2012 IEEE 53rd Annual Symposium on Foundations of Computer Science

LP Rounding for k-Centers with Non-uniform Hard Capacities

(Extended Abstract)

Marek Cygan*, MohammadTaghi Hajiaghayi†, Samir Khuller†
**IDSIA, University of Lugano, Switzerland. Email: marek@idsia.ch*
†*Department of Computer Science, University of Maryland, College Park, USA. Email: {hajiagha,samir}@cs.umd.edu*

***Abstract*—In this paper we consider a generalization of the classical k-center problem with capacities. Our goal is to select k centers in a graph, and assign each node to a nearby center, so that we respect the capacity constraints on centers. The objective is to minimize the maximum distance a node has to travel to get to its assigned center. This problem is NP-hard, even when centers have no capacity restrictions and optimal factor 2 approximation algorithms are known. With capacities, when all centers have identical capacities, a 6 approximation is known with no better lower bounds than for the infinite capacity version.**

While many generalizations and variations of this problem have been studied extensively, no progress was made on the capacitated version for a general capacity function. We develop the first constant factor approximation algorithm for this problem. Our algorithm uses an LP rounding approach to solve this problem, and works for the case of non-uniform *hard* capacities, when multiple copies of a node may not be chosen and can be extended to the case when there is a hard bound on the number of copies of a node that may be selected. Finally, for non-uniform soft capacities we present a much simpler 11-approximation algorithm, which we find as one more evidence that hard capacities are much harder to deal with.

***Keywords*-approximation algorithms; k-center; non-uniform capacities; hard capacities; LP rounding;**

I. Introduction

The k-center problem is a classical facility location problem and is defined as follows: given an edge-weighted graph $G = (V, E)$ find a subset $S \subseteq V$ of size at most k such that each vertex in V is "close" to some vertex in S. More formally, once we choose S the objective function is $\max_{u \in V} \min_{v \in S} d(u, v)$, where d is the distance function (a metric). The problem is known to be NP-hard [2]. Approximation algorithms for the k-center problem have been well studied and are known to be optimal [3]–[6]. In this paper we consider the k-center problem with *non-uniform capacities*. We have a capacity function L defined for each vertex, hence $L(u)$ denotes the capacity of vertex u. The goal is to identify a set S of at most k centers, as well as an *assignment* of vertices to "nearby" centers. No more than $L(u)$ vertices may be assigned to a chosen center at vertex u. Under these constraints we wish to minimize the maximum distance between a vertex v and its assigned center $\phi(v)$. Formally, the cost of a solution S is $\min_{S \subseteq V, |S|=k} \max_{v \in V} d(v, \phi(v))$ such that $|\{v \mid \phi(v) = u\}| \leq L(u) \quad \forall u \in S$ where $\phi : V \to S$.

For the special case when all the capacities are *identical*, a 6 approximation was developed by Khuller and Sussmann [7] improving the previous bound of 10 by Bar-Ilan, Kortsarz and Peleg [8]. In the special case when multiple copies of the same vertex may be chosen, the approximation factor was improved to 5. No improvements have been obtained on these results in the last 15 years. The assumption that the capacities are identical is crucial for both these approaches as it allows one to select centers and then "shift" to a neighboring vertex. In addition, one can use arguments such as $\lceil \frac{N}{L} \rceil$ is a lower bound on the optimal solution; with non-uniform capacities we cannot use such a bound. This problem has resisted any progress at all, and no constant approximation algorithm was developed for the non-uniform capacity version.

In this work we present the first constant factor approximations for the k-center problem with arbitrary capacities. Moreover, our algorithm satisfies hard capacity constraints and only one copy of any vertex is chosen. When multiple copies of a vertex can be chosen then a constant factor approximation is implied by our result for the hard capacity version. For convenience, we discuss the algorithm for the case when at most one copy of a vertex may be chosen. Our algorithms use a novel LP rounding method to obtain the result. In fact this is the first time that LP techniques have been applied for any variation of the k-center problem.

While our constants are large, we do show via integrality gap examples that the problem with non-uniform capacities is significantly harder than the basic k-center problem. In addition we establish that if there is a $(3-\epsilon)$-approximation for the k-center problem with non-uniform capacity constraints then $P = NP$. Such a result is known for the cost k-center problem [9] and from that one can infer the result for the unit cost capacitated k-center problem with non-uniform capacities, but our reduction is a direct reduction from Exact

The full version of this work can be found at [1]. The first author was partially supported by ERC grant NEWNET reference 279352, ERC Starting Grant PAAl 259515, Foundation for Polish Science. The first and second author were supported in part by NSF CAREER award 1053605, ONR YIP award N000141110662, DARPA/AFRL award FA8650-11-1-7162, and a University of Maryland Research and Scholarship Award (RASA). Research of the third author was supported by NSF CCF-0728839, NSF CCF-0937865 and a Google Research Award.

0272-5428/12 $26.00 © 2012 IEEE
DOI 10.1109/FOCS.2012.63

Cover by 3-Sets and considerably simpler. We would like to note that for the k-supplier problem, which is k-center with disjoint sets of clients and potential centers, a simple proof of $(3 - \epsilon)$ approximation hardness under $P \neq NP$ was obtained by Karloff and can be found in [5].

In all cases of studying covering problems, the hard capacity restriction makes the problems very challenging. For example, for the simple capacitated vertex cover problem with soft capacities, a 2 approximation can be obtained by a variety of methods [10], [11] – however imposing a hard capacity restriction makes the problem as hard as set cover [12]. In the special case of unweighted graphs it was shown that a 3 approximation is possible [12], which was subsequently improved to 2 [13].

A. Related Facility Location Work

The *facility location* problem is a central problem in operations research and computer science and has been a testbed for many new algorithmic ideas resulting a number of different approximation algorithms. In this problem, given a metric (via a weighted graph G), a set of nodes called *clients*, and opening costs on some nodes called *facilities*, the goal is to open a subset of facilities such that the sum of their opening costs and connection costs of clients to their nearest open facilities is minimized. When the facilities have capacities, the problem is called the *capacitated facility location* problem. The first constant-factor approximation algorithm for the (uncapacitated) version of this problem was given by Shmoys, Tardos, and Aardal [14] and was based on LP rounding and a filtering technique due to Lin and Vitter [15]. A long series of improvements culminated in a 1.5 approximation due to Byrka [16]. Up to now, the best known approximation ratio is 1.488, due to Li [17] who uses a randomized selection in Byrka's algorithm [16]. Guha and Khuller [18] showed that this problem is hard to approximate within a factor better than 1.463, assuming $NP \nsubseteq DTIME[n^{O(\log \log n)}]$.

Capacitated facility location has also received a great deal of attention in recent years. Two main variants of the problem are soft-capacitated facility location and hard-capacitated facility location: in the latter problem, each facility is either opened at some location or not, whereas in the former, one may specify any integer number of facilities to be opened at that location. Soft capacities make the problem easier and by modifying approximation algorithms for the uncapacitated problems, we can also handle this case [14], [19]. Korupolu, Plaxton, and Rajaraman [20] gave the first constant-factor approximation algorithm that handles hard capacities, based on a local search procedure, but their approach works only if all capacities are equal. Chudak and Williamson [21] improved this performance guarantee to 5.83 for the same uniform capacity case. Pál, Tardos, and Wexler [22] gave the first constant performance guarantee for the case of non-uniform hard capacities. This was recently improved by Mahdian and Pál [23] and Zhang, Chen, and Ye [24] to yield a 5.83-approximation algorithm. All these approaches are based on local search. The only LP-relaxation based approach for this problem is due to Levi, Shmoys and Swamy [25] who gave a 5-approximation algorithm for the special case in which all facility opening costs are equal (otherwise the LP does not have a constant integrality gap). The above approximation algorithms for hard capacities are focused on the uniform demand case or the splittable case in which each unit of demand can be served by a different facility. Recently, Bateni and Hajiaghayi [26] considered the unsplittable hard-capacitated facility location problem when we allow violating facility capacities by a $1+\epsilon$ factor (otherwise, it is NP-hard to obtain any approximation factor) and obtain an $O(\log n)$ approximation algorithm for this problem.

A problem very close to both facility location and k-center is the k-*median* problem in which we want to open at most k facilities (like in the k-center problem) and the goal is to minimize the sum of connection costs of clients to their nearest open facilities (like facility location). If facilities have capacities the problem is called *capacitated k-median*. The approaches for uncapacitated facility location often work for k-median. In particular, Charikar, Guha, Tardos, and Shmoys [27] gave the first constant factor approximation for k-median based on LP rounding. The best approximation factor for k-median is $3 + \epsilon$, for an arbitrary positive constant ϵ, via the local search algorithm of Arya et al. [28]. Unfortunately obtaining a constant factor approximation algorithm for capacitated k-median still remains open despite consistent effort. The methods used to solve uncapacitated k-median or even the local search technique for capacitated facility location all seem to suffer from serious drawbacks when trying to apply them for capacitated k-median. For example standard LP relaxation is known to have an unbounded integrality gap [27]. The only previous attempts with constant approximation factors for this problem violate the capacities within a constant factor for the uniform capacity case [27] and the non-uniform capacity case [29] or exceed the number k of facilities by a constant factor [30].

CAPACITATED k-CENTER PROBLEM
Input: An undirected graph $G = (V, E)$, a capacity function $L : V \to \mathbb{N}$ and an integer k.
Output: A set $S \subseteq V$ of size k, and a function $\phi : V \to S$, such that for each $u \in S$, $|\phi^{-1}(u)| \leq L(u)$.
Goal: Minimize $\max_{v \in V} \text{dist}_G(v, \phi(v))$.

Removing the metric: We employ the standard "thresholding" method used for bottleneck optimization problems. We can assume that we guess the optimal solution, since there are polynomially many distinct distances between pairs of nodes. Once we guess the distance correctly, we create an unweighted graph consisting of those edges uv such that $d(u,v) \leq OPT$. We henceforth assume that we are considering the problem for an undirected graph G.

By a c-approximation algorithm we denote a polynomial time algorithm, that for an instance for which there exists a solution with objective function equal to 1, returns a solution using distances at most c. Note that the distance function $\mathrm{dist}(u,v)$, measures the distance in the unweighted undirected graph.

In the soft-capacitated version S can be a multiset, that is one can open more than one center at a vertex. To avoid confusion we call the standard version of the problem hard-capacitated.

B. Our results

While LP based algorithms have been widely used for uncapacitated facility location problems as well as capacitated versions of facility location with soft capacities, these methods are not of much use for problems in dealing with hard capacities due to the fact that they usually have an unbounded integrality gap [22], [27].

For general undirected graphs this is also the case for the capacitated k-center problem. Consider the LP relaxation for the natural IP, which we denote as LP1. We use y_u as an indicator variable for open centers.

$$\sum_{u \in V} y_u = k; \quad (1)$$
$$x_{u,v} \leq y_u \qquad \forall u,v \in V \quad (2)$$
$$\sum_{v \in V} x_{u,v} \leq L(u) y_u \qquad \forall u \in V \quad (3)$$
$$\sum_{u \in V} x_{u,v} = 1 \qquad \forall v \in V \quad (4)$$
$$0 \leq y_u \leq 1 \qquad \forall u \in V \quad (5)$$
$$x_{u,v} = 0 \qquad \forall u,v \in V \ \mathrm{dist}_G(u,v) > 1 \quad (6)$$
$$x_{u,v} \geq 0 \qquad \forall u,v \in V \quad (7)$$

For the sake of presentation we have introduced variables $x_{u,v}$ for all u, v, even if the distance between u and v in G is greater than one. We will use those variables in our rounding algorithm. Furthermore in constraints (1) and (4) we used equality instead of inequality to make our rounding algorithm and lemma formulations simpler. In the soft-capacitated version the $y_u \leq 1$ part of constraint (5) should be removed. Note that we are only interested in feasibility of LP1, and there is no objective function.

For an undirected graph $G = (V,E)$ and a positive integer δ, by G^δ we denote the graph (V,E'), where $uv \in E'$ iff $\mathrm{dist}_G(u,v) \leq \delta$. By an integrality gap of LP1 we mean the minimum positive integer δ such that if LP1 has a feasible solution, then the graph G^δ admits a capacitated k-center solution. As this is usually the case for capacitated problems, by a simple example we prove LP1 has unbounded integrality gap for general graphs. Due to space limitations, proofs of theorems marked with a spade symbol (♠) are postponed to the full version of this paper.

Theorem I.1 (♠). *LP1 has unbounded integrality gap, even for uniform capacities.*

However, interestingly, if we assume that the given graph is connected, the situation changes dramatically. Our main result is, that both for hard and soft capacitated version of the k-center problem, even for non-uniform capacities, LP1 has constant integrality gap for connected graphs. Moreover by using novel techniques we show a corresponding polynomial time rounding algorithm, which consists of several steps, described at high level in the following subsection. The actual algorithm is somewhat complex, although it can be implemented quite efficiently.

Theorem I.2. *There is a polynomial time algorithm, which given an instance of the hard-capacitated k-center problem for a connected graph, and a fractional feasible solution for LP1, can round it to an integral solution that uses non-zero $x_{u,v}$ variables for pairs of nodes with distance at most c.*

Corollary I.3. *The integrality gap of LP1 for connected graphs is bounded by a constant, and there is a constant factor approximation algorithm for connected graphs.*

To simplify the presentation we do not calculate the exact constant proved in the above corollary, but it is in the order of hundreds. As a counterposition, for soft capacities in the full version we present a much simpler 11-approximation algorithm, which we find as one more evidence that hard capacities are much harder to deal with.

Theorem I.4 (♠). *For connected graphs there is a polynomial time rounding algorithm, upper bounding the integrality gap of LP1 by 11 for soft-capacities.*

By using standard techniques one can restrict the capacitated k-center problem to connected graphs.

Theorem I.5 (♠). *If there exists a polynomial time c-approximation algorithm for the capacitated k-center problem in connected graphs, then there exists a polynomial time c-approximation algorithm for general graphs.*

Therefore we prove there is a constant factor approximation algorithm for the hard-capacitated k-center problem[1]. Our results easily extend to the case when there is an upper bound $U(u)$ of the number of times vertex u may be chosen as a center. Constraint 5 should be modified to be $0 \leq y_u \leq U(u)$ to yield a relaxation LP2. We can employ the same rounding procedure as discussed for the hard capacity case with $U(u) = 1$.

The proof of the following theorem is omitted.

Theorem I.6 (♠). *There is a polynomial time algorithm, which given an instance of the hard-capacitated k-center problem for a connected graph, and a fractional feasible solution for LP2, can round it to an integral solution that*

[1] With some care, perhaps some of the constants can be improved, however our focus was to show that a constant approximation is obtainable using LP rounding.

uses non-zero $x_{u,v}$ variables for pairs of nodes with distance at most c.

While our constants are large, we do show via integrality gap examples that the problem with non-uniform capacities is significantly harder than the basic k-center problem.

Theorem I.7 (♠). *For connected graphs the integrality gap of LP1 is at least 5 for uniform-hard-capacities and at least 4 for uniform-soft-capacities.*

Moreover in the non-uniform hard-capacitated case, the integrality gap of LP1 for connected graphs is at least 7, even if all the non-zero capacities are equal.

Despite the fact, that the algorithm of [7] for uniform capacities was obtained more than a decade ago, no lower bound for the capacity version (neither soft nor hard), better than the trivial $2-\epsilon$, derived from the uncapacitated version, is known. We believe that the integrality gap examples, presented in this paper, are of independent interest since they may help in proving a stronger lower bound for the capacitated k-center problem with uniform capacities.

To make a step in this direction we investigate lower bounds for the non-uniform case. By a reduction from the cost k-center problem [9] one can show that there is no $(3-\epsilon)$-approximation for the capacitated k-center problem with non-uniform capacities. By a simple reduction from Exact Cover by 3-Sets, in the full version, we prove the same result under the assumption $P \neq NP$.

Finally we give evidence that our LP approach might be the proper tool for solving the capacitated k-center problem. The proof of the following theorem shows that when the Khuller-Sussmann algorithm fails to find a solution then in fact there is no feasible LP solution for that guess of distance. The smallest radius guess for which the algorithm succeeds, proves an integrality gap on the LP. Considering the result of Theorem I.7, it follows that for uniform capacities the gap in the analysis is small, since our bounds are tight up to an additive $+1$ error.

Theorem I.8 (♠). *For connected graphs the integrality gap of LP1 is at most 6 for uniform-hard-capacities and at most 5 for uniform-soft-capacities.*

C. Our techniques

We assume that G is connected and that LP1 has a feasible solution for the graph G. We call two functions $x : V \times V \to \mathbb{R}_+ \cup \{0\}$ and $y : V \to \mathbb{R}_+ \cup \{0\}$ an *assignment* even if (x, y) is potentially infeasible for LP1. In other words initially we have a feasible fractional solution, in the end we will obtain a feasible integral solution, although during the execution of our rounding algorithm an assignment (x, y) is not required to be feasible. Furthermore without loss of generality we assume that for a vertex v with $L(v) = 0$ we have $y_v = 0$.

We need to show that there exists a constant δ such that if for a connected component LP1 has a feasible solution, then one can (in polynomial time) find an integral feasible solution for G^δ.

Definition I.9 (δ-feasible solution). An assignment is called *δ-feasible* if it is feasible for the graph G^δ.

Note that the only difference between LP1's for the graphs G and G^δ is constraint (6).

Definition I.10 (radius$_{(x,y)}$). For a δ-feasible solution (x, y) to LP1 we define a function $\mathrm{radius}_{(x,y)} : V \to \{0, \ldots, \delta\}$ which for a vertex u assigns the greatest integer i such that there exists a vertex v with $\mathrm{dist}_G(v, u) = i$ and $x_{u,v} > 0$ (if no such i exists then $\mathrm{radius}_{(x,y)}(u) = 0$).

We give a brief overview of the following sections. Initially we start with a 1-feasible (fractional) solution (x, y) to LP1 and our goal is to make it integral. We perform several steps where in each step we get more structure on the δ-feasible solution but at the same time the value of δ will increase.

In Sections II-A-II-D in four non-trivial steps we round the y-values of a feasible solution. First, in Section II-A we define a *caterpillar structure* which is a key structure in the rounding process. In Section II-B we define the *y-flow* and *chain shifting* operations which allow for transferring y-values between distant vertices using intermediate vertices on the caterpillar structure. Unfortunately, because the capacities are non-uniform and hard, to find a rounding flow for a caterpillar structure we need more assumptions. To overcome this difficulty in the most challenging part of the rounding process, that is in Section II-C, we define a *safe* caterpillar structure and show how to split a given caterpillar structure into a set of safe caterpillar structures (at the cost of increasing radius of the δ-feasible solution). In Section II-D we design a rounding procedure for a safe caterpillar structure, obtaining a c-feasible solution with integral y-values, for some constant c. We would like to note, that for uniform capacities every caterpillar structure is safe, therefore for non-uniform capacities we have to design much more involved tools comparing to the previously known uniform capacities case.

Finally in Section II-E we show, that using standard techniques, when we have integral y-values then rounding x-values is simple, obtaining a constant factor approximation algorithm.

II. LP ROUNDING FOR HARD-CAPACITIES

A. Group shifting and caterpillar structure

In the first phase of our procedure we obtain a path-like structure containing all vertices with non-integral y-values. We first define the notion of shifting values between variables of LP1 relaxation.

Definition II.1 (**shifting).** For an assignment (x, y) for the LP, two distinct vertices $a, b \in V$ and a positive real $\alpha \leq \min(y_a, 1 - y_b)$ such that $L(a) \leq L(b)$ by *shifting* α from a to b we consider the following operation:

1) Let $\epsilon = \frac{\alpha}{y_a}$; for each $v \in V$ let $\Delta_v = \epsilon x_{a,v}$, decrease $x_{a,v}$ by Δ_v and increase $x_{b,v}$ by Δ_v.
2) Increase y_b by α, and decrease y_a by α.

Lemma II.2 (♠). *Let (x, y) be a δ-feasible solution to LP. Let (x', y') be a result of shifting α from a to b, for some α, a, b such that $L(a) \leq L(b)$, $0 < \alpha \leq \min(y_a, 1 - y_b)$. Then (x', y') is a $(\delta + \mathrm{dist}_G(a, b))$-feasible solution and for each vertex $v \neq b$ we have $\mathrm{radius}_{(x',y')}(v) \leq \mathrm{radius}_{(x,y)}(v)$ whereas $\mathrm{radius}_{(x',y')}(b) \leq \max(\mathrm{radius}_{(x,y)}(a) + \mathrm{dist}_G(a, b), \mathrm{radius}_{(x,y)}(b))$.*

Definition II.3 (**group shifting).** For a δ-feasible solution (x, y) and a set $V_0 \subseteq V$ by a *group shifting* we denote the following operation. Assume $V_0 = \{v_1, \ldots, v_\ell\}$, where $L(v_i) \leq L(v_{i+1})$ for $1 \leq i < \ell$. As long as there are at least two vertices in V_0 with fractional y-values, let a be the smallest, and b the greatest integer such that $v_a, v_b \in V_0$ are vertices with fractional y-values. Shift $\min(y_a, 1 - y_b)$ from a to b.

Lemma II.4. *Let (x, y) be a δ-feasible solution, V_0 be a subset of V and $d = \max_{a,b \in V_0} \mathrm{dist}_G(a, b)$. After group shifting on V_0 we obtain a $(\delta + d)$-feasible solution (x', y'), where there is at most one vertex in V_0 with fractional y-value and moreover for $v \in V \setminus V_0$ we have $\mathrm{radius}_{(x',y')}(v) \leq \mathrm{radius}_{(x,y)}(v)$.*

To make a graph Hamiltonian we use the following lemma known from 1960 [31], [32].

Lemma II.5. *For any undirected connected graph G there always exists a Hamiltonian path in G^3 and one can find it in polynomial time.*

We define a caterpillar structure which is one of the key ingredients of our rounding process. Intuitively we want to define an auxiliary path-like tree, where adjacent vertices are close in the original graph G, vertices with fractional y-values are leaves of the tree, and all non-leaf vertices have y-values equal to 1.

Definition II.6 (**caterpillar structure).** By a δ-caterpillar structure for an assignment (x, y) we denote a sequence of distinct vertices $P = (v_1, \ldots, v_p)$ together with a sequence $P' = (v'_0, \ldots, v'_{p+1})$ where:

1) for each $i = 1, \ldots, p$ we have $y_{v_i} = 1$,
2) for each $i = 1, \ldots, p-1$ we have $\mathrm{dist}_G(v_i, v_{i+1}) \leq \delta$,
3) for each $i = 0, \ldots, p+1$ either $v'_i = \mathrm{nil}$ or $v'_i \in V \setminus \{v_j : j = 1, \ldots, p\}$,
4) for each $i = 1, \ldots, p$ if $v'_i \neq \mathrm{nil}$ then $L(v_i) \geq L(v'_i)$, $0 < y_{v'_i} < 1$, $\mathrm{dist}_G(v_i, v'_i) \leq \delta$,
5) if $v'_0 \neq \mathrm{nil}$ then $\mathrm{dist}_G(v'_0, v_1) \leq \delta$, $0 < y_{v'_0} < 1$,
6) if $v'_{p+1} \neq \mathrm{nil}$ then $\mathrm{dist}_G(v'_{p+1}, v_p) \leq \delta$, $0 < y_{v'_{p+1}} < 1$,
7) for each $0 \leq i < j \leq p+1$ if $v'_i \neq \mathrm{nil}$ and $v'_j \neq \mathrm{nil}$ then $v'_i \neq v'_j$,
8) $\sum_{v \in V(P')} y_v$ is integral.

We sometimes omit δ and simply write "caterpillar structure" when the value of δ is irrelevant.

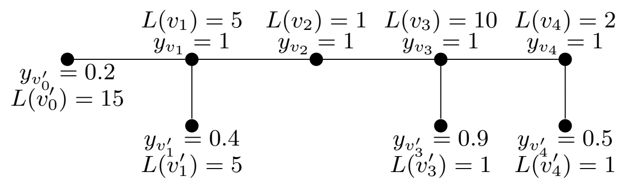

Figure 1. Example of a δ-caterpillar structure $((v_1, v_2, v_3, v_4), (v'_0, v'_1, \mathrm{nil}, v'_3, v'_4, \mathrm{nil}))$. Vertices connected by edges are within distance δ in the graph G. Note that the sum of y-values over all vertices is integral.

Lemma II.7. *For a given feasible LP solution (x, y) we can find a 5-feasible solution (x', y') together with a 21-caterpillar structure (P, P') such that each vertex $v \in V \setminus (V(P) \cup V(P'))$ has an integral y-value in (x', y'), and the first and last element of the sequence P' equals* nil.

Proof: Consider the following algorithm for constructing sets S, S' and a function $\Phi : V \to S'$. The set S will be an inclusionwise maximal independent set in G^2 and moreover we ensure that $L(\Phi(v)) \geq L(v)$, for any $v \in V$.

1) Set $V_0 := V$ and $S := S' := \emptyset$.
2) As long as $V_0 \neq \emptyset$ let v be a highest capacity vertex in V_0.
   - Let $f(v)$ be a highest capacity vertex in $N_G[v]$ (potentially $f(v) \notin V_0$).
   - Add $f(v)$ to S' and for each $u \in N_G[N_G[v]] \cap V_0$ set $\Phi(u) = f(v)$.
   - Add v to S and set $V_0 := V_0 \setminus N_G[N_G[v]]$.

Observe that each time we remove from the set V_0 all vertices that are within distance two from v, hence the set S is an inclusion maximal independent set in G^2. For this reason vertices in the set S have disjoint neighborhoods and moreover by constraints (4) and (2) of the LP1 we infer that for each $v \in V$ we have:

$$\sum_{u \in N[v]} y_u \geq \sum_{u \in N[v]} x_{u,v} = 1 \tag{8}$$

We perform shifting operations to make sure all vertices in the set S' have y-value equal to one. Consider a vertex $v \in S$ and the corresponding vertex $f(v)$ chosen by the algorithm. As long as $y_{f(v)} < 1$ take any $u \in N[v], u \neq f(v)$ such that $y_u > 0$ and shift $\min(y_u, 1 - y_{f(v)})$ from u to $f(v)$. Note that $L(u) \leq L(f(v))$ by the definition of $f(v)$ and for this reason shifting is possible. By Lemma II.2 after all the shifting operations we have a 3-feasible solution (x, y), since before a shift from u to $f(v)$ we have $\mathrm{radius}_{(x,y)}(u) \leq 1$, $\mathrm{radius}_{(x,y)}(f(v)) \leq 3$ and $\mathrm{dist}_G(u, f(v)) \leq 2$. Moreover by Inequality (8) we infer, that all the vertices in the set S' have y-value equal to one, since otherwise a shifting operation from some $u \in N[v]$ to $f(v)$ would be possible.

Observe that by the maximality of the independent set S in G^2 the graph $G^5[S]$ is connected, otherwise we could add a vertex to S still obtaining an independent set in G^2. Moreover for any two adjacent vertices $u, v \in S$ in $G^5[S]$, the vertices $f(u), f(v)$ are adjacent in $G^7[S']$. By the connectivity of $G^5[S]$, the graph $G^7[S']$ is also connected. By Lemma II.5 we can in polynomial time order the vertices of S' to obtain a Hamiltonian path P in $G^{21}[S']$.

Currently for each vertex v from the set $V \setminus S'$ we have $\mathrm{radius}_{(x,y)}(v) \leq 1$. For each $v \in S$ we use group shifting on the set $\Phi^{-1}(f(v)) \setminus S'$. Since

$$\max_{a,b\in\Phi^{-1}(f(v))\setminus S'} \mathrm{dist}_G(a, b) \leq \max_{a,b\in\Phi^{-1}(f(v))\setminus S'} \mathrm{dist}_G(a, v) + \mathrm{dist}_G(v, b) \leq 4\,,$$

by Lemma II.4 we obtain a 5-feasible solution (x, y) such that all vertices in the set S' have y-value equal to one and moreover for each $f(v) \in S'$ the set $\Phi^{-1}(f(v)) \setminus S'$ contains at most one vertex with fractional y-value. Let us assume that the already constructed path P is of the form $P = (v_1, \dots, v_p)$. We construct a sequence $P' = (\mathrm{nil}, v'_1, \dots, v'_p, \mathrm{nil})$ where as v'_i we take the only vertex from $\Phi^{-1}(v_i) \setminus S'$ that has fractional y-value, or we set $v'_i := \mathrm{nil}$ if $\Phi^{-1}(v_i) \setminus S'$ has no vertices with fractional y-value. Note that since the way we select vertices to the sets S, S' is capacity driven (recall as v we select the highest capacity vertex in V_0 and as $f(v)$ we select a highest capacity vertex in $N[v]$), for each vertex $u \in \Phi^{-1}(v_i)$ we have $L(u) \leq L(v_i)$. In this way we have constructed a 5-feasible solution (x, y) together with a desired 21-caterpillar structure (P, P'). ■

As the reader might notice in the above proof we always construct a caterpillar structure with $v'_0 = v'_{p+1} = \mathrm{nil}$. The reason why the definition of a caterpillar structure allows for v'_0 and v'_{p+1} have non-nil values is that in Section II-C we will split a caterpillar structure into two smaller pieces and in order to have those pieces satisfy Definition II.6 we need v'_0 and v'_{p+1}.

B. y-flow and chain shifting

In the previous section we defined a group shifting operation. Unfortunately we can only perform such an operation if vertices are close. In this section we define notions of *y-flow* and *chain shifting* which allow us to transfer y-value between distant vertices. We will use those tools in Sections II-C and II-D.

Definition II.8 (y-flow). For a given assignment (x, y) let $S \subseteq V$ and $T \subseteq V$ be two disjoint sets and let $\mathcal{F}$ be a set containing sequences of the form $(\alpha, v_1, \dots, v_t)$ representing paths, where α is a positive real, each $v_i \in V$ is a vertex (for $i = 1, \dots, t$), $v_1 \in S$, $v_t \in T$, $L(v_1) \leq L(v_t)$ and for $i = 2, \dots, t-1$ we have $v_i \notin S \cup T, y_{v_i} = 1, L(v_i) \geq L(v_1)$. We call $(\alpha, v_1, \dots, v_t)$ a *path* transferring α from v_1 to v_t through $v_2, \dots, v_{t-1}$. We denote $v_2, \dots, v_{t-1}$ as *internal* vertices of the path $(\alpha, v_1, \dots, v_t)$.

The set $\mathcal{F}$ is a *y-flow* from S to T iff:

- for each $v \in S$ the sum of values transferred from v in $\mathcal{F}$ is at most y_v,
- for each $v \in T$ the sum of values transferred to v in $\mathcal{F}$ is at most $1 - y_v$,
- for each $v \in V \setminus (S \cup T)$ the sum of values transferred through v in $\mathcal{F}$ is at most 1.

For a given y-flow $\mathcal{F}$ from S to T we define $G_{\mathcal{F}} = (V, A)$ as an auxiliary directed graph with the same vertex set as G, where an arc (u, v) belongs to A iff there is a path in $\mathcal{F}$ containing u and v as consecutive vertices in exactly this order. We call the y-flow $\mathcal{F}$ *acyclic* iff the directed flow graph $G_{\mathcal{F}}$ is acyclic. Furthermore we define a function $f_{\mathcal{F}} : A \to (0, 1]$, which for an arc (u, v) assigns the sum of α values in all the paths in $\mathcal{F}$ that contain u and v as consecutive vertices. Moreover by $fl_{\mathcal{F}} : A \to \mathbb{R}_+$ we denote a function, which for an arc (u, v) assigns the sum of terms $L(s)\alpha$ over all paths from $\mathcal{F}$ that start with α and $s \in S$ and contain u, v as consecutive elements. Intuitively by $f_{\mathcal{F}}((u, v))$ we denote the fractional number of centers that are transferred from u to v, whereas by $fl_{\mathcal{F}}((u, v))$ we denote the fractional number of vertices (clients) that were previously covered by u and will be covered by v after the shifting operation (see Fig. 2).

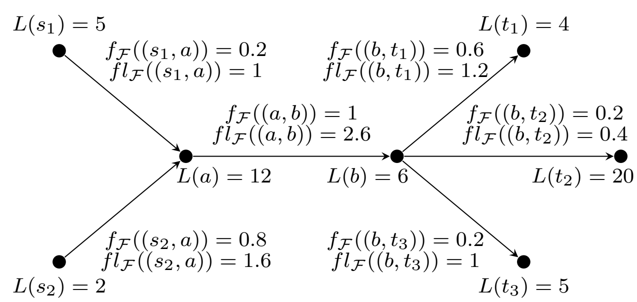

Figure 2. The graph $G_{\mathcal{F}}$ for an acyclic y-flow $\mathcal{F} = \{(0.2, s_1, a, b, t_3), (0.6, s_2, a, b, t_1), (0.2, s_2, a, b, t_2)\}$ from $S = \{s_1, s_2\}$ to $T = \{t_1, t_2, t_3\}$, where $y_{s_1} = 0.4$, $y_{s_2} = y_a = y_b = 1$, $y_{t_1} = 0$, $y_{t_2} = 0.8$, $y_{t_3} = 0.1$. Note that even though each path in $\mathcal{F}$ has starting point capacity not greater than its ending point capacity the vertex $t_1 \in T$ is reachable from $s_1 \in S$ in $G_{\mathcal{F}}$ despite the fact that $L(s_1) > L(t_1)$.

Now we show that if we are given an acyclic y-flow $\mathcal{F}$ then we can transfer y-values using a *chain shifting* method without increasing the radius of vertices by too much. Formal definitions and lemmas follow.

Definition II.9 (chain shifting). Let $\mathcal{F}$ be an acyclic y-

flow from S to T and let (x, y) be a δ-feasible solution. Let $G_{\mathcal{F}} = (V, A)$ be the auxiliary acyclic flow graph.

By *chain shifting* we denote the following operation:

- For each $u, v \in V$, set $\Delta_{u,v} = 0$.
- For each arc $(u, a) \in A$ in reverse topological ordering of $G_{\mathcal{F}}$:
  - For each $v \in V$, let $\Delta = x_{u,v} fl_{\mathcal{F}}(u, a)/(L(u)y_u)$, set $\Delta_{a,v} = \Delta_{a,v} + \Delta$ and $\Delta_{u,v} = \Delta_{u,v} - \Delta$.
- For each $u, v \in V$, set $x_{u,v} = x_{u,v} + \Delta_{u,v}$.
- For each $s \in S$ decrease y_s by $\sum_{(s,u)\in A} f_{\mathcal{F}}((s, u))$.
- For each $t \in T$ increase y_t by $\sum_{(u,t)\in A} f_{\mathcal{F}}((u, t))$.

For a directed graph $G = (V, A)$, for a vertex v, we denote $N^{in}(v) = \{u : (u, v) \in A\}$ and $N^{out}(v) = \{u : (v, u) \in A\}$.

Lemma II.10 (♠). *Let (x', y') be the result of the chain shifting operation on a δ-feasible solution (x, y) according to an acyclic y-flow $\mathcal{F}$ from S to T. If d is the greatest distance in G between two adjacent vertices in $G_{\mathcal{F}}$, then (x', y') is a $(\delta + d)$-feasible solution, and for each vertex v of indegree zero in $G_{\mathcal{F}}$, we have $\text{radius}_{(x',y')}(v) \leq \text{radius}_{(x,y)}(v)$, whereas for other vertices v, we have*

$$\text{radius}_{(x',y')}(v) \leq \max(\text{radius}_{(x,y)}(v), \max_{a\in N^{in}_{G_{\mathcal{F}}}(v)}(\text{radius}_{(x,y)}(a) + \text{dist}_G(a, v))) .$$

Furthermore for each $v \in V \setminus (S \cup T)$ its y-value is the same in (x, y) and (x', y').

C. Separable caterpillar structure

If we knew that in the caterpillar structure (P, P') produced by Lemma II.7 the capacity of each vertex in P is not smaller than the capacity of each vertex in P' then we could skip this section. Unfortunately some vertices of $V(P)$ may have smaller capacity than some vertices of $V(P')$ and for this reason we define the notion of *dangerous*, *safe* and *separable* caterpillar structures.

Definition II.11 (safe, dangerous). For a caterpillar structure $\mathcal{P} = (P = (v_1, \ldots, v_p), (v'_0, \ldots, v'_{p+1}))$, by $\Gamma(\mathcal{P}) \subseteq V(P)$ we denote the set containing all vertices v_i, such that there exist $0 \leq i_0 < i < i_1 \leq p + 1$, such that $v'_{i_0} \neq \text{nil}$, $L(v'_{i_0}) > L(v_i)$ and $v'_{i_1} \neq \text{nil}$, $L(v'_{i_1}) > L(v_i)$.

A caterpillar structure $\mathcal{P}$ is *safe* if $\Gamma(\mathcal{P}) = \emptyset$ and *dangerous* otherwise.

Definition II.12 (separable). Let (x, y) be a δ-feasible solution and let $\mathcal{P} = (P = (v_1, \ldots, v_p), P' = (v'_0, \ldots, v'_{p+1}))$ be a dangerous caterpillar structure. We call $\mathcal{P}$ *separable* iff there exists $1 \leq i \leq p$ such that $v_i \in \Gamma(\mathcal{P})$, $L(v_i) = \min_{v\in\Gamma(\mathcal{P})} L(v)$ and either:

- $S_1 \geq \lceil S_2 \rceil - S_2$, where $S_2 = \sum_{\substack{j=i+1,\ldots,p+1 \\ v'_j \neq \text{nil}}} y_{v'_j}$ and S_1 is the sum of values $(1 - y_v)$ where $v \in V, v = v'_j, L(v) > L(v_i)$ for some $i < j \leq p + 1$, or,

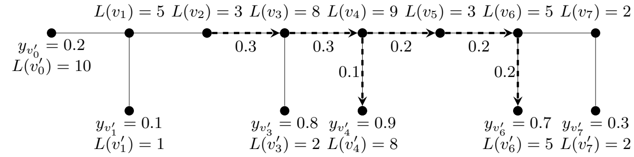

Figure 3. A separable caterpillar structure (P, P'), where $\Gamma((P, P')) = \{v_1, v_2, v_5\}$ (note that $v_7 \notin \Gamma((P, P'))$, since $v'_8 = \text{nil}$). By dashed edges an acyclic flow $\mathcal{F} = \{(0.1, v_2, v_3, v_4, v'_4), (0.2, v_2, v_3, v_4, v_5, v_6, v'_6)\}$ from $\{v_2\}$ to $\{v'_4, v'_6\}$ is marked with values $f_{\mathcal{F}}$ printed in the middle of each arc.

- $S_1 \geq \lceil S_2 \rceil - S_2$, where $S_2 = \sum_{\substack{j=0,\ldots,i-1 \\ v'_j \neq \text{nil}}} y_{v'_j}$ and S_1 is the sum of values $(1-y_v)$ where $v \in V, v = v'_j, L(v) > L(v_i)$ for some $0 \leq j < i$.

We call such i as above a *witness of separability* of $\mathcal{P}$. A caterpillar structure that is not separable is called *non-separable*.

The intuition behind the sums S_1, S_2 is as follows. The sum S_2 contains all the y-values of vertices of P' to the right (or left) of i. Since we want to round all the y-values of vertices of P', if we want to split the caterpillar structure (P, P') by removing the edge $v_i v_{i+1}$ (or $v_{i-1} v_i$), we need to send $\lceil S_2 \rceil - S_2$ units of flow to the part that does not contain v_i, in order to make the sum of y-values over all the leaves in both new caterpillar structures integral. That is to satisfy (8) of Definition II.6. In S_1 we sum over all vertices, that can potentially receive flow if we start a path at v_i, and the value $(1 - y_v)$ is the y-value a vertex v may receive.

An example of a separable caterpillar structure is depicted in Fig. 3. Observe that a non-separable path structure may be dangerous as in Fig. 4.

Lemma II.13 (♠). *Let $\mathcal{P} = ((v_1, \ldots, v_p), (v'_0, \ldots, v'_{p+1}))$ be a dangerous caterpillar structure and let i be an index such that $v_i \in \Gamma(\mathcal{P})$ and $L(v_i) = \min_{v\in\Gamma(\mathcal{P})} L(v)$. Moreover let j be an index such that $v'_j \neq \text{nil}, L(v'_j) > L(v_i)$. Then for any $a \in [\min(i, j), \max(i, j)]$ we have $L(v_a) \geq L(v_i)$.*

Lemma II.14. *Let $\mathcal{P} = ((v_1, \ldots, v_p), (v_0, \ldots, v_{p+1}))$ be a dangerous non-separable caterpillar structure and let $\ell = \min_{v\in\Gamma(\mathcal{P})} L(v)$. For $I = \{i : 0 \leq i \leq p + 1 \wedge v'_i \neq \text{nil} \wedge L(v'_i) > \ell\}$ we have $\sum_{i\in I}(1 - y_{v'_i}) < 2$.*

Proof: Consider any $v_i \in \Gamma(\mathcal{P})$ such that $L(v_i) = \ell$. Let $I_1 = I \cap [0, i - 1]$ and $I_2 = I \cap [i + 1, p + 1]$ (note that $I = I_1 \cup I_2$). We know that v_i is not a witness of separability hence each of the two sums S_1 in Definition II.12 is strictly smaller than 1, since otherwise we would have $S_1 \geq 1 \geq \lceil S_2 \rceil - S_2$. Consequently $\sum_{i\in I_1}(1 - y_{v'_i}) < 1$ and similarly $\sum_{i\in I_2}(1 - y_{v'_i}) < 1$. ■

In the following lemma we use a procedure which given a δ-caterpillar structure (P, P') produces a set of non-separable δ-caterpillar structures. At very high level it checks

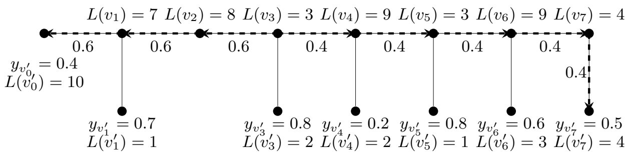

Figure 4. A dangerous caterpillar structure (P, P'), where $\Gamma((P, P')) = \{v_3, v_5\}$. The caterpillar structure is non-separable because both for $i = 3$ and $i = 5$ in Definition II.12 the sum S_1 is at most 0.6, while $\lceil S_2 \rceil - S_2$ is equal to 0.9. By dashed edges an acyclic flow $\mathcal{F} = \{(0.6, v_3, v_2, v_1, v'_0), (0.4, v_3, v_4, v_5, v_6, v_7, v'_7)\}$ from $\{v_3\}$ to $\{v'_0, v'_7\}$ is marked with values $f_\mathcal{F}$ printed in the middle of each arc.

whether (P, P') is separable, and if yes it sets as i a witness from Definition II.12 with the smallest value of $L(v_i)$. Next an acyclic flow from v_i to leaves of (P, P') is constructed (see Fig. 3), and afterwards the procedure is run on two caterpillar structures induced by the parts to the left, and to the right of v_i.

Lemma II.15 (♠). *For a given feasible LP solution (x, y) we can find a 68-feasible solution (x', y') together with a set of vertex disjoint non-separable 21-caterpillar structures $\mathcal{S}$ such that each vertex v outside of the set has an integral y-value in (x', y'). Furthermore for each vertex v that belongs to some caterpillar structure from $\mathcal{S}$ we have* $\mathrm{radius}_{(x',y')}(v) \leq 47$.

In the following lemma we transform non-separable caterpillar structures into safe caterpillar structures.

Lemma II.16. *There exist constants c, δ such that for a given feasible LP solution (x, y) we can find a c-feasible solution (x', y') together with a set of vertex disjoint safe δ-caterpillar structures $\mathcal{S}$ such that each vertex v outside of the set has an integral y-value in (x', y').*

Proof: We use Lemma II.15 to obtain a set $\mathcal{S}$ of vertex disjoint non-separable 21-caterpillar structures. Our goal is to transform each dangerous caterpillar structure in $\mathcal{S}$ into a safe caterpillar structure.

Consider a dangerous non-separable δ_0-caterpillar structure $\mathcal{P} = ((v_1, \ldots, v_p), (v'_0, \ldots, v'_{p+1})) \in \mathcal{S}$ and let v_a be a minimum capacity vertex in $\Gamma(\mathcal{P})$. Moreover let $I = \{i : 0 \leq i \leq p + 1 \wedge v'_i \neq \mathrm{nil} \wedge L(v'_i) > L(v_a)\}$. Construct any acyclic y-flow which sends $\min(1, \sum_{i \in I}(1 - y_{v'_i}))$ from $\{v_a\}$ to $\{v'_i : i \in I\}$ (see Fig. 4). Such flow always exists due to Lemma II.13.

Let $Y = \{v_a, v'_a, v'_{a-1}\} \setminus \{\mathrm{nil}\}$ and perform group shifting on Y (note that $a \geq 1$, since $v_a \in \Gamma(\mathcal{P})$). Replace $\mathcal{P}$ in $\mathcal{S}$ with the $(2\delta_0)$-caterpillar structure $((v_1, \ldots, v_{a-1}, v_{a+1}, \ldots, v_p), (v'_0, \ldots, v'_{a-2}, u, v'_{a+1}, \ldots, v'_{p+1}))$, where as u we set the only vertex from Y with fractional y-value after group shifting or we set $u = \mathrm{nil}$ if all vertices in Y have integral y-values. We need to argue, that when $u \neq \mathrm{nil}$, we have $L(u) \leq L(v_{a-1})$, in order to satisfy (4) of Definition II.6. Observe, that if $v'_a \neq \mathrm{nil}$, then $L(v'_a) \leq L(v_a)$, and similarly if $v'_{a-1} \neq \mathrm{nil}$, then $L(v'_{a-1}) \leq L(v_{a-1})$. Hence to show $L(u) \leq L(v_{a-1})$ it is enough to show $L(v_a) \leq L(v_{a-1})$, but this follows from Lemma II.13, since $v_a \in \Gamma(\mathcal{P})$.

Note, that each caterpillar structure will be modified according to the above procedure at most twice, since after one iteration the sum $\sum_{i \in I}(1 - y_{v'_i})$ either equals zero or decreases by one, and by Lemma II.14 we have $\sum_{i \in I}(1 - y_{v'_i}) < 2$. Consequently by Lemmas II.10, II.4 we obtain the desired set of vertex disjoint δ-caterpillar structure together with a c-feasible solution. ■

D. Rounding safe caterpillar structures

In this section we describe how to round the c-feasible solution (x', y') using the set of vertex disjoint safe caterpillar structures $\mathcal{S}$ from Lemma II.16. In order to do that we introduce a notion of *rounding flow* which is a special kind of y-flow defined for a caterpillar structure.

Definition II.17 (rounding flow). For a caterpillar structure (P, P') and an assignment (x, y) we call $\mathcal{F}$ a *rounding flow* iff $\mathcal{F}$ is a y-flow from S to T where $S \cup T = V(P')$, for each $v'_i \in S$ we have $f_\mathcal{F}((v'_i, v_i)) = y_{v'_i}$ and for each $v'_i \in T$ we have $f_\mathcal{F}((v_i, v'_i)) = 1 - y_{v'_i}$. Furthermore each flow path from $\mathcal{F}$ can not go through a vertex from $V \setminus (V(P) \cup V(P'))$.

In order to obtain a rounding flow for each vertex of $V(P')$ (which by definition have fractional y-values), we have to decide whether it will be a source (member of S) or a sink (member of T). After chain shifting according to $\mathcal{F}$ all sources should have y-value equal to zero whereas all sinks should have y-value equal to one and consequently all vertices from the caterpillar structure will have integral y-value. In the following lemma we show that for each non-separable caterpillar structure we can always find a rounding flow in polynomial time.

Lemma II.18. *For any safe δ-caterpillar structure (P, P') and an assignment (x, y) there exists a rounding flow $\mathcal{F}$ such that for any two adjacent vertices in $G_\mathcal{F}$ their distance in G is at most δ. Furthermore we can find such a rounding flow in polynomial time.*

Proof: We present a recursive procedure which constructs a desired rounding flow. Note that some recursive calls of the procedure might potentially involve infeasible assignments (x', y'), however we prove that if the initial call gives the procedure a safe δ-caterpillar structure, then as a result we obtain a valid rounding flow.

Let us describe a procedure which is given a caterpillar structure (P, P') together with an assignment y (the procedure does not need the x part of an assignment). Denote $P = (v_1, \ldots, v_p)$ and $P' = (v'_0, \ldots, v'_{p+1})$. If $V(P') = \emptyset$ then we simply return the empty rounding flow. Otherwise let i be the smallest integer such that the sum of y-values of $X = \{v'_0, \ldots, v'_i\} \setminus \mathrm{nil}$ is at least one (such i always exists since the sum of all y-values in $V(P')$ is integral by (8) of Def. II.6). Note that since all vertices in $V(P')$ have fractional y-values we have $i > 0$. Let $0 \leq i_0 \leq i$ be an index such that $v'_{i_0} \neq \text{nil}$ and v'_{i_0} has the biggest capacity in X. Let $\alpha = \sum_{v \in X} y_v$. If $\alpha = 1$ then we recursively construct a rounding flow $\mathcal{F}$ from S to T for a smaller caterpillar structure $((v_{i+1}, \ldots, v_p), (\text{nil}, v'_{i+1}, \ldots, v'_{p+1}))$ and (i) add to S the set of vertices $X \setminus \{v'_{i_0}\}$ (ii) add to T the vertex v'_{i_0} (iii) for each $v'_j \in X \setminus \{v'_{i_0}\}$ add to $\mathcal{F}$ a flow path $(y_{v'_j}, v'_j, v_j, \ldots, v_{i_0}, v'_{i_0})$. In this case we return $\mathcal{F}$ as a desired rounding flow for (P, P'). Hence from now on we assume $\alpha > 1$ and $\alpha - 1 < y_{v'_i}$. Consider two cases: $i_0 < i$ and $i_0 = i$.

First let us assume that $i_0 < i$. We store $z := y_{v'_i}$ and temporarily set $y_{v'_i} = \alpha - 1$. Next recursively construct a rounding flow $\mathcal{F}$ from $S \subseteq V(P'')$ to $T \subseteq V(P'')$ for a smaller caterpillar structure $((v_i, \ldots, v_p), P'')$, where $P'' = (\text{nil}, v'_i, \ldots, v'_{p+1})$ (note that the sum of y-values in P'' is integral). Now consider two cases:

- if $v'_i \in S$ then: (i) add to S vertices from $X \setminus \{v'_i, v'_{i_0}\}$ (ii) add to T the vertex v'_{i_0} (iii) for each $v'_j \in X \setminus \{v'_{i_0}, v'_i\}$ add to $\mathcal{F}$ a flow path $(y_{v'_j}, v'_j, v_j, \ldots, v_{i_0}, v'_{i_0})$ (iv) add to $\mathcal{F}$ a flow path $(z - y_{v'_i}, v'_i, v_i, \ldots, v_{i_0}, v'_{i_0})$ (v) set $y_{v'_i} := z$ (vi) return $\mathcal{F}$.
- if $v'_i \in T$ then: (i) add to S vertices from $X \setminus \{v'_i, v'_{i_0}\}$ (ii) add to T the vertex v'_{i_0} (iii) out of the flow paths in $\mathcal{F}$ that end in v'_i leave only that many, that send exactly $1 - z$ units of flow and reroute the rest paths to v'_{i_0} through vertices $v_{i-1}, v_{i-2}, \ldots, v_{i_0}$ (iv) for each $v'_j \in X \setminus \{v'_{i_0}, v'_i\}$ add to $\mathcal{F}$ a flow path $(y_{v'_j}, v'_j, v_j, \ldots, v_{i_0}, v'_{i_0})$ (v) return $\mathcal{F}$.

Now assume that $i_0 = i$. We create a smaller caterpillar structure $((v_a, v_{i+1}, v_{i+2}, \ldots, v_p), (\text{nil}, v'_a, v'_{i+1}, \ldots, v'_{p+1}))$, where v_a, v'_a are two newly created vertices with $y_{v'_a} := \alpha - 1$ and $L(v'_a) := L(v_a) := L(v'_{i_1})$, where v'_{i_1} is the second biggest capacity vertex in the set X. Next run recursively our procedure on the newly created caterpillar structure to obtain a rounding flow $\mathcal{F}$ from S to T. Again, consider two cases:

- if $v'_a \in S$ then: (i) set $S := (S \setminus \{v'_a\}) \cup (X \setminus \{v'_i\})$ (ii) set $T := T \cup \{v'_i\}$ (iii) change in $\mathcal{F}$ all the paths that start in v'_a to start in $X \setminus \{v'_i\}$ (iv) add to $\mathcal{F}$ paths that start in X and transfer $1 - y_{v'_i}$ units of flow from X to v'_i (v) return $\mathcal{F}$.
- if $v'_a \in T$ then: (i) set $S := S \cup (X \setminus \{v'_i, v'_{i_1}\})$ (ii) set $T := (T \setminus \{v'_a\}) \cup \{v'_i, v'_{i_1}\}$ (iii) reroute some of the flow paths from $\mathcal{F}$ that end in v'_a to that transfer exactly $1 - y_{v'_i}$ units of flow to $v_{i'}$ (that is remove v'_a as the last vertex on those paths and extend the paths by v_i, v'_i) (iv) reroute all the remaining flow paths in $\mathcal{F}$ that end in v'_a to v'_{i_1} (that is remove v'_a and extend those paths by $v_i, v_{i-1}, \ldots, v_{i_1}, v'_{i_1}$) (v) for each $v'_j \in X \setminus \{v'_i, v'_{i_1}\}$ add to $\mathcal{F}$ a flow path $(y_{v'_j}, v'_j, v_j, \ldots, v_{i_1}, v'_{i_1})$ (v) return $\mathcal{F}$.

Finally we prove that if the procedure receives a safe caterpillar structure then it returns a desired rounding flow. The only property of the rounding flow that needs detailed analysis is the assumption that each internal vertex of a flow path has capacity not smaller than its the capacity of its starting point. Let us assume that there exists a path in $\mathcal{F}$ that starts in v'_a, goes though v_b and ends in v'_c, where $L(v'_c) \geq L(v'_a) > L(v_b)$. This contradicts the assumption that $\mathcal{P}$ is safe because $v_b \in \Gamma(\mathcal{P})$. ■

The following theorem summarizes Sections II-A, II-B, II-C, II-D.

Theorem II.19. *For a connected graph G, if LP1 has a feasible solution then we can find a c-feasible solution with integral y-values.*

Proof: Using a feasible solution to LP1, by Lemma II.16, we obtain a c-feasible solution (x', y'), together with a set of vertex disjoint safe δ-caterpillar structures $\mathcal{S}$, such that vertices that do not belong to any caterpillar structure in $\mathcal{S}$ have integral y-value in (x', y'). Next by Lemma II.18 for each δ-caterpillar structure $(P, P') \in \mathcal{S}$ we find a rounding flow $\mathcal{F}_{(P,P')}$. Finally for each δ-caterpillar structure (P, P') we perform chain shifting with respect to $\mathcal{F}_{(P,P')}$, and by Lemma II.10 we obtain a c'-feasible solution (x'', y'') to LP1.

By Lemma II.16, vertices outside of $\mathcal{S}$ have integral y-value in (x', y'). Moreover by Definition II.17, after chain shifting all the vertices in each caterpillar structure of $\mathcal{S}$ have integral y-values in (x'', y''). ■

E. Rounding x-values

In this section we show how to extend Theorem II.19 to obtain not only integral y-values, but also integral x-values. The following lemma is standard (using network flows).

Lemma II.20. *Let (x, y) be a δ-feasible solution such that all y-values are integral. There is a polynomial time algorithm that creates a δ-feasible solution which has both x- and y-values integral.*

As a consequence of Theorem II.19 and the above lemma the proof Theorem I.2 follows.

III. Conclusions and Open Problems

We have obtained the first constant approximation ratio for the k-center problem with non-uniform hard capacities. The approximation ratio we obtain is in the order of hundreds (however we do not calculate it explicitly), so the natural open problem is to give an algorithm with a reasonable approximation ratio. Moreover, we have shown that the integrality gap of the standard LP formulation for connected graphs in the uniform capacities case is either 5 or 6, which we think might be an evidence, that it should be possible to narrow the gap between the known lower bound of $(2 - eps)$ and upper bound 6 in the uniform capacities case.

Acknowledgements

We are thankful to anonymous referees for their helpful comments and remarks.

References

[1] M. Cygan, M. Hajiaghayi, and S. Khuller, "LP rounding for k-centers with non-uniform hard capacities," *CoRR*, vol. abs/1208.3054, 2012.

[2] M. R. Garey and D. S. Johnson, *Computers and intractability*. Freeman, 1979.

[3] T. Gonzalez, "Clustering to minimize the maximum intercluster distance," *Theoretical Computer Science*, vol. 38, pp. 293–306, 1985.

[4] D. Hochbaum and D. Shmoys, "A best possible heuristic for the k-center problem," *Mathematics of Operations Research*, vol. 10, pp. 180–184, 1985.

[5] ——, "A unified approach to approximation algorithms for bottleneck problems," *Journal of the ACM*, vol. 33, pp. 533–550, 1986.

[6] W. Hsu and G. Nemhauser, "Easy and hard bottleneck location problems," *Discrete Applied Mathematics*, vol. 1, pp. 209–216, 1979.

[7] S. Khuller and Y. J. Sussmann, "The capacitated *k*-center problem," *SIAM J. Discrete Math.*, vol. 13, no. 3, pp. 403–418, 2000.

[8] J. Bar-Ilan, G. Kortsarz, and D. Peleg, "How to allocate network centers," *Journal of Algorithms*, vol. 15, pp. 385–415, 1993.

[9] J. Chuzhoy, S. Guha, E. Halperin, S. Khanna, G. Kortsarz, R. Krauthgamer, and J. Naor, "Asymmetric -center is log* -hard to approximate," *J. ACM*, vol. 52, no. 4, pp. 538–551, 2005.

[10] S. Guha, R. Hassin, S. Khuller, and E. Or, "Capacitated vertex covering with applications," *J. Algorithms*, vol. 48, no. 1, pp. 257–270, 2003.

[11] R. Gandhi, S. Khuller, S. Parthasarathy, and A. Srinivasan, "Dependent rounding and its applications to approximation algorithms," *J. ACM*, vol. 53, no. 3, pp. 324–360, 2006.

[12] J. Chuzhoy and J. Naor, "Covering problems with hard capacities," in *Proc of. FOCS'02*, 2002, pp. 481–489.

[13] R. Gandhi, E. Halperin, S. Khuller, G. Kortsarz, and A. Srinivasan, "An improved approximation algorithm for vertex cover with hard capacities," *JCSS*, vol. 72, no. 1, pp. 16–33, 2006.

[14] D. Shmoys, E. Tardos, and K. Aardal, "Approximation algorithms for facility location problems," in *Proc. of STOC'97*, 1997, pp. 265–274.

[15] J. Lin and J. Vitter, "ϵ-approximations with minimum packing constraint violation," in *Proc. of STOC'92*, 1992, pp. 771–782.

[16] J. Byrka, "An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem," in *Proc. of APPROX'07*, 2007, pp. 29–43.

[17] S. Li, "A 1.488 approximation algorithm for the uncapacitated facility location problem," in *Proc. of ICALP'11*, 2011, pp. 77–88.

[18] S. Guha and S. Khuller, "Greedy strikes back: Improved facility location algorithms," *Journal of Algorithms*, vol. 31, pp. 228–248, 1999.

[19] K. Jain and V. V. Vazirani, "Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and Lagrangian relaxation," *Journal of the ACM*, vol. 48, no. 2, pp. 274–296, 2001.

[20] M. R. Korupolu, C. G. Plaxton, and R. Rajaraman, "Analysis of a local search heuristic for facility location problems," *Journal of Algorithms*, vol. 37, no. 1, pp. 146–188, 2000.

[21] F. A. Chudak and D. P. Williamson, "Improved approximation algorithms for capacitated facility location problems," *Math. Program.*, vol. 102, no. 2, Ser. A, pp. 207–222, 2005.

[22] M. Pál, É. Tardos, and T. Wexler, "Facility location with nonuniform hard capacities," in *Proc. of FOCS'01*, 2001, pp. 329–338.

[23] M. Mahdian and M. Pál, "Universal facility location," in *Proc. of ESA'03*, 2003, pp. 409–421.

[24] J. Zhang, B. Chen, and Y. Ye, "A multi-exchange local search algorithm for the capacitated facility location problem," in *Proc. of IPCO'04*, 2004, pp. 219–233.

[25] R. Levi, D. B. Shmoys, and C. Swamy, "LP-based approximation algorithms for capacitated facility location," in *Proc. of IPCO'04*, 2004, pp. 206–218.

[26] M. H. Bateni and M. T. Hajiaghayi, "Assignment problem in content distribution networks: unsplittable hard-capacitated facility location," *ACM Trans. Algorithms*, to appear. A preliminary version appeared in *Proc. of SODA'09*, 2009, pages 805–814.

[27] M. Charikar, S. Guha, É. Tardos, and D. Shmoys, "A constant-factor approximation algorithm for the k-median problem," in *Proc. of STOC'99*, May 1999, pp. 1–10.

[28] V. Arya, N. Garg, R. Khandekar, K. Munagala, and V. Pandit, "Local search heuristics for k-median and facility location problems," in *Proc. of STOC'01*, 2001, pp. 21–29.

[29] J. Chuzhoy and Y. Rabani, "Approximating k-median with non-uniform capacities," in *Proc. of SODA'05*, 2005, pp. 952–958.

[30] Y. Bartal, M. Charikar, and D. Raz, "Approximating min-sum k-clustering in metric spaces," in *Proc. of STOC'01*. New York, NY, USA: ACM, 2001, pp. 11–20.

[31] J. J. Karaganis, "On the cube of a graph," *Canad. Math. Bull.*, vol. 11, pp. 295–296, 1968.

[32] M. Sekanina, "On an ordering of the set of vertices of a connected graph," *Technical Report Publ. Fac. Sci. Univ. Brno*, vol. 412, 1960.