57º SBPO
SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL
Gramado • RS
DE 5 A 9 DE OUTUBRO DE 2025

NP-Completeness, Approximation Results and Mixed Integer Linear Programming Methods for the Minimum Labeling Spanning k-Forest Problem

Kennedy Corrêa
Instituto de Matemática e Estatística - Universidade de São Paulo
R. do Matão, 1010 - Butantã, São Paulo-SP, 05508-090
kennedycorrea2001@gmail.com

Manoel Campêlo
Departamento de Estatística e Matemática Aplicada - Universidade Federal do Ceará
Bloco 910, Campus do Pici - Fortaleza-CE, 60440-900
mcampelo@ufc.br

RESUMO

Um grafo colorido em arestas (GCA) é um grafo tal que cada aresta tem uma cor associada. Dado um GCA G, o problema da k-floresta geradora minimamente rotulada se consiste em encontrar uma floresta geradora F de G com número mínimo de cores e número de componentes limitado por k. Neste trabalho, um estudo de complexidade é realizado e um algoritmo aproximativo para o problema é proposto. Continuando em um estudo anterior sobre um modelo de programação matemática, propomos duas desigualdades válidas para reforçá-lo e analisamos experimentos computacionais.

PALAVRAS CHAVE. Florestas Geradoras, Algoritmo Aproximativo, Grafos Coloridos em Arestas.

Tópicos (Otimização Discreta, Teoria dos Grafos e Algoritmos Relacionados)

ABSTRACT

An edge-labeled graph (ELG) is a graph such that each edge has a label associated. Given an ELG G, the minimum labeling spanning k-forest problem consists in finding a spanning forest F of G with minimum number of labels and number of components bounded by k. In this work, a complexity study is realized and an approximation algorithm for the problem is proposed. Continuing on a previous study of a mathematical programming model, we propose two valid inequalities to reinforce it and analyze computational experiments.

KEYWORDS. Spanning Forests. Approximation Algorithms. Edge-labeled Graphs.

Paper topics (Discrete Optimization, Graph Theory and Related Algorithms)

1. Introduction

An edge-labeled graph (ELG) G consists of a tuple (V, E, L, ℓ) where V is the set of vertices, E is the set of edges, L is a set of labels/colors and $\ell : E \to L$ is a function that label each edge. Given an ELG G with number of components $c(G) \leq k$, the minimum spanning labeling $k-$forest problem (MSLkFP) consists in finding a spanning forest of G with at most k components and minimum number of labels. An example is depicted in Figure 1. It is also useful to represent G as (V_G, E_G, L_G, ℓ_G)

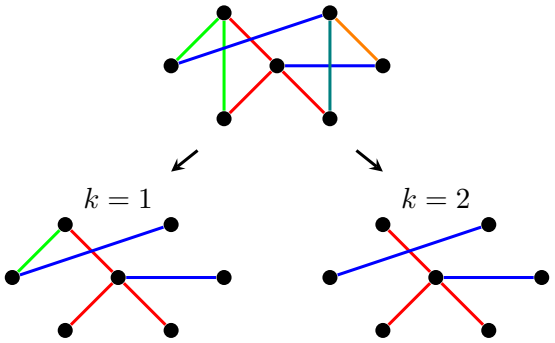


Figure 1: An edge-labeled graph with solutions of the MSLkFP for $k = 1$ and $k = 2$.

This problem was introduced in [Corrêa e Campêlo, 2024] where a flow-based mixed integer linear programming (MILP) formulation was proposed and the authors conducted computational experiments to evaluate the formulation.

2. Related Work

When $k = 1$, the MSLkFP is known as minimum spanning labeling tree problem (MSLTP), being first introduced in [Chang e Leu, 1997], where it was proved that the MLSTP is NP-Complete for complete graphs and it was proposed the first version of the Maximum Vertex Covering Algorithm (MVCA), a heuristic for solving the problem. In [Krumke e Wirth, 1998] it was proved that there is no α-approximation algorithm for the MSLTP for any constant α, unless P = NP. The authors also proposed a revision of the MVCA heuristic and proved that it possesses an approximation factor of $1 + 2ln(|V|)$.

The approximation factor of the MVCA was further proven to be smaller in [Wan et al., 2002], where it decreased to $1 + ln(|V|)$. Finally, in [Xiong et al., 2005], the approximation algorithm decreased to $H_b = \sum_{i=1}^{b} \frac{1}{i}$, where $b = \max_{l \in L} |\{e \in E : \ell(e) = l\}|$, the maximum frequency of a label. The following result was also proven.

Theorem 2.1 (Xiong et al. [2005])**.** *H_b is a tight approximation factor for the MVCA algorithm, that is, there exists an ELG G such that the value of the solution obtained by the algorithm on G is H_b times the value of the optimal solution.*

Spanning trees have been one of the fundamental topics in graph theory, the MSLTP has found applications in telecommunications [Chwatal et al., 2007] and has been extended to other important problems such as the minimum spanning labeling Steiner tree [Cerulli et al., 2006] or the budgeted labeled minimum spanning tree problem [Cerulli et al., 2024].

3. Computational Complexity

As already mentioned, the MSLkFP is NP-Complete for $k = 1$. It remains to answer if this result follows for every positive integer k

A certifier for MSLkFP is obtained by (i) counting the number of labels of the edges, (ii) checking for the existence of cycles, and (iii) verifying that the number of components is at most k. Since all of the these steps can be done in polynomial time, the problem is NP.

NP-hardness can be proven by reducing the MSLTP to the general MSLkFP by adjoining $k-1$ artificial vertices in a complete graph G such that each artificial vertex is adjacent to only one vertex of G.

Theorem 3.1. *For any positive integer k, the MSLkFP is NP-Complete in split graphs.*

The NP-completeness of the problem reveals the unlikeliness of the existence of a polynomial-time algorithm for solving the MSLkFP, hence, approaches of approximation algorithms and mathematical programming are useful.

4. Approximation Algorithm

In this section, a modification of the MVCA heuristic is proposed with the same approximation factor. The algorithm here called generalized MVCA is described in Algorithm 1. When $k = 1$, this algorithm is the same as the MVCA as written in [Xiong et al., 2005].

Algorithm 1: Generalized MVCA

input: A ELG G with vertices V, edges E and labels L and a positive integer k.

Let $C \leftarrow \emptyset$ the set of used labels;
Let $H = G[C]$ be the subgraph of G restricted with labels from C;
while *H has more then k components* **do**
  for all $i \in L - C$ **do**
    Determine the number of connected components when inserting all edges of label i in H;
  end
  Choose label i with the smallest number of components;
  $C \leftarrow C \cup \{i\}$;
  $H \leftarrow G[C]$;
end

Theorem 4.1. *The algorithm 1 has an approximation factor of H_b for the MSLkFP, where $b = \max_{l \in L} |\{e \in E : \ell(e) = l\}|$ and $H_b = \sum_{i=1}^{b} \frac{1}{i}$.*

The proof of this theorem is very similar to the proof of approximation factor for the MVCA given in [Xiong et al., 2005]. It consists of finding upper bounds for a number of successive linear programming models that describe the steps of the algorithm. The main difference is that the number of edges in the ongoing problem is $|V| - k$ as opposed of only $|V| - 1$ in the original paper.

Another important result is the fact that the approximation factor is tight for every positive integer k. For this, the concept of isomorphism is useful.

Definition 4.1. *The ELGs $G = (V, E, L, \ell)$ and $G' = (V', E', L', \ell')$ are isomorphic if there exist bijective functions $f : V \to V'$ and $g : L \to L'$ such that $uv \in E \iff f(u)f(v) \in E'$ and $\ell'(f(u)f(v)) = g(\ell(uv))$. When then say that G and G' are isomorphic.*

Given an ELG G and a positive integer k, then define the ELG G^* as the union of k ELGs isomorphic to G such that, for each pair of ELGs their vertices are disjoint as well as their labels (Figure 2). Lemma 4.2 is a intuitive result that tells the form of the solution on G^*.

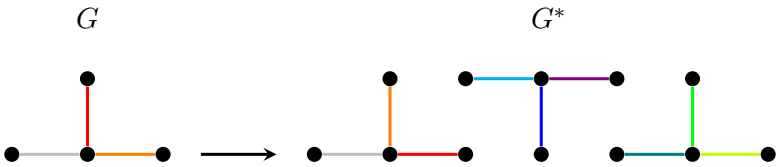


Figure 2: Extension of an ELG G to G^* for $k = 3$.

Lemma 4.2. *Suppose that the MSLTP in the ELG G has optimal solution A with value v. then an optimal solution for the MSLkFP in G^* is obtained by choosing a forest F such that each component of F is isomorphic to the tree A. The number of labels in such solution is kv.*

Lemma 4.2 allows the study of Algorithm 1 in the ELG G^* induced by the ELG G such that G is the graph of Theorem 2.1 because it must construct a spanning tree in each of its components in a way that the number of labels in each component is H_b, hence, Algorithm 1 returns a solution with value kH_b.

Theorem 4.3. *For any positive integer k, the approximation factor for algorithm 1 in solving the MSLkFP is tight, that is, exists a graph whose solution given by the algorithm has number of labels given by H_b times the number of labels of an optimal solution.*

5. Mathematical Programming Model

In [Corrêa e Campêlo, 2024], a mixed linear integer programming model for the MSLkFP is created by the use of the extended ELG (Figure 3). Such extension is used in [Figueredo, 2020] to solve the k-labeled spanning forest problem that consists in, given an ELG, minimizing the number of components while maintaining the number of labels bounded by k.

Definition 5.1. *Given an ELG $G = (V, E, L, \ell)$, The extended ELG $G' = (V', E', L', \ell')$ is defined such that $V' = V \cup \{s\}, s \notin V$, $E' = E \cup \{(s, v) : v \in V\}$, $L' = L \cup C$, $C = \{c_v : v \in V\}$ is a set of new distinct labels $C \cap L = \emptyset$, and ℓ' such that*

$$\ell'(e) = \begin{cases} \ell(e), & e \in E, \\ c_v, & e = (s, v), v \in V. \end{cases}$$

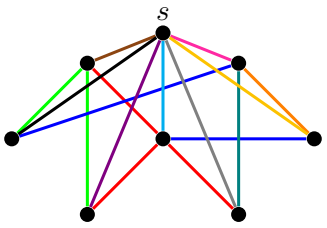


Figure 3: Extended ELG G' of the graph given in figure 1.

The main idea is that solving the MSLkFP in G is equivalent to finding a spanning tree A of G' such that $L_A = L^* \cup C^*$, where $L^* \subseteq L_G, C^* \subseteq C$, $|C^*| \leq k$ and $|L_A|$ is minimal (Figure 4).

The MILP is then based on sending a flow from the artificial vertex s to ensure connectivity. The model variables are $z_l \in \{0, 1\}$ is 1 if and only if the label $l \in L \cup C$ is used in the solution, $f_{ij} \geq 0$ is the amount of flow from the vertex i to the vertex j and M is a big-M factor.

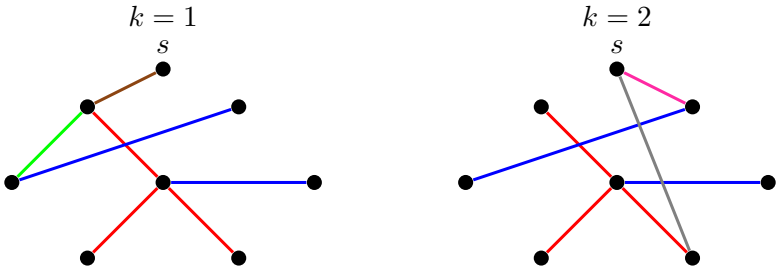


Figure 4: Equivalent solutions to the solutions in Figure 1 generated on the extended ELG.

$$(flow)\ \min \quad \sum_{l \in L} z_l \tag{1a}$$

$$\text{s.t.} \quad f_{sj} + \sum_{i:(i,j)\in E(G)} f_{ij} - \sum_{i:(i,j)\in E(G)} f_{ji} = 1 \qquad \forall j \in V_G, \tag{1b}$$

$$0 \leq f_{ij}, f_{ji} \leq M z_{\ell(ij)} \quad \forall\ (i,j) \in E_G, \tag{1c}$$

$$0 \leq f_{sj} \leq M z_{\ell(sj)} \quad \forall\ j \in V_G, \tag{1d}$$

$$\sum_{l \in C} z_l \leq k, \tag{1e}$$

$$z_l \in \{0,1\} \quad \forall\ l \in L \cup C \tag{1f}$$

5.1. Valid Inequalities

Two valid inequalities can be used to strengthen the model 1a-1f, these are Inequalities 2 and 3, both proposed in [da Silva, 2018] for solving the MSLTP.

The first inequality follows because, for each vertex $v \in V$, the neighborhood of v in G' must not be empty. Hence, there must be at least one label in the edges of the neighborhood.

$$\sum_{l \in K(\{v\})} z_l \geq 1, \quad \forall v \in V, \tag{2}$$

where $K(\{v\})$ is the set of labels of edges in the neighborhood of v.

For inequality 3, every spanning tree of G' must have at least $|V'| - 1 = |V|$ edges and each label chosen in a solution contributes with $|E'(l)|$ edges, where $E'(l)$ is the set of edges in G' with label l.

$$\sum_{l \in L \cup C} |E'(l)| z_l \geq |V|. \tag{3}$$

6. Computational Experiments

In this section it is discussed computational experiments for solving the MLSkFP with the methods described throughout Sections 4 and 5.

The experiments were executed in a computer with processor intel i3 - 12100, with 16 GB of RAM and operational system Ubuntu 22.04. The implementations were done in Python 3.10.12 with the CPLEX solver in version 22.1.1.0. The instances were pseudo-randomly generated using the package NetworkX in a manner to guarantee that each label required was in the ELGs and the

distribution of labels on edges became uniform. The parameters for generating the instances were $|V| = n \in \{50, 100, 250, 500\}$, the density $d = \frac{|E|}{\binom{n}{2}} \in \{0.2, 0.5, 0.8\}$, $|L| \in \{\lfloor \frac{n}{4} \rfloor, \frac{n}{2}, n, \lfloor 1.25n \rfloor\}$, $k \in \{1, \frac{n}{25}, \frac{n}{10}\}$. For each parameter combination, 5 instances were created what resulted in a total of 720 instances.

About the MILP model, four methods were used for solving it: The standard CPLEX branch-and-cut applied to the the model (1a)-(1f) here refereed to as "Flow"; "Flow + Inequalities" being the same as "Flow", but reinforced with the inequalities (2) and (3); "Flow + Benders" is the same as "Flow", but it is used the CPLEX standard Benders decomposition; Finally, "Flow + Benders + Inequality" method is a combination of both "Flow + Benders" and "Flow + Inequalities".

A time limit of 1 hour and a memory limit of 10 GB were also established, since preliminary tests revealed that the methods based on Benders decomposition actually surpassed the computer RAM.

Of the four methods discussed in the last subsection, the "Flow + Benders + Inequality" has proven to be the most efficient. In Figure 5 is seen that "Flow + Benders + Inequality" solved the most amount of instances while has fewer cases of instances that exceeded time limit. The mean time taken in processing each instance is displayed in Figure 6 and the "Flow + Benders + Inequality" was faster than the other methods even on small time scales.

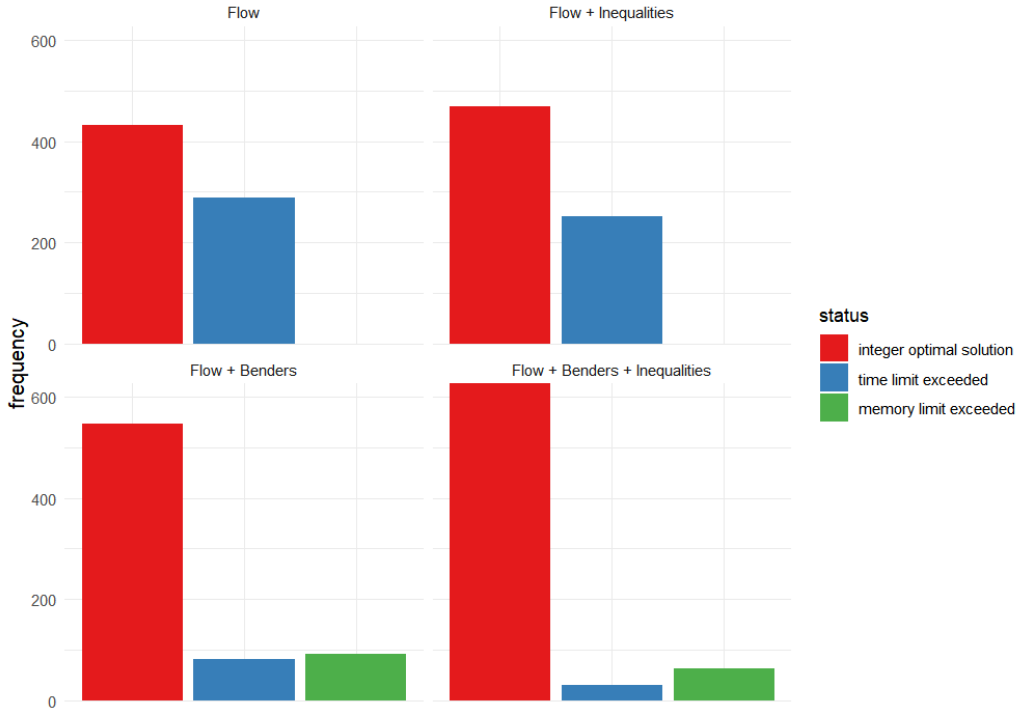


Figure 5: Frequency of statuses returned for every each method used.

Experiments were also made with the approximation algorithm, as seen in Figure 7, that gives the ratios between the number of labels given by the generalized MVCA and the number of labels in the optimal solution that were possible to obtain by the exact methods. It's perceptible that no instance solved returned a ratio than $1.5 = H_2$, in fact, for some combinations of parameters the worst obtained ration was the optimal solution itself, what suggests that the algorithm works well in practice.

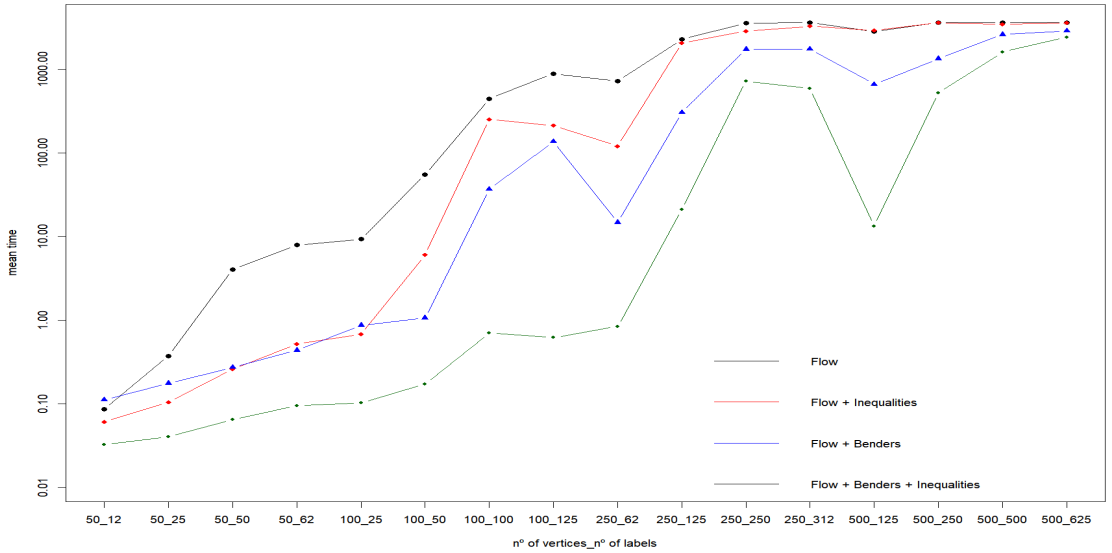


Figure 6: Mean processing time of instances in logarithmic scale categorized by vertex number and label number.

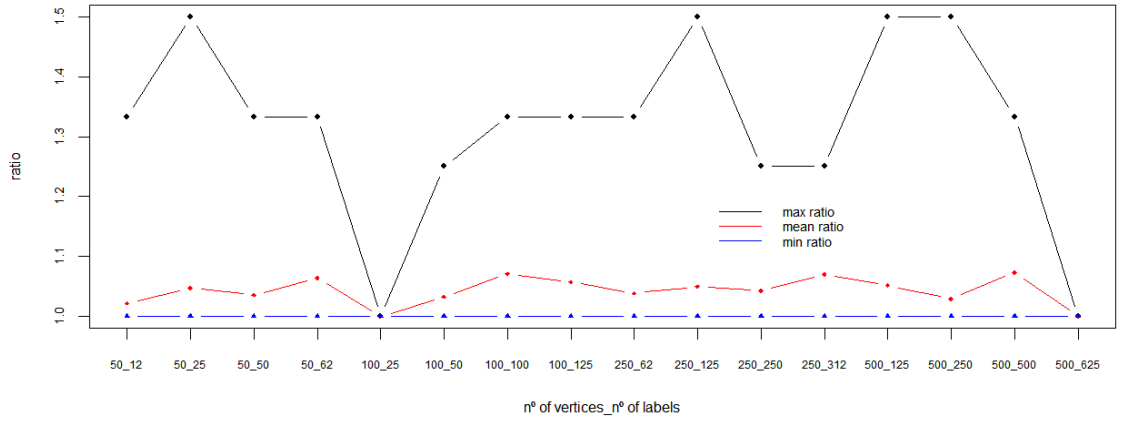


Figure 7: Ratios between the number of labels obtained by the algorithm 1 and optimal solutions categorized by number of vertices and number of labels.

7. Conclusion

The concept of finding a spanning forest with at most k components and a minimum number of labels is a natural extension of the MSLTP. It was shown that the MSLkFP is NP-complete for every k and an H_b-approximation factor algorithm is proposed, together with the fact that such factor is tight. At last, improvements were made for the MILP model proposed in [Corrêa e Campêlo, 2024] and the computational experiments were incremented with an extended database. Possible future research topics include the studies of heuristics and the behavior of solutions in specific graph structures, new mathematical programming models and analyses involving the parametrized complexity of the problem.

References

Cerulli, R., Ambrosio, C., Serra, D., e Sorgente, C. (2024). The budgeted labeled minimum spanning tree problem. *Mathematics*, 12(2). ISSN 2227-7390. URL https://www.mdpi.com/2227-7390/12/2/230.

Cerulli, R., Fink, A., e Gentili, M. (2006). Extensions of the minimum labelling spanning tree problem. *Journal of Telecommunications and Information Technology*, 26(4):39–45. URL https://jtit.pl/jtit/article/view/396.

Chang, R.-S. e Leu, S.-J. (1997). The minimum labeling spanning trees. *Inf. Process. Lett.*, 63 (5):277–282. ISSN 0020-0190. URL https://doi.org/10.1016/S0020-0190(97)00127-0.

Chwatal, A. M., Raidl, G. R., e Dietzel, O. (2007). Compressing fingerprint templates by solving an extended minimum label spanning tree problem. URL https://api.semanticscholar.org/CorpusID:6693734.

Corrêa, L. e Campêlo, M. (2024). O problema da k-floresta geradora minimamente rotulada. In *Anais do LVI SBPO*, Fortaleza.

da Silva, T. G. (2018). *The Minimum Labeling Spanning Tree and Related Problems*. PhD thesis, Université d'Avignon; Universidade Federal Fluminense.

Figueredo, P. J. d. A. (2020). O problema da floresta geradora k-rotulada. Mestrado em ciência da computação, Programa de Pós-Graduação em Ciência da Computação, Centro de Ciências, Universidade Federal do Ceará, Fortaleza.

Krumke, S. O. e Wirth, H.-C. (1998). On the minimum label spanning tree problem. *Information Processing Letters*, 66(2):81–85. ISSN 0020-0190. URL https://www.sciencedirect.com/science/article/pii/S0020019098000349.

Wan, Y., Chen, G., e Xu, Y. (2002). A note on the minimum label spanning tree. *Information Processing Letters*, 84(2):99–101. ISSN 0020-0190. URL https://www.sciencedirect.com/science/article/pii/S0020019002002302.

Xiong, Y., Golden, B., e Wasil, E. (2005). Worst-case behavior of the mvca heuristic for the minimum labeling spanning tree problem. *Operations Research Letters*, 33(1):77–80. ISSN 0167-6377. URL https://www.sciencedirect.com/science/article/pii/S0167637704000483.

Este trabalho foi parcialmente financiado por CNPQ (132983/2024-9, 442977/2023-9, 404479/2023-5 e 312417/2022-5) e FUNCAP (ITR-0214-00096.01.00/23).