Julius-Maximilians-Universität Würzburg
Institut für Informatik
Lehrstuhl für Informatik I
Algorithmen, Komplexität und wissensbasierte Systeme

Kumulative Habilitation

Approximation Algorithms for Network Design and Location Problems

Joachim Spoerhase

Eingereicht am 6. März 2017

Kommission:
Prof. Dr. Alexander Wolff
Prof. Dr. Victor Chepoi
Prof. Dr. Christian Glaßer

Abstract

The task of computing an optimal solution to a given problem instance is ubiquitous in computer science and is usually referred to as an *optimization problem.* In a *combinatorial* optimization problem, any problem instance or feasible solution is described by a *discrete* structure involving, for example, finite sets, graphs, or integral variables. This is in contrast to *continuous* optimization where the focus is on numerical problems and where instances and solutions are described by real-valued variables. Interestingly enough, algorithmic approaches for *combinatorial* optimization often rely on solving a *continuous* relaxation of the problem; we employ this idea several times in this thesis, too.

Most of the combinatorial optimization problems arising in practice are *NP-hard*, which implies that (under usual complexity-theoretic assumptions) there is no efficient algorithm to compute an optimum solution efficiently, that is, in polynomial time.

A common way to attack NP-hard optimization problems is to devise efficient algorithms that are not guaranteed to find an optimal but always a "good" solution. An *approximation algorithm* is such an algorithm that has a *provable performance guarantee.* That is, we can show an upper bound on the deviation of any solution computed by this algorithm from the optimal solution of the respective problem instance. A prominent example is the travelling salesperson problem (TSP) where the goal is to find a shortest round trip that visits a given set of locations in a metric space. The well-known approximation algorithm by Christofides is guaranteed to efficiently compute a tour that is at most 3/2 times as expensive as an optimal tour.

Combinatorial problems usually give rise to a big variety and diversity in structure. Changing a problem even in a seemingly simple way may change its algorithmic solvability dramatically. Hence, discrete algorithmic techniques have often been developed for a specific problem rather than in a general manner. And it is the (often non-trivial) task of the algorithm designer to tailor such techniques to the particular problem at hand. In the particular case of approximation algorithms, the community working on such algorithms has singled out certain central combinatorial problems (such as set cover, TSP, Steiner tree, or facility location) that serve—due to their simple, fundamental structure—as a test-bed for new algorithmic tools. It has, in fact, turned out that substantial improvements for these problems usually go hand in hand with the development of new techniques that can be applied to a variety of other, more specific problems as well.

The objective of this thesis is to study approximation algorithms for NP-hard problems that are motivated by real-world *network design* and *location* problems. Following the above-outlined methodology, we aim at examining a variety of structurally different problems with the idea of covering several types of problems. Our focus lies on investigating problems that are considered to be among the most central ones in the

field of approximation algorithms; but we also examine new or more recently proposed problems—in particular in the case of geometric optimization problems. A recurring theme of this thesis is that we shed new light on these classical optimization problems. We study these problems under a new perspective such as adding an additional constraint (for example, a packing, covering, cardinality, or length constraint) that makes the problem substantially harder, studying a new parameterization of it, or investigating it in a more general search space.

In this summary, we give a brief overview over the results that we obtained for the particular problems under investigation. We also sketch the algorithmic techniques used.

Contents

1 Network Design Problems

In a network design problem we are looking for a subgraph of a given graph that satisfies certain structural constraints (such as spanning a certain subset of the vertices) and that optimizes a certain quality measure (such as minimizing the number of edges). Classical examples are the well-known Steiner tree problem (see Section 1.5) and the travelling salesperson problem, which we mentioned in the abstract.

In this chapter, we discuss our results on approximation algorithms for network design problems. We distinguish two types of such problems. First, in Sections 1.1–1.3, we consider problems where the input involves an *arbitrary* graph. Second, in Sections 1.4–1.6, we discuss problems arising in a *geometric* context (such as rectilinear networks).

1.1 Maximum Edge-Disjoint Paths

The *maximum edge-disjoint paths* problem is one of the classical NP-hard routing problems. The input is an undirected graph G with n nodes and a set of k node pairs (s_i, t_i) called *terminal pairs*. The objective is to find a subset of terminal pairs of maximum cardinality that can be routed via edge-disjoint paths. The approximability of the problem is currently not well understood. There is no $2^{o(\sqrt{\log n})}$-approximation algorithm unless all problems in NP have algorithms with running time $n^{O(\log n)}$ [CKN17]. This inapproximability bound constitutes a significant gap to the best known approximation upper bound of $\mathcal{O}(\sqrt{n})$ due to Chekuri et al. [CKS06]; closing this gap is currently one of the big open problems in approximation algorithms. In their seminal paper, Raghavan and Thompson [RT87] introduce the technique of randomized rounding for LPs, which yields an $\mathcal{O}(1)$-approximation when edges may be used by $\mathcal{O}\left(\frac{\log n}{\log \log n}\right)$ paths.

To gain a deeper understanding of the problem, the approximability parameterized with the *tree-width* w of the input graph has been studied [CNS13]. For $w = 1$, that is on trees, the problem is efficiently solvable [GVY97]. For general w, an $O(w^3)$-approximation algorithm is known [EMPR16]. It has been conjectured that the best possible bound is $O(w)$ since the largest known integrality gap of the standard LP relaxation of the problem is $\Omega(w)$ [CNS13].

Motivated by this gap between upper and lower bound in terms of the tree-width, we [FMS16] propose to study the problem with respect to another parameter that measures how tree-like a graph is and that is lower bounded by the tree-width. In particular, we analyze the approximability in terms of the *feedback vertex set number* r of a graph, which measures its vertex deletion distance to a forest. In fact, we can show that the problem is already NP-hard for $r = 1$. The bounds that we obtain strengthen the above fundamental results for *general* graphs. In particular, we obtain first an $\mathcal{O}(\sqrt{r} \cdot \log kr)$-approximation algorithm where k is the number of terminal pairs in the input. As $r \leq n$, up to logarithmic factors, our result strengthens the best known ratio $\mathcal{O}(\sqrt{n})$

due to Chekuri et al. Second, we show how to route $\Omega(\text{OPT})$ pairs with congestion $\mathcal{O}\left(\frac{\log kr}{\log\log kr}\right)$, strengthening the bound obtained by the classic approach of Raghavan and Thompson.

Our algorithms employ the standard multi-commodity flow relaxation of the problem [CKS06], which finds for each terminal pair a collection of (possibly fractional) flow paths and ensures that each terminal and each edge is passed by at most one unit of flow. The total flow of the optimum flow provides an upper bound on OPT.

In the randomized rounding approach of Raghavan and Thompson [RT87] every flow path is routed with probability equal to its flow value. This is done in a natural dependent manner that ensures that each terminal pair is routed by at most one of the picked paths. In order to prove our strengthened congestion bound $O(\log kr/\log\log kr)$, we propose a non-trivial *preprocessing* of the optimum LP solution that is applied prior to the randomized rounding. In this step, we aggregate the flow paths by a careful rerouting so that the flow "concentrates" in special nodes (so-called *hot spots*) in the sense that if all edges incident on hot spots have low congestion, then so have all edges in the graph. A crucial point that allow us to show the improved bound is that the number of these hot spots is polynomial in k and r.

Our $O(\sqrt{r}\log kr)$-approximation algorithm is also based on rounding the multi-commodity flow relaxation. Similarly to the algorithm of Chekuri et al. [CKS06], we distinguish the two cases where the majority of flow paths is, in a certain sense, short or long, respectively. In particular, in the first case many flow paths visit a large number of nodes in the feedback vertex set R. Then there must be a single node carrying a significant fraction of the total flow and a good fraction of this flow can be realized by integral paths by solving a single-source flow problem. This case is analogous to the approach of Chekuri et al. The second case where a majority of the flow paths visit only a few nodes in R turns out to be more challenging, since any such path may still visit an unbounded number of edges in terms of k and r (in contrast to the work of Chekuri et al.). We use two main ingredients to overcome this difficulty. First, we apply our first result as a building block to obtain a solution with logarithmic congestion while losing only a constant factor in the approximation ratio. Secondly, we introduce the idea of *irreducible routings with low congestion*, which allows us to exploit the structural properties of the graph and the congestion property to identify a sufficiently large number of flow paths blocking only a small amount of flow. These paths can then be routed in a greedy manner.

1.2 Network Design with Bounded Distances

In a directed graph with n nodes, non-correlated edge lengths and costs, the *network design problem with bounded distances* asks for a cost-minimal spanning subgraph subject to a length bound d for all node pairs. The best known algorithm for this problem is by Dodis and Khanna [DK99] and has a linear approximation ratio of $O(n\log d)$. The algorithm is based on rounding a linear programming relaxation of the problem.

We [CS15b] give a bi-criteria approximation for this problem that achieves for any

$\epsilon > 0$ an approximation ratio $n^{0.5+\epsilon}$ but guarantees only a pair-wise distance of $(2+\epsilon)d$ rather than d. The running time of the algorithm depends on n and ϵ and is polynomial for any fixed $\epsilon > 0$.

As a starting point, our algorithm uses a two-stage approach originally proposed by Feldman et al. [FKN12] for directed Steiner forest and that has later been used for sparse directed spanners [BGJ$^+$12, DK11]. We divide the considered node pairs into *thin* and *thick* pairs. Here, a node pair (s, t) is thin if the set of s–t paths of length at most d covers a "small" number of nodes. Otherwise, the pair is called thick. We settle the thin pairs by LP-rounding, as we have to cover certain cuts w.r.t. shortest paths. For the latter, we sample nodes and construct short in- and out-trees for each of them. This latter part is a main technical challenge: In contrast to the case of sparse spanners, we cannot simply use shortest-path trees, as they could have arbitrarily high costs.

To solve this issue, we turn our attention to a second problem, which is also of independent interest, called *directed shallow-light Steiner tree*. In this problem we are given a digraph with non-correlated edge lengths and costs, a distinguished root node r, a set of k nodes called *terminals*, and for each terminal an individual length bound. The objective is to find an r-rooted directed subtree that spans all terminals, ensures that the distance of each terminal from the root is upper bounded by the length bound of the terminal and that minimizes the total cost among all such trees.

Kortsarz and Peleg [KP97] gave an $O(k^\varepsilon)$-approximation for undirected graphs with uniform edge lengths and uniform distance bound. We obtain the first non-trivial result for the general directed problem. In fact, at the cost of violating the length bounds by a factor of at most $1 + \varepsilon$, we obtain the same approximation ratio as [KP97], but for directed graphs and without the restrictions to uniform lengths and costs.

Finally, we show how to apply our results to obtain an $(\alpha+\varepsilon, O(n^{0.5+\varepsilon}))$-approximation for *light-weight directed α-spanners*. For this, no non-trivial approximation algorithm has been known before. All running times depend on n and ε and are polynomial in n for any fixed $\varepsilon > 0$.

1.3 Degree-Based Spanning Tree Problems

The notion of *degree-based spanning tree problems* has been introduced by Salamon [Sal10]. In such a problem we look for a spanning tree of a given undirected graph that optimizes an objective function that depends on the distribution of *node degrees* in the spanning tree. Optimizing this *structural* aspect is in contrast to many other network design problem where the objective is to minimize total edge or node weights (such as in minimum spanning tree or minimum Steiner tree problem). Degree-based spanning tree problems can be motivated by the design of communication networks where nodes represent communication devices whose cost depend on their functionality, which is, in turn, reflected by the degree of the node. For example, a receiver may be represented by a leaf, a forward node by a node of degree two, and a router or splitter by a node of degree at least three.

A variety of such problems has been studied in the approximation algorithms literature. In this thesis, we consider three degree-based spanning tree problems. In the *maximum leaf spanning tree problem* we aim at finding a spanning tree that maximizes the number of leaves. In the *maximum internal spanning tree problem* we wish to maximize the number of internal nodes (non-leaves) in the tree. Finally, in the *maximum path-node spanning tree problem* we maximize the number of nodes one or two (which can be motivated by the design of optical networks).

In contrast to the previous two problems that we tackled by LP rounding approaches, known algorithms for degree-based spanning tree problems are *combinatorial.* Below we discuss two local search and one greedy algorithm for the above-mentioned problems. The analyses of the two local search algorithms are based on sophisticated charging schemes that relate a locally optimal solution with a globally optimal one.

Maximum Leaf Spanning Tree We [SSW11] consider the *maximum leaf spanning tree* problem (MLST) on digraphs. MLST is NP-hard [SO98] and existing approximation algorithms for MLST on digraphs have ratios of $O(\sqrt{\mathrm{OPT}})$ [DV10] and 92 [DT09].

We focus on the special case of *acyclic* digraphs and propose two linear-time approximation algorithms; one with ratio 4 that uses a result of Daligault and Thomassé [DT09] and one with ratio 2 based on a greedy 3-approximation algorithm of Lu and Ravi [LR98] for the undirected version of the problem. Our analysis of the greedy algorithm is inspired by a clever analysis of Solis-Oba [SO98] for the undirected case. We complement these positive results by observing that MLST is MaxSNP-hard on acyclic digraphs. Hence, this special case does not admit a PTAS (unless P = NP).

Maximum Internal Spanning Tree The best approximation algorithm known prior for the *maximum internal spanning tree* problem is due to Prieto and Sloper [PS05] and has a ratio of 2. For graphs without pendant nodes, Salamon [Sal09] has lowered this factor to 7/4 by means of local search. However, the approximative behaviour of his algorithm on general graphs has remained open. We [KS15] show that a simplified and faster version of Salamon's algorithm yields a 5/3-approximation even on general graphs. In addition to this, we investigate a node weighted variant of the problem for which Salamon achieved a ratio of $2 \cdot \Delta(G) - 3$. Extending Salamon's approach we obtain a factor of $3 + \epsilon$ for any $\epsilon > 0$. We complement our results with worst case instances showing that our analyses are tight.

Maximum Path-Node Spanning Tree Given an undirected, connected graph, the aim of the *minimum branch-node spanning tree problem* is to find a spanning tree with the minimum number of nodes of degree larger than two. The problem is motivated by optical network design problems where junctions are significantly more expensive than simple end- or through-nodes, and are thus to be avoided [GHSV02, GH03]. Unfortunately, it is NP-hard to recognize instances that admit an objective value of zero, rendering the search for guaranteed approximation ratios futile.

We [CS15a] suggest to investigate a complementary formulation, called *maximum path-node spanning tree*, where the goal is to find a spanning tree that maximizes the number of nodes with degree at most two. While the optimal solutions (and the practical applications) of both formulations coincide, our formulation proves more suitable for approximation. In fact, it admits a trivial 1/2-approximation algorithm. Our main contribution is a local search algorithm that guarantees a ratio of $6/11$, as well as showing that the problem is APX-hard, that is, it does not allow a polynomial time approximation scheme (PTAS).

1.4 Manhattan Network Problems

In contrast to the problems discussed above, Manhattan network problems are *geometric* network design problems. In particular, we [DGK$^+$15, DFK$^+$17] consider the *generalized minimum Manhattan network problem* (GMMN). The input to this problem is a set R of n pairs of terminals, which are points in $\mathbb{R}^2$. The goal is to find a minimum-length rectilinear network that connects every pair in R by a *Manhattan path*, that is, a path of axis-parallel line segments whose total length equals the pair's Manhattan distance. This problem is a natural generalization of the extensively studied *minimum Manhattan network problem* (MMN) in which R consists of all possible pairs of terminals [GLN01]. Another important special case is the well-known *rectilinear Steiner arborescence problem* (RSA) [LR00]. As a generalization of these problems, GMMN is NP-hard. No approximation algorithms were known for general GMMN.

We obtain an $O(\log n)$-approximation algorithm for GMMN. First, we use a simple (yet powerful) divide-and-conquer scheme to reduce the problem to RSA. This yields a ratio of $O(\log^2 n)$. To bring down the ratio to $O(\log n)$ we develop a new *stabbing technique*, which is a novel way to approach Manhattan network problems and constitutes the main technical contribution of this work.

Our result is a first step towards answering the open question of Chepoi et al. [CNV08] whether or not there is a constant-factor approximation algorithm for GMMN. We give indications that it may be challenging to obtain an $O(1)$-approximation and demonstrate why techniques working for MMN and RSA seem to fail.

We also study the case of higher dimensions. In fact, some parts of our algorithm generalize to higher dimensions, yielding a simple $O(\log^{d+1} n)$- approximation algorithm for the problem in arbitrary fixed dimension d. As a corollary, we obtain an exponential improvement upon the previously best $O(n^\epsilon)$-ratio for MMN in d dimensions (an earlier result of us [DGK$^+$15]). En route, we show that an existing $O(\log n)$-approximation algorithm for 2D-RSA generalizes to higher dimensions.

For dimension $d = 3$, we also give a $4(k-1)$-approximation algorithm for the case that the terminals are contained in the union of $k \geq 2$ parallel planes [DGK$^+$15]. This result is based on an interesting connection to a rectangle piercing problem on the plane that can be solved efficiently.

1.5 Non-Crossing Steiner Forest

Steiner tree is a fundamental problem in combinatorial optimization. Given an edge-weighted graph and a set of vertices called *terminals*, the task is to find a minimum-weight subgraph that connects the terminals. For the closely related and well-studied

Steiner forest problem, the terminals are colored, and the desired subgraph must connect, for each color, the terminals of that color.

We [BFK+15] consider a geometric variant of Steiner forest where we add the constraint of planarity and require that terminals with distinct colors lie in distinct connected components. More precisely, we consider the problem of computing, for a k-colored set of points in the plane (which we also call *terminals*), k pairwise non-crossing planar Euclidean Steiner trees, one for each color. Note that such trees exist for every given set of points. The problem was introduced by Efrat et al. [EHKP15] and can be used for visualizing embedded and clustered graphs. We call the problem of minimizing the total length of these trees *k-Colored Non-Crossing Euclidean Steiner Forest* (k-CESF).

For $k = 1$, this is the well-known Euclidean Steiner tree problem. For k-CESF, we present a deterministic $(k + \epsilon)$-approximation algorithm (improving on a known $1.21k$-approximation algorithm [EHKP15]) and a randomized $O(\sqrt{n} \log k)$-approximation algorithm (this bound was previously only known for matchings [CHKL13]). Our main result is that 2-CESF admits a PTAS. By a non-trivial modification of this PTAS, we prove that 3-CESF admits a $(5/3 + \epsilon)$-approximation algorithm. Our PTAS for 2-CESF uses some ideas of Arora's algorithm [ARR98] for Euclidean Steiner tree, which is equivalent to 1-CESF. Since, in a solution to 2-CESF, the two trees are not allowed to cross, our approach differs from Arora's algorithm in several respects. We use a different notion of *r-lightness*, and by a *portal-crossing reduction* we achieve that each portal is crossed at most three times. More care is also needed in the perturbation step and in the base case of the dynamic program.

1.6 Box Representations

We [BvDF+17] study the following geometric representation problem: Given a graph whose vertices correspond to axis-aligned rectangles with fixed dimensions, arrange the rectangles without overlaps in the plane such that two rectangles touch if the graph contains an edge between them. This problem is called Contact Representation of Word Networks (Crown) since it formalizes the geometric problem behind drawing word clouds in which semantically related words are close to each other. Crown is known to be NP-hard, and there are approximation algorithms by Barth et al. [BFK+14] for certain graph classes for the optimization version, in which realizing each desired adjacency yields a certain profit. This optimization version can be viewed as a network design problem, in which we aim at realizing the maximum profit subgraph.

We present the first $O(1)$-approximation algorithm for the general case, when the input is a complete weighted graph, and a stronger bound for the bipartite case. Since the subgraph of realized adjacencies is necessarily planar, we also consider several planar graph classes (namely stars, trees, outerplanar, and planar graphs), improving upon the known results for these graph classes [BFK+14]. For some graph classes, we also describe improvements in the unweighted case, where each adjacency yields the same profit. Finally, we show that the problem is APX-complete on bipartite graphs of bounded maximum degree.

The results of Barth et al. are simply based on existing decompositions of the respective graph classes into star forests or cycles. For stars and cycles straightforward or existing algorithms are used.

Our results rely on a variety of algorithmic tools. First, we devise sophisticated decompositions of the input graphs into heterogeneous classes of subgraphs, which also requires a more general combination method than that of Barth et al. Second, we use randomization to obtain a simple constant-factor approximation for general weighted graphs. Previously, such a result was not even known for unweighted bipartite graphs. Third, to obtain an improved algorithm for the unweighted case, we prove a lower bound on the size of a matching in a planar graph of high average degree. Fourth, we use a planar separator result of Frederickson [Fre91] to obtain a polynomial-time approximation scheme (PTAS) for degree-bounded planar graphs.

2 Location Problems

In metric location problems, the input consists of a set of clients, a set F of facilities and a metric distance function between clients and facilities. The goal is to select a subset $S \subseteq F$ of facilities, and an assignment of clients to the selected facilities, that together minimize a certain problem-specific cost function. One can think of F being a set of potential facility locations, whereas S contains the locations where we decide to open (that is, to build) facilities.

For example, in the *k-center* problem, we aim at opening k facilities such that the maximum distance of a client to the facility serving it is minimized. In the well-studied problem *facility location* there is no bound on the number of facilities but instead each facility has an individual opening cost and the objective is to minimize the total opening cost plus the total connection cost. Finally, in the *k-median problem* we want to open k facilities so as to minimize the total connection cost. For k-center and facility location (nearly) optimal constant-factor approximation algorithms are known, that is, the proven approximation bounds [HS85, Li11] match (or nearly match) the best inapproximability bound [HS85, GK99]. Also for the k-median problem constant-factor approximation algorithms are known [CGTS99], but there is still a significant gap between the best known upper bound of 2.675 [BPR$^+$15] and the best known inapproximability bound of 1.73 [GK99]. The k-median problem seems the hardest one among the above-mentioned three location problems in terms of approximation algorithms.

2.1 Capacitated k-Median

Also *capacitated* variants of all three above-mentioned (and also further) location problems have been studied extensively [CHK12, PTW01, CR05]. In the capacitated version each facility has an upper bound on the number of clients it may serve. While the capacitated variants turn out substiantially harder to solve than the uncapacitated counterparts, constant-factor approximation algorithms have been obtained for capacitated k-center [CHK12] and capacitated facility location problems [PTW01]. For capacitated k-median, however, the approximability status is still unknown despite significant efforts by the community. And it is one of the central open questions in approximation algorithms whether or not also this problem admits a constant-factor approximation algorithm.

A main difficulty in approximating the problem is that the standard linear programming formulation (even in the case of uniform capacities) has unbounded integrality gap unless we violate the capacity bound by a factor of at least 2 [CGTS99]. We [BFRS15] construct approximation algorithms for capacitated k-median violating the capacities based on rounding a fractional solution to this relaxation. We show that a violation factor of $2 + \varepsilon$ is in fact sufficient to obtain a constant-factor approximation algorithm with ratio $O(1/\varepsilon^2)$ with respect to the connection cost in the case of uniform capacities. Prior to our work, the best algorithm had a violation factor of 3 [CGTS99] and required to open multiple copies of facilities whereas our algorithm opens each facility at most once.

We extend our $(2+\varepsilon)$-violation algorithm in the following two directions. On one hand, we obtain a $2+\varepsilon$ capacity violating algorithm to the more general k-facility location problem with uniform capacities, where opening facilities incurs a location specific opening cost. On the other hand, we show that violating capacities by a slightly bigger factor of $3+\varepsilon$ is sufficient to obtain a constant factor approximation of the connection cost also in the case of the non-uniform hard capacitated k-median problem. This substantially improves upon the previously known algorithm with a violation of 50 and that required (in contrast to ours) to open multiple copies of facilities [CR05].

Our algorithms first use the clustering of Charikar et al. [CGTS99] to partition the facilities into sets of total fractional opening at least $1-1/\ell$ for some fixed ℓ. Then we exploit the technique of Levi, Shmoys, and Swamy [LSS12], which they developed for the capacitated facility location problem, which is to locally group the demand from clients to obtain a system of single node demand instances. Next, depending on the setting, we either work with stars of facilities (for non-uniform capacities), or we use a dedicated routing tree on the demand nodes (for non-uniform opening cost), to redistribute the demand that cannot be satisfied locally within the clusters.

In a recent work subsequent to ours, Demirci and Li [DL16] study a much stronger LP relaxation by means of which they can even achieve a $(1+\varepsilon)$-capacity violation. The question for a constant-factor approximation algorithm for capacitated k-median remains, however, open.

2.2 Knapsack Median

Knapsack median is a generalization of the classic k-median problem in which we replace the cardinality constraint with a knapsack constraint. More precisely, we are given for each facility an individual opening cost and a bound on the total opening cost. A major difficulty in approximating this problem lies in the fact that the standard LP has an unbounded integrality gap. Kumar [Kum12] was the first to get around this difficulty by combining the lower bound of the LP with a clever *combinatorial* bound to obtain the first constant-factor approximation. There is a series of improved bounds and the best algorithm prior to our work has a ratio of 32 and is due to Swamy [Swa16]. We [BPR+17] improve on the best known algorithms in several ways, including adding randomization and applying sparsification as a preprocessing step. The latter improvement produces the first LP for this problem with bounded integrality gap. The new algorithm obtains an approximation factor of 17.46.

Our algorithm has a flow similar to Swamy's: we first get a half-integral solution (except for a few "bad" facilities), and then create pairs of half-facilities, opening one facility in each pair. By making several improvements, we reduce the approximation ratio to 17.46. The first improvement is a simple modification to the pairing process so that every half-facility is guaranteed either itself or its closest neighbor to be open (versus having to go through two "jumps" to get to an open facility). The second improvement is to *randomly* sample the half-integral solution, and condition on the probability that any given facility is "bad". The algorithm can be derandomized with linear loss in the runtime. The third improvement deals with the bad facilities, which inevitabley arise due to the knapsack constraint. All previous algorithms used Kumar's bound [Kum12] to bound the cost of nearby clients when bad facilities must be closed. However, we show that by using a sparsification technique similar in spirit to—but distinct from—that used by Li and Svensson [LS13], we can focus on a subinstance in which the connection costs of clients are guaranteed to be evenly distributed throughout the instance. This allows for a much stronger bound than Kumar's, and also results in an LP with bounded integrality gap, unlike previous algorithms.

We also give a bi-criteria algorithm with factor 3.05 and with a budget violation of $1 + \epsilon$ for any $\epsilon > 0$. (The running time of this algorithm depends on ϵ.)

2.3 Maximum Betweenness Centrality

A question that frequently arises in the analysis of complex networks is how central or important a given node is. Examples of such complex networks are communication or logistical networks. In the previous sections we discussed k-median and k-center problems aiming at minimizing connections costs. Here, in contrast, we [FS11] consider a centrality measure that aims at monitoring communication and is called *shortest path betweenness centrality* [Fre77, Bra08]. This measure can be motivated by the following scenario that relies only on very basic assumptions.

The *maximum betweenness centrality* problem (MBC) can be defined as follows. Given a graph, find a k-element node set C that maximizes the probability of detecting communication between a pair of nodes s and t chosen uniformly at random. It is assumed that the communication between s and t is realized along a shortest s–t path which is, again, selected uniformly at random. The communication is detected if the communication path contains a node of C. Dolev et al. [DEPZ09] showed that MBC is NP-hard and gave a $(1 - 1/e)$-approximation algorithm using a greedy approach. We provide a reduction of MBC to the maximum coverage problem that simplifies the analysis of the algorithm of Dolev et al. considerably. Our reduction allows us to obtain a new algorithm with the same approximation ratio for a (generalized) budgeted version of MBC, in which every node has a cost and we are given a budget that specifies an upper bound on the total cost of the chosen node set C. We provide tight examples showing that the analyses of both algorithms are best possible. Moreover, we prove that MBC is APX-complete by a reduction from the maximum k-vertex cover problem and provide an exact polynomial-time algorithm for MBC on tree graphs based on dynamic programming.

2.4 Maximizing Monotone Submodular Functions Subject to a Covering and a Packing Constraint

Suppose you are the producer of a certain good and you wish to produce a given minimum amount P of this good, for example, because of projections, customer demands or to ensure a certain market share. To this end, you want to open at most k factories. Assume that you have a certain ground set U of options for opening factories. Each option e in this set specifies the amount $p(e)$ of the goods that e can produce. Your aim is to find a k-subset S of U to maximize the *gain* $f(S)$ (revenue minus production cost) and so that S can produce P units of the good. In practice, the actual gain may be a complicated function depending, for example, on the location and the precise demand distribution of the customers. We will work with the assumption that the gain function f is *submodular* [NWF78]. Here, submodular means that f has the property of *diminishing returns*, which is a reasonable assumption in most practical settings. More precisely, if $e \in U$ is an element and $X \subseteq Y \subseteq U$ are two subsets not containing e then its incremental gain $f(X + e) - f(X)$ with respect to X is not smaller than its incremental gain $f(Y + e) - f(Y)$ with respect to Y. The function f is called *monotone* if $f(X) \leq f(Y)$ for all $X \subseteq Y \subseteq U$.

Submodular functions are a general means of modelling the principle of *diminishing returns* in discrete optimization [NWF78]. They are the discrete analogous of convexity and capture, for example, problems such as maximum cut, maximum di-cut, generalized assignment, maximum coverage, maximum bi-section and maximum facility location.

Optimization of submodular functions has attracted lots of attention from the combinatorial optimization community in the last few years. Initially, combinatorial algorithms using greedy and local search techniques were proposed for maximizing submodular functions subject to cardinality or packing constraints [NWF78, Svi04]. But as the complexity of the constraints increased (such as matroid or multiple knapsack constraints), it became necessary to look at more sophisticated techniques using a continuous relaxation known as the *multilinear relaxation* [CCPV11, KST09, VCZ11, EN16].

We [SU17] take a detour from the recently used techniques and propose a new combinatorial algorithm that achieves constant-factor approximations for maximizing monotone submodular functions subject to a *covering* constraint and a *packing* constraint for maximizing monotone submodular functions. Our algorithm violates both constraints by a factor of $1+\epsilon$ for any $\epsilon > 0$. We remark that a violation cannot be avoided since already checking existence of a feasible solution is NP-hard.

Existing approaches for maximizing submodular functions usually exploit that the underlying polytope describing the constraints is down-closed. Our results are the first to handle a general covering constraint, which is not down-closed. We propose a combinatorial approach that seems novel to us. We combine the greedy approach with a *dynamic programming (DP) table* that can handle more complex constraints and controls the greedy process. As a result, our dynamic programming table does not contain optimum partial solutions (as it is common for DPs) but rather approximate solutions and the propagation steps in the DP are made according to a greedy rule. We are not aware of a similar usage of DP in submodular optimization and more general in approximation algorithms. We also feel that our approach is simple and natural enough so that it may have applications for other problems as well. It would, for example, be interesting to see if this approach can handle other problems with additional complex constraints where the basic variants (with simple constraints) can be tackled by a greedy approach. In particular, we point out an interesting connection of our approach to the capacitated

k-median problem and give a non-trivial approximation algorithm *without any violation* of the constraints for metrics with two distances.

2.5 Geometric Coverage Problems

The *maximum coverage* (MC) problem is one of the classic combinatorial optimization problems which is well studied due to its wealth of applications. In this problem, we are given a family of sets over a given universe of ground elements. The objective is to find a given number k of sets in the family such that the number of covered ground elements is maximized. A possible application of maximum coverage lies in the area of location problems where each set in the family corresponds to the set of clients that can be served by a potential facility and where the objective is to open k facilities so as to maximize the number of clients served. The problem is closely related to the well-known *set cover* (SC) problem where we want to find the smallest number of sets that cover the *all* ground elements.

In their seminal work, Mustafa and Ray [MR09] showed that a wide class of *geometric* SC problems admit a PTAS via local search—this is one of the most general approaches known for such problems. Their result applies if a naturally defined "exchange graph" for two feasible solutions is planar and is based on subdividing this graph via a planar separator theorem due to Frederickson [Fre91]. Obtaining similar results for the related MC seems non-trivial due to the hard cardinality constraint. In fact, while Badanidiyuru, Kleinberg, and Lee [BKL12] have shown (via a different analysis) that local search yields a PTAS for MC with two-dimensional real halfspaces, they only conjectured that the same holds true for dimension three. Interestingly, at this point it was already known that local search provides a PTAS for the corresponding set cover case and this followed directly from the approach of Mustafa and Ray.

It is possible to construct the same exchange graphs as in the case of SC also for MC. However, the hard cardinality constraint given by input parameter k poses an obstacle. In particular, when considering a swap corresponding to a part of the subdivision, this swap might be infeasible as it may contain (substantially) more sets from the global optimum than from the local optimum. Another issue is that MC has a different objective function than SC. Namely, the goal is to maximize the number of covered elements rather than minimizing the number of used sets.

We [CDRS16] provide a way to address the above-mentioned issues. First, we propose a *color-balanced* version of the planar separator theorem. The resulting subdivision approximates locally in each part the global distribution of the colors. Second, we show how this roughly balanced subdivision can be employed in a more careful analysis to strictly obey the hard cardinality constraint. More specifically, we obtain a PTAS for any "planarizable" instance of MC and thus essentially for all cases where the corresponding SC instance can be tackled via the approach of Mustafa and Ray. As a corollary, we confirm the conjecture of Badanidiyuru, Kleinberg, and Lee [BKL12] regarding real half spaces in dimension three. We feel that our ideas could also be helpful in other geometric settings involving a cardinality constraint.

Conclusion

In this thesis, we have studied approximation algorithms for combinatorial optimization problems. The focus was on examining the effect of adding a new constraint (such as a cardinality, knapsack, capacity, covering or length constraint) to a classical optimization problem. In the case of edge-disjoint paths, we have considered a new parameterization, and for Manhattan networks we studied a generalized version. While the particular problems under consideration were structurally quite different, the presence of such an additional constraint (or new aspect) always rendered the problem under investigation substantially harder to tackle.

For some of the problems, rounding continuous relaxations of the problem has turned out to be a powerful approach even under the presence of additional constraints but we were sometimes required to relax those constraints.

For strongly constrained problems where "structure" is a decisive factor (such as for geometric problems or degree-based spanning tree problems), combinatorial algorithmic approaches such as divide-and-conquer, greedy, or local search have turned out effective.

In the future, it would be interesting to obtain more general results on how the particular constraints impact the approximability of a problem. It would also be interesting to devise stronger continuous relaxations that do not require the violation of constraints or that are able to cope with highly constrained and structured problems.

Publications Contributing to the Habilitation

[BFK+15] Sergey Bereg, Krzysztof Fleszar, Philipp Kindermann, Sergey Pupyrev, Joachim Spoerhase, and Alexander Wolff. Colored non-crossing Euclidean Steiner forest. In *Proc. 26th International Symposium on Algorithms and Computation (ISAAC'15)*, pages 429–441, 2015.

[BFRS15] Jarosław Byrka, Krzysztof Fleszar, Bartosz Rybicki, and Joachim Spoerhase. Bi-factor approximation algorithms for hard capacitated k-median problems. In *Proc. 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'15)*, pages 722–736, 2015.

[BPR+17] Jarosław Byrka, Thomas Pensyl, Bartosz Rybicki, Joachim Spoerhase, Aravind Srinivasan, and Khoa Trinh. An improved approximation algorithm for knapsack median using sparsification. *Algorithmica*, 2017. accepted.

[BvDF+17] Michael A. Bekos, Thomas C. van Dijk, Martin Fink, Philipp Kindermann, Stephen G. Kobourov, Sergey Pupyrev, Joachim Spoerhase, and Alexander Wolff. Improved approximation algorithms for box contact representations. *Algorithmica*, 77(3):902–920, 2017.

[CDRS16] Steven Chaplick, Minati De, Alexander Ravsky, and Joachim Spoerhase. Approximation schemes for geometric coverage problems. *CoRR*, 2016. http://arxiv.org/abs/1607.06665, submitted.

[CS15a] Markus Chimani and Joachim Spoerhase. Approximating spanning trees with few branches. *Theory Comput. Syst.*, 56(1):181–196, 2015.

[CS15b] Markus Chimani and Joachim Spoerhase. Network design problems with bounded distances via shallow-light Steiner trees. In *Proc. 32nd International Symposium on Theoretical Aspects of Computer Science (STACS'15)*, pages 238–248, 2015.

[DFK+17] Aparna Das, Krzysztof Fleszar, Stephen Kobourov, Joachim Spoerhase, Sankar Veeramoni, and Alexander Wolff. Approximating the generalized minimum Manhattan network problem. *Algorithmica*, 2017. accepted.

[DGK+15] Aparna Das, Emden R. Gansner, Michael Kaufmann, Stephen G. Kobourov, Joachim Spoerhase, and Alexander Wolff. Approximating minimum Manhattan networks in higher dimensions. *Algorithmica*, 71(1):36–52, 2015.

[FMS16] Krzysztof Fleszar, Matthias Mnich, and Joachim Spoerhase. New algorithms for maximum disjoint paths based on tree-likeness. In *Proc. 24th European Symposium on Algorithms (ESA'16)*, pages 42:1–42:17, 2016.

[FS11] Martin Fink and Joachim Spoerhase. Maximum betweenness centrality: Approximability and tractable cases. In *Proc. 5th Workshop on Algorithms and Computation (WALCOM'11)*, pages 9–20, 2011.

[KS15] Martin Knauer and Joachim Spoerhase. Better approximation algorithms for the maximum internal spanning tree problem. *Algorithmica*, 71(4):797–811, 2015.

[SSW11] Nadine Schwartges, Joachim Spoerhase, and Alexander Wolff. Approximation algorithms for the maximum leaf spanning tree problem on acyclic digraphs. In *Proc. 9th Workshop on Approximation and Online Algorithms (WAOA'11)*, pages 77–88, 2011.

[SU17] Joachim Spoerhase and Sumedha Uniyal. Maximizing monotone submodular functions subject to a covering and a packing constraint. 2017, submitted.

References

[ARR98] Sanjeev Arora, Prabhakar Raghavan, and Satish Rao. Approximation schemes for Euclidean k-medians and related problems. In *Proc. 30th Annu. ACM Symp. on the Theory of Computing (STOC'98)*, pages 106–113, 1998.

[BFK+14] Lukas Barth, Sara Irina Fabrikant, Stephen G. Kobourov, Anna Lubiw, Martin Nöllenburg, Yoshio Okamoto, Sergey Pupyrev, Claudio Squarcella, Torsten Ueckerdt, and Alexander Wolff. Semantic word cloud representations: Hardness and approximation algorithms. In *Proc. 11th Latin American Symposium on Theoretical Informatics (LATIN'14)*, pages 514–525, 2014.

[BGJ+12] Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova, and David P. Woodruff. Transitive-closure spanners. *SIAM J. Comput.*, 41(6):1380–1425, 2012.

[BKL12] Ashwinkumar Badanidiyuru, Robert Kleinberg, and Hooyeon Lee. Approximating low-dimensional coverage problems. In *Proc. ACM Symposium on Computational Geometry (SoCG'12)*, pages 161–170, 2012.

[BPR+15] Jaroslaw Byrka, Thomas Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median, and positive correlation in budgeted optimization. In *Proc. 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'15)*, pages 737–756, 2015.

[Bra08] Ulrik Brandes. On variants of shortest-path betweenness centrality and their generic computation. *Social Networks*, 30(2):136–145, 2008.

[CCPV11] Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 40(6):1740–1766, 2011.

[CGTS99] Moses Charikar, Sudipto Guha, Eva Tardos, and David B. Shmoys. A constant factor approximation algorithm for the k-median problem. In *Proc. 31st Annu. ACM Symp. on the Theory of Computing (STOC'99)*, 1999.

[CHK12] Marek Cygan, MohammadTaghi Hajiaghayi, and Samir Khuller. LP rounding for k-centers with non-uniform hard capacities. In *Proc. 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'12)*, pages 273–282, 2012.

[CHKL13] Timothy M. Chan, Hella-Franziska Hoffmann, Stephen Kiazyk, and Anna Lubiw. Minimum length embedding of planar graphs at fixed vertex locations. In *Proc. 21st International Symposium on Graph Drawing (GD'13)*, pages 376–387, 2013.

[CKN17] Julia Chuzhoy, David H.K. Kim, and Rachit Nimavat. New hardness results for routing on disjoint paths. In *Proc. 49th Annual ACM Symposium on the Theory of Computing (STOC'17)*, page (to appear), 2017.

[CKS06] Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. An $O(\sqrt{n})$ approximation and integrality gap for disjoint paths and unsplittable flow. *Theory of Computing*, 2(7):137–146, 2006.

[CNS13] Chandra Chekuri, Guyslain Naves, and F. Bruce Shepherd. Maximum edge-disjoint paths in k-sums of graphs. In *Proc. 40th International Colloquium Automata, Languages, and Programming (ICALP'13)*, pages 328–339, 2013.

[CNV08] Victor Chepoi, Karim Nouioua, and Yann Vaxès. A rounding algorithm for approximating minimum Manhattan networks. *Theor. Comput. Sci.*, 390(1):56–69, 2008.

[CR05] Julia Chuzhoy and Yuval Rabani. Approximating k-median with non-uniform capacities. In *Proc. 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'05)*, pages 952–958, 2005.

[DEPZ09] Shlomi Dolev, Yuval Elovici, Rami Puzis, and Polina Zilberman. Incremental deployment of network monitors based on group betweenness centrality. *Inf. Process. Lett.*, 109(20):1172–1176, 2009.

[DK99] Yevgeniy Dodis and Sanjeev Khanna. Designing networks with bounded pairwise distance. In *Proc. 31st Annu. ACM Symp. on the Theory of Computing (STOC'99)*, pages 750–759, 1999.

[DK11] Michael Dinitz and Robert Krauthgamer. Directed spanners via flow-based linear programs. In *Proc. 43rd ACM Symposium on Theory of Computing (STOC'11)*, pages 323–332, 2011.

[DL16] H. Gökalp Demirci and Shi Li. Constant approximation for capacitated k-median with $(1+\epsilon)$-capacity violation. In *Proc. 43rd International Colloquium on Automata, Languages, and Programming (ICALP'16)*, pages 73:1–73:14, 2016.

[DT09] Jean Daligault and Stéphan Thomassé. On finding directed trees with many leaves. In *Proc. 4th International Workshop Parameterized and Exact Computation (IWPEC'09)*, pages 86–97, 2009.

[DV10] Matthew Drescher and Adrian Vetta. An approximation algorithm for the maximum leaf spanning arborescence problem. *ACM Trans. Algorithms*, 6(3):1–18, 2010.

[EHKP15] Alon Efrat, Yifan Hu, Stephen G. Kobourov, and Sergey Pupyrev. Mapsets: Visualizing embedded and clustered graphs. *J. Graph Algorithms Appl.*, 19(2):571–593, 2015.

[EMPR16] Alina Ene, Matthias Mnich, Marcin Pilipczuk, and Andrej Risteski. On routing disjoint paths in bounded treewidth graphs. In *Proc. 15th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT'16)*, pages 15:1–15:15, 2016.

[EN16] Alina Ene and Huy L. Nguyen. Constrained submodular maximization: Beyond 1/e. In *Proc. 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS'16)*, pages 248–257, 2016.

[FKN12] Moran Feldman, Guy Kortsarz, and Zeev Nutov. Improved approximation algorithms for directed steiner forest. *J. Comput. Syst. Sci.*, 78(1):279–292, 2012.

[Fre77] L.C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.

[Fre91] Greg N. Frederickson. Planar graph decomposition and all pairs shortest paths. *J. ACM*, 38(1):162–204, 1991.

[GH03] Luisa Gargano and Mikael Hammar. There are spanning spiders in dense graphs (and we know how to find them). In *Proc. 30th Int. Colloq. Automata, Lang. Progr. (ICALP'03)*, pages 802–816, 2003.

[GHSV02] Luisa Gargano, Pavol Hell, Ladislav Stacho, and Ugo Vaccaro. Spanning trees with bounded number of branch vertices. In *Proc. 29th Int. Colloq. Automata, Lang. Program. (ICALP'02)*, pages 355–365, 2002.

[GK99] Sudipto Guha and Samir Khuller. Greedy strikes back: Improved facility location algorithms. *J. Algorithms*, 31(1):228–248, 1999.

[GLN01] Joachim Gudmundsson, Christos Levcopoulos, and Giri Narasimhan. Approximating a minimum manhattan network. *Nord. J. Comput.*, 8(2):219–232, 2001.

[GVY97] Naveen Garg, Vijay V. Vazirani, and Mihalis Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18(1):3–20, 1997.

[HS85] Dorit S. Hochbaum and David B. Shmoys. A best possible heuristic for the k-center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985.

[KP97] Guy Kortsarz and David Peleg. Approximating shallow-light trees. In *Proc. 8th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'97)*, pages 103–110, 1997.

[KST09] Ariel Kulik, Hadas Shachnai, and Tami Tamir. Maximizing submodular set functions subject to multiple linear constraints. In *Proc. 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'09)*, pages 545–554, 2009.

[Kum12] Amit Kumar. Constant factor approximation algorithm for the knapsack median problem. In *Proc. 23rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'12)*, pages 824–832, 2012.

[Li11] Shi Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. In *Proc. 38th International Colloquium on Automata, Languages and Programming (ICALP'11)*, pages 77–88, 2011.

[LR98] Hsueh-I Lu and R. Ravi. Approximating maximum leaf spanning trees in almost linear time. *J. Algorithms*, 29(1):132–141, 1998.

[LR00] Bing Lu and Lu Ruan. Polynomial time approximation scheme for the rectilinear Steiner arborescence problem. *J. Comb. Optim.*, 4(3):357–363, 2000.

[LS13] Shi Li and Ola Svensson. Approximating k-median via pseudo-approximation. In *Proc. 45th Symposium on Theory of Computing Conference (STOC'13)*, pages 901–910, 2013.

[LSS12] Retsef Levi, David B. Shmoys, and Chaitanya Swamy. LP-based approximation algorithms for capacitated facility location. *Math. Program.*, 131(1-2):365–379, 2012.

[MR09] Nabil H. Mustafa and Saurabh Ray. PTAS for geometric hitting set problems via local search. In *Proc. 25th ACM Symposium on Computational Geometry (SoCG'09)*, pages 17–22, 2009.

[NWF78] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Math. Program.*, 14(1):265–294, 1978.

[PS05] Elena Prieto and Christian Sloper. Reducing to independent set structure – the case of k-internal spanning tree. *Nord. J. Comput.*, 12(3):308–318, 2005.

[PTW01] Martin Pál, Éva Tardos, and Tom Wexler. Facility location with nonuniform hard capacities. In *Proc. 42nd Annual Symposium on Foundations of Computer Science (FOCS'01)*, pages 329–338, 2001.

[RT87] Prabhakar Raghavan and Clark D. Thompson. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374, 1987.

[Sal09] Gábor Salamon. Approximating the maximum internal spanning tree problem. *Theoretical Comput. Sci.*, 410(50):5273–5284, 2009.

[Sal10] Gábor Salamon. *Degree-Based Spanning Tree Optimization.* PhD thesis, Department of Computer Science and Information Theory, Budapest University of Technology and Economics, 2010.

[SO98] Roberto Solis-Oba. 2-approximation algorithm for finding a spanning tree with maximum number of leaves. In *Proc. 6th Annu. European Symp. Algorithms (ESA'98)*, pages 441–452, 1998.

[Svi04] Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Let.*, 32(1):41–43, 2004.

[Swa16] Chaitanya Swamy. Improved approximation algorithms for matroid and knapsack median problems and applications. *ACM Trans. Algorithms*, 12(4):49, 2016.

[VCZ11] Jan Vondrák, Chandra Chekuri, and Rico Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. In *Proc. 43rd ACM Symposium on Theory of Computing (STOC'11)*, pages 783–792, 2011.