An Exact Approach for the Precedence Constrained Generalized Traveling Salesman Problem

Daniil Khachai, Olga Battaïa, Michael Khachay, Ruslan Sadykov

An Exact Approach for the Precedence Constrained Generalized Traveling Salesman Problem

Daniil Khachai*[1,2], Olga Battaia†[1], Michael Khachay‡[3], and Ruslan Sadykov§[2]

[1]Kedge Business School, 680 Cours de la Liberation, Talence, France
[2]Inria Bordeaux Sud-Ouest, 200 Avenue de la Vieille Tour, Talence 33405, France
[3]Ivannikov Institute for System Programming of RAS, Moscow, Russia

20 November 2022

Abstract

The Precedence Constrained Generalized Traveling Salesman Problem (PCGTSP) is an extension of two well-known combinatorial optimization problems — the Generalized Traveling Salesman Problem (GTSP) and the Precedence Constrained Asymmetric Traveling Salesman Problem (PCATSP), whose path version is known as the Sequential Ordering Problem (SOP). Similarly to the classic GTSP, the goal of the PCGTSP, for a given input digraph and partition of its node set into clusters, is to find a minimum cost cyclic route (tour) visiting each cluster in a single node. In addition, as in the PCATSP, feasible tours are restricted to visit the clusters with respect to the given partial order. Unlike the GTSP and SOP, to the best of our knowledge, the PCGTSP still remain to be weakly studied both in terms of polyhedral theory and algorithms. In this paper, for the first time for the PCGTSP, we propose several families of valid inequalities, establish dimension of the PCGTS polytope and prove sufficient conditions ensuring that the extended Balas' π- and σ-inequalities become facet-inducing. Relying on these theoretical results and evolving the state-of-the-art algorithmic approaches for the PCATSP and SOP, we introduce a family of MILP-models (formulations) and several variants of the branch-and-cut algorithm for the PCGTSP. We prove their high performance in a competitive numerical evaluation against the public benchmark library PCGTSPLIB, a known adaptation of the classic SOPLIB to the problem in question.

1 Introduction

Introduced in the seminal paper by Srivastava et al. (1969), the Generalized Traveling Salesman Problem (GTSP) is one of the most well-known generalizations of the classic Traveling Salesman Problem (TSP). It has numerous industrial applications including air time minimization in metal sheet cutting (Dewil et al. (2016); Chentsov et al. (2018); Makarovskikh et al. (2018)) and coordinate measuring machinery (Salman et al. (2016)).

An instance of the GTSP can be defined informally as follows. A salesperson travels across a given transportation network consisting of *cities* and *roads* connecting them, represented by the nodes and arcs of some directed graph, respectively. The set of cities is partitioned into subsets called *clusters*. By traveling on any road, the salesperson is charged with a corresponding

*daniil.khachai@kedgebs.com
†olga.battaia@kedgebs.com
‡mkhachay@imm.uran.ru
§ruslan.sadykov@inria.fr

transportation cost. The goal is to construct a closed tour that visits each cluster in one city exactly and minimizes the accumulated transportation costs.

Being an extension of the classic Traveling Salesman Problem (TSP), the GTSP is strongly NP-hard even on the Euclidean plane (Papadimitriou (1977)) any time when number of clusters m is a part of the input. On the other hand, an adaptation to this problem of the well-known Held and Karp dynamic programming scheme (Held and Karp (1962)) has running-time bound $O(n^3m^2 \cdot 2^m)$, i.e. the GTSP belongs to the class of Fixed-Parameter Tractable (FPT) problems, being parameterized by the number of clusters. Furthermore, it can be solved to optimality in polynomial time, provided that $m = O(\log n)$. In the case of PCGTSP, the running time is $O(n^3m^2 \cdot |\mathcal{J}|)$, where $\mathcal{J}$ is a set of ideals of the given partial order (see e.g. Khachay et al. (2021)). In particular, if the order specifying the precedence constraints is of fixed *width* w, then $|\mathcal{J}| = O(m^w)$ (Steiner (1990)). Thus, in this case, the PCGTSP can be solved to optimality in polynomial time.

The algorithmic design for the GTSP has been developed in the literature in several directions. The first approach is based on the reduction of the initial problem to an appropriate instance of the Asymmetric TSP (ATSP) (Noon and Bean (1993)), which at first glance gives an opportunity to employ a vast variety of known algorithms designed for the ATSP (see e.g. Roberti and Toth (2012)). However, despite its mathematical elegance, this approach suffers from several technical shortcomings. First, even a close-to-optimal solutions of such auxiliary ATSP instances may correspond to infeasible solutions of the initial GTSP. Furthermore, such instances may have a quite unusual shape and thus difficult to solve for the existing TSP solvers (Karapetyan and Gutin (2012), see also Yuan et al. (2020)).

Another approach provides various heuristics and meta-heuristics. Among them are the memetic algorithms (Gutin and Karapetyan (2010)), an extension of the Lin-Kernighan-Helsgaun heuristic solver (Helsgaun (2015)), and the GLNS meta-heuristic (Smith and Imeson (2017)) based on the Adaptive Large Neighborhood Search (ALNS) framework, which appears to be the most efficient at the moment.

Finally, the third research direction is related to design of approximation algorithms with theoretical performance guarantees (see e.g. Feremans et al. (2006); Khachai and Neznakhina (2017)) and problem-specific branch-and-bound and branch-and-cut algorithms (Fischetti et al. (1997); Yuan et al. (2020)).

The Sequential Ordering Problem (SOP), which is extremely close to the PCATSP, was introduced in (Escudero (1988)). We should mention three groups of important results obtained for the both problems on which the current research for the PCGTSP is based on.

The first of them, in the field of polyhedral study of the PCATSP, was obtained in the seminal paper (Balas et al. (1995)), where sufficient conditions for the π- and σ-inequalities to be facet-inducing were proved.

The second group comprises valid inequalities that exploit precedence constraints explicitly and approaches to their strengthening, as well as the design of MILP-models (formulations) in order to obtain better lower bounds while decreasing time complexity of the appropriate LP-relaxations. Among them are compact formulations proposed in (Sarin et al. (2005)) as an extension of results of (Gouveia and Pires (1999, 2001); Sherali and Driscoll (2002)), and formulations whose exponential-size sets of valid inequalities are supplemented with polynomial-time separation techniques (Gouveia and Pesneau (2006)). To the best of our knowledge, to the date, the models providing the tightest lower bounds were introduced in (Gouveia et al. (2018)).

The last group of results relies on design and implementation of problem-specific branch-and-cut algorithms including ones proposed in (Ascheuer et al. (2000)), (Cire and van Hoeve (2013)) and (Gouveia and Ruthmair (2015)), where the last one is regarded to be state-of-the-art on the topic.

In this paper, we consider the PCGTSP, which is an extension of the GTSP, where the feasible tours are restricted to visit all the clusters with respect to the partial order. At the same time, PCGTSP extends the PCATSP as follows. Any instance of PCATSP is considered to be

the instance of PCGTSP, where all clusters are singletons. Unlike both GTSP and PCATSP, the PCGTSP considered in this paper still remains weakly studied. To the best of our knowledge, all the related published results are exhausted by:
(i) efficient algorithms for several specific precedence constraints including partial orders of Balas-type (see e.g. Balas and Simonetti (2001); Chentsov et al. (2016)) and the orders that lead to quasi- and pseudo-pyramidal optimal tours (Khachay and Neznakhina (2020));
(ii) the PCGLNS heuristic solver proposed in Khachay et al. (2020) that extends the results obtained in (Smith and Imeson (2017)) to the case of PCGTSP;
(iii) branch-and-bound and DP-and-bound algorithms for this problem (Khachay et al. (2021)), based on Balas instance preprocessing (Balas et al. (1995)), Held and Karp branching framework (see e.g. Morin and Marsten (1976)), and the combinatorial lower bounds from Salman et al. (2020),
(iv) the public PCGTSPLIB benchmark library proposed in (Salman et al. (2020)) as an extension of the well-known SOPLIB library. According to the literature (Salman et al. (2020); Khachay et al. (2021)), 12 out of 40 instances of this library were solved to optimality. Meanwhile, their solutions can be found within a competitive time by Gurobi solver supplied with our extension of the L1PCATSPxy compact model, previously introduced in (Sarin et al. (2005)) for the PCATSP, built-in cutting planes, and PCGLNS primal heuristic.

In addition, we should mention the branch-and-cut algorithm proposed recently in Yuan et al. (2020) for the GTSP with time windows. This result seems to be relevant as the time windows defined on clusters induce natural precedence constraints. Unfortunately, this approach is hardly applicable to the general PCGTSP, since a partial order defined on the set of clusters not necessarily admits encoding in terms of time windows.

In this paper, we try to bridge the gap both in the context of polyhedral theory and in the field of branch-and-cut algorithms for the considered problem. Contribution of this paper is three-fold:
(i) by evolving the inductive framework developed in Fischetti et al. (1995) for the symmetric GTSP, we establish dimension of the PCGTS polytope and extend the sufficient facet-inducing conditions for π- and σ-inequalites proved initially in (Balas et al. (1995)) for the PCATSP, to the more general case of the PCGTSP;
(ii) relying on the known results on formulations for the PCATSP (Sarin et al. (2005); Gouveia and Pesneau (2006); Gouveia et al. (2018)), we propose novel valid inequalities for the PCGTSP and a family of compact and exponential-size MILP-models for this problem aimed to increase tightness of their lower bounds and speed-up the solution procedure for the appropriate LP-relaxations;
(iii) by combining the best formulations (in terms of lower bounds and running times) and the PCGLNS primal heuristic, for the first time, we propose several variants of the branch-and-cut algorithm for the PCGTSP, and compare their performance with aforementioned best known results and our adaptation of the state-of-the-art algorithm proposed in Gouveia and Ruthmair (2015) for the SOP.

As a result, the number of PCGTSPLIB instances solved to optimality has increased almost twice, to 23 out of 40 instances. Furthermore, the carried out numerical evaluation confirm that the considered MILP-models and branch-and-cut algorithm for the PCGTSP benefit well from the incorporation of the predecessor/successor inequalities.

The rest of the paper is organized as follows. In Section 2, we give a mathematical statement of the considered problem, introduce some necessary definitions and notation, discuss the instance preprocessing, and describe the compact MILP-model used throughout the paper. In Section 3, we propose novel families of valid inequalities for the problem in question and explain the corresponding separation procedures. Section 4 deals with the polyhedral study of the PCGTSP. By extending the seminal results of Balas et al. (1995) and Fischetti et al. (1995), we establish dimension of the PCGTS polytope and prove the conditions sufficient for π- and σ-inequalities to be facet-inducing. Further, Section 5 represents the proposed formulations for the PCGTSP, while Section 6 gives an overview of our branch-and-cut algorithm. In Section 7 we report the results of the numerical evaluation, both for the proposed formulations and suggested variants of the branch-and-cut algorithm. Finally, Section 8 concludes the paper.

2 Problem statement

An instance of PCGTSP is given by a triple $(G, \mathcal{C}, \mathcal{G})$, where
- the complete loopless arc-weighted digraph $G = (V, E, c)$, $|V| = n$, defines a groundset network supplemented with transportation costs $c(u, v)$ for an arbitrary arc $(u, v) \in E$;
- the partition $\mathcal{C} = \{C_1, \dots, C_m\}$ splits the nodeset V of the graph G into m non-empty pairwise-disjoint *clusters*, where the cluster C_1 is referred to as *depot*;
- the directed acyclic graph $\mathcal{G} = (\mathcal{C}, A)$ defines a partial order (*precedence constraints*) on the set of clusters $\mathcal{C}$. Further, without loss of generality, we assume $\mathcal{G}$ to be *transitively closed*, i.e. $(C_i, C_j) \in A$ and $(C_j, C_k) \in A$ imply $(C_i, C_k) \in A$, and that $(C_1, C_p) \in A$ for any $p \in \{2, \dots, m\}$.

A closed m-tour T is called a *feasible* solution of the PCGTSP, if
- it departs from and arrives at some node $v_1 \in C_1$;
- it visits each cluster $C_p \in \mathcal{C}$ exactly once;
- the tour T is consistent with the partial order $\mathcal{G}$, i.e. no cluster C_q can be visited by the tour T before its arbitrary predecessor in the order $\mathcal{G}$.

The cost of a tour $T = (v_1, v_2, \dots, v_m)$ is the sum of costs of its arcs $cost(T) = c(v_m, v_1) + \sum_{i=1}^{m-1} c(v_i, v_{i+1})$. The objective of the PCGTSP is to find a feasible m-tour of the minimum cost.

2.1 Preliminaries

We start with some necessary definitions and notation. For any pair of clusters C_p and C_q except the depot cluster C_1, for which $(C_p, C_q) \in A$, we refer to C_p as a *predecessor* of C_q (and C_q as a *successor* of C_p) or shortly $C_p \prec C_q$. Further, to any non-depot cluster C, we assign subsets $\pi(C) = \{C_i \neq C_1 \colon C_i \prec C\}$ and $\sigma(C) = \{C_i \neq C_1 \colon C \prec C_i\}$ of its predecessors and successors, respectively. This notation can be easily extended to an arbitrary nonempty subset of clusters $\mathcal{C}' \subset \mathcal{C} \setminus \{C_1\}$: $\pi(\mathcal{C}') = \bigcup_{C \in \mathcal{C}'} \pi(C)$, $\quad \sigma(\mathcal{C}') = \bigcup_{C \in \mathcal{C}'} \sigma(C)$. In turn, by $\tilde{\pi}(C)$ and $\tilde{\sigma}(C)$ we denote the subsets of $\pi(C)$ and $\sigma(C)$ respectively consisting of the direct parents and children of the cluster C. Finally, by $\mathcal{C}^+ = \bigcup_{i=2}^m \pi(C_i)$ and $\mathcal{C}^- = \bigcup_{i=2}^m \sigma(C_i)$ we denote the sets of all aforementioned predecessors and successors, respectively.

If, for $C \neq C_1$, $\pi(C) \cup \sigma(C) = \varnothing$, we call C a *free* cluster. In terms of polyhedral results, we restrict ourselves to the setting of PCGTSP with a singleton free cluster, which we call C_{Balas}.

In the following, by $C(v)$ we denote (the only) cluster that contains an arbitrary node $v \in V$. We call v a *non-individual* node, if $|C(v)| > 1$, otherwise v is called *individual.* To simplify the problem at hand, we use the instance preprocessing technique proposed in Balas et al. (1995). We exclude any arc $(i, j) \in E$, for which at least one of the following options holds:

$$(i \in C_1) \;\&\; (j \in \mathcal{C}^-) \tag{1}$$
$$(i \in \mathcal{C}^+) \;\&\; (j \in C_1) \tag{2}$$
$$C(j) \prec C(i) \tag{3}$$
$$\exists\, \tilde{C} \in \mathcal{C} \colon (C(i) \prec \tilde{C}) \;\&\; (\tilde{C} \prec C(j)) \tag{4}$$
$$C(i) = C(j). \tag{5}$$

For any proper subset $\varnothing \neq S \subset V$, we use the standard notation $\delta^-(S) = \{(i, j) \in E \colon i \notin S, j \in S\}$, $\delta^+(S) = \{(i, j) \in E \colon i \in S, j \notin S\}$, and $\delta(S) = \delta^+(S) \cup \delta^-(S)$ for the appropriate incoming and outgoing cuts, and their union, respectively.

In the case of a singleton $S = \{v\}$, we use simple notation $\delta^+(v)$ and $\delta^-(v)$.

Without loss of generality, we assume that graph G has no isolated nodes after preprocessing.

Furthermore, we can assume that, for any node $v \in V$, $\delta^+(v) \neq \varnothing$ and $\delta^-(v) \neq \varnothing$. As a simple consequence, we obtain that $\delta^+(C) \neq \varnothing$ and $\delta^-(C) \neq \varnothing$ for any cluster C as well.

2.2 Compact MILP model

To obtain a basic compact model for the considered problem, we extend the known L1PCATSPxy formulation, proposed in Sarin et al. (2005) for the PCATSP, which is the best performer among compact models in terms of LP-relaxation bounds for that problem.

For any $(i,j) \in E$ and node $v \in V$, we introduce the following binary decision variables:

$$x_{ij} = \begin{cases} 1, & \text{if } (i,j) \text{ belongs to the solution} \\ 0, & \text{otherwise,} \end{cases} \qquad z_v = \begin{cases} 1, & \text{if } v \text{ is visited by the solution} \\ 0, & \text{otherwise.} \end{cases}$$

In addition, we introduce auxiliary variables y_{pq} and u_{pq}:

$$y_{pq} = \begin{cases} 1, & \text{if cluster } C_p \text{ precedes } C_q \text{ in the solution (not necessarily immediately)} \\ 0, & \text{otherwise,} \end{cases}$$

$$u_{pq} = \begin{cases} 1, & \text{if cluster } C_p \text{ immediately precedes } C_q \text{ in the solution} \\ 0, & \text{otherwise.} \end{cases}$$

The proposed MILP model for the PCGTSP is as follows:

$$\min \sum_{(i,j)\in E} c_{ij} x_{ij}, \tag{6}$$

$$\text{s.t.} \quad \sum_{i \in C_k} z_i = 1 \quad (k \in \{1,\dots,m\}) \tag{7}$$

$$\sum_{(i,j)\in\delta^+(i)} x_{ij} = z_i \quad (i \in V) \tag{8}$$

$$\sum_{(i,j)\in\delta^-(i)} x_{ji} = z_i \quad (i \in V) \tag{9}$$

$$\sum_{q=1, q\neq p}^{m} u_{pq} = 1 \ \ (p \in \{1,\dots,m\}), \qquad \sum_{p=1, p\neq q}^{m} u_{pq} = 1 \ \ (q \in \{1,\dots,m\}) \tag{10}$$

$$\sum_{i\in\delta^+(C_p)} \sum_{j\in\delta^-(C_q)} x_{ij} = u_{pq} \quad (p,q \in \{1,\dots,m\}, p \neq q) \tag{11}$$

$$(y_{pq} + u_{qp}) + y_{qr} + y_{rp} \leqslant 2 \quad (p,q,r \in \{2,\dots,m\},\ p \neq q \neq r) \tag{12}$$

$$u_{pq} - y_{pq} \leqslant 0 \quad (p,q \in \{2,\dots,m\},\ p \neq q) \tag{13}$$

$$y_{pq} + y_{qp} = 1 \quad (\{p,q\} \subset \{2,\dots,m\}) \tag{14}$$

$$y_{pq} = 1 \quad (p,q \in \{2,\dots,m\},\ C_p \prec C_q) \tag{15}$$

$$x_{ij}, z_i \in \{0,1\}, u_{pq} \geqslant 0, y_{pq} \geqslant 0 \tag{16}$$

The objective is to minimize the total traveling cost (6). Constraints (7) ensure that exactly one node from each cluster is visited. Constraints (8) and (9) are flow conservation constraints in terms of nodes, while constraints (10) are flow conservation constraints in terms of clusters. Technical constraints (11) establish the link between the decision and auxiliary variables. Similarly to the initial L1PCATSPxy model, constraints (12)-(15) ensure subtour elimination and establish the given precedence constraints simultaneously.

By evolving the arguments of Sarin et al. (2005), it is easy to verify the following observation.

Observation 1. *For any feasible solutions* $[x', z', u', y']$ *and* $[x'', z'', u'', y'']$ *of the model* (6)-(16), $(x' = x'') \wedge (z' = z'') \Rightarrow (u' = u'') \wedge (y' = y'')$.

3 Valid inequalities

In this section, we extend to the case of PCGTSP some known families of valid inequalities initially introduced in papers Balas et al. (1995); Gouveia and Ruthmair (2015); Gouveia et al. (2018) for the PCATSP. It is convenient to specify these inequalities in terms of the following standard notation. For any non-empty disjoint cluster subsets $\mathcal{U}',\mathcal{U}'' \subset \mathcal{C}$,

$$x(\mathcal{U}',\mathcal{U}'') = \sum_{C_p\subset\mathcal{U}'}\sum_{C_q\subset\mathcal{U}''}\sum_{i\in C_p}\sum_{j\in C_q} x_{ij} \equiv \sum_{C_p\subset\mathcal{U}'}\sum_{C_q\subset\mathcal{U}''} u_{pq}.$$

3.1 Predecessor and successor inequalities

Proposition 1. *For an arbitrary non-empty $\mathcal{S}\subset\mathcal{C}\setminus\{C_1\}$, $\bar{\mathcal{S}}=\mathcal{C}\setminus\mathcal{S}$, the predecessor-inequality (π-inequality):*

$$x(\mathcal{S}\setminus\pi(\mathcal{S}),\bar{\mathcal{S}}\setminus\pi(\mathcal{S})) \geqslant 1 \tag{17}$$

is valid for the PCGTSP.

Proof. Let T be an arbitrary tour that satisfies the precedence constraints and C_p be the last cluster in $\mathcal{S}$ visited by T. Then, $C_p\in\mathcal{S}\setminus\pi(\mathcal{S})$ and for the next cluster visited by T, $C_q\in\bar{\mathcal{S}}\setminus\pi(\mathcal{S})$. Such a cluster exists, since the tour T should depart from and arrive at C_1. Therefore, $x(\mathcal{S}\setminus\pi(\mathcal{S}),\bar{\mathcal{S}}\setminus\pi(\mathcal{S}))\geqslant u_{pq}=1$. □

Since the following two propositions can be treated similarly, we skip their proofs for the sake of brevity.

Proposition 2. *For an arbitrary non-empty $\mathcal{S}\subset\mathcal{C}\setminus\{C_1\}$, $\bar{\mathcal{S}}=\mathcal{C}\setminus\mathcal{S}$, the successor-inequality (σ-inequality):*

$$x(\bar{\mathcal{S}}\setminus\sigma(\mathcal{S}),\mathcal{S}\setminus\sigma(\mathcal{S})) \geqslant 1 \tag{18}$$

is valid for the PCGTSP.

Proposition 3. *Let $\mathcal{X},\mathcal{Y}\subset\mathcal{C}\setminus\{C_1\}$ be non-empty subsets such that, for an arbitrary clusters $C'\in\mathcal{X}$ and $C''\in\mathcal{Y}$, $C'\prec C''$, and let $\mathcal{Q}=\{C_1\}\cup\pi(\mathcal{X})\cup\sigma(\mathcal{Y})$. Then for any $\mathcal{S}\subset\mathcal{C}$, $\bar{\mathcal{S}}=\mathcal{C}\setminus\mathcal{S}$ such that $\mathcal{X}\subseteq\mathcal{S}$, $\mathcal{Y}\subseteq\bar{\mathcal{S}}$, the (π,σ)-inequality:*

$$x(\mathcal{S}\setminus\mathcal{Q},\bar{\mathcal{S}}\setminus\mathcal{Q}) \geqslant 1 \tag{19}$$

is valid for the PCGTSP.

3.2 Precedence cycle breaking inequalities

For some natural t, consider a subset $\mathcal{C}'=\{C_{i_1},\dots,C_{i_{2t+1}}\}\subset\mathcal{C}\setminus\{C_1\}$, such that $C_{i_1}\prec\ldots\prec C_{i_{2t+1}}$. Introduce the subsets $\mathcal{C}'_{odd}=\{C_{i_{2s+1}}: s\in\{0,\dots,t\}\}$ and $\mathcal{C}'_{even}=\{C_{i_{2s}}: s\in\{1,\dots,t\}\}$ of $\mathcal{C}'$, that contain C_{i_j} with odd and even j respectively.

Proposition 4. *For an arbitrary non-empty $\mathcal{S}\subset\mathcal{C}\setminus\{C_1\}$, $\bar{\mathcal{S}}=\mathcal{C}\setminus\mathcal{S}$, such that $\mathcal{C}'_{odd}\subset\mathcal{S}$ and $\mathcal{C}'_{even}\subset\bar{\mathcal{S}}$,*

$$x(\mathcal{S},\bar{\mathcal{S}}) \geq t+1 \tag{20}$$

is valid for the PCGTSP.

Proof. Indeed, consider an arbitrary feasible tour T. Since clusters $C_{i_1}\ldots C_{i_{2t+1}}$ are linearly ordered and $C_1\notin\mathcal{S}$, the tour T crosses the border from $\mathcal{S}$ to $\bar{\mathcal{S}}$ at least $t+1$ times. □

Following (Gouveia and Ruthmair (2015)), without loss of generality, we can assume that $C_{i_j}\in\tilde{\pi}(C_{i_{j+1}})$ for each $j\in\{1,\dots,2t\}$. Furthermore, we can strengthen inequality (20) as follows.

Proposition 5. *For an arbitrary non-empty* $\mathcal{S} \subset \mathcal{C} \setminus \{C_1\}$, $\bar{\mathcal{S}} = \mathcal{C} \setminus \mathcal{S}$, *such that* $\mathcal{C}'_{odd} \subset \mathcal{S}$ *and* $\mathcal{C}'_{even} \subset \bar{\mathcal{S}}$, *the condition* $\tilde{\sigma}(C_{i_{2t+1}}) \not\subset \mathcal{S}$ *implies the validity of inequality*

$$x(\mathcal{S} \setminus \mathcal{S}', \bar{\mathcal{S}} \setminus \mathcal{S}') \geq t + 1, \tag{21}$$

where $\mathcal{S}' = \pi(C_{i_1}) \cup \sigma(C_{i_{2t+1}}) \setminus \tilde{\sigma}(C_{i_{2t+1}})$.

3.3 Single-option inequalities

In this subsection, we extend the family of simple (but powerful) inequalities proposed in Gouveia and Ruthmair (2015) for the PCATSP, whose validity can be easily obtained from (7)-(11) and precedence constraints.

Proposition 6. *For an arbitrary* $\{C_i, C_j\} \subset \mathcal{C} \setminus \{C_1\}$, *the following inequalities*

$$\begin{aligned}
u_{ij} + u_{ji} + u_{kl} + u_{lk} &\leq 1 \quad (C_k \in \pi(C_i),\ C_l \in \sigma(C_j)) && (22)\\
u_{ij} + u_{ji} + \sum_{C_l \in \sigma(C_j)} u_{kl} &\leq 1 \quad (C_k \in \pi(C_i)) && (23)\\
u_{ij} + u_{ji} + \sum_{C_l \in \sigma(C_j)} u_{lk} &\leq 1 \quad (C_k \in \pi(C_i)) && (24)\\
u_{ij} + u_{ji} + \sum_{C_k \in \pi(C_i)} u_{kl} &\leq 1 \quad (C_l \in \sigma(C_j)) && (25)\\
u_{ij} + u_{ji} + \sum_{C_k \in \pi(C_i)} u_{lk} &\leq 1 \quad (C_l \in \sigma(C_j)) && (26)
\end{aligned}$$

are valid for the PCGTSP.

3.4 Strengthened precedence variables and network flow based inequalities

The authors of (Gouveia et al. (2018)) introduced a novel exponential-size families of valid inequalities augmented with polynomial-time separation procedures, their strengthened counterparts, and the related formulations for the PCATSP. Comprehensive numerical analysis carried out there showed that more tight lower bounds were provided by the formulations based on strengthened inequalities. Therefore, in this paper, we restrict ourselves only on extension to the PCGTSP of these families.

Proposition 7. *For an arbitrary clusters* C_p *and* C_q *not equal to* C_1, *where* $p \neq q$, *the strengthened simple-cut inequality* $x(\mathcal{S}, \bar{\mathcal{S}}) \geq y_{pq}$ *is valid for the PCGTSP, for any partition*

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus C^1_{pq}) \cup \{C_p, C_q\}, \text{ such that } C_p \in \mathcal{S},\ C_q \in \bar{\mathcal{S}}, \tag{27}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus C^2_{pq}) \cup \{C_1, C_p\}, \text{ such that } C_1 \in \mathcal{S},\ C_p \in \bar{\mathcal{S}}, \tag{28}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus C^3_{pq}) \cup \{C_1, C_q\}, \text{ such that } C_q \in \mathcal{S},\ C_1 \in \bar{\mathcal{S}}, \tag{29}$$

where $C^1_{pq} = \{C_1\} \cup \pi(C_p) \cup \sigma(C_q)$, $C^2_{pq} = \{C_q\} \cup \sigma(C_p) \cup \sigma(C_q)$ *and* $C^3_{pq} = \{C_p\} \cup \pi(C_p) \cup \pi(C_q)$.

Proposition 8. *For an arbitrary triple* (C_p, C_q, C_r) *of distinct clusters not equal to* C_1, *the strengthened GDDL inequality*[1]

$$x(\mathcal{S}, \bar{\mathcal{S}}) \geq y_{pr} + y_{rq} \tag{30}$$

is valid for the PCGTSP for any partition $(\mathcal{S}, \bar{\mathcal{S}})$ *of* $\big(\mathcal{C} \setminus \mathcal{C}_{pqr}\big) \cup \{C_1, C_p, C_q, C_r\}$, *such that* $\{C_1, C_r\} \subset \mathcal{S}$, $\{C_p, C_q\} \subset \bar{\mathcal{S}}$ *and* $\mathcal{C}_{pqr} = \big(\sigma(C_p) \cap \sigma(C_q)\big) \cup \big(\pi(C_r) \cap \sigma(C_p)\big) \cup \big(\sigma(C_q) \cap \sigma(C_r)\big)$.

[1]Generalized Disaggregated Desrochers-Laporte inequality

Proposition 9. *For an arbitrary triple (C_p, C_q, C_r) of distinct clusters not equal to C_1, the strengthened Reversed GDDL inequality*

$$x(\mathcal{S}, \bar{\mathcal{S}}) \geq y_{pr} + y_{rq} \tag{31}$$

is valid for the PCGTSP for any partition $(\mathcal{S}, \bar{\mathcal{S}})$ of $\big(\mathcal{C} \setminus \mathcal{C}^R_{pqr}\big) \cup \{C_1, C_p, C_q, C_r\}$, such that $\{C_p, C_q\} \subset \mathcal{S}$, $\{C_1, C_r\} \subset \bar{\mathcal{S}}$ and $\mathcal{C}^R_{pqr} = \big(\pi(C_p) \cap \pi(C_r)\big) \cup \big(\pi(C_p) \cap \pi(C_q)\big) \cup \big(\sigma(C_r) \cap \pi(C_q)\big)$.

Proposition 10. *For an arbitrary triple (C_p, C_q, C_r) of distinct clusters not equal to C_1, the strengthened 2-path inequality $x(\mathcal{S}, \bar{\mathcal{S}}) \geq y_{pq} + y_{qr} - 1$ is valid for the PCGTSP, for any partition*

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^1_{pqr}) \cup \{C_1, C_p\}, \text{ such that } C_1 \in \mathcal{S},\ C_p \in \bar{\mathcal{S}}, \tag{32}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^2_{pqr}) \cup \{C_p, C_q\}, \text{ such that } C_p \in \mathcal{S},\ C_q \in \bar{\mathcal{S}}, \tag{33}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^3_{pqr}) \cup \{C_q, C_r\}, \text{ such that } C_q \in \mathcal{S},\ C_r \in \bar{\mathcal{S}}. \tag{34}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^4_{pqr}) \cup \{C_1, C_r\}, \text{ such that } C_r \in \mathcal{S},\ C_1 \in \bar{\mathcal{S}}, \tag{35}$$

where $\mathcal{C}^1_{pqr} = \{C_q, C_r\} \cup \sigma(C_p) \cup \sigma(C_q) \cup \sigma(C_r)$, $\mathcal{C}^2_{pqr} = \{C_1, C_r\} \cup \pi(C_p) \cup \sigma(C_q) \cup \sigma(C_r)$, $\mathcal{C}^3_{pqr} = \{C_1, C_p\} \cup \pi(C_p) \cup \pi(C_q) \cup \sigma(C_r)$, and $\mathcal{C}^4_{pqr} = \{C_p, C_q\} \cup \pi(C_p) \cup \pi(C_q) \cup \pi(C_r)$, respectively.

Proposition 11. *For an arbitrary quadruple (C_p, C_q, C_r, C_s) of distinct clusters not equal to C_1, the strengthened 3v GDDL-like inequality $x(\mathcal{S}, \bar{\mathcal{S}}) \geq y_{pq} + y_{qr} + y_{rs} - 1$ is valid for the PCGTSP, for any partition:*

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^1_{pqrs}) \cup \{C_p, C_q, C_r, C_s\}, \text{ such that } \{C_p, C_r\} \subset \mathcal{S},\ \{C_q, C_s\} \subset \bar{\mathcal{S}}, \tag{36}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^2_{pqrs}) \cup \{C_1, C_p, C_q, C_s\}, \text{ such that } \{C_p, C_s\} \subset \mathcal{S},\ \{C_q, C_1\} \subset \bar{\mathcal{S}}, \tag{37}$$

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } (\mathcal{C} \setminus \mathcal{C}^3_{pqrs}) \cup \{C_1, C_p, C_r, C_s\}, \text{ such that } \{C_1, C_r\} \subset \mathcal{S},\ \{C_p, C_s\} \subset \bar{\mathcal{S}}, \tag{38}$$

where $C^1_{pqrs} = \{C_1\} \cup \big((\pi(C_q) \cup \pi(C_r) \cup \sigma(C_s)) \cap (\pi(C_p) \cup \sigma(C_q))\big)$, $C^2_{pqrs} = \{C_r\} \cup \big((\pi(C_r) \cup \pi(C_s)) \cap (\pi(C_p) \cup \sigma(C_q) \cup \sigma(C_r))\big)$, and $C^3_{pqrs} = \{C_q\} \cup \big((\sigma(C_p) \cup \sigma(C_q)) \cap (\pi(C_q) \cup \pi(C_r) \cup \sigma(C_s))\big)$.

Proposition 12. *For an arbitrary quintuple $(C_p, C_q, C_k, C_r, C_s)$ of distinct clusters not equal to C_1, the strengthened 4v GDDL-like inequality $x(\mathcal{S}, \bar{\mathcal{S}}) \geq y_{pq} + y_{qk} + y_{kr} + y_{rs} - 2$ is valid for the PCGTSP, for any partition*

$$(\mathcal{S}, \bar{\mathcal{S}}) \text{ of } \mathcal{C} \setminus \mathcal{C}_{pqkrs} \cup \{C_p, C_q, C_r, C_s\}, \text{ such that } \{C_p, C_r\} \subset \mathcal{S},\ \{C_q, C_s\} \subset \bar{\mathcal{S}}, \tag{39}$$

where $\mathcal{C}_{pqkrs} = \{C_1, C_k\} \cup \big((\pi(C_p) \cup \sigma(C_q) \cup \sigma(C_k)) \cap (\pi(C_k) \cup \pi(C_r) \cup \sigma(C_s))\big)$.

Proofs of all the propositions of this subsection can be obtained by extension of the arguments presented in Gouveia et al. (2018).

3.5 Separation procedures

All the aforementioned families of valid inequalities are augmented with polynomial-time separation procedures, which extend he seminal unit flow propagation approach introduced in Balas et al. (1995). In Algorithm 1, we present the proposed separation technique for π-inequalities (17).

For the sake of brevity, we restrict our further discussion to precedence cycle breaking inequalities (20). Other procedures evolve the similar results obtained in (Gouveia and Ruthmair (2015); Gouveia et al. (2018)) for the PCATSP and (Yuan et al. (2020)) for the GTSP with time windows and can be retrieved from the supplemented source code (Khachai (2022)).

Indeed, suppose we are given by the current fractional solution (x, z, u, y). For a sequence of non-depot clusters $C_{i_1} \prec \ldots \prec C_{i_{2t+1}}$, we construct an auxiliary cluster digraph $H = (\mathcal{C} \cup \{s, t\}, E')$, where s and t are artificial source and destination nodes connected by incapacitated arcs with clusters from $\mathcal{C}'_{odd}$ and $\mathcal{C}'_{even} \cup \{C_1\}$, respectively. For each other arc $(C_p, C_q) \in E'$, its capacity is defined by u_{pq}. Next, if the value of the maximum s-t-flow in the digraph H appears to be less than $t + 1$, an arbitrary minimum cut $(\mathcal{S}, \bar{\mathcal{S}})$, where $\mathcal{S} \subset \mathcal{C} \cup \mathcal{C}'_{\text{odd}} \setminus (\{C_1\} \cup \mathcal{C}'_{\text{even}})$ and $\bar{\mathcal{S}} = \mathcal{C} \setminus \mathcal{S}$, defines inequality (20) violated by the given solution.

Algorithm 1 Separation technique for π-inequalities

Input: current (fractional) solution $(x_{ij}, z_i, u_{pq}, y_{pq})$, a non-depot cluster $C \neq C_1$
Output: π-inequality for an appropriate $\mathcal{S}$ (if any)

1: create an auxiliary cluster digraph $G_C = (\mathcal{C}_C, E_C)$, where $\mathcal{C}_C = \mathcal{C} \setminus \pi(C)$ and (C_p, C_q) belongs to E_C and has capacity u_{pq} if and only if $u_{pq} > 0$
2: find a maximum C-to-C_1 flow F in the graph G_C
3: **if** $val(F) < 1$ **then**
4: find a minimum cut $\mathcal{U}', \mathcal{U}'' \subset \mathcal{C}_C$
5: set $\mathcal{S} = \mathcal{U}' \cup \pi(C)$ and $\bar{\mathcal{S}} = \mathcal{C} \setminus \mathcal{S} = \mathcal{U}''$
6: **return** π-inequality

$$x(\mathcal{S} \setminus \pi(\mathcal{S})), \bar{\mathcal{S}} \setminus \pi(\mathcal{S})) \geqslant 1$$

7: **end if**

4 Facets of the PCGTS polytope

In this section, we study a polyhedral structure of the PCGTS polytope. To elaborate this task, we employ the classic approach relying on *dimensions* of the studied polytope and its faces.

By definition, for an arbitrary polytope P, its dimension is equal to the dimension of its affine hull $\dim P = \dim(\text{aff}(P))$, which in turn is one less than the number of affinely independent extreme points this polytope.

An intersection of a polytope P with an arbitrary *support* hyperplane is called a *face* of this polytope. Usually, for the sake of convenience, the family of faces of a polytope is extended by *improper* faces $\varnothing$ and P. A face F of a polytope P is called a *facet* (of this polytope), if $\dim F = \dim P - 1$.

The PCGTSP is an extension of an Equality GTSP (E-GTSP) introduced in Fischetti et al. (1995), where E-GTSP polytope was denoted by $P^{=}$. Therefore, we keep the same notation for the PCGTS polytope, i.e. the convex hull of the incidence vectors $[x, z]$ encoding all the feasible tours of the problem in question. As it follows from Observation 1, $P^{=} = \text{conv}\{[x, z] \in \mathbb{R}^{E \cup V} : (7) - (16) \text{ holds}\}$. Since $[x, z]$ could be obviously extended to the feasible solution $[x, z, u, y]$ of (7)-(16), the polytope $P^{=}$ is isomorphic to the convex hull of the feasible set of the initial non-relaxed MILP model from Subsection 2.2. In the sequel, for the simplicity, we will not distinguish them. Our goal is to derive conditions sufficient for an arbitrary inequality

$$\alpha^T x - \beta^T z \geqslant \gamma \tag{40}$$

to induce a facet of the polytope $P^{=}$.

4.1 Dimension of the PCGTS polytope

In this section, we prove the following

Theorem 1. *For any instance of PCGTSP, the following equation:*

$$\dim(P^{=}) = |E| - n - m + 1 \tag{41}$$

holds.

To prove Theorem 1, we employ an inductive approach similar to Fischetti et al. (1995) on the number of excessive nodes ρ within clusters:

$$\rho = \sum_{h=1}^{m} (|C_h| - 1) = n - m. \tag{42}$$

Here, the base case $\rho = 0$ corresponds to the Precedence Constrained Asymmetric Traveling Salesman Problem (PCATSP) and follows from

Theorem 2 (Balas et al. (1995))**.** *For an arbitrary instance of PCATSP, dimension of its polytope* $P^{=}_{ATSP}$ *is as follows:* $\dim P^{=}_{ATSP} = |E| - 2n + 1$.

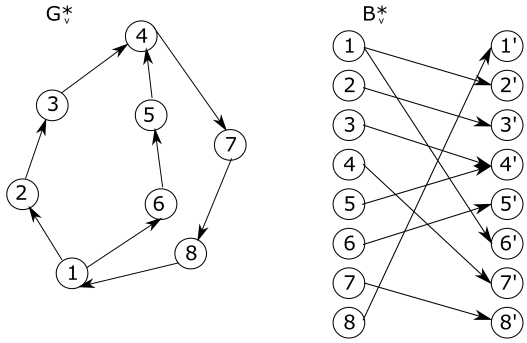

Figure 1: Example of a directed graph and its bipartite representation

Remark 1. In the paper by Balas et al. (1995), the polytope is denoted in $\mathbb{R}^E$. However, it can be unambigiously extended to $\mathbb{R}^{E\cup V}$ by setting $z_v = 1$ for each node $v \in V$ as it was done in Fischetti et al. (1995).

In order to prove the inductive step, we need additional notation and technical lemmas. Let inequality (40) be valid for $P^=$, i.e. $P^= \subset \{[x,z] \in \mathbb{R}^{E\cup V} : \alpha^T x \geqslant \beta^T z + \gamma\}$. Consider the appropriate face $\mathcal{H}(\alpha,\beta,\gamma) = P^= \cap \{[x,z] \in \mathbb{R}^{E\cup V} : \alpha^T x = \beta^T z + \gamma\}$ of the polytope $P^=$.

Further, to any non-individual node $v \in V$, we assign:
(i) a PCGTSP polytope $P_v^=$ associated with the subgraph of G induced by $V \setminus \{v\}$,
(ii) the v-restriction of inequality (40) obtained by dropping variables z_v and x_e for all $e \in \delta(v)$,
(iii) the v-compatibility digraph of (40) $G_v^* = (V \setminus C(v), E_v^*)$, where

$$E_v^* = \{(i,j) \colon i,j \in V \setminus C(v), i \neq j, \exists\, [x,z] \in \mathcal{H}(\alpha,\beta,\gamma),\ x_{iv} = x_{vj} = 1\},$$

(iv) its bipartite representation B_v^* (see Bang-Jensen and Gutin (2009) and Fig. 1).

Lemma 3. *For any valid inequality $\alpha^T x \geqslant \beta^T z + \gamma$, and an arbitrary non-individual node $v \in V$, $\dim \mathcal{H}(\alpha,\beta,\gamma) \geqslant \dim \mathcal{H}(\alpha,\beta,\gamma)_v + rank(B_v^*)$ where $\mathcal{H}(\alpha,\beta,\gamma)_v$ is the face of polytope $P_v^=$ induced by its v-restriction.*

Proof. Consider the matrix M, whose rows are extreme points of the face $\mathcal{H}(\alpha,\beta,\gamma)$ (Fig. 2). By construction, $\mathcal{H}(\alpha,\beta,\gamma)$ is contained in a hyperplane of $\mathbb{R}^{E\cup V}$ not passing through the origin (due to equation (7)). Therefore, for any subset of rows of M, the affine independence is equivalent to the linear one. Thus, $\dim \mathcal{H}(\alpha,\beta,\gamma) = \operatorname{rank}(M) - 1$.

Matrix M can be represented as follows:

$$M = \begin{pmatrix} M_{11} & 0 & 0 \\ M_{21} & M_{22} & 1 \end{pmatrix},$$

where the last column corresponds to node v, and the columns left to it correspond to the arcs incident with v. By construction, block M_{11} corresponds to the extreme points of face $\mathcal{H}(\alpha,\beta,\gamma)_v$. Thus, $\operatorname{rank}(M_{11}) = \dim \mathcal{H}(\alpha,\beta,\gamma)_v + 1$.

On the other hand, matrix M_{22} is located in the part of the tour visiting node v. By construction, it should visit it only once. Therefore, each row of M_{22} has exactly two 1s. Consider an arbitrary row of block M_{22}. Suppose that 1s are located in the columns (i,v) and (v,j). Hence, in graph B_v^*, nodes i and j are adjacent and the considered row is a column in the incidence matrix $M_{B_v^*}$ of B_v^*. Thus, $M_{22} = M_{B_v^*}^T$ (see Fig. 3).

Therefore, $\operatorname{rank}(M_{22}) = \operatorname{rank}(B_v^*) = N_{B_v^*} - \kappa(B_v^*)$, where $N_{B_v^*}$ is a size of the nodeset of bipartite graph B_v^* and $\kappa(B_v^*)$ is the number of its connected components (see e.g. Biggs (1974)).

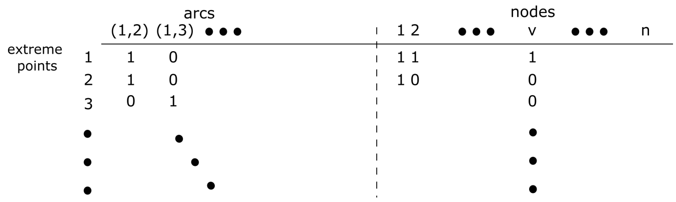

Figure 2: Matrix of extreme points of $\mathcal{H}(\alpha, \beta, \gamma)$

Finally, $\text{rank}(M) \geqslant \text{rank}(M_{11}) + \text{rank}(M_{22})$. Since $\text{rank}(M_{11}) = \dim \mathcal{H}(\alpha, \beta, \gamma)_v + 1$, $\text{rank}(M_{22}) = \text{rank}(B_v^*)$. Lemma 1 is proved. □

The claim of Lemma 1 is valid for an arbitrary face $\mathcal{H}(\alpha, \beta, \gamma)$. Now, to determine dimension of polytope $P^=$, we consider its improper face $\mathcal{H}(0, 0, 0) = P^=$. To emphasize the associated bipartite graph B_v^* in this special case, denote it by $\bar{B}_v^*$.

Lemma 4. *For any non-individual node v, rank$(\bar{B}_v^*) = |\delta(v)| - 1$.*

Proof. We prove Lemma 4 by enumeration of all the possible options to relate cluster $C(v)$ with the given precedence constraints. In the sequel, we use the following notation. By $\tilde{\pi}$ and $\tilde{\sigma}$, for cluster $C(v)$, we denote subsets of nodes belonging to its direct parents and children, respectively. Similarly, we introduce subsets $\hat{\pi}$ and $\hat{\sigma}$ of nodes that belong to other ancestors and descendants of this cluster. In addition, by r, we denote a union of all clusters except C_{Balas} incompatible with $C(v)$.

Observation 2. *For any cluster, its parents (if any) are mutually incomparable. For its children the same claim is valid as well.*

Case 1 ($\tilde{\pi} \neq \varnothing$ and $\tilde{\sigma} = \varnothing$). In this case, cluster $C(v)$ is one of the minimal descendants in the given partial order. Here, for cut $\delta(v)$ in graph G (see Fig. 4), we have $|\delta(v)| = |C_1| + |\tilde{\pi}| + 2|r| + 2$, since $|C_{Balas}| = 1$. Consider the appropriate bipartite graph $\bar{B}_v^*$ (Fig. 5). It has

$N_{\bar{B}_v^*} = 2|C_1| + 2|\hat{\pi}| + 2|\tilde{\pi}| + 2|r| + 2$ nodes. By definition, an arbitrary node i from the left part and j' from the right part of graph $\bar{B}_v^*$ are incident if and only if there is a feasible tour with the fragment i-v-j. If such an arc exists, then graph $\bar{B}_v^*$ has a complete bipartite subgraph, whose parts are induced by clusters $C(i)$ and $C(j)$. In Fig. 5, we encode such subgraphs by straight line segments. By construction, all non-isolated nodes of $\bar{B}_v^*$ belong to the only connected component. Furthermore, the number of connected components is $\kappa(\bar{B}_v^*) = 1 + |C_1| + 2|\hat{\pi}| + |\tilde{\pi}|$. Indeed, for instance, verify the incidence of some $i \in \tilde{\pi}$ and node j' corresponding to the only node of cluster C_{Balas}. Take an arbitrary node $v_1 \in C_1$ from the depot and construct a feasible tour as follows. Departing from v_1 the tour visits all the clusters preceding $C(v)$ such that the cluster $C(i)$ is visited last, at node i. Then, we traverse arcs (i, v) and (v, j) directly, visit all the remaining clusters (respecting the precedence constraints) and complete the tour by returning to node v_1.

Finally, $\text{rank}(\bar{B}_v^*) = N_{\bar{B}_v^*} - \kappa(\bar{B}_v^*) = |C_1| + |\tilde{\pi}| + 2|r| + 1 = |\delta(v)| - 1$.

Case 2 ($\tilde{\sigma} \neq \varnothing$ and $\tilde{\pi} = \varnothing$). This case is dual to Case 1, here $C(v)$ is the maximal ancestor in the partial order. In the similar sense (see Fig. 6 and Fig. 7), we obtain $|\delta(v)| =$

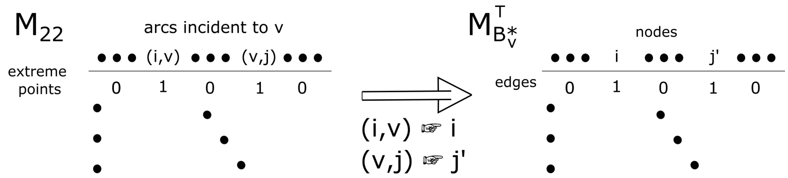

Figure 3: Block M_{22} and the incidence matrix of B_v^*

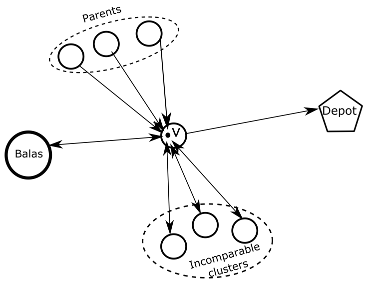

Figure 4: Cut $\delta(v)$ for Case 1

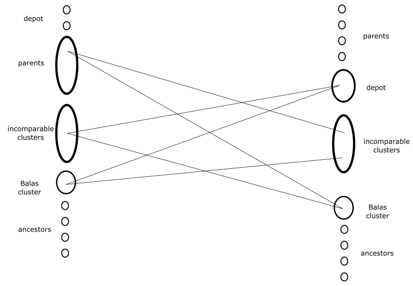

Figure 5: Bipartite graph $\bar{B}_v^*$ for Case 1

$|C_1| + |\tilde{\sigma}| + 2|r| + 2$, $N_{\bar{B}_v^*} = 2|C_1| + 2|\hat{\sigma}| + 2|\tilde{\sigma}| + 2|r| + 2$, $\kappa(\bar{B}_v^*) = 1 + |C_1| + 2|\hat{\sigma}| + |\tilde{\sigma}|$, and finally, $\mathrm{rank}(\bar{B}_v^*) = |C_1| + |\tilde{\sigma}| + 2|r| + 1 = |\delta(v)| - 1$.

Case 3 ($\tilde{\pi} \neq \varnothing$, $\tilde{\sigma} \neq \varnothing$). The only difference here is that cluster $C(v)$ has both parents and children, which slightly impacts the structure of $\bar{B}_v^*$. Proceeding with the proof in a similar way, we obtain (Fig. 8 and Fig. 9): $|\delta(v)| = |\tilde{\pi}|+|\tilde{\sigma}|+2|r|+2$, $N_{\bar{B}_v^*} = 2(|C_1|+|\hat{\sigma}|+|\tilde{\sigma}|+|\hat{\pi}|+|\tilde{\pi}|+|r|+1)$, $\kappa(\bar{B}_v^*) = 1 + 2|C_1| + |\tilde{\sigma}| + |\tilde{\pi}| + 2|\hat{\pi}| + 2|\hat{\sigma}|, \mathit{and}\ \mathrm{rank}(\bar{B}_v^*) = 2|\tilde{\pi}| + 2|\tilde{\sigma}| + 2|r| + 1 = |\delta(v)| - 1$.

Case 4 ($\tilde{\pi} = \varnothing$, $\tilde{\sigma} = \varnothing$). Without loss of generality, we restrict ourselves to the case where the set of free clusters is exhausted by $C(v)$ and C_{Balas} (if the set of free clusters has more than two elements, the case is similar to this one). Since this case is different from the discussed above, we provide an argument in detail. Since v is a non-individual node, $C(v) \neq C_{Balas}$. Again for cut $\delta(v)$, we have $|\delta(v)| = 2|C_1| + 2|r| + 2$. We show that in this case $\bar{B}_v^*$ is a connected bipartite graph. We skip the trivial option of the empty order, since here $\bar{B}_v^*$ is a complete graph. Otherwise, there are always at least two clusters C_p and C_q, such that C_p is the parent of C_q. Obviously, these clusters induce a complete bipartite subgraph of graph $\bar{B}_v^*$. Since both copies of C_{Balas} are incident with all other clusters from the opposite part, $\bar{B}_v^*$ is connected (see Fig. 10). Finally, we obtain $N_{\bar{B}_v^*} = 2|C_1|+2|r|+2$, $\kappa(\bar{B}_v^*) = 1$, and $\mathrm{rank}(\bar{B}_v^*) = 2|C_1|+2|r|+1 = |\delta(v)|-1$.

Case 5 ($C(v) = C_1$). This is another unique case. To proceed with our proof, we need additional notation. By Σ, we denote the set of all nodes from minimal descendants, Π consists of all nodes from maximal ancestors in the given partial order. Also, let F be the set of all nodes from free clusters, except C_{Balas}, and R are the remaining nodes. Then, $|\delta(v)| = 2|F| + |\Pi| + |\Sigma| + 2$.

As for the graph $\bar{B}_v^*$, it is constructed in the same sense as for the previous cases. The only difference here, is that the depot is departure and arrival node at the same time. However, this won't be a problem, since any feasible tour is closed (see Fig. 11). Finally, $N_{\bar{B}_v^*} = 2(|R| + |\Pi| + |F| + 1 + |\Sigma|)$, $\kappa(\bar{B}_v^*) = 1 + |\Pi| + |\Sigma| + 2|R|$, and $\mathrm{rank}(\bar{B}_v^*) = |\delta(v)| - 1$. Lemma 4 is proved. □

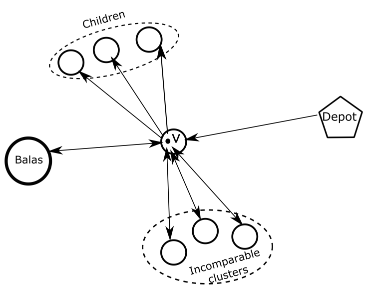

Figure 6: Cut $\delta(v)$ for Case 2

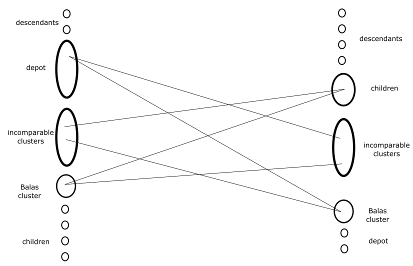

Figure 7: Bipartite graph $\bar{B}_v^*$ for Case 2

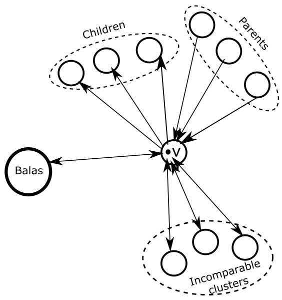

Figure 8: Cut $\delta(v)$ for Case 3

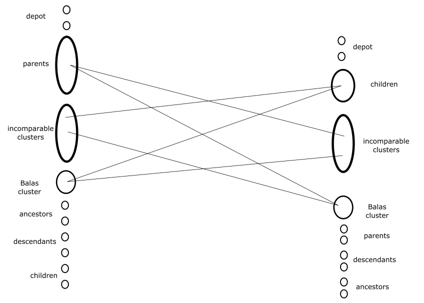

Figure 9: Bipartite graph $\bar{B}_v^*$ for Case 3

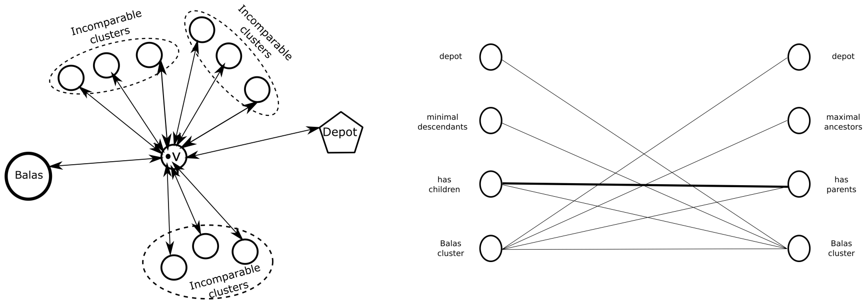

Figure 10: Representation of $\delta(v)$ and the connected components of $\bar{B}_v^*$ for Case 4. Bold line provides connectivity of the $\bar{B}_v^*$

Now, we ready to establish dimension of the polytope $P^=$ and prove Theorem 1.

Proof. By construction, the PCGTS polytope $P^=$ is a part of a solution set of inequality system (7)-(11). Hence, $\dim P^=$ cannot be greater than dimension of this solution set. In turn, for an arbitrary feasible system of linear equations $Ax = b$ with $m \times d$ coefficient matrix, dimension of its solution set is $d - \operatorname{rank}(A)$.

Let A be the coefficient matrix of system (7)-(11) (Fig. 12). By construction, $\operatorname{rank}(A) \geqslant \operatorname{rank}(D) + \operatorname{rank}(K) = \operatorname{rank}(D) + m$. We show that $\operatorname{rank}(D) = 2n - 1$. To the initial graph G, we assign cluster digraph $G_c = (\mathcal{C}, E_c)$, for which $(C', C'') \in E_c$ if and only if there exist $i \in C'$ and $j \in C''$, such that $(i, j) \in E$. Let B_G and B_{G_c} be bipartite representations of digraphs G and G_c respectively.

Observation 3. *Evidently, if* $(C', C'') \in E_c$*, then* $(i, j) \in E \; \forall i \in C' \; \forall j \in C''$.

Observation 4. *By construction,* D *is the incidence matrix of* B_G.

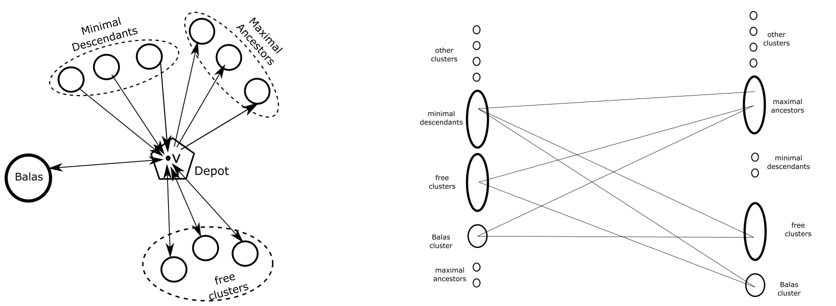

Figure 11: Representation of $\delta(v)$ and the connected components of the $\bar{B}_v^*$ for Case 5

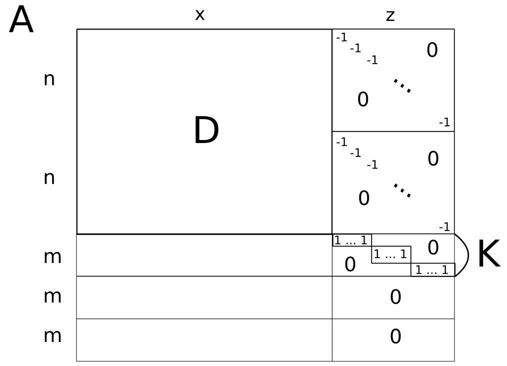

Figure 12: Structure of matrix A.

As a simple consequence, we obtain that graphs B_G and B_{G_c} have the same number of connected components.

Since the initial graph G has at least one free cluster, then by Proposition 5.3 from Balas et al. (1995), B_{G_c} is connected, i.e. $\kappa(B_{G_c}) = \kappa(B_G) = 1$, $\text{rank}(D) = 2n-1$, and $\text{rank}(A) \geqslant 2n+m-1$. Therefore,

$$\dim P^{=} \leqslant |E| + |V| - \text{rank}(A) \leqslant |E| + n - 2n - m + 1 = |E| - n - m + 1. \tag{43}$$

To complete the argument, we need to prove the lower bound

$$\dim P^{=} \geqslant |E| - n - m + 1. \tag{44}$$

We proceed with induction on the number of excessive nodes within clusters:

$$\rho = \sum_{h=1}^{m} (|C_h| - 1) = n - m$$

Base Case ($\rho = 0$) follows from Theorem 2 for the PCATSP.

Inductive Step. Assume that inequality (44) holds for some ρ. To prove it for $\rho + 1$, take an arbitrary non-individual node v. By Lemma 3 and Lemma 4,

$$\dim P^{=} \geqslant \dim P_v^{=} + \text{rank}(\bar{B}_v^*) = \dim P_v^{=} + |\delta(v)| - 1.$$

Recall that $P_v^{=}$ corresponds to the graph of $n-1$ nodes and $|E \setminus \delta(v)|$ arcs. By induction hypothesis, $\dim P_v^{=} \geqslant |E| - |\delta(v)| - n - m + 2$, and the claim follows.

Combination of (43) and (44) concludes the proof. □

4.2 Facet-inducing inequalities

By extending the results obtained in Balas et al. (1995), in this subsection we establish the sufficient conditions ensuring that π- and σ-inequalities ((17) and (18)) introduced in Subsection 3.1 are facet-inducing.

Theorem 5. *For $\mathcal{S} \subset \mathcal{C} \setminus \{C_1, C_{Balas}\}$ and $\bar{\mathcal{S}} = \mathcal{C} \setminus \mathcal{S}$, an inequality*

$$x(\mathcal{S} \setminus \pi(\mathcal{S}), \bar{\mathcal{S}}) \geqslant 1 \tag{45}$$

induces a facet of the polytope $P^{=}$, if $\pi(\mathcal{S}) \subset \mathcal{S}$, $\sigma(\mathcal{S}) \subset \mathcal{S}$, and $\mathcal{S}$ contains at least 3 free clusters.

Theorem 6. *For $\mathcal{S} \subset \mathcal{C} \setminus \{C_1, C_{Balas}\}$ and $\bar{\mathcal{S}} = \mathcal{C} \setminus \mathcal{S}$, an inequality*

$$x(\bar{\mathcal{S}}, \mathcal{S} \setminus \sigma(\mathcal{S})) \geqslant 1 \tag{46}$$

induces a facet of the polytope $P^{=}$, if $\pi(\mathcal{S}) \subset \mathcal{S}$, $\sigma(\mathcal{S}) \subset \mathcal{S}$, and $\mathcal{S}$ contains at least 3 free clusters.

Similarly to Theorem 1, our proof is based on the inductive framework developed in Fischetti et al. (1995) for the symmetric GTSP. The induction is carried out on the number of excessive nodes (42) in clusters. Since the base case corresponds to the classic PCATSP, our claim follows from the known result (Balas et al. (1995), Theorem 5.5). In turn, proof of the inductive step relies on Lemma 3 and our adaptation of Lemma 4 to the case of the proper face $\mathcal{H}^\pi = \mathcal{H}(\alpha, \beta, \gamma)$, where

$$\beta = 0,\ \gamma = 1, \quad \alpha_{i,j} = \begin{cases} 1, & \exists C_p \in \mathcal{S} \setminus \pi(\mathcal{S}), \exists C_q \in \bar{\mathcal{S}} \colon i \in C_p, j \in C_q, \\ 0, & \text{otherwise} \end{cases}$$

induced by inequality (45).

Lemma 7. *Let $\mathcal{H}^\pi$ be the face of $P^{=}$ induced by π-inequality* (45)*. The hypothesis of Theorem 5 implies that, for an arbitrary non-individual node v, rank$(B_v^*) = |\delta(v)| - 1$.*

Proof. Our argument is based on enumeration of all the possible options to establish a relation between cluster $C(v)$ and the given partial order. Previously, in the proof of Lemma 4, for each case, we explored properties of the associated cut $\delta(v)$ and bipartite graph $\bar{B}_v^*$. Now, each of these options can be split into several sub-options in correspondence to the ways to locate $C(v)$ with respect to the face $\mathcal{H}^\pi$ (see Table 1).

Table 1: Options for cluster $C(v)$.

case #	relation to the partial order	relation to the face $\mathcal{H}_\pi$
1	minimal descendant	$\mathcal{S}' = \mathcal{S} \setminus \pi(\mathcal{S})$, $\bar{\mathcal{S}}$
2	maximal ancestor	$\pi(\mathcal{S})$, $\bar{\mathcal{S}}$
3	has parents and children	$\pi(\mathcal{S})$, $\bar{\mathcal{S}}$
4	free cluster	$\mathcal{S}'$, $\bar{\mathcal{S}}$
5	depot	$\bar{\mathcal{S}}$

It is easy to verify that all subcases of any unique case presented at a single line of Table 1 share the same cut $\delta(v)$, while their associated bipartite graphs B_v^* are spanning subgraphs of graph $\bar{B}_v^*$ constructed in Lemma 4 for the entire polytope $P^{=}$. In its proof, we showed that, for any v, graph $\bar{B}_v^*$ contains a single connected component. Therefore, to prove Lemma 7, it is sufficient to show that the same node subset induces a connected component in any mentioned graph B_v^* as well.

For the sake of brevity, we restrict ourselves to cases 3 and 4 (see Table 1), since they appear to be the most common. For the other cases, the argument can be obtained in a similar way.

As in the proof of Lemma 4, by $\tilde{\pi}$, $\tilde{\sigma}$, and r, we denote the subsets of nodes (in graph G) belonging to parent, child and incomparable clusters with respect to cluster $C(v)$, respectively.

Case 3 ($\tilde{\pi} \neq \varnothing$, $\tilde{\sigma} \neq \varnothing$). In both subcases, for $C(v) \in \pi(\mathcal{S})$ and $C(v) \in \bar{\mathcal{S}}$, we verify the connectivity of the subgraphs induced by the connected component found in the proof of Lemma 4, Case 3. We present these subgraphs in Fig. 13 in more detail. To prove their connectivity, it is sufficient (a) to show that the single node from C_{Balas} is adjacent to any other node from the opposite part of graph B_v^*; (b) to present at least one additional arc connecting nodes from any two clusters other than C_{Balas}.

(a) For instance, we establish the existence of an arc connecting node $i \in C_{Balas}$ and some node j belonging to some child cluster $C \in \mathcal{S}' = \mathcal{S} \setminus \pi(\mathcal{S})$ of cluster $C(v)$ (Fig. 13). Departing from the depot, we start with construction of a tour T by visiting all the clusters in $\bar{S}$ except C_{Balas} (regarding the precedence constraints). Then, we proceed with all the ancestors of cluster $C(j)$ except $C(v)$. This is possible due to Proposition 2.

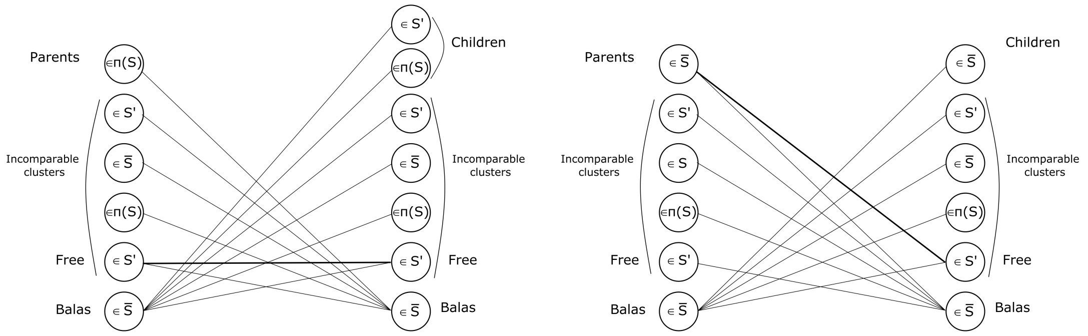

Figure 13: Connected components of B_v^* for Case 3. $C(v) \in \bar{\mathcal{S}}$ (left) and $C(v) \in \bar{\mathcal{S}}$ (right)

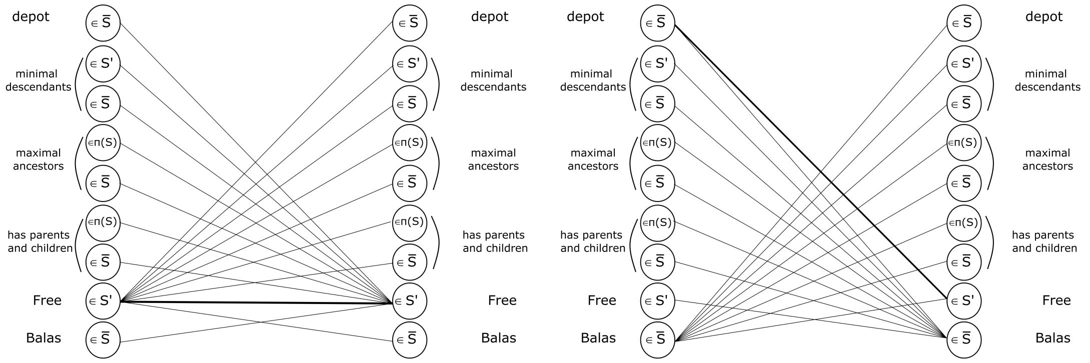

Figure 14: Connected components of B_v^* for Case 4. $C(v) \in \bar{\mathcal{S}}'$ (left) and $C(v) \in \bar{\mathcal{S}}$ (right)

Further, we traverse the i-v-j fragment and proceed with visiting all the remaining clusters in $\pi(\mathcal{S})$. Finally, we randomly visit all the clusters in $\mathcal{S}'$ and return to the depot by a direct arc. By construction, it is the only arc in the proposed tour that belongs to the cut $\delta^+(\mathcal{S} \setminus \pi(\mathcal{S}), \bar{\mathcal{S}})$ (in graph G). Therefore, for this tour, inequality (45) becomes tight.

(b) Without loss of generality, provide an argument for subcase $C(v) \in \bar{\mathcal{S}}$ (Fig. 13). Let i be any node from some parent $C(i)$ of $C(v)$, and j belongs to a free cluster $C(j)$. Again, we consider the tour T departing from an arbitrary depot node. We visit all the ancestors of $C(v)$, except $C(i)$. Next, we pass through the i-v-j fragment and continue from $C(j)$ by visiting all the clusters in $\pi(\mathcal{S})$. Then, we proceed with traveling over the rest of $\mathcal{S}'$. Finally, we return to $\bar{\mathcal{S}}$ by an arc that belongs to the cut $\delta^+(\mathcal{S} \setminus \pi(\mathcal{S}), \bar{\mathcal{S}})$, and complete the tour by visiting the remaining clusters, arriving at the depot.

Case 4 ($\tilde{\pi} = \varnothing$, $\tilde{\sigma} = \varnothing$). Generally speaking, the argument for this case is close to the previous one. However, we mention it separately, since this case appears to be the only reason for requiring at least three free clusters from $\mathcal{S}'$ in the hypothesis of Theorem 5. As it follows from Fig. 14, for $C(v) \in \mathcal{S}'$, cluster C_{Balas} does no longer induce a dominating set in the considered subgraph (of graph B_v^*). Instead, free clusters take its place.

Furthermore, these free clusters ensure the connectivity of the subgraph. Indeed, consider free clusters $C(i), C(j) \in \mathcal{S}'$, such that $C(i) \neq C(v) \neq C(j)$. Construct a feasible tour T with the fragment i-v-j in graph G. Since $C(i), C(v)$ and $C(j)$ are free and belong to $\mathcal{S}'$, we are allowed to move i-v-j directly after the departure from the depot. Then, after visiting all the clusters in $\pi(\mathcal{S})$, we come to the remaining clusters from $\mathcal{S}'$, cross the border between $\mathcal{S}'$ and $\bar{\mathcal{S}}$ (at once), move through all the clusters in $\bar{\mathcal{S}}$ and return to the depot.

In subcase $C(v) \in \bar{\mathcal{S}}$ (Fig. 14), the proof can be obtained in a similar way to the Case 3. Lemma 7 is proved. $\square$

Now, we are ready to establish the proof of Theorem 5.

Proof. Let $\mathcal{H}^\pi$ be the face of polytope $P^=$ induced by π-inequality. By Theorem 1, we have $\dim \mathcal{H}^\pi \leqslant \dim P^= = |E| - n - m + 1$. By induction on number ρ (see eqn. (42)), we show that

$$\dim \mathcal{H}^\pi \geqslant |E| - n - m. \tag{47}$$

Base case of $(\rho = 0)$ is proved in Balas et al. (1995), since, in this case, the problem at hand is equivalent to the PCATSP.

Inductive step. Assuming that (47) holds for some ρ, prove it for $\rho+1$. Combining claims of Lemma 3 and Lemma 7, we have $\dim \mathcal{H}^\pi \geqslant \dim \mathcal{H}^\pi_v + \operatorname{rank}(B^*_v) = \dim \mathcal{H}^\pi_v + |\delta(v)| - 1$. Since, by induction hypothesis, $\dim \mathcal{H}^\pi_v \geqslant |E| - |\delta(v)| - n + 1 - m$, we obtain the desired lower bound (47).

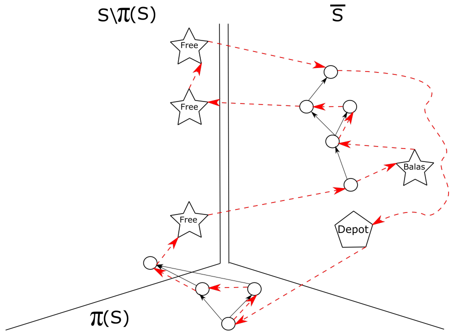

Figure 15: Example of the contradicting solution

To finalize the proof, we show that inequality (47) is tight. Indeed, suppose by contradiction that it is not. But, under this assumption, $\dim \mathcal{H}^\pi = \dim P^=$ and, consequently the face $\mathcal{H}^\pi$ coincides with the polytope $P^=$. However, we can always provide a feasible solution crossing the outgoing cut $\delta^+(\mathcal{S} \setminus \pi(\mathcal{S}))$ at least twice (see, e.g., Fig. 15). Theorem 5 follows from the obtained contradiction. □

For the sake of brevity, we omit the proof of Theorem 6, which can be obtained in a similar way.

5 Formulations

In this section we describe novel MILP-models (formulations) for the PCGTSP. Almost all of them are extensions of the known formulations proposed initially in Gouveia and Pesneau (2006); Gouveia and Ruthmair (2015); Gouveia et al. (2018) for the PCATSP and incorporate exponential size families of valid inequalities introduced for the PCGTSP in Section 3.

Following to Gouveia et al. (2018), we start with the sequence of models obtained incrementally as follows:
- M_1 is our basic compact model described in Subsection 2.2,
- M_2 is M_1 augmented with strengthened simple-cut (27)–(29) and both strengthened regular (30), and reversed GDDL (31) inequalities,
- M_3 is M_2 with strengthened 2-path inequalities (32)–(35),
- M_4 is M_3 enforced by strengthened 3v GDDL-like inequalities (36)–(38),
- M_5 is M_4 supplied with strengthened 4v GDDL-like inequalities (39).

In addition, we propose the model incorporating the inequalities described in Subsection 3.1, Subsection 3.2, and Subsection 3.3:

- M_1^* is M_1 augmented with π-, σ-, and (π, σ)-inequalities (17)–(19), precedence cycle breaking inequalities (20) and (21), and inequalities (22)–(26).

In order to increase the tightness of the lower bounds, we combine M_1^* with other best performers of our exploratory Experiment I (see Subsection 7.2):
- M_3^*, which is M_1^* + M_3 and
- $M_5^* = M_1^* + M_5$.

In all these models, families of valid inequalities are separated exactly, following to the incremental pattern proposed in Gouveia et al. (2018). Although the models M_3 and M_5 clearly benefit from the combination with M_1^* in terms of the lower bounds, they still remain to be rather time-consuming.

Therefore, by evolving the well-known *roulette-wheel* sampling principle (see ex. Gendreau and Potvin (2019)) and simple *online learning* technique, we propose a novel heuristic separation procedure and the corresponding models M_{3s}^* and M_{5s}^*, which we call *sampled* as well. The main idea of the proposed procedure is as follows:
- to each family of valid inequalities, we assign an appropriate probabilistic measure; for instance, in the case of 3v GDDL-like inequalities (36)-(38), it is sufficient to define a discrete distribution on the set of ordered quadruples (C_p, C_q, C_r, C_s) of non-depot clusters;
- given by a sample size, at each separation epoch, we apply cut generation technique at this epoch only to the entries of a sample drawn from the defined distribution;
- each time, when a tuple managed to produce a cut, we increase its probability.

Generally speaking, the proposed separation heuristic is a compromise between the tightness of the LP-relaxation bounds and numerical performance. However, we decide to evaluate it in our experiments along with the known incremental separation pattern, because the sampling gives us an opportunity to adopt powerful but large families of valid inequalities from the very beginning of the LP-relaxation solution process.

6 Branch-and-Cut Algorithm

Our branch-and-cut algorithm extends the algorithm proposed in Gouveia and Ruthmair (2015) for the SOP and has a component-wise structure based on few main building blocks. Among them are instance preprocessing routine, primal heuristic, and a formulation of the problem in question that specifies a family of cutting planes.

In its current version, the proposed algorithm is restricted to use the same instance preprocessing routine. The arcs violating precedence constraints are excluded from the given graph by preprocessing rules (1)-(5), previously introduced in Balas et al. (1995) for the PCATSP. In addition, as the only primal heuristic, the algorithm uses PCGLNS, proposed in Khachay et al. (2020) and briefly described in Subsection 6.1. Thus, all the proposed variants of the algorithm (refer to Subsection 7.3 for details) were obtained by varying the problem formulation.

6.1 PCGLNS Primal Heuristic

The PCGLNS heuristic extends the recent GLNS algorithm proposed in Smith and Imeson (2017) for the common GTSP. PCGLNS is designed to take into account additional precedence constraints defined on a set of clusters. In a nutshell, PCGLNS appears to be an original implementation of the seminal Adaptive Large Neighborhood Search (ALNS) metaheuristic (see, e.g. Gendreau and Potvin (2019)) and combines the well-known *ruin* and *recreate* principle with online learning over a given sets of basic removal and insertion local search heuristics.

Table 2: PCGTSPLIB library.

instance	n	m	PC density	instance	n	m	PC density
ESC07	39	8	14	p43.1	203	43	53
ESC12	65	13	23	p43.2	198	43	76
ESC25	133	26	36	p43.3	211	43	138
ESC47	244	48	79	p43.4	204	43	538
ESC63	349	64	296	prob.100	510	99	139
ESC78	414	79	361	prob.42	208	41	59
br17.10	88	17	31	rbg048a	255	49	495
br17.12	92	17	38	rbg050c	259	51	558
ft53.1	281	53	64	rbg109a	573	110	5438
ft53.2	274	53	82	rbg150a	871	151	10484
ft53.3	281	53	269	rbg174a	962	175	14129
ft53.4	275	53	811	rbg253a	1389	254	30434
ft70.1	346	70	86	rbg323a	1825	324	48525
ft70.2	351	70	117	rbg341a	1822	342	56644
ft70.3	347	70	284	rbg358a	1967	359	56894
ft70.4	353	70	1394	rbg378a	1973	379	63963
kro124p.1	514	100	132	ry48p.1	256	48	59
kro124p.2	524	100	169	ry48p.2	250	48	73
kro124p.3	534	100	365	ry48p.3	254	48	179
kro124p.4	526	100	2404	ry48p.4	249	48	643

n and m are the number of nodes and clusters respectively
'PC density' is the number of arcs in the transitively closed precedence DAG

6.2 Implementation

The proposed algorithm is implemented on top of the Gurobi 9.3 framework. Primal heuristic and cutting planes are provided as user callback functions. For the sampled models, all the parameters of the heuristic separation including sample sizes and learning rates are tuned within preliminary testing stage. All the built-in Gurobi heuristics and cutting plane algorithms are disabled, while other parameters of the solver keep their default values. The suggested implementation is carried out in Python 3 leveraging NetworkX software package for internal graph processing tasks and fully cross-platform. All source code together with the reported experimental results are open for public access at Khachai (2022).

7 Numerical evaluation

In this section, we report results of the competitive numerical experiments that show how each proposed formulation and variant of the branch-and-cut algorithm could be useful for the PCGTSP. In particular, these results reveal the notable impact contributed by predecessor/successor inequalities in terms of accuracy and running time, which can be considered as an additional support of the theoretical results obtained in Section 4. We proceed with two separate experiments. In the former one, we evaluate the proposed formulations with respect to their LP-relaxation bounds and the time consumption. In turn, the purpose of the latter one is to compare the best performers of the first experiment with known results within the branch-and-cut setting. All the computations are carried out on the 16-core Intel Xeon 128G RAM server [2] against the same public benchmark library PCGTSPLIB.

7.1 PCGTSPLIB Benchmark library

The PCGTSPLIB library was derived in Salman et al. (2020) from the well-known SOPLIB library in order to provide a test-bed for PCGTSP. To the best of our knowledge, it is the only public library for the problem in question. We provide a short overview of this library in Table 2.

Since computational complexity of the PCGTSP depends mostly on the number of clusters m (rather than the size of a node set n, as it is for SOP), it is convenient to partition all 40 instances of this library into small (up to 30 clusters), medium (up to 70 clusters), large (up to 120 clusters), and huge ones (more than 120 clusters). In addition, the instances differ substantially in terms the density of the constituent partial orders.

For each instance, we round the transportation costs to the nearest integral values. For the sake of convenience, we provide the converted instances along with our source codes Khachai

[2] provided by Supercomputer 'Uran' at Krasovsky Institute of Mathematics and Mechanics

(2022).

7.2 Experiment I: Comparison of the LP-relaxations

Inspired by the results of Gouveia et al. (2018), we start with the comparison of the formulations M_1–M_5 and M_1^* in terms of their LP-relaxation bounds and time complexity. In this experiment, for each competing model, computation time was limited to 10 hours (36000 seconds).

Since the separation procedures for M_2–M_5 follow the incremental pattern initially proposed for the SOP, more complex formulations provide tighter lower bounds, perhaps with substantially increased computation time. Therefore, for each instance, whose optimum value is achieved by some M_i model, we do not solve it by M_j, for any $j > i$. As it follows from Table 3, the optimum values were found for 8 out of 40 instances: for ESC63 - by model M_1, for br17.10 and br17.12 - by model M_2, for other five - by model M_1^* (along with M_2 and M_5 for rbg048a and rbg050c, respectively). For the remaining instances, M_2 found the tightest lower bound once, M_3 three times as well as M_5, and M_1^* — 25 times.

Table 3: Comparison of formulations M_1–M_5 and M_1^*.

Instance	OPT	M_1		M_2		M_3		M_4		M_5		M_1^*	
		LPB	t	LPB	t	LPB	t	LPB	t	LPB	t	LPB	t
ESC07	1730	1683	0	1730	0.28	1730	0.29	1730	0.29	1730	0.3	$\underline{\mathbf{1730}}$	0.09
ESC12	1390	1238	0.02	1387	5.91	1387	7.1	1387	8.21	1387	9.97	$\underline{\mathbf{1390}}$	0.8
ESC25	1383	1296	0.21	1362	229	1362	300	1362	364	1362	448	$\underline{1363}$	7
ESC47	1063	1001	8.46	1012	2982	1012	3545	1013	4655	1016	7420	$\underline{1023}$	119
ESC63	62	$\underline{\mathbf{62}}$	207.36	–	–	–	–	–	–	–	–	62	318
ESC78	[14673, 14808]	14629	3829.32	14640	23287	14641	36000	14641	36000	14641	36000	$\underline{14659}$	5477
br17.10	43	15	0.05	$\underline{\mathbf{43}}$	6.81	–	–	–	–	–	–	32	5
br17.12	43	15	0.05	$\underline{\mathbf{43}}$	6.54	–	–	–	–	–	–	35	6
ft53.1	6194	4981	9.71	5780	7045	5781	9646	5781	10540	5781	12688	$\underline{5833}$	400
ft53.2	[6581, 6619]	5079	19.45	5951	4381	5960	8084	5961	10924	5962	17774	$\underline{5982}$	174
ft53.3	[8323, 8446]	5928	114.76	7168	5761	7169	8637	7169	11541	7169	13080	$\underline{7178}$	200
ft53.4	11822	9850	2.6	11443	4828	$\underline{11449}$	5964	11449	9858	11449	14692	11437	49
ft70.1	32608	31543	228.15	32258	19022	32258	36000	32258	36000	32258	36000	$\underline{32348}$	3069
ft70.2	[33037, 33448]	31820	395.64	32556	19258	32556	23596	32556	24649	32556	36000	$\underline{32561}$	673
ft70.3	[34761, 35234]	32842	712.56	33960	9728	$\underline{33961}$	25165	33961	36000	33961	36000	33856	1764
ft70.4	[44368, 44451]	40068	115.96	41080	9196	$\underline{42116}$	23481	42116	36000	42116	36000	42043	182
kro124p.1	[31812, 32825]	29337	3589.11	29647	36000	29647	36000	29647	36000	29647	36000	$\underline{30663}$	36000
kro124p.2	[32320, 34253]	29544	3036	29544	36000	29923	36000	29923	36000	29923	36000	$\underline{30259}$	9791.41
kro124p.3	[34961, 40906]	30424	17364.51	30424	36000	30425	36000	30425	36000	30425	36000	$\underline{31840}$	21149
kro124p.4	[56261, 62818]	43495	2310.91	43495	36000	47023	36000	47023	36000	47023	36000	$\underline{49019}$	4776
p43.1	22545	879	3.19	22414	1702	22414	2175	22414	2997	22414	4363	$\underline{\mathbf{22545}}$	308
p43.2	22837	985	5.17	$\underline{22651}$	1858	22651	2465	22651	3525	22651	4350	22645	409
p43.3	23119	1076	3.38	22802	1956	22802	2532	22802	3689	22802	5636	$\underline{22848}$	400
p43.4	66848	44854	1.56	53858	1622	53858	2648	53858	3844	$\underline{66678}$	4951	56071	73
prob.100	[838, 1343]	803	428.58	815	36000	816	36000	816	36000	816	36000	$\underline{822}$	3457.78
prob.42	202	183	5.33	190	1429	191	1632	192	2718	$\underline{193}$	3040	188	201
rbg048a	282	273	4.8	**282**	2901	–	–	–	–	–	–	$\underline{\mathbf{282}}$	61
rbg050c	378	376	7.14	377	3311	377	5763	377	12218	**378**	14430	$\underline{\mathbf{378}}$	38
rbg109a	848	803	1.87	803	36000	832	36000	832	36000	832	36000	$\underline{840}$	427
rbg150a	1414	1381	7.44	1381	36000	1381	36000	1381	36000	1381	36000	$\underline{1411}$	1519
rbg174a	1641	1606	8.19	1606	36000	1606	36000	1606	36000	1606	36000	$\underline{1631}$	2512
rbg253a	2372	2308	20.05	2308	36000	2308	36000	2308	36000	2308	36000	$\underline{2342}$	850
rbg341a	[2062, 2147]	1961	82.02	1961	36000	1961	36000	1961	36000	1961	36000	$\underline{2019}$	1604.05
rbg358a	[2037, 2172]	1967	7028	1967	36000	1967	36000	1967	36000	1967	36000	$\underline{2001}$	36000
rbg378a	[2233, 2385]	2132	35422	2132	36000	2132	36000	2132	36000	2132	36000	$\underline{2166}$	36000
ry48p.1	[13084, 13135]	11617	22.54	11952	3413	11966	5311	11984	6772	11988	10910	$\underline{12158}$	440
ry48p.2	[13401, 13802]	11721	12.24	12188	3529	12216	4375	12216	6812	12216	8171	$\underline{12357}$	379
ry48p.3	[15768, 16533]	12520	112.79	13873	1749	13879	4392	13879	4717	13879	5888	$\underline{13937}$	235
ry48p.4	25977	20378	3.46	21888	1844.32	21888	2081	22670	6564	$\underline{23049}$	8669	22861	33

Notes: column 'OPT' provides optimum values of the instances (if known) or the best bounds;
columns M_1–M_5 and M_1^* present LP-relaxation lower bounds and the corresponding running times;
optimum values highlighted in bold, best lower bounds underlined

Although the model M_1^* appears to be the best performer for the most cases, there exist instances, e.g. ft53.4, ft70.4, and ry48.p4, where some other competitors found more tight lower bounds. Therefore, we evaluate models M_3^* and M_5^* obtained by combination M_1^* with M_3 and M_1^* with M_5, where M_5 and M_3 are chosen for the combination as the most powerful and well-balanced[3] models among M_2–M_5 respectively.

According to results presented in Table 4, formulations M_3, M_5 and M_1^* collaborate quite well. In particular, for instances ft53.2, ft70.1 and p43.2, M_3^* provides better lower bounds than both initial models M_3 and M_1^*. The similar result can be observed for instances ry48p.3 and ft53.4 with respect to formulations M_5, M_1^* and their combination M_5^*.

While the combined models perform better than their initial counterparts, they still remain to be quite expensive to be applied in the branch-and-cut algorithm. On the other hand, comparing

[3] with respect to accuracy and time consumption

the model M_3^* with the sampled one M_{3s}^* and excluding tiny instances ESC07, ESC12, br17.10 and br17.12, we observe the significant decrease of the time complexity, i.e. LP-relaxation was solved 16 times faster in average. Furthermore, the better lower bounds were obtained in 18 out of 36 remaining instances. For those instances where M_{3s}^* found less accurate results, the lower bound decreased at most by 1.7%. In addition, we should emphasize one large instance rbg109a, where M_{3s}^* found an optimum value of the LP-relaxation faster than all other competitors.

Table 4: Performance of the combined and sampled formulations.

Instance	OPT	M_1^*		M_3		M_5		M_3^*		M_5^*		M_{3s}^*		M_{5s}^*	
		LPB	t	LPB	t	LPB	t	LPB	t	LPB	t	LPB	t	LPB	t
ESC07	1730	**1730**	0.09	1730	0.29	1730	0.3	1730	0.29	1730	0.3	**1730**	0.04	**1730**	0.06
ESC12	1390	**1390**	0.8	1387	7.1	1387	9.97	1387	7.2	1387	10	**1390**	0.35	**1390**	0.46
ESC25	1383	1363	7	1362	300	1362	448	1362	205	1362	221	1363	13	**1383**	4.31
ESC47	1063	1023	119	1012	2982	1016	7420	1014	5899	1018	6658	1026	247	1030	589
ESC63	62	**62**	318	**62**	8491	**62**	36000	**62**	13790	**62**	16506	**62**	250	**62**	243
ESC78	[14673, 14808]	14659	5477	14640	23287	14641	36000	14660	36000	14660	36000	14660	6106	14660	5312
br17.10	43	32	5	43	7	43	12.63	43	6.87	43	8.1	35	6	33	5
br17.12	43	35	6	43	7	43	35.05	43	6.69	43	7.99	34	5	34	4
ft53.1	6194	5833	400	5781	9646	5781	12688	5803	4786	5803	6144	5895	910	5833	375
ft53.2	[6581, 6619]	5982	174	5960	8084	5962	17774	6035	7282	6035	10144	5981	237	5982	124
ft53.3	[8323, 8446]	7178	200	7169	8637	7169	13080	7169	7717	7171	10286	7176	300	7180	204
ft53.4	11822	11437	49	11449	5964	11449	14692	11450	1450	11457	9649	11498	62	11463	43
ft70.1	32608	32348	3069	32258	36000	32258	36000	32380	13700	32380	36000	32385	4484	32348	2191
ft70.2	[33037, 33448]	32561	673	32556	23596	32556	36000	32563	24445	32564	28444	32598	862	32559	586
ft70.3	[34761, 35234]	33856	1764	33961	25165	33961	36000	33990	21890	33995	29579	33855	1275	33852	1432
ft70.4	[44368, 44451]	42043	182	42116	23481	42116	36000	42120	11022	42120	12040	42019	287	41900	198
kro124p.1	[31812, 32825]	30663	36000	29647	36000	29647	36000	29647	36000	29647	36000	30174	36000	30563	36000
kro124p.2	[32320, 34253]	30259	9791.41	29923	36000	29923	36000	30082	36000	30082	36000	30274	36000	30326	5527
kro124p.3	[34961, 40906]	31840	21149	30425	36000	30425	36000	30425	36000	30425	36000	31780	21225	31871	36000
kro124p.4	[56261, 62818]	49019	4776	47023	36000	47023	36000	47023	36000	47023	36000	48660	2579	48560	2513
p43.1	22545	**22545**	308	22414	2175	22414	4363	22415	1774	22415	3816	**22545**	206	**22545**	167
p43.2	22837	22645	409	22651	2465	22651	4350	22653	2316	22655	3649	22650	445	22642	338
p43.3	23119	22848	400	22802	2532	22802	5636	22870	1272	22872	2361	22915	389	22870	325
p43.4	66848	56071	73	53858	2648	66678	4951	53859	1328	66679	3400	56053	80	66700	340
prob.100	[838, 1343]	822	3457.78	816	36000	816	36000	816	36000	816	36000	823	4736	822	3596
prob.42	202	188	201	191	1632	193	3040	192	1325	193	2043	190	322	188	196
rbg048a	282	**282**	61	282	3544	282	15951	**282**	2128	**282**	21449	**282**	38	**282**	46
rbg050c	378	**378**	38	377	5763	**378**	14430	**378**	2880	**378**	12682	**378**	29	**378**	65
rbg109a	848	840	427	832	36000	832	36000	832	36000	832	36000	**848**	3530	840	952
rbg150a	1414	1411	1519	1381	36000	1381	36000	1381	36000	1381	36000	1411	17762	1411	2280
rbg174a	1641	1631	2512	1606	36000	1606	36000	1606	36000	1606	36000	1635	36000	1632	2455
rbg253a	2372	2342	850	2308	36000	2308	36000	2308	36000	2308	36000	2342	9059	2342	2135
rbg323a	2533	2515	3654.61	2491	36000	2491	36000	2491	36000	2491	36000	2517	5580	2515	7737
rbg341a	[2062, 2147]	2019	1604.05	1961	36000	1961	36000	1961	36000	1961	36000	2017	4190	2021	3533
rbg358a	[2037, 2172]	2001	36000	1967	36000	1967	36000	1967	36000	1967	36000	2013	36000	2013	36000
rbg378a	[2233, 2385]	2166	36000	2132	36000	2132	36000	2132	36000	2132	36000	2189	36000	2191	36000
ry48p.1	[13084, 13135]	12158	440	11966	5311	11988	10910	12052	3539	12053	8878	12458	926	12167	252
ry48p.2	[13401, 13802]	12357	379	12216	4375	12216	8171	12217	5176	12217	8459	12780	1436	12366	253
ry48p.3	[15768, 16533]	13937	235	13879	4392	13879	5888	14011	5084	14597	13794	13783	179	13840	172
ry48p.4	25977	22861	33	21888	2081	23049	8669	22781	1874	23050	4988	22674	34	22677	46

Notes: column 'OPT' provides optimum values of the instances (if known) or the best bounds; optimum values highlighted in bold

As for the models M_5^* and M_{5s}^*, we observe average speed-up by 59 times and better lower bounds in 22 out the same 36 instances. For that instances, where M_5^* outperform its sampled counterpart, the lower bound decreased at most by 5.2%. In addition, we should emphasize the instance ESC25, for which M_{5s}^* was the only competitor, who found the optimum value.

To summarize, we conclude that the addition of predecessor/successor inequalities and application of the proposed heuristic separation procedure can provide significant improvement in LP-relaxation of the PCGTSP.

7.3 Experiment II: Comparison of Branch-and-Cut Algorithms

This experiment is intended to assess variants of the branch-and-cut algorithm proposed in Section 6 induced by several formulations introduced in Section 5.

For the first competition, we choose variants bc_1^*, bc_{3s}^*, and bc_{5s}^* induced by the best performers of Experiment I, the models M_1^*, M_{3s}^*, and M_{5s}^* respectively. In addition, we consider the variant bc_{MTZ}^* induced by the formulation M_1^*, where the compact model is replaced with an adapted to the PCGTSP classic compact Miller-Tucker-Zemlin model (Miller et al. (1960)). It can be obtained from the considered compact model by replacing constraints (12)–(15) with

$$(m-1)u_{pq} + v_p \leq v_q + m - 2 \quad (C_p, C_q \in \mathcal{C} \setminus \{C_1\},\ p \neq q),$$

for $0 \leq v_p, v_q \leq (m-2)$, and exclusion of y variables. Such a model was chosen intentionally, as one of the lightest known compact models ensuring efficient enumeration of the nodes of branching tree. As baselines, we use:

- Gurobi solver applied to the model M_1 with default configuration (including built-in heuristics and cutting planes),
- our PCGTSP adaptation bc^*_{DFJ} of the state-of-the-art branch-and-cut algorithm for the SOP proposed in Gouveia and Ruthmair (2015). This algorithm tackles the similar partial classic Dantzig-Fulkerson-Johnson (DFJ) model Dantzig et al. (1954) ((6)–(11), and (16) without y variables) and separates corresponding families of valid inequalities (17)–(26). In addition, we replace the initial primal heuristic with our GLNS-based heuristic PCGLNS, since GLNS appears to be more efficient for the GTSP-like problems (see Smith and Imeson (2017)).

Table 5(a): Comparison of the branch-and-cut algorithms

Instance	OPT		Gurobi	bc^*_{DFJ}	bc^*_1	bc^*_{3s}	bc^*_{5s}	bc^*_{MTZ}	bc_{3s}	bc_{MTZ}
ESC07	1730	UB	1730	1730	1730	**1730**	1730	1730	1730	1730
		LB	1730	1730	1730	**1730**	1730	1730	1730	1730
		gap	0	0	0	**0**	0	0	0	0
		t	0.05	0.06	0.09	**0.04**	0.06	0.05	0.06	0.05
ESC12	1390	UB	1390	1390	1390	1390	1390	**1390**	1390	1390
		LB	1390	1390	1390	1390	1390	**1390**	1390	1390
		gap	0	0	0	0	0	**0**	0	0
		t	5.05	0.58	0.8	0.35	0.46	**0.19**	2.14	0.37
ESC25	1383	UB	1383	**1383**	1383	1383	1383	1383	1383	1383
		LB	1383	**1383**	1383	1383	1383	1383	1383	1383
		gap	0	**0**	0	0	0	0	0	0
		t	7.56	**1.98**	8	15	4.31	3.75	8.55	4
ESC47	1063	UB	1063	1063	1063	1063	1063	**1063**	1063	1063
		LB	1063	1063	1063	1063	1063	**1063**	1063	1063
		gap	0	0	0	0	0	**0**	0	0
		t	6963	43.06	623	1520	837	**29**	1355	85
ESC63	62	UB	62	62	62	62	62	**62**	62	62
		LB	62	62	62	62	62	**62**	62	62
		gap	0	0	0	0	0	**0**	0	0
		t	209	7.52	318	250	243	**4.2**	244	4.4
ESC78	[14673, 14808]	UB	14808	14808	14808	14808	14808	14808	14808	14808
		LB	14633	14657	14661	14666	14667	**14673**	14553	14633
		gap	1.2	1	1	1	1	**0.9**	1.8	1.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
br17.10	43	UB	43	43	43	43	43	**43**	43	43
		LB	43	43	43	43	43	**43**	43	43
		gap	0	0	0	0	0	**0**	0	0
		t	232	11.74	12	10	13	**9.75**	10.76	4183
br17.12	43	UB	43	43	**43**	43	43	43	43	43
		LB	43	43	**43**	43	43	43	43	43
		gap	0	0	**0**	0	0	0	0	0
		t	75	58.09	**11**	12	12	12.1	12	362
ft53.1	6194	UB	6194	6194	6194	6194	6194	**6194**	6194	6194
		LB	5933	6194	6169	6176	6177	**6194**	6176	6022
		gap	4.4	0	0.4	0.3	0.3	**0**	0.3	2.9
		t	72000	3398	72000	72000	72000	**2054**	72000	72000
ft53.2	[6581, 6619]	UB	6619	6619	6619	6619	6619	6619	6619	6619
		LB	6087	6442	6410	6490	6439	**6581**	6487	6248
		gap	8.7	2.7	3.3	2	2.8	**0.6**	2	5.9
		t	72000	72000	72000	72000	72000	72000	72000	72000

Notes: best performers are highlighted in bold

All the competitors are supplied with the same primal heuristic PCGLNS. The time limit is set to 20 hours (72000 seconds). We report cost of the best found solution (UB), the best lower bound (LB), an accuracy measure (gap, in percentage)

$$\text{gap} = \frac{\text{UB}-\text{LB}}{\text{LB}} \geqslant \frac{\text{UB}-\text{OPT}}{\text{OPT}} = \varepsilon,$$

for the relative error ε of the obtained solution, and the elapsed time (in seconds).

As it follows from Tables 5(a)–5(d), both baseline algorithms solved to optimality 17 out of 40 instances in total, where the instances rbg150a and rbg174a (which are huge ones) were solved by Gurobi solely, and the instances ft53.1, ft70.1, p43.1, p43.4, prob.42 – by bc^*_{DFJ}. In turn, the proposed algorithms bc^*_1, bc^*_{3s}, bc^*_{5s}, bc^*_{MTZ} managed to solve to optimality 23 out of 40 instances in total including all the mentioned above. Regarding to the new six instances, ft53.4 and ry48p.4 were solved by all of them, the instance rbg253a was solved by bc^*_1, the instance rbg323a – by bc^*_{3s}, bc^*_{5s} and bc^*_{MTZ}. Finally, the optimal solutions of the instances p43.2 and p43.3 were found by both bc^*_{3s} and bc^*_{5s}. In addition, each of 15 instances solved to optimality by bc^*_{DFJ} is also solved exactly by one of the proposed variants about 8 times faster in average. Nevertheless, we should mention instances ESC25, rbg048a, and rbg050c, where bc^*_{DFJ} outperforms other competitors in terms of the elapsed time.

In the residual 17 open instances, the proposed algorithms managed to significantly increase lower bounds and close the average gap value about 4.5 times better than both baselines and complement each other quite well.

Our second observation is related to the comparison of variants bc^*_{3s} and bc^*_{MTZ} with the corresponding counterparts bc_{3s} and bc_{MTZ} obtained by exclusion the predecessor / successor inequalities from the separation pipeline. Regarding to bc_{3s} and bc^*_{3s}, we observe that inclusion of such inequalities allows to solve to optimality three additional instances (p43.2, p43.3, and rbg323a). Furthermore, for 12 out of 16 instances solved by both competitors exactly, we observe notable decrease of the running rime. In addition, for the remaining 21 instances, bc^*_{3s} closed the gap by 1.7 times better in average. In turn, we should note that bc^*_{MTZ} significantly outperforms bc_{MTZ} in terms of instances solved to optimality, gap values and elapsed time.

Therefore, the predecessor/successor inequalities are proved to be useful for the PCGTSP in the branch-and-cut setting as well.

Table 5(b): Comparison of the branch-and-cut algorithms

Instance	OPT		Gurobi	bc^*_{DFJ}	bc^*_1	bc^*_{3s}	bc^*_{5s}	bc^*_{MTZ}	bc_{3s}	bc_{MTZ}
ft53.3	[8323, 8446]	UB	8446	8446	8446	8446	8446	8446	8446	8446
		LB	7135	7786	7992	8186	8222	**8323**	8108	7393
		gap	18.4	8.5	5.7	3.2	2.7	**1.5**	4.2	14.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
ft53.4	11822	UB	11822	11822	**11822**	11822	11822	11822	11822	11822
		LB	11253	11687	**11822**	11822	11822	11822	11822	11707
		gap	5.1	1.2	**0**	0	0	0	0	1
		t	72000	72000	**129**	163	192	162	733	72000
ft70.1	32608	UB	32614	32608	32614	32614	32614	**32608**	32614	32614
		LB	31765	32608	32466	32480	32455	**32608**	32475	31968
		gap	2.7	0	0.5	0.4	0.5	**0**	0.4	2
		t	72000	6523	72000	72000	72000	**5573**	72000	72000
ft70.2	[33037, 33448]	UB	33448	33448	33448	33448	33448	33448	33448	33448
		LB	32029	32889	32799	32890	32805	**33037**	32740	32725
		gap	4.4	1.7	2	1.7	2	**1.2**	2.2	2.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
ft70.3	[34761, 35234]	UB	35234	35234	35234	35234	35234	35234	35234	35234
		LB	33232	34105	34304	**34761**	34719	34736	34629	34167
		gap	6	3.3	2.7	1.4	1.5	1.4	1.7	3.1
		t	72000	72000	72000	72000	72000	72000	72000	72000
ft70.4	[44368, 44451]	UB	44451	44451	44451	44451	44451	44451	44451	44451
		LB	41634	41388	44051	43998	44033	**44368**	43990	41459
		gap	6.8	7.4	0.9	1	0.9	**0.2**	1	7.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
kro124p.1	[31812, 32825]	UB	32835	32835	32825	32835	32835	32835	32835	32835
		LB	29704	30858	30827	30174	30182	**31812**	29530	30454
		gap	10.5	6.4	6.5	8.8	8.8	**3.2**	11.2	7.8
		t	72000	72000	72000	72000	72000	72000	72000	72000
kro124p.2	[32320, 34253]	UB	34253	34253	34253	34253	34253	34253	34253	34253
		LB	30084	30722	30509	30448	30448	**32320**	29881	31657
		gap	13.9	11.5	12.3	12.5	12.5	**6**	14.6	8.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
kro124p.3	[34961, 40906]	UB	40906	40906	40906	40906	40906	40906	40906	40906
		LB	30945	31930	32734	32954	32674	**34961**	31122	33738
		gap	32.2	28.1	25	24.1	25.2	**17**	31.4	21.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
kro124p.4	[56261, 62818]	UB	62818	62818	62818	62818	62818	62818	62818	62818
		LB	46861	45720	54993	55329	53841	**56261**	51210	51495
		gap	34.1	37.4	14.2	13.5	16.7	**11.7**	22.7	22
		t	72000	72000	72000	72000	72000	72000	72000	72000

Notes: best performers are highlighted in bold

8 Conclusion

In this paper, we addressed the Precedence Constrained Generalized Traveling Salesman Problem (PCGTSP) both in terms of the polyhedral study and algorithmic analysis. By evolving the results previously introduced for PCATSP, we proposed several novel families of the valid

Table 5(c): Comparison of the branch-and-cut algorithms

Instance	OPT		Gurobi	bc^*_{DFJ}	bc^*_1	bc^*_{3s}	bc^*_{5s}	bc^*_{MTZ}	bc_{3s}	bc_{MTZ}
p43.1	22545	UB	22545	22545	22545	22545	22545	**22545**	22545	22545
		LB	22408	22545	22545	22545	22545	**22545**	22545	22545
		gap	0.6	0	0	0	0	**0**	0	0
		t	72000	195	308	206	167	**27**	2583	35722
p43.2	22837	UB	22837	22837	22837	**22837**	22837	22837	22837	22837
		LB	22461	22711	22731	**22837**	22837	22765	22801	22639
		gap	1.7	0.6	0.5	**0**	0	0.3	0.2	0.9
		t	72000	72000	72000	**22780**	39365	72000	72000	72000
p43.3	23119	UB	23119	23119	23119	**23119**	23119	23119	23119	23119
		LB	22399	22293	22970	**23119**	23119	23085	23104	22821
		gap	3.2	3.7	0.6	**0**	0	0.1	0.1	1.3
		t	72000	72000	72000	**8672**	11665	72000	72000	72000
p43.4	66848	UB	66848	66848	66848	66848	66848	**66848**	66848	66848
		LB	45266	66848	66848	66848	66848	**66848**	66848	45169
		gap	47.7	0	0	0	0	**0**	0	48
		t	72000	2596	131	283	587	**86**	212	72000
prob.100	[838, 1343]	UB	1343	1516	1343	1343	1343	1343	1343	1343
		LB	813	824	826	824	826	**838**	813	790
		gap	65.2	84	62.6	63	62.6	**60.3**	65.2	70
		t	72000	72000	72000	72000	72000	72000	72000	72000
prob.42	202	UB	204	202	202	202	202	**202**	202	202
		LB	198	202	202	202	202	**202**	202	202
		gap	3	0	0	0	0	**0**	0	0
		t	72000	832	767	869	3559	**326**	1155	3987
rbg048a	282	UB	282	**282**	282	282	282	282	282	282
		LB	282	**282**	282	282	282	282	282	282
		gap	0	**0**	0	0	0	0	0	0
		t	57	**13**	61	38	46	22	61	46
rbg050c	378	UB	378	**378**	378	378	378	378	378	378
		LB	378	**378**	378	378	378	378	378	378
		gap	0	**0**	0	0	0	0	0	0
		t	42	**21**	38	29	65	47	34	673
rbg109a	848	UB	848	848	**848**	848	848	848	848	848
		LB	848	848	**848**	848	848	848	848	834
		gap	0	0	**0**	0	0	0	0	1.7
		t	1942	51757	**781**	3567	3530	842	6583	72000
rbg150a	1414	UB	1414	1414	**1414**	1414	1414	1414	1414	1414
		LB	1414	1400	**1414**	1414	1414	1414	1414	1400
		gap	0	1	**0**	0	0	0	0	1
		t	21725	72000	**7674**	27154	70000	22134	42549	72000

Notes: best performers are highlighted in bold

Table 5(d): Comparison of the branch-and-cut algorithms

Instance	OPT		Gurobi	bc^*_{DFJ}	bc^*_1	bc^*_{3s}	bc^*_{5s}	bc^*_{MTZ}	bc_{3s}	bc_{MTZ}
rbg174a	1641	UB	1641	1641	**1641**	1641	1641	1641	1641	1641
		LB	1641	1602	**1641**	1636	1637	1638	1630	1607
		gap	0	2.4	**0**	0.3	0.2	0.2	0.7	2.1
		t	62657	72000	**14448**	72000	72000	72000	72000	72000
rbg253a	2372	UB	2373	2372	**2372**	2372	2372	2373	2372	2373
		LB	2369	2358	**2372**	2357	2357	2365	2350	2301
		gap	0.2	0.6	**0**	0.6	0.6	0.3	0.9	3.1
		t	72000	72000	**27642**	72000	72000	72000	72000	72000
rbg323a	2533	UB	2595	2597	2586	**2533**	2533	2533	2594	2586
		LB	2528	2517	2531	**2533**	2533	2533	2530	2488
		gap	2.7	3.2	2.2	**0**	0	0	2.5	3.9
		t	72000	72000	72000	**71550**	71800	71900	72000	72000
rbg341a	[2062, 2147]	UB	2180	2195	2199	2184	2147	2184	2184	2184
		LB	2047	2017	2056	2061	**2062**	2059	2060	1928
		gap	6.5	8.8	7	6	**4.1**	6.1	6	13.3
		t	72000	72000	72000	72000	72000	72000	72000	72000
rbg358a	[2037, 2172]	UB	2172	2174	2175	2172	2174	2185	2172	2176
		LB	1996	2009	2025	2025	2013	**2037**	2002	1956
		gap	8.8	8.2	7.4	7.3	8	**7.3**	8.5	11.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
rbg378a	[2233, 2385]	UB	2390	2385	2404	2400	2402	2404	2400	2404
		LB	2185	2191	2210	2214	2205	**2233**	2132	2086
		gap	9.4	8.9	8.8	8.4	8.9	**7.7**	12.6	15.2
		t	72000	72000	72000	72000	72000	72000	72000	72000
ry48p.1	[13084, 13135]	UB	13135	13135	13135	13135	13135	13135	13135	13135
		LB	12065	12732	12634	**13084**	12914	13039	12669	12596
		gap	8.9	3.2	4	**0.4**	1.7	0.7	3.7	4.3
		t	72000	72000	72000	72000	72000	72000	72000	72000
ry48p.2	[13401, 13802]	UB	13802	13802	13802	13802	13802	13802	13802	13802
		LB	12217	12963	12917	**13401**	13327	13223	13019	12729
		gap	13	6.5	6.9	**3**	3.6	4.4	6	8.4
		t	72000	72000	72000	72000	72000	72000	72000	72000
ry48p.3	[15768, 16533]	UB	16533	16533	16533	16553	16533	16533	16533	16533
		LB	13387	14753	14825	15147	15441	**15768**	14672	14268
		gap	23.5	12.1	11.5	9.3	7.1	**4.9**	12.7	15.9
		t	72000	72000	72000	72000	72000	72000	72000	72000
ry48p.4	25977	UB	25977	25977	25977	25977	25977	**25977**	25977	25977
		LB	22732	24079	25977	25977	25977	**25977**	25977	23644
		gap	14.3	7.9	0	0	0	**0**	0	9.9
		t	72000	72000	11182	22106	29865	**4626**	25000	72000

Notes: best performers are highlighted in bold inequalities. Then, we established dimension of the PCGTS polytope and proved sufficient conditions for the predecessor/successor inequalities to be facet-inducing.

Further, we offered a sequence of novel formulations for the PCGTSP and proposed the first branch-and-cut algorithm relying on these fomulations. In the numerical evaluation, we reported the most efficient formulations in terms of LP-relaxation bounds and suggested several well-collaborating variants of the proposed branch-and-cut. As a result, the number of PCGTSPLIB instances solved to optimality became 23 out of 40, where for 11 instances it was done for the first time.

In addition, the obtained results confirmed the importance of the predecessor/successor inequalities for the PCGTSP, both for LP-relaxation and branch-and-cut framework.

References

N. Ascheuer, M. Júnger, and G. Reinelt. A branch & cut algorithm for the asymmetric traveling salesman problem with precedence constraints. *Computational Optimization and Applications*, 17(1):61–84, 2000. doi: 10.1023/A:1008779125567.

Egon Balas and Neils Simonetti. Linear time dynamic-programming algorithms for new classes of restricted TSPs: A computational study. *INFORMS J. on Computing*, 13(1):56–75, February 2001. ISSN 1526-5528. doi: 10.1287/ijoc.13.1.56.9748.

Egon Balas, Matteo Fischetti, and William Pulleyblank. The precedence-constraint asymmetric traveling salesman polytope. *Math. Program.*, 68:241–265, 03 1995. doi: 10.1007/BF01585767.

Jørgen Bang-Jensen and Gregory Z. Gutin. *Digraphs: Theory, Algorithms and Applications.* Springer London, 2009. ISBN 978-1-84800-998-1. doi: 10.1007/978-1-84800-998-1_1.

Norman Biggs. *Algebraic Graph Theory.* Cambridge Tracts in Mathematics 67. Cambridge University Press, 2 edition, 1974. ISBN 052120335X,9780521203357.

A. G. Chentsov, M. Yu. Khachai, and D. M. Khachai. An exact algorithm with linear complexity for a problem of visiting megalopolises. *Proceedings of the Steklov Institute of Mathematics*, 295(1):38–46, 2016. ISSN 1531-8605. doi: 10.1134/S0081543816090054.

Alexander G. Chentsov, Pavel A. Chentsov, Alexander A. Petunin, and Alexander N. Sesekin. Model of megalopolises in the tool path optimisation for CNC plate cutting machines. *International Journal of Production Research*, 56(14):4819–4830, 2018. doi: 10.1080/00207543.2017.1421784.

Andre A. Cire and Willem-Jan van Hoeve. Multivalued decision diagrams for sequencing problems. *Operations Research*, 61(6):1411–1428, 2013. doi: 10.1287/opre.2013.1221.

G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. *Journal of the Operations Research Society of America*, 2(4):393–410, 1954. doi: 10.1287/opre.2.4.393.

Reginald Dewil, Pieter Vansteenwegen, and Dirk Cattrysse. A review of cutting path algorithms for laser cutters. *The International Journal of Advanced Manufacturing Technology*, 87(5):1865–1884, 2016. doi: 10.1007/s00170-016-8609-1.

L.F. Escudero. An inexact algorithm for the sequential ordering problem. *European Journal of Operational Research*, 37(2):236–249, 1988. doi: 10.1016/0377-2217(88)90333-5.

Corinne Feremans, Alexander Grigoriev, and René Sitters. The geometric generalized minimum spanning tree problem with grid clustering. *4OR*, 4(4):319–329, 2006. ISSN 1614-2411. doi: 10.1007/s10288-006-0012-6.

Matteo Fischetti, Juan Josè Salazar González, and Paolo Toth. The symmetric generalized traveling salesman polytope. *Networks*, 26:113–123, 09 1995. doi: 10.1002/net.3230260206.

Matteo Fischetti, Juan José Salazar González, and Paolo Toth. A branch-and-cut algorithm for the symmetric Generalized Traveling Salesman Problem. *Operations Research*, 45(3):378–394, 1997. doi: 10.1287/opre.45.3.378.

Michel Gendreau and Jean-Yves Potvin. *Handbook of Metaheuristics*. International Series in Operations Research & Management Science 272. Springer International Publishing, 3rd edition, 2019. ISBN 978-3-319-91085-7; 978-3-319-91086-4.

L. Gouveia and P. Pesneau. On extended formulations for the precedence constrained asymmetric traveling salesman problem. *Networks*, 48(2):77–89, 2006. doi: 10.1002/net.20122.

L. Gouveia and J. M. Pires. The asymmetric travelling salesman problem and a reformulation of the miller–tucker–zemlin constraints. *European Journal of Operational Research*, 112(1): 134–146, 1999. doi: 10.1016/S0377-2217(97)00358-5.

L. Gouveia and J. M. Pires. The asymmetric travelling salesman problem: on generalizations of disaggregated miller–tucker–zemlin constraints. *Discrete Applied Mathematics*, 112(1):129–145, 2001. doi: 10.1016/S0166-218X(00)00313-9.

Luis Gouveia and Mario Ruthmair. Load-dependent and precedence-based models for pickup and delivery problems. *Computers & Operations Research*, 63:56–71, 2015. ISSN 0305-0548. doi: 10.1016/j.cor.2015.04.008.

Luis Gouveia, Pierre Pesneau, Mario Ruthmair, and Daniel Santos. Combining and projecting flow models for the (precedence constrained) asymmetric traveling salesman problem. *Networks*, 71(4):451–465, 2018. doi: 10.1002/net.21765.

Gregory Gutin and Daniel Karapetyan. A memetic algorithm for the generalized traveling salesman problem. *Natural Computing*, 9(1):47–60, 2010. doi: 10.1007/s11047-009-9111-6.

Michael Held and Richard M. Karp. A dynamic programming approach to sequencing problems. *Journal of the Society for Industrial and Applied Mathematics*, 10(1):196–210, 1962. ISSN 03684245. URL `http://www.jstor.org/stable/2098806`.

Keld Helsgaun. Solving the equality Generalized Traveling Salesman Problem using the Lin–Kernighan–Helsgaun algorithm. *Mathematical Programming Computation*, pages 1–19, 2015.

D. Karapetyan and G. Gutin. Efficient local search algorithms for known and new neighborhoods for the generalized traveling salesman problem. *European Journal of Operational Research*, 219(2):234–251, 2012. ISSN 0377-2217. doi: 10.1016/j.ejor.2012.01.011.

Daniil Khachai. Branch-and-cut algorithm for the precedence constrained generalized traveling salesman problem: Gurobi callbacks implementation, 2022. URL `https://github.com/EnsignDaniels/BnC`.

M. Yu. Khachai and E. D. Neznakhina. Approximation schemes for the generalized traveling salesman problem. *Proceedings of the Steklov Institute of Mathematics*, 299(1):97–105, 2017. ISSN 1531-8605. doi: 10.1134/S0081543817090127.

Michael Khachay and Katherine Neznakhina. Complexity and approximability of the Euclidean Generalized Traveling Salesman Problem in grid clusters. *Annals of Mathematics and Artificial Intelligence*, 88(1):53–69, 2020. doi: 10.1007/s10472-019-09626-w.

Michael Khachay, Andrei Kudriavtsev, and Alexander Petunin. PCGLNS: A heuristic solver for the Precedence Constrained Generalized Traveling Salesman Problem. *LNCS*, 12422:196–208, 2020. doi: 10.1007/978-3-030-62867-3_15.

Michael Khachay, Stanislav Ukolov, and Alexander Petunin. Problem-specific Branch-and-Bound algorithms for the Precedence Constrained Generalized Traveling Salesman Problem. *LNCS*, pages 136–148, 2021. doi: 10.1007/978-3-030-91059-4_10.

T.A. Makarovskikh, A.V. Panyukov, and E.A. Savitskiy. Mathematical models and routing algorithms for economical cutting tool paths. *International Journal of Production Research*, 56(3):1171–1188, 2018. doi: 10.1080/00207543.2017.1401746.

C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *J. ACM*, 7(4):326–329, 1960. doi: 10.1145/321043.321046.

Thomas L. Morin and Roy E. Marsten. Branch-and-bound strategies for dynamic programming. *Operations Research*, 24(4):611–627, 1976.

Charles E. Noon and James C. Bean. An efficient transformation of the generalized traveling salesman problem. *INFOR: Information Systems and Operational Research*, 31(1):39–44, 1993. doi: 10.1080/03155986.1993.11732212.

Christos Papadimitriou. Euclidean TSP is NP-complete. *Theoret. Comput. Sci.*, 4:237–244, 1977.

Roberto Roberti and Paolo Toth. Models and algorithms for the asymmetric traveling salesman problem: an experimental comparison. *EURO Journal on Transportation and Logistics*, 1(1): 113–133, 2012. ISSN 2192-4376. doi: 10.1007/s13676-012-0010-0.

Raad Salman, Johan S. Carlson, Fredrik Ekstedt, Domenico Spensieri, Johan Torstensson, and Rikard Söderberg. An industrially validated CMM inspection process with sequence constraints. *Procedia CIRP*, 44:138 – 143, 2016. doi: 10.1016/j.procir.2016.02.136.

Raad Salman, Fredrik Ekstedt, and Peter Damaschke. Branch-and-bound for the Precedence Constrained Generalized Traveling Salesman Problem. *Operations Research Letters*, 48(2): 163–166, 2020. doi: 10.1016/j.orl.2020.01.009.

Subhash C. Sarin, Hanif D. Sherali, and Ajay Bhootra. New tighter polynomial length formulations for the Asymmetric Traveling Salesman Problem with and without precedence constraints. *Oper. Res. Lett.*, 33(1):62–70, 2005. ISSN 0167-6377. doi: 10.1016/j.orl.2004.03.007.

Hanif D. Sherali and Patrick J. Driscoll. On tightening the relaxations of miller-tucker-zemlin formulations for asymmetric traveling salesman problems. *Operations Research*, 50(4):656–669, 2002. doi: 10.1287/opre.50.4.656.2865.

Stephen L. Smith and Frank Imeson. GLNS: An effective large neighborhood search heuristic for the Generalized Traveling Salesman Problem. *Computers & Operations Research*, 87:1–19, 2017. doi: 10.1016/j.cor.2017.05.010.

S. Srivastava, S Kumar, R Garg, and P. Sen. Generalized Traveling Salesman Problem through n sets of nodes. *CORS journal*, 7(2):97–101, 1969.

George Steiner. On the complexity of dynamic programming for sequencing problems with precedence constraints. *Annals of Oper. Res.*, 26(1):103–123, 1990.

Yuan Yuan, Diego Cattaruzza, Maxime Ogier, and Frédéric Semet. A branch-and-cut algorithm for the generalized traveling salesman problem with time windows. *European Journal of Operational Research*, 286(3):849–866, 2020. ISSN 0377-2217. doi: 10.1016/j.ejor.2020.04.024.