover and independent set problems for low-degree graphs. Chinese J. of Computers **28**(2) (2005) 153–160

[18] Xiao, M., Nagamochi, H.: Confining sets and avoiding bottleneck cases: A simple maximum independent set algorithm in degree-3 graphs. Theoretical Computer Science **469** (2013) 92–104

[19] Xiao, M., Nagamochi, H.: Further Improvement on Maximum Independent Set in Graphs with Maximum Degree 4. Technical report 2012-003, Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University (2012)

Proof of Lemma 5. Lemma 5 follows from the next lemma and its proof.

Lemma 17 *It holds that* $\alpha(G_1) \geq \alpha(G_1^v)$ *and*

$$\alpha(G) = \begin{cases} \alpha(G_1^v)+\alpha(G[V_2+v]) & \text{if } \alpha(G_1)=\alpha(G_1^v) \\ \alpha(G_1)+\alpha(G_2) & \text{if } \alpha(G_1) > \alpha(G_1^v). \end{cases}$$

Proof. Since G_1^v is an induced subgraph of G, we know that $\alpha(G_1) \geq \alpha(G_1^v)$ holds. Let S be a maximum independent set of G. Then we have $|S \cap V_1| \leq \alpha(G_1)$ and $|S \setminus V_1| \leq \alpha(G[V_2+v])$.

First assume that $\alpha(G_1) = \alpha(G_1^v)$. Then we have $|S \cap V_1| \leq \alpha(G_1) = \alpha(G_1^v) = |S_1^v|$ for any maximum independent set S_1^v of G_1^v. Then $\alpha(G) = |S \cap V_1| + |S \setminus V_1| \leq \alpha(G_1^v) + \alpha(G[V_2+v])$. On the other hand, $|S_1^v \cup (S \setminus V_1)| \leq \alpha(G)$ since $S_1^v \cup (S \setminus V_1)$ is an independent set of G since $G[V_2+v]$ and G_1^v are separated by cut $V_1 \setminus V_1^v$. Therefore, $\alpha(G) = \alpha(G_1^v) + \alpha(G[V_2+v])$.

Next, we consider the case of $\alpha(G_1) > \alpha(G_1^v)$. Let $\hat{S}_1$ be a maximum independent set of $G[V_1+v]$. We have that $|\hat{S}_1| \leq \min\{\alpha(G_1^v)+1, \alpha(G_1)\} \leq \alpha(G_1)$. We have that $|S \cap V_2| \leq \alpha(G_2)$ and $|S \setminus V_2| \leq |\hat{S}_1| \leq \alpha(G_1)$. Then $\alpha(G) = |S \cap V_2| + |S \setminus V_2| \leq \alpha(G_2) + \alpha(G_1)$. On the other hand, $\alpha(G) \geq |(S \cap V_2) \cup S_1| = |S \cap V_2| + \alpha(G_1)$ for any maximum independent set S_1 of G_1, since $(S \cap V_2) \cup S_1$ is also an independent set of G since G_2 and G_1 are separated by cut $\{v\}$. Therefore, $\alpha(G) = \alpha(G_1) + \alpha(G_2)$. ∎

Proof of Lemma 6. Lemma 6 follows from the next lemma and its proof.

Lemma 18 *Assume without loss of generality that* $\alpha(G_1^u) \leq \alpha(G_1^v)$. *Then it holds that* $\alpha(G_1^{uv}) \leq \alpha(G_1^u) \leq \alpha(G_1^v) \leq \alpha(G_1)$ *and*

$$\alpha(G)=\begin{cases} \alpha(G_1^{uv})+\alpha(G[V_2\cup\{u,v\}]) & \text{if } \alpha(G_1^{uv})=\alpha(G_1), & \text{(i)} \\ \alpha(G_1)+\alpha(\widetilde{G_2}) & \text{if } \alpha(G_1^{uv})<\alpha(G_1^u)=\alpha(G_1^v)=\alpha(G_1), & \text{(ii)} \\ \alpha(G_1^v)+\alpha(G[V_2+v]) & \text{if } \alpha(G_1^u)<\alpha(G_1^v)=\alpha(G_1), & \text{(iii)} \\ \alpha(G_1)+\alpha(\widehat{G_2}) & \text{if } \alpha(G_1^{uv})+1=\alpha(G_1) \text{ and } \alpha(G_1^v)<\alpha(G_1), & \text{(iv)} \\ \alpha(G_1)+\alpha(G_2) & \text{if } \alpha(G_1^{uv})+2\leq\alpha(G_1) \text{ and } \alpha(G_1^v)<\alpha(G_1), & \text{(v)} \end{cases}$$

where $\widetilde{G_2}$ *is the graph obtained from* $G[V_2\cup\{u,v\}]$ *by adding an edge* uv *if* v *and* u *are not adjacent and* $\widehat{G_2} = G/(V_1\cup\{u,v\})$ *is the graph obtained from* G *by contracting* $V_1\cup\{u,v\}$ *into a single vertex* z *and deleting multi-edges and self-loops.*

Proof. Since G_1^{uv} is an induced subgraph of G_1^u (resp., G_1^v), and G_1^u (resp., G_1^v) is an induced subgraph of G_1, we know that $\alpha(G_1^{uv}) \le \alpha(G_1^u) \le \alpha(G_1)$ and $\alpha(G_1^{uv}) \le \alpha(G_1^v) \le \alpha(G_1)$. We consider the other five possible relations among $\alpha(G_1^{uv})$, $\alpha(G_1^u)$, $\alpha(G_1^v)$ and $\alpha(G_1)$. In the following, S (resp., S_1^{uv}, S_1^u, S_1^v and S_1) denotes an arbitrary maximum independent set of G (resp., G_1^{uv}, G_1^u, G_1^v and G_1^u).

Case (i). $\alpha(G_1^{uv}) = \alpha(G_1)$: We partition $V(G)$ into V_1^{uv}, $Z = V_1 \setminus V_1^{uv}$ and $V_2 \cup \{u, v\}$ so that there is no edge between V_1^{uv} and $V_2 \cup \{u, v\}$. Hence we have $\alpha(G) \ge \alpha(G_1^{uv}) + \alpha(G[V_2 \cup \{u, v\}])$. The converse can be obtained by

$$\begin{aligned}\alpha(G) = |S| &= |S \cap V_1| + |S \cap (V_2 \cup \{u, v\})| & \le \alpha(G_1) + |S \cap (V_2 \cup \{u, v\})| \\ &\le \alpha(G_1^{uv}) + |S \cap (V_2 \cup \{u, v\})| & = |S_1^{uv}| + |S \cap (V_2 \cup \{u, v\})| \\ &\le \alpha(G_1^{uv}) + \alpha(G[V_2 \cup \{u, v\}]). &\end{aligned}$$

Case (ii). $\alpha(G_1^{uv}) < \alpha(G_1^u) = \alpha(G_1^v) = \alpha(G_1)$: This holds because $V_2 \cup \{u, v\}$ is a subgraph of $\widetilde{G_2}$. We first show that G has a maximum independent set S containing at most one vertex in $\{u, v\}$. If $u, v \in S$, then can replace $S \cap (V_1 \cup \{u\})$ with S_1^v in S to get another maximum independent set $S' = S_1^v \cup \{v\} \cup (S \cap (V_2 \setminus N(\{u, v\})))$ of G, since

$$\begin{aligned}\alpha(G) = |S| &= |S \cap V_1^{uv}| + |\{u, v\}| + |S \cap (V_2 \setminus N(\{u, v\}))| \\ &\le \alpha(G_1^{uv}) + |\{u, v\}| + |S \cap (V_2 \setminus N(\{u, v\}))| \\ &\le \alpha(G_1^v) - 1 + |\{u, v\}| + |S \cap (V_2 \setminus N(\{u, v\}))| \\ &= |S_1^v| + |\{v\}| + |S \cap (V_2 \setminus N(\{u, v\}))| \\ &\le \alpha(G).\end{aligned}$$

Hence G has a maximum independent set S such that $|\{u, v\} \cap S| = 1$. Now we observe that

$$\begin{aligned}\alpha(G) = |S| &= |S \cap V_1| + |S \cap (V_2 \cup \{u, v\})| \\ &\le |S \cap V_1| + \alpha(\widetilde{G_2}) \\ &\le \max\{\alpha(G_1^u), \alpha(G_1^v), \alpha(G_1)\} + \alpha(\widetilde{G_2}) \\ &= \min\{\alpha(G_1^u), \alpha(G_1^v), \alpha(G_1)\} + \alpha(\widetilde{G_2}) \\ &\le \alpha(G),\end{aligned}$$

indicating that $\alpha(G) = \alpha(G_1) + \alpha(\widetilde{G_2})$.

Case (iii). $\alpha(G_1^u) < \alpha(G_1^v) = \alpha(G_1)$: We partition $V(G)$ into V_1^v, $Z = \{u\} \cup (V_1 \setminus V_1^v)$ and $V_2 + v$ such that there is no edge between V_1^v and $V_2 + v$. Hence $\alpha(G) \ge \alpha(G_1^v) + \alpha(G[V_2 + v])$. If $u \notin S$, then $|S \cap (V_1 + u)| = |S \cap V_1| \le \alpha(G_1) = \alpha(G_1^v)$. If $u \in S$, then $|S \cap (V_1 + u)| = |S \cap V_1^u| + 1 \le \alpha(G_1^u) + 1 \le \alpha(G_1^v)$. In any case we have

$$\begin{aligned}\alpha(G) = |S| &= |S \cap (V_1 + u)| + |S \cap (V_2 + v)| \le \alpha(G_1^v) + |S \cap (V_2 + v)| \\ &= |S_1^v| + |S \cap (V_2 + v)| \le \alpha(G).\end{aligned}$$

Case (vi). $\alpha(G_1^{uv}) + 1 = \alpha(G_1)$ and $\alpha(G_1^v) < \alpha(G_1)$: We first observe that assumption $\alpha(G_1^u) \le \alpha(G_1^v) < \alpha(G_1)$ implies that G has a maximum independent set S with $|\{u, v\} \cap S| = 0$ or 2. If $\{u, v\} \cap S = \{u\}$, then we can replace $S \cap (V_1 \cup \{u, v\})$ with S_1 in S to get another maximum independent $S' = S_1 \cup (S \cap V_2)$ set of G, where

$$\begin{aligned}\alpha(G) = |S| &= |S \cap V_1| + |S \cap \{u, v\}| + |S \cap V_2| \\ &\le \alpha(G_1^u) + 1 + |S \cap V_2| \\ &\le \alpha(G_1) + |S \cap V_2| \\ &= |S_1| + |S \cap V_2| \le \alpha(G).\end{aligned}$$

Symmetrically if $\{u,v\} \cap S = \{v\}$, then $\alpha(G_1^v) < \alpha(G_1)$ implies that we can replace $S \cap (V_1 \cup \{u,v\})$ with S_1 in S to get another maximum independent set $S' = S_1 \cup (S \cap V_2)$ of G. Hence G has a maximum independent set S with $|\{u,v\} \cap S| = 0$ or 2.

When $|\{u,v\} \cap S| = 2$, we have $\alpha(G) = \alpha(G_1^{uv}) + |S \cap (V_2 \cup \{u,v\})| \leq \alpha(G_1^{uv}) + \alpha(\widehat{G_2}) + 1$. When $|\{u,v\} \cap S| = 0$, we have $\alpha(G) = \alpha(G_1) + |S \cap (V_2 \cup \{u,v\})| \leq \alpha(G_1) + \alpha(\widehat{G_2})$. In any case, we have $\alpha(G) \leq \alpha(G_1^{uv}) + 1 + \alpha(\widehat{G_2}) = \alpha(G_1) + \alpha(\widehat{G_2})$. We show the converse. For a maximum independent set S^* of $\widehat{G_2}$, if $z \in S^*$ (resp., $z \notin S^*$) then we have an independent set S' of G such that $S' = S_1^{uv} \cup (S^* \setminus \{z\}) \cup \{u,v\}$ (resp., $S' = S_1 \cup S^*$) and $\alpha(G) \geq |S'| = \alpha(G_1^{uv}) + \alpha(\widehat{G_2}) - 1 + 2$ (resp., $\alpha(G) \geq |S'| = \alpha(G_1) + \alpha(\widehat{G_2})$), where $\alpha(G_1^{uv}) + 1 + \alpha(\widehat{G_2}) = \alpha(G_1) + \alpha(\widehat{G_2})$ by assumption.

Case (v). $\alpha(G_1^v) < \alpha(G_1)$ and $\alpha(G_1^{uv}) + 2 \leq \alpha(G_1)$: As in Case (iv), assumption $\alpha(G_1^u) \leq \alpha(G_1^v) < \alpha(G_1)$ implies that G has a maximum independent set S with $|\{u,v\} \cap S| = 0$ or 2. If $|\{u,v\} \cap S| = 2$, then we can replace $S \cap (V_1 \cup \{u,v\})$ with S_1 in S to get another maximum independent set $S' = S_1 \cup (S \cap V_2)$ of G with $\{u,v\} \cap S' = \emptyset$, where

$$\begin{aligned}
\alpha(G) = |S| &= |S \cap V_1^{uv}| + |S \cap \{u,v\}| + |S \cap V_2| \\
&\leq \alpha(G_1^{uv}) + 2 + |S \cap V_2| \\
&\leq \alpha(G_1) + |S \cap V_2| \\
&= |S_1| + |S \cap V_2| \leq \alpha(G).
\end{aligned}$$

Hence G has a maximum independent set S with $S \cap \{u,v\} = \emptyset$, indicating that This means that $\alpha(G) = \alpha(G_1) + \alpha(G_2)$. ∎