

Fare Zone Assignment with Hard Cut Budgets under Bounded Congestion

Alper Ferudun*

October 15, 2026

Abstract

At the Oberwolfach workshop on Combinatorial Optimization in November 2024, Peis, with Pabst, Kauther and Fiorini, posed the following problem (Oberwolfach Report 50/2024, p. 2992). Commodities on a tree have paths, weights and budgets. A set of tree edges, the zone borders, is to be chosen; every commodity pays its weight once for each chosen edge on its path, the path of every commodity may contain at most its budget of chosen edges, and the total payment is to be maximized. The source states that the problem is polynomial on paths and strongly NP-hard on stars, that it has a 2-approximation when all budgets are even, and asks for its approximability when the congestion, the largest number of commodity paths through one edge, is at most k . We answer this question up to polylogarithmic factors in k , partly by citing known results. For $k \leq 2$ the problem is a maximum-weight simple b -matching problem and is solvable in polynomial time on every host graph. For $k \geq 3$ it is strongly NP-hard and APX-hard already on stars with unit weights and budgets, by a reduction from independent set in cubic graphs. For every k it is a k -column-sparse packing integer program; hence known algorithms give the ratios $8k$ and $2k + o(k)$ in expectation, $ek + o(k)$ deterministically, and $(k + 1)/2 + \varepsilon$ when all budgets are 1. Known hardness results for independent set in graphs of maximum degree k transfer to stars: for all sufficiently large k there is no $k/(C \log^4 k)$ -approximation unless $P = NP$ (by a theorem of Chan), and none within $ck/\log^2 k$ under the Unique Games Conjecture and $NP \not\subseteq BPP$ (Austrin, Khot and Safra); for $k = 3$ we only have APX-hardness. For unbounded congestion there is no $|E|^{1-\varepsilon}$ -approximation on stars unless $NP = ZPP$, while an $|E|$ -approximation is trivial. The other two facts stated in the source, polynomiality on paths and the 2-approximation, are proved by total unimodularity, and all hardness found here comes from commodities with budget 1. The cited approximation and hardness theorems are not ours. The closest work, by Hoefer, Kauther, Pabst, Peis and Tran, studies a different model in which a commodity over its budget drops out, and proves APX-hardness on stars with congestion 3 for that model. The exact dependence on k remains open. This is an unrefereed note.

1 Introduction

1.1 The problem

A partition of a transport network into fare zones prices a journey by the number of zone borders it crosses. On a tree, a partition into connected zones is the same as a set of edges, the borders, and a journey along a path P crosses the borders in P . At the Oberwolfach workshop on Combinatorial Optimization in November 2024, Peis, in joint work with Pabst, Kauther and Fiorini, posed the following problem [RSW25, p. 2992]. Given are a tree $G = (V, E)$ and commodities (P_i, w_i, u_i) ,

*Mercury Software GmbH. Email: alper@mercurycodelab.com. GitHub: <https://github.com/AlperTheKing>. Unrefereed preprint. CC BY 4.0.

$i = 1, \dots, m$, where $P_i \subseteq E$ is (the edge set of) a path in G , w_i is its weight and u_i is the “maximum number of allowed cuts on P_i ”. The problem is the integer program

$$\max \sum_{e \in E} y_e w_e \quad \text{subject to} \quad \sum_{e \in P_i} y_e \leq u_i \quad (i = 1, \dots, m), \quad y \in \{0, 1\}^E, \quad (1)$$

where $w_e = \sum_{i: e \in P_i} w_i$. An edge with $y_e = 1$ is a border (a *cut*); every commodity pays its weight once for each cut on its path, so the objective equals $\sum_i w_i |\{e \in P_i : y_e = 1\}|$, and the budget u_i is a hard constraint. The source states, without proof, that (1) is solvable in polynomial time when G is a path and strongly NP-hard when G is a star, and that there is a 2-approximation when all u_i are even, implying a 3-approximation when all $u_i \geq 2$. It then asks what else can be said about approximation results on trees, and in particular about the approximability when the *congestion*, “the maximum number of paths that share the same edge”, is bounded by a parameter k . The problem is record OWR-14298804-012 of the UnsolvedMath collection [ula26], where it is listed as partially solved; the statement in the record is a garbled copy of the source, and we work with the source.

This note is about the second question. We write FZ_k for (1) restricted to instances of congestion at most k , and we ask, for each k , for the best ratio of a polynomial-time approximation algorithm. In Section 2 the congestion counts commodities, which gives the same results as counting distinct paths (Remark 2.2); weights are nonnegative (the source gives no sign condition), and budgets are nonnegative integers.

1.2 Results

The following theorem collects the answer to the congestion question; the parts are proved, or derived from the cited theorems, in Sections 3 to 5.

Theorem 1.1. *The following hold for instances of (1) with nonnegative weights.*

- (a) FZ_2 is a maximum-weight simple b -matching problem and can be solved exactly in polynomial time. This holds on every host graph, indeed for every family of sets in which each element lies in at most two sets (Theorem 3.3).
- (b) FZ_3 is strongly NP-hard and APX-hard, already on stars with all weights 1, all budgets 1 and all paths of two edges (Theorem 4.2).
- (c) For every k , FZ_k has polynomial-time approximation algorithms with the ratios $8k$ and $2k + o(k)$ (randomized; the ratio holds in expectation), $ek + o(k)$ (deterministic) and, if all budgets are 1, $\frac{k+1}{2} + \varepsilon$ for every fixed $\varepsilon > 0$ (Theorem 5.1).
- (d) There are absolute constants $C, c > 0$ such that for all sufficiently large k the problem FZ_k , restricted to stars with all budgets 1, has no polynomial-time $k/(C(\log k)^4)$ -approximation unless $\text{P} = \text{NP}$, and no polynomial-time $ck/(\log k)^2$ -approximation if the Unique Games Conjecture holds and $\text{NP} \not\subseteq \text{BPP}$ (Theorem 4.3).
- (e) For every $\varepsilon > 0$, problem (1) on stars, with unbounded congestion, has no polynomial-time $|E|^{1-\varepsilon}$ -approximation unless $\text{NP} = \text{ZPP}$, and by a later theorem none unless $\text{P} = \text{NP}$; the best single edge that may be cut is an $|E|$ -approximation on every instance (Corollary 4.4).

Parts (a) and (b) are proved here, (b) from the APX-hardness of independent set in cubic graphs [AK97, AK00]. Parts (c), (d), (e) apply known theorems [BKNS12, BSSX20, Ber00], [Cha16, AKS11], [Hå99, Zuc07] through reductions proved here. Given these theorems, the best ratio for FZ_k is $\Theta(k)$

congestion	complexity	ratio achieved	lower bound on the ratio
$k \leq 2$	polynomial on every host graph (Theorem 3.3)	1	—
$k = 3$	strongly NP-hard and APX-complete, already on stars (Theorem 4.2)	24 in expectation; 1.786 if all $u_i = 1$	a constant > 1 unless $P = NP$
large k	as for $k = 3$	$ek + o(k)$; $2k + o(k)$ in expectation; $\frac{k+1}{2} + \varepsilon$ if all $u_i = 1$	$k/(C \log^4 k)$ unless $P = NP$; $ck/\log^2 k$ under UGC and $NP \not\subseteq BPP$
unbounded	as for $k = 3$	$ E $	$ E ^{1-\varepsilon}$ unless $P = NP$

Table 1: Approximability of the hard-cap fare zone problem (1) with congestion k (trees; the lower bounds hold already on stars with all budgets 1). Upper bounds from Theorem 5.1, Corollary 4.4; lower bounds from Theorems 4.2 and 4.3, Corollary 4.4.

up to polylogarithmic factors: for all sufficiently large k it lies between $k/(C(\log k)^4)$ (unless $P = NP$) and $ek + o(k)$ (Corollary 5.2); for $k = 3$ the lower bound is only APX-hardness (Table 1).

On stars of congestion k the problem lies between independent set in graphs of maximum degree k (unweighted) and weighted independent set in graphs of maximum degree k ; this is a sandwich, not an equivalence (Proposition 4.5). Section 6 proves the other two facts of the source by total unimodularity: (1) is polynomial on paths (Proposition 6.2), and splitting the paths at their highest vertices gives the ratio $\max_i u_i / \lfloor u_i/2 \rfloor$, which is 2 for even budgets and at most 3 for budgets at least 2 (Proposition 6.3). All hardness found here comes from commodities with budget 1 (Remark 6.4).

1.3 What is new, what is not, and what stays open

The problem and the three facts quoted above are due to the proposers [RSW25, p. 2992]; the source gives no proofs, and our proofs in Sections 4 and 6 may well be theirs.

The closest work we know is the paper of Hoefer, Kautner, Pabst, Peis and Tran [HKP⁺25]. In its model a commodity whose path contains more than u_i cut edges is not forbidden but *drops out* and pays nothing, and a served commodity pays $w_i f(|P_i \cap F|)$ for a non-decreasing subadditive function f of the number of cuts [HKP⁺25, Definition 1]. For this drop-out model they give $O(\log n)$ - and $O(\log n / \log \log n)$ -approximations in expectation on trees with n vertices, solve exactly the instances in which all paths share an end vertex, prove APX-hardness on stars with maximum congestion 3 [HKP⁺25, Theorem 18] and strong NP-hardness on paths [HKP⁺25, Theorem 22], and treat paths further (an outlined PTAS, and exact algorithms parameterized by the congestion and by other parameters). Their Theorem 18 is the drop-out analogue of Theorem 1.1(b) and precedes it; the proofs differ (from Max 2-Sat with bounded occurrences there, from cubic independent set here), and neither statement implies the other. The models differ genuinely: on paths (1) is polynomial, the drop-out model strongly NP-hard. The paper contains no approximation result for its problem parameterized by the congestion, and we claim nothing about the drop-out model. A 2024 seminar abstract of Pabst [Pab24] announces, besides the drop-out model, a variant that is polynomial on paths, as joint work with Kautner, Müller and Peis; (1) may be that variant.

Not ours are the algorithms for column-sparse packing [BKNS12, BSSX20] and weighted set packing [Ber00, TW23, TX26], the hardness results for independent set [AK97, AK00, Cha16, AKS11, Hå99, Zuc07], Tutte’s gadget [Tut54] with Edmonds’ matching algorithm [Edm65], and total unimodularity [HK56, FG65]. New, to our knowledge, is the classification of the bounded-congestion problem (1) with hard budgets in the form of Theorem 1.1, with its proofs: congestion at most 2 as simple b -matching (an instance of the standard correspondence between 0/1 packing programs

with at most two nonzero entries per column and simple b -matching problems; for unit budgets it is weighted matching, as noted in [BKNS12, Section 1.2]), the reductions between stars and independent set in graphs of bounded degree (Lemma 4.1, Proposition 4.5), and the consequences for bounded and unbounded congestion. These observations are elementary, and we do not claim that the proposers were unaware of them; the searches are described under “Scope and priority” at the end.

Open remain the exact dependence on k , that is, the polylogarithmic gap between $k/(C(\log k)^4)$ and $ek + o(k)$ (on stars this gap is tied to that for independent set in graphs of bounded degree, Proposition 4.5), the best constants for small $k \geq 3$, and the first, open-ended question of the source beyond the statements made here. The corpus record therefore remains partially solved.

2 The model

Definition 2.1. An *instance* consists of a finite graph $G = (V, E)$ (the host graph; a tree in the source), and commodities (P_i, w_i, u_i) , $i = 1, \dots, m$, where $P_i \subseteq E$ is the edge set of a path in G with at least one edge, $w_i \geq 0$ is a real number and $u_i \geq 0$ is an integer; several commodities may have the same path. The *edge weights* are $w_e = \sum_{i: e \in P_i} w_i$. A vector $y \in \{0, 1\}^E$ is *feasible* if $\sum_{e \in P_i} y_e \leq u_i$ for all i ; its *value* is $w(y) = \sum_e y_e w_e$, and OPT is the largest value of a feasible vector. For $e \in E$ let $S_e = \{i : e \in P_i\}$. The *congestion* is $\text{cong} = \max_{e \in E} |S_e|$, and FZ_k is the problem restricted to instances with $\text{cong} \leq k$.

The zero vector is feasible, so $\text{OPT} \geq 0$. A ρ -*approximation algorithm* ($\rho \geq 1$) returns a feasible y with $w(y) \geq \text{OPT}/\rho$; for a randomized one the output is always feasible and the inequality holds in expectation. Theorems 3.3 and 5.1 hold for any host graph, indeed for any family of subsets P_i of a finite set E .

Remark 2.2 (merging identical paths). If $P_i = P_j$ for $i \neq j$, replace the two commodities by one commodity with path P_i , weight $w_i + w_j$ and budget $\min(u_i, u_j)$. Every edge weight w_e is unchanged, and so is the set of feasible vectors, since the two constraints on $\sum_{e \in P_i} y_e$ are equivalent to the one with the smaller budget. After merging all groups of identical paths, $|S_e|$ is the number of distinct paths through e . Hence the upper bounds of this note (Theorems 3.3 and 5.1), proved for the congestion counted with commodities, hold also when the congestion is read as the maximum number of distinct paths through an edge; and our lower-bound instances (Lemma 4.1) have pairwise distinct paths, so that for them the two readings agree.

Two simplifications are used repeatedly. Call an edge *blocked* if it lies on the path of a commodity with budget 0, and call a commodity *vacuous* if its budget is at least the number of unblocked edges on its path.

Lemma 2.3 (preprocessing). *Let B be the set of blocked edges, $E' = E \setminus B$, and let I' be the set of commodities that are not vacuous. The program*

$$\max \left\{ \sum_{e \in E'} w_e y_e : \sum_{e \in P_i \cap E'} y_e \leq u_i \ (i \in I'), \ y \in \{0, 1\}^{E'} \right\} \quad (2)$$

has the optimal value OPT , and a feasible (optimal) solution of (2), extended by $y_e = 0$ on B , is feasible (optimal) for the instance. In (2) every budget satisfies $1 \leq u_i < |P_i \cap E'|$, and every edge of E' lies in at most cong of the sets $P_i \cap E'$, $i \in I'$.

Proof. Every feasible y vanishes on B . For y vanishing on B , the constraint of a commodity with budget 0 holds, and so does the constraint of a vacuous commodity, since it has at most u_i unblocked edges. Hence the feasible vectors are exactly the extensions by zeros of the feasible solutions of (2),

with the same values. A commodity with budget 0 has no unblocked edge and is vacuous, so every $i \in I'$ has $u_i \geq 1$, and $u_i < |P_i \cap E'|$ by definition of I' . Deleting edges and commodities does not increase the number of sets containing an edge. $\square$

All computations of this kind take polynomial time. Throughout, $\log$ is the natural logarithm.

3 Congestion at most two

In this section E is a finite set, $P_1, \dots, P_m \subseteq E$, $u \in \mathbb{Z}_{\geq 0}^m$, $w \in \mathbb{R}_{\geq 0}^E$ (not necessarily induced by commodity weights), and every element lies in at most two of the sets: $|S_e| \leq 2$ for all e , where $S_e = \{i : e \in P_i\}$. Feasibility and value are defined as in Definition 2.1. Elements with $S_e = \emptyset$ are called *free*.

Let H be the multigraph with vertex set $\{1, \dots, m\} \cup \{z\}$, where z is a new vertex, and with one edge h_e for every non-free element e : its ends are the two elements of S_e if $|S_e| = 2$, and i and z if $S_e = \{i\}$; its weight is $w(h_e) = w_e$. H has no loops, but it may have parallel edges (two sets sharing several elements). Put $b(i) = u_i$ for $i \leq m$ and $b(z) = |E(H)|$. A *simple b -matching* of H is a set $M \subseteq E(H)$ with $\deg_M(v) \leq b(v)$ for every vertex v .

Lemma 3.1. *For $y \in \{0, 1\}^E$ let $M_y = \{h_e : y_e = 1\}$. Then y is feasible if and only if M_y is a simple b -matching of H , and $w(y) = w(M_y) + \sum_{e \text{ free}, y_e=1} w_e$. Consequently*

$$\text{OPT} = \sum_{e \text{ free}} w_e + \max\{w(M) : M \text{ a simple } b\text{-matching of } H\}.$$

Proof. For $i \leq m$ we have $\deg_{M_y}(i) = |\{e : y_e = 1, i \in S_e\}| = \sum_{e \in P_i} y_e$, because $i \in S_e$ if and only if $e \in P_i$; so the constraint of P_i says $\deg_{M_y}(i) \leq b(i)$. At z the bound $\deg_{M_y}(z) \leq |E(H)| = b(z)$ always holds. The weights agree term by term. The map $y \mapsto (M_y, \{e \text{ free} : y_e = 1\})$ is a bijection onto the pairs consisting of a subset of $E(H)$ and a set of free elements, and the free elements occur in no constraint; since $w \geq 0$, an optimal y may set all of them to 1. $\square$

The following construction is a variant of Tutte's reduction of factor problems to matchings [Tut54]; we include the short proof, in the form needed here.

Lemma 3.2. *Let H be a loopless multigraph with capacities $b : V(H) \rightarrow \mathbb{Z}_{\geq 0}$ and edge weights $w \geq 0$, and let $\beta(v) = \min(b(v), \deg_H(v))$. Let G' be the simple graph with the vertices (v, κ) for $v \in V(H)$ and $1 \leq \kappa \leq \beta(v)$ (the copies of v) and, for every edge h of H with ends a, c , two vertices h_a, h_c (the pair of h); its edges are $h_a h_c$, $(a, \kappa) h_a$ for $\kappa \leq \beta(a)$ and $(c, \kappa) h_c$ for $\kappa \leq \beta(c)$, all of weight $w(h)$. For a matching N of G' let M_N be the set of edges h of H such that both vertices of the pair of h are matched by N to copies. Then M_N is a simple b -matching of H with $w(M_N) \geq w(N) - w(E(H))$, and*

$$\max\{w(N) : N \text{ a matching of } G'\} = w(E(H)) + \max\{w(M) : M \text{ a simple } b\text{-matching of } H\}.$$

In particular M_N is a maximum-weight simple b -matching if N is a maximum-weight matching.

Proof. ($\geq$) Let M be a simple b -matching. At every vertex v , $\deg_M(v) \leq b(v)$ and $\deg_M(v) \leq \deg_H(v)$, so $\deg_M(v) \leq \beta(v)$ and the edges of M at v can be assigned to distinct copies of v . For $h = ac \in M$ match h_a and h_c to the copies of a and c assigned to h (weight $2w(h)$); for $h \notin M$ take the edge $h_a h_c$ (weight $w(h)$). No copy and no pair vertex is used twice, so this is a matching of weight $w(E(H)) + w(M)$.

($\leq$) and the statement on M_N . Every edge of G' meets the pair of exactly one edge h of H (the edge $h_a h_c$ meets both of its vertices, every other edge one), and all edges meeting the pair of h have weight $w(h)$. Hence $w(N) = \sum_h w(h) n_h$, where n_h is the number of edges of N meeting the pair of h . Since N covers each of h_a, h_c at most once, $n_h \leq 2$; if $h_a h_c \in N$ then $n_h = 1$; and $n_h = 2$ if and only if h_a and h_c are both matched to copies, that is, $h \in M_N$. So $n_h \leq 1$ for $h \notin M_N$ and $n_h \leq 2$ for $h \in M_N$, and as $w \geq 0$, $w(N) \leq w(E(H)) + w(M_N)$. Finally, distinct edges of M_N at a vertex v are matched to distinct copies of v , so $\deg_{M_N}(v) \leq \beta(v) \leq b(v)$. $\square$

Theorem 3.3. *Let $E, P_1, \dots, P_m, u$ and $w \geq 0$ be as above, with every element of E in at most two of the sets P_i . A feasible y of maximum value can be computed in polynomial time. In particular FZ₂ is solvable in polynomial time on every host graph.*

Proof. Build H and G' ; then $|V(G')| \leq \sum_v \deg_H(v) + 2|E(H)| \leq 4|E|$ and $|E(G')| \leq |E(H)| + \sum_v \deg_H(v)^2 \leq |E| + 2|E|^2$. Compute a maximum-weight matching N of G' in polynomial time [Edm65], and let $y_e = 1$ for the free elements and for the elements with $h_e \in M_N$, and $y_e = 0$ otherwise. By Lemma 3.2 the set M_N is a maximum-weight simple b -matching of H , and by Lemma 3.1 the vector y is feasible with $w(y) = \text{OPT}$. $\square$

Remark 3.4. (a) For congestion 1 every edge of H ends at z , and the problem splits: for each i one cuts the u_i heaviest edges of P_i (all of them if $|P_i| \leq u_i$), and every free edge. (b) The natural linear relaxation of (1), with $0 \leq y \leq 1$, is not integral for congestion 2, so the matching algorithm cannot be replaced by linear programming over this relaxation. On a star with spokes s_1, s_2, s_3 , take three commodities with paths $\{s_1, s_2\}, \{s_2, s_3\}, \{s_1, s_3\}$, weights 1 and budgets 1: the congestion is 2, every spoke has weight 2, $\text{OPT} = 2$, and $y \equiv \frac{1}{2}$ is feasible for the relaxation with value 3. Here H is a triangle.

4 Stars and independent sets

A path in a star has one or two edges (spokes). We write $\alpha(H)$ for the largest size of an independent set of a graph H , and $\alpha_W(H)$ for the largest weight when the vertices carry weights W .

Lemma 4.1. *Let H be a simple graph with vertex set $\{1, \dots, n\}$, $n \geq 1$, and let $d \geq \max(\Delta(H), 1)$ be an integer. Consider the star with centre v and spokes $s_a = va$, $a = 1, \dots, n$, and the following instances.*

- $\Sigma(H, d)$: for every edge ab of H an edge commodity with path $\{s_a, s_b\}$, weight 1 and budget 1; for every vertex a with $\deg_H(a) < d$ a filler with path $\{s_a\}$, weight $d - \deg_H(a)$ and budget 1.
- $\Sigma(H, d, W)$, for vertex weights $W_a \geq 1$: the same edge commodities, and for every vertex a a filler with path $\{s_a\}$, weight $(d + 1)W_a - \deg_H(a)$ and budget 1.

Then:

- (a) In both instances all budgets are 1, all paths have at most two edges and are pairwise distinct, and all weights are positive. In $\Sigma(H, d)$ the congestion is at most d and $w_{s_a} = d$ for every a ; if H is d -regular, there are no fillers, all weights are 1 and the congestion is d . In $\Sigma(H, d, W)$ the congestion is at most $d + 1$ and $w_{s_a} = (d + 1)W_a$.
- (b) In both instances a vector y is feasible if and only if $I_y = \{a : y_{s_a} = 1\}$ is an independent set of H .

- (c) $w(y) = d|I_y|$ in $\Sigma(H, d)$ and $w(y) = (d+1)W(I_y)$ in $\Sigma(H, d, W)$. Hence $\text{OPT} = d\alpha(H)$, resp. $(d+1)\alpha_W(H)$, and for every $\rho \geq 1$ a feasible y with $w(y) \geq \text{OPT}/\rho$ gives an independent set I_y of size at least $\alpha(H)/\rho$, resp. of weight at least $\alpha_W(H)/\rho$.
- (d) The instances are computed in polynomial time, and all their numbers are at most d , resp. at most $(d+1)\max_a W_a$.

Proof. (a) The spoke s_a lies in the $\deg_H(a)$ edge commodities at a and in the filler at a , if there is one. In $\Sigma(H, d)$ this gives congestion $\deg_H(a) \leq d$ at s_a when $\deg_H(a) = d$, and $\deg_H(a) + 1 \leq d$ otherwise, and $w_{s_a} = \deg_H(a) + (d - \deg_H(a)) = d$ in both cases. In $\Sigma(H, d, W)$ the congestion at s_a is $\deg_H(a) + 1 \leq d+1$, the filler weight is at least $(d+1) - d = 1$, and $w_{s_a} = \deg_H(a) + (d+1)W_a - \deg_H(a) = (d+1)W_a$. Distinct edges give distinct paths, and fillers have one spoke. (b) A filler constraint reads $y_{s_a} \leq 1$ and always holds; the constraint of the edge commodity of ab reads $y_{s_a} + y_{s_b} \leq 1$. (c) $w(y) = \sum_a y_{s_a} w_{s_a}$; now use (a) and (b). (d) is clear. $\square$

Theorem 4.2. *Problem (1) is strongly NP-hard and APX-hard on the instances $\Sigma(H, 3)$ with H cubic: stars with congestion 3, all weights 1, all budgets 1 and all paths of two edges. Consequently, for every $k \geq 3$, FZ_k is strongly NP-hard and APX-hard on stars, hence on trees; with Theorem 5.1 it is APX-complete for every fixed $k \geq 3$.*

Proof. Maximum independent set is APX-complete on cubic graphs [AK97, AK00]; so it is NP-hard there, and there is $\delta > 0$ such that no polynomial-time algorithm approximates it within $1 + \delta$ on cubic graphs unless $P = NP$. For a cubic graph H the instance $\Sigma(H, 3)$ has no fillers, and by Lemma 4.1(a) it is of the stated form. By Lemma 4.1(c), $\text{OPT} = 3\alpha(H)$, so deciding whether $\text{OPT} \geq 3t$ decides whether $\alpha(H) \geq t$, and a $(1 + \delta)$ -approximation algorithm for these instances would give one for independent set in cubic graphs. All numbers in the instances are 1, so the NP-hardness is strong. Since the reduction preserves the approximation ratio exactly, it is approximation-preserving, and APX-hardness transfers. Instances of congestion 3 have congestion at most k for every $k \geq 3$. Membership in APX for fixed k follows from the deterministic constant-factor algorithm of Theorem 5.1(b). $\square$

Theorem 4.3. *Let $k \geq 1$. A polynomial-time ρ -approximation algorithm for FZ_k on stars with all budgets 1 yields a polynomial-time ρ -approximation algorithm for maximum independent set in graphs of maximum degree at most k . Consequently there are absolute constants $C, c > 0$ such that for all sufficiently large k :*

- (i) unless $P = NP$, no polynomial-time algorithm approximates FZ_k on stars with all budgets 1 within $k/(C(\log k)^4)$;
- (ii) if the Unique Games Conjecture holds and $NP \not\subseteq BPP$, no polynomial-time algorithm approximates FZ_k on stars with all budgets 1 within $ck/(\log k)^2$.

Proof. For a graph H of maximum degree at most k , the instance $\Sigma(H, k)$ of Lemma 4.1 has congestion at most k and all budgets 1, and by Lemma 4.1(c) a ρ -approximate solution gives a ρ -approximate independent set. This proves the first statement; the two consequences are known hardness results for independent set, transported by it.

(i) Chan [Cha16, Theorem 1.4 of the ECCC version] proves that for all sufficiently large D there is $\nu = o_{n,D}(1)$ such that it is NP-hard to approximate independent set on graphs of degree D beyond $O(\log D)^4/D + \nu$. Here the hardness ratio is the ratio of the soundness to the completeness value, the hard graphs are unweighted and have maximum degree at most D [Cha16, Appendix D of the ECCC version], and for fixed D the term ν tends to 0 as the size of the instance grows, so that it is absorbed into the constant C for all instances of sufficient size. Apply this with $D = k$.

(ii) Austrin, Khot and Safra [AKS11, Theorem 1.2] prove that it is Unique-Games-hard, under randomized reductions, to approximate independent set in graphs of degree d within a factor $O(d/\log^2 d)$, a statement asymptotic in d [AKS11, Section 6]; their hard graphs are unweighted, of maximum degree at most d , with independent sets of size $\Theta(1/\log d) \cdot n$ in the completeness case and none of size $\Theta(\log d/d) \cdot n$ in the soundness case [AKS11, Section 1, Theorem 4.1]. Under the Unique Games Conjecture this is a randomized polynomial-time reduction from an NP-complete problem, so a polynomial-time algorithm with ratio $ck/\log^2 k$, for a suitable absolute $c > 0$, would put NP into BPP. Apply this with $d = k$. $\square$

For $k = 3$ no explicit constant is claimed here; Theorem 4.2 gives APX-hardness only.

Corollary 4.4. *For every $\varepsilon > 0$ there is no polynomial-time $|E|^{1-\varepsilon}$ -approximation algorithm for (1) on stars with all budgets 1 unless $\text{NP} = \text{ZPP}$ [Hå99], and none unless $\text{P} = \text{NP}$ [Zuc07]. On the other hand, on every host graph, the vector with a single 1 at an unblocked edge of maximum weight (or $y = 0$ if every edge is blocked) is an $|E|$ -approximation.*

Proof. Let H be a graph with $n \geq 2$ vertices and $d = \max(\Delta(H), 1)$. The star of $\Sigma(H, d)$ has $|E| = n$ edges, and by Lemma 4.1(c) an $|E|^{1-\varepsilon}$ -approximation for it gives an $n^{1-\varepsilon}$ -approximation of $\alpha(H)$. Independent sets of H are the cliques of its complement, which has the same number of vertices; so such an algorithm would contradict Håstad's theorem [Hå99] that maximum clique cannot be approximated within $n^{1-\varepsilon}$ in polynomial time unless $\text{NP} = \text{ZPP}$, and Zuckerman's theorem [Zuc07] that this approximation is NP-hard. For the upper bound, a feasible vector vanishes on the blocked edges (Lemma 2.3), so $\text{OPT} \leq \sum_{e \notin B} w_e \leq |E| \max_{e \notin B} w_e$; and a single unblocked edge e is feasible, since every commodity through e has budget at least 1. $\square$

The next proposition places stars between independent set problems in graphs of bounded degree; it is a sandwich, not an equivalence.

Proposition 4.5. *Let $k \geq 1$.*

- (a) *Every instance on a star with congestion at most k can be transformed in polynomial time into a graph H of maximum degree at most k with vertex weights $w \geq 0$, such that, after the blocked spokes are fixed to 0, the feasible vectors are the incidence vectors of the independent sets of H , with equal values. Hence a ρ -approximation algorithm for weighted independent set in graphs of maximum degree at most k is a ρ -approximation algorithm for FZ_k on stars.*
- (b) *Unweighted independent set in graphs of maximum degree at most k , and, for $k \geq 2$, weighted independent set with positive weights in graphs of maximum degree at most $k - 1$, reduce to FZ_k on stars with all budgets 1, by the instances $\Sigma(H, k)$ and $\Sigma(H, k - 1, W)$ of Lemma 4.1; the reductions preserve approximation ratios exactly.*
- (c) *The converse of (a) fails with the same graph: for every k , the complete graph K_{k+1} with vertex weights $(k + 1, 1, \dots, 1)$ is not the graph H of (a) for any star instance of congestion at most k whose spoke weights are proportional to these vertex weights. For $k = 1$ the same holds for K_2 with the weights 3 and 1.*

Proof. (a) Apply Lemma 2.3. A commodity that is not vacuous has budget 1 and two unblocked spokes s_a, s_b , and its constraint is $y_{s_a} + y_{s_b} \leq 1$. Let H have the unblocked spokes as vertices, with weights $w_{s_a} \geq 0$, and an edge ab whenever some such commodity has the path $\{s_a, s_b\}$. Then y (with $y = 0$ on blocked spokes) is feasible if and only if $\{a : y_{s_a} = 1\}$ is independent in H , the values agree, and $\deg_H(a)$ is at most the number of commodities through s_a , hence at most k . (b) is Lemma 4.1(b),(c), after scaling the weights so that $\min_a W_a = 1$.

(c) Suppose that $H = K_{k+1}$, on the unblocked spokes $s_1, \dots, s_{k+1}$. For every pair $a \neq b$ some non-vacuous commodity has the path $\{s_a, s_b\}$. As every spoke lies in at most k commodities, there is exactly one such commodity for each pair, say of weight $x_{ab} \geq 0$, and no other commodity contains a spoke s_a . Hence $w_{s_a} = \sum_{b \neq a} x_{ab}$ and

$$w_{s_1} = \sum_{b \neq 1} x_{1b} \leq \sum_{b \neq 1} \sum_{c \neq b} x_{bc} = \sum_{b \neq 1} w_{s_b}.$$

Positive multiples of $(k+1, 1, \dots, 1)$ violate this inequality, and so do those of $(3, 1)$ for $k = 1$. $\square$

So algorithms for (a) and hardness results for (b) transfer to stars; we do not claim that every improvement for weighted independent set in graphs of maximum degree k transfers to stars, or conversely.

Remark 4.6. On $\Sigma(K_{k+1}, k)$ the natural relaxation of (1) has value $k(k+1)/2$: the vector $y \equiv \frac{1}{2}$ is feasible, and summing the $\binom{k+1}{2}$ constraints $y_{s_a} + y_{s_b} \leq 1$ shows $\sum_a y_{s_a} \leq (k+1)/2$. Since $\text{OPT} = k$, the ratio of the value of the relaxation to OPT is $(k+1)/2$ on these instances, so the integrality gap of the natural relaxation is at least $(k+1)/2$ already on stars with congestion k .

5 Upper bounds: column-sparse packing

A *k-column-sparse packing integer program* in the normalization of Bansal, Korula, Nagarajan and Srinivasan [BKNS12, Section 1, “Notation”] is a program $\max\{w^\top x : Ax \leq b, x \in \{0, 1\}^n\}$ with $A \in [0, 1]^{m \times n}$, $b \in [1, \infty)^m$, $w \in \mathbb{R}_{\geq 0}^n$, in which every column of A has at most k nonzero entries.

Theorem 5.1. *Let $k \geq 1$. On every host graph, FZ_k has polynomial-time approximation algorithms with the following guarantees.*

- (a) *A randomized algorithm whose output has expected value at least $\text{OPT}/(8k)$ [BKNS12, Theorem 3.3].*
- (b) *A deterministic algorithm with ratio $ek + o(k)$ [BKNS12, Theorem 1.2 and the Remark after its proof]; the theorem gives a randomized algorithm, and the remark explains its derandomization.*
- (c) *A randomized algorithm whose output has expected value at least $\text{OPT}/(2k + o(k))$ [BSSX20, Theorem 1]; there the ratio is $2k + \Theta(k^{0.8} \text{polylog}(k))$ and holds in expectation.*
- (d) *If every commodity that is not vacuous has budget 1 (for instance, if all budgets are 1), an algorithm with ratio $\frac{k+1}{2} + \varepsilon$, for every fixed $\varepsilon > 0$, by Berman’s local search for weighted set packing [Ber00], as stated in [BKNS12, Section 1.2] and [TW23]; for $k = 3$ the algorithm of Thiery and Ward [TW23] has ratio 1.786 (a recent preprint of Tong and Xu [TX26] announces 1.6908).*

Proof. By Lemma 2.3 it suffices to approximate the reduced program (2), and approximate solutions extend by zeros with the same value. Program (2) has the form $\max\{w^\top y : Ay \leq u, y \in \{0, 1\}^{E'}\}$, where the rows of A are the incidence vectors of the sets $P_i \cap E'$, $i \in I'$; so A is a 0/1-matrix with at most k ones in every column, every budget is at least 1, and $w \geq 0$. This is a *k-column-sparse packing integer program* in the normalization above, and (a), (b), (c) are the cited theorems. For (d), let all budgets in (2) be 1, and let $S'_e = \{i \in I' : e \in P_i\}$ for $e \in E'$, a set of size at most k . A vector y is feasible if and only if the sets S'_e with $y_e = 1$ are pairwise disjoint, so after setting $y_e = 1$ for the edges with $S'_e = \emptyset$, which lie in no constraint, (2) is weighted set packing with sets of size at most k ; its conflict graph is $(k+1)$ -claw-free, which is the setting of [Ber00]. For this problem the

ratio $\frac{k+1}{2} + \varepsilon$ is the one stated in [BKNS12, Section 1.2], and 1.786 for $k = 3$ is the main result of [TW23]. $\square$

For fixed small k the asymptotic ratios $ek + o(k)$ and $2k + o(k)$ give unspecified constants; explicit ones are $8k$ in expectation and $\frac{k+1}{2} + \varepsilon$ for unit budgets.

Corollary 5.2. *Assume the theorems cited in Theorems 4.3 and 5.1, and let $\rho^*(k)$ be the infimum of the ratios of deterministic polynomial-time approximation algorithms for FZ_k , either on trees or on stars with all budgets 1. There are absolute constants $C, c > 0$ such that for all sufficiently large k*

$$\rho^*(k) \leq ek + o(k),$$

$$\rho^*(k) \geq \frac{k}{C(\log k)^4} \quad \text{unless } \text{P} = \text{NP}, \quad \rho^*(k) \geq \frac{ck}{(\log k)^2} \quad \text{under UGC and } \text{NP} \not\subseteq \text{BPP},$$

and a randomized algorithm achieves the ratio $2k + o(k)$ in expectation. Moreover $\rho^*(k) = 1$ for $k \leq 2$, and $\rho^*(3) > 1$ unless $\text{P} = \text{NP}$. In this sense the approximability of FZ_k is $\Theta(k)$ up to polylogarithmic factors.

Proof. Combine Theorems 3.3, 4.2, 4.3 and 5.1; the algorithms for trees work in particular on stars, and the lower bounds for stars hold a fortiori on trees. $\square$

6 The facts stated in the source

Of the three facts stated in the source, strong NP-hardness on stars is contained in Theorem 4.2; the other two follow from one lemma on total unimodularity.

Let T be a tree rooted at a vertex r . A set of edges is *vertical* if it is the edge set of the path from a vertex x to a descendant y of x (it is empty if $x = y$). The *bottom edge* of a nonempty vertical set is its edge farthest from r .

Lemma 6.1. *Every 0/1-matrix whose columns are indexed by the edges of T and whose rows are incidence vectors of vertical sets is totally unimodular.*

Proof. Let D be a square submatrix, with rows given by vertical sets $R_1, \dots, R_t$ and columns by a set F of t edges. Contract the edges of T not in F . The result is a tree T' with edge set F , rooted at the image of r , and the ancestor relation between edges of F is the same in T and in T' ; the path from x down to y becomes the path from the image of x down to the image of y , so every $R_j \cap F$ is vertical in T' . Hence it suffices to show $\det D \in \{0, \pm 1\}$ when the columns of D are all t edges of a rooted tree and the rows are t vertical sets; we use induction on $\sum_j |R_j|$. If some row is zero, $\det D = 0$. If two rows $j \neq j'$ (whose sets may coincide) have the same bottom edge f , both sets are final segments of the path from the root to f , so one contains the other, say $R_{j'} \subseteq R_j$; subtracting row j' from row j does not change the determinant and replaces R_j by the vertical set $R_j \setminus R_{j'}$, which decreases $\sum_j |R_j|$. Otherwise the t rows have t distinct bottom edges, so every edge f is the bottom edge of exactly one row $R_{(f)}$. Order the edges so that every edge comes after the edges between it and the root, and order the rows as the $R_{(f)}$ in the same order. The set $R_{(f)}$ consists of f and of edges between f and the root, which come earlier; so the reordered matrix is lower triangular with ones on the diagonal, and $\det D = \pm 1$. $\square$

If A is totally unimodular and u is integral, every vertex of the polytope $Q = \{y : Ay \leq u, 0 \leq y \leq 1\}$ is integral [HK56]: the matrix obtained from A by appending the rows of I and $-I$ is again totally unimodular, a vertex is the unique solution of a nonsingular square subsystem with integral right-hand side, and Cramer's rule applies. A linear program over Q can be solved in polynomial time

[Kha80], and an optimal vertex can be obtained from an optimal solution in polynomial time; for nonnegative w , an integral optimal vertex of $\max\{w^\top y : y \in Q\}$ is an optimal solution of the integer program.

Proposition 6.2. *Problem (1) is solvable in polynomial time, for all budgets, weights and congestions,*

- (a) *when G is a path;*
- (b) *when G is a tree and all paths P_i have a common end vertex.*

Proof. (a) Root the path at one of its ends; then every subpath is vertical. (b) Root the tree at the common end vertex; then every P_i is vertical. In both cases the constraint matrix of (1) is totally unimodular by Lemma 6.1, and the remarks above apply. $\square$

Part (a) is the first fact of the source; its constraint matrix is an interval matrix, whose total unimodularity is classical [FG65]. Instances as in (b) are called rooted in [HKP⁺25], where they are solved exactly for the drop-out model [HKP⁺25, Proposition 2]; they are not mentioned in the source.

Proposition 6.3. *Let G be a tree in which every commodity that is not vacuous has budget at least 2, and let $t = \max u_i / \lfloor u_i/2 \rfloor$ over these commodities ($t = 1$ if there is none). A feasible y with $w(y) \geq \text{OPT}/t$ can be computed in polynomial time. Here $t \leq 2$ if all these budgets are even, and $t \leq 3$ in all cases.*

Proof. Root G at any vertex. Call a commodity *active* if its budget is 0 or if it is not vacuous. Split the path of every active commodity i at its vertex nearest to the root into at most two vertical sets P_i^1, P_i^2 (one of them may be empty), and consider the auxiliary program with the constraints $\sum_{e \in P_i^j} y_e \leq \lfloor u_i/2 \rfloor$ for the active i and $j = 1, 2$, and $y \in \{0, 1\}^E$.

(1) Every y that satisfies the auxiliary constraints is feasible. For an active i , $\sum_{e \in P_i} y_e = \sum_{e \in P_i^1} y_e + \sum_{e \in P_i^2} y_e \leq 2\lfloor u_i/2 \rfloor \leq u_i$. In particular y vanishes on the blocked edges, and then the constraint of a vacuous commodity holds, as it has at most u_i unblocked edges.

(2) The auxiliary constraint matrix is totally unimodular by Lemma 6.1, and its right-hand sides are integers. So an integral optimal vertex $\bar{y}$ of its relaxation with $0 \leq y \leq 1$ is computed in polynomial time, and $\bar{y}$ is feasible by (1).

(3) Let y^* be an optimal solution of the instance. Then y^*/t satisfies the relaxed auxiliary constraints: $0 \leq y^*/t \leq 1$; if $u_i = 0$, then y^* vanishes on P_i ; and otherwise $u_i \geq 2$ and $\sum_{e \in P_i^j} y_e^*/t \leq u_i/t \leq \lfloor u_i/2 \rfloor$ by the choice of t . Hence $w(\bar{y}) \geq w(y^*)/t = \text{OPT}/t$.

Finally $u/\lfloor u/2 \rfloor = 2$ for even $u \geq 2$, and $u/\lfloor u/2 \rfloor = 2 + 2/(u-1) \leq 3$ for odd $u \geq 3$. $\square$

This gives the 2- and 3-approximations of the source; it may well be the proposers' argument. The ratio t is attained by this algorithm on some instances (Section 7); it is not a lower bound for the problem.

Remark 6.4 (budget 1). All hardness found in this note comes from commodities with budget 1. With congestion at most 2 every budget is harmless (Theorem 3.3). If no commodity of budget 1 is non-vacuous, Proposition 6.3 gives a 3-approximation on trees for every congestion, and on stars such instances are trivial (a budget of at least 2 is vacuous there). All lower-bound instances of Theorems 4.2 and 4.3 and Corollary 4.4 have all budgets equal to 1, and the splitting argument fails exactly at budget 1, since $\lfloor 1/2 \rfloor = 0$. We make no further claim about instances that mix commodities of budget 1 with others.

7 Computations and reproducibility

No proof depends on a computation. The program `check_note.py` of the reproducibility package tests the statements on small random instances with fixed seeds, in exact arithmetic (`networkx` is used only for maximum-weight matching with integer weights), in about ten seconds: (i) Lemmas 3.1 and 3.2 and Theorem 3.3 against brute force on 3,000 random trees of congestion at most 2 and on 3,000 set systems that are not path systems (Lemma 3.2 also for 18,000 arbitrary matchings), Remark 3.4(b) and the merge remark; (ii) Lemma 4.1 for every 0/1 vector on all 2,497 pairs (H, d) with H a graph on at most 7 vertices and $d \in \{\max(\Delta(H), 1), \Delta(H) + 1\}$, and on random and cubic graphs; Proposition 4.5(a) on 3,000 random stars; Proposition 4.5(c), Remark 4.6, Corollary 4.4, and the reduction of Theorem 5.1 with its set-packing form; (iii) Lemma 6.1 by the exact determinants of all square submatrices of 2,000 matrices, with a negative control (34 of 500 matrices of arbitrary tree paths are not totally unimodular), and the steps of the proof of Proposition 6.3 on 3,000 random trees. The largest ratio $\text{OPT}/\text{OPT}(\text{split})$ observed is 3, and 2 when the budgets of the non-vacuous commodities are even. All checks pass. The programs and the recorded outputs of the third verification run, which examined the final text (see the next paragraph), are in the folder `independent_run_2/` of the package.

Verification. The results were first obtained with proofs and with three check programs. Three independent verification runs, all AI-assisted, followed; each re-derived the proofs and wrote its own programs. The first two examined the results before this note was written, the third examined the final text.

The first run re-derived every proof, compared the statement with the source (locating the summation signs of (1), which the text layer of the PDF drops) and the quoted theorems with the open versions of [BKNS12, BSSX20, AKS11, Cha16, HKP⁺25], and tested the constructions with its own programs, among them Lemma 4.1 on all graphs with at most 7 vertices and Theorem 3.3 on trees with up to 60 vertices against an integer-programming solver. It found no false theorem, proposition or lemma and no gap in a proof, but an overstated claim and wrong statements about sources, and it required twelve corrections, all made. The mathematical ones: the first version claimed an equivalence between stars of congestion k and weighted independent set in graphs of maximum degree k , where only the sandwich of Proposition 4.5 is proved (the run gave the example of part (c) for $k = 1$); the assumption of Theorem 4.3(ii) must include $\text{NP} \not\subseteq \text{BPP}$; the qualifiers “in expectation” and “ $+\varepsilon$ ” must be kept in all summaries; the merge remark and Corollary 4.4 were to be added. The other seven concern presentation and sources; for instance, the source does not mention rooted instances, and two related papers were described incorrectly. The second run implemented the problem from the source definition, with a brute-force and an integer-programming solver, confirmed Theorem 3.3, Lemma 4.1, Proposition 4.5(a) and Propositions 6.2 and 6.3 on several thousand instances, with negative controls, read the cited theorems in the open versions, repeated the literature search and checked the bibliographic data with Crossref. It found no error and recommended that Theorem 4.3(i) be stated for sufficiently large k , with APX-hardness alone for $k = 3$, that the dependence of Corollary 5.2 on the cited theorems be stated, and that the online date of [BSSX20] be given; this was done. After these runs, Theorem 4.2 was stated for cubic graphs, Proposition 6.3 was restated with blocked edges and vacuous commodities, and Lemma 6.1 was given a self-contained proof; the exact program of this note tests the statements as printed.

The third run examined the final text and the package line by line, as if nothing had been checked. It read the source and the quoted parts of [HKP⁺25, BKNS12, BSSX20, AKS11, Cha16] again, re-derived every proof, and wrote its own programs before reading `check_note.py`. They test Lemma 3.2 for every matching of the gadget graph on 1,726 small instances (more than ten million matchings), Theorem 3.3 against brute force and, on trees with up to 45 vertices, against an integer-programming

solver, Lemma 4.1 on all graphs with at most 7 vertices for three values of d , Proposition 4.5(c) by an exhaustive search over the realizations of K_{k+1} for $k \leq 4$, Lemma 6.1 for all rooted trees with at most 7 edges and all their vertical sets, and Proposition 6.3 on 3,000 instances; seven deliberately wrong variants of the constructions were all detected. The run re-ran the package from the archive, which reproduced the recorded output, and repeated the literature search. It found no mathematical error and no gap, and it required ten corrections of wording and credit, all made: the abstract attributed all facts of the source to total unimodularity; this paragraph said that the first run had found no false statement; the result of the zbMATH search was described inaccurately; Table 1 gave a weaker ratio for $k = 3$ than Theorem 5.1(d); the reduction for congestion at most 2 is now credited as an instance of a standard correspondence; and five points of precision (Proposition 4.5(a), Remark 4.6, Proposition 6.3, the description of [HKP⁺25], and the record of this run).

Scope and priority. The note answers the congestion question of the source up to polylogarithmic factors in k , partly through cited theorems; the first, open-ended question is addressed only by the statements above, and the corpus record remains partially solved. The problem and the three facts of Section 1.1 are due to Peis, Pabst, Kautner and Fiorini [RSW25]. The closest work is [HKP⁺25], on the drop-out model; its Theorem 18, APX-hardness on stars with congestion 3 for that model, precedes our Theorem 4.2. The approximation and hardness theorems for packing problems and for independent set [BKNS12, BSSX20, Ber00, TW23, TX26, AK97, AK00, AKS11, Cha16, Hå99, Zuc07] are cited, not ours; b -matching, including the reduction used in Section 3, and total unimodularity are standard [Tut54, Edm65, HK56, FG65]. New, to our knowledge, is the classification of the bounded-congestion problem with hard budgets in the form of Theorem 1.1, with the reductions of Section 4, the application of the standard b -matching reduction in Section 3, and the consequence for unbounded congestion. What was read: the source in the publisher’s PDF; [HKP⁺25] completely, in version 3; the parts of [BKNS12, BSSX20, AKS11] and of the ECCC version of [Cha16] that contain the quoted statements (the journal version of [Cha16] was not compared); the abstracts of [TW23, TX26, AK97, Zuc07, SU22, SU25] and the seminar abstract [Pab24]. The texts of [AK00, Ber00, Hå99] were not read; what we use from them is stated in the abstract of [AK97], in [BKNS12, Section 1.2] and in [AKS11, Section 1]. The proofs of the cited theorems were not checked. The searches were made on 15 October 2026: the arXiv API (fare zones, zone assignment, tariff zones, zone tariffs, fare structures), the citing works in OpenAlex (five works cite the source volume, none on this problem; none cites [HKP⁺25]), zbMATH (eight documents for “fare zone”, none on this problem), Crossref, and web searches; the verification runs also searched the arXiv listings of the proposers, and the third run repeated these searches on the same day. Apart from [HKP⁺25, Pab24], the work found treats other objectives, such as minimizing deviations from reference prices [HS04, SU25] or finding a cheapest ticket [SU22]. No earlier treatment of the bounded-congestion fare zone problem with hard budgets was found in these sources; a search that finds nothing is not a proof of novelty, and we claim no priority.

AI-assisted tools were used to search the problem corpus and the literature, to find the arguments, to write and run the verification programs, and to help prepare this manuscript. The author has reviewed the proofs, the computations and the sources and is responsible for all claims. The note has not been peer reviewed.

References

- [AK97] Paola Alimonti and Viggo Kann. Hardness of approximating problems on cubic graphs. In *Algorithms and Complexity (CIAC 1997)*, volume 1203 of *Lecture Notes in Computer Science*, pages 288–298. Springer, 1997. doi:10.1007/3-540-62592-5_80.

- [AK00] Paola Alimonti and Viggo Kann. Some APX-completeness results for cubic graphs. *Theoretical Computer Science*, 237(1–2):123–134, 2000. doi:10.1016/S0304-3975(98)00158-3.
- [AKS11] Per Austrin, Subhash Khot, and Muli Safra. Inapproximability of vertex cover and independent set in bounded degree graphs. *Theory of Computing*, 7:27–43, 2011. doi:10.4086/toc.2011.v007a003.
- [Ber00] Piotr Berman. A $d/2$ approximation for maximum weight independent set in d -claw free graphs. In *Algorithm Theory – SWAT 2000*, Lecture Notes in Computer Science, pages 214–219. Springer, 2000. Journal version: *Nordic J. Comput.* 7(3) (2000) 178–184. doi:10.1007/3-540-44985-X_19.
- [BKNS12] Nikhil Bansal, Nitish Korula, Viswanath Nagarajan, and Aravind Srinivasan. Solving packing integer programs via randomized rounding with alterations. *Theory of Computing*, 8:533–565, 2012. doi:10.4086/toc.2012.v008a024.
- [BSSX20] Brian Brubach, Karthik A. Sankararaman, Aravind Srinivasan, and Pan Xu. Algorithms to approximate column-sparse packing problems. *ACM Transactions on Algorithms*, 16(1):1–32, 2020. Published online 15 November 2019. Theorem numbers refer to arXiv version 6. arXiv:1711.02724, doi:10.1145/3355400.
- [Cha16] Siu On Chan. Approximation resistance from pairwise-independent subgroups. *Journal of the ACM*, 63(3):1–32, 2016. Preliminary version: Electronic Colloquium on Computational Complexity, Report TR12-110 (2012), whose theorem numbers we use. doi:10.1145/2873054.
- [Edm65] Jack Edmonds. Maximum matching and a polyhedron with 0,1-vertices. *Journal of Research of the National Bureau of Standards, Section B*, 69B:125–130, 1965. doi:10.6028/jres.069B.013.
- [FG65] Delbert R. Fulkerson and Oliver A. Gross. Incidence matrices and interval graphs. *Pacific Journal of Mathematics*, 15(3):835–855, 1965. doi:10.2140/pjm.1965.15.835.
- [HK56] Alan J. Hoffman and Joseph B. Kruskal. Integral boundary points of convex polyhedra. In Harold W. Kuhn and Albert W. Tucker, editors, *Linear Inequalities and Related Systems*, volume 38 of *Annals of Mathematics Studies*, pages 223–246. Princeton University Press, 1956. doi:10.1515/9781400881987-014.
- [HKP⁺25] Martin Hoefer, Lennart Kautner, Philipp Pabst, Britta Peis, and Khai Van Tran. Fare zone assignment on trees, 2025. Version 3 of 24 April 2026, to which our theorem numbers refer. arXiv:2512.19493.
- [HS04] Horst W. Hamacher and Anita Schöbel. Design of zone tariff systems in public transportation. *Operations Research*, 52(6):897–908, 2004. doi:10.1287/opre.1040.0120.
- [Hå99] Johan Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182(1):105–142, 1999. doi:10.1007/BF02392825.
- [Kha80] Leonid G. Khachiyan. Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics*, 20(1):53–72, 1980. doi:10.1016/0041-5553(80)90061-0.

- [Pab24] Philipp Pabst. Tariff zone assignment: Complexity results and solution approaches. Abstract of a talk on 24 April in the AGCO Seminar, Centro de Modelamiento Matemático, Universidad de Chile, 2024. URL: <https://www.cmm.uchile.cl/wp-content/uploads/2024/04/Philipp-Pabst-24-04-24.pdf>.
- [RSW25] Thomas Rothvoss, Laura Sanità, and Robert Weismantel. Combinatorial optimization. *Oberwolfach Reports*, 21(4):2949–3000, 2025. Report No. 50/2024 of the workshop held 10–15 November 2024. Open problem “Fare Zone Assignment” presented by B. Peis (joint work with P. Pabst, L. Kautner and S. Fiorini), p. 2992. doi:10.4171/OWR/2024/50.
- [SU22] Anita Schöbel and Reena Urban. The cheapest ticket problem in public transport. *Transportation Science*, 56(6):1432–1451, 2022. arXiv:2106.10521, doi:10.1287/trsc.2022.1138.
- [SU25] Anita Schöbel and Reena Urban. Fare structure design in public transport, 2025. arXiv:2502.08228.
- [Tut54] William T. Tutte. A short proof of the factor theorem for finite graphs. *Canadian Journal of Mathematics*, 6:347–352, 1954. doi:10.4153/CJM-1954-033-3.
- [TW23] Theophile Thiery and Justin Ward. An improved approximation for maximum weighted k -set packing. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1138–1162. SIAM, 2023. arXiv:2301.07537, doi:10.1137/1.9781611977554.ch42.
- [TX26] Weitian Tong and Yao Xu. Breaking the $\sqrt{3}$ barrier for maximum weighted 3-set packing, 2026. Version 1 of 7 October 2026. arXiv:2610.10036.
- [ula26] ulamai. UnsolvedMath. Hugging Face dataset, <https://huggingface.co/datasets/ulamai/UnsolvedMath>, 2026. Version 1.6.0; record OWR-14298804-012.
- [Zuc07] David Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. *Theory of Computing*, 3:103–128, 2007. doi:10.4086/toc.2007.v003a006.