



HAL
open science

An Approximation Algorithm for the k -Connected Location Set Cover Problem with Color-Spanning Constraint

Yin Wang, Yinfeng Xu

► **To cite this version:**

Yin Wang, Yinfeng Xu. An Approximation Algorithm for the k -Connected Location Set Cover Problem with Color-Spanning Constraint. IFIP International Conference on Advances in Production Management Systems (APMS), Sep 2021, Nantes, France. pp.317-325, 10.1007/978-3-030-85906-0_36 . hal-04022156

HAL Id: hal-04022156

<https://inria.hal.science/hal-04022156>

Submitted on 9 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

An Approximation Algorithm for the k -Connected Location Set Cover Problem with Color-spanning Constraint

Yin Wang^{1,2}[0000-0002-2558-5004] and Yinfeng Xu^{1,2}

¹ School of Management, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China

² The State Key Lab for Manufacturing Systems Engineering, Xi'an, Shaanxi 710049, China

yinaywang@stu.xjtu.edu.cn

Abstract. Motivated by the need for a more sensitive server assignment strategy in supply-chain network management, our total cost comprises coverage area (i.e., disk) sizes and “moving” service modes that facilitate multiple and flexible demand fulfillment. Selection of k color-spanning centers to achieve cost minimization is the aim of our k -Connected Location Set Cover Problem with Color-spanning Constraint (k -CLSCPCC). The cost reflects the sum of the radii of the color-spanning disks plus the cost of connecting to disk regions. The farthest-color Voronoi diagram(FCVD) helps to assign an individual radius to each selected color-spanning center with aims to minimal cost. The main idea behind our greedy algorithm, which integrates the ideas of the classical minimum-power coverage problem and k -maximum coverage problem, is to minimize the measurable gap between the cost of connecting all nodes and the reduced cost of coverage with k disks. Our proposed algorithm can approximate a 3.368-factor solution within $O(n^2m \log m)$ running time, equal to time cost of generating FCVD, where n is the number of input nodes and m is the number of demand types.

Keywords: Color-spanning set · Approximation algorithm · Maximum coverage problems · farthest-color Voronoi diagram(FCVD) · Minimum spanning tree.

1 Introduction

Demand fulfillment, one of the most important components of complex supply-chain network management, is the basic aim for both location and routing problems. Demand in a cluster with requirement of a multi-type facility, represents same type or the possible locations of an object. This multiple-choice problem as “color-spanning set” should cover each given subset but not all given elements[1, 2, 3].

Facility location always involve long-term decisions and large investments in supply-chain management, and is difficult to correlate them with organizational

changes, particularly those involving diversification of products and varying demand. To fulfil multiple and varying demands in a wide variety of domains, location plans must account for both stable and dynamic demands that arise in a wide variety of domains.

A center with multiple service types is always appropriate to a long-term location plan for stable and multi-type demands. However, once the organization commits to a location, resource and location-related costs are often difficult to reduce. Dynamic dispatching can facilitate rapid response times and accurately adapt to variability in customer requirements. A location decision process with appropriate parameter settings should reduce operating costs. We are interested in establishing covered zones using fixed multi-type server and allocate regions via routing to make server assignments more sensitive to diversification of products and varying demand.

The problem discussed in this paper can be traced back to the location set covering problem (LSCP) [4]. The goal of LSCP is to meet the customer demand using the minimum number of facilities. The classical k -center location model aims to find optimal locations for k facilities while minimizing the coverage radii. In most cases, the classical LSCP is required to satisfy demands within a certain distance or standardized arrival time. The fixed coverage radius was relaxed by employing variable radii as the influence of facilities [5]. Some special constraints on facilities help to generate coverage regions. Considering demands with different types or uncertainty, the minimum diameter color-spanning set problem focused on finding the smallest color-spanning regions[1]. As its NP-hardness [6], designing an approximation is a natural choice.

The Farthest Color Voronoi Diagram (FCVD) is a color weighted version of the furthest point Voronoi diagram to find the furthest colored elements [7]. The minimum or maximum color-spanning objects can be find via the FCVD graph[2, 8, 9]. Especially, the approximated smallest disk is no more twice size than its optimal one in \mathbb{R}^2 within time complexity of $O(n \log n)$ based on generated FCVD graph[3]. The time cost of FCVD construction was reduce from $\Omega(mn)$ [7] to $O(n^2 \alpha(m) \log m)$, where m is the number of given color sets[10]. Under the condition of $m \leq \frac{n}{2}$, its performance was improved to $O(n^2 \log n)$ [11, 12].

The color-spanning problems focus on finding k objects whose union could cover a fixed number of points for each given color[6]. The connectivity between objects is approximated to be bounded by the minimum color-spanning set (MCSS) [13].

For node set without colors, a constraint specifying an equal maximum coverage distance is a classical setting for the connected set cover location problem. The coverage tour problem (MCTP) always defines this influence as coverage based on the lengths of paths and trees [14]. MCTP aims to generate a tour with exactly k elements selected from a ground set. An algorithm for a generalized version of MCLP improved the performance based on a tree allowing individual radii [15]. Focusing on minimization of total radius cost, the minimum sensor coverage problem and the minimum-power coverage (MPC) problem de-

fine a “sensing area” as a disk centered at one of sensors , and finds the minimum radius that covers the required area of targets.

In our variations, the connective costs arise from locating the facilities, as well as transportation costs for distributing the remaining nodes. Differing from previous studies, this paper introduces the k-Connected Location Set Covering Problem with Color-spanning Constraint(k-CLSCPCC), which seeks to find a selection set of exact k color-spanning disks such that the total cost is as less as possible. The total cost considered in our study is motivated by the need for a more sensitive server assignment strategy and arises from coverage area sizes and “moving” service modes that allow stable and flexible demand fulfillment.

The objective value of the MPC problem were within 2 times the length of a generated MST [16]. A minimum spanning tree (MST) is a natural choice for ensuring that facilities and uncovered nodes can connect with each other. We first recall how the MST plays an important role in assigning the node connective weights by ranges from the MPC problem. The main idea behind our strategy is to minimize the measurable gap between the cost of connecting all nodes and the reduced cost of coverage with a specified number of disks (center/radius combinations).

This is the background and related works of our study. Our problem description and approximation design were then emphasized. The remainder of this paper is organized as follows. Our problem description and approximation design were then emphasized in Section 2 and Section 3. Section 4 presents our main results relating to the approximation ratio and overall complexity. At last, section 5 contains our concluding thoughts and scope for future work.

2 Problem Definition

Consider a finite set of potential facility locations that are available; however, without loss of generality, we assume that the demand points and the potential locations are a set of nodes $V(|V| = n)$ of an all-connected network G with shortest Euclidean distances $dist_{xy}, \forall v_x, v_y \in V$. Any pair of nodes (i, j) and (i', j') cannot have the same weights to ensure the uniqueness of a generated MST[17].

Given integer $1 \leq k \leq n$ as number of selected centers. The selection $D = \{o_1, o_2, \dots, o_k\}$ with k sites is our decision variable, and generate the assigned radii as $R(D) = \{r(o_1), r(o_2), \dots, r(o_k)\}$ under the “color-spanning” constraint.

Color-spanning constraint A finite family $\mathcal{P} = \{P_1, P_2, \dots, P_j, \dots, P_m\}$ is m distinct colored sets for all given nodes V in \mathbb{R}^2 . Each node only belongs to one of m colored set and all nodes in each set $P_j \in \mathcal{P}$ have the same color. Our selected regions with center o_i and range r_i should cover all types in P with minimum distance to each set $P_j \in P$, i.e. $dist(o_i, P_j) = \min_{x \in P_j} dist(o_i, x) \leq r_i$. Conceptually, we define any range of each selected region $R_i \in R(D)$ as $r_i = \max_{P_j \subset \mathcal{P}} \min_{x \in P_j} dist(o_i, x)$, and generated a set of subsets $S(D) = \{s_i\} = \{s(o_1), \dots, s(o_k)\}$.

Each selected color-spanning region $s_i \in S(D)$ contains all types of colors by the smallest enclosing disk of radius r_i . Given integer k as size of selection, let $Cov(D) = \bigcup_i^k s(o_i)$ be the nodes covered by color-spanning sets and rest node set $B = D \cup Cov(D)$.

This paper focus on achieving “accessibility”. All demand points should be covered by color-spanning disks regions or through reallocation with minimum total covering cost $F(D)$. This study addresses the total cost arising from two defined service types: centers providing coverage with cost as the color-spanning radius $r \in R(D)$, and remaining uncovered nodes whose connective cost is the “moving” length from paths or subtrees to covered elements. Thus, our objective function $F(D)$ is the cost obtained by summation of coverage radius $r \in R(D)$ and the cost $Cost(B)$ of connecting centers or uncovered parts to covered regions. We have $F(D) = \sum^k r(i) + Cost(B)$, where $r(i)$ is assigned color-spanning radius of selected cite $o_i \in D$, and $Cost(B)$ is allocation cost depending on the connective cost or accessibility cost between uncovered node set B and covered elements $Cov(D)$.

3 Proposed Algorithm

3.1 Algorithm Design Notations

After generating the given n nodes with all edge weights for an all-connected undirected graph $G = (V, E)$, our approximation algorithm performs three main procedures: The color-spanning set generation, the MPC procedure and a deducting procedure based on the greedy strategy of the Max-k-cover problem.

The Voronoi diagram of high order is a common method to solving color-spanning set [3, 8]. The farthest color Voronoi diagram (FCVD) is a *max – min* diagram generating the bounds on maximal distance to each type[?]. Our approximation algorithm decides the color-spanning radius sequence $CSR(V) = \{r(v_1), r(v_2), \dots, r(v_n)\}$ for all points by FCVD at first.

The result of the MCP problem can provide a measurable approximation processing to assign all nodes weights. The MST generation helps to solve the MCP problem [15, 16]. With ranges $W(V) = \{w(v_1), w(v_2), \dots, w(v_n)\}$ of the MCP problem, all nodes can be connected, where w_i is range of cite v_i .

After these two steps, each set s_i captures the nodes in disk $\odot_i(v_i, r_i)$ centered on v_i with radius r_i , but $v_i \notin s_i$. Once two element $v_x, v_y \in V$ was selected to center set D , $dist(v_x, v_y) = w_x$ or $dist(v_x, v_y) = w_y$, these two elements set need w_x or w_y ranges to connected with each other. Then weight of each s_i is $H(i) = \sum_{x \in s_i} w_x$. The final Max-k-cover subproblem is seeking the “best” collection to “peel off” the weights of nodes covered by k disks as much as possible. When computing the intersection of two or more sets, our only concern is determining whether these sets have any common union values arising from the elements such as $\Delta_t = f(\bigcup_i^{t-1} s_{t-1} \cup s_t) - f(\bigcup_i^{t-1} s_{t-1})$. The first one of center sequence sorted by $f(s_i)$ in nondecreasing order is the initial solution D_1 . The rest $k - 1$ ones should ensure that each addition Δ_t at step t is maximal. Let $Cov_{t-1} = \bigcup_i^{t-1} s(t-1)$ with initial solution $Cov_0 = \emptyset$.

3.2 Approximation Algorithm Design

We provide a step-by-step description of our algorithm, followed by details on constraint identification.

Algorithm 1 Approximation Algorithm

```

1: Generate the FCVD for given  $G = (V, E)$  with  $m$  colors
2: for each  $v_i \in V$  do
3:    $r(v_i) \leftarrow \max_{P_j \subset \mathcal{P}} \min_{x \in P_j} \text{dist}(v_i, x)$ ;
4: end for
5: Save  $CSR(V) = \{r(v_1), r(v_2), \dots, r(v_n)\}$ 
6: Compute the minimum spanning tree (MST)  $G_{MST} = (V, E^{MST})$ 
7:  $\forall w(u) \leftarrow 0$ 
8:  $S_0 = \emptyset$ 
9: for each  $v_i \in V$  do
10:   $w(v_i) \leftarrow \max_{e(i,x) \in E^{MST}} w(v_i, v_x)$ ;
11: end for
12:  $Solution(MPC) \leftarrow \sum_{v_i \in V} w(v_i)$ 
13: for each  $v_i \in V$  do
14:  if  $\text{dist}(v_i, v_x) \leq r(v_i)$  then
15:     $s(v_i) \leftarrow v_x$ ;
16:  end if
17:  Save  $s(v_i)$ , short as  $s(i)$ , and count  $|s(v_i)|$ ;
18: end for
19: for  $v_i \in V$  do
20:   $f(v_i) \leftarrow \sum_{v_x \in s(i)}^{|s(i)|} w(v_x) - w(v_i)$ ;
21: end for
22: Save sorted sequence  $V^I = \{v_1, v_2, \dots, v_i, \dots, v_n\}$  in nonincreasing order by value
     $f(v_i)$  as the covering set;
23: Save subsets to  $S = \{s(v_1), s(v_2), \dots, s(v_i), \dots, s(v_n)\}$ ;
24:  $D_0 \leftarrow \emptyset$ ,  $Cov(D_0) \leftarrow \emptyset$ ;
25: Select the first item with maximal  $f(v)$  in  $L = (V^I, S)$ 
26:  $r_1 \leftarrow r(v_1)$ , and  $D \leftarrow v_1$ 
27:  $S(D_1) \leftarrow s_1$ , and  $Cov(D_1) \leftarrow s_1$ 
28: for  $t = 2$  to  $k$  do
29:  choose  $v_t \in V$  to maximize  $\Delta_t f(\bigcup_i^{t-1} Cov_{t-1} \cup s_t)$ 
30:   $D_t \leftarrow D_{t-1} \cup \{v_t\}$ ,  $Cov(t) \leftarrow Cov(t-1) \cup \{s_t\}$ 
31: end for
32: return Selected center set  $D^A$  with color-spanning radii  $R^A(D) = \{r_1, \dots, r_k\}$ ;
33: return covered node subset  $S(D^A) = \{s_1, \dots, s_k\}$ ,  $Cov(D^A) = \bigcup_i^k s(D_i)$ , and the
    uncovered node set  $B = U \setminus (Cov(D) \cup D)$ ;
34: return  $F^A = Solution(MPC) - \max_{D \subseteq V} f(D) + \sum^k r(i)$ 

```

4 The Performance of our Algorithm

4.1 The Approximation Ratio

The classical and k -maximal cover problem are NP-hard. When objective function is under submodular and monotone condition, Feige et al. [18, 19] proved a

best-possible factor approximation ratio of $1 - e^{-1}$ by the greedy algorithm unless $NP = P$. Submodular is a very natural property of various functions. In set function $f : 2^V \mapsto \mathbb{R}$, f is submodular iff. $f(X \cup Y) + f(X \cap Y) \leq f(X) + f(Y)$ for any $X, Y \subseteq V$. The corollary below helps us to estimate the gap between our approximation value f^A and the optimal solution f^* of the Max-k-cover step.

Lemma 1. *The function in our Max-k-cover subproblem is submodular and monotone.*

Proof. Let D_t be the selected center by greedy strategy up to t iterations and $Cov(D_t)$ be the corresponding sets.

The $Cov(D_1)$ is the maximum coverage one in all $s_i \subseteq S$. Then the rest $k - 1$ selection is added by maximum added coverage value at each iteration. It is obvious that the improvement from $f(t)$ to $f(t + 1)$ is always positive and monotone decreasing or nonincreasing for any integer $t \in [1, k]$. This step shows serious monotonicity firstly.

Let $w(i) = \sum_{v_x \in s_i} p(v_x) - p(i)$ be the value covered by disk $\odot(o_i, r_i)$ centered on node o_i with radius r_i , and $Cov(D_k) = \{s_1 \cup \dots \cup s_k\} - D_k$ be union of each covered nodes by selected centers except centers themselves. Next proof of submodularity is based the definition above. As we know, our Max-k-cover subproblem aims to find k sets with maximal coverage value $f(D_k) = \sum_{v_x \in Cov(k)} p(v_x) - \sum_{v_i \in D_k} p(i)$, where $r(i)$ is the radius of color-spanning disk, and $p(i)$ is assigned weight to node $v_i \in V$ for connectivity based on of MCP subproblem.

It has an influence on the total coverage $f(D_t)$ that whether the new added set s_t in t -th iteration contains previous selected centers D_{t-1} . Based on this property, we divide two cases by whether $s_t \cap D_{t-1} \neq \emptyset$.

One case is $s_t \cap D_{t-1} = \emptyset, \forall t \in [2, k]$, when no selected centers is contained by new added coverage sets. The total coverage value would be summation. We have $f(A \cup B) = f(A) + f(B)$ for any $A, B \subseteq V$, as $f(A \cap B) = 0$. Another case exists $s_t \cap D_{t-1} \neq \emptyset$. We assume it appeared at the t -th iteration. Let $v_x \in D_{t-1}$ be the covered element by s_t . When element v_t added, function $f(t)$ would move out value $p(x)$ counted in $w(t)$. As $\forall p(x) \geq 1$, we have $f(D_{t-1} \cap X) < f(D_{t-1}) + f(X) - f(D_{t-1} \cup X)$, where X is any other set contained $v_x \in D_{t-1}$.

Combined these two cases, we can conclude that $f(A \cup X) + f(A \cap X) \leq f(A) + f(X)$ with $A, X \subset V$. Submodularity is proved based on its definition.

Theorem 1. *Our approximation algorithm can solve the k-CCSLPCC within a ratio of $3 + \frac{1}{e}$.*

Proof. There are three main parts for our algorithm.

The first FCVD graph for color-spanning constraints can find an approximation solution r_i^A no more twice than optimal one for $v_i \in V$. Note the case with ratio 2 as instance I_1 . Let $R^A(I)$ be the solution of our approximation algorithm for any instance I , and R^* be optimal solution. We have $\frac{R^A(I)}{R^*} \leq \frac{R^A(I_1)}{R^*(I_1)} \leq 2$.

Let $P(I_2) = Length(MST)$ is the solution under instance I_2 with gap 2 to our MCP solutions [16]. Thus, we have $\frac{Solution(MCP)}{OPT(MCP)} \leq \frac{Solution(MCP)}{P(I_2)} \leq 2$.

The Max-k-cover part is the last step of our approximation algorithm. Let $F(D)$ be our objective function and $f(D_i)$ be value function of the Max-k-cover value. Let $f(I_3)$ be the results in instance I_3 with the maximal gap between our greedy strategy and the optimal one f^* . As Corollary 1 proved, we claim that $\frac{f^A(I)}{f^*} \geq \frac{f^A(I_2)}{f^*(I_2)} \geq 1 - \frac{1}{e}$ [19].

We denote an instance I_{worst} arising the upper bound of gap between our approximation solution $A(I)$ and the optimal result $OPT(I)$ for our k-CLSCPCC. With the help of case I_{worst} , the approximation ratio $ratio(A)$ of our algorithm can be estimated as follow:

$$ratio(A) = \max \frac{A(I)}{OPT(I)} = \frac{Solution(MCP) - f^A(I) + R^A(I)}{OPT(I)} \quad (1)$$

$$\leq \frac{2P(I_2)}{OPT(I_{worst})} - \frac{f^A(I_3)}{OPT(I_{worst})} + \frac{R^A(I_1)}{OPT(I_{worst})} \quad (2)$$

$$\leq \frac{2P(I_2)}{OPT(I_2)} - \frac{f^A(I_3)}{f^*(I_3)} + \frac{R^A(I_1)}{R^*(I_1)} \leq 2 - (1 - \frac{1}{e}) + 2 = 3 + \frac{1}{e} < 3.368 \quad (3)$$

4.2 The Complexity

Theorem 2. *The computing time of our approximation algorithm costs no more than $O(n^2m \log m)$, where n is the input size of ground set, m is the number of colors.*

Proof. Our total running time is the longest time of our three main steps.

Firstly, the disk coverage region must satisfy the color-spanning constraint. Generation of FCVD costs $O(n^2m \log m)$ for given n nodes with m colors [10]. Based on FCVD graph, n nodes need at most n times $n \log n$ to get their diameter of color-spanning circle.

Secondly, Primal Algorithm helps generate the minimum spanning tree T_n on dense graph with timing of $O(n^2 \log n)$. The selection of the maximum weighted runs n times for all nodes on tree. Then, the total cost of second main step is $O(n^2 \log n) + O(n^2 \log n) = O(n^2 \log n)$.

The set operations help compute covered value $u_i = \sum_{v_j \in Cov(v_i, r_i)} p_j$ and n elements cost $n \times O(n)$ time. It costs $O(n^2)$ time to sort elements by value u_i . At most k times loops to finish the selection of rest D_i . Even based on the basic List and Array List data structure, all selections cost $O(kn^3)$ time at most.

All in all, as parameter m is much less than n and k is given integer, the total computing time is $T(n^2m \log m + n^2 \log n + kn^3 + n) = O(n^2m \log m)$, where n and m is the number of nodes and colors of in a given node set V .

5 Conclusions

Our k-CLSCPCC assigns an individual radius to each selected color-spanning center such that the total cost is minimized. A more sensitive server assignment is achieved using the nearest-first rule in disk coverage for stable performance

in high-density and multi-class scenarios. A moving service mode to fulfill the remaining demand arises rest part of cost except all k individual radius. In this study, we propose a $3 + \frac{1}{\epsilon}$ ratio algorithm for a variant connected location problem within $O(n^2 m \log m)$ running time.

Applying our method or adjusting our method on a wider range of networks can be considered in the future.

Bibliography

- [1] Ju, W., Fan, C., Luo, J., et al.: On some geometric problems of color-spanning sets. *Journal of Combinatorial Optimization* **26**(2), 266–283(2013).
- [2] Graf, T., Hinrichs, K.: Algorithms for proximity problems on colored point sets. In *Proc.5th Canad. Conf.Comput. Geom.*, Universität Münster. *Ange wandte Mathematik und Informatik*, 420–425 (1993).
- [3] Abellanas, M., Hurtado, F., Icking, C., et al.: Smallest color-spanning objects. In *Proceedings of the 9th Annual European Symposium on Algorithms 2001*, LNCS, vol. 2161, pp. 278–289. Springer, Berlin, Heidelberg (2001). https://doi.org/10.1007/3-540-44676-1_23
- [4] Toregas, C., Swain, R., ReVelle, C., et al.: The location of emergency service facilities. *Operations Research* **19**(6), 1363–1373 (1971).
- [5] Berman, O., Ingco, D. I., Odoni, A.: Improving the location of minimax facilities through network modification. *Networks* **24**(1), 31–41 (1994).
- [6] Banerjee S., Misra N., Nandy S.C.: Color Spanning Objects: Algorithms and Hardness Results. In: Govindarajan S., Maheshwari A. (eds) *Algorithms and Discrete Applied Mathematics, CALDAM 2016*, Lecture Notes in Computer Science, vol 9602, pp. 14–22. Springer, Cham. (2016).https://doi.org/10.1007/978-3-319-29221-2_4
- [7] Huttenlocher D. P., Kedem K., Sharir M.: The upper envelope of Voronoi surfaces and its applications. *Discrete Comput. Geom.* **9**(3), 267–291 (1993).
- [8] Fleischer, R., Xu, X.: Computing minimum diameter color-spanning sets. In *International Workshop on Frontiers in Algorithmics, FAW 2010*, LNCS, vol. 6213, pp. 285–292. Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/2F978-3-642-14553-7_27
- [9] Fleischer, R., Xu, X.: Computing minimum diameter color-spanning sets is hard. *Information Processing Letters* **111**(21-22), 1054–1056 (2011).
- [10] Abellanas, M., Hurtado, F., Icking, C., et al.: The farthest color voronoi diagram and related problems. In *Abstracts 17th European Workshop Comput. Geom.*, CG 2001, pp. 113–116. Freie, Universität Berlin (2001).
- [11] Eppstein, D., Overmars, M., Rote, G., et al.: (1992). Finding minimum area k-gons. *Discrete & Computational Geometry* **7**(1), 45–58 (1992).
- [12] Mantas, I., Papadopoulou, E., Sacristán, V., et al.: Farthest Color Voronoi Diagrams: Complexity and Algorithms. In *Latin American Symposium on Theoretical Informatics, LATIN 2020*, LNCS, vol. 12118, pp. 283–295. Springer, Cham. (2021). https://doi.org/10.1007/978-3-030-61792-9_23
- [13] Fan C., Luo J., Zhu B.(2013). Tight Approximation Bounds for Connectivity with a Color-Spanning Set. *ISAAC 2013*, 590-600. https://doi.org/10.1007/978-3-642-45030-3_55
- [14] Current, J. R., Schilling, D. A.: The median tour and maximal covering tour problems: Formulations and heuristics. *European Journal of Operational Research* **73**(1), 114–126 (1994).

- [15] Spoerhase, J., Wirth, H. C.: An $O(n(\log n)^2/\log \log n)$ algorithm for the single maximum coverage location or the $(1, X_p)$ -medianoid problem on trees. *Information Processing Letters* **109**(8), 391–394 (2009).
- [16] Chen, W. T., Huang, N. F.: The strongly connecting problem on multihop packet radio networks. *IEEE Transactions on Communications*, **37**(3), 293–295 (1989).
- [17] Sedgewick, R., Wayne F. K.: *Algorithms*. 4th edn. Addison-Wesley, Boston, USA (2011).
- [18] Nemhauser, G. L., Wolsey, L. A. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research* **3**(3), 177–188 (1978).
- [19] Feige, U.: A threshold of $\ln n$ for approximating set cover. *Journal of the ACM (JACM)* **45**(4), 634–652 (1998).