



The limits of local search for weighted k -set packing

Meike Neuwöhner¹

Received: 5 August 2022 / Accepted: 26 September 2023
© The Author(s) 2023

Abstract

We consider the weighted k -set packing problem, where, given a collection \mathcal{S} of sets, each of cardinality at most k , and a positive weight function $w : \mathcal{S} \rightarrow \mathbb{Q}_{>0}$, the task is to find a sub-collection of \mathcal{S} consisting of pairwise disjoint sets of maximum total weight. As this problem does not permit a polynomial-time $o(\frac{k}{\log k})$ -approximation unless $P = NP$ (Hazan et al. in *Comput Complex* 15:20–39, 2006. <https://doi.org/10.1007/s00037-006-0205-6>), most previous approaches rely on local search. For twenty years, Berman’s algorithm *SquareImp* (Berman, in: *Scandinavian workshop on algorithm theory*, Springer, 2000. https://doi.org/10.1007/3-540-44985-X_19), which yields a polynomial-time $\frac{k+1}{2} + \epsilon$ -approximation for any fixed $\epsilon > 0$, has remained unchallenged. Only recently, it could be improved to $\frac{k+1}{2} - \frac{1}{63,700,993}$ by Neuwöhner (38th International symposium on theoretical aspects of computer science (STACS 2021), *Leibniz international proceedings in informatics (LIPIcs)*, 2021. <https://doi.org/10.4230/LIPIcs.STACS.2021.53>). In her paper, she showed that instances for which the analysis of *SquareImp* is almost tight are “close to unweighted” in a certain sense. But for the unit weight variant, the best known approximation guarantee is $\frac{k+1}{3} + \epsilon$ (Fürer and Yu in *International symposium on combinatorial optimization*, Springer, 2014. https://doi.org/10.1007/978-3-319-09174-7_35). Using this observation as a starting point, we conduct a more in-depth analysis of close-to-tight instances of *SquareImp*. This finally allows us to generalize techniques used in the unweighted case to the weighted setting. In doing so, we obtain approximation guarantees of $\frac{k+\epsilon_k}{2}$, where $\lim_{k \rightarrow \infty} \epsilon_k = 0$. On the other hand, we prove that this is asymptotically best possible in that local improvements of logarithmically bounded size cannot produce an approximation ratio below $\frac{k}{2}$.

This is the full version of the paper [18], which appeared as an extended abstract in the proceedings of IPCO 2022.

✉ Meike Neuwöhner
neuwohner@or.uni-bonn.de

¹ Research Institute for Discrete Mathematics, University of Bonn, Lennéstraße 2, 53113 Bonn, Germany

Keywords Weighted k -set packing · Local search · d -Claw free graphs · Independent set

Mathematics Subject Classification 05C69

1 Introduction

For a positive integer k , the *weighted k -Set Packing problem* is defined as follows: Given a family \mathcal{S} of sets each of size at most k together with a positive weight function $w : \mathcal{S} \rightarrow \mathbb{Q}_{>0}$, the task is to find a sub-collection A of \mathcal{S} of maximum weight such that the sets in A are pairwise disjoint. For $k \leq 2$, the weighted k -Set Packing problem reduces to the Maximum Weight Matching problem by adding dummy elements to sets of cardinality less than two. The Maximum Weight Matching problem is known to be solvable in polynomial time [8].

For $k \geq 3$, the weighted k -Set Packing problem is NP -hard since it generalizes the optimization variant of the NP -complete 3-Dimensional Matching problem [14]. In addition, Hazan, Safra and Schwartz [12] have shown that there cannot even be a polynomial-time $o\left(\frac{k}{\log k}\right)$ -approximation for the weighted k -Set Packing problem unless $P = NP$.

On the positive side, a simple greedy algorithm yields an approximation guarantee of k (see for example [5]). In order to improve on this, the technique that has proven most successful so far is *local search*. The basic idea is to start with an arbitrary solution (e.g. the empty one) and to iteratively improve the current solution by applying some sort of local modifications until no more of these exist.

More precisely, given a feasible solution A to the weighted k -Set Packing problem, we call a collection $X \subseteq \mathcal{S} \setminus A$ consisting of pairwise disjoint sets a *local improvement of A of size $|X|$* if

$$w(X) > w(\{a \in A : \exists x \in X : a \cap x \neq \emptyset\}),$$

that is, if replacing the collection of sets in A that intersect sets in X by the sets in X increases the weight of the solution. Note that whenever A is sub-optimum and A^* is a solution of maximum weight, then $A^* \setminus A$ defines a local improvement of A . However, if one aims at designing a polynomial-time algorithm, it is of course infeasible to check subsets of \mathcal{S} of arbitrarily large size.

1.1 The unit weight case

For the special case of unit weights, Hurkens and Schrijver [13] showed that local improvements of arbitrarily large but constant size result in approximation guarantees arbitrarily close to $\frac{k}{2}$. Their paper also provides matching lower bound examples proving their result to be tight. Since then, a lot of progress has been made regarding the special case where $w \equiv 1$, which we will also refer to as the *unweighted k -Set Packing problem*. In 1995, at the cost of a quasi-polynomial running time, Halldórsson

[11] achieved an approximation factor of $\frac{k+2}{3}$ by applying local improvements of size logarithmic in the total number of sets. Cygan, Grandoni and Mastrolilli [7] managed to get down to an approximation factor of $\frac{k+1}{3} + \epsilon$, still with a quasi-polynomial running time.

The first polynomial-time algorithm improving on the result by Hurkens and Schrijver [13] was obtained by Sviridenko and Ward [21] in 2013. By combining color coding with the algorithm presented in [11], they achieved an approximation ratio of $\frac{k+2}{3}$. This result was further improved to $\frac{k+1}{3} + \epsilon$ for any fixed $\epsilon > 0$ by Cygan [6], obtaining a polynomial running time doubly exponential in $\frac{1}{\epsilon}$. The best approximation algorithm for the unweighted k -Set Packing problem in terms of performance ratio and running time is due to Fürer and Yu [10] from 2014. They achieve the same approximation guarantee as Cygan [6], but a running time only singly exponential in $\frac{1}{\epsilon^2}$. Moreover, they show that their result is best possible in that there exist arbitrarily large instances that feature solutions that do not permit any local improvement of size $o(|S|^{\frac{1}{5}})$, but that are by a factor of $\frac{k+1}{3}$ smaller than the optimum.

1.2 General weights and the MWIS in $(k+1)$ -claw free graphs

In the weighted setting, much less is known. Arkin and Hassin [2] have shown that unlike the unit weight case, local improvements of constant size cannot produce an approximation ratio better than $k - 1$ for general weights. Both papers improving on this deal with a more general problem, the *Maximum Weight Independent Set problem (MWIS) in $k + 1$ -claw free graphs* [3, 5]:

For $d \geq 1$, a d -claw C [3] is defined to be a star consisting of one *center node* and a set T_C of d *talons* connected to it. An undirected graph is said to be d -claw free if none of its induced subgraphs form a d -claw.

For $d \leq 3$, the MWIS in d -claw free graphs can be solved in polynomial time (see [15, 19] for the unweighted, [16] for the weighted variant), while for $d \geq 4$, again no $o(\frac{d}{\log d})$ -approximation algorithm is possible unless $P = NP$ [12].

Given an undirected graph $G = (V, E)$ and an independent set $A \subseteq V$, we call an independent set $X \subseteq V \setminus A$ a local improvement of A if the weight of X exceeds the weight of its neighborhood in A . Using this definition, most of the previous results for the (weighted or unweighted) k -Set Packing problem also apply to the more general context of the MWIS in $k + 1$ -claw free graphs. However, it is not known how to get down to a polynomial (instead of quasi-polynomial) running time for the algorithms in [6, 21] and [10] since there is no obvious equivalent to applying color coding to the underlying universe.

By considering the *conflict graph* G_S associated with an instance of the weighted k -Set Packing problem, we obtain a weight preserving one-to-one correspondence between feasible solutions to the weighted k -Set Packing problem and independent sets in G_S . The vertices of G_S are given by the sets in S and are equipped with the respective weights. The edges represent non-empty set intersections. To see that G_S is indeed $k + 1$ -claw free, observe that the set corresponding to the center of a $k + 1$ -claw in G_S would need to intersect each of the $k + 1$ pairwise disjoint sets corresponding to the talons. But this contradicts the fact that its cardinality is bounded by k .

The first significant improvement over the approximation guarantee of k achieved by the greedy approach for the MWIS in $k + 1$ -claw free graphs was made by Chandra and Halldórsson [5]. In each iteration, their algorithm *BestImp* picks a certain type of local improvement that maximizes the ratio between the total weight of the vertices added to and removed from the current solution. By further scaling and truncating the weight function to ensure a polynomial number of iterations, Chandra and Halldórsson [5] obtain a $\frac{2(k+1)}{3} + \epsilon$ -approximation algorithm for the MWIS in $k + 1$ -claw free graphs.

1.3 Berman's algorithm *SquareImp*

For 20 years, Berman's algorithm *SquareImp* [3] has been the state-of-the-art for both the MWIS in $k + 1$ -claw free graphs and the weighted k -Set Packing problem. *SquareImp* proceeds by iteratively applying local improvements of the *squared weight function* that arise as sets of talons of claws in G , until no more exist. In doing so, *SquareImp* achieves an approximation ratio of $\frac{k+1}{2}$, leading to a polynomial-time $\frac{k+1}{2} + \epsilon$ -approximation algorithm for both the MWIS in $k + 1$ -claw free graphs and the weighted k -Set Packing problem for any fixed $\epsilon > 0$.

Berman [3] also provides an example for $w \equiv 1$ showing that his analysis of *SquareImp* is tight. As the example uses unit weights, he concludes that applying the same type of local improvement algorithm for a different power of the weight function does not provide further improvements. However, as also implied by the result in [13], while no small improvements *forming the set of talons of a claw* in the input graph exist in the tight example given by Berman [3], once this additional condition is dropped, improvements of small constant size can be found quite easily. This observation is the basis of a recent paper by Neuwöhner [17], who managed to obtain an approximation guarantee slightly below $\frac{k+1}{2}$ by taking into account a broader class of local improvements, namely all improvements of the squared weight function of size at most $k^2 + k$.

2 Our contribution

The main result of this paper is the following theorem:

Theorem 1 *There exists a polynomial-time algorithm that obtains approximation guarantees of $\frac{k+\epsilon_k}{2}$ for the weighted k -Set Packing problem for $k \geq 3$, where $0 \leq \epsilon_k \leq 1$ for all $k \geq 3$ and $\lim_{k \rightarrow \infty} \epsilon_k = 0$.*

In order to prove Theorem 1, it will be convenient to phrase our algorithm and most of our results in terms of the more general MWIS in $k + 1$ -claw free graphs instead of the weighted k -Set Packing problem. In fact, the algorithm we propose also yields a quasi-polynomial time $\frac{k+\epsilon_k}{2}$ -approximation for the MWIS in $k + 1$ -claw free graphs, but to get down to a polynomial running time, we exploit the additional structure (the conflict graph of) an instance of the weighted k -Set Packing problem features.

The algorithm *LogImp* we propose builds on Berman's algorithm *SquareImp* in that it also searches for local improvements with respect to the squared weight function that constitute the set of talons of a claw in the input graph, as well as an additional

type of improvement of up to logarithmic size (again with respect to the squared weight function). In particular, we can apply Berman’s analysis of SquareImp to our algorithm. Following [17], we show that whenever Berman’s analysis of SquareImp is close to being tight, the instance is locally unweighted in the sense that almost every time that a vertex from the solution chosen by SquareImp and a vertex from any optimum solution share an edge, their weights must be very similar. While [17] merely focuses on one of the two major steps in Berman’s analysis, we consider both of them, allowing us to derive much stronger statements concerning the structure of instances where SquareImp does not do much better than a $\frac{k+1}{2}$ -approximation. In particular, we are able to transfer techniques that are used in the state-of-the-art works on the unweighted k -Set Packing problem [6, 10] to a setting where weights are locally similar. This idea gives rise to the precise notion of a local improvement of logarithmic size we consider and is the main ingredient towards Theorem 1.

We further prove the approximation guarantees stated in Theorem 1 to be asymptotically best possible by showing that a local improvement algorithm that, for an arbitrarily chosen, but fixed parameter $\alpha \in \mathbb{R}$, starts with the empty solution and iteratively applies local improvements of w^α of size $\mathcal{O}(\log(|\mathcal{S}|))$ until no more exist, cannot produce an approximation guarantee better than $\frac{k}{2}$ for the weighted k -Set Packing problem with $k \geq 3$.¹

Theorem 2 *Let $k \geq 3$, $\alpha \in \mathbb{R}$, $0 < \epsilon < 1$ and $C > 0$. Then for each $N_0 \in \mathbb{N}$, there exists an instance (\mathcal{S}, w) of the weighted k -Set Packing problem with $|\mathcal{S}| \geq N_0$ and such that there are feasible solutions $A, A^* \subseteq \mathcal{S}$ with the following two properties:*

- *There is no local improvement of A of size at most $C \cdot \log(|\mathcal{S}|)$ with respect to w^α , that is, there is no disjoint sub-collection $X \subseteq \mathcal{S} \setminus A$ of sets such that*

$$|X| \leq C \cdot \log(|\mathcal{S}|) \text{ and } w^\alpha(X) > w^\alpha(\{a \in A : \exists x \in X : a \cap x \neq \emptyset\}).$$

- *We have $w(A^*) \geq \frac{k-\epsilon}{2} \cdot w(A)$.*

Theorem 2 significantly extends the state of knowledge in terms of lower bound examples. Even more importantly, we can finally (at least asymptotically) answer the long-standing question of how far one can get by using pure local search in the weighted setting. In doing so, we are also the first ones to port the idea of searching for local improvements of logarithmic size, which has proven very successful for unit weights [6, 7, 10, 11, 21], to the weighted setting.

The rest of this paper is organized as follows: Sect. 3 recaps some of the definitions and main results from [3] that we will employ in the analysis of our local improvement algorithm *LogImp* in Sect. 4. Section 5 then shows how to make *LogImp* run in polynomial time by means of color coding. In Sect. 6, we explain our lower bound construction. Finally, Sect. 7 provides a brief conclusion.

¹ Observe that such an algorithm might just pick the solution A from Theorem 2 set by set.

Algorithm 1: SquareImp [3]**Input:** an undirected $k + 1$ -claw free graph $G = (V, E)$, $w : V \rightarrow \mathbb{Q}_{>0}$ **Output:** an independent set $A \subseteq V$

```

1  $A \leftarrow \emptyset$ 
2 while there exists a claw  $C$  in  $G$  that improves  $w^2(A)$  do
3    $A \leftarrow (A \setminus N(T_C, A)) \cup T_C$ 
4 return  $A$ 

```

3 Preliminaries

Definition 3 (*Neighborhood* [3]) Given an undirected graph $G = (V, E)$ and subsets $U, W \subseteq V$ of its vertices, we define the *neighborhood* $N(U, W)$ of U in W as $N(U, W) := \{w \in W : \exists u \in U : \{u, w\} \in E \vee u = w\}$. For $u \in V$ and $W \subseteq V$, we write $N(u, W)$ instead of $N(\{u\}, W)$.

Notation Given $w : V \rightarrow \mathbb{Q}$ and $U \subseteq V$, we write $w^2(U) := \sum_{u \in U} w^2(u)$.

Definition 4 (*Local improvement, claw-shaped improvement* [3]) Given an undirected graph $G = (V, E)$, a weight function $w : V \rightarrow \mathbb{Q}_{>0}$ and an independent set $A \subseteq V$, we say that a vertex set $B \subseteq V$ *improves* $w^2(A)$ *if* B is a *local improvement* of A if B is independent in G and we have

$$w^2((A \setminus N(B, A)) \cup B) > w^2(A).$$

We define a 0-claw to consist of a single talon and an empty center. For a d -claw C in G with $d \geq 0$, we say that C improves $w^2(A)$ if T_C does and call T_C a *claw-shaped improvement* in this case.

Note that in contrast to the introduction, we do not require a local improvement B to be disjoint from A anymore. However, it is not hard to see that B defines a local/claw-shaped improvement of A if and only if $B \setminus A$ does. Further observe that an independent set B improves A if and only if we have $w^2(B) > w^2(N(B, A))$.

Using the notation introduced above, Berman's algorithm SquareImp [3] can now be formulated as in Algorithm 1.

As all weights are positive, every $v \notin A$ such that $A \cup \{v\}$ is independent constitutes the talon of a 0-claw improving $w^2(A)$, so the algorithm returns a maximal independent set.

Theorem 5 ([3]) *Let $k \geq 3$, let $G = (V, E)$ be a $k + 1$ -claw free graph and let $w : V \rightarrow \mathbb{Q}_{>0}$. Let further $A \subseteq V$ be independent such that no claw improves A , and let A^* be an independent set of maximum weight in G .*

Then $w(A^) \leq \frac{k+1}{2} \cdot w(A)$.*

In particular, Theorem 5 implies that SquareImp obtains an approximation guarantee of $\frac{k+1}{2}$. The main idea of the proof of Theorem 5 presented in [3] is to charge the vertices in A for preventing adjacent vertices in an optimum solution A^* from being included into A . The latter is done by spreading the weight of the vertices in A^* among

their neighbors in the maximal independent set A in such a way that no vertex in A receives more than $\frac{k+1}{2}$ times its own weight. The suggested distribution of weights proceeds in two steps:

First, each vertex $u \in A^*$ invokes costs of $\frac{w(v)}{2}$ at each $v \in N(u, A)$.

In a second step, each $u \in A^*$ sends the remaining amount of $w(u) - \frac{w(N(u,A))}{2}$ to a heaviest neighbor it possesses in A , which is captured by the following definition of charges:

Definition 6 (*Charges* [3]) For each $u \in A^*$, pick a vertex $v \in N(u, A)$ of maximum weight and call it $n(u)$ (recall that A is maximal).

For $u \in A^*$ and $v \in A$, define

$$\text{charge}(u, v) := \begin{cases} w(u) - \frac{w(N(u,A))}{2}, & \text{if } v = n(u) \\ 0, & \text{otherwise} \end{cases}.$$

Definition 6 directly implies the following statement.

Proposition 7 *In the situation of Theorem 5, we have*

$$w(A^*) = \underbrace{\sum_{u \in A^*} \sum_{v \in N(u,A)} \frac{w(v)}{2}}_{\text{weight distributed in step 1}} + \underbrace{\sum_{u \in A^*} \text{charge}(u, n(u))}_{\text{weight distributed in step 2}}.$$

In order to prove Theorem 5, it suffices to see that each vertex $v \in A$ receives at most $\frac{k+1}{2}$ times its own weight. To prove this, we analyze the two steps of the weight distribution separately and show that in the first step, each $v \in A$ has to pay at most $\frac{k}{2} \cdot w(v)$, while in the second step, the total charges v has to pay can be bounded by $\frac{w(v)}{2}$. In doing so, we roughly follow [3], but introduce some new notation that we will need later for the analysis of our improved algorithm.

Looking at the first step of the weight distribution, the total weight a vertex $v \in A$ receives equals

$$\sum_{u \in N(v, A^*)} \frac{w(v)}{2} = \frac{|N(v, A^*)|}{2} \cdot w(v) = \frac{k}{2} \cdot w(v) - \frac{k - |N(v, A^*)|}{2} \cdot w(v).$$

In particular, we can bound the total weight distributed in the first step by

$$\sum_{u \in A^*} \sum_{v \in N(u,A)} \frac{w(v)}{2} = \frac{k}{2} \cdot w(A) - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v). \tag{1}$$

Proposition 8 further bounds the right-hand side by $\frac{k}{2} \cdot w(A)$ as claimed.

Proposition 8 *In the situation of Theorem 5, we have $|N(v, A^*)| \leq k$ for every $v \in A$.*

Proof As G is $k + 1$ -claw free, no vertex $v \in A \setminus A^*$ can have more than k neighbors in the independent set A^* . Moreover, if $v \in A \cap A^*$, then $N(v, A^*) = \{v\}$ and $|N(v, A^*)| = 1 \leq k$. \square

The weight distributed in the second step equals

$$\sum_{u \in A^*} \text{charge}(u, n(u)) = \sum_{v \in A} \sum_{u \in A^*: v=n(u)} \text{charge}(u, v).$$

Our next goal is to show that the total amount of charges a vertex $v \in A$ has to pay is bounded by $\frac{w(v)}{2}$. To this end, we need to exploit the fact that when SquareImp terminates, there is no improving claw centered at v . Suppose that we want to construct an improving claw C centered at v and consider adding $u \in N(v, A^*)$ to its set T_C of talons. On the one hand, this increases $w^2(T_C)$ by $w^2(u)$. On the other hand, $w^2(N(T_C, A))$ may also increase by up to $w^2(N(u, A) \setminus \{v\})$ (if our claw should be centered at v , we have to pay for v anyways). In case $w^2(u) > w^2(N(u, A) \setminus \{v\})$, we surely want to add u to our claw, otherwise, we may choose not to. This is captured by the definition of the *contribution*:

Definition 9 (*Contribution* [17]) Define a contribution map $\text{contr} : A^* \times A \rightarrow \mathbb{Q}_{\geq 0}$ by setting

$$\text{contr}(u, v) := \begin{cases} \max \left\{ 0, \frac{w^2(u) - w^2(N(u, A) \setminus \{v\})}{w(v)} \right\}, & \text{if } v \in N(u, A) \\ 0, & \text{otherwise} \end{cases}.$$

The fact that there is no improving claw directly implies that the total contribution to $v \in A$ is bounded by $w(v)$, which is the statement of Proposition 10.

Proposition 10 *In the situation of Theorem 5, we have*

$$\sum_{u \in A^*: n(u)=v} \text{contr}(u, v) \leq \sum_{u \in A^*} \text{contr}(u, v) \leq w(v)$$

for every $v \in A$. In particular, $\sum_{u \in A^*} \text{contr}(u, n(u)) \leq w(A)$.

Proof For $v \in A^*$, the independence of both A and A^* implies

$$\{u \in A^* : \text{contr}(u, v) > 0\} \subseteq N(v, A^*) = \{v\} = N(v, A)$$

and, hence,

$$\sum_{u \in A^*} \text{contr}(u, v) \leq \text{contr}(v, v) = \max \left\{ 0, \frac{w^2(v) - w^2(N(v, A) \setminus \{v\})}{w(v)} \right\} = w(v).$$

Assume towards a contradiction that there is $v \in A \setminus A^*$ such that

$$\sum_{u \in A^*} \text{contr}(u, v) > w(v).$$

Then

$$T := \{u \in A^* : \text{contr}(u, v) > 0\} \subseteq N(v, A^*)$$

forms the set of talons of a claw centered at v . As $w(v) > 0$, we obtain

$$\begin{aligned} w^2(v) &< w(v) \cdot \sum_{u \in A^*} \text{contr}(u, v) = w(v) \cdot \sum_{u \in T} \text{contr}(u, v) \\ &= \sum_{u \in T} w^2(u) - w^2(N(u, A) \setminus \{v\}) \leq w^2(T) - w^2(N(T, A) \setminus \{v\}). \end{aligned}$$

Hence, $w^2(N(T, A)) < w^2(T)$, contradicting the fact that no claw improves $w^2(A)$. \square

We will further show that for all $u \in A^*$, we have

$$2 \cdot \text{charge}(u, n(u)) \leq \text{contr}(u, n(u)).$$

This implies that $\sum_{u \in A^*: v=n(u)} \text{charge}(u, n(u)) \leq \frac{w(v)}{2}$ for every $v \in A$ as claimed. However, for the analysis of our algorithm, where we are not only interested in the statement of Theorem 5 itself, but would also like to know when this is close to being tight, it is more convenient to write

$$\begin{aligned} \sum_{u \in A^*} \sum_{v \in A} \text{charge}(u, v) &= \sum_{u \in A^*} \text{charge}(u, n(u)) \\ &= \frac{1}{2} \cdot \sum_{u \in A^*} \text{contr}(u, n(u)) - \sum_{u \in A^*} \left[\frac{1}{2} \cdot \text{contr}(u, n(u)) - \text{charge}(u, n(u)) \right] \\ &\stackrel{\text{Prop. 10}}{\leq} \frac{w(A)}{2} - \sum_{u \in A^*} \left[\frac{1}{2} \cdot \text{contr}(u, n(u)) - \text{charge}(u, n(u)) \right], \end{aligned} \tag{2}$$

and show that the total slack $\sum_{u \in A^*} \left[\frac{1}{2} \cdot \text{contr}(u, n(u)) - \text{charge}(u, n(u)) \right]$ we have in this estimate is non-negative. This is taken care of by Definition 11 and Lemma 12.

Definition 11 (*Slack*) In the situation of Theorem 5, for $u \in A^*$, we define

$$\text{slack}(u) := \frac{1}{2} \cdot \text{contr}(u, n(u)) - \text{charge}(u, n(u)).$$

Lemma 12 In the situation of Theorem 5, we have $\text{slack}(u) \geq 0$ for every $u \in A^*$.

We prove Lemma 13, a slightly stronger statement which we will need again at a later point. It directly implies Lemma 12 since all weights are strictly positive, meaning that we can divide the inequality $2 \cdot \text{slack}(u) \cdot w(v) \geq 0$ by $2 \cdot w(v)$.

Lemma 13 *Let $u \in A^*$ and let $v := n(u)$. Then*

$$\begin{aligned}
 2 \cdot \text{slack}(u) \cdot w(v) &\geq (w(u) - w(v))^2 \\
 &\quad + \sum_{y \in N(v, A) \setminus \{v\}} (w(v) - w(y)) \cdot w(y) \\
 &\geq (w(u) - w(v))^2 + (w(N(u, A)) - w(v)) \cdot (w(v) \\
 &\quad - \max_{y \in N(u, A) \setminus \{v\}} w(y)) \\
 &\geq (w(u) - w(v))^2 \geq 0,
 \end{aligned}$$

where $\max \emptyset := 0$.

Proof We have

$$\begin{aligned}
 2 \cdot \text{slack}(u) \cdot w(v) &= (\text{contr}(u, v) - 2 \cdot \text{charge}(u, v)) \cdot w(v) \\
 &\geq w^2(u) - w^2(N(u, A) \setminus \{v\}) - 2 \cdot w(u) \cdot w(v) + w(N(u, A)) \cdot w(v) \\
 &= (w(u) - w(v))^2 + (w(N(u, A)) - w(v)) \cdot w(v) - w^2(N(u, A) \setminus \{v\}) \\
 &= (w(u) - w(v))^2 + \sum_{y \in N(u, A) \setminus \{v\}} (w(v) - w(y)) \cdot w(y) \\
 &\geq (w(u) - w(v))^2 + (w(N(u, A)) - w(v)) \cdot (w(v) - \max_{y \in N(u, A) \setminus \{v\}} w(y)) \\
 &\geq (w(u) - w(v))^2 \geq 0.
 \end{aligned}$$

Then penultimate inequality follows since $v = n(u) \in N(u, A)$ is of maximum weight and all weights are positive. \square

We have now assembled all ingredients to prove Theorem 14, which in particular implies Theorem 5.

Theorem 14 *Let $k \geq 3$, let $G = (V, E)$ be a $k + 1$ -claw free graph and let $w : V \rightarrow \mathbb{Q}_{>0}$. Let further $A \subseteq V$ be independent such that no claw improves A , and let A^* be an independent set of maximum weight in G . Then*

$$\begin{aligned}
 w(A^*) &\leq \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \sum_{v \in A} \sum_{u \in A^*: v=n(u)} \text{contr}(u, v) \\
 &\quad - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) - \sum_{u \in A^*} \text{slack}(u) \\
 &\leq \frac{k+1}{2} \cdot w(A) \\
 &\quad - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) - \sum_{u \in A^*} \text{slack}(u) \leq \frac{k+1}{2} \cdot w(A).
 \end{aligned}$$

Algorithm 2: LogImp

Input: an undirected $k + 1$ -claw free graph $G = (V, E)$, $w : V \rightarrow \mathbb{Q}_{>0}$

Output: an independent set $A \subseteq V$

- 1 $A \leftarrow \emptyset$
 - 2 **while** there exists a claw-shaped or circular local improvement X of $w^2(A)$ **do**
 - 3 $A \leftarrow (A \setminus N(X, A)) \cup X$
 - 4 **return** A
-

Proof

$$\begin{aligned}
 w(A^*) &\stackrel{\text{Prop. 7}}{=} \sum_{u \in A^*} \sum_{v \in N(u, A)} \frac{w(v)}{2} + \sum_{u \in A^*} \text{charge}(u, n(u)) \\
 &\stackrel{(1)}{=} \frac{k}{2} \cdot w(A) - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) + \sum_{u \in A^*} \text{charge}(u, n(u)) \\
 &\stackrel{\text{Def. 11}}{=} \frac{k}{2} \cdot w(A) - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) \\
 &\quad + \sum_{u \in A^*} \left(\frac{1}{2} \cdot \text{contr}(u, n(u)) - \text{slack}(u) \right) \\
 &= \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \sum_{v \in A} \sum_{u \in A^*: v=n(u)} \text{contr}(u, v) \\
 &\quad - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) - \sum_{u \in A^*} \text{slack}(u) \\
 &\stackrel{\text{Prop. 10}}{\leq} \frac{k+1}{2} \cdot w(A) - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) - \sum_{u \in A^*} \text{slack}(u) \\
 &\stackrel{\text{Prop. 8}}{\leq} \frac{k+1}{2} \cdot w(A). \\
 &\stackrel{\text{Lem. 12}}{\leq} \frac{k+1}{2} \cdot w(A).
 \end{aligned}$$

□

This concludes the analysis of SquareImp.

4 Our algorithm LogImp

In this section, we introduce our algorithm *LogImp* for the MWIS in $k + 1$ -claw free graphs and analyze its performance ratio.

LogImp (Algorithm 2) is a local search algorithm that starts with the empty solution, and then iteratively checks for the existence of improving claws as in Berma’s algorithm SquareImp and another type of local improvement, which we call *circular*. It corresponds to a cycle of logarithmically bounded size in a certain auxiliary graph.

Definition 15 Let $G = (V, E)$ be a $k + 1$ -claw free graph, let $w : V \rightarrow \mathbb{Q}_{>0}$ and let $A \subseteq V$ be independent.

Let further two maps $n : \{u \in V : |N(u, A)| \geq 1\} \rightarrow A$ mapping u to an element of $N(u, A)$ of maximum weight, and $n_2 : \{u \in V : |N(u, A)| \geq 2\} \rightarrow A$ mapping u to an element of $N(u, A) \setminus \{n(u)\}$ of maximum weight be given. We call an independent set $X \subseteq \{u \in V \setminus A : |N(u, A)| \geq 1\}$ a *circular improvement* if

- (i) $\exists U \subseteq X$ s.t. $|U| \leq 8 \cdot \lceil \log(|V|) \rceil$, each $u \in U$ has at least two neighbors in A and $C := (\bigcup_{u \in U} \{n(u), n_2(u)\}, \{e_u = \{n(u), n_2(u)\}, u \in U\})$ is a cycle.
- (ii) If we let $Y_v := \{x \in X \setminus U : n(x) = v\}$, then $X = U \cup \bigcup_{v \in V(C)} Y_v$.
- (iii) For every $u \in U$, we have

$$\begin{aligned}
 &w^2(u) + \frac{1}{2} \cdot \left(\sum_{z \in Y_{n(u)}} w^2(z) + \sum_{z \in Y_{n_2(u)}} w^2(z) \right) \\
 &> \frac{w^2(n(u)) + w^2(n_2(u))}{2} + w^2(N(u, A) \setminus \{n(u), n_2(u)\}) \\
 &\quad + \frac{1}{2} \cdot \left(\sum_{z \in Y_{n(u)}} w^2(N(z, A) \setminus \{n(u)\}) + \sum_{z \in Y_{n_2(u)}} w^2(N(z, A) \setminus \{n_2(u)\}) \right).
 \end{aligned}$$

Proposition 16 *If X is a circular improvement and U, C and $Y_v, v \in V(C)$ are as in Definition 15, then we have $|Y_v| \leq k$ for all $v \in V(C)$. In particular, we obtain $|X| \leq 8 \cdot (k + 1) \cdot \lceil \log(|V|) \rceil$.*

Proof As X is independent and $Y_v \subseteq N(v, V \setminus A)$ by definition of Y_v and the map n , we know that for every $v \in V(C)$, Y_v constitutes the set of talons of a claw centered at v , provided it is non-empty. Hence, $|Y_v| \leq k$ for all $v \in V(C)$. In particular, we obtain

$$|X| = |U| + \sum_{v \in V(C)} |Y_v| \leq |U| + k \cdot |V(C)| = (k + 1) \cdot |U| \leq 8 \cdot (k + 1) \cdot \lceil \log(|V|) \rceil,$$

where $|U| = |V(C)|$ follows by definition of C . □

Proposition 17 *Every circular improvement X satisfies $w^2(X) > w^2(N(X, A))$ and, thus, constitutes a local improvement.*

Proof Let X be a circular improvement and let U, C and $Y_v, v \in V(C)$ as in Definition 15. We compute

$$\begin{aligned}
 w^2(X) &= w^2(U) + \sum_{v \in V(C)} w^2(Y_v) \stackrel{(*)}{=} \sum_{u \in U} \\
 &\quad \left[w^2(u) + \frac{1}{2} \cdot \left(\sum_{z \in Y_{n(u)}} w^2(z) + \sum_{z \in Y_{n_2(u)}} w^2(z) \right) \right]
 \end{aligned}$$

$$\begin{aligned}
 & \stackrel{\text{(iii)}}{>} \sum_{u \in U} \left[\frac{w^2(n(u)) + w^2(n_2(u))}{2} + w^2(N(u, A) \setminus \{n(u), n_2(u)\}) \right] \\
 & + \frac{1}{2} \cdot \sum_{u \in U} \left[\sum_{z \in Y_{n(u)}} w^2(N(z, A) \setminus \{n(u)\}) + \sum_{z \in Y_{n_2(u)}} w^2(N(z, A) \setminus \{n_2(u)\}) \right] \\
 & \stackrel{(*)}{=} \sum_{v \in V(C)} \left[w^2(v) + \sum_{z \in Y_v} w^2(N(z, A) \setminus \{v\}) \right] + \sum_{u \in U} w^2(N(u, A) \setminus \{n(u), n_2(u)\}) \\
 & \geq w^2(V(C)) + \sum_{v \in V(C)} w^2(N(Y_v, A) \setminus V(C)) + \sum_{u \in U} w^2(N(u, A) \setminus V(C)) \\
 & \geq w^2(N(X, A)).
 \end{aligned}$$

The equations labeled $(*)$ follow from the fact that

$$C = \left(\bigcup_{u \in U} \{n(u), n_2(u)\}, \{e_u = \{n(u), n_2(u)\}, u \in U\} \right)$$

forms a cycle, meaning that each $v \in V(C)$ occurs exactly twice among all of the sets $\{n(u), n_2(u)\}, u \in U$. □

The intuition behind Definition 15 is the following: Similar to the state-of-the-art algorithm by Fürer and Yu [10] for the unweighted case, we build up an auxiliary graph H on the vertex set A , where each vertex $u \in V \setminus A$ with at least two neighbors in A induces an edge between its two heaviest ones $n(u)$ and $n_2(u)$. The backbone of our circular improvement is given by a cycle C of logarithmic size in H . Additionally, for each $v \in V(C)$, we can add some additional vertices u with $n(u) = v$ that contribute a positive amount to v . Now, we want to cover for the weight of each $v \in V(C) \subseteq N(X, A)$ by using the vertices corresponding to the two incident edges in C as well as the contributions from the vertices in Y_v . More precisely, for each edge induced by $u \in U$, we would like $w^2(u)$, together with half of the contributions to $n(u)$ and $n_2(u)$, to be able to pay for all neighbors of u in A other than $n(u)$ and $n_2(u)$, as well as half of $n(u)$ and $n_2(u)$, which is precisely the constraint (iii).

When LogImp terminates, there cannot be any further improving claw, so we can apply the analysis of SquareImp presented in Sect. 3 to our algorithm. Similar to [17], the key idea in analyzing LogImp is the following: Either the analysis of SquareImp is far enough from being tight to achieve the desired approximation ratio, or the instance at hand bears a certain structure that allows us to derive the existence of a circular improvement. But this contradicts the termination criterion of LogImp.

The remainder of this section is dedicated to the proof of Theorem 18, which bounds the approximation guarantee obtained by LogImp. In Sect. 5, we then derive our main result, Theorem 1, by showing how to implement each iteration of LogImp in polynomial time, provided the input graph is the conflict graph of an (explicitly given) instance of the weighted k -Set Packing problem, and by explaining how to obtain a polynomial number of iterations by scaling and truncating the weight function.

Theorem 18 For $k \geq 3$, *LogImp* yields a $\frac{k+\delta_k}{2}$ -approximation for the MWIS in $k + 1$ -claw free graphs, where

$$\delta_k := \begin{cases} 1, & k = 3 \\ 1 - 2 \cdot 10^{-4} \cdot (k - 3.59), & 4 \leq k \leq 2153 \\ 0.57 + \frac{1.55}{k}, & 2154 \leq k \leq 5007 \\ \frac{40.25}{\sqrt{k}} + \frac{3.59}{k}, & k > 5007 \end{cases} \tag{3}$$

We have $\lim_{k \rightarrow \infty} \delta_k = 0$ and a simple calculation shows $\delta_k \in (0, 1)$ for all $k \geq 4$. We further point out that for $k \geq 4$, we obtain an approximation guarantee strictly better than the ratio of $\frac{k+1}{2}$ obtained by *SquareImp*, but also than the improvement to $\frac{k+1}{2} - \frac{1}{63,700,993}$ (or $\frac{k+1}{2} - \frac{1}{63,700,992} + \epsilon$, to be precise) by Neuwöhner [17].

For $k = 3$, Theorem 18 is implied by Theorem 5. To see why *LogImp* cannot yield an improvement over the approximation guarantee of $\frac{k+1}{2}$ for $k = 3$, consider a graph G that consists of a cycle of even length $2m$ the vertices of which alternately belong to the solution A that *LogImp* may find vertex by vertex and an optimum solution A^* . Moreover, each vertex from A has an additional neighbor from A^* of degree 1. All weights are 1. As the maximum degree in G equals 3, G is clearly 4-claw free. Furthermore, G is (isomorphic to) the conflict graph of the instance of the weighted 3-Set Packing problem we obtain by identifying each vertex with its set of incident edges.

We have $w(A^*) = |A^*| = 2 \cdot m = 2 \cdot |A| = \frac{3+1}{2} \cdot w(A)$. It is not hard to see that no claw improves A .² But for $m \geq 40$, there also is no circular improvement because the set U from Definition 15 would need to consist of all vertices from A^* that have two neighbors in A . But there are m such vertices and since $|V| = 3 \cdot m \geq 120$, we have $m = \frac{|V|}{3} > 8 \cdot (\log(|V|) + 1)$. Hence, there exists no circular improvement, either.

Now, we deal with the case $k \geq 4$. We introduce two auxiliary constants ϵ'_k and $\tilde{\epsilon}_k$ and we consider vertex weights to be “close” if they differ by a factor of at most $1 \pm \sqrt{\epsilon'_k}$. Moreover, for a vertex $v \in A$, $\tilde{\epsilon}_k \cdot w(v)$ constitutes a threshold telling us whether the total contribution to v should be regarded as small or large. We choose our constants to be

$$\epsilon'_k := \begin{cases} 10^{-3}, & 4 \leq k \leq 2153 \\ \frac{5 \cdot (1 - 18 \cdot 10^{-\frac{3}{2}})}{k}, & 2154 \leq k \leq 5007 \\ \frac{5}{k}, & 5007 < k \end{cases} \quad \text{and} \quad \tilde{\epsilon}_k := 18 \cdot \sqrt{\epsilon'_k}. \tag{4}$$

Our choices of ϵ'_k , $\tilde{\epsilon}_k$ and δ_k satisfy a bunch of inequalities that pop up during our analysis and are listed in Sect. 4.4. However, the main intuition the reader should have about them is that $0 < \epsilon'_k \ll \sqrt{\epsilon'_k} < \tilde{\epsilon}_k < \delta_k < 1$.

² By symmetry, one can, for example, just pick one vertex from A as a potential center and check all 8 subsets of its neighbors in $A^* = V \setminus A$.

For the analysis, we fix an instance (G, w) of the MWIS in $k + 1$ -claw free graphs consisting of a $k + 1$ -claw free graph $G = (V, E)$ and a positive weight function $w : V \rightarrow \mathbb{Q}_{>0}$. Denote the solution returned by LogImp when applied to (G, w) by A , and let A^* be an optimum solution for this instance. Finally, fix two maps n and n_2 according to Definition 15. We point out that A is maximal when LogImp terminates since there is no claw-shaped improvement. This implies that the domain of n is V .

4.1 Classification of vertices from A^*

We now provide a classification of the vertices in A^* that helps us to understand the structural properties of near-tight instances.

Definition 19 Let $u \in A^*$. We call u *contributive* if

$$\text{slack}(u) \geq \frac{\epsilon'_k}{10} \cdot w(N(u, A)).$$

If u is not contributive, but

$$\frac{w(u)}{w(n(u))} \in \left[1 - \sqrt{\epsilon'_k}, 1 + \sqrt{\epsilon'_k} \right] \text{ and } w(N(u, A)) \leq \left(1 + \sqrt{\epsilon'_k} \right) \cdot w(n(u)),$$

we call u *single*. Finally, if u is neither contributive nor single, and $|N(u, A)| \geq 2$,

$$\frac{w(u)}{w(n(u))} \in \left[1 - \sqrt{\epsilon'_k}, 1 + \sqrt{\epsilon'_k} \right], \frac{w(n_2(u))}{w(n(u))} \in \left[1 - \sqrt{\epsilon'_k}, 1 \right], \text{ and}$$

$w(N(u, A)) < (2 + \epsilon'_k) \cdot w(u)$, we call u *double*.

For a *contributive* vertex u , we gain a constant fraction of the weight of its neighborhood in A in our bound on $w(A^*)$ (cf. Theorem 14). For a *single* vertex u , the heaviest neighbor $n(u)$ of u in A has almost the same weight as u and makes up almost all of $N(u, A)$ in terms of weight. Single vertices correspond to vertices of degree 1 to A in the unit weight case.

For a *double* vertex u , its two heaviest neighbors in A , $n(u)$ and $n_2(u)$, have roughly the same weight as u and make up most of $N(u, A)$. Double vertices can be thought of as having degree 2 to A .

Lemma 20 Each $u \in A^*$ is single, double, or contributive.

Proof Pick $u \in A^*$ and let $v := n(u)$.

Case 1: $w(N(u, A)) \geq (2 + \epsilon'_k) \cdot w(u)$.

By non-negativity of the contribution, we obtain

$$\begin{aligned} 2 \cdot \text{slack}(u) &= \text{contr}(u, v) - 2 \cdot \text{charge}(u, v) \geq 0 - (2 \cdot w(u) - w(N(u, A))) \\ &= w(N(u, A)) \cdot \left(1 - \frac{2 \cdot w(u)}{w(N(u, A))} \right) \geq w(N(u, A)) \cdot \left(1 - \frac{2}{2 + \epsilon'_k} \right) \end{aligned}$$

$$= \frac{\epsilon'_k}{2 + \epsilon'_k} \cdot w(N(u, A)) \stackrel{(19)}{\geq} \frac{\epsilon'_k}{5} \cdot w(N(u, A)).$$

This implies that u is contributive.

Case 2: $w(N(u, A)) < (2 + \epsilon'_k) \cdot w(u)$. If u is contributive, we are done, so assume that this is not the case. In particular,

$$\text{slack}(u) < \frac{\epsilon'_k}{10} \cdot w(N(u, A)) < \epsilon'_k \cdot \frac{2 + \epsilon'_k}{10} \cdot w(u) \stackrel{(19)}{\leq} \frac{\epsilon'_k}{4} \cdot w(u).$$

We further claim that $w(v) \geq \frac{w(u)}{2}$, implying

$$\text{slack}(u) < \frac{\epsilon'_k}{2} \cdot w(v). \quad (5)$$

Indeed, if we had $w(v) < \frac{w(u)}{2}$, then Lemma 13 would yield

$$\text{slack}(u) \geq \frac{(w(u) - w(v))^2}{2 \cdot w(v)} > \frac{\frac{w(u)}{2} \cdot w(v)}{2 \cdot w(v)} = \frac{w(u)}{4} \stackrel{(19)}{>} \frac{\epsilon'_k}{4} \cdot w(u),$$

a contradiction. By combining (5), multiplied with $2 \cdot w(v)$, and Lemma 13, we obtain

$$(w(u) - w(v))^2 + (w(N(u, A)) - w(v)) \cdot (w(v) - \max_{y \in N(u, A) \setminus \{v\}} w(y)) < \epsilon'_k \cdot w^2(v).$$

Note that both summands on the left hand side are non-negative since real squares are non-negative, all weights are positive, and $v = n(u) \in N(u, A)$ is of maximum weight. In particular, both

$$\epsilon'_k \cdot w^2(v) \geq (w(u) - w(v))^2 \quad \text{and} \quad (6)$$

$$\epsilon'_k \cdot w^2(v) \geq (w(N(u, A)) - w(v)) \cdot (w(v) - \max_{y \in N(u, A) \setminus \{v\}} w(y)) \quad (7)$$

need to hold. From (6), we can infer that $|w(u) - w(v)| \leq \sqrt{\epsilon'_k} \cdot w(v)$, which in turn implies that

$$\frac{w(u)}{w(v)} \in \left[1 - \sqrt{\epsilon'_k}, 1 + \sqrt{\epsilon'_k} \right]. \quad (8)$$

In addition to that, (7) tells us that at least one of the two inequalities

$$\sqrt{\epsilon'_k} \cdot w(v) \geq w(N(u, A)) - w(v) \quad \text{or} \quad (9)$$

$$\sqrt{\epsilon'_k} \cdot w(v) \geq w(v) - \max_{y \in N(u, A) \setminus \{v\}} w(y) \quad (10)$$

must hold. If we have (9), then together with (8) and our assumption that u is not contributive, we may conclude that u is single. Thus, we are left with the case where u is neither contributive nor single, and (10) applies. The fact that $\epsilon'_k < 1$ by (19), together with $w(v) > 0$, implies that $N(u, A) \setminus \{v\} \neq \emptyset$, so let $v_2 := n_2(u)$. Then $w(v) - w(v_2) \leq \sqrt{\epsilon'_k} \cdot w(v)$ and, hence, $(1 - \sqrt{\epsilon'_k}) \cdot w(v) \leq w(v_2) \leq w(v)$ by maximality of $w(v)$ in $N(u, A)$. Thus, together with (8) and our case assumption $w(N(u, A)) < (2 + \epsilon'_k) \cdot w(u)$, all conditions for u being double are fulfilled. \square

4.2 Missing, profitable and helpful vertices

We now discuss the role the different types of vertices in A^* play in our analysis. We start by recalling that in the first step of the weight distribution in the analysis of SquareImp, each $v \in A$ pays $|N(v, A^*)| \cdot \frac{w(v)}{2}$, which we bound by $k \cdot \frac{w(v)}{2}$. In particular, if the number of neighbors of v in A^* happens to be less than k (we say that v has $k - |N(v, A^*)|$ missing neighbors in this case), we gain $\frac{w(v)}{2}$ for each missing neighbor of v .

We partition the neighbors a vertex $v \in A$ has in A^* into those that are helpful to v , and those that are profitable for v . While helpful vertices are those vertices that would be considered neighbors of v in an unweighted approximation of our instance, and that help us to construct local improvements of logarithmic size, profitable vertices are the remaining neighbors that in some sense keep the instance from being close to unweighted and hence, tight. Therefore, they improve the analysis (i.e. our bound on $w(A^*)$ profits from these).

Definition 21 Let $u \in A^*$ and $v \in N(u, A)$. We say that u is helpful to v if

- u is single and $v = n(u)$, or if
- u is double and $v \in \{n(u), n_2(u)\}$.

Otherwise, we call u profitable for v . Moreover, for $u \in A^*$, we define

$$\text{help}(u) := \{v \in N(u, A) : u \text{ is helpful to } v\},$$

and for $v \in A$, we let $\text{helped}(v) := \{u \in A^* : v \in \text{help}(u)\}$.

One can show that for each profitable neighbor that a vertex $v \in A$ possesses in A^* , we gain a constant fraction of $w(v)$ in bounding $w(A^*)$. Intuitively, this is because v increases $\text{slack}(u)$ for every profitable neighbor u of v . Lemma 22 formalizes the way missing and profitable neighbors improve our bound on $w(A^*)$.

Lemma 22

$$\begin{aligned} w(A^*) &\leq \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \sum_{v \in A} \sum_{u \in A^* : v = n(u)} \text{contr}(u, v) \\ &\quad - \sum_{v \in A} \underbrace{(k - |\text{helped}(v)|)}_{\substack{\# \text{ missing or profitable} \\ \text{neighbors of } v}} \cdot \frac{\epsilon'_k}{10} \cdot w(v) \end{aligned}$$

Note that Proposition 10 tells us that $\sum_{u \in A^*: v=n(u)} \text{contr}(u, v) \leq w(v)$ for $v \in A$. To prove Lemma 22, we provide a lower bound on the slack of a vertex in A^* (see Lemma 23). Then, we distribute the slack among the vertices in $N(u, A) \setminus \text{help}(u)$ proportional to their weight.

Lemma 23 *Let $u \in A^*$. Then $\text{slack}(u) \geq \frac{\epsilon'_k}{10} \cdot w(N(u, A) \setminus \text{help}(u))$.*

Proof of Lemma 22, assuming Lemma 23 We calculate

$$\begin{aligned}
 & \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) + \sum_{u \in A^*} \text{slack}(u) \\
 \stackrel{(19)}{\geq} & \sum_{v \in A} (k - |N(v, A^*)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) + \sum_{u \in A^*} \frac{\epsilon'_k}{10} \cdot w(N(u, A) \setminus \text{help}(u)) \\
 = & \sum_{v \in A} (k - |N(v, A^*)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) + \sum_{u \in A^*} \sum_{v \in N(u, A) \setminus \text{help}(u)} \frac{\epsilon'_k}{10} \cdot w(v) \\
 = & \sum_{v \in A} (k - |N(v, A^*)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) \\
 & + \sum_{v \in A} (|N(v, A^*)| - |\text{helped}(v)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) \\
 = & \sum_{v \in A} (k - |\text{helped}(v)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v).
 \end{aligned}$$

By Theorem 14, we, hence, obtain

$$\begin{aligned}
 w(A^*) & \leq \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \sum_{v \in A} \sum_{\substack{u \in A^* \\ v=n(u)}} \text{contr}(u, v) \\
 & \quad - \sum_{v \in A} \frac{k - |N(v, A^*)|}{2} \cdot w(v) - \sum_{u \in A^*} \text{slack}(u) \\
 & \leq \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \sum_{v \in A} \sum_{\substack{u \in A^* \\ v=n(u)}} \text{contr}(u, v) \\
 & \quad - \sum_{v \in A} (k - |\text{helped}(v)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v).
 \end{aligned}$$

□

Proof of Lemma 23 By Lemma 20, it suffices to verify the statement for the three cases where u is contributive, double, or single, respectively.

Case 1: u is contributive. Then $\text{slack}(u) \geq \frac{\epsilon'_k}{10} \cdot w(N(u, A))$ by definition.

Case 2: u is double. Let $v_1 := n(u)$ and $v_2 := n_2(u)$. Then $\text{help}(u) = \{v_1, v_2\}$. By Lemma 13, we calculate

$$\begin{aligned}
 2 \cdot \text{slack}(u) \cdot w(v_1) &\geq (w(u) - w(v_1))^2 + \sum_{y \in N(u, A) \setminus \{v_1\}} (w(v_1) - w(y)) \cdot w(y) \\
 &\geq \sum_{y \in N(u, A) \setminus \{v_1, v_2\}} (w(v_1) - w(y)) \cdot w(y) \\
 &\geq \sum_{y \in N(u, A) \setminus \{v_1, v_2\}} (w(v_1) - w(N(u, A) \setminus \{v_1, v_2\})) \cdot w(y) \\
 &= (w(v_1) - w(N(u, A) \setminus \{v_1, v_2\})) \cdot w(N(u, A) \setminus \{v_1, v_2\}) \\
 &= (w(v_1) - w(N(u, A) \setminus \{v_1, v_2\})) \cdot w(N(u, A) \setminus \text{help}(u)). \tag{11}
 \end{aligned}$$

The second and third inequality follow from the facts that all weights are positive and that v_1 is of maximum weight in $N(u, A)$. As u is double, we further know that

$$\begin{aligned}
 w(N(u, A) \setminus \{v_1, v_2\}) &= w(N(u, A)) - w(v_1) - w(v_2) \\
 &\leq (2 + \epsilon'_k) \cdot w(u) - w(v_1) - \left(1 - \sqrt{\epsilon'_k}\right) \cdot w(v_1) \\
 &\leq (2 + \epsilon'_k) \cdot \left(1 + \sqrt{\epsilon'_k}\right) \cdot w(v_1) - w(v_1) - \left(1 - \sqrt{\epsilon'_k}\right) \cdot w(v_1) \\
 &= \left(\epsilon'_k \cdot \left(1 + \sqrt{\epsilon'_k}\right) + 3 \cdot \sqrt{\epsilon'_k}\right) \cdot w(v_1).
 \end{aligned}$$

Plugging this into (11) yields

$$\begin{aligned}
 \text{slack}(u) &\geq \frac{1}{2} \cdot \left(1 - \epsilon'_k \cdot \left(1 + \sqrt{\epsilon'_k}\right) - 3 \cdot \sqrt{\epsilon'_k}\right) \cdot \\
 w(N(u, A) \setminus \text{help}(u)) &\stackrel{(20)}{\geq} \frac{\epsilon'_k}{10} \cdot w(N(u, A) \setminus \text{help}(u)).
 \end{aligned}$$

Case 3: u is single.

Let $v_1 := n(u)$. Then $\text{help}(u) = N(u, A) \setminus \{v_1\}$ and $w(N(u, A)) \leq (1 + \sqrt{\epsilon'_k}) \cdot w(v_1)$ by definition of single. By Lemma 13, we obtain

$$\begin{aligned}
 2 \cdot \text{slack}(u) \cdot w(v_1) &\geq (w(u) - w(v_1))^2 + (w(v_1) - \max_{y \in N(u, A) \setminus \{v_1\}} w(y)) \cdot w(N(u, A) \setminus \{v_1\}) \\
 &\geq (w(u) - w(v_1))^2 + (w(v_1) - w(N(u, A) \setminus \{v_1\})) \cdot w(N(u, A) \setminus \{v_1\}) \\
 &= (w(u) - w(v_1))^2 + (2 \cdot w(v_1) - w(N(u, A))) \cdot w(N(u, A) \setminus \{v_1\}) \\
 &\geq \left(1 - \sqrt{\epsilon'_k}\right) \cdot w(v_1) \cdot w(N(u, A) \setminus \{v_1\})
 \end{aligned}$$

$$= \left(1 - \sqrt{\epsilon'_k}\right) \cdot w(v_1) \cdot w(N(u, A) \setminus \text{help}(u)).$$

Division by $2 \cdot w(v_1) > 0$ yields

$$\text{slack}(u) \geq \frac{1 - \sqrt{\epsilon'_k}}{2} \cdot w(N(u, A) \setminus \text{help}(u)) \stackrel{(20)}{\geq} \frac{\epsilon'_k}{10} \cdot w(N(u, A) \setminus \text{help}(u))$$

as claimed. \square

4.3 Local improvements of logarithmic size

While missing and profitable neighbors improve our bound on $w(A^*)$, our goal is to use helpful neighbors towards a local improvement of logarithmic size. In order to get a better idea of what we are aiming at, we take a brief detour and recapitulate how these vertices are handled in the unit weight case. In [10], an auxiliary graph G_A is constructed, the vertices of which correspond to the vertices in the current solution A . Each vertex from an optimum solution A^* with exactly one neighbor in A creates a loop on that neighbor, while a vertex from A^* with exactly two neighbors in A results in an edge connecting these. Now, it is not hard to see that there is a one-to-one correspondence between local improvements only featuring vertices from A^* with degree one or two to A , and subgraphs of G_A with more edges than vertices. A minimal such subgraph is called a *binocular* [4]. Now, a result by Berman and Fürer [4] comes into play:

Lemma 24 ([4]) *Let $s \in \mathbb{N}_{>0}$ and let $G = (V, E)$ be a graph with $|E| \geq \frac{s+1}{s} \cdot |V|$. Then G contains a binocular with at most $4 \cdot s \cdot \lceil \log(|V|) \rceil$ vertices.*

In particular, Lemma 24 implies the existence of a cycle of logarithmically bounded size. Moreover, if the number of vertices from A^* of degree 1 or 2 to A exceeds $(1 + \epsilon) \cdot |A|$, one can find local improvements of size $\mathcal{O}(\frac{1}{\epsilon} \cdot \log(|A|))$, which is one of the key ingredients of the result in [10].

We would like to port this approach to our weighted setting where vertex weights are locally similar. If we could ensure that for each double vertex u , its squared weight is at least as large as the average squared weight of its neighbors $n(u)$ and $n_2(u)$, plus the squared weight of the neighbors of u in A other than $n(u)$ and $n_2(u)$, then we would be done since in a binocular, every vertex has degree at least two, and there is at least one vertex of degree more than two. However, this need not be the case in general, and even if we only lose an ϵ'_k fraction of the weight of each vertex involved, the total loss might be arbitrarily large if we consider improvements of logarithmic size. In order to overcome this issue, we have to make sure that locally, we have some additional vertices with a positive contribution to the endpoints of the edges that we can add to guarantee that we can make up for the slight inaccuracies caused by the weight differences. This idea is captured by the following definition.

Definition 25 (T_v and B) For $v \in A$, let

$$T_v := \{u \in A^* : n(u) = v \wedge \text{contr}(u, v) > 0\}.$$

(Note that $T_v \subseteq N(v, A^*)$ and that $n(u) = v$ for all $u \in T_v$, implying that the sets $(T_v)_{v \in A}$ are pairwise disjoint.) We define

$$B := \left\{ v \in A : \sum_{u \in T_v} \text{contr}(u, v) > \tilde{\epsilon}_k \cdot w(v) \right\}.$$

For a vertex $v \in B$, we can use the contributions of the vertices in T_v to cover for the inaccuracies caused by local weight differences as discussed before. In order to make sure that the neighborhood of our local improvement in A features sufficiently many vertices from B , we will only allow edges in our auxiliary graph that have at least one endpoint in B . The idea is that then each (sufficiently short) cycle in our auxiliary graph gives rise to a circular improvement by choosing $Y_v := T_v \setminus U$ for $v \in B$, and $Y_v := \emptyset$ otherwise, where U denotes the set of vertices from $V \setminus A$ inducing the edges of the cycle. To find such a cycle, we need to see that the weighted sum of the number of incident edges (induced by helpful vertices) of vertices in B is large enough. Lemma 26 tells us that if

$$\sum_{v \in B} w(v) \cdot |\text{helped}(v)| \leq \left(\frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} + 1 \right) \cdot w(A),$$

then we reach the desired approximation guarantee of $\frac{k+\delta_k}{2}$ stated in Theorem 18. On the other hand, Lemma 27 shows that if this condition is not fulfilled, we can indeed find a circular improvement, contradicting the termination criterion of LogImp. This then concludes the analysis.

Lemma 26 *If*

$$\sum_{v \in B} w(v) \cdot |\text{helped}(v)| \leq \left(\frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} + 1 \right) \cdot w(A),$$

then $w(A^) \leq \frac{k+\delta_k}{2} \cdot w(A)$ holds.*

Proof For $v \in A$, we get $\sum_{u \in A^*: v=n(u)} \text{contr}(u, v) = \sum_{u \in T_v} \text{contr}(u, v)$ by non-negativity of the contribution. For $v \in A \setminus B$, we have

$$\sum_{\substack{u \in A^* \\ v=n(u)}} \text{contr}(u, v) = \sum_{u \in T_v} \text{contr}(u, v) \leq \tilde{\epsilon}_k \cdot w(v),$$

while for $v \in B$, Proposition 10 tells us that

$$\sum_{\substack{u \in A^* \\ v=n(u)}} \text{contr}(u, v) = \sum_{u \in T_v} \text{contr}(u, v) \leq w(v).$$

By Lemma 22, we obtain

$$\begin{aligned}
 w(A^*) &\leq \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \sum_{v \in A} \sum_{u \in A^*: v=n(u)} \text{contr}(u, v) \\
 &\quad - \sum_{v \in A} (k - |\text{helped}(v)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) \\
 &\leq \frac{k}{2} \cdot w(A) + \frac{\tilde{\epsilon}_k}{2} \cdot w(A \setminus B) + \frac{1}{2} \cdot w(B) - \sum_{v \in A} (k - |\text{helped}(v)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) \\
 [1] \quad &\stackrel{\text{Prop. 8}}{\leq} \frac{k + \tilde{\epsilon}_k}{2} \cdot w(A) + \frac{1 - \tilde{\epsilon}_k}{2} \cdot w(B) - \sum_{v \in B} (k - |\text{helped}(v)|) \cdot \frac{\epsilon'_k}{10} \cdot w(v) \\
 &= \frac{k + \tilde{\epsilon}_k}{2} \cdot w(A) + \left(\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \right) \cdot w(B) + \frac{\epsilon'_k}{10} \cdot \sum_{v \in B} w(v) \cdot |\text{helped}(v)| \\
 &\leq \left(\frac{k + \tilde{\epsilon}_k}{2} + \frac{\epsilon'_k}{10} \cdot \left(\frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} + 1 \right) \right) \cdot w(A) \\
 &\quad + \left(\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \right) \cdot w(B).
 \end{aligned}$$

If $\left(\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \right) \geq 0$, using $w(B) \leq w(A)$ as $B \subseteq A$, the latter can be upper bounded by

$$\begin{aligned}
 &\left(\frac{k + 1}{2} - \frac{\epsilon'_k}{10} \cdot \left(k - \frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} - 1 \right) \right) \cdot w(A) \\
 &\stackrel{(19)}{\leq} \left(\frac{k + 1}{2} - \frac{\epsilon'_k}{10} \cdot \left(k - \frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} - 1 \right) \right) \cdot w(A).
 \end{aligned}$$

On the other hand, if $\left(\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \right) \leq 0$, we obtain an upper bound of

$$\begin{aligned}
 &\left(\frac{k + \tilde{\epsilon}_k}{2} + \frac{\epsilon'_k}{10} \cdot \left(\frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} + 1 \right) \right) \cdot w(A) \\
 &\stackrel{(19)}{\leq} \left(\frac{k + \tilde{\epsilon}_k}{2} + \frac{\epsilon'_k}{10} \cdot \left(\frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} + 1 \right) \right) \cdot w(A)
 \end{aligned}$$

since $w(B) \geq 0$. For $4 \leq k \leq 2153$, we have

$$\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \stackrel{(4)}{\geq} \frac{1}{2} - 9 \cdot 10^{-\frac{3}{2}} - 2153 \cdot 10^{-4} > 0.215395 - 0.2153 > 0.$$

Hence,

$$\begin{aligned} w(A^*) &\leq \left(\frac{k+1}{2} - \frac{\epsilon'_k}{10} \cdot \left(k - \frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} - 1 \right) \right) \cdot w(A) \\ &\stackrel{(4)}{=} \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \left(1 - \frac{10^{-3}}{5} \cdot \left(k - \frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} - 1 \right) \right) \cdot w(A) \\ &< \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \left(1 - 2 \cdot 10^{-4} \cdot (k - 3.59) \right) \cdot w(A) \stackrel{(3)}{=} \frac{k + \delta_k}{2} \cdot w(A). \end{aligned}$$

For $2154 \leq k \leq 5007$, we obtain $\tilde{\epsilon}_k = 18 \cdot \sqrt{\epsilon'_k} \stackrel{(19)}{\leq} 18 \cdot 10^{-\frac{3}{2}}$, which implies

$$\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \stackrel{(4)}{\geq} \frac{1 - 18 \cdot 10^{-\frac{3}{2}}}{2} - \frac{1 - 18 \cdot 10^{-\frac{3}{2}}}{2} = 0.$$

Thus, we may conclude that

$$\begin{aligned} w(A^*) &\leq \left(\frac{k+1}{2} - \frac{\epsilon'_k}{10} \cdot \left(k - \frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} - 1 \right) \right) \cdot w(A) \\ &= \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \left(1 - \frac{\epsilon'_k \cdot k}{5} + \frac{\epsilon'_k \cdot k}{5} \cdot \left(\frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} + 1 \right) \cdot \frac{1}{k} \right) \cdot w(A) \\ &\stackrel{(4)}{=} \frac{k}{2} \cdot w(A) \\ &\quad + \frac{1}{2} \cdot \left(18 \cdot 10^{-\frac{3}{2}} + (1 - 18 \cdot 10^{-\frac{3}{2}}) \cdot \left(\frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} + 1 \right) \cdot \frac{1}{k} \right) \cdot w(A) \\ &< \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \left(0.57 + \frac{1.55}{k} \right) \cdot w(A) \stackrel{(3)}{=} \frac{k + \delta_k}{2} \cdot w(A). \end{aligned}$$

Finally, for $k > 5007$, we have

$$\frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \epsilon'_k}{10} \stackrel{(4)}{=} \frac{1 - \tilde{\epsilon}_k}{2} - \frac{k \cdot \frac{5}{k}}{10} = -\frac{\tilde{\epsilon}_k}{2} < 0,$$

so

$$\begin{aligned}
 w(A^*) &\leq \left(\frac{k + \tilde{\epsilon}_k}{2} + \frac{\epsilon'_k}{10} \cdot \left(\frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} + 1 \right) \right) \cdot w(A) \\
 &\stackrel{(4)}{=} \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \left(18 \cdot \sqrt{\frac{5}{k}} + \left(\frac{5}{2 \cdot (1 - 10^{-\frac{3}{2}})} + 1 \right) \cdot \frac{1}{k} \right) \cdot w(A) \\
 &< \frac{k}{2} \cdot w(A) + \frac{1}{2} \cdot \left(\frac{40.25}{\sqrt{k}} + \frac{3.59}{k} \right) \cdot w(A) \stackrel{(3)}{=} \frac{k + \delta_k}{2} \cdot w(A).
 \end{aligned}$$

This concludes the proof. \square

Lemma 27 *If*

$$\sum_{v \in B} w(v) \cdot |\text{helped}(v)| > \left(\frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} + 1 \right) \cdot w(A),$$

then there exists a circular improvement.

To prove Lemma 27, we first state Lemma 28, which tells us that any graph in which the average degree is by some constant factor larger than 2 contains a cycle of logarithmically bounded size. It is implied by the proof of Lemma 3.1 in [4].

Lemma 29 then provides a generalization to a setting where we are given positive vertex weights and only have a bound on the average weighted degree, but know that the weights of neighboring vertices can only differ by a constant factor.

We employ it in our proof of Lemma 27 to derive the existence of a cycle of logarithmically bounded size in a certain auxiliary graph, which then gives rise to a circular improvement.

Lemma 28 *Let $s \in \mathbb{N}_{>0}$ and let $G = (V, E)$ be an undirected graph with $|V| \geq 2$ such that $|E| \geq \frac{s+1}{s} \cdot |V|$. Then G contains a cycle of size at most $2 \cdot s \cdot \lceil \log(|V|) \rceil$.*

Lemma 29 *Let $G = (V, E)$ be an undirected graph with $|V| \geq 2$ equipped with positive vertex weights $w : V(G) \rightarrow \mathbb{Q}_{>0}$.*

Let further $\alpha \geq 1$ such that for every edge $\{u, v\} \in E(G)$, we have

$$\alpha^{-1} \cdot w(u) \leq w(v) \leq \alpha \cdot w(u),$$

and let $s \in \mathbb{N}$ such that

$$\sum_{v \in V} w(v) \cdot |\delta(v)| \geq \frac{s+1}{s} \cdot 2 \cdot \alpha \cdot \sum_{v \in V} w(v),$$

where $\delta(v)$ denotes the set of incident edges of v in G . Then G contains a cycle of length at most $2 \cdot s \cdot \lceil \log(|V|) \rceil$.

Proof If G contains a loop, we are done since $\log(|V|) \geq \log(2) = 1$, so assume that this is not the case. By Lemma 28, it suffices to show that G contains a non-empty subgraph G' with $|E(G')| \geq \frac{s+1}{s} \cdot |V(G')|$. Note that each such subgraph satisfies $|V(G')| \geq 2$ by our assumption that G does not contain any loop. Let $w_{max} := \max_{v \in V} w(v)$. For $x \in (0, \infty)$, let $V_{\geq x} := \{v \in V : w(v) \geq x\}$ be the set of vertices of weight at least x , and let $G_{\geq x} := (V_{\geq x}, E_{\geq x}) := G[V_{\geq x}]$ denote the subgraph of G induced by $V_{\geq x}$. Observe that by definition of w_{max} , $V_{\geq x} \neq \emptyset$ for $x \in (0, w_{max})$. We claim that there is $x \in (0, w_{max})$ such that $|E_{\geq x}| \geq \frac{s+1}{s} \cdot |V_{\geq x}|$. Assume towards a contradiction that this were not the case, i.e.

$$\forall x \in (0, w_{max}) : |E_{\geq x}| < \frac{s+1}{s} \cdot |V_{\geq x}|. \tag{12}$$

By our assumption on α , we know that for every $x \in (0, \infty)$ and every $v \in V_{\geq \alpha \cdot x}$, the neighborhood of v in G is contained in $V_{\geq x}$. Hence, we obtain

$$\forall x \in (0, w_{max}) : 2 \cdot |E_{\geq x}| = \sum_{v \in V_{\geq x}} |\delta(v) \cap E_{\geq x}| \geq \sum_{v \in V_{\geq \alpha \cdot x}} |\delta(v)|. \tag{13}$$

This yields

$$\begin{aligned} & 2 \cdot \frac{s+1}{s} \cdot \sum_{v \in V} w(v) \leq \sum_{v \in V} \alpha^{-1} \cdot w(v) \cdot |\delta(v)| = \sum_{v \in V} |\delta(v)| \cdot \int_0^{\alpha^{-1} \cdot w(v)} 1 \, dx \\ & \stackrel{\alpha \geq 1}{=} \sum_{v \in V} |\delta(v)| \cdot \int_0^{w_{max}} \mathbf{1}_{\alpha^{-1} \cdot w(v) \geq x} \, dx = \sum_{v \in V} |\delta(v)| \cdot \int_0^{w_{max}} \mathbf{1}_{w(v) \geq \alpha \cdot x} \, dx \\ & = \int_0^{w_{max}} \sum_{v \in V} \mathbf{1}_{w(v) \geq \alpha \cdot x} \cdot |\delta(v)| \, dx = \int_0^{w_{max}} \sum_{v \in V_{\geq \alpha \cdot x}} |\delta(v)| \, dx \\ & \stackrel{(13)}{\leq} \int_0^{w_{max}} 2 \cdot |E_{\geq x}| \, dx \stackrel{(12)}{<} \int_0^{w_{max}} 2 \cdot \frac{s+1}{s} \cdot |V_{\geq x}| \, dx \\ & = 2 \cdot \frac{s+1}{s} \cdot \int_0^{w_{max}} \sum_{v \in V} \mathbf{1}_{w(v) \geq x} \, dx = 2 \cdot \frac{s+1}{s} \cdot \sum_{v \in V} \int_0^{w_{max}} \mathbf{1}_{w(v) \geq x} \, dx \\ & = 2 \cdot \frac{s+1}{s} \cdot \sum_{v \in V} w(v), \end{aligned}$$

i.e. $2 \cdot \frac{s+1}{s} \cdot \sum_{v \in V} w(v) < 2 \cdot \frac{s+1}{s} \cdot \sum_{v \in V} w(v)$, a contradiction. Note that when applying (12) in the fourth line, we indeed inherit the strict inequality because both $|E_{\geq x}|$ and $|V_{\geq x}|$ are constant on the interval $(0, \min_{v \in V} w(v))$ of positive length. \square

Proof of Lemma 27 Let $v \in B$. We show that there is at most one vertex $u \in N(v, A^*)$ such that u is single and $v = n(u)$. Observe that for each such vertex u , we have

$$\text{contr}(u, v) \cdot w(v) \geq w^2(u) - w^2(N(u, A) \setminus \{v\})$$

$$\geq \left(1 - \sqrt{\epsilon'_k}\right)^2 \cdot w^2(v) - \left(\sqrt{\epsilon'_k} \cdot w(v)\right)^2 = \left(1 - 2\sqrt{\epsilon'_k}\right) \cdot w^2(v) \stackrel{(22)}{>} \frac{w^2(v)}{2},$$

so $\text{contr}(u, v) > \frac{w(v)}{2}$. But by Proposition 10, the total contribution to $v \in B \subseteq A$ is bounded by $w(v)$ since there is no claw-shaped improvement when LogImp terminates. Hence, there can be at most one such vertex u since the contribution is non-negative.

Moreover, recall that $v \in \text{help}(u)$ if and only if u is single and $v = n(u)$, or u is double and $v \in \{n(u), n_2(u)\}$. As a consequence, we obtain

$$\begin{aligned} & \sum_{v \in B} w(v) \cdot |\{u \in N(v, A^*) : u \text{ double}, v \in \{n(u), n_2(u)\}\}| \\ &= \sum_{v \in B} w(v) \cdot |\{u \in N(v, A^*) : v \in \text{help}(u), u \text{ not single}\}| \\ &\geq \sum_{v \in B} w(v) \cdot (|\text{helped}(v)| - 1) \\ &> \left(\frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} + 1 \right) \cdot w(A) - w(B) \geq \frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} \cdot w(A). \end{aligned}$$

Now, we want to apply Lemma 29. We consider the following graph G^* :

- $V(G^*) := A$
- $E(G^*) := \{\{n(u), n_2(u)\} : u \text{ is double} \wedge \{n(u), n_2(u)\} \cap B \neq \emptyset\}$

The vertex weights are given by w . By definition of being double, we can choose $\alpha := \left(1 - \sqrt{\epsilon'_k}\right)^{-1}$. Then

$$\begin{aligned} & \sum_{v \in V(G^*)} w(v) \cdot |\delta_{G^*}(v)| \geq \sum_{v \in B} w(v) \cdot |\delta_{G^*}(v)| \\ &\geq \sum_{v \in B} w(v) \cdot |\{u \in N(v, A^*) : u \text{ double}, v \in \{n(u), n_2(u)\}\}| \\ &> \frac{5}{2 \cdot (1 - \sqrt{\epsilon'_k})} \cdot w(A) = \frac{5}{2} \cdot \alpha \cdot w(A) = \frac{4+1}{4} \cdot 2 \cdot \alpha \cdot w(V(G^*)). \end{aligned}$$

In particular, as G^* does not contain any loops by definition, we must have $|V(G^*)| \geq 2$. Hence, we can apply Lemma 29 to obtain a cycle C of size at most $8 \cdot \lceil \log(|A|) \rceil \leq 8 \cdot \lceil \log(|V(G)|) \rceil$. We need to see how to get a circular local improvement out of C . To this end, denote the set of vertices from A^* corresponding to the edges of C by U_e , and define

$$X := U_e \cup \bigcup_{v \in V(C) \cap B} T_v.$$

Then $X \subseteq A^*$ is independent. Let $e = \{n(u), n_2(u)\} \in E(C)$ and let $\{v, z\} := e$ such that $v \in B$ (and we do not make any assumptions on whether or not $z \in B$). This is possible since every edge of G^* and hence of C must intersect B by definition. We claim that

$$w^2(u) + \frac{1}{2} \cdot w^2(T_v \setminus U_e) > \frac{w^2(v) + w^2(z)}{2} + w^2(N(u, A) \setminus \{v, z\}) + \frac{1}{2} \cdot \sum_{x \in T_v \setminus U_e} w^2(N(x, A) \setminus \{v\}). \tag{14}$$

We have $\sum_{x \in T_v} w^2(x) - w^2(N(x, A) \setminus \{v\}) \geq \tilde{\epsilon}_k \cdot w^2(v)$ by definition of B and T_v and the contribution. Moreover, we know that each vertex $x \in T_v \cap U_e$ satisfies $n(x) = v$ since it is in T_v , and is double since it is in U_e . Hence,

$$\begin{aligned} w^2(x) - w^2(N(x, A) \setminus \{v\}) &\leq w^2(x) - w^2(n_2(x)) \\ &\leq \left(1 + \sqrt{\epsilon'_k}\right)^2 \cdot w^2(v) - \left(1 - \sqrt{\epsilon'_k}\right)^2 \cdot w^2(v) = 4 \cdot \sqrt{\epsilon'_k} \cdot w^2(v). \end{aligned}$$

Next, as the edge in $E(C)$ that $x \in T_v \cap U_e$ induces is $\{n(x), n_2(x)\} = \{v, n_2(x)\}$, this edge must be incident to v . But in the cycle C , v has exactly two incident edges. Therefore, $|T_v \cap U_e| \leq 2$ and

$$\sum_{x \in T_v \setminus U_e} w^2(x) - w^2(N(x, A) \setminus \{v\}) \geq \left(\tilde{\epsilon}_k - 8\sqrt{\epsilon'_k}\right) \cdot w^2(v) \stackrel{(4)}{\geq} 0. \tag{15}$$

We have $e = \{n(u), n_2(u)\} = \{v, z\}$, which implies

$$w(v) \geq \min\{w(n(u)), w(n_2(u))\} = w(n_2(u)).$$

As u is double, we either get $w(n_2(u)) \leq w(n(u)) \leq w(u)$ and, hence,

$$\frac{w^2(v) + w^2(z)}{2} = \frac{w^2(n(u)) + w^2(n_2(u))}{2} \leq w^2(u),$$

or

$$w(u) < w(n(u)) \leq \frac{1}{1 - \sqrt{\epsilon'_k}} \cdot w(u) \stackrel{(21)}{\leq} \left(1 + 2\sqrt{\epsilon'_k}\right) \cdot w(u)$$

and

$$\left(1 - \sqrt{\epsilon'_k}\right) \cdot w(u) < \left(1 - \sqrt{\epsilon'_k}\right) \cdot w(n(u)) \leq w(n_2(u)) \leq w(n(u)),$$

leading to

$$\begin{aligned} \frac{w(v)^2 + w(z)^2}{2} &= \frac{w(n(u))^2 + w(n_2(u))^2}{2} \leq \frac{2 + 8 \cdot \sqrt{\epsilon'_k} + 8\epsilon'_k}{2} \cdot w^2(u) \\ &= w^2(u) + \left(4\sqrt{\epsilon'_k} + 4\epsilon'_k\right) \cdot w^2(u) \leq w^2(u) + \frac{4\sqrt{\epsilon'_k} + 4\epsilon'_k}{\left(1 - \sqrt{\epsilon'_k}\right)^2} \cdot w^2(v). \end{aligned}$$

Hence, the weaker inequality

$$\frac{w(v)^2 + w(z)^2}{2} \leq w^2(u) + \frac{4\sqrt{\epsilon'_k} + 4\epsilon'_k}{\left(1 - \sqrt{\epsilon'_k}\right)^2} \cdot w^2(v) \quad (16)$$

holds in either case. Additionally, as

$$w(v) \geq w(n_2(u)) \geq \left(1 - \sqrt{\epsilon'_k}\right) \cdot w(n(u)) \geq \frac{1 - \sqrt{\epsilon'_k}}{1 + \sqrt{\epsilon'_k}} \cdot w(u) \geq \left(1 - \sqrt{\epsilon'_k}\right)^2 \cdot w(u)$$

and $w(N(u, A)) \leq (2 + \epsilon'_k) \cdot w(u)$ by definition of double, we get

$$\begin{aligned} w^2(N(u, A) \setminus \{v, z\}) &= w^2(N(u, A) \setminus \{n(u), n_2(u)\}) \\ &\leq (w(N(u, A)) - w(n(u)) - w(n_2(u)))^2 \end{aligned} \quad (17)$$

$$\begin{aligned} &\leq \left(2 + \epsilon'_k - \left(1 - \sqrt{\epsilon'_k}\right) - \left(1 - \sqrt{\epsilon'_k}\right)\right)^2 \cdot w^2(u) \\ &= \left(3\sqrt{\epsilon'_k}\right)^2 \cdot w^2(u) = 9 \cdot \epsilon'_k \cdot w^2(u) \leq 9 \cdot \epsilon'_k \cdot \frac{w^2(v)}{\left(1 - \sqrt{\epsilon'_k}\right)^4} \end{aligned} \quad (18)$$

This gives

$$\begin{aligned} &w^2(u) + \frac{1}{2} \sum_{x \in T_v \setminus U_e} w^2(x) - w^2(N(x, A) \setminus \{v\}) \\ &\stackrel{(15)}{\geq} w^2(u) + \frac{\tilde{\epsilon}_k}{2} \cdot w^2(v) - 4\sqrt{\epsilon'_k} \cdot w^2(v) \\ &\stackrel{(16), (18)}{\geq} \frac{w^2(v) + w^2(z)}{2} + w^2(N(u, A) \setminus \{v, z\}) + \frac{\tilde{\epsilon}_k}{2} \cdot w^2(v) \\ &\quad - \left(4\sqrt{\epsilon'_k} + \frac{4\sqrt{\epsilon'_k} + 4\epsilon'_k}{\left(1 - \sqrt{\epsilon'_k}\right)^2} + \frac{9\epsilon'_k}{\left(1 - \sqrt{\epsilon'_k}\right)^4}\right) \cdot w^2(v) \end{aligned}$$

$$> \frac{w^2(v) + w^2(z)}{2} + w^2(N(u, A) \setminus \{v, z\}),$$

where the last strict inequality follows from (23) and $w(v) > 0$ since weights are positive. This proves our claim (14).

Moreover, we have $|U_e| = |E(C)| \leq 8 \cdot \lceil \log(|V(G)|) \rceil$. Hence, choosing $U := U_e$ and $Y_v := T_v \setminus U_e$ for $v \in V(C) \cap B$ and $Y_v := \emptyset$ for $v \in V(C) \setminus B$, we see that our local improvement is circular by (14) and (15), provided it is a subset of $V \setminus A$. To this end, note that $V(C) \subseteq A \setminus A^*$ since vertices in $A \cap A^*$ are only adjacent to themselves in A^* , and, hence, isolated in G^* . As a consequence, as all vertices in $X = U_e \cup \bigcup_{v \in V(C) \cap B} T_v$ are adjacent to some $v \in V(C) \subseteq A \setminus A^*$, they cannot be contained in $A^* \cap A$ either, because in this case, they could only be adjacent to themselves in A and in particular not to a vertex in $A \setminus A^*$. Hence, $X \subseteq A^* \setminus A \subseteq V \setminus A$ defines a circular improvement. \square

4.4 Conditions satisfied by our choices of $\tilde{\epsilon}_k, \epsilon'_k$ and δ_k

The following constraints can be easily verified by plugging in our choices of $\tilde{\epsilon}_k, \epsilon'_k$ and δ_k . Due to page limit, we omit the calculations.

$$0 < \epsilon'_k \leq 10^{-3} < \frac{1}{20} \tag{19}$$

$$\frac{1 - \sqrt{\epsilon'_k}}{2} \geq \frac{1}{2} \cdot \left(1 - \epsilon'_k \cdot \left(1 + \sqrt{\epsilon'_k} \right) - 3 \cdot \sqrt{\epsilon'_k} \right) \geq \frac{\epsilon'_k}{10} \tag{20}$$

$$\frac{1}{1 - \sqrt{\epsilon'_k}} \leq 1 + 2 \cdot \sqrt{\epsilon'_k} \tag{21}$$

$$\frac{1}{2} < 1 - 2\sqrt{\epsilon'_k} \tag{22}$$

$$8 \cdot \sqrt{\epsilon'_k} \leq 8\sqrt{\epsilon'_k} + \frac{8\sqrt{\epsilon'_k} + 8\epsilon'_k}{(1 - \sqrt{\epsilon'_k})^2} + \frac{18\epsilon'_k}{(1 - \sqrt{\epsilon'_k})^4} < \tilde{\epsilon}_k \tag{23}$$

5 A polynomial running time

In this section, we complete the proof of Theorem 1 by showing how to obtain a polynomial running time for (conflict graphs of) instances of the weighted k -Set Packing problem. In doing so, Lemma 30, which we prove in Sect. 5.1, guarantees a polynomial number of iterations, while Lemma 31 tells us that we can implement each iteration of LogImp in polynomial time. We prove it in Sect. 5.2.

Lemma 30 *Let $k \geq 3$ and denote the approximation guarantee that LogImp achieves for the MWIS in $k + 1$ -claw free graphs by ρ . Given an instance (G, w) of the MWIS*

in $k + 1$ -claw free graphs and $N \in \mathbb{N}_{>0}$, we can, in polynomial time, compute an induced subgraph G' of G and a weight function w' with the following two properties:

- When applied to (G', w') , *LogImp* terminates after a number of iterations that is polynomial in $|V(G)|$ and N .
- The output of *LogImp* on (G', w') yields an $\frac{N}{N-1} \cdot \rho$ -approximation for the MWIS on (G, w) .

Lemma 31 *There is an algorithm, which, given a family \mathcal{S} consisting of sets each of cardinality at most k , a positive weight function $w : \mathcal{S} \rightarrow \mathbb{Q}_{>0}$ and $A \subseteq \mathcal{S}$ such that the sets in A are pairwise disjoint, in polynomial time either returns a claw-shaped or circular improvement X of A in $G_{\mathcal{S}}$, or decides that none exists.*

Proof of Theorem 1 For $k = 3$, we can, for example, run the polynomial time $\frac{k+1}{2} - \frac{1}{63,700,993}$ -approximation algorithm for the MWIS in $k + 1$ -claw free graphs given in [17], yielding $\epsilon_3 = 1 - \frac{2}{63,700,993}$. Now, let $k \geq 4$. Theorem 18 tells us that *LogImp* achieves an approximation guarantee of $\frac{k+\delta_k}{2}$ for the MWIS in $k + 1$ -claw free graphs, where $\delta_k \in (0, 1)$ and $\lim_{k \rightarrow \infty} \delta_k = 0$. Define

$$N_k := \max \left\{ (k+1)^2 + 1, \left\lceil \frac{k+1}{1-\delta_k} \right\rceil + 1 \right\}.$$

This is a constant that only depends on k , but not on the input instance. We can obtain an $\frac{N_k}{N_k-1} \cdot \frac{k+\delta_k}{2}$ -approximation for the weighted k -Set Packing problem by

- computing the conflict graph $G_{\mathcal{S}}$ of a given instance (\mathcal{S}, w) ,
- applying Lemma 30 to obtain an induced subgraph G' of $G_{\mathcal{S}}$ (which equals the conflict graph of the corresponding sub-instance of \mathcal{S}) and a weight function w' and
- finally running *LogImp* on (G', w') .

By Lemmas 30 and 31, we can perform each of these steps in polynomial time. Concerning the approximation guarantee, we have

$$\frac{N_k}{N_k-1} \cdot \frac{k+\delta_k}{2} = \left(1 + \frac{1}{N_k-1}\right) \cdot \frac{k+\delta_k}{2} = \frac{k}{2} + \frac{1}{2} \cdot \left(\delta_k + \frac{k+\delta_k}{N_k-1}\right) = \frac{k+\epsilon_k}{2},$$

where $\epsilon_k := \delta_k + \frac{1}{N_k-1} \cdot (k+\delta_k)$. Then $\delta_k \in (0, 1)$ and our choice of N_k imply

$$0 \leq \epsilon_k \leq \delta_k + \frac{1}{N_k-1} \cdot (k+1) \leq \delta_k + 1 - \delta_k = 1.$$

Moreover,

$$\epsilon_k \leq \delta_k + \frac{1}{N_k-1} \cdot (k+1) \leq \delta_k + \frac{1}{(k+1)^2} \cdot (k+1) \leq \delta_k + \frac{1}{k+1},$$

and $\lim_{k \rightarrow \infty} \delta_k = 0$ implies $\lim_{k \rightarrow \infty} \epsilon_k = 0$. This concludes the proof of Theorem 1. \square

5.1 A polynomial number of iterations

Proof of Lemma 30 Let $G = (V, E)$. We first apply the greedy algorithm, which considers the vertices of G by decreasing weight and picks every vertex that is not adjacent to an already chosen one. Denote the solution it returns by A' . It is known that A' is a k -approximate solution to the MWIS in (G, w) (see for example [5]). Now, we construct a weight function w_s from w by re-scaling it such that $w_s(A') = N \cdot |V|$ holds. Then A' also constitutes a k -approximate solution to the MWIS in (G, w_s) . Next, we define w' by $w'(v) := \lfloor w_s(v) \rfloor$ and obtain G' by deleting vertices with $w'(v) = 0$, which does not change the weight of an optimum solution (w.r.t. w'). Consider a run of LogImp on (G', w') and denote the solution that LogImp maintains by A . As w' is positive and integral, we know that $w'^2(A)$ must increase by at least 1 in every iteration except for the last one. On the other hand, at each point, we must have

$$w'^2(A) \leq w_s^2(A) \leq (w_s(A'))^2 \leq (k \cdot w_s(A'))^2 = k^2 \cdot N^2 \cdot |V|^2,$$

which bounds the total number of iterations by $k^2 \cdot N^2 \cdot |V|^2 + 1$. Finally, if A now denotes the solution LogImp returns and A^* is an independent set in G of maximum weight with respect to w (and w_s), we know that

$$\rho \cdot w_s(A) \geq \rho \cdot w'(A) \geq w'(A^*) \geq w_s(A^*) - |A^*| \geq w_s(A^*) - |V| \geq \frac{N-1}{N} \cdot w_s(A^*),$$

so $w_s(A^*) \leq \frac{N}{N-1} \cdot \rho \cdot w_s(A)$ and thus, also $w(A^*) \leq \frac{N}{N-1} \cdot \rho \cdot w(A)$. □

5.2 A polynomial running time per iteration

This section is dedicated to the proof of Lemma 31, which consists of two parts. First, Proposition 32 tells us that we can search for a claw-shaped improvement in polynomial time (for general $k + 1$ -claw free graphs). Then, Lemma 33 makes sure that for conflict graphs of instances of the weighted k -Set Packing problem, we can also find a circular improvement, or decide that none exists, in polynomial time.

Proposition 32 *Given a $k + 1$ -claw free graph $G = (V, E)$, a positive weight function $w : V \rightarrow \mathbb{Q}_{>0}$ and an independent set $A \subseteq V$, we can, in polynomial time (considering k a constant), either return a claw-shaped improvement, or decide that none exists.*

Proof We can check for the existence of a claw-shaped improvement in polynomial time by looping over all $\mathcal{O}((|V| + 1)^k) = \mathcal{O}(|V|^k)$ subsets X of V of size at most k . For each of them, we can check in polynomial time whether it is independent and whether it is claw-shaped (either $|X| = 1$ or all vertices in X have a common neighbor in A). Finally, we can check whether $w^2(X) > w^2(N(X, A))$ in polynomial time. □

It is not hard to see that we can search for a circular improvement in quasi-polynomial time. By Proposition 16, there are

$$\mathcal{O}\left((|V| + 1)^{8 \cdot (k+1) \cdot \lceil \log(|V|) \rceil}\right) = \mathcal{O}\left(|V|^{8 \cdot (k+1) \cdot (\log(|V|)+1)}\right)$$

many possible choices for X and we can check in polynomial time whether a given one constitutes a circular improvement. More precisely, we can first check whether X is independent, and then try all of the at most

$$2^{|X|} \leq 2^{8 \cdot (k+1) \cdot (\log(|V|)+1)} = (2 \cdot |V|)^{8 \cdot (k+1)}$$

possible choices for $U \subseteq X$ with $|U| \leq 8 \cdot \lceil \log(|V|) \rceil$. For each choice of U , we can check in polynomial time whether each vertex in U has at least two neighbors in A , and if so, compute $V(C)$ (cf. Def. 15). Then, the partition into the sets Y_v is uniquely determined by the map n , although it might happen that for some vertex $x \in X \setminus U$, $n(x)$ is not contained in $V(C)$, meaning that our choice of X and U does not meet the requirements of Definition 15. Finally, we can check the condition (iii) from Definition 15 in polynomial time.

For the general case of the MWIS in $k + 1$ -claw free graphs, it is not clear whether and how one can, in polynomial time, either find a circular improvement X , or decide that none exists. However, for the weighted k -Set Packing problem, which is the main application we have in mind, we will show in the remainder of this section how to perform this task in polynomial time. For this purpose, we employ the color coding technique similar as in [10] (but in a much simpler way).

Lemma 33 *There is an algorithm, which, given a family \mathcal{S} consisting of sets each of cardinality at most k , a positive weight function $w : \mathcal{S} \rightarrow \mathbb{Q}_{>0}$ and $A \subseteq \mathcal{S}$ such that the sets in A are pairwise disjoint and there is no claw-shaped improvement of A in $G_{\mathcal{S}}$, in polynomial time (considering k a constant) either returns a circular improvement X of A in $G_{\mathcal{S}}$, or decides that none exists.*

The remainder of this section is dedicated to the proof of Lemma 33. To simplify notation, we identify a set $S \in \mathcal{S}$ and its corresponding vertex in the $k + 1$ -claw free conflict graph $G_{\mathcal{S}} =: G =: (V, E)$.

In order to search for a circular improvement, we first compute two maps $n : V \setminus A \rightarrow A$ mapping u to an element of $N(u, A)$ of maximum weight, and $n_2 : \{u \in V \setminus A : |N(u, A)| \geq 2\} \rightarrow A$ mapping u to an element of $N(u, A) \setminus \{n(u)\}$ of maximum weight. These maps are guaranteed to exist since the assumption that there is no claw-shaped improvement implies that A constitutes a maximal independent set in $G_{\mathcal{S}}$. n and n_2 can be easily computed in polynomial time. We define an auxiliary multi-graph H as follows:

The vertices of H are pairs (v, Y) , where $v \in A$ and $Y \subseteq \{u \in V \setminus A : n(u) = v\}$ is an independent set of cardinality at most k . Then for each $v \in A$, there are at most $\sum_{i=0}^k |V|^i = \frac{|V|^{k+1}-1}{|V|-1} \in \mathcal{O}(|V|^k)$ many possible choices for Y , so we have $|V(H)| \in \mathcal{O}(|V|^{k+1})$.

As far as the set of edges is concerned, for each $u \in \{x \in V \setminus A : |N(x, A)| \geq 2\}$ and each pair of vertices $(n(u), Y_1), (n_2(u), Y_2) \in V(H)$ such that

$$w^2(u) + \frac{1}{2} \cdot \sum_{x \in Y_1} w^2(x) + \frac{1}{2} \cdot \sum_{x \in Y_2} w^2(x)$$

$$\begin{aligned}
 &> \frac{w^2(n(u)) + w^2(n_2(u))}{2} + w^2(N(u, A) \setminus \{n(u), n_2(u)\}) \\
 &\quad + \frac{1}{2} \cdot \sum_{x \in Y_1} w^2(N(x, A) \setminus \{n(u)\}) + \frac{1}{2} \cdot \sum_{x \in Y_2} w^2(N(x, A) \setminus \{n_2(u)\}),
 \end{aligned}$$

and $\{u\}$, Y_1 and Y_2 are pairwise disjoint and their union is independent, we add a copy of the edge $\{(n(u), Y_1), (n_2(u), Y_2)\}$ to the multi-set $E(H)$. We say that this edge is *induced by u* . As we have as most $|V|$ many possible choices for u and $\mathcal{O}(|V|^k)$ possibilities to choose each of Y_1 and Y_2 , we get $|E(H)| \in \mathcal{O}(|V|^{2k+1})$.

For convenience, let $\mu : E(H) \rightarrow V \setminus A$ such that $e \in E(H)$ is induced by the vertex $\mu(e)$.

We claim that there is a correspondence between circular improvements and cycles \bar{C} in H of length at most $8 \cdot \lceil \log(|V|) \rceil$ such that for every $v \in A$, there is at most one set Y with $(v, Y) \in V(\bar{C})$ and such that $\{\mu(e), e \in E(\bar{C})\} \cup \bigcup_{(v,Y) \in V(\bar{C})} Y$ defines an independent set a.k.a. a disjoint sub-family of \mathcal{S} , and the union is disjoint. More precisely, we prove the following statement:

Proposition 34 *An independent set $X \subseteq V \setminus A$ constitutes a circular improvement if and only if there is a cycle \bar{C} in H of length at most $8 \cdot \lceil \log(|V|) \rceil$ such that for every $v \in A$, there is at most one set Y with $(v, Y) \in V(\bar{C})$ and such that*

$$X = \{\mu(e), e \in E(\bar{C})\} \cup \bigcup_{(v,Y) \in V(\bar{C})} Y$$

and the union is disjoint.

Proof First, assume that X is a circular improvement. Then X constitutes an independent set. Let U, C and $Y_v, v \in V(C)$ be as in Definition 15. By (ii) and (iii) and since X is independent, we have $\bar{e}_u := \{(n(u), Y_{n(u)}), (n_2(u), Y_{n_2(u)})\} \in E(H)$ for every $u \in U$, and $\mu(\bar{e}_u) = u$. By (i), $\bar{C} := (\{(v, Y_v), v \in V(C)\}, \{\bar{e}_u, u \in U\})$ yields a cycle in H of length at most $8 \cdot \lceil \log(|V|) \rceil$. By construction, for every $v \in A$, there is at most one set Y with $(v, Y) \in V(\bar{C})$ and we have

$$X = U \cup \bigcup_{v \in V(C)} Y_v = \{\mu(e), e \in E(\bar{C})\} \cup \bigcup_{(v,Y) \in V(\bar{C})} Y.$$

By definition of the sets $Y_v, v \in V(C)$, this union is disjoint.

Now, let $X \subseteq V \setminus A$ be independent and let \bar{C} be a cycle in H as described in the proposition. Define $U := \{\mu(e) : e \in E(\bar{C})\}$. Then $|U| \leq 8 \cdot \lceil \log(|V|) \rceil$. By definition of H ,

$$C := \left(\bigcup_{u \in U} \{n(u), n_2(u)\}, \{e_u = \{n(u), n_2(u)\}, u \in U\} \right)$$

is a closed edge sequence because it arises from \bar{C} by projecting every vertex $(v, Y) \in V(\bar{C})$ to its first component. As for every $v \in A$, there is at most one set Y with

$(v, Y) \in V(\bar{C})$, C is a cycle, so condition (i) from Definition 15 holds. Moreover, as for every $(v, Y) \in V(\bar{C})$, we have $Y \subseteq \{u \in V \setminus A : n(u) = v\}$, we can infer that for every $(v, Y) \in V(\bar{C})$, we have $Y \cap X \setminus U = \{z \in X \setminus U : n(z) = v\} = Y_v$. In particular, $X = U \cup \bigcup_{v \in V(C)} Y = U \cup \bigcup_{v \in V(C)} Y_v$, which yields condition (ii). As the first union is disjoint, we can further conclude that $Y = Y_v$ for all $v \in V(C)$. Hence, the definition of $E(H)$ implies that condition (iii) from Definition 15 is satisfied. \square

It remains to see how to, in polynomial time, find a cycle in H with the properties described in Prop. 34, or decide that none exists.

First, for each pair of parallel edges in $E(H)$, we can check in polynomial time whether or not it yields such a cycle, so we can restrict ourselves to cycles of length at least 3 (and at most $8 \cdot \lceil \log(|V|) \rceil$) in the following.

To find these, we want to apply the *color coding* technique. For this purpose, we introduce the following terminology:

Definition 35 (*t*-perfect family of hash functions, [1]) For $t, m \in \mathbb{N}$ with $t \leq m$, a family $\mathcal{F} \subseteq \{1, \dots, m\}^{\{1, \dots, t\}}$ of functions mapping $\{1, \dots, m\}$ to $\{1, \dots, t\}$ is called a *t*-perfect family of hash functions if for all $I \subseteq \{1, \dots, m\}$ of size at most t , there is $f \in \mathcal{F}$ with $f \upharpoonright I$ injective.

Theorem 36 (Stated in [1] referring to [20]) For $t, m \in \mathbb{N}$ with $t \leq m$, a *t*-perfect family \mathcal{F} of hash functions of cardinality $2^{\mathcal{O}(t)} \cdot (\log(m))^2$, where each function is encoded using $\mathcal{O}(t) + 2 \log \log m$ many bits, can be explicitly constructed such that the query time is constant.

In order to find a cycle \bar{C} in H of logarithmically bounded size such that each vertex $v \in A$ occurs at most once as the first entry of a vertex $(z, Y) \in V(\bar{C})$ and $X := \{\mu(e), e \in E(\bar{C})\} \cup \bigcup_{(v, Y) \in V(\bar{C})} Y$ defines an independent set in G (i.e. a disjoint family of k -sets) and the union is disjoint, the idea is to color both the vertices in A and the elements of the k -sets corresponding to the vertices in X , and to replace the distinctness/disjointness constraints by the stronger property that for each of the vertices $(v, Y) \in V(\bar{C})$, v receives a different color, and moreover, the colors of the elements of the k -sets in X are pairwise distinct. We call a cycle with this property *colorful* (see Definition 37). If we only use a logarithmically bounded number of colors, then we can use dynamic programming to check for the existence of a colorful cycle. Moreover, we can show that if we pick our colorings from appropriately chosen families of perfect hash functions (of polynomial size), we can guarantee that for each cycle \bar{C} in H meeting the requirements of Prop. 34, there exist colorings from our families for which \bar{C} is colorful. This means that by looping over all colorings from our families and checking for a colorful cycle, we will find a circular improvement, if one exists.

Let \mathcal{F} be a $t_1 := 8 \cdot \lceil \log(|V|) \rceil$ perfect family of hash functions with domain $A \subseteq V$. There exists such a family of size

$$2^{\mathcal{O}(\lceil \log(|V|) \rceil)} \cdot \log(|V|)^2 = |V|^{\mathcal{O}(1)} \cdot \log(|V|)^2 = |V|^{\mathcal{O}(1)},$$

which is polynomial.

In addition, let \mathcal{G} be a $t_2 := 8 \cdot (k + 1) \cdot k \cdot \lceil \log(|V|) \rceil$ -perfect family of functions with domain $\bigcup \mathcal{S}$, i.e. the underlying universe of the k -Set Packing problem (or, more precisely, its restriction to the set of elements that appear in at least one set). Clearly, $|\bigcup \mathcal{S}| \leq k \cdot |\mathcal{S}| = k \cdot |V|$, so we obtain such a family \mathcal{G} of size

$$2^{\mathcal{O}(k^2 \cdot \lceil \log(|V|) \rceil)} \cdot (\log(k \cdot |V|))^2 = |V|^{\mathcal{O}(k^2)},$$

which is polynomial. For each pair of functions $(f, g) \in \mathcal{F} \times \mathcal{G}$, we do the following:

For every $v = (z, Y) \in V(H)$, we set

$$\text{col}^f(v) := f(z),$$

meaning that we color v with the color assigned to the vertex z by f . Additionally, we define

$$\text{col}^g(v) := g\left(\bigcup Y\right) = \{g(x) : \exists u \in Y : x \in u\},$$

i.e. we color v with all of the colors occurring among the elements of the k -sets represented by the vertices in Y .

Finally, we assign to an edge $e = \{(v_1, Y_1), (v_2, Y_2)\} \in E(H)$ the set of colors

$$\text{col}^g(e) := g(\mu(e)) = \{g(x), x \in \mu(e)\},$$

where we interpret $\mu(e)$ as the corresponding k -set.

Definition 37 Following [10], we call a path or cycle P in H *colorful* if

- The colors $\text{col}^f(v)$, $v \in V(P)$ are pairwise distinct.
- The color sets $\text{col}^g(e)$, $e \in E(P)$ and $\text{col}^g(v)$, $v \in V(P)$ are pairwise disjoint.

Lemma 38 Let \bar{C} be a cycle of length at least 3 and at most $8 \cdot \lceil \log(|V|) \rceil$ in H . Then the following are equivalent:

- There exist $f \in \mathcal{F}$ and $g \in \mathcal{G}$ for which \bar{C} is colorful.
- \bar{C} meets the requirements of Prop. 34.

Proof For a colorful cycle \bar{C} , the sets of colors assigned (by g) to the (union of) k -sets in the sets Y , $(v, Y) \in V(\bar{C})$ and the sets of colors assigned (by g) to the k -sets $\mu(e)$, $e \in E(\bar{C})$ are pairwise disjoint. In particular, the (unions of) k -sets in the list $\bigcup Y$, $(v, Y) \in V(\bar{C})$, $\mu(e)$, $e \in E(\bar{C})$ are pairwise disjoint. Moreover, when constructing H , we made sure that for each $(v, Y) \in V(H)$, Y is independent in $G_{\mathcal{S}}$, meaning that the contained k -sets are pairwise disjoint. Hence, the k -sets in $X := \bigcup_{(v, Y) \in V(\bar{C})} Y \cup \{\mu(e), e \in E(\bar{C})\}$ are pairwise disjoint and the union is disjoint. In particular, X constitutes an independent set in $G_{\mathcal{S}}$. Moreover, the colors $f(v)$, $(v, Y) \in V(\bar{C})$ are pairwise distinct, meaning that for each $v \in A$, there is at most one Y with $(v, Y) \in V(\bar{C})$. Hence, \bar{C} meets the requirements of Prop. 34.

On the other hand, if \bar{C} satisfies the conditions from Prop. 34, then we have $|V(\bar{C})| \leq 8 \cdot \lceil \log(|V|) \rceil = t_1$ and for each $v \in A$, there is at most one Y such that $(v, Y) \in V(\bar{C})$.

Hence, there is $f \in \mathcal{F}$ that assigns a different color to each one of the vertices $v, (v, Y) \in V(\bar{C})$. Moreover, the k -sets contained in the sets $Y, (v, Y) \in V(H)$ and the k -sets $\mu(e), e \in E(\bar{C})$ are pairwise distinct and pairwise disjoint. In addition, as every set Y with $(v, Y) \in V(H)$ contains at most k k -sets, the total number of elements from $\bigcup \mathcal{S}$ contained in the k -sets in the sets $Y, (v, Y) \in V(\bar{C})$ or the k -sets $\mu(e), e \in E(\bar{C})$ is bounded by

$$\left| \bigcup_{(v,Y) \in V(\bar{C})} Y \cup \bigcup_{e \in E(\bar{C})} \mu(e) \right| \leq k^2 \cdot |V(\bar{C})| + k \cdot |E(\bar{C})| \leq (k + 1) \cdot k \cdot 8 \cdot \lceil \log(|V|) \rceil = t_2.$$

Hence, there exists $g \in \mathcal{G}$ assigning distinct colors to all of these elements. Thus, \bar{C} is colorful for f and g . □

As the sizes of \mathcal{F} and \mathcal{G} are polynomially bounded, Lemma 39 concludes the proof of Lemma 33.

Lemma 39 *Let $f \in \mathcal{F}$ and $g \in \mathcal{G}$. We can, in polynomial time, either find a colorful cycle \bar{C} of length at least 3 and at most $8 \cdot \lceil \log(|V|) \rceil$ in H , or decide that none exists.*

Proof We observe that there is a colorful cycle \bar{C} meeting the above size bounds if and only if there exist an edge $e_0 = \{s, t\} \in E(H)$ and a colorful s - t -path P of length at least 2 and at most $8 \cdot \lceil \log(|V|) \rceil - 1$ such that $\text{col}^g(e_0)$ and $\bigcup_{e \in E(P)} \text{col}^g(e) \cup \bigcup_{v \in V(P)} \text{col}^g(v)$ are disjoint. For $s, t \in V(H)$, $C^f \subseteq \{1, \dots, t_1\}$, $C^g \subseteq \{1, \dots, t_2\}$ and $i \in \{0, \dots, 8 \cdot \lceil \log |V| \rceil - 1\}$, we define the Boolean value $\text{Path}(s, t, C^f, C^g, i)$ to be true if and only if there exists a colorful s - t -path P of length i with

$$\{\text{col}^f(v), v \in V(P)\} = C^f \text{ and } \bigcup_{e \in E(P)} \text{col}^g(e) \cup \bigcup_{v \in V(P)} \text{col}^g(v) = C^g.$$

We apply dynamic programming to compute these values in order of non-decreasing i , and we use backlinks to be able to retrace a corresponding path for each value that is set to true. The total number of values we have to compute is bounded by

$$\begin{aligned} & |V(H)|^2 \cdot 2^{t_1} \cdot 2^{t_2} \cdot 8 \cdot \log \lceil |V| \rceil \\ & \in \mathcal{O}(|V|^{2(k+1)} \cdot 2^{8 \cdot \lceil \log(|V|) \rceil} \cdot 2^{8 \cdot k \cdot (k+1) \cdot \lceil \log(|V|) \rceil} \cdot 8 \cdot \lceil \log |V| \rceil) = |V|^{\mathcal{O}(k^2)}, \end{aligned}$$

which is polynomial bounded. Hence, it suffices to show how to, in polynomial time, determine $\text{Path}(s, t, C^f, C^g, i)$, provided that all of the values $\text{Path}(s', t', C^{f'}, C^{g'}, i')$ with $i' < i$ are already known to us. We have $\text{Path}(s, t, C^f, C^g, 0) = \text{true}$ if and only if $s = t$, $C^f = \{\text{col}^f(s)\}$ and $C^g = \text{col}^g(s)$. For $i \geq 1$, we observe that $\text{Path}(s, t, C^f, C^g, i) = \text{true}$ if and only if $\text{col}^f(s) \in C^f$ and there exists an edge $e = \{s, v\} \in E(H)$ such that $\text{col}^g(s) \cap \text{col}^g(e) = \emptyset$, $\text{col}^g(s) \cup \text{col}^g(e) \subseteq C^g$ and $\text{Path}(v, t, C^f \setminus \{\text{col}^f(s)\}, C^g \setminus (\text{col}^g(s) \cup \text{col}^g(e)), i - 1) = \text{true}$. We can check this in polynomial time. □

6 The lower bound

In this section, we show that our result is asymptotically best possible in the sense of Theorem 2.

Theorem 2 *Let $k \geq 3$, $\alpha \in \mathbb{R}$, $0 < \epsilon < 1$ and $C > 0$. Then for each $N_0 \in \mathbb{N}$, there exists an instance (\mathcal{S}, w) of the weighted k -Set Packing problem with $|\mathcal{S}| \geq N_0$ and such that there are feasible solutions $A, A^* \subseteq \mathcal{S}$ with the following two properties:*

- *There is no local improvement of A of size at most $C \cdot \log(|\mathcal{S}|)$ with respect to w^α , that is, there is no disjoint sub-collection $X \subseteq \mathcal{S} \setminus A$ of sets such that*

$$|X| \leq C \cdot \log(|\mathcal{S}|) \text{ and } w^\alpha(X) > w^\alpha(\{a \in A : \exists x \in X : a \cap x \neq \emptyset\}).$$

- *We have $w(A^*) \geq \frac{k-\epsilon}{2} \cdot w(A)$.*

Proof For $\alpha \leq 0$, we can just choose the weights in A to be arbitrarily small. Hence, we can assume $\alpha > 0$. Now, there is a result by Erdős and Sachs [9] telling us that for every $N_1 \geq N_0$, there is a k -regular graph H on $|V(H)| \geq N_1$ vertices such that the girth of H , i.e. the minimum length of a cycle in H , is at least $\frac{\log(|V(H)|)}{\log(k-1)} - 1$. Consider the graph G with vertex set $V(G) := V(H) \cup E(H)$ and edge set $E(G) := \{\{v, e\} : v \in e \in E(H)\}$, i.e. each edge of H is connected via edges in G to both of its endpoints. We define $\mathcal{S} := \{\delta_G(x), x \in V(G)\}$, where $\delta_G(x)$ is the set of incident edges of x in G . By k -regularity of H , $|\delta_G(v)| = k \geq 3$ for $v \in V(H)$ and $|\delta_G(e)| = 2$ for $e \in E(H)$, so each element of \mathcal{S} has cardinality at most k . By definition, G is simple, so no two vertices share more than one edge. As all degrees are at least two, the sets $\delta_G(x), x \in V(G)$ are pairwise distinct. Finally, $V(H)$ and $E(H)$ constitute independent sets in G , implying that $A := \{\delta_G(v), v \in V(H)\}$ and $A^* := \{\delta_G(e), e \in E(H)\}$ each consist of pairwise disjoint sets. We define positive weights on \mathcal{S} by setting $w(\delta_G(v)) := 1$ for $v \in V(H)$ and $w(\delta_G(e)) := (1 - \bar{\epsilon})^{\frac{1}{\alpha}}$, where $\bar{\epsilon} > 0$ is chosen such that $\frac{1}{\bar{\epsilon}} \in \mathbb{N}$ and $(1 - \bar{\epsilon})^{\frac{1}{\alpha}} \geq 1 - \frac{\epsilon}{k}$. By k -regularity of H , $|A^*| = |E(H)| = \frac{k}{2} \cdot |V(H)| = \frac{k}{2} \cdot |A|$, so $w(A^*) \geq \frac{k-\epsilon}{2} \cdot w(A)$. To see that there is no local improvement $X \subseteq \mathcal{S} \setminus A$ of $w^\alpha(A)$ with

$$|X| \leq C \cdot \log(|\mathcal{S}|) = C \cdot \log(|V(G)|) = C \cdot \log\left(\frac{k+2}{2} \cdot |V(H)|\right),$$

we observe that X being a local improvement would imply

$$|X| > \frac{1}{1 - \bar{\epsilon}} \cdot |\{y \in A : \exists x \in X : y \cap x \neq \emptyset\}|$$

by our choice of weights. But the sets from A that X intersects are precisely the sets $\delta_G(v)$ for those vertices $v \in V(H)$ that are endpoints of edges $e \in E(H)$ with $\delta_G(e) \in X$. Hence, we have found a subgraph J of H with

$$C \cdot \log\left(\frac{k+2}{2} \cdot |V(H)|\right) \geq |X| = |E(J)| > \frac{1}{1 - \bar{\epsilon}} \cdot |V(J)| \geq (1 + \bar{\epsilon}) \cdot |V(J)|,$$

which implies the existence of a cycle of length $\frac{2}{\epsilon} \cdot \lceil \log(C \cdot \log(\frac{k+2}{2} \cdot |V(H)|)) \rceil$ by Lemma 28. But as a function of $|V(H)|$, this grows asymptotically slower than our lower bound of $\frac{\log(|V(H)|)}{\log(k-1)} - 1$ on the girth, resulting in a contradiction for N_1 and, hence, $|V(H)|$ chosen large enough. \square

7 Conclusion

In this paper, we have seen how to use local search to approximate the weighted k -Set Packing problem with an approximation ratio that gets arbitrarily close to $\frac{k}{2}$ as k approaches infinity. At the cost of a quasi-polynomial running time, this result applies to the more general setting of the Maximum Weight Independent Set problem in $k + 1$ -claw free graphs. Moreover, we have seen that our result is asymptotically best possible in the sense that for every $\alpha \in \mathbb{R}$, a local improvement algorithm for the weighted k -Set Packing problem that considers local improvements of w^α of logarithmically bounded size cannot produce an approximation guarantee better than $\frac{k}{2}$.

As a consequence, our paper seems to conclude the story of (pure) local improvement algorithms for both the MWIS in $k + 1$ -claw free graphs and the weighted k -Set Packing problem. Hence, the search for new techniques beating the threshold of $\frac{k}{2}$ is one of the next goals for research in this area.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Alon, N., Yuster, R., Zwick, U.: Color-coding. *J. ACM* **42**(4), 844–856 (1995). <https://doi.org/10.1145/210332.210337>
2. Arkin, E.M., Hassin, R.: On local search for weighted k -set packing. *Math. Oper. Res.* **23**(3), 640–648 (1998). <https://doi.org/10.1287/moor.23.3.640>
3. Berman, P.: A $d/2$ approximation for maximum weight independent set in d -claw free graphs. In: *Scandinavian Workshop on Algorithm Theory*, pp. 214–219. Springer (2000). https://doi.org/10.1007/3-540-44985-X_19
4. Berman, P., Fürer, M.: Approximating maximum independent set in bounded degree graphs. In: *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 365–371 (1994). <https://dl.acm.org/doi/pdf/10.5555/314464.314570>
5. Chandra, B., Halldórsson, M.M.: Greedy local improvement and weighted set packing approximation. *J. Algorithms* **39**(2), 223–240 (2001). <https://doi.org/10.1006/jagm.2000.1155>
6. Cygan, M.: Improved approximation for 3-dimensional matching via bounded pathwidth local search. In: *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26–29 October, 2013, Berkeley, CA, USA*, pp. 509–518. IEEE Computer Society (2013). <https://doi.org/10.1109/FOCS.2013.61>

7. Cygan, M., Grandoni, F., Mastrolilli, M.: How to sell hyperedges: the hypermatching assignment problem. In: Proceedings of the 2013 Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 342–351. SIAM (2013). <https://doi.org/10.1137/1.9781611973105.25>
8. Edmonds, J.: Maximum matching and a polyhedron with 0,1-vertices. J. Res. Natl. Bureau Stand. Sect. B Math. Math. Phys. **69B**, 125–130 (1965). <https://doi.org/10.6028/jres.069b.013>
9. Erdős, P., Sachs, H.: Reguläre Graphen gegebener Taillenweite mit minimaler Knotenzahl. Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math.-Natur. Reihe **12**(3), 251–257 (1963)
10. Fürer, M., Yu, H.: Approximating the k -set packing problem by local improvements. In: International Symposium on Combinatorial Optimization, pp. 408–420. Springer (2014). https://doi.org/10.1007/978-3-319-09174-7_35
11. Halldórsson, M.M.: Approximating discrete collections via local improvements. In: Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 160–169. Society for Industrial and Applied Mathematics, USA (1995). <https://dl.acm.org/doi/10.5555/313651.313687>
12. Hazan, E., Safra, S., Schwartz, O.: On the complexity of approximating k -Set Packing. Comput. Complex. **15**, 20–39 (2006). <https://doi.org/10.1007/s00037-006-0205-6>
13. Hurkens, C.A.J., Schrijver, A.: On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems. SIAM J. Discrete Math. **2**(1), 68–72 (1989). <https://doi.org/10.1137/0402008>
14. Karp, R.M.: Reducibility among combinatorial problems. In: Miller, R.E., Thatcher, J.W., Bohlinger, J.D. (eds.) Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations. Plenum Press (1972). https://doi.org/10.1007/978-1-4684-2001-2_9
15. Minty, G.J.: On maximal independent sets of vertices in claw-free graphs. J. Combin. Theory Ser. B **28**(3), 284–304 (1980). [https://doi.org/10.1016/0095-8956\(80\)90074-X](https://doi.org/10.1016/0095-8956(80)90074-X)
16. Nakamura, D., Tamura, A.: A revision of Minty’s algorithm for finding a maximum weight stable set of a claw-free graph. J. Oper. Res. Soc. Japan **44**(2), 194–204 (2001). <https://doi.org/10.15807/jorsj.44.194>
17. Neuwöhner, M.: An improved approximation algorithm for the maximum weight independent set problem in d -claw free graphs. In: 38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021), Leibniz International Proceedings in Informatics (LIPIcs), vol. 187, pp. 53:1–53:20 (2021). <https://doi.org/10.4230/LIPIcs.STACS.2021.53>
18. Neuwöhner, M.: The limits of local search for weighted k -set packing. In: Aardal, K., Sanità, L. (eds.) Integer Programming and Combinatorial Optimization—23rd International Conference, IPCO 2022, Eindhoven, The Netherlands, June 27–29, 2022, Proceedings, Lecture Notes in Computer Science, vol. 13265, pp. 415–428. Springer (2022). https://doi.org/10.1007/978-3-031-06901-7_31
19. Sbihi, N.: Algorithme de recherche d’un stable de cardinalité maximum dans un graphe sans étoile. Discrete Math. **29**(1), 53–76 (1980). [https://doi.org/10.1016/0012-365X\(90\)90287-R](https://doi.org/10.1016/0012-365X(90)90287-R)
20. Schmidt, J.P., Siegel, A.: The spatial complexity of oblivious k -probe hash functions. SIAM J. Comput. **19**(5), 775–786 (1990). <https://doi.org/10.1137/0219054>
21. Sviridenko, M., Ward, J.: Large neighborhood local search for the maximum set packing problem. In: International Colloquium on Automata, Languages, and Programming, pp. 792–803. Springer (2013). https://doi.org/10.1007/978-3-642-39206-1_67