Capacitated Partition Vertex Cover and Partition Edge Cover

Rajni Dabas
Northwestern University, Evanston, IL, USA

Samir Khuller
Northwestern University, Evanston, IL, USA

Emilie Rivkin
Northwestern University, Evanston, IL, USA

Abstract

We study generalizations of the classical Vertex Cover and Edge Cover problems that incorporate group-wise coverage.

Our first focus is the Capacitated Partition Vertex Cover (C-PVC) problem in hypergraphs. In C-PVC, we are given a hypergraph with capacities on its vertices and a partition of the hyperedge set into ω distinct groups. The objective is to select a minimum size subset of vertices that satisfies two main conditions: (1) in each group, the total number of covered hyperedges meets a specified threshold, and (2) the number of hyperedges assigned to any vertex respects its capacity constraint. A covered hyperedge is required to be assigned to a selected vertex that belongs to the hyperedge. This formulation generalizes classical Vertex Cover, Partial Vertex Cover, and Partition Vertex Cover.

We investigate two primary variants: soft capacitated (multiple copies of a vertex are allowed) and hard capacitated (each vertex can be chosen at most once). Let f denote the rank of the hypergraph (i.e., the maximum number of vertices contained in any single hyperedge). Our main contributions are: (i) an $(f+1)$-approximation algorithm for the weighted soft-capacitated C-PVC problem, which runs in polynomial time for constant ω, and (ii) an $(f+\epsilon)$-approximation algorithm for the unweighted hard-capacitated C-PVC problem, which runs in $n^{O(\omega/\epsilon)}$ time.

We also study a natural generalization of the edge cover problem, the *Weighted Partition Edge Cover* (W-PEC) problem, where each edge has an associated weight, and the vertex set is partitioned into groups. For each group, the goal is to cover at least a specified number of vertices using incident edges, while minimizing the total weight of the selected edges. We present the first exact polynomial-time algorithm for the weighted case, improving runtime from $O(\omega n^3)$ to $O(mn + n^2 \log n)$ and simplifying the algorithmic structure over prior unweighted approaches (that rely on the tropical matching problem).

2012 ACM Subject Classification Theory of computation → Packing and covering problems

Keywords and phrases Approximation algorithms, capacitated vertex cover, iterative rounding

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2026.8

Category APPROX

Funding Supported by NSF 2216970 (IDEAL) as well as a gift from Adobe Labs.

Acknowledgements We would like to thank Amol Deshpande, Hal Gabow, Tanmay Inamdar, Julian Mestre and Emily Pitler for useful discussions.

1 Introduction

The classical *Vertex Cover* (VC) problem in graphs is defined as follows: given a graph $G = (V, E)$ with weights on the vertices, find a minimum size subset of vertices S so that every edge of the graph is covered, in other words, at least one of its end points is in S. This extremely simple problem has a rich history. It was one of the earliest problems shown to be NP-complete by Karp [30] and was shown to have a polynomial time solution when the graph is bipartite. On the hardness side, [31] assumes the Unique Games Conjecture and shows that Vertex Cover is hard to approximate within a factor $(2-\epsilon)$ for every $\epsilon > 0$. Gavril [22] showed that a simple algorithm that takes all the vertices of any maximal matching is a 2-approximation (because the cardinality of any maximal matching is a lower bound on the optimal solution size). Subsequently, this simple algorithm was extended to weighted[1] graphs as well, but this required significantly deeper understanding of the role played by linear programming (LP) relaxations [24] and finally a simple combinatorial algorithm was developed based on LP duality [6]. This problem also paved the way for improvements and the well known local ratio technique [7] that found numerous applications. In addition, approximation algorithms easily follow from the work of Nemhauser and Trotter who showed that an optimal LP solution has a very simple structure [35]. In the Fixed Parameter Tractability arena, again Vertex Cover has played a crucial role in the understanding of kernelization and efficient algorithms.

Several extensions of VC have been studied. One major direction introduces capacities on vertices, leading to the *Capacitated Vertex Cover* (CVC) problem. Here, each selected vertex may cover only a limited number of incident edges. In this problem, a key distinction arises between *soft capacities* (where multiple copies of a vertex can be purchased) and *hard capacities* (where each vertex is chosen at most once). For soft capacities, Guha et al.[23] first introduced the problem and provided a 2-approximation using a primal-dual algorithm; another 2-approximation using dependent randomized rounding was later given by Gandhi et al. [20]. Notably, these results often hold for the weighted setting where vertices have costs. The hard capacity variant (VC-HC) presents significantly more challenges. Chuzhoy and Naor [12] initiated its study, giving a 3-approximation for the unweighted version on simple graphs. Crucially, they also showed that the weighted version of VC-HC is at least as hard as Set Cover, which is NP-hard to approximate within a factor of $(1-\epsilon)\ln n$ for every $\epsilon > 0$ [16]. Focusing on the unweighted VC-HC, Gandhi et al. [21] improved the approximation to a tight 2. When generalizing these problems to hypergraphs, approximation guarantees typically depend on the maximum size of a hyperedge. Let f denote the rank of the hypergraph, defined as the maximum number of vertices contained in any single hyperedge. A hypergraph whose rank is bounded by f is commonly referred to as an f-hypergraph. For f-hypergraphs, Saha and Khuller [36] provided the first $O(f)$-approximation (specifically $\min\{6f, 65\}$). Cheung, Goemans, and Wong [11] then improved the bounds to $2f$ using a deterministic two-stage LP rounding approach. The gap was finally closed to a tight f-approximation for unweighted VC-HC on f-hypergraphs, achieved independently by Kao [28] and Wong [37], using iterative rounding techniques.

A separate parallel line of work relaxes the requirement to cover all edges. *Partial Vertex Cover* asks to cover at least a specified number of edges [10, 5, 21]. A further generalization is *Partition Vertex Cover* (PVC), where the edge set is partitioned into ω groups, the solution must cover a specified minimum number of edges from each group. PVC was introduced by Bera et al. [9], who obtained an $O(\log \omega)$-approximation via a strengthened LP with exponentially many constraints. They also proved that the problem is at least as hard as Set Cover when ω is unbounded. For f-hypergraphs, Hong and Kao [25] obtained a $(f+1)\log \omega$ approximation for the uncapacitated PVC problem. This is particularly relevant as the techniques in Bera et al. for graphs do not extend naturally to hypergraphs.

[1] Every vertex has a weight and the aim is to find a minimum weighted subset of vertices.

For constant ω, and unweighted graphs, Bandyapadhyay et al. [3] gave a $(2+\epsilon)$-approximation in time $n^{O(\omega/\epsilon)}$, later improved to the weighted case with running time $n^{O(\omega)}$ independently by Liu and Li [32] and Dabas et al. [13].

Combining both dimensions - vertex capacities and partition-based coverage - yields richer and significantly more challenging models. While Partial VC with capacities has been investigated[2], these methods fail when coverage requirements are partitioned across multiple groups. In this paper, we close this gap by introducing the *Capacitated Partition Vertex Cover* (C-PVC) problem, which unifies the capacitated and partition variants of the Vertex Cover problem. We study the problem in the general setting of hypergraphs.

Beyond theoretical generalization, C-PVC provides a framework for fair resource allocation in systems where demands can be served by multiple potential providers. For example, companies like Amazon solve a massive "Capacitated Coverage" problem to assign specific fulfillment centers to zip codes for same-day delivery. In 2016, Bloomberg published an investigation showing that Amazon's Prime Free Same-Day Delivery excluded predominantly Black neighborhoods (like Roxbury in Boston and the Bronx in New York) while covering the white neighborhoods surrounding them [26]. Similar fairness concerns arise in urban mobility, where car-sharing fleets are often concentrated in high-income and advantaged zones, failing to improve accessibility in underserved communities [15]. The direct applicability of the C-PVC framework stems from the fact that these resource allocation challenges are naturally modeled as Hypergraph Vertex Cover problems rather than simple graph problems. C-PVC formulation addresses equity by enforcing a coverage threshold ρ_g for each group ensuring that every demographic/geographic sector receives a minimum standard of service.

▶ **Definition 1** (Capacitated Partition Vertex Cover (C-PVC)). *We are given a hypergraph $H=(V,E)$, where $f=\max_{e\in E}|e|$ is the rank of the hypergraph. We are also given, a vertex weight function $w: V\to\mathbb{R}_{\geq 0}$, a vertex capacity function $k: V\to\mathbb{Z}^+$, a partition of the hyperedge set $E=E_1\cup E_2\cup\cdots\cup E_\omega$, and a coverage requirement (threshold) $\rho_g\in\mathbb{Z}^+$ for each group E_g.*

We denote the capacity of a vertex v by k_v and its weight by w_v. A feasible solution consists of a set of integer copy counts $\{x_v\}_{v\in V}$ (where $x_v\in\mathbb{Z}_{\geq 0}$) and an assignment $\varphi: E'\to V$ for a subset of covered edges $E'\subseteq E$, satisfying:

1. ***Valid Assignment:*** *For every edge $e\in E'$, the assigned vertex must be in the edge, i.e., $\varphi(e)\in e$.*
2. ***Capacity Constraint:*** *The number of edges assigned to any vertex v is at most its total capacity:*

$$|\{e\in E' \mid \varphi(e)=v\}|\leq x_v\cdot k_v \quad \forall v\in V$$

3. ***Partition Constraint:*** *For each group $g\in[\omega]$, the number of covered edges from that group meets its threshold:*

$$|E'\cap E_g|\geq\rho_g$$

The objective is to find a feasible solution that minimizes $\sum_{v\in V} x_v\cdot w_v$. We refer to this as the soft-capacitated variant. The hard-capacitated variant adds the constraint that $x_v\in\{0,1\}$ for all $v\in V$.

[2] For hard capacities, Kao et al [29] give an f-approximation using iterative rounding. For soft capacities, Bar-Yehuda et al [8] give an f-approximation using the local ratio method and Mestre [34] gives a 2-approximation using a primal-dual algorithm.

In this work, we present two algorithms for the C-PVC problem. We obtain an $(f+1)$-approximation for the soft-capacitated weighted case.(Theorem 2) and an $(f+\epsilon)$-approximation for the hard-capacitated unweighted case (Theorem 3). Both algorithms run in polynomial time with respect to n and m but exhibit exponential dependence on ω, which yields an efficient polynomial-time solution when the number of groups ω is constant. This dependence on ω is consistent with previous work on partition-constrained problems. While the exact problem is known to be $W[1]$-hard parameterized by the number of groups ω (which rules out exact FPT algorithms), the existence of an FPT approximation algorithm parameterized by ω remains an intriguing open question.

For the weighted variant of the hard-capacitated case, Chuzhoy and Naor [12] showed a hardness of $\Omega(\log n)$ of any poly-time approximation. Briefly, we can give the same upper bound for Hard C-PVC. Let H be a hypergraph of rank f. For each group $g \in [\omega]$, add one vertex v_g with capacity $|E_g| - \rho_g$ (the allowed number of uncovered edges of that group) and $w_{v_g} = 0$. For each hyperedge $e \in E_g$, add v_g to e. Now, we can solve a full-coverage instance of weighted vertex cover with hard capacities on hypergraph H' with rank $f+1$ using Chuzhoy and Naor's approximation algorithm for hypergraph vertex cover. This[3] will give an approximation with factor $O(\log(n+\omega)) = O(\log(n))$ for constant ω. For this reason, we only consider the unweighted variant of Hard C-PVC.

▶ **Theorem 2.** *There exists a $(f+1)$-factor approximation algorithm for the* Soft Capacitated Weighted Partition Vertex Cover *problem that runs in polynomial time for constant ω.*

▶ **Theorem 3.** *For any fixed $\epsilon > 0$, there exists an $(f+\epsilon)$-factor approximation algorithm for the* Hard Capacitated Unweighted Partition Vertex Cover *problem that runs in time $n^{O(\omega/\epsilon)}$.*

While the vertex cover problem focuses on selecting a subset of vertices to cover all edges, a natural dual problem is the *edge cover* problem. In this variant, the goal is to select a subset of edges such that every vertex is incident to at least one selected edge. Although structurally similar to vertex cover, the edge cover problem has very different algorithmic properties—it can be solved in polynomial time via reductions to maximum matching—and it arises in applications such as network design and resource allocation.

In this paper, we study a natural generalization of the edge cover problem called the *Partition Edge Cover* (Partition-EC) problem which is defined as follows.

▶ **Definition 4** (Weighted Partition-EC). *We are given a graph $G = (V, E)$ with a weight function $p : E \to \mathbb{R}_{\geq 0}$, a partition of the vertex set $V = V_1 \cup \cdots \cup V_\omega$, and a parameter ρ_g for each group. We want to find a minimum-weight subset of edges such that at least ρ_g vertices from group V_g are covered.*

Bandyapadhyay et al. [4] studied this problem in the unweighted setting and gave an exact polynomial-time algorithm with a runtime of $O(\omega n^3)$. Their approach first reduces the problem to *Budgeted Matching*, which is then reduced to *Tropical Matching*, allowing the use of known algorithms for the latter to obtain an optimal solution.

In this work, we present a simpler algorithm with improved running time (Theorem 5). Our algorithm avoids the use of tropical matching and runs in time $O(mn + n^2 \log n)$. Moreover, our approach naturally extends to the more general *weighted* setting, where edges have arbitrary non-negative weights.

[3] This does not generalize to soft capacities, since we could pick unlimited copies of these additional zero weight vertices.

▶ **Theorem 5.** *The Weighted Partition Edge Cover problem can be solved exactly in $O(mn + n^2 \log n)$ time.*

Related Work. The closely related k-center problem has also been well studied in the partition framework. The first algorithm was proposed by Bandyapadhyay et al. [4], providing a polynomial time 2-approximation while allowing $k+\omega$ centers. A subsequent result by Anegg et al. [1] eliminated the violation in cardinality of centers and obtained a 4-approximation in time $O(n^\omega)$, which was later improved to a 3-approximation in time $O(n^{\omega^2})$ by Jia et al. [27]. The same framework for facility location and k-median objectives has also been studied; see, for example, Dabas et al. [14], Bajpai et. al. [2], and the references therein.

Organization of the paper. The remainder of the paper is organized as follows. We highlight the technical challenges and key ideas used to achieve our results in Section 1.1. Sections 2 and 3 present the $(f+1)$-approximation and $(f+\epsilon)$-approximation algorithms for soft and hard capacitated PVC, respectively. In Section 4, we describe the algorithm for W-PEC. Proofs marked by (†) are deferred to the Appendix.

1.1 Technical Challenges and Key Ideas

Covering problems are extensively studied, but partial versions of these questions (that require coverage of a given number of elements) make the problems much harder and significant new ideas and techniques were required to obtain approximation algorithms for Facility Location, k-median, k-center, Vertex Cover, etc. Despite the seemingly "small" generalization from $\omega = 1$ to $\omega > 1$, the problems again become much harder. For example, for the k-center problem with outliers, there is a simple greedy algorithm with a factor 3 and a simple LP rounding algorithm with a factor 2. As mentioned in the related work, the problem for $\omega > 1$ is considerably more involved; the best approximation factor of 3 requires a very high run time of $O(n^{\omega^2})$.

The same situation also arises for Vertex Cover, where for $\omega = 1$ there is basically no fundamental difference between full coverage and partial coverage and a matching bound of 2 can be obtained. But for $\omega > 1$ there is still no combinatorial algorithm that gives a $O(1)$ bound.

Soft C-PVC. For the soft-capacitated variant, the main challenge is managing fractional values in the LP relaxation while respecting the lower bound threshold for each group. Our algorithm first transforms an optimal fractional solution so that edge coverage is handled either by an already rounded high vertex or by a unique low vertex. We then formulate a configuration LP to handle low vertices. A configuration for a low vertex records, for each group g, how many edges of E_g are served by one copy of that vertex (therefore has one block of variables for each low vertex) and ω coverage constraints. The key insight here is that this configuration LP has at most ω fractional vertex blocks. We round one copy for each fractional block and retain its fractional configuration mixture - a max-flow computation determines the actual edge assignments. In the weighted case, we control the additive cost of the at most ω rounded blocks by guessing the ω most expensive relevant vertices. Our approach also eliminates the enumerative step for the unweighted case, thereby removing the running time's dependence on ϵ present in Bandyapadhyay et al. [3]. While this $(f+1)$ ratio is higher than the graph-specific $(2+\epsilon)$ ratio in prior work, the removal of $1/\epsilon$ from the exponent of the runtime is a significant algorithmic simplification and efficiency gain, particularly for the weighted case.

Hard C-PVC. Our algorithm uses an iterative rounding algorithm to fix vertices and assignments with large fractional values in each iteration. A final enumeration handles the remaining small solution space. This structure matches the known tight techniques for hard-capacitated hypergraph VC [37] but the core technical difficulty stems from integrating the ω partition constraints into the iterative rounding framework. The algorithm in Wong [37] crucially rely on a *counting argument* applied to the final LP's extreme point solution. This argument bounds the cost of rounding the remaining "small" fractional vertices by bounding their total number. However, introducing ω new partition constraints fundamentally alters the structure of this extreme point, as these ω constraints can also be tight. This breaks the standard counting argument. Our key technical contribution is a modified iterative algorithm that demonstrates that the number of remaining fractional variables is bounded by a function of ω. We then manage this remaining set via a final enumeration step, yielding an $(f+\epsilon)$-approximation algorithm whose runtime depends exponentially on ω.

Weighted Partition-EC. Prior work by Bandyapadhyay et al. [4] for the unweighted setting relied on a reduction to *Tropical Matching*. A major challenge is that it is not clear whether the tropical matching-based approach can be extended to handle weighted instances, and the existing method requires heavy algorithmic machinery. We circumvent this by identifying a simpler structure that allows us to avoid tropical matching entirely. By reducing the problem to a simpler matching formulation, we are able to handle arbitrary weights and significantly improve the runtime to $O(mn + n^2 \log n)$.

2 Soft-Capacitated Weighted Partition Vertex Cover

We first present an integer program for the SC-PVC problem. Note that we add an upper bound of m on the multiplicity x_v, since any vertex v has $k_v \geq 1$ and $|\delta(v)| \leq m$.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{v\in V} w_v x_v & & \text{(IP-SC)}\\
\text{subject to} \quad & \sum_{v\in e} y_{e,v} \leq 1 & & \forall e \in E\\
& \sum_{e\in E_g}\left(\sum_{v\in e} y_{e,v}\right) \geq \rho_g & & \forall g \in \{1,\dots,\omega\}\\
& \sum_{e\in\delta(v)} y_{e,v} \leq k_v x_v & & \forall v \in V\\
& 0 \leq y_{e,v} \leq x_v & & \forall v \in e \in E\\
& x_v \in \{0,1,\dots,m\} & & \forall v \in V\\
& y_{e,v} \in \{0,1\} & & \forall v \in e \in E
\end{aligned}
$$

We will say that an integral solution is *feasible* if we have an integral multiplicity x_v for each vertex and an assignment function $\varphi : E \to V \cup \{\perp\}$ such that $|\{e \in E_g : \varphi(e) \neq \perp\}| \geq \rho_g$ for all $g \in [\omega]$ and $|\{e \in \delta(v) : \varphi(e) = v\}| \leq k_v x_v$.

We only need x to be integral in a feasible solution $\langle x, y\rangle$ in order to find a fully integral solution. The following lemma is utilized in several prior works on capacitated vertex cover and involves a simple reduction to a network flow problem.

▶ **Lemma 6** (†). *Let $\langle x, y\rangle$ be a feasible solution to the linear relaxation of IP-SC such that x is integral. In polynomial time, we can compute an integral solution $\hat{y}$ such that $\langle x, \hat{y}\rangle$ is feasible.*

2.1 The Residual LP

We now partition the vertex set and generate a modified instance of the problem.

- Let $M \subseteq V$ with $|M| = \omega$. For each $v \in M$, let m_v be a fixed integer multiplicity.
- Define $R = \{v \in V \setminus M : w_v \leq \min_{v \in M} w_v\ \}$.
- Define $Z = V \setminus (M \cup R)$.

We now give a modified linear program for each choice of M and guessed multiplicty for nodes in M. Denote its value by $\mathrm{OPT}_{\mathrm{res}}(M)$.

Our guessing step separates the expensive part of the solution from the residual rounding problem. If the optimal solution uses at most ω vertices, we can enumerate to find the exact solution. We may therefore assume that the optimal solution uses more than ω distinct vertices. In this case, we guess the set M of ω highest weight vertices and their multiplicities. Under the correct guess, every remaining optimal vertex is no more expensive than the least expensive guessed vertex, so we may safely exclude all more expensive unguessed vertices. Later, the guessed vertices also provide the budget for a set of rounded fractional vertices.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{v \in R} w_v x_v & & \qquad (\text{LP-RES}_{\mathrm{M}})\\
\text{subject to} \quad & \sum_{v \in e} y_{e,v} \leq 1 & & \forall e \in E\\
& \sum_{e \in E_g} \left(\sum_{v \in e} y_{e,v} \right) \geq \rho_g & & \forall g \in \{1, \ldots, \omega\}\\
& \sum_{e \in \delta(v)} y_{e,v} \leq k_v x_v & & \forall v \in V\\
& 0 \leq y_{e,v} \leq x_v & & \forall v \in e \in E\\
& 0 \leq x_v \leq m & & \forall v \in R\\
& x_v = m_v & & \forall v \in M\\
& x_v = 0 & & \forall v \in Z
\end{aligned}
$$

For ease of notation, we will use $c_e := \sum_{v \in e} y_{e,v}$ to refer to the coverage of an edge.

▶ **Observation 7.** *Let OPT^I be the value of the optimal integral solution to IP-SC. There exists a set $M \subseteq V$ with selected multiplicites $\{m_v\}_{v \in M}$ such that the resulting* LP-RES$_{\mathrm{M}}$ *has integral solution of value $\mathrm{OPT}^I_{\mathrm{res}}(M)$ with*

$$\mathrm{OPT}^I_{\mathrm{res}}(M) + \sum_{v \in M} w_v m_v = \mathrm{OPT}^I .$$

The proof of the following lemma is somewhat technical and deferred to Subsection 2.3.

▶ **Lemma 8.** *For any feasible solution $\langle x, y \rangle$ to* LP-RES$_{\mathrm{M}}$*, we can construct a feasible solution $\langle \hat{x}, \hat{y} \rangle$ such that $\hat{x}$ is integral and has the following properties:*

1. *for vertices in M,*
$$\sum_{v \in M} w_v \hat{x}_v \leq f \sum_{v \in M} w_v x_v;$$
2. *for the vertices involved in* LP-RES$_{\mathrm{M}}$*,*
$$\sum_{v \in R} w_v \hat{x}_v \leq (f+1)\,\mathrm{OPT}_{\mathrm{res}}(M) + \sum_{v \in F} w_v,$$
where $F \subseteq R$ with $|F| \leq \omega$.

In order to complete the proof of Theorem 2, we need to show how the correct choice of $M \subseteq V$ eliminates the term $\sum_{v \in F} w_v$. Recall that for all $v \in F \subseteq R$ and all $v_j \in M$, $w_v \leq w_{v_j}$. By combining this with Observation 7, we get

$$\begin{aligned}
\sum_{v \in V} w_v \hat{x}_v &= \sum_{v \in M} w_v \hat{x}_v + \sum_{v \in R} w_v \hat{x}_v \\
&\leq f \sum_{v \in M} w_v m_v + (f+1)\,\mathrm{OPT}_{\mathrm{res}}(M) + \sum_{v \in F} w_v \\
&\leq f \sum_{v \in M} w_v m_v + (f+1) \left(\mathrm{OPT}^I - \sum_{v \in M} w_v m_v \right) + \sum_{v \in F} w_v \\
&= (f+1)\,\mathrm{OPT}^I + f \sum_{v \in M} w_v m_v - (f+1) \sum_{v \in M} w_v m_v + \sum_{v \in F} w_v \\
&= (f+1)\,\mathrm{OPT}^I - \sum_{v \in M} w_v m_v + \sum_{v \in F} w_v \\
&\leq (f+1)\,\mathrm{OPT}^I .
\end{aligned}$$

Thus our rounded solution, which is feasible with respect to both group and capacity constraints, exceeds the integral optimal solution by a factor of no more than $f+1$. There are $n^{O(\omega)}$ possible sets M, and each set has m^ω different sets of multiplicities. After checking all possibilities, we output the lowest-cost feasible solution; since this is the correct guess, its cost is bounded by Theorem 2.

Before we prove Lemma 8, we first describe a rounding scheme starting from the solution to LP-RES$_{\mathrm{M}}$.

2.2 The Rounding Scheme

We give a partition of V:

- $H := M \cup \left\{ v \in R : x_v \geq \frac{1}{f+1} \right\}$
- $L := \left\{ v \in R : 0 < x_v < \frac{1}{f+1} \right\}$

Any vertices with $x_v = 0$ are disregarded, and we set $\hat{x}_v = 0$. Now, we round the edge variables. If any edge is such that all vertices have $\hat{x}_v = 0$, that edge cannot be covered. For simplicity, we will remove such edges from consideration going forward (these will remain uncovered). For each $v \in L$, create an empty set L_v. Some edges may include vertices that are already rounded to 0. For clarity, let $L' = L \cup \{v \in V : \hat{x}_v = 0\}$.

The next step preserves the total coverage c_e of every edge. It only moves this coverage between endpoints such that the coverage of each edge is partitioned between vertices in H or given to one unique vertex in L.

1. $e \cap L' = \emptyset$: all coverage is handled by vertices in H. Set $\hat{y}_{e,v} = y_{e,v}$ for all $v \in e$.
2. $e \cap H = \emptyset$: all coverage is handled by vertices in L. Let $v_\ell = \arg\max_{v \in e \cap L} y_{e,v}$. Set $\hat{y}_{e,v_\ell} = c_e$ and $\hat{y}_{e,v} = 0 \; \forall \; v \in e \setminus v_\ell$. Add e to L_{v_ℓ}.
3. $e \cap H, e \cap L \neq \emptyset$: coverage is distributed between vertices in L and vertices in H.
   a. If $y_{e,v_\ell} \geq \frac{c_e}{f+1}$, let $v_\ell = \arg\max_{v \in e \cap L} y_{e,v}$. Set $\hat{y}_{e,v_\ell} = c_e$ and $\hat{y}_{e,v} = 0$ for all other $v \in e$. Add e to L_{v_ℓ}.
   b. Otherwise, let $c_e^\ell = \sum_{v \in e \cap L} y_{e,v}$ and $c_e^h = \sum_{v \in e \cap H} y_{e,v}$. Then, $\hat{y}_{e,v} = y_{e,v} + \frac{y_{e,v}}{c_e^h} c_e^\ell$ for $v \in e \cap H$ and $\hat{y}_{e,v} = 0$ for $v \in e \cap L$.

Note that, in case (3), if an endpoint in L is above the threshold $\frac{c_e}{f+1}$, it can absorb the entire edge with an $(f+1)$-fold increase; otherwise, the coverage provided by all endpoints in L is proportionately transfered to the endpoints in H. Therefore, the threshold is carefully chosen to work for both cases. Also observe that the sets L_v are disjoint, meaning that each edge has no more than one $v \in L$ that is responsible for its coverage. For now, we only round $v \in H$: $\hat{x}_v = \left\lceil \frac{f+1}{2} x_v \right\rceil$.

▶ **Lemma 9** (†). *For all $v \in H$,*

$$\sum_{e \in \delta(v)} \hat{y}_{e,v} \leq \frac{f+1}{2} k_v x_v \leq k_v \hat{x}_v.$$

Moreover, if $v \in H \cap R$, $\hat{x}_v \leq (f+1)x_v$, and if $v \in M$, $\hat{x}_v \leq f \cdot x_v$.

We are now covering some subset of edges in each group. For $g \in [\omega]$,

$$h_g := \sum_{v \in H} \sum_{e \in \delta(v) \cap E_g} \hat{y}_{e,v}.$$

In Lemma 9, we showed that the rounded multiplicities $\tilde{x}_v$ for $v \in H$ can cover all rounded fractional edges that they are responsible for. Now, we have to open some vertices in L to cover the remaining $\rho_g - h_g$ edges of each group.

2.3 The Configuration LP

Our final task is to round the vertices in L. For these vertices, we will only consider whether or not we will open one copy of each vertex. For $v \in L$ and $g \in [\omega]$, let $N_{v,g} = E_g \cap L_v$ be the set of edges of group g that v is responsible for, and let $\eta_{v,g} = |N_{v,g}|$.

Since each L_v is disjoint, some vertex v covering some edge $e \in L_v$ has no bearing on whether another vertex v' can cover some edge $e' \in L_{v'}$, even if v and v' are adjacent. Therefore, we consider the following question: if we were to open one copy of v, how many different ways could v cover up to k_v edges in L_v? With this framing, we simplify the remaining problem by rendering edges of the same group in each L_v indistinguishable.

A feasible configuration for a vertex $v \in L$ is a vector $A = (A_1, \ldots, A_\omega) \in \mathbb{Z}^{\omega}_{\geq 0}$ with $A_g \leq \eta_{v,g}$ for all $g \in [\omega]$ and $\sum_{g=1}^{\omega} A_g \leq k_v$. The coordinate A_g records how many owned edges of group g are served by one copy. The bounds $A_g \leq \eta_{v,g}$ and $\sum_g A_g \leq k_v$ enforce edge availability and capacity, respectively. Let $\mathcal{A}_v$ be the set of all feasible configurations for $v \in L$. We now solve this configuration LP:

$$\begin{aligned}
\text{minimize} \quad & \sum_{v \in L} w_v \sum_{A \in \mathcal{A}_v \setminus \mathbf{0}} \lambda_{v,A} && \text{(LP-CONFIG)} \\
\text{subject to} \quad & \sum_{A \in \mathcal{A}_v} \lambda_{v,A} = 1 \qquad \forall v \in L && (1) \\
& \sum_{v \in L} \sum_{A \in \mathcal{A}_v} A_g \lambda_{v,A} \geq \rho_g - h_g \qquad \forall g \in \{1, \ldots, \omega\} && (2) \\
& \lambda_{v,A} \geq 0 \qquad \forall v \in L, A \in \mathcal{A}_v &&
\end{aligned}$$

We consider the zero configuration $\mathbf{0}$ to mean that the vertex is not chosen, so each $\lambda_{v,\mathbf{0}}$ does not contribute to the cost.

We first need to connect the solution of LP-CONFIG to the solution of LP-RES$_\mathrm{M}$ restricted to the vertices in L. In order to do this, we will construct a feasible solution to LP-CONFIG using the solution $\langle x, y \rangle$ restricted to vertices in L and edges in $\bigcup_{v \in L} L_v$ and the rounding scheme described in Subsection 2.2.

▶ **Lemma 10.** *LP-CONFIG has a feasible solution of cost no more than* $(f+1)\sum_{v\in L} w_v x_v$.

Proof. We start by considering the edges in the sets L_v for each $v \in L$. If an edge was rounded using case (2), we have

$$\hat{y}_{e,v} = c_e \leq f \cdot y_{e,v} \leq (f+1)x_v.$$

If it was rounded using case (3a), v must be such that $y_{e,v} \geq \frac{c_e}{f+1}$, so

$$\hat{y}_{e,v} = c_e \leq (f+1)y_{e,v} \leq (f+1)x_v.$$

We also observe that, because L contains only vertices with $x_v < \frac{1}{f+1}$, we must have $(f+1)x_v < 1$. Finally, by summing over edges in L_v, we have

$$\sum_{e\in L_v} c_e \leq (f+1)\sum_{e\in L_v} y_{e,v} \leq (f+1)k_v x_v$$

from the feasibility of the LP solution. This implies that

$$\sum_{e\in L_v} \frac{c_e}{(f+1)x_v} \leq k_v.$$

Consider the following set of vectors:

$$\mathcal{P}(d,k) = \left\{ z \in \mathbb{R}^d : 0 \leq \|z\|_\infty \leq 1, \|z\|_1 \leq k \right\}.$$

This defines a concave set with integral extremal points. Assume that we ordered the edges of L_v. Then, the vector

$$z_v = \left(\frac{c_{e_1}}{(f+1)x_v}, \frac{c_{e_2}}{(f+1)x_v}, \dots \right)$$

is contained within $\mathcal{P}(|L_v|, k_v)$. Let $\mathcal{S}_v = \{S \subseteq L_v : |S| \leq k_v\}$. We can now use the set of indicators $\mathbf{1}_S$ of $S \in \mathcal{S}_v$ as a basis for z_v. Write $z_v = \sum_{S\in\mathcal{S}_v} \zeta_{v,S}\mathbf{1}_S$ for $\sum_{S\in\mathcal{S}_v} \zeta_{v,S} = 1$.

Next, we want to move from the vectors $\mathbf{1}_S$ to feasible edge configurations. Define the projection $\pi_v : \mathbb{R}^{|L_v|} \to \mathbb{R}^\omega$ such that column g is $\mathbf{1}_{N_{v,g}}$. Each $\pi_v \mathbf{1}_S$ is a feasible edge configuration. For each $A \in \mathcal{A}_v \setminus \mathbf{0}$, let $S_A = \{S \in \mathcal{S}_v : \pi_v S = A\}$. Define $\lambda_{v,A} = (f+1)x_v \sum_{S\in S_A} \zeta_{v,S}$ and $\lambda_{v,\mathbf{0}} = 1 - \sum_{A\neq\mathbf{0}} \lambda_{v,A}$. Therefore, λ satisfies the constraints (1).

Now, we look at the constraints (2). For one edge, $\sum_{S\ni e} \zeta_{v,S} = \frac{c_e}{(f+1)x_v}$. We know that the edges are sufficiently covered in the rounded solution, so

$$\begin{aligned}
\sum_{v\in L}\sum_{A\in\mathcal{A}_v} A_g \lambda_{v,A} &= \sum_{v\in L}\sum_{A\in\mathcal{A}_v} A_g \sum_{S\in S_A} \zeta_{v,S}(f+1)x_v \\
&= \sum_{v\in L}(f+1)x_v \sum_{S\in\mathcal{S}} \mathbf{1}_{N_{v,g}} \cdot \mathbf{1}_S \zeta_{v,S} \\
&= \sum_{v\in L}\sum_{S\in\mathcal{S}_v}\sum_{e\in S\cap N_{v,g}} (f+1)x_v\zeta_{v,S} \\
&= \sum_{v\in L}(f+1)x_v \sum_{e\in N_{v,g}}\sum_{S\ni e} \zeta_{v,S} \\
&= \sum_{v\in L}(f+1)x_v \sum_{e\in N_{v,g}} \frac{c_e}{(f+1)x_v} \\
&= \sum_{v\in L}\sum_{e\in N_{v,g}} c_e \geq \rho_g - h_g
\end{aligned}$$

Now that the projection defines a feasible solution to LP-CONFIG, we need to show that the objective function value is bounded:

$$\begin{aligned}\sum_{v\in L} w_v \sum_{A\in\mathcal{A}_v\setminus\mathbf{0}} \lambda_{v,A} &= \sum_{v\in L} w_v \sum_{A\in\mathcal{A}_v\setminus\mathbf{0}} (f+1)x_v \sum_{S\in S_A} \zeta_{v,S} \\ &= \sum_{v\in L} w_v (f+1)x_v \sum_{S\in\mathcal{S}\setminus\mathbf{0}} \zeta_{v,S} \\ &\le (f+1)\sum_{v\in L} w_v x_v. \end{aligned}$$ ◀

Let $F \subseteq L$ be the set of vertices with fractional $\lambda_{v,A}$. While we also want to open this set of vertices, bounding the cost is made difficult by the lack of an integral configuration.

▶ **Lemma 11.** *Any optimal basic feasible solution λ to LP-CONFIG has $|F| \le \omega$.*

Proof. We start by turning constraints (2) into equalities. Our new LP is

$$\begin{aligned} \text{minimize} \quad & \sum_{v\in L} w_v \sum_{A\in\mathcal{A}_v\setminus\{0\}} \lambda_{v,A} \\ \text{subject to} \quad & \sum_{A\in\mathcal{A}_v} \lambda_{v,A} = 1 && \forall v\in L \\ & \sum_{v\in L}\sum_{A\in\mathcal{A}} A_g \lambda_{v,A} = \rho_g - h_g + s_g && \forall g\in\{1,\dots,\omega\} \\ & \lambda_{v,A} \ge 0 && \forall v\in L, A\in\mathcal{A}_v \\ & s_g \ge 0 && \forall g\in\{1,\dots,\omega\} \end{aligned}$$

A solution to this LP has up to $|L|+\omega$ non-tight variables, meaning that up to $|L|+\omega$ of the variables $\lambda_{v,A}$ and s_g are positive. Constraints (1) ensures that at least one $\lambda_{v,A}$ per $v\in L$ is positive, and at least $|L|-\omega$ of those constraints has only one positive $\lambda_{v,A}$, so up to ω of these constraints can have fractional $\lambda_{v,A}$. ◀

We now have the machinery to prove Lemma 8. Recall the statement:

▶ **Lemma 8.** *For any feasible solution $\langle x, y\rangle$ to* LP-RES$_\mathrm{M}$*, we can construct a feasible solution $\langle \hat{x}, \hat{y}\rangle$ such that $\hat{x}$ is integral and has the following properties:*

1. *for vertices in M,*

$$\sum_{v\in M} w_v \hat{x}_v \le f \sum_{v\in M} w_v x_v;$$

2. *for the vertices involved in* LP-RES$_\mathrm{M}$*,*

$$\sum_{v\in R} w_v \hat{x}_v \le (f+1)\,\mathrm{OPT}_{\mathrm{res}}(M) + \sum_{v\in F} w_v,$$

where $F \subseteq R$ with $|F| \le \omega$.

Proof. For each $v\in L$, if $\lambda_{v,\mathbf{0}} = 1$, set $\hat{x}_v = 0$ and $\hat{y}_{e,v} = 0$ for all $e \in L_v$. Otherwise, set $\hat{x}_v = 1$. If $v \notin F$, there is one integral edge configuration. Select a set $N(A)$ consisting of A_g edges from each $N_{v,g}$, and set $\hat{y}_{e,v} = \mathbf{1}_{e\in N(A)}$. If $v\in F$, let $\mathcal{F}_v = \{A \in \mathcal{A}_v : \lambda_{v,A}\} > 0$. For each configuration $A\in\mathcal{F}_v$, define a set of edges $N(A)$. Then, set $\hat{y}_{e,v} = \sum_{A\in\mathcal{F}_v} \lambda_{v,A}\mathbf{1}_{e\in N(A)}$.

Since the solution λ is feasible for LP-CONFIG,

$$\sum_{v\in L}\sum_{e\in N_{v,g}} \hat{y}_{e,v} = \sum_{v\in L\setminus F}\sum_{e\in N_{v,g}} \mathbf{1}_{e\in N(A)} + \sum_{v\in F}\sum_{A\in\mathcal{F}_v}\sum_{e\in N_{v,g}} \lambda_{v,A}\mathbf{1}_{e\in N(A)}$$
$$= \sum_{v\in L} A_g + \sum_{v\in F}\sum_{A\in\mathcal{F}_v}\sum_{e\in N_{v,g}} \lambda_{v,A}A_g \geq \rho_g - h_g,$$

since the L_v are disjoint and no edge can be double-counted. Moreover, if $v \in L \setminus F$,

$$\sum_{e\in\delta(v)} \hat{y}_{e,v} = \sum_{g=1}^{\omega} A_g \leq k_v = k_v\hat{x}_v.$$

If $v \in F$,

$$\sum_{e\in\delta(v)} \hat{y}_{e,v} = \sum_{e\in\delta(v)}\sum_{A\in\mathcal{F}_v} \lambda_{v,A}\mathbf{1}_{e\in N(A)} = \sum_{e\in\delta(v)}\sum_{A\in\mathcal{F}_v} \lambda_{v,A}A_g \leq k_v = k_v\hat{x}_v.$$

From Lemma 9, the vertices in H respect the capacity constraints. Thus, we can conclude that $\langle\tilde{x},\tilde{y}\rangle$ over all $v \in L \cup H$ satisfies both the group and capacity constraints. We now bound the cost of the rounded vertices. We use Lemma 9. For part (1), we can sum over the bound on $\hat{x}_v$ for all vertices in M. For part (2),

$$\sum_{v\in R\cap H} w_v\hat{x}_v \leq (f+1)\sum_{v\in R\cap H} w_v x_v.$$

From Lemma 10,

$$\sum_{v\in L} w_v\hat{x}_v \leq (f+1)\sum_{v\in L} w_v x_v + \sum_{v\in F} w_v.$$

By combining these statements, we have

$$\sum_{v\in R} w_v\hat{x}_v \leq (f+1)\sum_{v\in R} w_v x_v + \sum_{v\in F} w_v = (f+1)\,\mathrm{OPT}_{\mathrm{res}}(M) + \sum_{v\in F} w_v,$$

which concludes the proof of Lemma 8. ◀

2.4 The Full Algorithm

We end this section with an overview of the algorithm and a discussion of the runtime.

1. Enumerate through every set of vertices of size at most ω and every integral multiplicity from 1 to m. If any set is feasible, take the cheapest feasible set. Otherwise, continue to the main algorithm. There are $(|V| + |E|)^{O(\omega)}$ such sets.
2. For every set $M \subseteq V$ with $|M| \leq \omega$ and every integral multiplicity from 1 to m, solve LP-RES$_\mathrm{M}$. There are $(|V| + |E|)^{O(\omega)}$ such sets.
   a. Round the solution using Lemma 9.
   b. Solve LP-CONFIG for the vertices in L. The LP has up to $|V|(|E| + 1)^\omega$ variables and $|V| + \omega$ constraints.
   c. Round the fractional vertices using Lemmas 10 and 11.
   d. Continue with the lowest-cost feasible solution, which corresponds to the correct guess of M.
3. Find an integral edge assignment using Lemma 6. The network has $O(|V| + |E| + \omega)$ nodes and $O(|V| + |E| + \omega)$ arcs.

Each step requires time that is polynomial in the $|V| + |E|$, and we repeat each step a number of times that is polynomial in $|V| + |E|$. Since ω is assumed to be constant, the overall algorithm is polynomial in $|V| + |E|$. Note that this runtime is independent of the hyperedge size f. This completes the proof of Theorem 2.

3 Hard Capacitated Unweighted Partition Vertex Cover

In this section, we study the Hard Capacitated Unweighted Partition Vertex Cover problem in hypergraphs using iterative rounding. Before detailing the iterative steps, we define the LP relaxation using fractional vertex variables x and assignment variables y. These variables are constrained to satisfy the partition thresholds and vertex capacities while ensuring that no edge assignment exceeds the fractional status of its endpoints. This formulation minimizes the total fractional vertex count, establishing a baseline solution that informs the subsequent rounding iterations. The integer program closely resembles IP-SC, so we will not restate it. The formal relaxation is presented below:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{v \in V} x_v \\
\text{subject to:} \quad & \\
& \sum_{v \in e} y_{e,v} \leq 1 && \forall e \in E \\
& \sum_{e \in E_g} \left(\sum_{v \in e} y_{e,v} \right) \geq \rho_g && \forall g \in \{1, \dots, \omega\} \\
& y_{e,v} \leq x_v && \forall v \in e \in E \\
& \sum_{e \in \delta(v)} y_{e,v} \leq k_v x_v && \forall v \in V \\
& 0 \leq x_v \leq 1 && \forall v \in V \\
& y_{e,v} \geq 0 && \forall v \in e \in E
\end{aligned}
$$

It is sufficient to find an integral solution for x; an integral assignment y can then be found efficiently via a max-flow computation. Our algorithm therefore focuses on finding an integral x with a bounded cost.

3.1 Algorithm

Our algorithm is based on an iterative rounding framework. The core idea is to repeatedly solve an LP relaxation, fix a subset of variables to integral values based on the solution, and then re-solve a smaller, updated LP on the remaining problem. This process is guided by a dynamic LP formulation that adapts to the decisions made in each step (see LP-IP).

We maintain several sets for the iterative rounding algorithm. Let D be the set of vertices whose final integral counts have been determined, and let T_v^g denote the set of edges of group g that have been definitively assigned to vertex v. We define $T = \bigcup_{v \in V} \bigcup_{g=1}^{\omega} T_v^g$ as the set of all covered edges. Based on the fractional solution from the previous step, we partition the remaining undecided vertices into two groups: $U = \{v \in V \setminus D \mid x_v^* \geq 1/f\}$, which contains vertices with high fractional values, and $W = \{v \in V \setminus D \mid 0 < x_v^* < 1/f\}$, which contains those with low fractional values. U is further divided into $U_> = \{v \in V \setminus D \mid x_v^* > 1/f\}$ and $U_= = \{v \in V \setminus D \mid x_v^* = 1/f\}$. Let $Z = \{v \in V \setminus D \mid x_v^* = 0\}$. Initially the sets T and D are empty.

$$\text{minimize} \quad \sum_{v \in V \setminus D} x_v \tag{LP-IP}$$

subject to:

$$\begin{aligned}
&\sum_{v \in e \setminus D} y_{e,v} \leq 1 - \sum_{v' \in e \cap D} \bar{y}_{e,v'} && \forall e \in E \setminus T \\
&\sum_{e \in E_g \setminus T} \sum_{v \in e \setminus D} y_{e,v} \geq \rho_g \\
&\qquad - \sum_{v \in V} |T_v^g| - \sum_{e \in E_g \setminus T} \left(\sum_{v' \in e \cap D} \bar{y}_{e,v'} \right) && g \in \{1, \ldots, \omega\} \\
&y_{e,v} \leq x_v && \forall e \in E \setminus T, v \in e \setminus D \\
&\sum_{e \in \delta(v) \setminus T} y_{e,v} \leq \left(k_v - \sum_{g \in \omega} |T_v^g| \right) x_v && \forall v \in U_> \\
&\sum_{e \in \delta(v) \setminus T} y_{e,v} \leq k_v x_v && \forall v \in W \\
&1/f \leq x_v \leq 1 && \forall v \in U_> \\
&0 \leq x_v \leq 1/f && \forall v \in W \\
&y_{e,v} \geq 0 && \forall e \in E \setminus T, v \in e \setminus D
\end{aligned}$$

Note that there is no capacity constraint included for vertices in $U_=$ within LP-IP. This is because any vertex u entering $U_=$ is immediately processed by the algorithm (its value is fixed to $\bar{x}_u = 1$ and it is moved to the decided set D). Thus, it is no longer an active variable in the updated LP over $V \setminus D$, and omitting it does not cause the feasible region to grow. The complete iterative rounding framework is detailed in Algorithm 1. It formalizes the process of identifying which variables to fix, updating the problem state, and iterating until a fully integral solution is found.

Note that when a tight edge is found and permanently assigned (Lines 8-11), we intentionally do not fix x_u to 1. The variable x_u remains active and fractional in the updated LP so its residual capacity can be utilized for other incident edges, and to prevent a premature increase in the LP objective. Since $u \in U$, Lemma 6 guarantees its fractional value remains at least $1/f$, ensuring it will be safely rounded to 1 at the end of the algorithm.

3.2 Analysis

▶ **Lemma 12.** *Suppose a vertex u satisfies $x_u^* \geq 1/f$ at some iteration. Then $x_u^* \geq 1/f$ in all subsequent iterations as long as $u \notin D$. Consequently, the final integral value $\bar{x}_u$ will be at least 1.*

Proof. When $x_u^* \geq 1/f$, the vertex u is placed in the set U. We consider two cases based on its value.

- If $x_u^* = 1/f$, then $u \in U_=$. In the same iteration, the algorithm adds u to the set of decided vertices D and fixes its final integral value as $\bar{x}_u = 1$.
- If $x_u^* > 1/f$, then $u \in U_>$. For all subsequent iterations where u remains an undecided variable, the updated LP-IR includes the explicit constraint $x_u \geq 1/f$. This ensures its fractional value cannot drop below $1/f$.

Algorithm 1 Iterative Rounding Algorithm

Solve the initial LP Relaxation to get an extreme point solution $\langle x^*, y^* \rangle$.
$T_v^g \leftarrow \emptyset$ for all $v \in V$ (which implicitly sets $T = \emptyset$), $g \in \{1, \ldots, \omega\}$ and $D \leftarrow \emptyset$.
Initialize $U, U_>, U_=, W, Z$ based on $\langle x^*, y^* \rangle$.
repeat
  for $v \in Z$ **do** ▷ Handle zero-valued vertices
    Set $\bar{x}_v \leftarrow 0$.
    $D \leftarrow D \cup \{v\}$.
  end for
  if $y^*_{e,u} = x^*_u$ for some $u \in U$ and $e \in \delta(u) \setminus T$ **then** ▷ Assign tight edges
    Set $\bar{y}_{e,u} \leftarrow 1$, and $\bar{y}_{e,v} \leftarrow 0$ for $v \in e \setminus \{u\}$.
    $T_u^g \leftarrow T_u^g \cup \{e\}$ **for** $e \in E_g$.
  end if
  for $u \in U_=$ **do** ▷ Handle boundary-valued vertices
    Set $\bar{x}_u \leftarrow 1$.
    Set $\bar{y}_{e,u} \leftarrow y^*_{e,u}$ **for** $e \in \delta(u) \setminus T$. ▷ Fix edge assignments
    $D \leftarrow D \cup \{u\}$.
  end for
  Solve the updated LP-IR for a new solution $\langle x^*, y^* \rangle$.
  Update $U, U_>, U_=, W, Z$ based on the new $\langle x^*, y^* \rangle$.
until $U_= = \emptyset$ and $Z = \emptyset$ and no tight edges are found
for $v \notin D$ **do** ▷ Final rounding step
  Set $\bar{x}_v \leftarrow \lceil x^*_v \rceil$.
end for
return integral solution $\bar{x}$.

Eventually, either u enters $U_=$ (and is rounded to $\bar{x}_u = 1$) or the algorithm terminates with $u \notin D$. In the latter case, it is rounded up in the final step, yielding $\bar{x}_u = \lceil x^*_u \rceil \geq \lceil 1/f \rceil = 1$. In all possible outcomes, if u ever enters U, its final value $\bar{x}_u$ is at least 1. ◀

▶ **Lemma 13.** *The solution $\bar{x}$ returned by the algorithm is a feasible integral solution to the problem.*

Proof. The final solution $\bar{x}$ is integral by construction. Variables are set to integer values (0 or 1) during the loop, and the final step applies the ceiling function.

Next, the initial LP ensures that the group coverage ρ_g is satisfied fractionally. As the algorithm fixes variables and assigns edges to T, it reduces the coverage requirement for subsequent LPs. The final LP solution $\langle x^*, y^* \rangle$ satisfies the remaining coverage fractionally. The final rounding step, $\bar{x}_v = \lceil x^*_v \rceil$, only increases the available capacity, which means at least as many edges can be covered as in the fractional solution. Thus, the integral solution meets or exceeds the fractional coverage and satisfies all group coverage requirements.

Finally, we consider capacity constraints for each group of vertices:

- **For vertices in $U_>$:** The capacity used by any "tight" edges is explicitly subtracted and budgeted for in later steps. When the remaining vertices are rounded up, the new integer capacity is always greater than or equal to the fractional capacity required by the LP solution ($k_u \lceil x^*_u \rceil \geq k_u x^*_u$), ensuring the constraint holds.
- **For vertices in $U_=$:** When a vertex is rounded from $1/f$ to 1, the LP guarantees that its fractional edge assignments required a capacity of at most $(k_u - \sum_{g \in [\omega]} |T_u^g|)/f$. This is well within the newly allocated integer capacity of $(k_u - \sum_{g \in [\omega]} |T_u^g|)$, so the assignment is valid.
- **For vertices in W:** At the end of the algorithm, these vertices are rounded up from a value less than $1/f$ to 1. The LP solution's required capacity was less than the final integer capacity ($k_w x^*_w < k_w \cdot 1$), so the final assignment is feasible. ◀

▶ **Lemma 14.** *Let $\langle x^*, y^* \rangle$ be the extreme point solution upon which the algorithm terminates. $|W| \le |U_1| + \omega$ where $U_1 = \{u \in U : x^*_u = 1\}$.*

Proof. The proof is by a counting argument on the variables and constraints of the final LP-IR. As an extreme point solution, it is uniquely defined by a system of linear equations formed by setting a subset of the LP's inequality constraints to be tight. A fundamental property of such solutions is that the number of variables must equal the number of linearly independent tight constraints.

At termination, the set of undecided vertices is $V \setminus D = U_> \cup W$. The algorithm's termination conditions ensure that for any $u \in U_>$, we have $x^*_u > 1/f$ and $y^*_{e,u} < x^*_u$. For any $w \in W$, we have $0 < x^*_w < 1/f$.

The variables in the final LP are the $\{x_v\}$ for undecided vertices and the associated $\{y_{e,v}\}$ assignment variables. The total number of variables is

$$|U_>| + |W| + \sum_{e \in E \setminus T} |e \setminus D| = |U_>| + |W| + \sum_{e \in E \setminus T} |e \cap U_>| + \sum_{e \in E \setminus T} |e \cap W|.$$

Next, we establish an upper bound on the number of linearly independent constraints that can be tight. For the **partition constraints**, **capacity constraints for $\boldsymbol{U_>}$:**, and **upper bounds $\boldsymbol{x_u \le 1}$:** at most ω, $|U_>|$, and $|U_1|$ constraints can be tight.

The **constraints involving W vertices** include the capacity constraint for each $w \in W$ ($\sum y_{e,w} \le k_w x_w$), the coupling constraints ($y_{e,w} \le x_w$), and the non-negativity constraints ($y_{e,w} \ge 0$). The maximum number of constraints from this group is limited for two reasons. First, for any specific edge e, the constraints $y_{e,w} \ge 0$ and $y_{e,w} \le x_w$ cannot both be tight simultaneously. This is because $x^*_w > 0$. Thus, for each edge, at most one of these two boundary constraints can be part of the basis. Second, the entire collection of constraints for vertex w is linearly dependent. For a given vertex w, if the constraints corresponding to $y_{e,w}$ are chosen to be all tight, the tightness of the capacity constraint is a direct consequence and is therefore redundant; it cannot add a new linearly independent equation to the basis. Due to these two properties, the maximum number of independent tight constraints associated with any single vertex w is $|\delta(w) \setminus T|$. Summing over all vertices in W establishes the total contribution of at most

$$\sum_{w \in W} |\delta(w) \setminus T| = \sum_{e \notin T} |e \cap W|$$

constraints to the basis.

Finally, in order to handle the **constraints involving $\boldsymbol{U_>}$ vertices**, we do a per-edge analysis. For each edge $e \in E \setminus T$, we analyze the relationship between its edge coverage constraint and its non-negativity constraints for $U_>$ endpoints. Let C_e be the edge coverage constraint

$$\sum_{v \in e \setminus D} y_{e,v} \le 1 - \sum_{v' \in e \cap D} \bar{y}_{e,v'},$$

and let $\{N_{e,u}\}$ be the set of non-negativity constraints: $y_{e,u} \ge 0$ for all $u \in e \cap U_>$.

▷ Claim. For any edge e with at least one endpoint in $U_>$, it is impossible for C_e and all constraints in $\{N_{e,u}\}$ to be tight simultaneously.

Proof. Assume for contradiction that for an edge e where $|e \cap U_>| \ge 1$, constraint C_e and all constraints $N_{e,u}$ are tight. The tightness of all $N_{e,u}$ implies $y_{e,u} = 0$ for all $u \in e \cap U_>$.

Thus, the edge's total coverage comes only from W vertices:

$$\sum_{v \in e \setminus D} y_{e,v} = \sum_{w \in e \cap W} y_{e,w}.$$

Since $y_{e,w} < 1/f$ for any $w \in W$, we have a strict upper bound:

$$\sum_{v \in e \setminus D} y_{e,v} < |e \cap W|/f.$$

The tightness of C_e means

$$\sum_{v \in e \setminus D} y_{e,v} = 1 - \sum_{v' \in e \cap D} \bar{y}_{e,v'}.$$

This right-hand side has a lower bound of $|e \setminus D|/f$. Combining these gives

$$|e \setminus D|/f \leq \sum_{v \in e \setminus D} y_{e,v} < |e \cap W|/f.$$

This implies

$$|e \cap U_>| + |e \cap W| < |e \cap W|,$$

which simplifies to $|e \cap U_>| < 0$, a contradiction. ◁

This claim implies that for each edge e, the set of $1 + |e \cap U_>|$ constraints (consisting of C_e and $\{N_{e,u}\}$) are linearly dependent when all are tight. Therefore, at most $|e \cap U_>|$ of them can be chosen for a linearly independent basis. The total number of tight constraints from this group, summed over all edges, is at most $\sum_{e \in E \setminus T} |e \cap U_>|$.

By equating the number of variables to the upper bound on the number of tight, linearly independent constraints, we have:

$$|U_>|+|W|+\sum_{e \in E \setminus T} |e \cap U_>|+\sum_{e \in E \setminus T} |e \cap W| \leq \omega+|U_>|+|U_1|+\sum_{e \in E \setminus T} |e \cap W|+\sum_{e \in E \setminus T} |e \cap U_>|.$$

Canceling the common terms from both sides yields the desired result. ◀

We can now return to the proof of Theorem 3.

▶ **Theorem 3.** *For any fixed $\epsilon > 0$, there exists an $(f + \epsilon)$-factor approximation algorithm for the* Hard Capacitated Unweighted Partition Vertex Cover *problem that runs in time $n^{O(\omega/\epsilon)}$.*

Proof. The feasibility of the integral solution $\bar{x}$ produced by the algorithm is guaranteed by Lemma 13. We now bound its cost.

Let OPT_{LP} be the optimal value of the initial LP relaxation. First, observe that the optimal value of the LP relaxation does not decrease between iterations. When rounding decisions are made in the main loop, the problem is modified by either fixing variables or enforcing stricter inequalities (e.g., $x_v \geq 1/f$ instead of $x_v \geq 0$). Because we shrink the feasible region of a minimization problem, the objective value cannot improve. The cost added to our integral solution during the main loop can be bounded by charging it to the fractional value removed from the LP. Specifically, when $x^*_u \geq 1/f$ is rounded to 1, the cost ratio is at most $1/(1/f) = f$.

Let S_L be the set of vertices finalized during the iterative loop and $S_F = U_> \cup W$ be the set of vertices rounded at the end. An accounting argument based on the above observation shows that the cost from vertices rounded during the loop, $|S_L|$, plus f times the cost of the final LP, $f \cdot \mathrm{OPT}_F$, is at most $f \cdot \mathrm{OPT}_{LP}$.

The cost from the final rounding step is $\mathrm{Cost}(S_F) = \sum_{v \in U_> \cup W} \lceil x_v^* \rceil = |U_>| + |W|$. By Lemma 14, this is bounded by $|U_>| + |U_1| + \omega$. For any $u \in U_>$, we have $1 < f \cdot x_u^*$. This is sufficient to show that, for $f \geq 2$,

$$|U_>| + |U_1| \leq f \cdot \sum_{u \in U_>} x_u^* \leq f \cdot \mathrm{OPT}_F .$$

Therefore, the cost of the final step is bounded as

$$\mathrm{Cost}(S_F) = |U_>| + |W| \leq f \cdot \mathrm{OPT}_F + \omega$$

Combining the costs from both phases gives the total cost

$$\mathrm{Cost}(\bar{x}) = |S_L| + \mathrm{Cost}(S_F) \leq |S_L| + f \cdot \mathrm{OPT}_F + \omega$$

Since $|S_L| + f \cdot \mathrm{OPT}_F \leq f \cdot \mathrm{OPT}_{LP}$, we have a solution with a total cost bounded by $f \cdot \mathrm{OPT}_{LP} + \omega$.

We enumerate all possible solutions of size $\kappa = 1, 2, \ldots, \lceil \omega/\epsilon \rceil$. This step takes $n^{O(\omega/\epsilon)}$ time. If a feasible solution is found, the first one encountered must be an optimal solution, since we are checking solution sizes in increasing order, and we are done.

Otherwise, if this process is completed without finding a feasible solution, we know that the optimal integral solution, OPT, has a cost greater than ω/ϵ. In this case, we run our iterative rounding algorithm. By the analysis above, it returns a feasible solution with a cost bounded by

$$\mathrm{Cost}(\bar{x}) \leq f \cdot \mathrm{OPT}_{LP} + \omega$$

Since $\mathrm{OPT}_{LP} \leq \mathrm{OPT}$ and we have established that $\mathrm{OPT} > \omega/\epsilon$, we can bound the cost of our solution as

$$\mathrm{Cost}(\bar{x}) < f \cdot \mathrm{OPT} + \epsilon \cdot \mathrm{OPT} = (f + \epsilon) \cdot \mathrm{OPT} .$$

This provides an $(f + \epsilon)$-approximation for the problem. ◀

4 Weighted Partition Edge Cover

We consider the following generalization from [3]:

▶ **Definition 15** (Weighted Budgeted Matching). *We are given a graph $G = (V, E)$ with a weight function $p : E \to \mathbb{R}_{\geq 0}$, a partition of the vertex set $V = V_1 \cup \cdots \cup V_\omega$, and a parameter ρ_g for each group. We want to find a minimum-weight matching such that at least ρ_g vertices from group V_g are matched.*

First, we note that the reduction between Partition-EC and Budgeted Matching in Lemma 5 [3] holds in the weighted case.

▶ **Lemma 16** ([3]). *If Weighted Budgeted Matching can be solved in time $T(n, m)$, then Partition-EC can be solved in time $T(2n, m + n) + O(m + n)$. Here n and m refer to the number of nodes and edges respectively.*

In order to be matched or covered at minimum cost, a vertex will pay for its least expensive adjacent edge, so we can modify the reduction by giving the edge between each vertex and its auxillary vertex this minimum adjacent weight. Moreover, we eliminate the dependence on ω.

Now, we give a simple reduction to the standard problem of finding a maximum weight matching in a graph. This allows us to generalize the problem to weighted graphs, improve the running time, and avoid the reduction to tropical matching completely. Our algorithm also admits the use of standard libraries for weighted matching.

1. Let M be large with $M > \sum_e p_e$.
2. Let H be a copy of G with the following vertex sets added. For each group g add a set V'_g of $|V_g| - \rho_g$ nodes. Add edges between all nodes in V'_g and V_g for each g. The weight of an edge e in H is $2M - p_e$ when $e \in E$, and the weight of an edge (x, y) where $x \in V'_g, y \in V_g$ is M. See Figure 1 for an example
3. Find a maximum weight matching $\mathcal{M}_H$ in H using the algorithm by Gabow [17, 18] The matched edges induced by the nodes of G should be a minimum weight matching satisfying the group requirements. If the weight of the maximum weight matching is at most $(n-1)M$ then no solution exists.

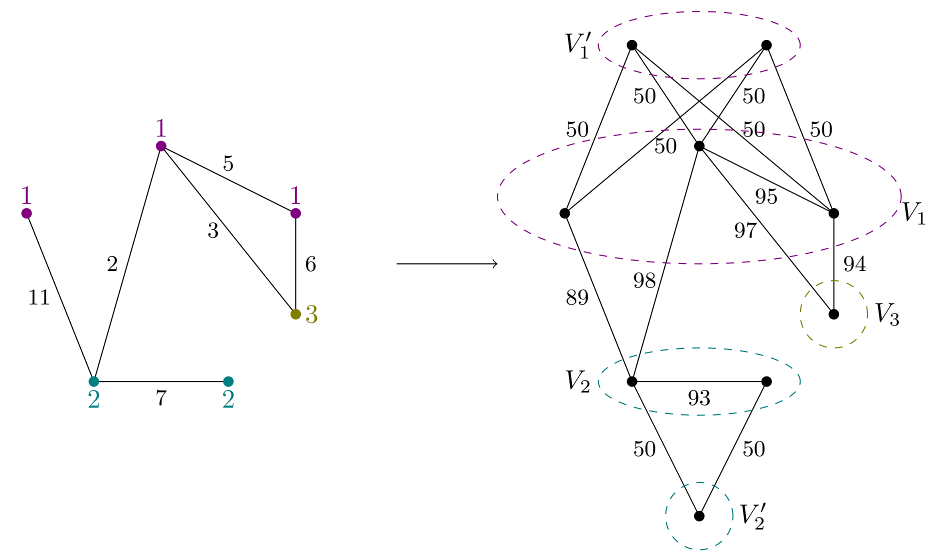

Figure 1 The reduction from Weighted Budgeted Matching to Maximum Matching for an instance with $\rho_1 = \rho_2 = \rho_3 = 1$. Here we let $M = 50$.

▶ **Lemma 17.** *If maximum-weight matching can be solved in time $T(n, m)$, then Weighted Budgeted Matching can be solved in time $T(2n, m + n^2)$.*

Proof. For each group g we add $|V'_g| \le |V_g|$ vertices for a total of at most n additional vertices. We then add a complete bipartite graph with $|V'_g| \cdot |V_g|$ edges for each color, which totals at most n^2 additional edges. ◀

▶ **Lemma 18** (†)**.** *In $\mathcal{M}_H$, at least ρ_g vertices from each group V_g are matched by edges from G.*

▶ **Lemma 19** (†). *The matching $\mathcal{M} = \mathcal{M}_H \cap E(G)$ with original edge weights has total weight equal to the optimum of the Weighted Budgeted Matching problem.*

To speed up the algorithm, we do a reduction to the problem of finding a max weight subgraph with upper bounds on degrees (f-factor). We contract the set V_g' into a single node V_g' with edges to all vertices in V_g of weight M and give a degree constraint of $|V_g| - \rho_g$ to V_g'. All nodes of V will have a degree constraint of 1. This new graph has only $n + \omega$ vertices, and at most n extra edges, preserving the sparsity of the graph. We now use the algorithm by Gabow [18] (see also [19]) which gives $O(nm + n^2 \log n)$ running time.

Lemmas 16, 17, 18 and 19 together imply Theorem 5.

5 Conclusion

While our algorithm achieves an $(f + \epsilon)$-approximation for the hard-capacitated problem, the algorithm for the weighted soft-capacitated (SC-PVC) variant has an approximation factor of $f + 1$. Bridging this gap by designing an f-approximation algorithm for the weighted SC-PVC problem remains an intriguing open question. A natural question is about the dependence on ω in the run-time. Is it possible to derive an f approximation in time $O(f(\omega)n^c)$ so that the dependence on ω in the run-time is not $O(n^\omega)$? We already know that the problem is Set Cover hard for unbounded ω. The same runtime issue arises in the k-center problem as mentioned earlier (in related work).

Another interesting direction for future research is to develop an efficient combinatorial algorithm for the Partition Vertex Cover problem, even in the uncapacitated case. While such primal-dual methods exist for the related Partial Vertex Cover problem [34], developing one for the partition generalization is an open question.

For the distinct but related problem of maximizing edge coverage given a budget on the cardinality of the chosen vertex cover, Marx [33] shows how to obtain an FPT approximation scheme for unweighted simple graphs for $\omega = 1$. It will be interesting to obtain similar results for arbitrary ω.

References

1 Georg Anegg, Haris Angelidakis, Adam Kurpisz, and Rico Zenklusen. A technique for obtaining true approximations for k-center with covering constraints. *Math. Program.*, 192(1):3–27, 2022. doi:10.1007/S10107-021-01645-Y.

2 Tanvi Bajpai, Chandra Chekuri, and Pooja Kulkarni. Covering a few submodular constraints and applications. In Alina Ene and Eshan Chattopadhyay, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2025, August 11-13, 2025, Berkeley, CA, USA*, volume 353 of *LIPIcs*, pages 25:1–25:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.APPROX/RANDOM.2025.25.

3 Sayan Bandyapadhyay, Aritra Banik, and Sujoy Bhore. On colorful vertex and edge cover problems. *Algorithmica*, 85(12):3816–3827, 2023. doi:10.1007/S00453-023-01164-6.

4 Sayan Bandyapadhyay, Tanmay Inamdar, Shreyas Pai, and Kasturi R. Varadarajan. A constant approximation for colorful k-center. In Michael A. Bender, Ola Svensson, and Grzegorz Herman, editors, *27th Annual European Symposium on Algorithms, ESA 2019, September 9-11, 2019, Munich/Garching, Germany*, volume 144 of *LIPIcs*, pages 12:1–12:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.ESA.2019.12.

5 Reuven Bar-Yehuda. Using homogenous weights for approximating the partial cover problem. In Robert Endre Tarjan and Tandy J. Warnow, editors, *Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms, 17-19 January 1999, Maryland, USA*, pages 71–75. ACM/SIAM, 1999. URL: http://dl.acm.org/citation.cfm?id=314500.314533.

6 Reuven Bar-Yehuda and Shimon Even. A linear-time approximation algorithm for the weighted vertex cover problem. *J. Algorithms*, 2(2):198–203, 1981. `doi:10.1016/0196-6774(81)90020-1`.

7 Reuven Bar-Yehuda and Shimon Even. A local-ratio theorem for approximating the weighted vertex cover problem. In Manfred Nagl and Jürgen Perl, editors, *Proceedings of the WG '83, International Workshop on Graphtheoretic Concepts in Computer Science, June 16-18, 1983, Haus Ohrbeck, Germany*, pages 17–28. Universitätsverlag Rudolf Trauner, Linz, 1983.

8 Reuven Bar-Yehuda, Guy Flysher, Julián Mestre, and Dror Rawitz. Approximation of partial capacitated vertex cover. *SIAM J. Discret. Math.*, 24(4):1441–1469, 2010. `doi:10.1137/080728044`.

9 Suman Kalyan Bera, Shalmoli Gupta, Amit Kumar, and Sambuddha Roy. Approximation algorithms for the partition vertex cover problem. *Theor. Comput. Sci.*, 555:2–8, 2014. `doi:10.1016/J.TCS.2014.04.006`.

10 Nader H. Bshouty and Lynn Burroughs. Massaging a linear programming solution to give a 2-approximation for a generalization of the vertex cover problem. In Michel Morvan, Christoph Meinel, and Daniel Krob, editors, *STACS 98, 15th Annual Symposium on Theoretical Aspects of Computer Science, Paris, France, February 25-27, 1998, Proceedings*, volume 1373 of *Lecture Notes in Computer Science*, pages 298–308. Springer, 1998. `doi:10.1007/BFB0028569`.

11 Wang Chi Cheung, Michel X. Goemans, and Sam Chiu-wai Wong. Improved algorithms for vertex cover with hard capacities on multigraphs and hypergraphs. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1714–1726. SIAM, 2014. `doi:10.1137/1.9781611973402.124`.

12 Julia Chuzhoy and Joseph Naor. Covering problems with hard capacities. *SIAM J. Comput.*, 36(2):498–515, 2006. `doi:10.1137/S0097539703422479`.

13 Rajni Dabas, Samir Khuller, and Emilie Rivkin. Weighted partition vertex and edge cover. *CoRR*, abs/2508.13055, 2025. `doi:10.48550/arXiv.2508.13055`.

14 Rajni Dabas, Samir Khuller, and Emilie Rivkin. Serving clients fairly: On facility location and k-median with fair outliers. In Huijia (Rachel) Lin, editor, *7th Symposium on Foundations of Responsible Computing, FORC 2026, Harvard University, Cambridge, MA, USA, June 3-5, 2026*, volume 368 of *LIPIcs*, pages 9:1–9:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2026. `doi:10.4230/LIPIcs.FORC.2026.9`.

15 Jennifer Dill and Nathan McNeil. Are shared vehicles shared by all? a review of equity and vehicle sharing. *Journal of Planning Literature*, 36 (1):5–30, 2020. `doi:10.1177/0885412220966732`.

16 Irit Dinur and David Steurer. Analytical approach to parallel repetition. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 624–633. ACM, 2014. `doi:10.1145/2591796.2591884`.

17 Harold N. Gabow. Data structures for weighted matching and nearest common ancestors with linking. In David S. Johnson, editor, *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms, 22-24 January 1990, San Francisco, California, USA*, pages 434–443. SIAM, 1990. URL: `http://dl.acm.org/citation.cfm?id=320176.320229`.

18 Harold N. Gabow. Data structures for weighted matching and extensions to b-matching and f-factors. *ACM Trans. Algorithms*, 14(3):39:1–39:80, 2018. `doi:10.1145/3183369`.

19 Harold N. Gabow and Piotr Sankowski. Algorithms for weighted matching generalizations II: f-factors and the special case of shortest paths. *SIAM J. Comput.*, 50(2):555–601, 2021. `doi:10.1137/16M1106225`.

20 Rajiv Gandhi, Eran Halperin, Samir Khuller, Guy Kortsarz, and Aravind Srinivasan. An improved approximation algorithm for vertex cover with hard capacities. *J. Comput. Syst. Sci.*, 72(1):16–33, 2006. `doi:10.1016/J.JCSS.2005.06.004`.

21 Rajiv Gandhi, Samir Khuller, and Aravind Srinivasan. Approximation algorithms for partial covering problems. In Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen, editors, *Automata, Languages and Programming, 28th International Colloquium, ICALP 2001, Crete, Greece, July 8-12, 2001, Proceedings*, volume 2076 of *Lecture Notes in Computer Science*, pages 225–236. Springer, 2001. doi:10.1007/3-540-48224-5_19.

22 Fănică Gavril. Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. *SIAM Journal on Computing*, 1(2):180–187, 1972. doi:10.1137/0201013.

23 Sudipto Guha, Refael Hassin, Samir Khuller, and Einat Or. Capacitated vertex covering. *J. Algorithms*, 48(1):257–270, 2003. doi:10.1016/S0196-6774(03)00053-1.

24 Dorit S. Hochbaum. Approximation algorithms for the set covering and vertex cover problems. *SIAM J. Comput.*, 11(3):555–556, 1982. doi:10.1137/0211045.

25 Eunpyeong Hong and Mong-Jen Kao. Approximation algorithm for vertex cover with multiple covering constraints. *Algorithmica*, 84(1):1–12, 2022. doi:10.1007/s00453-021-00885-w.

26 David Ingold and Spencer Soper. Amazon doesn't consider the race of its customers. should it?, April 2016. Accessed: 2026-01-20. URL: https://www.bloomberg.com/graphics/2016-amazon-same-day/.

27 Xinrui Jia, Kshiteej Sheth, and Ola Svensson. Fair colorful k-center clustering. *Math. Program.*, 192(1):339–360, 2022. doi:10.1007/S10107-021-01674-7.

28 Mong-Jen Kao. Iterative partial rounding for vertex cover with hard capacities. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, January 16-19*, pages 2638–2653. SIAM, 2017. doi:10.1137/1.9781611974782.174.

29 Mong-Jen Kao, Jia-Yau Shiau, Ching-Chi Lin, and D. T. Lee. Tight approximation for partial vertex cover with hard capacities. *Theor. Comput. Sci.*, 778:61–72, 2019. doi:10.1016/J.TCS.2019.01.027.

30 Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972. doi:10.1007/978-1-4684-2001-2_9.

31 Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-ϵ. *J. Comput. Syst. Sci.*, 74(3):335–349, May 2008. doi:10.1016/j.jcss.2007.06.019.

32 Xiaofei Liu and Weidong Li. Approximation algorithm for prize-collecting hypergraph vertex cover with fairness constraints. In *Computing and Combinatorics (COCOON 2025)*, volume 15984 of *Lecture Notes in Computer Science*, pages 41–52, Singapore, 2026. Springer. doi:10.1007/978-981-95-0218-9_4.

33 Dániel Marx. Parameterized complexity and approximation algorithms. *Comput. J.*, 51(1):60–78, 2008. doi:10.1093/COMJNL/BXM048.

34 Julián Mestre. A primal-dual approximation algorithm for partial vertex cover: Making educated guesses. *Algorithmica*, 55(1):227–239, 2009. doi:10.1007/S00453-007-9003-Z.

35 George L. Nemhauser and Leslie E. Trotter Jr. Vertex packings: Structural properties and algorithms. *Math. Program.*, 8(1):232–248, 1975. doi:10.1007/BF01580444.

36 Barna Saha and Samir Khuller. Set cover revisited: Hypergraph cover with hard capacities. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part I*, volume 7391 of *Lecture Notes in Computer Science*, pages 762–773. Springer, 2012. doi:10.1007/978-3-642-31594-7_64.

37 Sam Chiu-wai Wong. Tight algorithms for vertex cover with hard capacities on multigraphs and hypergraphs. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, 16-19*, pages 2626–2637. SIAM, 2017. doi:10.1137/1.9781611974782.173.

A Remaining Proofs from Section 2

▶ **Lemma 6** (†). *Let $\langle x, y\rangle$ be a feasible solution to the linear relaxation of IP-SC such that x is integral. In polynomial time, we can compute an integral solution $\hat{y}$ such that $\langle x, \hat{y}\rangle$ is feasible.*

Proof. We present the construction for completeness. Consider the input instance and a solution $\langle x, y\rangle$ with x integral. We construct a network flow instance F. The nodes of F are $s \cup E \cup (V \cup \{U_1, \ldots, U_\omega\}) \cup t$. The source and sink are s and t, respectively. The nodes $U_1, \ldots, U_\omega$ correspond to the uncovered sets for each group. We define the arcs and capacities:

- (s, e) for each $e \in E$ with capacity 1;
- (e, v) for each $v \in e$ with capacity 1;
- (e, U_g) where $e \in E_g$ with capacity 1;
- (v, t) for each $v \in V$ with capacity $k_v x_v$;
- (U_g, t) for each $g = 1, \ldots, \omega$ with capacity $|E_g| - \rho_g$.

We know that $\langle x, y\rangle$ is feasible, so there must be a flow of $|E|$ in F from s to t. The capacity on each arc is integral, so there exists and integral flow of value $|E|$ from s to t. Define $\hat{y}_{e,v} = 1$ (meaning that $\varphi(e) = v$) if the flow (e, v) is 1 and $\hat{y}_{e,v} = 0$ otherwise. If an edge goes unassigned, meaning that the flow from e to U_g is 1, $\varphi(e) = \perp$. Then, $\langle x, \hat{y}\rangle$ is a feasible integral solution. ◀

▶ **Lemma 9** (†). *For all $v \in H$,*

$$\sum_{e\in\delta(v)} \hat{y}_{e,v} \leq \frac{f+1}{2} k_v x_v \leq k_v \hat{x}_v.$$

Moreover, if $v \in H \cap R$, $\hat{x}_v \leq (f+1)x_v$, and if $v \in M$, $\hat{x}_v \leq f \cdot x_v$.

Proof. We only consider vertices affected by rounding cases (1) and (3b). For (1), the value of the edge variable is unchanged. In the case of (3b), we know that for all $v \in e \cap L$, $y_{e,v} < \frac{c_e}{f+1}$. We consider the worst possible case for rounding here. An endpoint in H handles $1 + \frac{c_e^\ell}{c_e^h}$ times more edge coverage, which increases as the mass c_e^ℓ increases. This added responsibility is therefore maximized when the edge contains $f-1$ endpoints in L and a single endpoint in $v_h \in H$. Hence it suffices to analyze this worst case. Then,

$$\hat{y}_{e,v_h} = c_e = y_{e,v_h} + \sum_{v\in e\cap L} y_{e,v} < y_{e,v_h} + (f-1)\frac{c_e}{f+1}.$$

We can then conclude that

$$\hat{y}_{e,v_h} = c_e < \frac{f+1}{2} y_{e,v_h}.$$

Taking the sum over edges incident to v gives us the first inequality.

Now, we bound $\hat{x}_v$. For $v \in H \cap R$, we consider two cases. If $\frac{1}{f+1} \leq x_v \leq \frac{2}{f+1}$, then

$$\hat{x}_v = \left\lceil \frac{f+1}{2} x_v \right\rceil = 1 \leq (f+1)x_v.$$

Otherwise, we have $x_v > \frac{2}{f+1}$. In that case,

$$\hat{x}_v = \left\lceil \frac{f+1}{2} x_v \right\rceil < \frac{f+1}{2} x_v + 1 < (f+1)x_v.$$

For $v \in M$, $x_v \in \mathbb{N}$. Therefore,

$$\hat{x}_v = \left\lceil \frac{f+1}{2} x_v \right\rceil \leq \lceil f \cdot x_v \rceil = f \cdot x_v.$$

where the first inequality follows simply because $\frac{f+1}{2} \leq f$ for $f \geq 2$ and last equality follows because both f and x_v are integers. ◀

B Remaining Proofs from Section 4

▶ **Lemma 18** (†). *In $\mathcal{M}_H$, at least ρ_g vertices from each group V_g are matched by edges from G.*

Proof. Let $\mathcal{M}^* \subseteq G$ be an optimal solution to the budgeted matching problem. We now construct a matching $\mathcal{M}_H^*$ in H by including all edges from $\mathcal{M}^*$, and for each vertex in V_g not matched in $\mathcal{M}^*$, adding an auxiliary edge to a unique free node in V_g'.

Let k_g^* be the number of vertices in V_g matched by $\mathcal{M}^*$. Since $k_g^* \geq \rho_g$, we have, $|V_g| - k_g^* \leq |V_g| - \rho_g = |V_g'|$, so there are enough auxiliary vertices in V_g' to complete the matching without conflict. Thus, $\mathcal{M}_H^*$ is a feasible matching in H that matches all the vertices in V_g for every g. This also ensures that the cost of the maximum matching in H is strictly greater than $(n-1)M$ since each matched node in V contributes M to the weight of the matching, and even after subtracting the weight of a few edges we have a solution of weight between nM and $(n-1)M$. Any matching that does not match all nodes of G (that also belong to H) cannot have weight larger than $(n-1)M$.

Now consider the maximum weight matching $\mathcal{M}_H$ in H. Suppose, for contradiction, that $\mathcal{M}_H$ matches fewer than ρ_g vertices from some group V_g via edges from G. The number of vertices matched by auxiliary edges is at most $|V_g'| = |V_g| - \rho_g$. Therefore, at least one vertex $u \in V_g$ must be unmatched by $\mathcal{M}_H$. Therefore, for large M, $\mathcal{M}_H^*$ is better than $\mathcal{M}_H$, which is a contradiction. ◀

▶ **Lemma 19** (†). *The matching $\mathcal{M} = \mathcal{M}_H \cap E(G)$ with original edge weights has total weight equal to the optimum of the Weighted Budgeted Matching problem.*

Proof. We assume $n = |V|$ is even. This is perfectly fine because the reduction from Partition Edge Cover in Lemma 15 yields a Budgeted Matching instance with exactly $2n$ vertices, ensuring it is always an even-size instance. Let $\mathcal{M}^* \subseteq G$ be an optimal solution to the budgeted matching problem with total weight OPT. As in proof of Lemma 18, we can extend $\mathcal{M}^*$ to a matching $\mathcal{M}_H^*$ in H by adding auxiliary edges of weight M to unmatched vertices in V. Since all $n = |V|$ vertices in V are matched in $\mathcal{M}_H^*$, and exactly $n - 2|\mathcal{M}^*|$ auxiliary edges are used, the total weight of $\mathcal{M}_H^*$ is:

$$\sum_{e \in \mathcal{M}_H^*} p_e^H = \sum_{e \in \mathcal{M}^*} (2M - p_e) + (n - 2|\mathcal{M}^*|)M = nM - \mathrm{OPT}.$$

Let $\mathcal{M}_H$ be a maximum weight matching in H, and let $\mathcal{M} = \mathcal{M}_H \cap E(G)$. Note that $\mathcal{M}_H$ also matches all n vertices in V. Therefore,

$$\sum_{e \in \mathcal{M}_H} p_e^H = \sum_{e \in \mathcal{M}} (2M - p_e) + (n - 2|\mathcal{M}|)M = nM - \sum_{e \in \mathcal{M}} p_e.$$

Since $\mathcal{M}_H$ maximizes the weight,

$$\sum_{e \in \mathcal{M}_H} p_e^H \geq \sum_{e \in \mathcal{M}_H^*} p_e^H \Rightarrow \sum_{e \in \mathcal{M}} p_e \leq \mathrm{OPT}.$$

By Lemma 18, $\mathcal{M}$ is a feasible solution to the budgeted matching problem, so equality must hold and $\mathcal{M}$ is optimal. ◀