Approximation Algorithms for Generalized TSP in Grid Clusters

Michael Khachay[1,2,3] and Katherine Neznakhina[1,2]

[1] Krasovskii Institute of Mathematics and Mechanics,
[2] Ural Federal University, Ekaterinburg, Russia
[3] Omsk State Technical University, Omsk, Russia
mkhachay@imm.uran.ru eneznakhina@yandex.ru

Abstract. The Generalized Traveling Salesman Problem (GTSP) is a generalization of the well known Traveling Salesman Problem (TSP), where along with a weighted graph $G = (V, E, w)$ we are given by a partition of its node set $V = V_1 \cup \ldots \cup V_k$ into disjunctive subsets or *clusters*. The goal is to find a minimum cost cycle such that each cluster is hit by exactly one node of this cycle. We consider a geometric setting of the GTSP, in which a partition is specified by cells of the integer 1×1 grid (on the Euclidean plane). Even in this special setting, the GTSP remains intractable enclosing the classic Euclidean TSP on the plane. Recently, it was shown that this problem has $(1.5 + 8\sqrt{2} + \varepsilon)$-approximation algorithm with complexity bound depending on n and k polynomially, where k is the number of clusters. We propose three approximation algorithms for the Euclidean GTSP on grid clusters. For any fixed k, all proposed algorithms are PTASs. Time complexities of the first two remain polynomial for $k = O(\log n)$ while the last one is a PTAS when $k = n - O(\log n)$.

Keywords: Generalized traveling salesman, Grid clusters, Approximation algorithm

1 Introduction

The Generalized Traveling Salesman Problem (GTSP) is an extension of the well known Traveling Salesman Problem (TSP). The main difference between these problems is that unlike the TSP, in the GTSP, it is required to find a shortest tour visiting all specified disjunctive subsets or *clusters* of the given node set. An instance of the GTSP is given by an undirected edge-weighted graph $G = (V, E, w)$, whose node set is partitioned into k clusters V_i, $i = 1, \ldots, k$. The goal is to find a minimum weight cycle hitting all the V_i, i.e. visiting one point from each cluster exactly. It is easy to see that the classic TSP is a special case of the GTSP, for which every cluster consists of a single node. Therefore, GTSP is strongly NP-hard even in the Euclidean plane.

To the best of our knowledge, the GTSP was introduced in the late 1960s in relation to the optimal sequencing of computer files [11] and the scheduling of clients through

In: A. Kononov et al. (eds.): DOOR 2016, Vladivostok, Russia, published at http://ceur-ws.org

welfare agencies [17]. Another practical application closely related to GTSP is about the optimal routing of vans used to empty post boxes in an urban setting described by Bovet in [4]. They show that, by means of statistical analysis for the mail traffic, it is possible to determine (sub)optimal locations for post boxes in a city in order to provide a good service for customers. At the first stage of analysis, these locations need not be specified exactly. In particular, if it is decided to locate a post box at a crossroads, it is not necessary to specify on which side of the street the post box is to be located. In [4], it is proposed to model such an uncertainty by specifying the *cloud* of possible locations. The precise locations can be determined at the second stage, simultaneously with establishing the routes for the postal vans. This final problem is a kind of the GTSP.

Although for any nonnegative weighting function w and for any fixed number of clusters k, the GTSP as an exact polynomial time algorithm [8], any time when k is a part of the input, the problem is strongly NP-hard, even in the Euclidean plane.

There exist many approaches to finding of optimal or approximate solutions of the GTSP, e.g. genetic algorithms [3, 9, 16], ant colony heuristic [12], algorithms taking into account additional precedence constraints [19, 5]. The most intuitive approach is to reduce the considered instance of GTSP to some appropriate instance of the ordinary TSP. According to Laporte et al. [15], there exists a cost-preserving reduction from the GTSP to the corresponding instance of the asymmetric TSP. Therefore, researchers can apply the diversity of algorithms and solvers developed for the classic TSP. Unfortunately, the resulting TSP does not inherit some useful structural features of the initial problem, e.g. TSP assigned to the Euclidean GTSP is not even a metric one, so, to approximate this problem we could not use efficient algorithms like well known Christofides 3/2-approximation algorithm [6] or Aroras PTAS [1] for the metric or Euclidean TSP, respectively.

First algorithms [11, 17, 18] for the Euclidean GTSP were based on dynamic programming. In [14], Laporte and Nobert described an integer linear programming approach suitable for Euclidean or non-Euclidean problems.

Recently [2], it was shown that the GTSP has $(1.5 + 8\sqrt{2} + \epsilon)$-approximation algorithm with complexity bound depending polynomially on n and k.

In the paper, we present three approximation schemes for the Euclidean GTSP in k Grid Clusters (EGTSP-k-GC) in the plane. For any fixed k, all schemes find $(1+\varepsilon)$-approximate solutions of the analysed problem in polynomial time. Furthermore, the first two are LTAS in this case and remain PTAS, when $k = O(\log n)$, while the last one remains PTAS for $k = n - O(\log n)$.

The rest of the paper is organised as follows. In Section 2, we remind a formal statement of the EGTSP-k-GC and introduce some necessary definitions and notation. In Section 3, we present our first two approximation schemes focused to the case of slowly growing $k = k(n)$. In particular, we present our extension of the famous PTAS developed by S.Arora for the Euclidean TSP to the case of EGTSP-k-GC. Further, in Section 4 we consider the case of fast growing $k = k(n)$ and propose a PTAS for this case based on Arora's scheme as well. Finally, in Section 5 we summarize the results obtained and give a short overview of future work.

2 Problem statement

We consider the following combinatorial optimization problem. For an undirected edge-weighted graph $G = (V, E, w)$, whose vertex set is partitioned into k clusters V_i, $i = 1, \ldots, k$, Generalized Traveling Salesman Problem (GTSP) is required to find a minimum weight cycle visiting one point from each cluster exactly.

We consider the Euclidian Generalized Traveling Salesman Problem in k Grid Clusters, EGTSP-k-GC for short. In this special case of the GTSP, an undirected edge-weighted graph $G = (V, E, w)$ is given where the set of vertices V correspond to a set of points in the planar integer grid. Every nonempty 1×1 cell of the grid forms a cluster. The weight of the edge between two vertices is given by their Euclidean distance. In Fig. 1, we present an instance of the EGTSP-k-GC for $k = 6$.

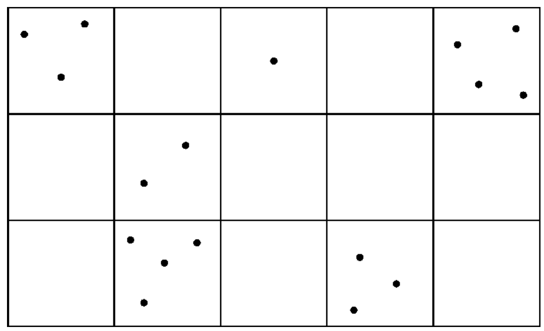

Fig. 1. An EGTSP-k-GC instance for $k = 6$

3 Schemes for the case of slowly growing *k*

For this special setting, we propose two approximation schemes. The former scheme is based on the classic Held-Karp dynamic programming algorithm used to finding exact solutions for a huge collection of the auxiliary instances of the Euclidean TSP. The latter one extends the Arora's PTAS to the case of the EGTSP-k-GC.

3.1 Approximation scheme based on dynamic programming

We start with a very simple algorithm. The main idea is to combine the exhaustive search for the suboptimal hitting sets for the set of clusters and subsequent dynamic programming-based estimation of their approximability.

To describe our scheme (see Algorithm 1), we introduce an auxiliary parameter t and partition all nonempty cells of the given grid into t^2 equal subcells of size $1/t$. Further, for any nonempty subcell obtained, we substitute all the nodes (of the graph G) belonging to it by the center of this subcell. This rounding procedure replaces each cluster by at most t^2 centers of the containing subcells. Therefore, there are at most t^{2k} ways to take k such centers as representatives of the given k clusters. Next, for any such a hitting set (containing k subcell centers exactly) we apply the standard Held-Karp dynamic programming technique [10] to find the exact solution of the corresponding TSP in time of $O(k^2 2^k)$. Therefore the overall time complexity is $O(t^{2k} 2^k k^2) + O(n)$.

Algorithm 1 Scheme based on DP

Input: a given instance of the Euclidean GTSP on k grid clusters and a required accuracy level ε.

Output: a $(1+\varepsilon)$-approximate solution.

1: partition all k nonempty cells of the given grid into t^2 smaller subcells; the value t will be specified later;
2: to each j-th cell assign a finite set C_j consisting of centers of nonempty subcells;
3: **for all** $(c_1, \ldots, c_k) \in C_1 \times \ldots \times C_k$ **do**
4:     using dynamic programming find an exact solution $S(c_1, \ldots, c_k)$ of the corresponding TSP instance;
5: **end for**
6: output the cheapest solution $S(c_1, \ldots, c_k)$.

Our accuracy bound for Algorithm 1 is based on the fact that, for any $v \in V$, the distance between the node v and the nearest subcell center is at most $\sqrt{2}/(2t)$. Consider an arbitrary optimal solution of the initial GTSP-k-GC instance. The accumulated error caused by substitution of the initial nodes by the nearest centers does not exceed $k\sqrt{2}/t$.

To estimate k in terms of optimum of the initial GTSP, we use a recent approximation result for another combinatorial optimization problem defined on clusters, Generalized Minimum Spanning Tree Problem (GMSTP). By the way, unlike to well known classic MSTP, which can be solved to optimality in polynomial time in the most general setting, GMSTP is NP-hard even in the Euclidean plane, where clusters are defined by cells of the unit grid. Nevertheless, in this case, the following claim is true.

Theorem 1 ([2]). *Let OPT_{GMSTP} be an optimum value of an instance of the Euclidean GMSTP on k grid clusters, then $k \leq 4OPT_{GMSTP} + 4$.*

Since any Hamiltonian cycle can be reduced to the corresponding spanning tree by excluding an arbitrary edge, therefore $OPT_{GTSP} \geq OPT_{GMSTP}$ any time when the weight function is nonnegative (which is true for the considered Euclidean case). Therefore, for the Euclidean GTSP, the same assertion is valid.

Corollary 1. *Let $OPT_{GTSP\text{-}k\text{-}GC}$ be the optimum value of an instance of the Euclidean GTSP on k grid clusters, then $k \leq 4OPT_{GTSP\text{-}k\text{-}GC} + 4$.*

So, for any $k > 4$ and $\varepsilon > 0$, taking a value of t such that

$$\frac{k\sqrt{2}}{t} \leq \frac{k-4}{4}\varepsilon \leq \varepsilon OPT_{\text{GTSP-}k\text{-GC}},$$

i.e.

$$t \geq \frac{4\sqrt{2}k}{(k-4)\varepsilon} = \frac{4\sqrt{2}}{\varepsilon}\left(1 + \frac{4}{k-4}\right) \geq \frac{20\sqrt{2}}{\varepsilon},$$

we guarantee that our accumulated error does not exceed $\varepsilon OPT_{\text{GTSP-}k\text{-GC}}$. It should be noticed that asymptotically we can obtain the same result even for $t \geq 4\sqrt{2}k/((k-4)\varepsilon)$ as $k \to \infty$. Hence, we proved the following theorem.

Theorem 2. *For any $\varepsilon > 0$ and any fixed $k > 4$ there exists an algorithm, which finds an $(1+\varepsilon)$-approximate solution of the GTSP on k grid clusters in time of $O(k^2(O(1/\varepsilon))^{2k}) + O(n)$.*

Corollary 2. *1. For any fixed number $k > 4$ and any $\varepsilon > 0$, Algorithm 1 finds an $(1+\varepsilon)$-approximate solution of the Euclidean GTSP on k clusters in a linear time with delay depending on ε.*
2. For the Euclidean GTSP on $k = O(\log n)$ clusters Algorithm 1 is a PTAS with time complexity of $O((\log n)^2 n^{O(\log(1/\varepsilon)))})$.

3.2 Extended Arora's scheme

We proceed with a scheme extending the famous PTAS proposed by S.Arora in [1] for the Euclidean TSP on the plane. Hereinafter we suppose that $k > 4$.

Similarly to Arora's PTAS, the main idea of the proposed approximation scheme is based on randomized recursive partitioning of the axis-aligned bounding box of the given instance into smaller squares and successive searching for the minimum weight closed tour subject to the following constraints:

(i) any cluster V_i is visited at once;
(ii) between-node segments of the route are continuous piece-wise linear curves crossing the borders of all squares only in predefined points called *portals*;
(iii) locations of the portals and the maximum count of crossings for each border-line of the squares depend on the given accuracy ε.

Following the main idea of the Arora's PTAS, we show that, to approximate well the EGTSP-k-GC in the plane, it is sufficient to design an efficient approximation algorithm for the special case of this problem, which is called a *well-rounded* EGTSP-k-GC. In particular, any PTAS for the well-rounded Euclidean EGTSP-k-GC induces the corresponding PTAS for the general case with the same complexity bound.

We call an instance of the EGTSP-k-GC *well-rounded* if

(i) where exists $L' = O(k)$ such that, for any node $v_i = [x_i, y_i]$ of the input graph G, its coordinates $x_i, y_i \in \{0, \ldots, L'\}$;
(ii) for any $u \neq v \in V$, $w(\{u, v\}) \geq 4$.

Indeed, consider an arbitrary instance of the EGTSP-k-GC and assign to in the corresponding well-rounded instance. Denoting the maximum distance between cluster cells by D, we obtain[1] that

$$OPT = OPT_{\text{GTSP-}k\text{-GC}} \geq 2D \geq 2$$

[1] for any $k > 4$

by the triangle inequality. Therefore, the size L of the minimal axis-aligned enclosing box for the given instance satisfies the equation

$$L \leq D + 2 \leq 1.5 \cdot OPT. \tag{1}$$

Next, we place an axis-aligned grid of granularity $L\varepsilon/(2k)$ and move any node to the nearest grid-point. Evidently, some nodes can map into the same grid-point, and, therefore, it is possible to obtain a well-rounded instance with a smaller number of nodes. Furthermore, the case when two or more different clusters share the same grid-point is possible as well. In this case, we assume that this grid-point hits all these clusters.

After such a transformation, every internode distance is changed at most by $L\varepsilon/k$. Also, the weight of any k-cycle can be changed at most by $L\varepsilon$.

By scaling the instance to the factor $8k/(L\varepsilon)$ and shifting the origin to the left-bottom corner of the scaled bounding box, we obtain the required well-rounded instance, since the size of this bounding box is $L' = O(k/\varepsilon) = O(k)$ for any fixed ε.

Further, consider an arbitrary k-cycle C in the initial instance and the corresponding cycle C' in the well-rounded one. Denoting their weights by W and W', respectively, we get

$$8k\,(W - L\varepsilon)\,/(L\varepsilon) \leq W' \leq 8k\,(W + L\varepsilon)\,/(L\varepsilon). \tag{2}$$

In particular, the RHS of equation (2) is valid for optimum values OPT and OPT' of the considered instances, i.e. $OPT' \leq 8k\,(OPT + L\varepsilon)\,/(L\varepsilon)$. Suppose $W' \leq (1+\varepsilon)OPT'$ for some $\varepsilon \in (0,1)$. Then,

$$8k\,(W - L\varepsilon)\,/(L\varepsilon) \leq W' \leq (1+\varepsilon)OPT' \leq 8k(1+\varepsilon)\,(OPT + L\varepsilon)\,/(L\varepsilon)$$

and, finally,

$$OPT \leq W \leq (1+\varepsilon)(OPT + L\varepsilon) + L\varepsilon \leq (1+4\varepsilon)OPT$$

due to (1). In such a way we have proved the following lemma.

Lemma 1. *Any PTAS for the well-rounded EGTSP-k-GC induces the appropriate PTAS for the EGTSP-k-GC with the same (up to the order) complexity bound.*

Let, further, $\mathcal{S}$ be the smallest axis-aligned square containing the instance of the EGTSP-k-GC. W.l.o.g. let the side-length L' of $\mathcal{S}$ be some power of two.

Following the approach introduced in [1], we construct a dissection of $\mathcal{S}$ into smaller squares using vertical and horizontal lines. These lines are crossing the coordinate axes in integer-coordinate points with a step of length 1. By construction, every smallest-size square contains at most one node of the given instance.

Further, we proceed with using of a 4-regular tree of a special kind known as a quadtree. In our case, the root of the tree is the bounding box $\mathcal{S}$. Each non-leaf square in the tree is partitioned into four equal child squares. This recursive partitioning stops on a square containing at most one node. By construction, the quadtree contains $O(k^2)$ leaves, $O(\log L') = O(\log k)$ levels and thus $O(k^2 \log k)$ squares in all.

The center point of the quadtree is the point of crossing of the inner edges of the squares with the side-length $L'/2$. We consider a shifted quadtree $T(a,b)$ with the center point $((L'/2 + a) \mod L', (L'/2 + b) \mod L')$, where $a, b \in \mathbb{N}_{L'}^0$ are constants. Any square of $T(a, b)$ with side-length less than L' is considered modulo L'.

To some parameter values $m, r \in \mathbb{N}$, and any node in the quadtree $T(a, b)$ [square S], we assign a regular partition of the border S consisting of $4(m+1)$ points including all the corners of S. We call this partition m-regular; its points are referred to portals. We consider the union of m-regular partitions of borders for all nodes of the quadtree $T(a, b)$.

Definition 1. *Let C be an arbitrary simple cycle in the graph G in the plane. The closed continuous piecewise linear route $l(C)$ is called an (m, r)-approximation of the cycle C if*

(i) $l(C)$ bends only at nodes of given graph and portals;
(ii) the nodes of G are visited by $l(C)$ in the same order as by C;
(iii) for any side of any node of $T(a, b)$, the route $l(C)$ crosses this side at portals and at most r times.

We use the following result from [13] claiming the existence of (m, r)-approximation with bounded weight.

Theorem 3. *(Structure Theorem) Let an instance of the well-rounded TSP in the plane be given by the graph G, let L be the side-length of the bounding box $\mathcal{S}$, and let constants $c > 1$ and $\eta \in (0, 1)$ be fixed. If the stochastic variables a and b are distributed uniformly in $\mathbb{N}_L$ and the parameters m and r are defined by the formulas*

$$m = \lceil 2s \log L \rceil, \quad r = s + 4, \quad \text{and } s = \lceil 36c/\eta \rceil .$$

Then, for an arbitrary simple cycle C of weight $W(C)$, with probability at least $1 - \eta$, there exists an (m, r)-approximation $l(C)$ of weight $W(l(C)) \leqslant (1 + 1/c)W(C)$.

Theorem 3 permit us to focus on finding the minimum cost (m, r)-approximation of an optimial solution for the well-rounded EGTSP-k-GC. For any shifted quadtree $T(a, b)$, denote such minimum cost (m, r)-approximation by $C(a, b)$. Similarly to [1], for each a and b, we can find $C(a, b)$ by dynamic programming except that, for each internal node of $T(a, b)$, the number of subtasks increases up to some factor depending on k. We present the overall scheme as Algorithm 2.

Following the Arora's argument [1] and the notices above, obtain the following theorem

Theorem 4. *For any fixed $\varepsilon \in (0, 1)$ and $k > 4$, Algorithm 2 finds a $(1+\varepsilon)$-approximate solution for the EGTSP-k-GC in time of*

$$2^{O(k)} k^4 (\log k)^{O(1/\varepsilon)} + O(n).$$

Corollary 3.
1. For any fixed $k > 4$, Algorithm 2 is a LTAS for the EGTSP-k-GC.
2. For $k = O(\log n)$, Algorithm 2 is PTAS for this problem with time complexity of $O(n(\log n)^4 (\log \log n)^{O(1/\varepsilon)})$.

Algorithm 2 Extended Arora's scheme

Input: a given instance of the Euclidean GTSP on k grid clusters and a required accuracy ε.
Output: a $(1+\varepsilon)$-approximate solution.

1: assign to the given instance the appropriate well-round instance enclosed in bouding box of size L';
2: **for all** $a, b \in \mathbb{N}^0_{L'}$ **do**
3: construct the shifted guadtree $T(a,b)$ and find $C(a,b)$ by dynamic procedure like proposed in [1] except that, for any internal node of the $T(a,b)$, the corresponding task along with conventional parameters depend on clusters $V_{i_1}, \ldots, V_{i_t}$ assigned to this node. Therefore, any child subtask of the Arora's DP produces up to 4^t copies according to all possible assignments of these clusters to this child (see Fig. 2);
4: **end for**
5: output the cheapest (m,r)-approximation $C(a,b)$.

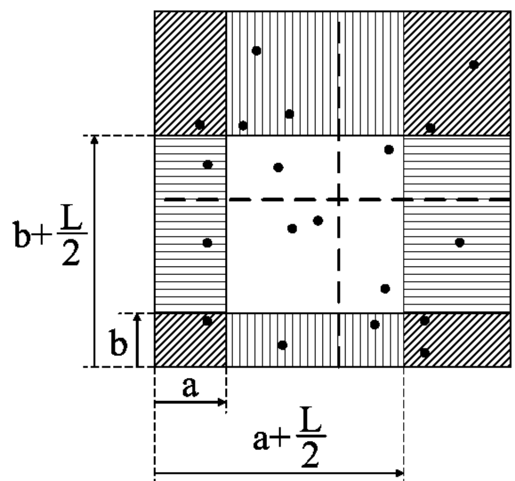

Fig. 2. Arrangement example of clusters among cells of the shifted quadtree

4 The case of fast growing k

Suppose now that k grows much faster. To this end we propose another approach (Algorithm 3) based on the famous S.Arora's PTAS [1] for the Euclidean TSP.

To prove the correctness of Algorithm 3, denote by t_i the number of nodes belonging to the i-th cluster. The number of ways to specify a TSP instance taking one node from each cluster is $t_1 \times \ldots \times t_k$. Maximizing this number subject to $\sum_{i=1}^{k} t_i = n$ we conclude that it does not exceed the value $(n/k)^k$ attained at point $t_i = n/k$.

Since, for any $\varepsilon > 0$, time complexity of the Arora's PTAS for k-node instance of the Euclidean TSP is $O(k^3 (\log k)^{O(\log(1/\varepsilon))})$, the time complexity of Algorithm 3 is

$$\left(\frac{n}{k}\right)^k k^3 (\log k)^{O(1/\varepsilon)}. \tag{3}$$

Algorithm 3 Scheme based on the classic Arora's PTAS

Input: a given instance of the Euclidean GTSP on k grid clusters and a required accuracy ε.
Output: a $(1+\varepsilon)$-approximate solution.

1: consider a partition $V_1 \dots V_k$ of the node set V of the given instance produced by the grid;
2: **for all** $(v_1, \dots, v_k) \in V_1 \times \dots \times V_k$ **do**
3: find an $(1+\varepsilon)$-approximate solution $S(v_1, \dots, v_k)$ of the corresponding TSP instance using Arora's PTAS;
4: **end for**
5: output the cheapest solution $S(c_1, \dots, c_k)$.

Evidently, for any fixed k, equation (3) depends on n polynomially and Algorithm 3 is a PTAS for the Euclidean GTSP on k grid clusters. To prove the same claim for k depending on n, we need to restrict $k = k(n)$ such that

$$\left(\frac{n}{k}\right)^k \le n^D \tag{4}$$

for some constant value $D > 0$. Suppose that $\frac{n-k(n)}{k(n)} \to 0$ as $n \to \infty$. Since, in this case,

$$\left(\frac{n}{k(n)}\right)^{k(n)} = \left(1 + \frac{n-k(n)}{k(n)}\right)^{k(n)} \le e^{n-k(n)},$$

the inequality $k(n) \ge n - D\log n$ implies equation (4).

Theorem 5. *1. For any fixed k and $\varepsilon > 0$, Algorithm 3 finds a $(1+\varepsilon)$-approximate solution of the Euclidean GTSP on k grid clusters in time of $n^k(\log k)^{O(1/\varepsilon)}$.*
2. If $k = n - D\log n$, then time complexity of Algorithm 3 is $n^{D+3}(\log n)^{O(1/\varepsilon)}$.

5 Conclusion

In this paper, we present three approximation schemes (Algorithms 1-3) for the Euclidean GTSP on k grid clusters. All proposed algorithms are PTAS for any fixed $k > 4$. Although none of the proposed algorithms has time complexity bound depending on n and k polynomially, we found two settings, for which such polynomial bounds exist. Actually, Algorithm 1 and 2 are PTAS for $k = O(\log n)$ while Algorhtm 3 is a PTAS for $k = n - O(\log n)$.

As for the future work, it would be interesting to answer the question *'Does the EGTSP-k-GC belong to PTAS or it is APX-complete?'*. Also, it would be interesting to extend the developed schemes to the case of clustered cycle cover problems (see, e.g. [7, 13]).

Acknowledgments. This research was supported by Russian Science Foundation, project no. 14-11-00109.

References

1. Arora, S.: Polynomial time approximation schemes for euclidean traveling salesman and other geometric problems. Journal of the ACM 45 (1998)
2. Bhattacharya, B., Ćustić, A., Rafiey, A., Rafiey, A., Sokol, V.: Combinatorial Optimization and Applications: 9th International Conference, COCOA 2015, Houston, TX, USA, December 18-20, 2015, Proceedings, chap. Approximation Algorithms for Generalized MST and TSP in Grid Clusters, pp. 110–125. LNCS, Springer International Publishing, Cham (2015)
3. Bontoux, B., Artigues, C., Feillet, D.: A memetic algorithm with a large neighborhood crossover operator for the generalized traveling salesman problem. Computers & Operations Research 37(11), 1844 – 1852 (2010)
4. Bovet, J.: Selective traveling salesman problem. In: Papers presented at the EURO VI Conference, Vienna (1983)
5. Chentsov, A., Khachay, M., Khachay, D.: An exact algorithm with linear complexity for a problem of visiting megalopolises. Proceedings of the Steklov Institute of Mathematics 295 (2016)
6. Christofides, N.: Worst-case analysis of a new heuristic for the traveling salesman problem. In: Symposium on New Directions and Recent Results in Algorithms and Complexity. p. 441 (1975)
7. Gimadi, E.K., Rykov, I.A.: On the asymptotic optimality of a solution of the euclidean problem of covering a graph by m nonadjacent cycles of maximum total weight. Doklady Mathematics 93(1), 117–120 (2016)
8. Grigoriev, A.: On fixed parameter tractability of GTSP. personal communication (2016)
9. Gutin, G., Karapetyan, D.: A memetic algorithm for the generalized traveling salesman problem 9(1), 47–60 (2010)
10. Held, M., Karp, R.M.: A dynamic programming approach to sequencing problems. In: Proceedings of the 1961 16th ACM National Meeting. pp. 71.201–71.204. ACM '61, ACM, New York, NY, USA (1961)
11. Henry-Labordere, A.: The record balancing problem: a dynamic programming solution of a generalized traveling salesman problem. In: RIBO, B-2. pp. 736–743 (1969)
12. Jun-man, K., Yi, Z.: Application of an improved ant colony optimization on generalized traveling salesman problem. Energy Procedia 17, Part A(0), 319 – 325 (2012)
13. Khachay, M., Neznakhina, K.: Approximability of the minimum-weight k-size cycle cover problem. Journal of Global Optimization (2015), http://dx.doi.org/10.1007/s10898-015-0391-3
14. Laporte, G., Nobert, Y.: Generalized travelling salesman problem through n sets of nodes: an integer programming approach. Informatica 21, 61–75 (1983)
15. Laporte, G., Mercure, H., Nobert, Y.: Generalized travelling salesman problem through n sets of nodes: the asymmetrical case. Discrete Applied Mathematics 18(2), 185 – 197 (1987)
16. Pop, P.C., Matei, O., Sabo, C.: Hybrid Metaheuristics: 7th International Workshop, HM 2010, Vienna, Austria, October 1-2, 2010. Proceedings, chap. A New Approach for Solving the Generalized Traveling Salesman Problem, pp. 62–72. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
17. Saksena, J.: Mathematical model for scheduling clients through welfare agencies. CORS Journal 8, 185–200 (1970)
18. Srivastava, S., Kumar, S., Garg, R., Sen, P.: Generalized travelling salesman problem through n sets of nodes. CORS Journal 7, 77–101 (1969)
19. Steiner, G.: On the complexity of dynamic programming for sequencing problems with precedence constraints. Annals of Operations Research 26(1), 103–123 (1990)