

Article

On Solving the Minimum Spanning Tree Problem with Conflicting Edge Pairs

Roberto Montemanni ^{1,*}  and Derek H. Smith ² 

¹ Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, 42122 Reggio Emilia, Italy

² Faculty of Computing, Engineering and Science, University of South Wales, Pontypridd CF37 1DL, UK; derek.smith@southwales.ac.uk

* Correspondence: roberto.montemanni@unimore.it; Tel.: +39-0522-522-126

Abstract

The Minimum Spanning Tree with Conflicting Edge Pairs is a generalization that adds conflict constraints to a classical optimization problem on graphs used to model several real-world applications. In recent years, several heuristic and exact approaches have been proposed to tackle this problem. In this paper, we present a mixed-integer linear program not previously applied to this problem, and we solve it with an open-source solver. Computational results for the benchmark instances commonly adopted in the literature of the problem are reported. The results indicate that the approach we propose obtains results aligned with those of the much more sophisticated approaches available, notwithstanding it being much simpler to implement. During the experimental campaign, six instances were closed for the first time, with nine improved best-known lower bounds and sixteen improved best-known upper bounds over a total of two hundred thirty instances considered.

Keywords: minimum spanning tree; conflict constraints; exact solutions; heuristic solutions



Academic Editor: Frank Werner

Received: 14 July 2025

Revised: 7 August 2025

Accepted: 14 August 2025

Published: 18 August 2025

Citation: Montemanni, R.; Smith, D.H. On Solving the Minimum Spanning Tree Problem with Conflicting Edge Pairs. *Algorithms* **2025**, *18*, 526. <https://doi.org/10.3390/a18080526>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Minimum Spanning Tree Problem with Conflicting Edge Pairs (MSTC) is a variant of the classical minimum spanning tree (MST) problem recently introduced [1]. Given a connected undirected graph with costs on the edges and a set of edge pairs in conflict with each other, the MSTC consists of finding the spanning tree with the minimum total cost that uses at most one of the edges of each conflicting pair. The MSTC was first introduced by Darmann et al. [2]. It was shown that under general settings, the problem cannot be solved in polynomial time, although some special cases are shown to be polynomially solvable. Several heuristic methods were subsequently proposed in [3], while a branch-and-cut based on a mathematical programming model and some valid inequalities was presented in [4]. More recently, a metaheuristic method was designed and presented in [5], while some new methods based on Lagrangian relaxations, able to produce high-quality lower bounds and heuristic solutions, were introduced in [6]. Finally, a new branch-and-cut approach based on some new valid inequalities was discussed in [7] and shown to represent the current state-of-the-art exact solving methods.

In the literature of graph theory and algorithms, several classic optimization problems have been studied in one or more variants involving conflict constraints, witnessing increasing popularity of the topic due to the number of related real-world applications. For

example, for the Assignment Problem with Conflicts, theoretical results on the computational complexity, together with some approximation results, can be found in [2]. Further results for special polynomially solvable settings were presented in [8], together with some heuristics and lower-bounding schema. More contributions can be found in [9,10], where some Mixed-Integer Linear Programming models, exact Branch-and-Bound methods, and heuristic ideas are introduced. The concepts of the last two papers were later summarized and extended in [11]. For the Knapsack Problem with Conflicts, a local-search heuristic and a branch-and-bound method based on Lagrangian relaxation are discussed in [12]. In [13], the authors proposed some preprocessing techniques and a different branch-and-bound algorithm. Heuristic methods for this problem were discussed in [14,15]. Metaheuristic approaches were finally presented in [16,17]. Finally, a branch-and-bound exact method, based on binary branching, was proposed in [18], while another branch-and-bound schema, based on n -ary branching, was developed and tested in [19]. Different Set-Covering Problems with Conflicts were proposed in [20,21], together with approximation algorithms and computational complexity results. Different families of valid inequalities for a model representing the problem were presented in [22], together with some preprocessing and speed-up routines. A specialization of the same problem in geometric settings was discussed in [23], where some approximation results were presented. The Shortest Path Problem with Conflicts can be found in [24], where critical variants of the shortest path problem arising in the automatic generation of test paths for programs were introduced, and in [25], where the first definition of conflict pairs was presented, although nodes instead of arcs were a part of the conflicts. A procedure to solve the problem was discussed in [26]. A polyhedral study of the problem was presented in [27], while branch-and-bound and dynamic programming approaches were recently proposed in [28,29]. The Maximum Flow Problem with Conflicts was first studied in [30], where the NP-hardness of the problem is shown. Later, some mixed-integer linear programming formulations and exact methods, such as a Benders Decomposition, a Branch-and-Bound, and a Russian Doll Search were presented in [31]. More recently, the authors of [32] proposed a heuristic approach for the problem called Kernousel, obtained by merging two metaheuristic approaches, one based on Carousel Greedy [33] and the other on Kernel Search [34].

The MSTC can be used to represent real applications arising in different domains. For example, the design of an offshore wind farm network, where the wind turbines have to be connected together, and the different possible point-to-point connections are characterized by different costs. Technical reasons prevent the use of overlapped cables in such a context [35], and this generates conflict constraints. Other practical applications are in the design and installation of long-distance pipelines, where conflicts typically represent the impossibility of using infrastructures owned by competitor service providers or, alternatively, the impossibility of crossing countries with reciprocal political issues with the same line [36].

In this paper, a compact mixed-integer linear programming model for the MSTC is introduced in the context of the MSTC. It is based on multi-commodity flow concepts and extends known formulations for the MST problem ([37,38]) to the case with conflicts. Notice that multi-commodity flow models are common for problems related to the MST problem (see, for example, [39] for the Min-degree Constrained MST and [40] for the Diameter-Constrained MST). The model we propose is solved by the open-source solver CP-SAT, which is a part of the Google OR-Tools [41] optimization suite. Normally, models based on multi-commodity flow are not very effective in the context of integer programming, due to the poor linear relaxation, but we believe such a formulation might perform well when solved with CP-SAT. Successful application of this solver on optimization problems with characteristics similar to the problem under investigation, such as the Parallel Drone-Scheduling Traveling Salesman Problem and Vehicle Routing Problems [42,43],

the Maximum Flow Problem with Conflict Constraints [44], and the Home Healthcare-Routing and Scheduling Problem [45], motivated this study. A comprehensive experimental campaign is conducted and reported on the benchmark instances previously used in the literature.

The overall organization of the paper can be summarized as follows: Section 2 is dedicated to the formal introduction and definition of the Minimum Spanning Tree Problem with Conflicting Edge Pairs. Section 3 presents a compact mixed-integer linear programming model to represent the problem. In Section 4, the new model is validated through an extensive experimental perspective. After the description of the benchmark instances considered and of the methods used for comparison purposes, the detailed results obtained by solving the new model are reported. Some conclusions about the work presented and the results achieved are drawn in Section 5 to close the paper.

2. Problem Description

The Minimum Spanning Tree Problem with Additional Conflicts can be defined on an undirected graph $(G = (V, E))$, with V denoting the set of nodes and E the set of edges. Each edge $(\{i, j\} \in E)$ is associated with a cost $(u_{ij} \in \mathbb{N}^+)$. The following feature differentiates the problem from the classic Minimum Spanning Tree Problem and increases the theoretical computational complexity of the problem to \mathcal{NP} -hard [2]: Two edges $(\{i, j\}, \{k, l\} \in E)$ are said to be *in conflict* if, at most, one of them can be included in a feasible solution. The set of edges in conflict with the edge $(\{i, j\} \in E)$ is contained in the set δ_{ij} . The goal of the MSTC is to find the minimum-cost-spanning tree that satisfies all the conflict constraints.

An example of an MFPC instance and the optimal solution for the same instance are provided in Figure 1.

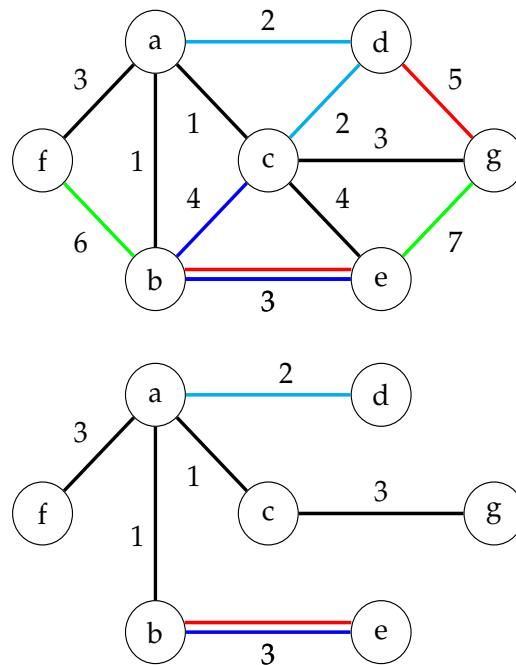


Figure 1. (Top) An example of a small MSTC instance is shown, where the edge costs are placed by the edges. Edges in black are not affected by conflicts, while any two edges sharing the same red, cyan, blue, or green color are in conflict. According to the notation $\delta_{ab} = \delta_{ac} = \delta_{af} = \delta_{ce} = \delta_{cg} = \emptyset$, $\delta_{ad} = \{\{c, d\}\}$, $\delta_{bc} = \{\{b, e\}\}$, $\delta_{be} = \{\{b, c\}, \{d, g\}\}$, $\delta_{bf} = \{\{e, g\}\}$, $\delta_{cd} = \{\{a, d\}\}$, $\delta_{dg} = \{\{b, e\}\}$, and $\delta_{eg} = \{\{b, f\}\}$. (Bottom) the optimal solution that does not violate any conflict is reported. Note that no edge of the green conflict is included.

3. A Mixed-Integer Linear Programming Model

In this section, a model for the MSTC—obtained by extending a classical model for the Minimum Spanning Tree Problem [37]—is proposed. The compact model is based on multi-commodity flow, characterized by the number of variables in the order of $O(|V|^2)$ and the number of constraints in the order of $O(|V|^2)$, making it compact and suitable to be attacked directly by a solver. Conversely, the approaches previously considered in the literature rely on models with an exponential number of constraints [4,7], which require a more complex implementation of dynamic and iterative mechanisms to generate violated constraints during the solving process. Our approach is expected to have a weaker linear relaxation than the exponential models, but we believe it has potential when approached with solvers able to do logical inferences, as is done herein. The new model uses a set (A) of (directed) arcs, obtained by including the two arcs $((i, j)$ and $(j, i))$ for each edge $(\{i, j\} \in E)$. A node (s) is arbitrarily selected as the root (node 0 in our implementation), and the model aims to send one unit of flow from the source to each other node.

The new model can be formally defined as follows: A variable $(x_{ij} \in \mathbb{Z}_0^+)$ is introduced for each $(i, j) \in A$ and takes the value of the flow transiting the arc $((i, j) \in A)$. An additional set of variables is introduced to link the multi-commodity flow formulation to the MSTC. Then, a binary variable (y_{ij}) is defined for each edge $(\{i, j\} \in E)$. It takes a value of 1 if the edge is a part of the solution and 0 otherwise. The resulting compact model, characterized by a polynomial number of variables and constraints, is as follows:

$$\begin{aligned} \min \quad & \sum_{\{i,j\} \in E} u_{ij} y_{ij} & (1) \\ \text{s.t.} \quad & \sum_{(j,i) \in A} x_{ji} - \sum_{(i,j) \in A} x_{ij} = \begin{cases} -|V| + 1 & \text{if } i = s \\ +1 & \text{if } i \neq s \end{cases} & i \in V \quad (2) \\ & x_{ij} \leq (|V| - 1) y_{ij} & \{i, j\} \in E \quad (3) \\ & x_{ji} \leq (|V| - 1) y_{ij} & \{i, j\} \in E \quad (4) \\ & y_{ij} + y_{kl} \leq 1 & \{i, j\} \in E, (k, l) \in \delta_{ij} \quad (5) \\ & x_{ij} \geq 0 & (i, j) \in A \quad (6) \\ & y_{ij} \in \{0, 1\} & \{i, j\} \in E \quad (7) \end{aligned}$$

The objective function (1) minimizes the cost of the spanning tree. Constraints (2) implement a classical multi-commodity flow model on the directed network and impose that one unit of the commodity has to exit the source (s) for each other node of V (case $i = s$) and that each node different from s has to retain exactly one unit of the commodity (case $i \neq s$). An example of the flow associated with the solution reported in Figure 1 is available in Figure 2.

Inequalities (3) and (4) connect the x and y variables by imposing that y_{ij} has to take a value of 1 as soon as either x_{ij} or x_{ji} takes a positive value. Constraints (5) encode conflict constraints by ensuring that conflicting edges are not selected simultaneously. Finally, the domains of the variables are specified in constraints (6) and (7) for the x and y variables, respectively.

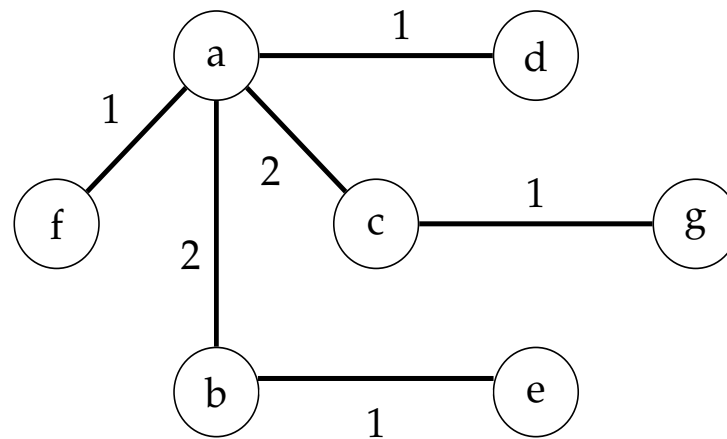


Figure 2. Assignment of flow variables (f) for the solution reported in Figure 1, assuming node a is designed to be the source node (s).

4. Computational Experiments

This section is devoted to the experimental validation of the new compact model introduced in Section 3. The model's results are compared to those of existing methods for the standard benchmark instances from the literature. In detail, Section 4.1 is devoted to the description of the benchmark instances previously introduced in the literature and used in the present study. An extensive comparison between the results achieved in this work and the results previously published is presented in Section 4.2.

4.1. Benchmark Instances

The benchmark sets originally proposed in [3,7], which have been widely used in the previous literature to validate and compare solving methods, are adopted for the experiments reported in the present work.

From the benchmark set proposed in [3], the 50 most challenging instances are considered, since the other ones can easily be solved to optimality by any recent method. The instances are generated at random and have a number of nodes that ranges between 50 and 300 and a number of edges between 200 and 1000. The edge costs are between 0 and 500, and different densities for the conflict graph are considered. We refer the interested reader to [3] for full details on instance generation.

A newer set of benchmark instances was recently introduced in [7] and is formed by random instances with the number of nodes between 25 and 100 and the edge density equal to 0.2, 0.3, or 0.4 (notice that the previous set of benchmarks was characterized by lower densities). Costs are between 1 and 30, and the density of the conflict graph is 0.01, 0.04, or 0.07. All 180 instances of the set are considered for the experiments reported in the present work. We refer the reader interested in the generation process of the instances to [7] for full details.

4.2. Experimental Results

The model discussed in Section 3 has been solved with the Google OR-Tools CP-SAT solver [41] version 9.12. The CP-SAT solver is designed to work in a multi-thread environment (compatible with all new processors) and implements a portfolio strategy with effective data exchange among the different threads. The main process runs a Constraint-Programming Solver based on Lazy Clause Generation (LCG) [46]. The concept behind LCG involves the (incremental) transformation of the problem to an SAT formula, subsequently employing an SAT solver to seek a solution (or prove bounds for infeasibility). The input model is also linearized, the corresponding linear program is (partially) solved with the (dual) simplex algorithm, and other classical MILP techniques are run to enhance bounds

and retrieve new solutions, aiming at supporting the satisfiability model. Finally, instances of metaheuristic methods, seeking high-quality feasible solutions, are executed.

We believe a compact model heavily based on binary variables, like those discussed in Section 3, might be very suitable to be attacked with a solver with the characteristics of CP-SAT.

The experiments have been run on a computer equipped with an Intel Core i7 12700F CPU, but the computational times were normalized using the Intel Core i7-3770 CPU used for the experiments reported in [5–7], as a baseline. This ensures fair comparisons across methods. The normalization of the computation times has been carried out according to the conversion ratios inferred from [47]. Notice that ref. [47] is commonly used in the community to compare results obtained on different machines (see, for example, [32,44]). In accordance with the tests previously reported in [7], a maximum computation time of 5010 s is allowed for all the methods for which a computation time is reported in the tables that follow.

The first experiments have been carried out on the instances proposed in [3] and are summarized in Table 1. The first columns identify each instance (n is the number of nodes, m is the number of edges, and p is the number of conflicts, following the parameters used to generate the instances), while the rest of the table summarizes all the relevant results presented in the previous literature. The (not necessarily) optimal results obtained by solving a Mixed-Integer Linear Programming (MILP) model and two lower-bounding procedures (LB-MST and LB-MI) are taken from [3] (with no computation time). The lower and upper bounds (HDA and HDA+, respectively) from [6] are reported together with the upper bounds obtained with the multi-ethnic genetic metaheuristic method that appeared in [5] (Mega) and the lower and upper bounds obtained by the two branch-and-cut approaches discussed in [4] (unfortunately, without computation times) and [7], respectively (BC). Finally, the results obtained by the method presented in this paper are presented in the last five columns (CP-SAT). On top of the lower and upper bounds obtained and the relative computation time, the percentage deviation (Dev%) against the best lower and upper bounds is also reported to better position the new approach. These deviations are calculated as $100 \cdot \frac{BK_{LB} - LB}{BK_{LB}}$ and $100 \cdot \frac{UB - BK_{UB}}{BK_{UB}}$, respectively, where BK_{LB} and BK_{UB} are the best-known lower and upper bounds available from the previous literature, and LB and UB are the values provided by CP-SAT. Negative deviations (corresponding to improved best-known results) are highlighted in bold.

The results in Table 1 suggest that the approach we propose (CP-SAT) is particularly effective in identifying infeasible instances: Five such instances were identified for the first time in this study by the powerful feasibility checker provided by the adopted solver (notice that infeasible instances are marked as Infeas in the table). This corroborates the intuition that the logical inference carried out by CP-SAT can be effective on the compact model we propose. Moreover, another best-known upper bound has been improved. In general, the new approach has a performance comparable with those of the other state-of-the-art methods available, with deviation gaps remaining below 8.5% (and, in general, substantially below) from best-known bounds (taken as the best of the results obtained by several different algorithms). We observe that the average deviations reported are not very significant because they are heavily affected by the figures for the infeasibility proofs. Notably, CP-SAT consistently provides meaningful lower and upper bounds, demonstrating a level of robustness—especially on the upper bounds—that is uncommon among other methods. On the other hand, the lower bounds found by CP-SAT appear sometimes worse than those of BC (from [7]) in some of the most challenging instances. This might indicate scalability issues. When observing computation times, CP-SAT often appears slower than the other methods. This likely depends on the model we propose, which is characterized by a poor linear relaxation, which, in turn, means that no strong lower bound

from linear programming is available to the solver. CP-SAT effectively compensates for this lack of information by applying logical inference, although this takes a longer time, showing that the solver is suitable for the new model, and the combination leads to very good results.

The percentage deviations from the best-known (before the present study) lower and upper bounds are plotted for all the methods considered and for each instance of the first benchmark set in Figures 3 and 4. The plots help to position the new approach even more clearly. A plot representing the computation times required to close each instance by the two exact methods for which such information is available (BC [7] and CP-SAT) is provided in Figure 5. A negative value is plotted when no result is available for the solver BC. The plot suggests how CP-SAT is often generally slower in the instances solved by the two methods, but, on the other hand, the number of instances solved by such an approach is higher.

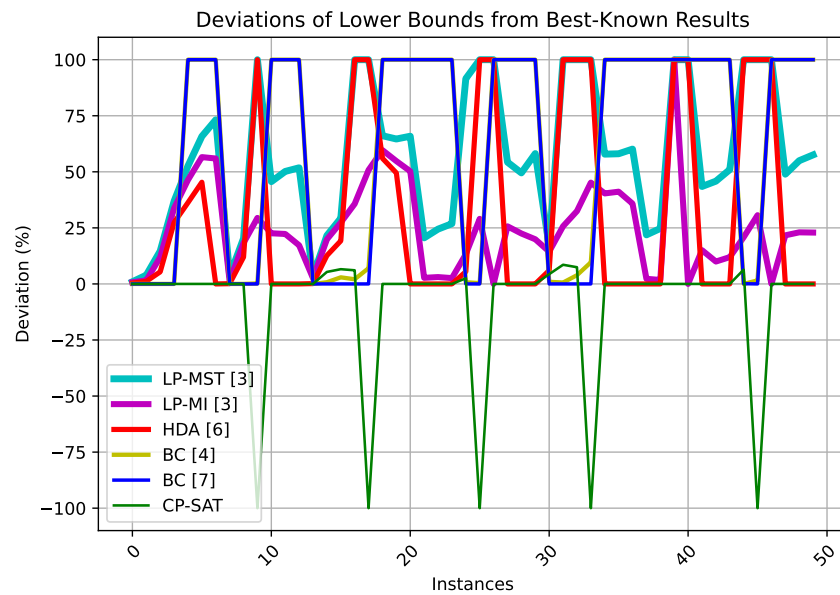


Figure 3. Percentage deviations from the best-known lower bounds in the instances of the first benchmark set.

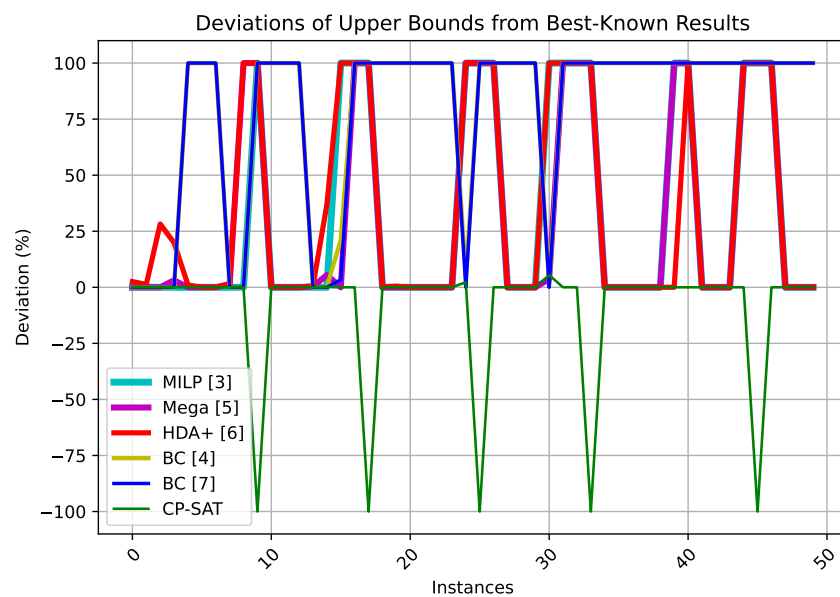


Figure 4. Percentage deviations from the best-known upper bounds in the instances of the first benchmark set.

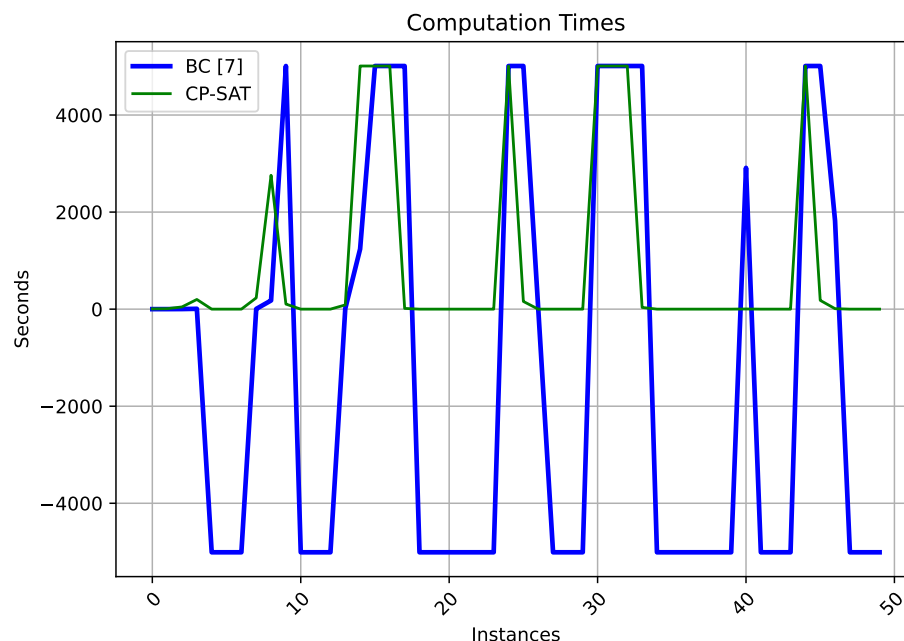


Figure 5. Computation times for two exact methods in the instances of the first benchmark set.

A second set of experiments has been carried out on the instances proposed in [7], and the results are summarized in Table 2. The meaning of the column matches that of the column in Table 1, with the addition of parameter s , which is the seed used to generate the instances and helps to distinguish among instances with the same characteristics. The results of all the methods for which results are available in this dataset (namely, the most recent approaches) are reported. Computational times are available for all the methods reported.

The comparison proposed in Table 2 shows that the approach we propose (CP-SAT) also performs well in this second set of benchmark instances, with average deviations from the previously best-known results of 0.4% concerning the lower bounds and 0.8% concerning the upper bounds. These figures are very good, considering the abundance of methods considered while calculating the (previous) best-known results. The maximum deviation is negligible for most of the instances, with only a few cases above 10% for upper bounds and never above 2.2% for lower bounds. Notice that for some of the instances of this benchmark set, CP-SAT suffers on the heuristic solution side (upper bounds). This indicates that several tailored heuristics presented for the problem were well motivated. A total of four new best-known lower bounds and ten new best-known upper bounds have been produced during the experimental campaign, with one instance (50-245-2093-349) closed for the first time during this experimental campaign. The same conclusion drawn for computation times in Table 1 also applies to these experiments too.

Analogous to what was done for the first benchmark set, the percentage deviations from the best-known (before the present study) lower and upper bounds are plotted for all the methods considered and for each instance of the first benchmark set in Figures 6 and 7. The plots help to position the new approach even more clearly, although the dominance of CP-SAT here is not as clear as in Figures 3 and 4. We plot the computation times required to close each instance by the two exact methods for which such information is available in Figure 8. The plot suggests that for the second benchmark set, BS converges faster (and closes a few more instances) than CP-SAT.

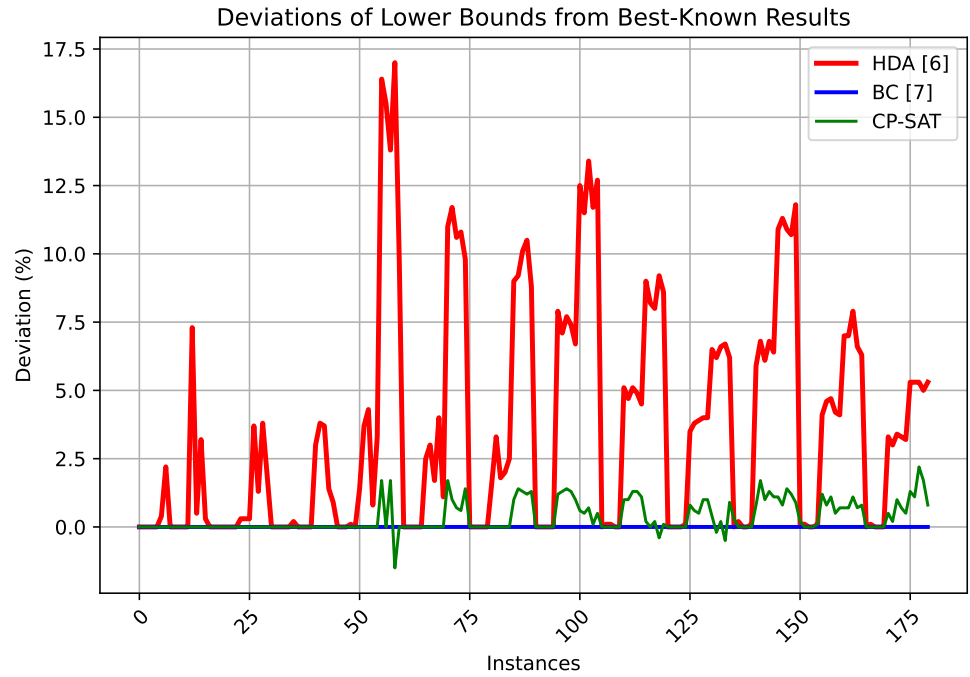


Figure 6. Percentage deviations from the best-known lower bounds in the instances of the second benchmark set.

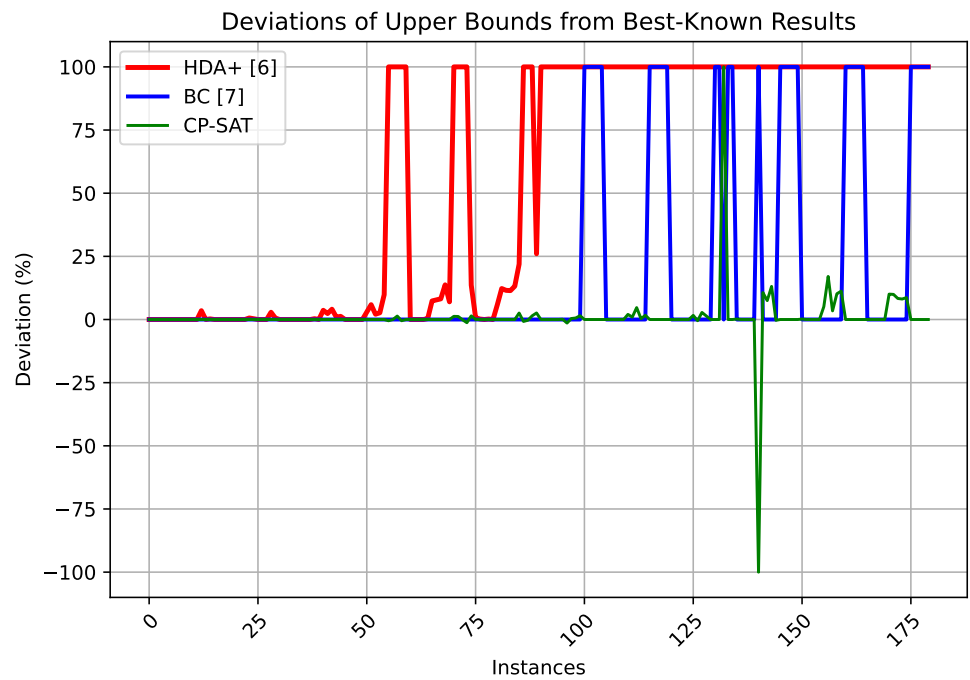


Figure 7. Percentage deviations from the best-known upper bounds in the instances of the second benchmark set.

Table 1. Computational results for the instances introduced in [3].

| Instances | | | MILP [3] | LB-MST [3] | LB-MI [3] | HDA [6] | | Mega [5] | | HDA+ [6] | | BC [4] | | BC [7] | | | CP-SAT | | | Dev% | |
|-----------|-----|-------|----------|------------|-----------|---------|------|----------|------|----------|------|---------|--------|---------|--------|--------|--------|--------|--------|--------|--------|
| n | m | p | UB | LB | LB | LB | Sec | UB | Sec | UB | Sec | LB | UB | LB | UB | Sec | LB | UB | Sec | LB | UB |
| 50 | 200 | 199 | 708 | 701.1 | 702.8 | 705.5 | 0.3 | 708 | 0.7 | 726 | 0.4 | 708.0 | 708 | 708.0 | 708 | 0.2 | 708 | 708 | 5.1 | 0.0 | 0.0 |
| 50 | 200 | 398 | 770 | 739.8 | 757.8 | 761.0 | 0.3 | 770 | 0.7 | 780 | 0.5 | 770.0 | 770 | 770.0 | 770 | 0.5 | 770 | 770 | 9.2 | 0.0 | 0.0 |
| 50 | 200 | 597 | 917 | 782.7 | 807.8 | 867.7 | 0.4 | 917 | 0.6 | 1176 | 0.6 | 917.0 | 917 | 917.0 | 917 | 1.8 | 917 | 917 | 46.0 | 0.0 | 0.0 |
| 50 | 200 | 995 | 1324 | 835.7 | 877.5 | 961.9 | 0.4 | 1365.4 | 0.6 | 1587 | 0.8 | 1324.0 | 1324 | 1324.0 | 1324 | 7.4 | 1324 | 1324 | 199.6 | 0.0 | 0.0 |
| 50 | 200 | 3903 | 1636 | 775.1 | 877.5 | 1042.8 | 1.2 | 1636 | 0.5 | 1653 | 2.2 | - | - | - | - | - | 1636 | 1636 | 0.1 | 0.0 | 0.0 |
| 50 | 200 | 4877 | 2043 | 698.7 | 887.5 | 1116.3 | 1.6 | 2043 | 0.5 | 2043 | 2.5 | - | - | - | - | - | 2043 | 2043 | 0.2 | 0.0 | 0.0 |
| 50 | 200 | 5864 | 2338 | 626.9 | 1030.3 | 2338.0 | 0.8 | 2338 | 0.6 | 2338 | 0.8 | - | - | - | - | - | 2338 | 2338 | 0.1 | 0.0 | 0.0 |
| 100 | 300 | 448 | 4041 | 3893.5 | 3991.2 | 4036.6 | 0.9 | 4099.2 | 0.6 | 4099 | 1.2 | 4041.0 | 4041 | 4041.0 | 4041 | 4.6 | 4041 | 4041 | 232.0 | 0.0 | 0.0 |
| 100 | 300 | 897 | 5658 | 4508.2 | 4624.2 | 4982.0 | 1.4 | - | 1.9 | - | 2.1 | 5658.0 | 5658 | 5658.0 | 5658 | 178.5 | 5658 | 5658 | 2758.6 | 0.0 | 0.0 |
| 100 | 300 | 1344 | - | - | 4681.3 | - | - | - | 2.9 | - | - | 6621.2 | - | 6635.4 | - | 5010.0 | Infeas | Infeas | 104.6 | -100.0 | -100.0 |
| 100 | 300 | 8609 | 7434 | 4043.4 | 5754.9 | 7434.0 | 0.6 | 7434 | 1.9 | 7434 | 0.6 | - | - | - | - | - | 7434 | 7434 | 0.1 | 0.0 | 0.0 |
| 100 | 300 | 10686 | 7968 | 3970.5 | 6192.3 | 7968.0 | 0.6 | 7968 | 1.8 | 7968 | 0.5 | - | - | - | - | - | 7968 | 7968 | 0.2 | 0.0 | 0.0 |
| 100 | 300 | 12761 | 8166 | 3936.3 | 6758.6 | 8166.0 | 0.5 | 8166 | 1.9 | 8166 | 0.5 | - | - | - | - | - | 8166 | 8166 | 0.1 | 0.0 | 0.0 |
| 100 | 500 | 1247 | 4275 | 4124.5 | 4165.7 | 4268.6 | 2.3 | 4291.2 | 5.3 | 4301 | 3.0 | 4275.0 | 4275 | 4275.0 | 4275 | 11.5 | 4275 | 4275 | 86.6 | 0.0 | 0.0 |
| 100 | 500 | 2495 | 5997 | 4701.9 | 4805.4 | 5238.2 | 2.5 | 6325 | 5.0 | 8214 | 5.5 | 5951.4 | 6006 | 5997.0 | 5997 | 1239.4 | 5676 | 6005 | 5010.0 | 5.4 | 0.1 |
| 100 | 500 | 3741 | 7665 | 4743.8 | 4871.3 | 5418.8 | 2.6 | 7788 | 3.2 | - | 6.8 | 6510.8 | 9440 | 6707.8 | 8049 | 5010.1 | 6267 | 7787 | 5010.0 | 6.6 | -0.013 |
| 100 | 500 | 6237 | - | - | 4969.0 | - | - | - | 5.0 | - | - | 7568.7 | - | 7729.3 | - | 5010.0 | 7259 | - | 5010.0 | 6.1 | 0.0 |
| 100 | 500 | 12474 | - | - | 5194.7 | - | - | - | 6.9 | - | - | 9816.9 | - | 10560.2 | - | 5010.0 | Infeas | Infeas | 15.1 | -100.0 | -100.0 |
| 100 | 500 | 24740 | 12652 | 4289.9 | 5104.9 | 5565.2 | 9.0 | 12652 | 3.4 | 12681 | 20.8 | - | - | - | - | - | 12652 | 12652 | 0.1 | 0.0 | 0.0 |
| 100 | 500 | 30886 | 11232 | 3972.7 | 5078.8 | 5672.7 | 15.5 | 11232 | 3.6 | 11287 | 26.4 | - | - | - | - | - | 11232 | 11232 | 0.1 | 0.0 | 0.0 |
| 100 | 500 | 36827 | 11481 | 3918.1 | 5710.8 | 11481.0 | 22.9 | 11481 | 3.6 | 11481 | 24.4 | - | - | - | - | - | 11481 | 11481 | 0.2 | 0.0 | 0.0 |
| 200 | 400 | 13660 | 17728 | 14085.8 | 17245.9 | 17728.0 | 1.3 | 17728 | 7.7 | 17728 | 1.4 | - | - | - | - | - | 17728 | 17728 | 0.1 | 0.0 | 0.0 |
| 200 | 400 | 17089 | 18617 | 14067.3 | 18048.2 | 18617.0 | 1.5 | 18617 | 7.7 | 18617 | 1.4 | - | - | - | - | - | 18617 | 18617 | 0.1 | 0.0 | 0.0 |
| 200 | 400 | 20470 | 19140 | 13998.7 | 18646.2 | 19140.0 | 1.4 | 19140 | 7.6 | 19140 | 1.4 | - | - | - | - | - | 19140 | 19140 | 0.1 | 0.0 | 0.0 |
| 200 | 600 | 1797 | 15029 | 1111.6 | 11425.8 | 12451.6 | 4.9 | - | 12.6 | - | 8.6 | 13072.9 | 14707 | 13171.2 | 14086 | 5010.0 | 12841 | 14405 | 5010.0 | 2.5 | 2.3 |
| 200 | 600 | 3594 | - | - | 12487.0 | - | - | - | 24.0 | - | - | 17532.7 | - | 17595.0 | - | 5010.0 | 0 | 0 | 158.9 | -100.0 | -100.0 |
| 200 | 600 | 5391 | - | - | 12873.2 | - | - | - | 31.6 | - | - | Infeas | Infeas | Infeas | Infeas | 16.4 | Infeas | Infeas | 0.0 | 0.0 | 0.0 |
| 200 | 600 | 34504 | 20716 | 9466.1 | 15393.1 | 20716.0 | 3.0 | 20716 | 11.6 | 20716 | 3.0 | - | - | - | - | - | 20716 | 20716 | 0.1 | 0.0 | 0.0 |
| 200 | 600 | 42860 | 18025 | 9100.6 | 13971.5 | 18025.0 | 2.1 | 18025 | 12.0 | 18025 | 2.2 | - | - | - | - | - | 18025 | 18025 | 0.2 | 0.0 | 0.0 |
| 200 | 600 | 50984 | 20864 | 8734.5 | 16708.1 | 20864.0 | 1.9 | 20864 | 12.5 | 20864 | 2.0 | - | - | - | - | - | 20864 | 20864 | 0.1 | 0.0 | 0.0 |
| 200 | 800 | 3196 | 22110 | 17428.8 | 17922.6 | 19685.1 | 8.6 | 22350.8 | 22.2 | - | 18.9 | 20744.2 | 21852 | 20941.5 | 21553 | 5010.1 | 20030 | 22722 | 5010.0 | 4.4 | 5.4 |
| 200 | 800 | 6392 | - | - | 19705.7 | - | - | - | 28.4 | - | - | 26361.3 | - | 26526.7 | - | 5010.1 | 24259 | - | 5010.0 | 8.5 | 0.0 |
| 200 | 800 | 9588 | - | - | 20684.8 | - | - | - | 38.3 | - | - | 29443.6 | - | 30634.2 | - | 5010.0 | 28351 | - | 5010.0 | 7.5 | 0.0 |
| 200 | 800 | 15980 | - | - | 20226.9 | - | - | - | 48.0 | - | - | 33345.1 | - | 36900.2 | - | 5010.0 | 0 | 0 | 35.4 | -100.0 | -100.0 |
| 200 | 800 | 62625 | 39895 | 16806.5 | 23792.3 | 39895.0 | 43.3 | 39895 | 16.7 | 39895 | 45.8 | - | - | - | - | - | 39895 | 39895 | 0.2 | 0.0 | 0.0 |
| 200 | 800 | 78387 | 37671 | 15803.1 | 22174.2 | 37671.0 | 7.7 | 37671 | 15.9 | 37671 | 8.2 | - | - | - | - | - | 37671 | 37671 | 0.2 | 0.0 | 0.0 |
| 200 | 800 | 93978 | 38798 | 15470.1 | 24907.0 | 38798.0 | 4.5 | 38798 | 17.3 | 38798 | 4.7 | - | - | - | - | - | 38798 | 38798 | 0.1 | 0.0 | 0.0 |

Table 1. Cont.

| Instances | | | MILP [3] | LB-MST [3] | LB-MI [3] | HDA [6] | | Mega [5] | | HDA+ [6] | | BC [4] | | BC [7] | | CP-SAT | | | Dev% | | |
|-----------|------|--------|----------|------------|-----------|---------|-----|----------|-------|----------|-----|---------|--------|---------|--------|--------|--------|--------|--------|--------|--------|
| n | m | p | UB | LB | LB | LB | Sec | UB | Sec | UB | Sec | LB | UB | LB | UB | Sec | LB | UB | Sec | LB | UB |
| 300 | 600 | 31000 | 43721 | 34154.2 | 42720.6 | 43721.0 | 3.0 | 43721 | 19.2 | 43721 | 3.4 | - | - | - | - | - | 43721 | 43721 | 0.1 | 0.0 | 0.0 |
| 300 | 600 | 38216 | 44267 | 33320.1 | 43486.7 | 44267.0 | 3.1 | 44267 | 22.3 | 44267 | 3.4 | - | - | - | - | - | 44267 | 44267 | 0.2 | 0.0 | 0.0 |
| 300 | 600 | 45310 | 43071 | 32072.3 | 42149.0 | 43071.0 | 3.0 | 43071 | 24.6 | 43071 | 3.1 | - | - | - | - | - | 43071 | 43071 | 0.3 | 0.0 | 0.0 |
| 300 | 800 | 3196 | - | - | 30190.1 | - | - | - | 62.8 | - | - | Infeas | Infeas | Infeas | Infeas | 2911.1 | Infeas | Infeas | 3.3 | 0.0 | 0.0 |
| 300 | 800 | 59600 | 43125 | 24384.3 | 36629.6 | 43125.0 | 3.3 | 43125 | 32.7 | 43125 | 3.1 | - | - | - | - | - | 43125 | 43125 | 0.1 | 0.0 | 0.0 |
| 300 | 800 | 74500 | 42292 | 22913.2 | 38069.3 | 42292.0 | 3.2 | 42292 | 35.6 | 42292 | 3.2 | - | - | - | - | - | 42292 | 42292 | 0.3 | 0.0 | 0.0 |
| 300 | 800 | 89300 | 44114 | 21624.6 | 38843.0 | 44114.0 | 3.4 | 44114 | 35.4 | 44114 | 3.2 | - | - | - | - | - | 44114 | 44114 | 0.1 | 0.0 | 0.0 |
| 300 | 1000 | 4995 | - | - | 40732.7 | - | - | - | 79.3 | - | - | 51451.3 | - | 51398.4 | - | 5010.0 | 48180 | - | 5010.0 | 6.4 | 0.0 |
| 300 | 1000 | 9990 | - | - | 42902.5 | - | - | - | 119.1 | - | - | 60907.8 | - | 61878.9 | - | 5010.0 | Infeas | Infeas | 181.3 | -100.0 | -100.0 |
| 300 | 1000 | 14985 | - | - | 44639.1 | - | - | - | 141.3 | - | - | Infeas | Infeas | Infeas | Infeas | 1820.0 | Infeas | Infeas | 9.7 | 0.0 | 0.0 |
| 300 | 1000 | 96590 | 71562 | 36544.7 | 56048.3 | 71562.0 | 9.2 | 71562 | 41.5 | 71562 | 9.5 | - | - | - | - | - | 71562 | 71562 | 0.1 | 0.0 | 0.0 |
| 300 | 1000 | 120500 | 76345 | 34380.8 | 58780.1 | 76345.0 | 6.6 | 76345 | 33.4 | 76345 | 6.7 | - | - | - | - | - | 76345 | 76345 | 0.2 | 0.0 | 0.0 |
| 300 | 1000 | 144090 | 78880 | 33481.2 | 60810.8 | 78880.0 | 5.2 | 78880 | 36.9 | 78880 | 5.4 | - | - | - | - | - | 78880 | 78880 | 0.1 | 0.0 | 0.0 |
| Averages | | | | | | | | | | | | | | | | | | | -9.1 | -9.8 | |

Table 2. Computational results for the instances introduced in [7].

| Instances | | | | HDA [6] | | HDA+ [6] | | BC [7] | | | CP-SAT | | | Dev% | |
|-----------|----------|----------|----------|---------|------|----------|------|--------|-----|--------|--------|-----|---------|------|------|
| <i>n</i> | <i>m</i> | <i>p</i> | <i>s</i> | LB | Sec | UB | Sec | LB | UB | Sec | LB | UB | Sec | LB | UB |
| 25 | 60 | 18 | 1 | 347.0 | 0.04 | 347 | 0.05 | 347.0 | 347 | 0.0 | 347 | 347 | 0.23 | 0.0 | 0.0 |
| 25 | 60 | 18 | 7 | 389.0 | 0.04 | 389 | 0.04 | 389.0 | 389 | 0.0 | 389 | 389 | 0.18 | 0.0 | 0.0 |
| 25 | 60 | 18 | 13 | 353.0 | 0.04 | 353 | 0.04 | 353.0 | 353 | 0.0 | 353 | 353 | 0.19 | 0.0 | 0.0 |
| 25 | 60 | 18 | 19 | 346.0 | 0.04 | 346 | 0.05 | 346.0 | 346 | 0.0 | 346 | 346 | 0.19 | 0.0 | 0.0 |
| 25 | 60 | 18 | 25 | 336.0 | 0.04 | 336 | 0.04 | 336.0 | 336 | 0.0 | 336 | 336 | 0.18 | 0.0 | 0.0 |
| 25 | 90 | 71 | 31 | 379.6 | 0.09 | 381 | 0.07 | 381.0 | 381 | 0.0 | 381 | 381 | 0.45 | 0.0 | 0.0 |
| 25 | 60 | 71 | 37 | 381.5 | 0.05 | 390 | 0.07 | 390.0 | 390 | 0.1 | 390 | 390 | 0.32 | 0.0 | 0.0 |
| 25 | 60 | 71 | 43 | 372.0 | 0.05 | 372 | 0.06 | 372.0 | 372 | 0.0 | 372 | 372 | 0.44 | 0.0 | 0.0 |
| 25 | 60 | 71 | 49 | 357.0 | 0.05 | 357 | 0.06 | 357.0 | 357 | 0.0 | 357 | 357 | 0.24 | 0.0 | 0.0 |
| 25 | 60 | 71 | 55 | 406.0 | 0.05 | 406 | 0.06 | 406.0 | 406 | 0.0 | 406 | 406 | 0.38 | 0.0 | 0.0 |
| 25 | 60 | 124 | 61 | 385.0 | 0.06 | 385 | 0.07 | 385.0 | 385 | 0.0 | 385 | 385 | 0.18 | 0.0 | 0.0 |
| 25 | 60 | 124 | 67 | 432.0 | 0.06 | 432 | 0.07 | 432.0 | 432 | 0.0 | 432 | 432 | 0.20 | 0.0 | 0.0 |
| 25 | 60 | 124 | 73 | 424.5 | 0.07 | 474 | 0.1 | 458.0 | 458 | 0.3 | 458 | 458 | 0.48 | 0.0 | 0.0 |
| 25 | 60 | 124 | 79 | 398.0 | 0.07 | 400 | 0.08 | 400.0 | 400 | 0.0 | 400 | 400 | 0.45 | 0.0 | 0.0 |
| 25 | 60 | 124 | 85 | 406.8 | 0.08 | 421 | 0.12 | 420.0 | 420 | 0.0 | 420 | 420 | 0.42 | 0.0 | 0.0 |
| 25 | 90 | 41 | 91 | 310.1 | 0.05 | 311 | 0.05 | 311.0 | 311 | 0.0 | 311 | 311 | 0.54 | 0.0 | 0.0 |
| 25 | 90 | 41 | 97 | 306.0 | 0.05 | 306 | 0.06 | 306.0 | 306 | 0.0 | 306 | 306 | 0.36 | 0.0 | 0.0 |
| 25 | 90 | 41 | 103 | 299.0 | 0.05 | 299 | 0.05 | 299.0 | 299 | 0.0 | 299 | 299 | 0.31 | 0.0 | 0.0 |
| 25 | 90 | 41 | 109 | 297.0 | 0.05 | 297 | 0.05 | 297.0 | 297 | 0.0 | 297 | 297 | 0.30 | 0.0 | 0.0 |
| 25 | 90 | 41 | 115 | 318.0 | 0.05 | 318 | 0.05 | 318.0 | 318 | 0.0 | 318 | 318 | 0.25 | 0.0 | 0.0 |
| 25 | 90 | 161 | 121 | 305.0 | 0.08 | 305 | 0.09 | 305.0 | 305 | 0.0 | 305 | 305 | 0.54 | 0.0 | 0.0 |
| 25 | 90 | 161 | 127 | 339.0 | 0.12 | 339 | 0.11 | 339.0 | 339 | 0.0 | 339 | 339 | 0.29 | 0.0 | 0.0 |
| 25 | 90 | 161 | 133 | 344.0 | 0.08 | 344 | 0.09 | 344.0 | 344 | 0.0 | 344 | 344 | 0.23 | 0.0 | 0.0 |
| 25 | 90 | 161 | 139 | 328.0 | 0.09 | 331 | 0.11 | 329.0 | 329 | 0.0 | 329 | 329 | 0.61 | 0.0 | 0.0 |
| 25 | 90 | 161 | 145 | 325.0 | 0.08 | 327 | 0.11 | 326.0 | 326 | 0.0 | 326 | 326 | 0.89 | 0.0 | 0.0 |
| 25 | 90 | 281 | 151 | 347.9 | 0.1 | 349 | 0.13 | 349.0 | 349 | 0.0 | 349 | 349 | 0.56 | 0.0 | 0.0 |
| 25 | 90 | 281 | 157 | 370.6 | 0.1 | 385 | 0.16 | 385.0 | 385 | 0.5 | 385 | 385 | 2.67 | 0.0 | 0.0 |
| 25 | 90 | 281 | 163 | 330.8 | 0.11 | 335 | 0.14 | 335.0 | 335 | 0.0 | 335 | 335 | 0.66 | 0.0 | 0.0 |
| 25 | 90 | 281 | 169 | 334.7 | 0.1 | 358 | 0.18 | 348.0 | 348 | 0.1 | 348 | 348 | 1.91 | 0.0 | 0.0 |
| 25 | 90 | 281 | 175 | 350.1 | 0.1 | 359 | 0.17 | 357.0 | 357 | 0.0 | 357 | 357 | 0.98 | 0.0 | 0.0 |
| 25 | 120 | 72 | 181 | 282.0 | 0.07 | 282 | 0.06 | 282.0 | 282 | 0.0 | 282 | 282 | 0.24 | 0.0 | 0.0 |
| 25 | 120 | 72 | 187 | 294.0 | 0.06 | 294 | 0.06 | 294.0 | 294 | 0.0 | 294 | 294 | 0.90 | 0.0 | 0.0 |
| 25 | 120 | 72 | 193 | 284.0 | 0.06 | 284 | 0.08 | 284.0 | 284 | 0.0 | 284 | 284 | 0.48 | 0.0 | 0.0 |
| 25 | 120 | 72 | 199 | 281.0 | 0.06 | 281 | 0.07 | 281.0 | 281 | 0.0 | 281 | 281 | 0.53 | 0.0 | 0.0 |
| 25 | 120 | 72 | 205 | 292.0 | 0.06 | 292 | 0.06 | 292.0 | 292 | 0.0 | 292 | 292 | 0.30 | 0.0 | 0.0 |
| 25 | 120 | 286 | 211 | 320.3 | 0.12 | 321 | 0.16 | 321.0 | 321 | 0.0 | 321 | 321 | 1.06 | 0.0 | 0.0 |
| 25 | 120 | 286 | 217 | 317.0 | 0.12 | 317 | 0.17 | 317.0 | 317 | 0.0 | 317 | 317 | 0.64 | 0.0 | 0.0 |
| 25 | 120 | 286 | 223 | 284.0 | 0.56 | 284 | 0.12 | 284.0 | 284 | 0.0 | 284 | 284 | 0.31 | 0.0 | 0.0 |
| 25 | 120 | 286 | 229 | 311.0 | 0.12 | 312 | 0.17 | 311.0 | 311 | 0.0 | 311 | 311 | 0.58 | 0.0 | 0.0 |
| 25 | 120 | 286 | 235 | 290.0 | 0.13 | 290 | 0.13 | 290.0 | 290 | 0.0 | 290 | 290 | 0.36 | 0.0 | 0.0 |
| 25 | 120 | 500 | 241 | 319.2 | 0.16 | 341 | 0.26 | 329.0 | 329 | 0.1 | 329 | 329 | 4.72 | 0.0 | 0.0 |
| 25 | 120 | 500 | 247 | 326.1 | 0.16 | 347 | 0.25 | 339.0 | 339 | 0.5 | 339 | 339 | 1.84 | 0.0 | 0.0 |
| 25 | 120 | 500 | 253 | 354.5 | 0.16 | 383 | 0.3 | 368.0 | 368 | 0.4 | 368 | 368 | 1.55 | 0.0 | 0.0 |
| 25 | 120 | 500 | 259 | 306.5 | 0.16 | 314 | 0.22 | 311.0 | 311 | 0.0 | 311 | 311 | 0.65 | 0.0 | 0.0 |
| 25 | 120 | 500 | 265 | 318.0 | 0.16 | 325 | 0.25 | 321.0 | 321 | 0.0 | 321 | 321 | 0.74 | 0.0 | 0.0 |
| 50 | 245 | 299 | 271 | 619.0 | 0.34 | 619 | 0.37 | 619.0 | 619 | 0.0 | 619 | 619 | 2.75 | 0.0 | 0.0 |
| 50 | 245 | 299 | 277 | 604.0 | 0.36 | 604 | 0.38 | 604.0 | 604 | 0.0 | 604 | 604 | 1.49 | 0.0 | 0.0 |
| 50 | 245 | 299 | 283 | 634.0 | 0.34 | 634 | 0.35 | 634.0 | 634 | 0.0 | 634 | 634 | 0.72 | 0.0 | 0.0 |
| 50 | 245 | 299 | 289 | 615.5 | 0.34 | 616 | 0.4 | 616.0 | 616 | 0.1 | 616 | 616 | 1.49 | 0.0 | 0.0 |
| 50 | 245 | 299 | 295 | 595.0 | 0.34 | 595 | 0.44 | 595.0 | 595 | 0.0 | 595 | 595 | 7.28 | 0.0 | 0.0 |
| 50 | 245 | 1196 | 301 | 668.5 | 0.57 | 698 | 0.96 | 678.0 | 678 | 1.4 | 678 | 678 | 27.19 | 0.0 | 0.0 |
| 50 | 245 | 1196 | 307 | 655.5 | 0.56 | 721 | 0.99 | 681.0 | 681 | 3.2 | 681 | 681 | 51.96 | 0.0 | 0.0 |
| 50 | 245 | 1196 | 313 | 678.2 | 0.59 | 725 | 1 | 709.0 | 709 | 6.3 | 709 | 709 | 612.25 | 0.0 | 0.0 |
| 50 | 245 | 1196 | 319 | 634.0 | 1.26 | 656 | 0.86 | 639.0 | 639 | 1.5 | 639 | 639 | 16.54 | 0.0 | 0.0 |
| 50 | 245 | 1196 | 325 | 658.8 | 0.56 | 748 | 1.01 | 681.0 | 681 | 3.8 | 681 | 681 | 57.54 | 0.0 | 0.0 |
| 50 | 245 | 2093 | 331 | 661.3 | 0.76 | - | - | 791.2 | 820 | 5010.0 | 778 | 816 | 5010.00 | 1.7 | -0.5 |
| 50 | 245 | 2093 | 337 | 706.0 | 0.75 | - | - | 835.0 | 835 | 1938.7 | 835 | 835 | 4603.22 | 0.0 | 0.0 |
| 50 | 245 | 2093 | 343 | 666.5 | 0.74 | - | - | 773.2 | 811 | 5010.1 | 760 | 822 | 5010.00 | 1.7 | 1.4 |
| 50 | 245 | 2093 | 349 | 680.8 | 0.76 | - | - | 820.0 | 836 | 5010.1 | 832 | 832 | 2874.48 | -1.5 | -0.5 |
| 50 | 245 | 2093 | 355 | 693.9 | 0.74 | - | - | 769.0 | 769 | 25.7 | 769 | 769 | 266.66 | 0.0 | 0.0 |
| 50 | 367 | 672 | 361 | 570.0 | 0.82 | 570 | 0.88 | 570.0 | 570 | 0.1 | 570 | 570 | 8.05 | 0.0 | 0.0 |
| 50 | 367 | 672 | 367 | 561.0 | 0.86 | 561 | 0.98 | 561.0 | 561 | 1.4 | 561 | 561 | 10.48 | 0.0 | 0.0 |
| 50 | 367 | 672 | 373 | 573.0 | 0.86 | 573 | 1.06 | 573.0 | 573 | 0.0 | 573 | 573 | 2.98 | 0.0 | 0.0 |
| 50 | 367 | 672 | 379 | 560.0 | 0.86 | 560 | 0.87 | 560.0 | 560 | 0.0 | 560 | 560 | 1.61 | 0.0 | 0.0 |
| 50 | 367 | 672 | 385 | 549.0 | 0.83 | 551 | 0.99 | 549.0 | 549 | 0.5 | 549 | 549 | 17.12 | 0.0 | 0.0 |
| 50 | 367 | 2687 | 391 | 596.4 | 1.26 | 657 | 2.37 | 612.0 | 612 | 7.5 | 612 | 612 | 230.76 | 0.0 | 0.0 |
| 50 | 367 | 2687 | 397 | 596.7 | 1.27 | 663 | 2.24 | 615.0 | 615 | 6.6 | 615 | 615 | 154.67 | 0.0 | 0.0 |
| 50 | 367 | 2687 | 403 | 577.1 | 1.26 | 635 | 2.33 | 587.0 | 587 | 3.0 | 587 | 587 | 35.79 | 0.0 | 0.0 |
| 50 | 367 | 2687 | 409 | 608.5 | 1.26 | 721 | 2.4 | 634.0 | 634 | 7.3 | 634 | 634 | 508.17 | 0.0 | 0.0 |
| 50 | 367 | 2687 | 415 | 636.1 | 3.08 | 688 | 2.45 | 643.0 | 643 | 3.2 | 643 | 643 | 72.48 | 0.0 | 0.0 |

Table 2. Cont.

| Instances | | | | HDA [6] | | HDA+ [6] | | BC [7] | | | CP-SAT | | | Dev% | |
|-----------|----------|----------|----------|---------|-------|----------|------|--------|------|--------|--------|------|---------|-------------|---------------|
| <i>n</i> | <i>m</i> | <i>p</i> | <i>s</i> | LB | Sec | UB | Sec | LB | UB | Sec | LB | UB | Sec | LB | UB |
| 50 | 367 | 4702 | 421 | 624.4 | 1.66 | - | - | 701.3 | 726 | 5010.1 | 689 | 735 | 5010.00 | 1.7 | 1.2 |
| 50 | 367 | 4702 | 427 | 635.4 | 1.65 | - | - | 719.5 | 753 | 5010.0 | 712 | 762 | 5010.00 | 1.0 | 1.2 |
| 50 | 367 | 4702 | 433 | 646.9 | 1.64 | - | - | 723.9 | 786 | 5010.0 | 719 | 784 | 5010.00 | 0.7 | -0.3 |
| 50 | 367 | 4702 | 439 | 597.5 | 1.72 | - | - | 669.8 | 711 | 5010.0 | 666 | 702 | 5010.00 | 0.6 | -1.3 |
| 50 | 367 | 4702 | 445 | 665.1 | 1.71 | 868 | 3.36 | 737.3 | 764 | 5010.0 | 727 | 775 | 5010.00 | 1.4 | 1.4 |
| 50 | 490 | 1199 | 451 | 548.0 | 1.68 | 552 | 2.03 | 548.0 | 548 | 0.1 | 548 | 548 | 8.41 | 0.0 | 0.0 |
| 50 | 490 | 1199 | 457 | 530.0 | 1.66 | 531 | 1.88 | 530.0 | 530 | 0.5 | 530 | 530 | 11.77 | 0.0 | 0.0 |
| 50 | 490 | 1199 | 463 | 549.0 | 1.65 | 549 | 1.76 | 549.0 | 549 | 0.0 | 549 | 549 | 3.08 | 0.0 | 0.0 |
| 50 | 490 | 1199 | 469 | 540.0 | 1.67 | 541 | 2.01 | 540.0 | 540 | 0.2 | 540 | 540 | 46.21 | 0.0 | 0.0 |
| 50 | 490 | 1199 | 475 | 540.0 | 1.65 | 540 | 1.71 | 540.0 | 540 | 0.0 | 540 | 540 | 4.77 | 0.0 | 0.0 |
| 50 | 490 | 4793 | 481 | 584.3 | 2.38 | 629 | 4.63 | 594.0 | 594 | 7.8 | 594 | 594 | 109.31 | 0.0 | 0.0 |
| 50 | 490 | 4793 | 487 | 559.9 | 2.4 | 650 | 4.9 | 579.0 | 579 | 13.8 | 579 | 579 | 1473.29 | 0.0 | 0.0 |
| 50 | 490 | 4793 | 493 | 578.3 | 2.43 | 657 | 4.75 | 589.0 | 589 | 3.0 | 589 | 589 | 31.17 | 0.0 | 0.0 |
| 50 | 490 | 4793 | 499 | 565.4 | 2.39 | 643 | 4.74 | 577.0 | 577 | 7.5 | 577 | 577 | 118.57 | 0.0 | 0.0 |
| 50 | 490 | 4793 | 505 | 577.4 | 2.4 | 670 | 4.71 | 592.0 | 592 | 6.0 | 592 | 592 | 40.34 | 0.0 | 0.0 |
| 50 | 490 | 8387 | 511 | 575.0 | 2.4 | 812 | 6.54 | 632.2 | 666 | 5010.0 | 626 | 683 | 5010.00 | 1.0 | 2.6 |
| 50 | 490 | 8387 | 517 | 569.0 | 3.07 | - | - | 626.7 | 651 | 5010.0 | 618 | 646 | 5010.00 | 1.4 | -0.8 |
| 50 | 490 | 8387 | 523 | 592.2 | 3.08 | - | - | 658.4 | 678 | 5010.0 | 650 | 676 | 5010.00 | 1.3 | -0.3 |
| 50 | 490 | 8387 | 529 | 592.5 | 3.07 | - | - | 662.2 | 682 | 5010.1 | 654 | 692 | 5010.00 | 1.2 | 1.5 |
| 50 | 490 | 8387 | 535 | 587.5 | 3.09 | 828 | 6.95 | 644.2 | 657 | 5010.0 | 636 | 674 | 5010.00 | 1.3 | 2.6 |
| 75 | 555 | 1538 | 541 | 868.0 | 2.51 | - | - | 868.0 | 868 | 0.7 | 868 | 868 | 75.24 | 0.0 | 0.0 |
| 75 | 555 | 1538 | 547 | 870.9 | 2.49 | - | - | 871.0 | 871 | 3.0 | 871 | 871 | 58.57 | 0.0 | 0.0 |
| 75 | 555 | 1538 | 553 | 838.0 | 2.45 | - | - | 838.0 | 838 | 0.3 | 838 | 838 | 33.73 | 0.0 | 0.0 |
| 75 | 555 | 1538 | 559 | 855.0 | 2.47 | - | - | 855.0 | 855 | 4.4 | 855 | 855 | 44.64 | 0.0 | 0.0 |
| 75 | 555 | 1538 | 565 | 857.0 | 2.47 | - | - | 857.0 | 857 | 4.1 | 857 | 857 | 48.19 | 0.0 | 0.0 |
| 75 | 555 | 6150 | 571 | 942.3 | 3.49 | - | - | 1023.7 | 1047 | 5010.0 | 1011 | 1048 | 5010.00 | 1.2 | 0.1 |
| 75 | 555 | 6150 | 577 | 937.6 | 3.54 | - | - | 1008.8 | 1069 | 5010.2 | 996 | 1054 | 5010.00 | 1.3 | -1.4 |
| 75 | 555 | 6150 | 583 | 910.9 | 3.48 | - | - | 987.3 | 1040 | 5010.1 | 973 | 1044 | 5010.00 | 1.4 | 0.4 |
| 75 | 555 | 6150 | 589 | 913.0 | 3.53 | - | - | 985.6 | 998 | 5010.1 | 973 | 1004 | 5010.00 | 1.3 | 0.6 |
| 75 | 555 | 6150 | 595 | 897.9 | 3.54 | - | - | 962.6 | 994 | 5010.1 | 953 | 1009 | 5010.00 | 1.0 | 1.5 |
| 75 | 555 | 10762 | 601 | 922.5 | 4.44 | - | - | 1054.5 | - | 4647.3 | 1048 | - | 5010.00 | 0.6 | 0.0 |
| 75 | 555 | 10762 | 607 | 948.0 | 12.01 | - | - | 1070.9 | - | 3483.6 | 1066 | - | 5010.00 | 0.5 | 0.0 |
| 75 | 555 | 10762 | 613 | 901.9 | 4.44 | - | - | 1041.0 | - | 5010.0 | 1034 | - | 5010.00 | 0.7 | 0.0 |
| 75 | 555 | 10762 | 619 | 888.4 | 4.44 | - | - | 1006.4 | - | 5010.0 | 1005 | - | 5010.00 | 0.1 | 0.0 |
| 75 | 555 | 10762 | 625 | 914.7 | 4.42 | - | - | 1048.1 | - | 3208.1 | 1043 | - | 5010.00 | 0.5 | 0.0 |
| 75 | 832 | 3457 | 631 | 797.8 | 6.86 | - | - | 798.0 | 798 | 4.8 | 798 | 798 | 78.24 | 0.0 | 0.0 |
| 75 | 832 | 3457 | 637 | 820.0 | 6.83 | - | - | 821.0 | 821 | 1.1 | 821 | 821 | 637.56 | 0.0 | 0.0 |
| 75 | 832 | 3457 | 643 | 815.3 | 6.87 | - | - | 816.0 | 816 | 4.0 | 816 | 816 | 262.59 | 0.0 | 0.0 |
| 75 | 832 | 3457 | 649 | 820.0 | 7.16 | - | - | 820.0 | 820 | 23.9 | 820 | 820 | 184.06 | 0.0 | 0.0 |
| 75 | 832 | 3457 | 655 | 815.0 | 7.23 | - | - | 815.0 | 815 | 1.4 | 815 | 815 | 84.81 | 0.0 | 0.0 |
| 75 | 832 | 13828 | 661 | 830.9 | 9.08 | - | - | 857.7 | 891 | 5010.0 | 867 | 909 | 5010.00 | 1.0 | 2.0 |
| 75 | 832 | 13828 | 667 | 859.5 | 9.05 | - | - | 901.8 | 947 | 5010.1 | 893 | 956 | 5010.00 | 1.0 | 1.0 |
| 75 | 832 | 13828 | 673 | 829.0 | 9.04 | - | - | 873.7 | 892 | 5010.0 | 862 | 934 | 5010.00 | 1.3 | 4.7 |
| 75 | 832 | 13828 | 679 | 842.2 | 8.98 | - | - | 885.6 | 915 | 4570.8 | 874 | 918 | 5010.00 | 1.3 | 0.3 |
| 75 | 832 | 13828 | 685 | 846.7 | 9.05 | - | - | 886.9 | 896 | 5010.0 | 877 | 911 | 5010.00 | 1.1 | 1.7 |
| 75 | 832 | 24199 | 691 | 870.0 | 12.24 | - | - | 955.8 | - | 1305.3 | 954 | - | 5010.00 | 0.2 | 0.0 |
| 75 | 832 | 24199 | 697 | 837.7 | 12.33 | - | - | 913.0 | - | 1161.2 | 913 | - | 5010.00 | 0.0 | 0.0 |
| 75 | 832 | 24199 | 703 | 842.0 | 27.17 | - | - | 915.0 | - | 953.0 | 913 | - | 5010.00 | 0.2 | 0.0 |
| 75 | 832 | 24199 | 709 | 863.0 | 11.98 | - | - | 950.4 | - | 5010.1 | 954 | - | 5010.00 | -0.4 | 0.0 |
| 75 | 832 | 24199 | 715 | 878.7 | 12.14 | - | - | 960.8 | - | 962.3 | 960 | - | 5010.00 | 0.1 | 0.0 |
| 75 | 1110 | 6155 | 721 | 787.0 | 14.88 | - | - | 787.0 | 787 | 2.1 | 787 | 787 | 237.82 | 0.0 | 0.0 |
| 75 | 1110 | 6155 | 727 | 785.0 | 14.82 | - | - | 785.0 | 785 | 5.4 | 785 | 785 | 214.71 | 0.0 | 0.0 |
| 75 | 1110 | 6155 | 733 | 783.0 | 14.78 | - | - | 783.0 | 783 | 0.0 | 783 | 783 | 102.86 | 0.0 | 0.0 |
| 75 | 1110 | 6155 | 739 | 783.9 | 14.94 | - | - | 784.0 | 784 | 3.3 | 784 | 784 | 128.21 | 0.0 | 0.0 |
| 75 | 1110 | 6155 | 745 | 796.5 | 14.86 | - | - | 797.0 | 797 | 5.7 | 797 | 797 | 479.97 | 0.0 | 0.0 |
| 75 | 1110 | 24620 | 751 | 816.9 | 19.16 | - | - | 846.5 | 857 | 5010.0 | 840 | 871 | 5010.00 | 0.8 | 1.6 |
| 75 | 1110 | 24620 | 757 | 798.0 | 19.28 | - | - | 829.2 | 851 | 4573.5 | 824 | 847 | 5010.00 | 0.6 | -0.5 |
| 75 | 1110 | 24620 | 763 | 808.4 | 19.44 | - | - | 841.5 | 892 | 2189.8 | 837 | 917 | 5010.00 | 0.5 | 2.8 |
| 75 | 1110 | 24620 | 769 | 808.0 | 19.31 | - | - | 841.6 | 864 | 2866.0 | 833 | 876 | 5010.00 | 1.0 | 1.4 |
| 75 | 1110 | 24620 | 775 | 804.0 | 19.32 | - | - | 837.8 | 868 | 5010.1 | 829 | 868 | 5010.00 | 1.0 | 0.0 |
| 75 | 1110 | 43085 | 781 | 817.0 | 27.33 | - | - | 873.5 | - | 2357.1 | 870 | - | 5010.00 | 0.4 | 0.0 |
| 75 | 1110 | 43085 | 787 | 804.0 | 26.74 | - | - | 857.0 | - | 3106.8 | 859 | - | 5010.00 | -0.2 | 0.0 |
| 75 | 1110 | 43085 | 793 | 827.3 | 26.9 | - | - | 885.9 | 1194 | 3443.8 | 884 | - | 5010.00 | 0.2 | 100.0 |
| 75 | 1110 | 43085 | 799 | 799.0 | - | - | - | 856.70 | - | 3448.3 | 861 | - | 5010.00 | -0.5 | 0.0 |
| 75 | 1110 | 43085 | 805 | 806.0 | 27.14 | - | - | 859.1 | - | 3146.2 | 851 | - | 5010.00 | 0.9 | 0.0 |
| 100 | 990 | 4896 | 811 | 1118.5 | 11.59 | - | - | 1119.0 | 1119 | 43.7 | 1116 | 1121 | 5010.00 | 0.3 | 0.2 |
| 100 | 990 | 4896 | 817 | 1134.7 | 11.43 | - | - | 1137.0 | 1137 | 11.8 | 1137 | 1137 | 1091.57 | 0.0 | 0.0 |
| 100 | 990 | 4896 | 823 | 1112.5 | 11.71 | - | - | 1113.0 | 1113 | 71.8 | 1113 | 1113 | 493.89 | 0.0 | 0.0 |
| 100 | 990 | 4896 | 829 | 1109.6 | 11.48 | - | - | 1110.0 | 1110 | 48.6 | 1110 | 1110 | 942.31 | 0.0 | 0.0 |
| 100 | 990 | 4896 | 835 | 1088.7 | 11.99 | - | - | 1090.0 | 1090 | 35.8 | 1090 | 1090 | 357.10 | 0.0 | 0.0 |
| 100 | 990 | 19583 | 841 | 1175.6 | 15.37 | - | - | 1249.4 | - | 5010.0 | 1239 | 1570 | 5010.00 | 0.8 | -100.0 |
| 100 | 990 | 19583 | 847 | 1142.0 | 14.98 | - | - | 1225.8 | 1491 | 5010.0 | 1205 | 1650 | 5010.00 | 1.7 | 10.7 |
| 100 | 990 | 19583 | 853 | 1140.7 | 14.8 | - | - | 1215.0 | 1510 | 5010.0 | 1203 | 1623 | 5010.00 | 1.0 | 7.5 |

Table 2. Cont.

| n | Instances | | | HDA [6] | | HDA+ [6] | | BC [7] | | | CP-SAT | | | Dev% | |
|----------|-----------|--------|------|---------|--------|----------|-----|--------|------|--------|--------|------|---------|------|------|
| | m | p | s | LB | Sec | UB | Sec | LB | UB | Sec | LB | UB | Sec | LB | UB |
| 100 | 990 | 19583 | 859 | 1178.1 | 14.82 | - | - | 1264.2 | 1441 | 5010.2 | 1248 | 1632 | 5010.00 | 1.3 | 13.3 |
| 100 | 990 | 19583 | 865 | 1176.4 | 15.36 | - | - | 1257.3 | 1560 | 5010.1 | 1244 | 1555 | 5010.00 | 1.1 | -0.3 |
| 100 | 990 | 34269 | 871 | 1126.4 | 22.2 | - | - | 1264.8 | - | 5010.0 | 1251 | - | 5010.00 | 1.1 | 0.0 |
| 100 | 990 | 34269 | 877 | 1148.9 | 24.59 | - | - | 1295.6 | - | 5010.0 | 1285 | - | 5010.00 | 0.8 | 0.0 |
| 100 | 990 | 34269 | 883 | 1179.9 | 21.17 | - | - | 1323.7 | - | 5010.0 | 1305 | - | 5010.00 | 1.4 | 0.0 |
| 100 | 990 | 34269 | 889 | 1146.7 | 20.3 | - | - | 1283.6 | - | 5010.0 | 1268 | - | 5010.00 | 1.2 | 0.0 |
| 100 | 990 | 34269 | 895 | 1155.0 | 20.42 | - | - | 1309.1 | - | 5010.0 | 1297 | - | 5010.00 | 0.9 | 0.0 |
| 100 | 1485 | 11019 | 901 | 1077.9 | 34.81 | - | - | 1079.0 | 1079 | 248.6 | 1077 | 1081 | 5010.00 | 0.2 | 0.2 |
| 100 | 1485 | 11019 | 907 | 1054.9 | 34.46 | - | - | 1056.0 | 1056 | 113.4 | 1056 | 1056 | 757.17 | 0.0 | 0.0 |
| 100 | 1485 | 11019 | 913 | 1059.0 | 34.29 | - | - | 1059.0 | 1059 | 47.9 | 1059 | 1059 | 503.94 | 0.0 | 0.0 |
| 100 | 1485 | 11019 | 919 | 1046.0 | 34.09 | - | - | 1046.0 | 1046 | 195.2 | 1046 | 1046 | 1206.51 | 0.0 | 0.0 |
| 100 | 1485 | 11019 | 925 | 1071.2 | 34.17 | - | - | 1072.0 | 1072 | 249.6 | 1071 | 1072 | 5010.00 | 0.1 | 0.0 |
| 100 | 1485 | 44075 | 931 | 1096.9 | 46.07 | - | - | 1144.0 | 1374 | 3018.3 | 1130 | 1444 | 5010.00 | 1.2 | 5.1 |
| 100 | 1485 | 44075 | 937 | 1091.0 | 45.6 | - | - | 1143.6 | 1291 | 2144.6 | 1134 | 1511 | 5010.00 | 0.8 | 17.0 |
| 100 | 1485 | 44075 | 943 | 1084.4 | 46.54 | - | - | 1137.6 | 1344 | 3075.6 | 1125 | 1390 | 5010.00 | 1.1 | 3.4 |
| 100 | 1485 | 44075 | 949 | 1089.4 | 46.01 | - | - | 1136.9 | 1286 | 3523.3 | 1131 | 1417 | 5010.00 | 0.5 | 10.2 |
| 100 | 1485 | 44075 | 955 | 1088.4 | 45.84 | - | - | 1134.6 | 1370 | 2954.2 | 1127 | 1524 | 5010.00 | 0.7 | 11.2 |
| 100 | 1485 | 77131 | 961 | 1083.5 | 60.68 | - | - | 1164.4 | - | 4773.8 | 1156 | - | 5010.00 | 0.7 | 0.0 |
| 100 | 1485 | 77131 | 967 | 1086.0 | 60.29 | - | - | 1168.2 | - | 5010.0 | 1160 | - | 5010.00 | 0.7 | 0.0 |
| 100 | 1485 | 77131 | 973 | 1094.0 | 60.69 | - | - | 1187.8 | - | 5010.0 | 1175 | - | 5010.00 | 1.1 | 0.0 |
| 100 | 1485 | 77131 | 979 | 1105.9 | 60.83 | - | - | 1183.7 | - | 5010.0 | 1175 | - | 5010.00 | 0.7 | 0.0 |
| 100 | 1485 | 77131 | 985 | 1090.9 | 60.63 | - | - | 1164.5 | - | 5010.0 | 1155 | - | 5010.00 | 0.8 | 0.0 |
| 100 | 1980 | 19593 | 991 | 1030.9 | 76.15 | - | - | 1031.0 | 1031 | 214.4 | 1031 | 1031 | 919.88 | 0.0 | 0.0 |
| 100 | 1980 | 19593 | 997 | 1034.9 | 76.02 | - | - | 1036.0 | 1036 | 42.8 | 1036 | 1036 | 1298.69 | 0.0 | 0.0 |
| 100 | 1980 | 19593 | 1003 | 1024.0 | 76.43 | - | - | 1024.0 | 1024 | 21.8 | 1024 | 1024 | 622.64 | 0.0 | 0.0 |
| 100 | 1980 | 19593 | 1009 | 1025.0 | 76.67 | - | - | 1025.0 | 1025 | 27.4 | 1025 | 1025 | 546.97 | 0.0 | 0.0 |
| 100 | 1980 | 19593 | 1015 | 1027.5 | 77.49 | - | - | 1028.0 | 1028 | 151.0 | 1028 | 1028 | 1964.48 | 0.0 | 0.0 |
| 100 | 1980 | 78369 | 1021 | 1060.5 | 106.68 | - | - | 1097.0 | 1234 | 5010.1 | 1091 | 1358 | 5010.00 | 0.5 | 10.0 |
| 100 | 1980 | 78369 | 1027 | 1034.0 | 106.82 | - | - | 1065.6 | 1187 | 5010.1 | 1064 | 1305 | 5010.00 | 0.2 | 9.9 |
| 100 | 1980 | 78369 | 1033 | 1050.5 | 104.96 | - | - | 1087.4 | 1213 | 5010.1 | 1076 | 1314 | 5010.00 | 1.0 | 8.3 |
| 100 | 1980 | 78369 | 1039 | 1048.1 | 103.83 | - | - | 1083.5 | 1221 | 5010.1 | 1076 | 1320 | 5010.00 | 0.7 | 8.1 |
| 100 | 1980 | 78369 | 1045 | 1048.9 | 103.99 | - | - | 1084.1 | 1245 | 5010.1 | 1079 | 1353 | 5010.00 | 0.5 | 8.7 |
| 100 | 1980 | 137145 | 1051 | 1041.8 | 129.47 | - | - | 1100.6 | - | 5010.1 | 1086 | - | 5010.00 | 1.3 | 0.0 |
| 100 | 1980 | 137145 | 1057 | 1066.5 | 129.26 | - | - | 1126.3 | - | 5010.1 | 1114 | - | 5010.00 | 1.1 | 0.0 |
| 100 | 1980 | 137145 | 1063 | 1052.4 | 130.38 | - | - | 1111.3 | - | 5010.1 | 1087 | - | 5010.00 | 2.2 | 0.0 |
| 100 | 1980 | 137145 | 1069 | 1062.9 | 130.01 | - | - | 1118.7 | - | 5010.1 | 1100 | - | 5010.00 | 1.7 | 0.0 |
| 100 | 1980 | 137145 | 1075 | 1055.5 | 130.01 | - | - | 1114.1 | - | 5010.1 | 1105 | - | 5010.00 | 0.8 | 0.0 |
| Averages | | | | | | | | | | | | | | 0.4 | 0.8 |

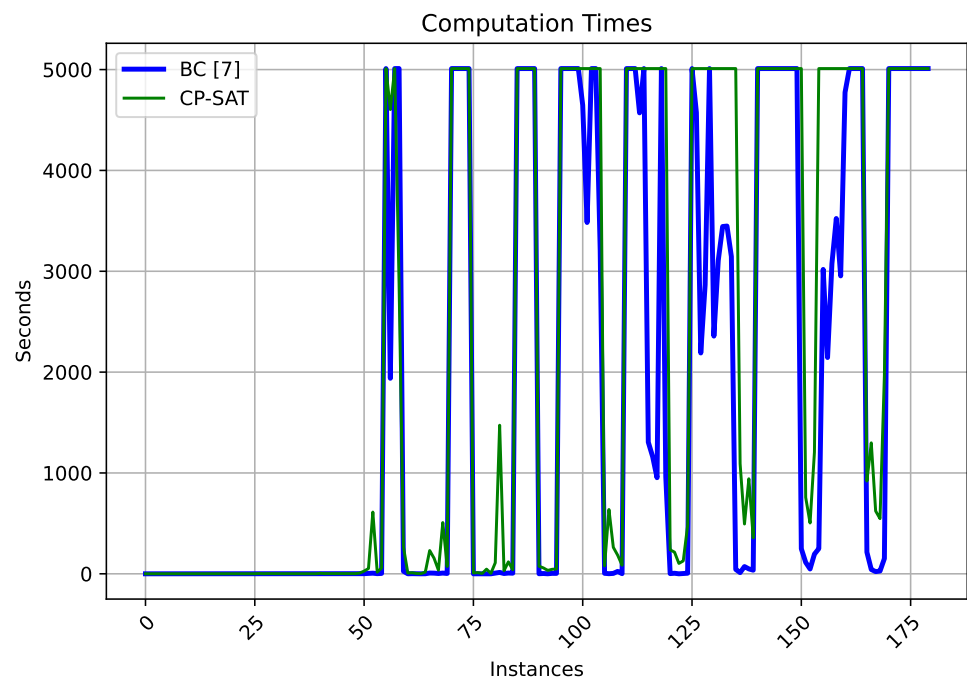


Figure 8. Computation times for two exact methods in the instances of the second benchmark set.

As a general conclusion, the proposed model closely matches or improves upon the best-known results available prior to this work. The proposed method also demonstrates consistent robustness across the entire dataset, although its convergence times appear slower than those of some of the methods previously proposed.

5. Conclusions

A compact mixed-integer linear formulation for the Minimum Spanning Tree Problem with Conflicting Edge Pairs was proposed for the first time in the context of this problem and solved via an open-source solver.

The results of a vast experimental campaign run on the benchmark instances adopted in the previous literature are reported. The results indicate that the proposed approach achieves computational results comparable with the best of those achieved by the substantially more complex methods that appeared in the prior literature, although, sometimes, the computation times are longer due to the mechanics of the solver. A total of six instances were closed for the first time, with nine best-known lower bounds and sixteen best-known upper bounds improved out of the two hundred thirty instances considered. In some of the instances, the new approach still performs slightly worse than some among the other methods it is compared against, but overall, it guarantees robust results, with improvements in instances attacked and solved in the research community for more than a decade.

Future research directions might be twofold. On the one hand, compact models, like the one we propose, might be employed on problems with conflicts similar to the one treated in this paper or employ soft constraints (that can be violated at the price of a penalty), which, by their nature, would pose new challenges. On the other hand, variants of the Minimum Spanning Tree, characterized by soft constraints, with penalties to be paid if violated, might be considered.

Author Contributions: Conceptualization, R.M. and D.H.S.; methodology, R.M.; software, R.M.; validation, R.M. and D.H.S.; formal analysis, R.M. and D.H.S.; investigation, R.M.; resources, R.M.; data curation, R.M.; writing—original draft preparation, R.M.; writing—review and editing, R.M. and D.H.S.; visualization, R.M. and D.H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Google Cloud Research Credits Program.

Data Availability Statement: The instances used for the experiments, originally introduced in [3,7], are available upon request to the corresponding author.

Acknowledgments: The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Prim, R.C. Shortest connection networks and some generalizations. *Bell Syst. Tech. J.* **1957**, *6*, 1389–1401.
2. Darmann, A.; Pfersch, U.; Schauer, J. and Woeginger, G. Paths, trees and matchings under disjunctive constraints. *Discret. Appl. Math.* **2011**, *16*, 1726–1735.
3. Zhang, R.; Kabadi, S.; Punnen, A. The minimum spanning tree problem with conflict constraints and its variations. *Discret. Optim.* **2011**, *2*, 191–205.
4. Samer, P.; Urrutia, S. A branch and cut algorithm for minimum spanning trees under conflict constraints. *Optim. Lett.* **2014**, *1*, 41–55.
5. Carrabs, F.; Cerrone, C.; Pentangelo, R. A multi-ethnic genetic approach for the minimum conflict weighted spanning tree problem. *Networks* **2019**, *2*, 134–147.
6. Carrabs, F.; Gaudio, M. A Lagrangian approach for the minimum spanning tree problem with conflicting edge pairs. *Networks* **2021**, *1*, 32–45.

7. Carrabs, F.; Cerulli, R.; Pentangelo, R.; Raiconi, A. Minimum spanning tree with conflicting edge pairs: A branch-and-cut approach. In *Annals of Operations Research*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 65–78.
8. Öncan, T.; Zhang, R.; Punnen, A.P. The minimum cost perfect matching problem with conflict pair constraints. *Comput. Oper. Res.* **2013**, *40*, 920–930.
9. Öncan, T.; Altinel, I.K. Iterated exact and heuristic algorithms for the minimum cost bipartite perfect matching problem with conflict constraints. In Proceedings of the 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, 10–13 December 2017; pp. 1032–1036.
10. Öncan, T.; Altinel, I.K. A Branch-and-Bound Algorithm for the Minimum Cost Bipartite Perfect Matching Problem with Conflict Pair Constraints. *Electron. Notes Discret. Math.* **2018**, *64*, 5–14.
11. Öncan, T.; Şuvak, Z.; Akyüz, M.H.; Altinel, I.K. Assignment problem with conflicts. *Comput. Oper. Res.* **2019**, *111*, 214–229.
12. Yamada, T.; Kataoka, S.; Watanabe, K. Heuristic and exact algorithms for the disjunctively constrained knapsack problem. *Inform. J. Comput.* **2002**, *43*, 2864–2870.
13. Hifi, M.; Michrafy, M. Reduction strategies and exact algorithms for the disjunctively constrained knapsack problem. *Comput. Oper. Res.* **2007**, *34*, 2657–2673.
14. Hifi, M.; Michrafy, M. A reactive local search-based algorithm for the disjunctively constrained knapsack problem. *J. Oper. Res. Soc.* **2006**, *57*, 718–726.
15. Akeb, H.; Hifi, M.; Ould Ahmed Mounir, M.E. Local branching-based algorithms for the disjunctively constrained knapsack problem. *Comput. Ind. Eng.* **2011**, *60*, 811–820.
16. Hifi, M.; Omani, N. An algorithm for the disjunctively constrained knapsack problem. *Int. J. Oper. Res.* **2012**, *13*, 22–43.
17. Hifi, M. An iterative rounding search-based algorithm for the disjunctively constrained knapsack problem. *Eng. Optim.* **2014**, *46*, 1109–1122.
18. Bettinelli, A.; Cacchiani, V.; Malaguti, E. A Branch-and-Bound Algorithm for the Knapsack Problem with Conflict Graph. *INFORMS Journal on Computing* **2017**, *29*, 457–473.
19. Coniglio, S.; Furini, F.; San Segundo, P. A new combinatorial branch-and-bound algorithm for the Knapsack Problem with Conflicts. *Eur. J. Oper. Res.* **2021**, *289*, 435–455.
20. Carrabs, F.; Cerulli, R.; Mansini, R.; Moreschini, L.; Serra, D. Solving the Set Covering Problem with Conflicts on Sets: A new parallel GRASP. *Comput. Oper. Res.* **2024**, *166*, 106620.
21. Jacob, A.; Majumdar, D.; Raman, V. Parameterized complexity of conflict-free set cover. *Lect. Notes Comput. Sci.* **2019**, *11532*, 191–202.
22. Saffari, S.; Fathi, Y. Set covering problem with conflict constraints. *Comput. Oper. Res.* **2022**, *143*, 105763.
23. Banik, A.; Panolan, F.; Raman, V.; Sahlot, V.; Saurabh, S. Parameterized Complexity of Geometric Covering Problems Having Conflicts. *Algorithmica* **2020**, *82*, 1–19.
24. Gabow, H.; Maheshwari, S.; Osterweil, L. On Two Problems in the Generation of Program Test Paths. *IEEE Trans. Softw. Eng.* **1976**, *SE-2*, 227–231.
25. Krause, K.W.; Goodwin, M.A.; Smith, R.W. *Optimal Software Test Planning Through Automated Network Analysis*; TRW Systems Group: California, USA, 1973; pp. 18–22.
26. Srimani, P.K.; Sinha, B.P. Impossible pair constrained test path generation in a program. *Inf. Sci.* **1982**, *28*, 87–103.
27. Blanco, M.; Borndörfer, R.; Brückner, M.; Hoàng, N.D.; Schlechte, T. On the Path Avoiding Forbidden Pairs Polytope. *Electron. Notes Discret. Math.* **2015**, *50*, 343–348. <https://doi.org/10.1016/j.endm.2015.07.057>.
28. Ferone, D.; Festa, P.; Salani, M., Branch and Bound and Dynamic Programming Approaches for the Path Avoiding Forbidden Pairs Problem. In *Optimization and Decision Science: ODS, Virtual Conference, November 19, 2020*; Cerulli, R., Dell’Amico, M., Guerriero, F., Pacciarelli, D., Sforza, A., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 227–235.
29. Cerulli, R.; Guerriero, G.; Scalzo, E.; Sorgente, C. Shortest paths with exclusive-disjunction arc pairs conflicts. *Comput. Oper. Res.* **2023**, *152*, 106158.
30. Pferschy, U.; Schauer, J. The maximum flow problem with disjunctive constraints. *J. Comb. Optim.* **2013**, *26*, 109–119.
31. Şuvak, Z.; Altinel, I.K.; Aras, N. Exact solution algorithms for the maximum flow problem with additional conflict constraints. *Eur. J. Oper. Res.* **2020**, *287*, 410–437.
32. Carrabs, F.; Cerulli, R.; Mansini, R.; Serra, D.; Sorgente, C. Hybridizing Carousel Greedy and Kernel Search: A new approach for the maximum flow problem with conflict constraints. *Eur. J. Oper. Res.* **2025**, *324*, 414–435.
33. Cerrone, C.; Cerulli, R.; Golden, B. Carousel greedy: A generalized greedy algorithm with applications in optimization. *Comput. Oper. Res.* **2017**, *85*, 97–112.
34. Angelelli, E.; Mansini, R.; Speranza, M.G. Kernel search: A new heuristic framework for portfolio selection. *Comput. Optim. Appl.* **2012**, *51*, 345–361.
35. Klein, A.; Haugland, D.; Bauer, J.; Mommer, M. An integer programming model for branching cable layouts in offshore wind farms. In *Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 27–36.

36. Darmann, A.; Pferschy, U.; Schauer, J. Determining a minimum spanning tree with disjunctive constraints. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 414–423.
37. Magnanti, T.L.; Wolsey, L., Optimal trees. In *Handbook in Operations Research and Management Science*; Ball, M.O., Magnanti, T.L., Monma, C.L., Nemhauser, G.L., Eds.; North Holland Publishing: Amsterdam, The Netherlands, 1995; pp. 503–615.
38. Abdelmaguid, T.F. An Efficient Mixed Integer Linear Programming Model for the Minimum Spanning Tree Problem. *Mathematics* **2018**, *6*, 183.
39. de Almeida, A.M.; Martins, P.; de Souza, M.C. Min-degree constrained minimum spanning tree problem: complexity, properties, and formulations. *Int. Trans. Oper. Res.* **2012**, *19*, 323–352.
40. Gouveia, L.; Magnanti, T.L. Network flow models for designing diameter-constrained minimum-spanning and Steiner trees. *Networks* **2003**, *41*, 159–173.
41. Perron, L.; Didier, F. Google OR-Tools-CP-SAT, 2025. Available online: https://developers.google.com/optimization/cp/cp_solver/ (accessed on 9 July 2025).
42. Montemanni, R.; Dell’Amico, M. Solving the parallel drone scheduling traveling salesman problem via constraint programming. *Algorithms* **2023**, *16*, 40.
43. Montemanni, R.; Dell’Amico, M.; Corsini, A. Parallel drone scheduling vehicle routing problems with collective drones. *Comput. Oper. Res.* **2024**, *163*, 106514.
44. Montemanni, R.; Smith, D.H. On Solving the Maximum Flow Problem with Conflict Constraints. In Proceedings of the 2025 Industrial Electronics and Applications Conference, Sabah, Malaysia, 8–9 September 2025.
45. Montemanni, R. Solving a Home Healthcare Routing and Scheduling Problem with Real-World Features. In *Machine Learning and Soft Computing, Proceedings of International Conference on Machine Learning and Soft Computing, Tokyo, Japan, 24–26 January 2025*; Springer: Singapore, 2025; pp. 111–119.
46. Stuckey, P.J. Lazy Clause Generation: Combining the Power of SAT and CP (and MIP?) Solving. In Proceedings of the International Conference on Integration of Artificial Intelligence and Operations Research Techniques in Constraint Programming (CPAIOR), Bologna, Italy, 14–18 June 2010; pp. 5–9.
47. GEne Network Expansion. CPU Performance, 2025. Available online: https://gene.disi.unitn.it/test/cpu_list.php (accessed on 9 July 2025).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.