Université de Montréal

Lagrangian-informed mixed integer programming reformulations

par
Paul Khuong

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures et postdoctorales
en vue de l'obtention du grade de Philosophiæ Doctor (Ph.D.)
en informatique

décembre, 2013

RÉSUMÉ

La programmation linéaire en nombres entiers est une approche robuste qui permet de résoudre rapidement de grandes instances de problèmes d'optimisation discrète. Toutefois, les problèmes gagnent constamment en complexité et imposent parfois de fortes limites sur le temps de calcul. Il devient alors nécessaire de développer des méthodes spécialisées afin de résoudre approximativement ces problèmes, tout en calculant des bornes sur leurs valeurs optimales afin de prouver la qualité des solutions obtenues.

Nous proposons d'explorer une approche de reformulation en nombres entiers guidée par la relaxation lagrangienne. Après l'identification d'une forte relaxation lagrangienne, un processus systématique permet d'obtenir une seconde formulation en nombres entiers. Cette reformulation, plus compacte que celle de Dantzig et Wolfe, comporte exactement les mêmes solutions entières que la formulation initiale, mais en améliore la borne linéaire : elle devient égale à la borne lagrangienne.

L'approche de reformulation permet d'unifier et de généraliser des formulations et des méthodes de borne connues. De plus, elle offre une manière simple d'obtenir des reformulations de moins grandes tailles en contrepartie de bornes plus faibles.

Ces reformulations demeurent de grandes tailles. C'est pourquoi nous décrivons aussi des méthodes spécialisées pour en résoudre les relaxations linéaires.

Finalement, nous appliquons l'approche de reformulation à deux problèmes de localisation. Cela nous mène à de nouvelles formulations pour ces problèmes ; certaines sont de très grandes tailles, mais nos méthodes de résolution spécialisées les rendent pratiques.

Mots clés : recherche opérationnelle, optimisation discrète, relaxation lagrangienne, programmation en nombres entiers, problèmes de localisation.

ABSTRACT

Integer linear programming is a robust and efficient approach to solve large-scale instances of combinatorial problems. However, problems constantly gain in complexity and sometimes impose strong constraints on computation times. We must then develop specialised methods to compute heuristic primal solutions to the problem and derive lower bounds on the optimal value, and thus prove the quality of our primal solutions.

We propose to guide a reformulation approach for mixed integer programs with Lagrangian relaxations. After the identification of a strong relaxation, a mechanical process leads to a second integer formulation. This reformulation is equivalent to the initial one, but its linear relaxation is equivalent to the strong Lagrangian dual.

We will show that the reformulation approach unifies and generalises prior formulations and lower bounding approaches, and that it exposes a simple mechanism to reduce the size of reformulations in return for weaker bounds.

Nevertheless, our reformulations are large. We address this issue by solving their linear relaxations with specialised methods.

Finally, we apply the reformulation approach to two location problems. This yields novel formulations for both problems; some are very large but, thanks to the aforementioned specialised methods, still practical.

Keywords: operations research, discrete optimisation, Lagrangian relaxation, mixed integer programming, location problems.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

“I am proposing [shadow prices], a new form of indirect indicator for labour value which would allow us to calculate, easily and straightforwardly, plans that are optimal all around.

Is this heresy?

Leonid V. Kantorovich in Red Plenty, *by Francis Spufford*”

REMERCIEMENTS

Merci à mon directeur Bernard Gendron : tu m'as invité et guidé dans le monde de la recherche dès ma première session de baccalauréat ; tu as su me proposer un style de direction très libre ; tu m'as même permis d'occuper un mandat d'un an et demi à la FAÉCUM. Ces preuves de confiance furent on ne peut plus gratifiantes et rassurantes.

Merci aussi à mes parents et à ma soeur. Vous m'avez appris à affronter les défis et encouragé à poursuivre mes passions alors que la médecine dentaire n'en était pas. Aussi, un *shout-out* tout particulier à mon père pour l'été de mes douze ans, période charnière pendant laquelle tu m'as confronté non seulement aux statistiques non-paramétriques, mais aussi à la programmation linéaire, au simplexe et à l'inimitable LINDO.

Les stages à Centrale-Lille avec Frédéric Semet et à Sophia-Antipolis avec Jean-Charles Régin furent aussi denses en apprentissage. Merci pour votre accueil et direction.

Sur une note plus institutionnelle, je suis reconnaissant au CRSNG, au FRQNT et au CIRRELT pour leur soutien financier : ces bourses et subventions de voyage m'ont permis de me concentrer sur mon doctorat et de découvrir les travaux qui se font ailleurs.

Véro, Maude, Delphine, Steph et Gen, gardiennes de ma santé mentale ♡.

Ross, Anne-Cath, Éloi, Max, Xavier, Nic, Mathieu, Stef et Jeahn-Gab, sans oublier Alex, Evelyne et Kate : merci de vous être adaptés à l'officier souvent présent mais pas toujours disponible, et d'avoir supporté et soutenu mes moments un peu plus difficiles.

Marius, Heinz, Catherine, Reb, Frank, David, Erick et Maxime : vous me rappelez qu'il est possible de demeurer humain dans le monde informatique.

Joonas, Barkley and Scott: I owe you my coding and design abilities and the survival of countless laptops and writing implements. I would never have been able to achieve this without your decade of constant guidance and rants about the real world, as well as the outlet you provided to vent about CPLEX and academic code.

To the Steel Bank Common Lisp team, particularly Christophe, Nikodemus and Juho: thank you for showing me how to work with a team on a medium-scale project, and for enabling me to procrastinate usefully. Here's to the next batch of grad students!

Ani... Merci. Je ne serais pas aujourd'hui sans toi.

INTRODUCTION

Lagrangian relaxation is a general approach to derive lower bounds for constrained optimisation problems; this thesis is concerned with its application to mixed integer linear programs. This approach is particularly attractive because it translates practical insights on the problem at hand – e.g., which constraints are less likely to be violated – into strong and theoretically sound lower bounding methods. However, it is difficult to materialise that strength in practice: the Lagrangian dual is an implicitly defined piecewise-linear function, and nondifferentiable optimisation is difficult in general. For example, Chapter 5 applies Lagrangian decomposition to an industrial location problem, and both a subgradient algorithm and a bundle algorithm make little progress after hundreds of iterations. Lagrangian relaxation is most attractive for large problems, but large-scale Lagrangian dual problems are difficult for general nondifferentiable optimisation methods.

This difficulty is partly caused by the fact that nondifferentiable convex optimisation methods imposes few assumptions on the objective functions: classical approaches to solve Lagrangian master problems are equally applicable to linear, mixed integer, or non-linear relaxed subproblems.

We propose to exploit the structure of mixed integer programs with *Lagrangian-informed* reformulation, a compact alternative to Dantzig-Wolfe reformulation. Specifically, we assume the existence of a Lagrangian relaxation scheme for the program, and that the Lagrangian subproblem may be described with an explicit mixed integer program. The resulting mixed integer reformulations are equivalent to the initial integer programs, but their linear relaxations are stronger: they are equivalent to the Lagrangian duals that gave rise to the reformulations.

We also describe methods to solve these linear relaxations (equivalently, Lagrangian duals) by leveraging scalable linear optimisation methods. Through case studies motivated by an industrial application, we will show that two classical nondifferentiable optimisation methods fail to scale to large instances of location problems, while our specialised methods are competitive with a state-of-the-art branch-and-cut algorithm.

Chapter 1 introduces Lagrangian-informed reformulation and shows how it unifies and generalises prior lower bounding approaches. The reformulation relies on the fact that an explicit mixed integer program describes the relaxed subproblem; once the discrete variables are fixed to feasible values, the restricted subproblem is a linear program. At a high level, we propose to enumerate all feasible assignments for the set of discrete variables and optimise over *multiple choice* formulation for the union of all restricted (linear) subproblems. When every variable is discrete, restricted subproblems are trivial, and the reformulation is equivalent to Dantzig-Wolfe reformulation. However, when some variables are continuous, our reformulation is more compact: each restricted subproblem captures several feasible solutions. We thus expose an additional dimension to balance bound strength and computational efficiency. All Lagrangian relaxation methods offer the possibility to relax more constraints and simplify the Lagrangian subproblem, at the expense of a weaker lower bound. Our reformulation method also exploits the presence of continuous variables in the subproblem; we can thus simplify the Lagrangian master problem (and the subproblems) by letting more decision variables take fractional values.

Despite their relative compactness, Lagrangian-informed reformulations may be too large to solve directly, as explicit linear programs. Chapter 2 describes two specialised methods to solve these linear relaxations. The first exploits the relationship between linear dual multipliers and Lagrange multipliers to warm start the solution of a large linear relaxation. The second applies Structured Dantzig-Wolfe decomposition [31] to reformulations that are too large for explicit formulations. We thus show how to solve any Lagrangian relaxation for a mixed integer program with this extension of bundle methods.

Chapters 3 to 5 apply the reformulation approach to derive novel formulations for two location problems. Chapters 3 and 4 are concerned with a variant of the Two-level Uncapacitated Facility Location Problem. Chapter 3 improves prior formulations with a Lagrangian-informed reformulation that we solve as an explicit mixed integer program. Chapter 4 instead applies the Structured Dantzig-Wolfe method from Chapter 2 to quickly compute lower bounds for a larger and stronger reformulation; the result is a Lagrangian heuristic method that is well suited to our practical instances. Finally, Chapter 5 builds on Chapter 3 to obtain a strong formulation for an industrial location problem. We tighten that formulation further by guiding our reformulation process with a Lagrangian decomposition scheme. The result is a large mixed integer program that we can nevertheless formulate explicitly; we approximately solves its linear relaxation with a specialised method that reaches stronger lower bounds than classical methods for an equivalent Lagrangian master problem, given the same time limit.

We hope that our work will improve the practical applicability of Lagrangian relaxation methods for mixed integer programs. The reformulation approach itself already offers an additional parameter to trade theoretical lower bound strength for runtime efficiency. We also describe methods that exploit high-performance linear optimisation software to optimise the large linear relaxations of our reformulations. In our experiments, these methods proved more efficient than generic nondifferentiable methods for equivalent Lagrangian dual functions. Thus, in addition to unifying prior linear formulations and relating them to intuitive Lagrangian relaxation schemes, we extend the design space of Lagrangian relaxation methods. In theory, the extension offers little advantage over prior results. In practice, it opens a new range of engineering trade-offs: it is usually preferable to reliably solve a weaker relaxation than to stall or cycle on a stronger one.

CHAPTER 1

LAGRANGIAN-INFORMED REFORMULATIONS

We propose to compute lower bounds for mixed integer programs with Lagrangian-informed reformulation, an approach based on the *multiple choice* model for the union of polytopes [4, 50, 67]. We define the reformulation on mixed integer programs of the form

$$\text{(P)} \qquad \min_{x \in \mathbb{R}^n, y} cx + fy$$

subject to

$$Ax + By \leq d, \tag{1.1}$$

$$Ex + Fy \leq g, \tag{1.2}$$

$$x \geq 0, \tag{1.3}$$

$$y \in Y, \qquad Y = \{y \in \mathbb{Z}^m \mid 0 \leq y \leq u\}, \tag{1.4}$$

where c, f, d, g and u are rational vectors of sizes m, m, p, q, respectively, and A, B, E, F are rational matrices of sizes $p \times n$, $p \times m$, $q \times n$, $q \times m$, respectively.

The reformulation approach only requires that Y be a finite set. We only assume that it is a bounded set of non-negative integers to simplify our analysis. We further assume that

$$T = \{(x, y) \in \mathbb{R}^n \times Y \mid Ex + Fy \leq g,\ x \geq 0\}$$

is non-empty and bounded.

The result of the Lagrangian-informed reformulation is a second mixed integer program with a stronger linear relaxation than (P), one equivalent to the Lagrangian dual with respect only to constraint (1.1).

Relaxing this constraint yields the Lagrangian subproblem

$$(\mathrm{S}_\lambda) \qquad \left[\min_{x,y} (\lambda A + c)x + (\lambda B + f)y\right] - \lambda d,$$

subject to constraints (1.2), (1.3) and (1.4), where $\lambda \geq 0$ is the vector of Lagrange multipliers associated with constraint (1.1). We compute the Lagrangian bound by maximising the Lagrangian dual

$$\max_{\lambda \geq 0} v(\mathrm{S}_\lambda),$$

where $v(\mathrm{S}_\lambda)$ is the value of the Lagrangian subproblem for a given vector λ. Computing that bound is equivalent to solving

$$\min_{x,y} cx + fy$$

over the intersection of

$$H = \mathrm{Conv}(\{(x,y) \in \mathbb{R}^n \times Y \mid Ex + Fy \leq g,\ x \geq 0\}) = \mathrm{Conv}(T),$$

the convex hull of the feasible set in the Lagrangian subproblem, and of

$$\{(x,y) \in \mathbb{R}^n \times \mathbb{R}^m \mid Ax + By \leq d\},$$

the subspace of $\mathbb{R}^n \times \mathbb{R}^m$ that satisfies constraint (1.1) [37]. Unless the formulation for (S_λ) possesses the integrality property, H is a strict subset of

$$\{(x,y) \in \mathbb{R}^n \times \mathbb{R}^m \mid Ex + Fy \leq g,\ x \geq 0,\ 0 \leq y \leq u\}$$

and the Lagrangian bound dominates the linear relaxation bound.

Our reformulation obtains a compact formulation for (S_λ) by enumerating every $y \in Y$. After fixing y to $s \in Y$ (in a fashion similar to Benders decomposition [10]), constraints

$$\begin{aligned} Ex &\leq g - Fs, \\ x &\geq 0 \end{aligned}$$

define the feasible set of the restricted Lagrangian subproblem. Each restricted subproblem captures a subspace of the feasible set that comprises multiple extreme points. Joined together, these subspaces span the feasible set of the Lagrangian subproblem.

We duplicate variables x in order to obtain one restricted subproblem for each $s \in Y$. We highlight the association between copies and subproblems by indexing the former as x^s: the exponent denotes the restricted subproblem to which each copy corresponds.

The multiple choice model [4, 50, 67] shows how to represent the union of restricted subspaces as a set of mixed integer linear constraints:

$$\begin{aligned} Ex^s &\leq \theta^s (g - Fs), && \forall s \in Y, \\ \sum_{s \in Y} \theta^s &= 1, \\ x^s &\geq 0, && \forall s \in Y. \\ x &= \sum_{s \in Y} x^s, \\ y &= \sum_{s \in Y} s\theta^s, \\ \theta^s &\in \{0, 1\}, && \forall s \in Y, \end{aligned}$$

where $s \in Y$ also serve as indices for the binary variables θ^s:

$$\theta^s = \begin{cases} 1 & \text{if } y = s \\ 0 & \text{otherwise} \end{cases} \quad \forall s \in Y.$$

This set has the same integer solutions as the feasible set in the Lagrangian subproblem; adding constraint (1.1) to it yields (P^+), the Lagrangian-informed reformulation of (P). We wish to solve the linear relaxation of (P^+) because of its strength: we prove in

Section 1.2 that it is equivalent to the Lagrangian dual of (P) with respect to (1.1).

$$(\mathrm{P}^+) \qquad \min_{x,y} cx + fy$$

subject to

$$Ax + By \leq d, \tag{1.5}$$

$$Ex^s \leq \theta^s(g - Fs), \qquad \forall s \in Y, \tag{1.6}$$

$$\sum_{s \in Y} \theta^s = 1, \tag{1.7}$$

$$x^s \geq 0, \qquad \forall s \in Y, \tag{1.8}$$

$$x = \sum_{s \in Y} x^s, \tag{1.9}$$

$$y = \sum_{s \in Y} s\theta^s, \tag{1.10}$$

$$\theta^s \in \{0,1\}, \qquad \forall s \in Y. \tag{1.11}$$

At first sight, the construction appears purely theoretical: Y can be large. Chapter 2 describes specialised methods to solve the linear relaxation of Lagrangian-informed reformulations. In Chapters 4 and 5, we apply these methods to novel reformulations and compare them with classical algorithms; the methods also yield primal information (*e.g.*, fractional solutions) that we exploit in heuristic algorithms.

The remainder of this chapter explores the theory of Lagrangian-informed reformulation. Section 1.1 gives a step-by-step example of the approach. Section 1.2 proves that the linear relaxation of every Lagrangian-informed reformulation is equivalent to the Lagrangian dual with respect to constraint (1.1). Section 1.3 shows how to obtain more compact reformulations at the expense of bound strength. Section 1.4 compares and relates the Lagrangian-informed reformulation approach to prior work. Section 1.5 offers hints to guide the implementation of Lagrangian-informed reformulations. Finally, Section 1.6 summarises this chapter.

1.1 Example of Lagrangian-informed reformulation

The multicommodity capacitated network design problem (MCND) is a classic example to illustrate Lagrangian relaxation (our presentation follows [31]). In this section, we apply Lagrangian-informed reformulation to a well known relaxation scheme for this problem.

We define the MCND on a directed graph $G = (V, A)$, where V is the set of vertices in the graph and A is the set of arcs between these vertices. We must allocate enough capacity for all origin–destination pairs: for each commodity $h \in H$, d^h units of flow depart from the origin vertex s_h and must arrive in the destination vertex t_h. Constants b_i^h represent this deficit or surplus:

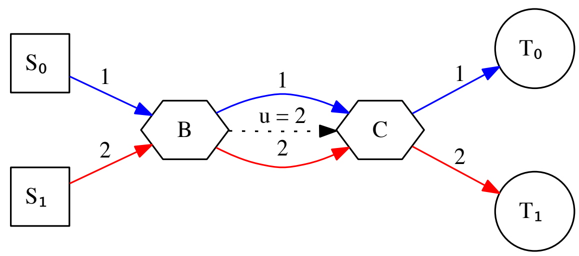
$$b_i^h = \begin{cases} -1, & \text{if } i = s_h, \\ 1, & \text{if } i = t_h, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i \in V, \forall h \in H.$$

Commodities also affect the cost of transitting through each arc $(i, j) \in A$: routing one unit of commodity $h \in H$ through arc (i, j) costs c_{ij}^h.

The other constraint is that there must be enough capacity on each arc. We can install an arbitrary number of facilities on each arc (i, j) to satisfy this constraint: each facility increases the capacity of the arc by u_{ij}, at cost f_{ij}.

The problem consists of determining the volume of each commodity to send through each arc and the number of facilities to install on each arc in order to satisfy all demand and capacity constraints at minimum cost. For example, the optimal integer solution to the instance in Fig. 1.1 installs two facilities on B→C.

Some versions of the MCND add the constraint that there must be exactly one path for each origin–destination pair, *i.e.*, that the flow for each commodity must not be split between multiple paths. We are currently interested in the version without this constraint,



One unit of flow must travel from S_0 to T_0, and two from S_1 to T_1; each facility installed on arc B→C increases its capacity by two.

Figure 1.1: A toy instance of the MCND.

the MCND with splittable flows; we will consider the MCND with unsplittable flows in Section 1.3.

We define (MC), a formulation for the MCND with splittable flows that comprises two classes of decision variables: flow variables w_{ij}^h determine the volume of commodity h (as a fraction of d^h) passing through each arc (i, j), and design variables y_{ij} represent the number of facilities installed on each arc.

(MC) $$\min_{w,y} \sum_{h\in H} \sum_{(i,j)\in A} d^h c_{ij}^h w_{ij}^h + \sum_{(i,j)\in A} f_{ij} y_{ij}$$

subject to

$$\sum_{(j,i)\in A} w_{ji}^h - \sum_{(i,j)\in A} w_{ij}^h = b_i^h, \qquad \forall i \in V, \forall h \in H, \tag{1.12}$$

$$\sum_{h\in H} d^h w_{ij}^h \leq u_{ij} y_{ij}, \qquad \forall (i,j) \in A, \tag{1.13}$$

$$0 \leq w_{ij}^h \leq 1, \qquad \forall (i,j) \in A, \forall h \in H, \tag{1.14}$$

$$0 \leq y_{ij} \leq T_{ij} \text{ and integer}, \qquad \forall (i,j) \in A,$$

where

$$T_{ij} = \left\lceil \sum_{h\in H} \frac{d^h}{u_{ij}} \right\rceil$$

is an upper bound on the number of facilities installed on arc (i,j).

We can adapt (MC) to the MCND with unsplittable flows by replacing constraints (1.14) with

$$w_{ij}^h \in \{0,1\}, \quad \forall (i,j) \in A, \forall h \in H.$$

Regardless of splittability, the linear relaxation of formulation (MC) (*i.e.*, relaxing the integrality constraint on y) is weak: it linearises the cost of capacity to f_{ij}/u_{ij} per unit of flow. On the example in Fig. 1.1, that relaxation would install $y_{\mathrm{BC}} = 3/2$ facilities to handle 3 units of flow.

We compute a stronger bound for the splittable case with the Lagrangian relaxation of flow conservation constraints (1.12): the Lagrangian subproblem decomposes into a set of independent integer programs (one per $(i,j) \in A$) with one constraint (1.13) and one integer variable (y_{ij}). For each arc (i, j), the set mixed integer constraints

$$
\begin{aligned}
&\sum_{h \in H} d^h w_{ij}^h \leq u_{ij} y_{ij}, && \\
&0 \leq w_{ij}^h \leq 1, && \forall h \in H, \\
&0 \leq y_{ij} \leq T_{ij} \text{ and integer} &&
\end{aligned}
$$

defines the feasible set of the decomposed Lagrangian subproblem.

Lagrangian-informed reformulation enumerates all $T_{ij}+1$ values for y_{ij} to find an equivalent, larger, mixed integer program. Fixing $y_{ij} = s$ leaves a linear knapsack set:

$$
\begin{aligned}
&\sum_{h \in H} d^h w_{ij}^h \leq u_{ij} s, && \\
&0 \leq w_{ij}^h \leq 1, && \forall h \in H.
\end{aligned}
$$

Let Y_{ij} be $\{0, 1, \ldots T_{ij}\}$, the domain of y_{ij}. Assembling the $T_{ij}+1$ restricted subproblems for $(i, j) \in A$ yields

$$
\begin{aligned}
&\sum_{h \in H} d^h w_{ij}^{hs} \leq u_{ij} s \theta_{ij}^s, && \forall s \in Y_{ij} \\
&0 \leq w_{ij}^{hs} \leq \theta_{ij}^s, && \forall h \in H, \forall s \in Y_{ij}, \\
&\sum_{s \in Y_{ij}} \theta_{ij}^s = 1, && \\
&w_{ij}^h = \sum_{s \in Y_{ij}} w_{ij}^{hs}, && \forall h \in H, \\
&y_{ij} = \sum_{s \in Y_{ij}} s \theta_{ij}^s, && \\
&\theta_{ij}^s \in \{0, 1\}, && \forall s \in Y_{ij}.
\end{aligned}
\tag{1.15}
$$

This mixed integer set possesses the integrality property, and Figure 1.2 shows its effect on the linear relaxation for arc B→C: it introduces copies of flow variables w_{BC}^h for each value y_{BC} could take. This, combined with constraints (1.15), is what improves

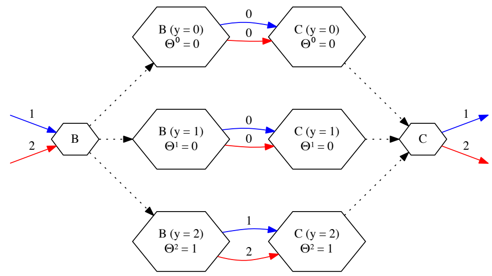


Figure 1.2: Effect of the reformulation on a single arc of the toy MCND instance.

the linear relaxation. The reformulation generalises the strong forcing constraints

$$w_{ij}^h \leq y_{ij}, \quad \forall (i,j) \in A, \forall h \in H,$$

constraints that are most effective when $y_{ij} \leq 1$. The forcing constraints in the reformulation extend to the case where y_{ij} is a general integer variable by mapping each possible value for y_{ij} to a dedicated binary decision variable.

We obtain the Lagrangian-informed reformulation by substituting this alternative mixed integer programming formulation for each subproblem—instead of constraints (1.13)—in (MC):

$$(\mathrm{MC}^+) \qquad \min_{w,y,\theta} \sum_{h\in H}\sum_{(i,j)\in A} d^h c_{ij}^h w_{ij}^h + \sum_{(i,j)\in A} f_{ij} y_{ij}$$

subject to

$$\begin{aligned}
\sum_{(j,i)\in A} w_{ji}^h - \sum_{(i,j)\in A} w_{ij}^h = b_i^h, & \qquad \forall i\in V, \forall h\in H,\\
\sum_{h\in H} d^h w_{ij}^{hs} \le u_{ij} s\theta_{ij}^s, & \qquad \forall (i,j)\in A, \forall s\in Y_{ij}, \qquad (1.16)\\
0 \le w_{ij}^{hs} \le \theta_{ij}^s, & \qquad \forall h\in H, \forall (i,j)\in A, \forall s\in Y_{ij},\\
\sum_{s\in Y_{ij}} \theta_{ij}^s = 1, & \qquad \forall (i,j)\in A,\\
w_{ij}^h = \sum_{s\in Y_{ij}} w_{ij}^{hs}, & \qquad \forall h\in H, \forall (i,j)\in A,\\
y_{ij} = \sum_{s\in Y_{ij}} s\theta_{ij}^s, & \qquad \forall (i,j)\in A,\\
\theta_{ij}^s \in \{0,1\}, & \qquad \forall (i,j)\in A, \forall s\in Y_{ij}.
\end{aligned}$$

The mixed integer formulation for each Lagrangian subproblem is nearly identical to the multiple choice formulation for piecewise linear cost functions [4, 18, 19, 50], and formulation (MC^+) to $\tilde{B}+$ of [30, 31]. We replaced constraints

$$u_{ij}(s-1)\theta_{ij}^s \le \sum_{h\in H} d^h w_{ij}^{hs} \le u_{ij} s\theta_{ij}^s, \quad \forall (i,j)\in A, \forall s\in Y_{ij}$$

with constraints (1.16), but the left-hand side is irrelevant when capacity modules incur a positive cost. This means we re-derived formulations that are known for their strong linear relaxations. For example, the linear relaxation of $\tilde{B}+$ (and of (MC^+)) yields the same bound as the Lagrangian dual of (MC) with respect to constraint (1.12) [31].

The value of Lagrangian-informed reformulation is that we only had to devise a Lagrangian relaxation scheme for (MC) in order to obtain (MC^+). The next section characterises the strength of our reformulation: the linear relaxation of a Lagrangian-informed reformulation is always equivalent to the Lagrangian dual for the relaxation that lead to it.

1.2 Why Lagrangian-informed reformulation works

The linear relaxation of a Lagrangian-informed reformulation enumerates subspaces of the convex hull of the subproblem that relaxes constraint (1.1) (*i.e.*, $Ax + By \leq d$). The resulting lower bound is thus always equal to maximising the Lagrangian dual with respect to constraint (1.1), which dominates the linear relaxation of (P). We now prove this equality.

Let ($\overline{\mathrm{P}}$) be the linear relaxation of (P)

$$(\overline{\mathrm{P}}) \qquad \min_{x,y} cx + fy$$

subject to

$$\begin{aligned} Ax + By &\leq d, \\ Ex + Fy &\leq g, \\ x &\geq 0, \\ 0 \leq y &\leq u, \end{aligned}$$

and ($\bar{\mathrm{S}}_\lambda$) the linear relaxation of (S_λ)

$$(\bar{\mathrm{S}}_\lambda) \qquad v(\mathrm{S}_\lambda) = \left[\min_{x,y} (\lambda A + c)x + (\lambda B + f)y\right] - \lambda d,$$

subject to

$$Ex + Fy \leq g,$$
$$x \geq 0,$$
$$0 \leq y \leq u.$$

Proposition 1. *The Lagrangian dual of* $(\overline{\mathrm{P}})$ *with respect to constraint* (1.1) *(i.e., with* $(\bar{\mathrm{S}}_\lambda)$ *as the Lagrangian subproblem) is equivalent to the linear programming dual of* $(\overline{\mathrm{P}})$. *[37]*

Proof. The Lagrangian dual of $(\overline{\mathrm{P}})$ with respect to constraint (1.1) is

$$\max_{\lambda \geq 0} v(\bar{\mathrm{S}}_\lambda).$$

The linear programming dual of $(\bar{\mathrm{S}}_\lambda)$ is

$$(\overline{\mathrm{DS}}_\lambda) \qquad \left[\max_{\alpha,\beta} -\alpha g - \beta u\right] - \lambda d$$

subject to

$$-\alpha E \leq c + \lambda A,$$
$$-\alpha F - \beta \leq f + \lambda B,$$
$$\alpha \geq 0,$$
$$\beta \geq 0.$$

Substituting this into the Lagrangian dual yields

$$\max_{\lambda,\alpha,\beta} -\lambda d - \alpha g - \beta u$$

subject to

$$\begin{aligned} -\lambda A - \alpha E &\leq c, \\ -\lambda B - \alpha F - \beta &\leq f, \\ \lambda &\geq 0, \\ \alpha &\geq 0, \\ \beta &\geq 0. \end{aligned}$$

The linear programming dual of $(\overline{\text{P}})$ is

$$(\overline{\text{D}}) \qquad \max_{\lambda,\alpha,\beta} -\lambda d - \alpha g - \beta u$$

subject to

$$\begin{aligned} -\lambda A - \alpha E &\leq c, \\ -\lambda B - \alpha F - \beta &\leq f, \\ \lambda &\geq 0, \\ \alpha &\geq 0, \\ \beta &\geq 0. \end{aligned}$$

and the Lagrangian dual of $(\overline{\text{P}})$ with respect to (1.1) is also the linear programming dual of $(\overline{\text{P}})$. □

Corollary 2. *The Lagrangian dual of* $(\overline{\text{P}})$ *with respect to* (1.1) *is maximised by letting* λ *be equal to its optimal value in* $(\overline{\text{D}})$*, i.e., to optimal multipliers for constraint* (1.1).

Lemma 3. *The Lagrangian subproblem* (S_λ) *can be solved with the linear program [37]*

$$(\mathrm{S}'_\lambda) \qquad v(S'_\lambda) = \left[\min_{x,y} (\lambda A + c)x + (\lambda B + f)y\right] - \lambda d,$$

subject to

$$(x,y) \in H = \mathrm{Conv}(\{(x,y) \in \mathbb{R}^n \times Y \mid Ex + Fy \le g,\ x \ge 0\}) = \mathrm{Conv}(T). \tag{1.17}$$

Proof. Since

$$T = \{(x,y) \in \mathbb{R}^n \times Y \mid Ex + Fy \le g,\ x \ge 0\}$$

is non-empty and bounded, we can express constraint (1.17) as the linear program

$$(x,y) = \sum_{t \in T} t\theta^t, \tag{1.18}$$

$$\sum_{t \in T} \theta^t = 1, \tag{1.19}$$

$$\theta^t \ge 0, \qquad \forall t \in T. \tag{1.20}$$

Thus, H is a polytope. Since the objective function of (S'_λ) is linear, there is an optimal solution to (S'_λ) that is an extreme point of H. Because every extreme point of H satisfies $(x,y) \in T$, an optimal solution to (S'_λ) is feasible for (S_λ), which implies that $v(\mathrm{S}'_\lambda) \ge v(\mathrm{S}_\lambda)$. Furthermore, $H \supseteq T$, and we also have that $v(\mathrm{S}'_\lambda) \le v(\mathrm{S}_\lambda)$. □

Proposition 4. *Maximising the Lagrangian dual*

$$\max_{\lambda \ge 0} v(\mathrm{S}_\lambda)$$

is equivalent to minimising this relaxation of (P)*: [37]*

$$(\mathrm{P}^*) \qquad \min_{x,y} cx + fy$$

subject to

$$Ax + By \leq d,$$
$$(x, y) \in H. \tag{1.21}$$

Proof. Proposition 1 states that the Lagrangian dual of a linear program ($\bar{\text{P}}$) is equivalent to the linear dual of ($\bar{\text{P}}$). We apply this result to the linear program (S'_λ), which is equivalent to (S_λ) by Lemma 3, to prove the proposition. □

Moreover, Lemma 3 shows how to solve (P^*) with a linear program: it suffices to replace constraint (1.21) with constraints (1.18)-(1.20). Alternatively, since H is a polytope, every point $(x, y) \in H$ can be expressed as a convex combination of the extreme points of H. Thus, if we denote by R the set of extreme points of H, we can solve (P^*) with the following linear program, which is known as the classical Dantzig-Wolfe reformulation [21, 22].

$$\text{(DW)} \qquad \min_{x,y} cx + fy$$

subject to

$$\begin{aligned} Ax + By &\leq d, \\ (x, y) &= \sum_{r \in R} r\theta^r, \\ \sum_{r \in R} \theta^r &= 1, \\ \theta^r &\geq 0, \qquad \forall r \in R. \end{aligned}$$

Proposition 5. *The linear relaxation of* (P^+)*, the Lagrangian-informed reformulation of* (P)*, dominates the linear relaxation of* (P)*: it is equivalent to* (P^*) *and thus to the Lagrangian dual of* (P) *with respect to constraint* (1.1).

Proof. Our reformulation exploits the fact that x is continuous to re-express

$$T = \{(x,y) \in \mathbb{R}^n \times Y \mid Ex + Fy \leq g,\ x \geq 0\}$$

as the union of simpler polytopes, one for each $s \in Y$:

$$T = \bigcup_{s \in Y} T(s),$$

where

$$T(s) = \{(x,y) \in \mathbb{R}^n \times \{s\} \mid Ex \leq g - Fs,\ x \geq 0\}.$$

The multiple choice model [4, 50, 67] for this union of polytopes is

$$Ex^s \leq \theta^s (g - Fs), \qquad \forall s \in Y, \tag{1.22}$$

$$\sum_{s \in Y} \theta^s = 1, \tag{1.23}$$

$$x^s \geq 0, \qquad \forall s \in Y, \tag{1.24}$$

$$x = \sum_{s \in Y} x^s, \tag{1.25}$$

$$y = \sum_{s \in Y} s\theta^s, \tag{1.26}$$

$$\theta^s \in \{0,1\}, \qquad \forall s \in Y. \tag{1.27}$$

Relaxing (1.27) into $\theta^s \in [0,1]$ yields a linear system that is equal to H, the convex hull of T [67]. Constraint (1.23) ensures that variables θ^s define weights for a convex combination of points in $T(s)$. The right-hand sides of constraints (1.22) are scaled by θ^s, so each x^s is a point in $T(s)$ scaled by θ^s; summing the scaled points in (1.25) yields a convex combination of points in $T(s)$. The same weights combine $s \in Y$ in constraint (1.26), so $(x,y) \in H$.

We obtain (P^+), the full Lagrangian-informed reformulation, by adding the relaxed constraint (1.1) to the multiple choice formulation for T. The linear relaxation of (P^+) relaxes the integrality constraints (1.11) into $\theta^s \in [0,1]$. As shown above, constraints

(1.6) to (1.11) then ensure that $(x,y) \in H$. Thus it is equivalent to (P*) and, therefore, it dominates the linear relaxation of (P). □

1.3 Trading bound strength for smaller formulations

Lagrangian-informed reformulations are more compact than Dantzig-Wolfe reformulations because we break the black box model: we equip the master problem with an explicit linear formulation for restricted subproblems. This allows us to adjust the size and strength of reformulations by relaxing more or fewer integrality constraints.

The previous section formalised the strength of Lagrangian bounds. Relaxing fewer constraints improves the lower bound, at the expense of a more complex subproblem. If we preserve every constraint, the relaxation is trivially equivalent to the original problem. If we instead relax all constraints, the Lagrangian bound is equal to the linear relaxation bound. Yet, this extreme choice fails to guarantee a subexponential sized Dantzig-Wolfe reformulation: the master problem always enumerates extreme points, even when we can describe its feasible set compactly.

In Lagrangian-informed reformulation, we assume that a linear formulation is available for restricted subproblems. That is why our reformulations shrink when more variables are continuous.

Letting a subset of integer variables y take fractional values always yields a valid lower bound. Once we have decided which constraints to relax, we can define intermediate reformulations by relaxing integrality for key variables. The linear relaxations of these intermediate reformulations are weaker than the initial Lagrangian relaxation, but stronger than the linear programming relaxation; more compact than the initial Lagrangian-informed relaxation, but larger than the original mixed integer formulation.

We will give one example of this trade-off by relaxing the unsplittable arc flow problem (UAFP) [3] into the splittable arc flow problem (SAFP) [3].

The UAFP appears in multicommodity capacitated network design problems when each commodity must follow a single path. We solve such MCND with unsplittable flows by restricting formulation (MC), presented in Section 1.1, with the constraint that variables w_{ij}^h are binary. Relaxing flow conservation constraints (1.12) in this pure integer formulation leaves independent subproblems (for each arc) of the form

$$\text{(UAFP)} \qquad \min_{x,y} \sum_{h \in H} c^h x^h + f y$$

subject to

$$\begin{aligned} \sum_{h \in H} d^h x^h &\leq uy, \\ x^h &\in \{0,1\}, & \forall h \in H, \\ y &\geq 0 \text{ and integer}, \end{aligned}$$

where $c^h \leq 0$, $f \geq 0$, $u > 0$, and H is a finite set of commodities.

There are no continuous variables, so both Lagrangian-informed reformulation and Dantzig-Wolfe decomposition represent the feasible set for UAFP as

$$\begin{aligned} (x,y) &= \sum_{r \in R} r\theta^r, \\ \sum_{r \in R} \theta^r &= 1, \\ \theta^r &\geq 0, & \forall r \in R, \end{aligned}$$

where R is the set of extreme points of

$$\text{Conv}(\{(x,y) \in \mathbb{B}^H \times \mathbb{N} \mid \sum_{h \in H} d^h x^h \leq uy\}).$$

Note that R grows exponentially with $|H|$.

The splittable arc flow problem (SAFP) [3] is a relaxation of the UAFP that dominates its linear relaxation and appears in the splittable MCND of Section 1.1: only y remains integer. We showed in Sections 1.1 and 1.2 that the Lagrangian-informed reformulation and the *multiple choice* model [4, 50, 67] express the SAFP as

$$\text{(SAFP)} \qquad \min_{x,y} \sum_{h \in H} c^h x^h + f y$$

subject to

$$\begin{aligned}
\sum_{h \in H} d^h x^{hs} &\leq u s \theta^s, && \forall s \in Y \\
0 \leq x^{hs} &\leq \theta^s, && \forall h \in H, \forall s \in Y, \\
\sum_{s \in Y} \theta^s &= 1, && \\
x^h &= \sum_{s \in Y} x^{hs}, && \forall h \in H, \\
y &= \sum_{s \in Y} s \theta^s, && \\
0 \leq \theta^s &\leq 1, && \forall s \in Y,
\end{aligned}$$

where

$$Y = 0, 1, \ldots, \left\lceil \sum_{h \in H} \frac{d^h}{u} \right\rceil$$

is the finite domain of y.

This reformulation includes one linear subproblem for each value in Y. However, Y is a smaller set than R in general (pseudopolynomial instead of exponential) and the reformulation is smaller than the one for UAFP. Solving the SAFP reformulation to optimality may yield stronger bounds more quickly than approximately maximising the Lagrangian dual for UAFP subproblems.

1.4 Relation to prior research

Lagrangian-informed reformulation builds on three areas: the multiple choice model for the union of polytopes [4, 50], non-differentiable optimisation methods for Lagrangian duals, and cross decomposition for mixed integer programming [49, 66].

Sections 1.1 and 1.2 explain how Lagrangian-informed reformulation decomposes the feasible set of the Lagrangian subproblem as the union of polytopes described by explicit linear constraints. It is then straightforward to represent this union with the multiple choice model.

This model brings us close to cutting-plane methods, one of the two general approaches for maximising Lagrangian duals. The other, subgradient methods [5, 9, 62, 64], shines when Lagrangian subproblems are simple. Solving subgradient master problems (*i.e.*, determining the next vector of Lagrange multipliers) is quick (*e.g.*, linear time) and sophisticated variants guarantee optimal convergence rates [40, 70]. However, the practical performance of these methods tends to be disappointing when instances grow larger and subproblems become more complex.

The family of cutting plane methods, including Dantzig-Wolfe decomposition [15, 21, 22, 54], decrease the number of subproblem evaluations by solving more demanding master problems, compared with subgradient methods. Later algorithms [13, 28, 41, 42, 48, 59, 68] focus on stabilising the master problem to obtain optimal convergence rates.

Lagrangian-informed reformulation continues a research line that aims to further reduce the number of subproblem evaluations (in practice if not in theory) by solving richer, more complex, master problems. Rich master problems violate the standard black box model of non-differentiable optimisation: additional information about the structure of the Lagrangian subproblem helps refine the master problem. Our reformulation also works in a translucent box model: we require access to linear programming formulations for restricted Lagrangian subproblems.

Previous work on rich Lagrangian master problems described schemes that disaggregate master problems when the subproblem separates into independent convex hulls [29, 52], tighten master problems with valid inequalities [33, 44, 65], or formulate parts of the convex hull explicitly [32]. These schemes are pragmatic when the master problem is a linear program: high-performance solvers optimise such problems quickly and reliably. We leverage these solvers with our own linear programs (*e.g.*, (P^+) and (MC^+)).

Our reformulation method is closer to the spirit of cross decomposition [49, 66] for mixed integer programs. Cross decomposition relies on the existence of both Benders [10] and Dantzig-Wolfe decompositions for the programs: an initial heuristic phase determines Lagrange multipliers by solving one Benders subproblem per iteration. The hope is that the Benders subproblem is a better model of the Lagrangian dual than early incarnations of the Dantzig-Wolfe master problem, and vice versa for the Lagrangian subproblem and the Benders master.

Lagrangian-informed reformulation as well exploits the quality of Benders subproblems as local approximations of the Lagrangian dual: reformulations consider all Benders subproblems simultaneously. The Benders subproblem in cross decomposition is a heuristic, but Lagrangian-informed reformulation is an exact method to maximise Lagrangian bounds and compute fractional solutions.

Structured Dantzig-Wolfe decomposition [31] is even more similar. It allies column generation with row generation to generalise Dantzig-Wolfe decomposition: each iteration adds a subspace of the feasible set to the master problem. We do the same in static, fully enumerated reformulations.

The correctness conditions for structured Dantzig-Wolfe decompositions are established based on "the apparently unrelated definition of a *multiple choice* binary formulation for the [MCND]" [31]. This contrasts with Lagrangian-informed reformulations: creativity is only necessary when choosing which constraints to relax and which to preserve in the Lagrangian subproblem. Afterwards, the reformulation is automatic, and Section 2.3 will show how to solve any Lagrangian-informed reformulation as a structured Dantzig-Wolfe decomposition.

Thanks to Lagrangian-informed reformulation, every Lagrangian relaxation with a mixed integer subproblem gives rise to a structured Dantzig-Wolfe decomposition. The reformulation also bridges between cross decomposition methods and structured Dantzig-Wolfe decomposition: the heuristic phase of the former solves a memoryless simplification of the latter's master problem.

1.5 Practical considerations

Lagrangian-informed reformulation is an approach to compute Lagrangian bounds, and the resulting lower bound is contingent on the same desiderata as Lagrangian relaxation. The parallel even extends to the size of the reformulation, which reflects the complexity of the underlying Lagrangian subproblem. Sadly, designing a Lagrangian relaxation that balances bound strength and subproblem complexity remains more of a craft than a science. There are however a few rules of thumb to direct our efforts.

Lagrangian relaxations in which the subproblem is so simple that we can solve it as a linear program are pointless: the Lagrangian bound is equal to the linear programming bound, and the only hope is to maximise the Lagrangian dual faster than simplex or interior point algorithms can solve the linear relaxation. That is an unlikely proposition, especially given that we optimise the dual with a linear master problem.

Experience shows that pure 0/1 problems are well suited to Lagrangian relaxation. A reasonable explanation is that the Lagrangian dual is more easily maximised when we relax constraints that are never strongly violated in the subproblem. For example, Lagrangian decomposition relaxes constraints of the form

$$x = y.$$

When x and y are binary, each of these constraints is violated at most by ± 1, and we expect decomposition to perform well. This agrees with Lagrangian decomposition folklore, which also favours duplicating binary variables over general integer ones.

Our reformulations are fully formulated linear programs, and linear programming solvers rarely suffer from numerical issues. However, reformulations still benefit from relaxing constraints that only involve binary variables. Lagrangian-informed reformulation enumerates all assignments for the set of discrete variables in the program and eliminate choices that are already infeasible in the relaxed subproblem. This pruning disregards relaxed constraints, and we can keep Y to a manageable size by preserving constraints that involve variables with wide domains.

Another way to reduce the size of reformulations is to identify when $y = 0$ forces $x = 0$, *e.g.*, when the discrete choices determine capacities. We can then eliminate all variables and constraints associated with $y = 0$: we replace Y with $Y' = Y \setminus \{0\}$, and constraint (1.7) with

$$\sum_{s \in Y'} \theta^s \leq 1.$$

For example, we can perform this substitution on the SAFP components of (MC^+).

This is a special case of the general observation that fixing $y = s$ exposes opportunities for simplification in constraint (1.6). We may fix some of x^s at precomputed values, and simplify or eliminate rows in (1.6). Automatic presolving will identify many of these cases, but may be disabled when re-optimising; we should explicitly specialise (1.6) to $y = s$ when possible.

Finally, one rare reliable fact about Lagrangian relaxations is that master problems require fewer subproblem evaluations when they are disaggregated [29, 52]. Disaggregation factorises H, the convex hull of the subproblem, into the cartesian product $H_1 \times H_2 \times \ldots \times H_m$, where each H_i is the convex hull for a subproblem involving fewer variables, and thus fewer extreme points. The same factorisation is useful for Lagrangian-informed reformulations as well: factorising

$$(Y, \mathbb{R}^n) = (Y_1, X_1) \times (Y_2, X_2) \times \ldots \times (Y_m, X_m)$$

yields a more compact reformulation. We give one example in Chapter 3, where the disaggregated reformulation *contracts* a formulation while improving its bound.

These rules of thumb are only that, folklore extracted from hard-earned experience. Lagrangian-informed reformulation does not offer more solid footing to devise relaxation schemes. However, the tools we describe in this dissertation make it more likely that we will achieve in practice the theoretical strength of Lagrangian relaxations.

1.6 Summary

We introduced Lagrangian-informed reformulation, a method to strengthen the linear relaxation of mixed integer programs. Prior work describes special cases of the method, *e.g.*, formulations for non-convex cost functions [18, 19, 67], cross decomposition heuristics for mixed integer programs [49, 66], or row and column generation for multicommodity capacitated network design [30, 31]. We unify and generalise this work to expose a range of intermediate relaxations and solution methods.

Our reformulation is a straightforward application of the multiple choice model [4, 50, 67] to reformulate the Dantzig-Wolfe master problem [21, 22] compactly. Both components were introduced decades ago, and it is surprising that the general reformulation was not described earlier.

Perhaps the reason is that our reformulated programs only became tractable recently. The next chapter describes methods that rely on the power of contemporary computers, coupled with efficient linear optimisation algorithms, to solve the linear relaxations of large Lagrangian-informed reformulations. Chapters 4 and 5 will show that it is now practical to optimise linear programs involving millions of variables and constraints, when the alternative is to solve (less) large integer programs.

CHAPTER 2

SOLVING LAGRANGIAN-INFORMED REFORMULATIONS

Computing solutions or bounds for Lagrangian-informed reformulations poses practical challenges: improvements in bound strength are usually accompanied by increases in formulation size.

Rarely, reformulations comprise fewer variables and constraints than the original mixed integer program. The reformulation in Chapter 3 is one specimen. Despite its small size, the integer reformulated program (*i.e.*, with weight variables $\theta \in \mathbb{B}^{|Y|}$) is difficult to solve to optimality: Lagrangian-informed reformulation strengthens the linear relaxation by discarding the logical structure encoded in discrete decision variables. Results in Chapter 3 show that we should re-introduce decision variables eliminated during reformulation: the final mixed integer program is larger but exposes better branching choices during branch-and-bound. Similar observations are frequent for branch-and-price methods [6, 23, 61], and the conclusion is not surprising.

This chapter describes methods for the common case: Lagrangian-informed reformulations with linear relaxations that, although smaller than equivalent Dantzig-Wolfe reformulations, are too large for standard linear programming solvers.

Section 2.1 quickly recalls the classical trick of initialising optimisation methods for the Lagrangian dual with an optimal solution to a simpler linear relaxation. We then extend this initialisation approach to Lagrangian decompositions.

Section 2.2 introduces a novel method to warm-start the dual simplex [16, 20, 60] algorithm by solving a smaller linear program. The method suits Lagrangian-informed reformulations when Y scales pseudopolynomially with the instance size. The linear relaxations of such reformulations may reach the edge of tractability: it can be practical to formulate the relaxations with off-the-shelf solvers, but not to solve them. A warm start quickly improves the initial approximate solution and enables the solution of larger relaxations. When we represent Lagrangian-informed reformulations explicitly, this method eliminates an important advantage of classical Lagrangian master problems:

the dual simplex algorithm now benefits from the same warm starts, and (in practice) will solve Lagrangian-informed reformulations faster than non-differentiable optimisation methods can maximise Lagrangian dual functions.

Chapter 5 applies this warm-starting method to a location problem with modular costs. Experiments on large-scale instances show that a warm-started dual simplex algorithm improves the same dual solutions more quickly than the bundle and volume algorithms. Chapter 5 also combines the warm start with a primal heuristic to reduce the optimality gap more quickly than a parallel branch-and-cut solver.

Section 2.3 points out the relationship between our Lagrangian-informed reformulation and structured (stabilised) Dantzig-Wolfe decomposition [31]: every reformulation satisfies the conditions for structured Dantzig-Wolfe decomposition. We conclude that, if the subproblem for a Lagrangian relaxation is a mixed integer program, we can solve that relaxation with structured Dantzig-Wolfe decomposition.

Converting a Lagrangian-informed reformulation to a structured Dantzig-Wolfe decomposition is useful when Y grows quickly, *e.g.*, exponentially in the instance size. Such reformulations are so large that it is impractical to even represent them explicitly. We propose to solve large reformulations through dynamic row and column generation, *i.e.*, a form of structured (stabilised) Dantzig-Wolfe decomposition [31]. We also describe three linear stabilisation terms for our restricted master problems. Results in Chapter 4 confirm that structured decomposition offers better performance than the bundle and volume algorithms. In fact, incorporating a primal heuristic yields a Lagrangian heuristic that is competitive with a branch-and-cut solver.

2.1 Linear warm start for Lagrangian relaxations

The equivalence between linear relaxations and Lagrangian duals for linear programming subproblems is a classical result of Lagrangian duality [36, 37].

Corollary 2 (Section 1.2) tells us that optimal dual multipliers for constraint (1.1) in the linear relaxation of (P) are optimal Lagrange multipliers for the linear relaxation of (S_λ). These multipliers are also valid—but not necessarily optimal—for the initial mixed integer Lagrangian subproblem (S_λ). (S_λ) optimises over a smaller feasible space than ($\bar{S}_\lambda$), and we obtain a Lagrangian bound that dominates the linear programming bound.

The practical consequence is that we can solve a linear program to extract dual multipliers and initialise a Lagrangian master problem with these heuristic multipliers. Even if we then maximise the Lagrangian dual approximately, our heuristic Lagrangian bound dominates the linear programming bound.

We extend this trick to compute initial multipliers for Lagrangian decompositions [45], *i.e.*, Lagrangian relaxations of artificial linking constraints, from solutions to the original problems without linking constraints.

Lagrangian decomposition is a special case of Lagrangian relaxation: the idea is to link components in the original formulation explicitly through artificial constraints, and then relax these linking constraints to decompose the formulation into independent Lagrangian subproblems. We just explained how to compute initial Lagrange multipliers for arbitrary relaxed constraints, but that solution is ill adapted to Lagrangian decomposition: the intermediate formulation with artificial linking constraints is larger than the initial one. We propose to derive the same multipliers from the latter more compact formulation.

Lagrangian decomposition for linear programs takes a problem of the form

(E) $$\min_x cx + fDx$$

subject to

$$Ax \leq b, \tag{2.1}$$

$$CDx \leq e, \tag{2.2}$$

transforms it into the equivalent

(E′) $$\min_{x,y} cx + fy$$

subject to

$$Ax \leq b, \tag{2.3}$$

$$Cy \leq e \tag{2.4}$$

$$Dx - y = 0, \tag{2.5}$$

and then relaxes constraint (2.5). Matrix D is often the identity, but becomes useful when aggregating variables with linking constraints of the form

$$\sum_{i \in I(j)} x_i = y_j.$$

We wish to turn a dual solution to (E) into dual multipliers for constraint (2.5) in (E′). The key is to compute reduced costs.

Let λ and μ be, respectively, optimal dual multipliers for constraints (2.1) and (2.2). The optimal multipliers for linking constraint (2.5) are then

$$\nu = \mu C - f$$

while those for (2.3) and (2.4) remain, respectively, λ and μ.

This solution is in the dual domain of (E′): λ and μ are nonnegative (they are feasible multipliers for inequality constraints in (E)), and multipliers for equality constraints can take arbitrary values. Moreover, because λ and μ are feasible dual multipliers for (E),

$$\lambda A + \mu CD = c + fD,$$

and we conclude that the dual constraints for x and y in (E′) are satisfied:

$$\lambda A + \nu D = \lambda A + \mu CD - fD = c,$$

and

$$\mu C - \nu I = f.$$

This dual solution is also optimal for (E′) iff the initial one is optimal for (E): both their objective values are $\lambda b + \mu e$ (the right-hand side of constraint (2.5) is 0).

Intuitively, the effect of constraint (2.2) on the reduced costs of x instead appears via constraint (2.5) in (E′): the multipliers for (2.5) are exactly the portion of the reduced costs of x (in (E)) attributable to (2.2) at optimum.

This equivalence holds for linear programs but not for more complex formulations in which x and y are discrete. Nevertheless, optimal multipliers for linear relaxations are useful initialisers for tighter relaxations.

2.2 Warm start for large reformulations

This section describes a novel method to solve large linear programs by first computing dual multipliers for a more tractable approximation. It is a clumsy workaround for the difficulty of warm-starting interior point methods, but Chapter 5 demonstrates its efficacy when combined with the heuristic of Section 2.1.

The simplex algorithm benefits naturally from advance bases. However, it is not obvious how to generate a basis from a dual solution to an unrelated linear program. The primal simplex algorithm re-optimises efficiently when the program gains variables and the dual simplex performs excellently after the addition of constraints, but neither is well suited to arbitrary modifications.

Contrary to simplex algorithms, interior point methods [69] work directly in terms of primal and dual solutions. However, warm-starting these methods is still an area of active research [25, 43, 51].

We propose to compute an advance simplex basis from a suboptimal dual solution. Dual multipliers help define a relaxation of the large initial linear program; we solve that relaxed subproblem to find an advance basis and we re-add the relaxed constraints to the subproblem; this final program is equivalent to the initial one and we re-optimise it with the dual simplex algorithm.

Formally, we wish to obtain an advance basis for linear programs of the form (B) starting from λ, a vector of potentially suboptimal multipliers for constraint (2.6).

(B) $$\min_x cx$$

subject to

$$Ax = b, \tag{2.6}$$
$$Dx \geq e.$$

Relaxing constraint (2.6) leaves a linear Lagrangian subproblem that we solve once, with multipliers λ:

(B′) $$-\lambda b + \min_x (\lambda A + c)x$$

subject to

$$Dx \geq e.$$

We need an optimal basis for this subproblem, and (B′) is smaller than (B); we propose to solve (B′) with the simplex algorithm.

We then re-insert constraint (2.6) in (B′) to obtain (B″). This last formulation is equivalent to (B): the only difference lies in the objective function, and constraint (2.6) forces the Lagrangian term $\lambda(Ax - b)$ to zero.

(B″) $$-\lambda b + \min_x (\lambda A + c)x$$

subject to

$$Ax = b,$$
$$Dx \geq e.$$

Formulation (B'') adds constraints to (B'); the dual simplex algorithm is well suited to re-optimisation after such modifications. Better, each iteration of the dual simplex algorithm will increase the lower bound: we can stop solving (B'') at any time and find a lower bound that dominates that of (B').

This is particularly interesting when λ is almost optimal: the lower bound computed with (B') is then close to the optimal value for (B), and (B'') quickly reaches optimality.

We apply this warm start to Lagrangian-informed reformulations that are too large for off-the-shelf solvers, but small enough to be represented explicitly. We obtain dual multipliers for constraints (1.9) and (1.10) by considering (P^+) as a reformulation of (P'), a version of (P) with explicit linking constraints (2.8) and (2.9).

$$(\mathrm{P}') \qquad \min_{x,y,x',y'} cx + fy$$

subject to

$$Ax + By \leq d,$$

$$Ex' + Fy' \leq g, \tag{2.7}$$

$$x = x', \tag{2.8}$$

$$y = y', \tag{2.9}$$

$$x' \geq 0, \tag{2.10}$$

$$y' \in Y. \tag{2.11}$$

Reformulation (P^+) replaces constraints (2.8) and (2.9) with (1.9) and (1.10), and optimal multipliers for the former pair of constraints (in the linear relaxation of (P')) are valid for the latter. They may however be suboptimal, as (P^+) replaces constraints (2.7), (2.10) and (2.11) with the tighter (1.6), (1.7), (1.8) and (1.11).

This section showed how to compute dual multipliers for artificial constraints (2.8) and (2.9) (and thus for (1.9) and (1.10)) from optimal dual multipliers for constraint (1.2). We exploit this in Algorithm 1 to warm-start the optimisation of (P^+). The first step only computes dual values for the initial (integrated) linear formulation (P); when it is large, we use an interior point method and disable crossover [12].

Algorithm 1 Warm-starting algorithm for Lagrangian-informed reformulations

Solve the linear relaxation of (P); let π be optimal multipliers for constraint (1.2).
Derive Lagrange multipliers λ for constraints (1.9) and (1.10) from π.
Relax (P^+) by dualising (some of) constraints (1.9) and (1.10) with multipliers λ.
Solve the linear relaxation of this Lagrangian subproblem; let b be the optimal basis.
Add constraints (1.9) and (1.10) to the formulation; the result is equivalent to (P^+).
Solve this new linear program with the dual simplex algorithm, starting from b.

2.3 Structured Dantzig-Wolfe decomposition

Structured (stabilised) Dantzig-Wolfe decomposition ($S^{(2)}$DW) [31] is an algorithmic scheme for solving specially structured large-scale linear programs that relies on three general assumptions, which we recall in Section 2.3.1. In this section, we show that all Lagrangian-informed reformulations possess the appropriate structure for $S^{(2)}$DW and describe our implementation of Lagrangian-informed structured Dantzig-Wolfe decomposition.

2.3.1 Dynamic generation of variables and constraints

The size blowup for (P^+) with respect to (P), the initial formulation, parallels the cardinality of Y: Lagrangian-informed reformulation introduces constraints over fresh variables for each $s \in Y$. However, row and column generation lore tells us that the majority of these variables and constraints are irrelevant at optimum; it is natural to solve a restriction of (P^+) and only enumerate Y on demand. We propose to implement an $S^{(2)}$DW and choose which $s \in Y$ to enumerate by solving the Lagrangian subproblem (S_λ). First, we show that this satisfies the assumptions for applying $S^{(2)}$DW [31].

If we disregard constraint (1.5) (*i.e.*, only consider the reformulation for H, the convex hull of (S_λ)), reformulation (P^+) satisfies

Assumption 1 For a finite vector of variables θ and matrices C, Γ and γ of appropriate dimension, $\text{Conv}(X) = \{x = C\theta \mid \Gamma\theta \leq \gamma\}$.

Variables θ^s and x^s in (P$^+$) correspond to θ in this assumption, constraints (1.9) and (1.10) to matrix C, and constraints (1.6), (1.7) and (1.8) to (Γ, γ).

The two remaining assumptions in are more technical and leak implementation details of bundle methods. Let $(\overline{\mathrm{P}^+})$ be the linear relaxation of a partially enumerated version of (P$^+$): rather than enumerating variables and constraints for all $s \in Y$, $(\overline{\mathrm{P}^+})$ is restricted to $s \in \mathscr{B} \subset Y$.

Assumption 2 $\Gamma_{\mathscr{B}} \bar{\theta}_{\mathscr{B}} \leq \gamma_{\mathscr{B}}$ and $\theta = \left[\bar{\theta}_{\mathscr{B}}, 0\right] \Rightarrow \Gamma\theta \leq \gamma$.

Assumption 3 Let $\bar{x}$ be a point such that $\bar{x} \in \mathrm{Conv}(X) \setminus X_{\mathscr{B}}$; then, it must be easy to update B and the associated $\Gamma_{\mathscr{B}}$, $\gamma_{\mathscr{B}}$ and $C_{\mathscr{B}}$ to a set $\mathscr{B}' \supsetneq \mathscr{B}$ (which satisfies Assumption 2) such that there exists $\mathscr{B}'' \supseteq \mathscr{B}'$ with $\bar{x} \in X_{\mathscr{B}''}$.

The second assumption means that $(\overline{\mathrm{P}^+})$ must be a restriction of (P$^+$); it must always be possible to "pad" a partial solution with zeros to obtain a feasible solution in the complete formulation. $(\overline{\mathrm{P}^+})$ satisfies this assumption: it is equivalent to adding constraints

$$\theta^s = 0, \quad \forall s \in Y \setminus \mathscr{B}$$

to the full formulation (P$^+$).

The third assumption is that, given $(\bar{x}, \bar{y}) \in H$, it is easy to extend $(\overline{\mathrm{P}^+})$ to (eventually) include $(\bar{x}, \bar{y})$. Lagrangian-informed reformulations satisfy a weaker form of this assumption: in practice, it suffices to be able to extend when $(\bar{x}, \bar{y})$ is an *extreme point* of H. We achieve that by adjoining $\bar{y}$ to $\mathscr{B}$.

The relationship between $(\overline{\mathrm{P}^+})$ and standard bundle master problems that enumerate extreme points of H is discussed in Frangioni and Gendron [31]. We only note that dual multipliers for constraints (1.5) serve as tentative Lagrange multipliers and that solving (S$_\lambda$) for these multipliers both computes a lower bound and separates an extreme point of H. If the Lagrangian bound is lower than the objective value of $(\overline{\mathrm{P}^+})$ with s restricted to $\mathscr{B} \subset Y$, the value of y in optimal solutions of the subproblem must not be in $\mathscr{B}$; we adjoin one such value to $\mathscr{B}$ and expand $(\overline{\mathrm{P}^+})$. Otherwise, any optimal solution of $(\overline{\mathrm{P}^+})$ is also optimal for (P$^+$).

2.3.2 Lagrangian-informed $S^{(2)}$DW

We have shown that any Lagrangian-informed reformulation can be cast in the framework of structured (stabilised) Dantzig-Wolfe decomposition ($S^{(2)}$DW). We now show how to use such a decomposition to solve instances of Lagrangian-informed reformulation.

Algorithm 2 provides an outline of the structured Dantzig-Wolfe algorithm [31]. We must determine how to initialise the bundle $\mathscr{B}$ and how to update the restricted master problem ($\overline{\mathrm{P}^+}$); we will also replace the termination condition.

Algorithm 2 The structured Dantzig-Wolfe algorithmic skeleton

Initialise the bundle $\mathscr{B}$.
repeat
  Solve the linear program ($\overline{\mathrm{P}^+}$) for $(\tilde{x}, \tilde{y})$;
    let $\tilde{\pi}$ be optimal dual multipliers of $Ax + By \leq d$.
  Solve (S_λ), with $\lambda = \tilde{\pi}$, for $(\bar{x}, \bar{y})$.
  Update $\mathscr{B}$ as in Assumption 3.
until $c\tilde{x} + f\tilde{y} \leq c\bar{x} + f\bar{y} + \tilde{\pi}(d - A\bar{x} - B\bar{y})$

We already defined the restricted master ($\overline{\mathrm{P}^+}$) as the linear relaxation of (P^+) with variables θ^s and x^s and constraints (1.22) and (1.24) only enumerated for $s \in \mathscr{B} \subset Y$. In our implementation of structured Dantzig-Wolfe decomposition, the bundle $\mathscr{B}$ is a set of assignments $s \in Y$ for y. It consists of partial solutions to the Lagrangian subproblem, not complete ones (*i.e.*, extreme points): solving ($\overline{\mathrm{P}^+}$) determines the value of x^s.

We propose to initialise $\mathscr{B}$ by solving the linear relaxation of the initial formulation (P). The dual multipliers for constraint (1.1) at optimum define a Lagrangian subproblem (S_λ) that we solve as a mixed integer program. Let $\bar{y}$ be the value of y in an optimal solution to that subproblem; we initialise $\mathscr{B}$ to the singleton $\{\bar{y}\}$.

We adjoin $\bar{y}$ to $\mathscr{B}$ by adding $x^{\bar{y}}$ and $\theta^{\bar{y}}$ to the restricted master and constraining them with new instances of (1.22) and (1.24). Convexity constraint (1.7) and linking constraints (1.25) and (1.26) are then altered to include $x^{\bar{y}}$ and $\theta^{\bar{y}}$ in their sums.

We propose to re-optimise the dual of ($\overline{\mathrm{P}^+}$) with the dual simplex algorithm. This may be surprising, as neither the primal nor the dual simplex algorithms are well suited to re-optimisation after general updates. Our expansion of the restricted master problem is a special case: the update is equivalent to relaxing the implicit constraint

$$\theta^{\bar{y}} = 0$$

into

$$0 \leq \theta^{\bar{y}} \leq 1,$$

and the previous basis is primal feasible for the expanded problem. The primal simplex algorithm thus benefits from a directly usable advance basis. However, the dual simplex algorithm offers superior numerical efficiency and we prefer to work with the dual of the restricted master problem.

The termination condition

$$c\tilde{x} + f\tilde{y} \leq c\bar{x} + f\bar{y} + \tilde{\pi}(d - A\bar{x} - B\bar{y})$$

is always correct, but overly general for our uses. We instead stop when $\bar{y}$, the value of y in an optimal solution of (S_λ), is already in the bundle $\mathscr{B}$. This alternative criterion is simple, avoids numerical issues of scale between the constraints and the objective function, and trivially guarantees finite convergence (Y is a finite set and $\mathscr{B}$ grows after each iteration).

These three changes yield Algorithm 3. The only remaining decisions are in the definition of the Lagrangian relaxation that guides the reformulation: which constraints to relax (are in (1.1)) and which discrete variables, if any, to turn into continuous ones (become part of x).

Chapter 4 will show that this unstabilised variant of Dantzig-Wolfe decomposition is practical only when the linear relaxation of (P) is a close approximation of (P^+). We now describe a linear stabilisation scheme that improves the empiric iteration and time complexity of the method.

Numerical results [31] show that the number of iterations is most reduced by a quadratic stabilisation term. However, the runtime performance of quadratic program-

Algorithm 3 Lagrangian-informed structured Dantzig-Wolfe algorithm

Solve the linear relaxation of (P);
  let $\tilde{\pi}$ be optimal dual multipliers for constraint (1.1).
Solve mixed integer program (S_λ), with $\lambda = \tilde{\pi}$, for $(\bar{x}, \bar{y})$.
Let $\mathscr{B} = \{\bar{y}\}$, and formulate $(\overline{\mathrm{P}^+})$ with the single block $s = \bar{y}$.
loop
  Solve (the dual of) $(\overline{\mathrm{P}^+})$ for $(\tilde{x}, \tilde{y})$;
    let $\tilde{\pi}$ be optimal dual multipliers for constraint (1.1).
  Solve (S_λ), with $\lambda = \tilde{\pi}$, for $(\bar{x}, \bar{y})$.
  if $\bar{y} \in \mathscr{B}$ **then**
    return $(\tilde{x}, \tilde{y})$, an optimal solution of (P^+).
  end if
  Update $\mathscr{B}$ to $\mathscr{B} \cup \{\bar{y}\}$.
  Extend the restricted master problem $(\overline{\mathrm{P}^+})$ with a new block for $s = \bar{y}$.
end loop

ming solvers lags behind that of linear programming solvers, and an ℓ_∞ constraint on dual multipliers delivered the best reduction in total runtime. That is why we only consider three linear trust region terms: a variant of the ℓ_∞ constraint, an ℓ_1 constraint, and the intersection of both.

Regardless of the stabilisation term, structured stabilised Dantzig-Wolfe decomposition algorithms are implementations of the trust region framework [17] for non-smooth optimisation (that leverage especially talkative oracles which produce more than individual subgradients). Algorithm 4 shows our specialisation of the basic trust region algorithm [17] for Lagrangian-informed structured Dantzig-Wolfe decomposition.

The first iterations are identical to that of the unstabilised decomposition (Algorithm 3): we leave the trust radius Δ at ∞ until two consecutive unstabilised iterations fail to improve the lower bound. Our aim is for easy instances to reach optimality before stalling and thus not suffer from any stabilisation overhead.

Once unstabilised structured Dantzig-Wolfe decomposition has failed to improve the best lower bound twice in a row, we insert the trust region term. We initialise the radius Δ to 10^{-3}, and the multipliers associated with the best bound to date become the stabilisation center λ.

These parameters are only updated when the stabilisation constraint is active and

Algorithm 4 Lagrangian-informed structured stabilised Dantzig-Wolfe algorithm

Solve the linear relaxation of (P); let $\tilde{\pi}$ be optimal dual multipliers for (1.1).
Solve mixed integer program (S_λ), with $\lambda = \tilde{\pi}$, for $(\bar{x}, \bar{y})$.
Let $\bar{z}_{-1} = (\tilde{\pi}A + c)\bar{x} + (\tilde{\pi}B + f)\bar{y} - \tilde{\pi}d$.
Set $k = 0$, the stabilisation center $\lambda_0 = \tilde{\pi}$ and the stabilisation radius $\Delta_k = \infty$.
Let $\mathscr{B} = \{\bar{y}\}$, and formulate the *stabilised* master $(\overline{P^+})$ with one block, $s = \bar{y}$.
for $k = 0, 1, 2, \ldots$ **do**
  Solve $(\overline{P^+})$ for $(\tilde{x}, \tilde{y})$; let $\tilde{\pi}$ be optimal multipliers for (1.1) and $\tilde{z} = c\tilde{x} + f\tilde{y}$.
  Solve (S_λ), with $\lambda = \tilde{\pi}$, for $(\bar{x}, \bar{y})$; let $\bar{z}_k = (\tilde{\pi}A + c)\bar{x} + (\tilde{\pi}B + f)\bar{y} - \tilde{\pi}d$.
  if the stabilisation contraint is inactive and $\bar{y} \in \mathscr{B}$ **then**
    return $(\tilde{x}, \tilde{y})$, an optimal solution of (P^+).
  else if $\Delta_k = \infty$ **then**
    if $\bar{z}_k > \bar{z}_{k-1}$ **then**
      Set $z^* = \bar{z}_k$. {New best multipliers}
      $\Delta_{k+1} = \infty$, $\lambda_{k+1} = \tilde{\pi}$.
    else
      $\lambda_{k+1} = \lambda_k$.
      if $\lambda_{k-1} = \lambda_k$ **then**
        $\Delta_{k+1} = 10^{-3}$. {Stalled twice in a row $\Rightarrow$ *unsuccessful* iteration}
      else
        $\Delta_{k+1} = \infty$.
      end if
    end if
  else if $\frac{\|\bar{z}_k - \tilde{z}\|}{\max(\|\bar{z}_k\|, \|\tilde{z}\|)} < 10^{-3}$ and the stabilisation constraint is active **then**
    $\lambda_{k+1} = \tilde{\pi}$.
    if the previous iteration was *unsuccessful* **then**
      $\Delta_{k+1} = \Delta_k$. {*Successful* iteration}
    else
      $\Delta_{k+1} = (2 + 8\mathbf{I}[\bar{y} \in \mathscr{B}])\Delta_k$. {*Very successful* iteration}
    end if
  else
    $\lambda_{k+1} = \lambda_k$, $\Delta_{k+1} = \Delta_k$. {*Unsuccessful* iteration}
  end if
  Update $\mathscr{B}$ to $\mathscr{B} \cup \{\bar{y}\}$.
  Extend the restricted stabilised master problem $(\overline{P^+})$ with the new block
    for $s = \bar{y}$, stabilisation center λ_{k+1} and radius Δ_{k+1}.
end for

$\tilde{z}$, the value estimate in the restricted master problem, and $\bar{z}$, the Lagrangian bound for the current multipliers, are almost identical. We leave the stabilisation parameters unchanged when the stabilisation constraint is inactive at optimum because any modification would likely have no effect but to hinder re-optimisation. Otherwise, the model of the Lagrangian dual defined by the bundle $\mathscr{B}$ is locally accurate and we update the stabilisation center to $\tilde{\pi}$, the current candidate for optimal Lagrange multipliers. The stabilisation radius Δ only grows when we update the stabilisation center two iterations in a row; we then multiply Δ by 10 if $\bar{y}$, (part of) the solution to the current Lagrangian subproblem, is already in the bundle $\mathscr{B}$, and by 2 otherwise.

The update strategy for the stabilisation radius differs from the one proposed by Conn et al [17]. Algorithm 4 modifies the radius more rarely, and never shrinks it. We made this choice because the master problem is large, and changes to the stabilisation term slow re-optimisation down. In our situation, we prefer to modify the stabilisation radius rarely; when we do increase it, we do so aggressively. We also exploit the high quality of our model. The restricted master problem eventually becomes locally *exact*; the bundle then fails to grow, and this triggers a greater increase of the radius.

2.3.3 Linear stabilisation terms in $\mathbf{S^2DW}$

The one remaining detail is the stabilisation term. We add these terms to keep the dual multipliers for constraint (1.5) in the vicinity of the stabilisation center, an acceptable solution whose neighbourhood is well approximated by the master problem.

A simple limit on the ℓ_∞ distance from the stabilisation center yields interesting results in Frangioni and Gendron [31]. The term constrains the dual of the master problem with

$$\begin{aligned} &\|\pi - \lambda\|_\infty \leq \Delta \\ \Leftrightarrow &\lambda - \Delta\mathbf{1} \leq \pi \leq \lambda + \Delta\mathbf{1}, \end{aligned}$$

where λ is the stabilisation center, Δ the stabilisation radius, and π the dual multipliers for constraint (1.5).

The effect of this term on the primal problem is to relax constraint (1.5) with slack variables that are then penalised: the Fenchel dual of an ℓ_∞ constraint is an ℓ_1 penalty [48]. Formulation (P^+) becomes

$$(\tilde{\text{P}}^+) \qquad \min_{x,y,z} cx + fy - \lambda z + \Delta\|z\|_1$$

subject to

$$\begin{aligned}
Ax + By - z &\leq d, && && (2.12)\\
Ex^s &\leq \theta^s(g - Fs), && \forall s \in Y,\\
\sum_{s\in Y} \theta^s &= 1,\\
x^s &\geq 0, && \forall s \in Y,\\
x &= \sum_{s\in Y} x^s,\\
y &= \sum_{s\in Y} s\theta^s,\\
0 \leq \theta^s &\leq 1, && \forall s \in Y,\\
z &\geq 0.
\end{aligned}$$

There are two additions to the objective function: $-\lambda z$, which favours dual solutions in which the multipliers for constraint (2.12) are parallel to the stabilisation center λ, and $\Delta\|z\|_1$, which brings slack variables z close to 0 as Δ grows. We compute the ℓ_1 norm as the sum of the non-negative elements of z,

$$\sum_{i\in I} z_i,$$

where I is a set of indices for the rows of (1.5).

Frangioni and Gendron [31] chose the stabilisation radius for the ℓ_∞ constraint by hand for each instance, after preliminary experiments. In Algorithm 4, we rely on a simple dynamic adjustment strategy that benefits from less fickle stabilisation constraints. Instead of a direct constraint on the ℓ_∞ distance, we consider a relative distance. Relative norms seem better suited to linear master problems: basic dual multipliers take arbitrarily large values while nonbasic multipliers equal zero. The constraint on dual variables π becomes

$$\lambda - \Delta \max(1, |\lambda|) \leq \pi \leq \lambda + \Delta \max(1, |\lambda|),$$

where $\max(1, |\lambda|)$ is computed element by element. The effect on the primal master problem is that constraint (2.12) becomes

$$Ax + By - \min(1, |\lambda|^{-1})z \leq d,$$

where $\min(1, |\lambda|^{-1})$ is also computed element-wise.

We also experimented with a relative ℓ_1 constraint on dual multipliers. In that stabilisation scheme, dual variables π are constrained with $\|\frac{\pi}{\max(1,|\lambda|)}\|_1 \leq \Delta$, where the division and max are element-wise, *i.e.*,

$$\sum_{i \in I} \frac{\pi_i}{\max(1, |\lambda_i|)} \leq \Delta.$$

The effect on $(\tilde{\mathrm{P}}^+)$ is that the objective function becomes

$$\min_{x,y,z} cx + fy - \lambda z + \Delta \|z\|_\infty,$$

(the Fenchel dual of the ℓ_1 constraint is an ℓ_∞ penalty) and that, as for the ℓ_∞ constraint, (2.12) becomes

$$Ax + By - \min(1, |\lambda|^{-1})z \leq d.$$

We express the ℓ_∞ penalty in this linear minimisation program by replacing $\Delta\|z\|_\infty$ in the objective function with Δd and adding constraints

$$z_i \leq d, \quad \forall i \in I.$$

The last stabilisation term we considered is the intersection of both the ℓ_∞ and the ℓ_1 (relative) constraints on dual variables π, with the same stabilisation center and radius. The effect of that term on the primal is to add two sets of non-negative slack variables and to penalise each set separately, one with an ℓ_1 penalty and the other with an ℓ_∞ penalty.

This concludes the complete description of our structured stabilised Dantzig-Wolfe decomposition for Lagrangian-informed reformulations. It is almost simplistic, compared with state-of-the-art bundle methods [28, 48, 59]. However, there are strong parallels between the latter methods and structured stabilised Dantzig-Wolfe decomposition [31]; we simply have yet to exploit them. Implementing replacement and aggregation strategies for the bundle should prove fruitful: the master problem grows quickly and smaller bundles make iterations quicker. These strategies will also expose a range of methods between cross decomposition [49, 66] (*i.e.*, bundles of size one) and structured Dantzig-Wolfe decompositions with unlimited memory. Aggressive aggregation schemes are particularly attractive: some guarantee convergence for bundles as small as two elements [55].

2.4 Summary

This chapter exploits basic duality results to develop a novel warm start method for Lagrangian-informed reformulations and other large linear programs. When solving an implicitly defined Lagrangian dual, multipliers from a related linear program can serve as initial stabilisation centers. We show how to obtain an advance simplex basis from such dual multipliers; this allows us to solve barely tractable linear reformulations that are equivalent to strong Lagrangian dual functions.

We also showed how every Lagrangian-informed reformulation can be solved with structured Dantzig-Wolfe decomposition [31]. The process is completely mechanical and only depends on the identification of relaxed constraints. Every Lagrangian relaxation with a mixed integer subproblem yields a structured Dantzig-Wolfe decomposition.

This result contrasts with the assumptions for structured Dantzig-Wolfe decomposition given in Frangioni and Gendron [31]. Our reformulations automatically define a master problem and a bundle extension method that satisfy these assumptions.

The two methods let us solve the linear relaxations of large Lagrangian-informed reformulations. Without them, the reformulations would be impractical or trivially equivalent to Dantzig-Wolfe decomposition. Chapters 4 and 5 will demonstrate that, thanks to these methods, we maximise Lagrangian duals more efficiently than some classical non-differentiable optimisation methods.

CHAPTER 3

A SIMPLE LAGRANGIAN-INFORMED REFORMULATION FOR TUFLP-S

This chapter describes a novel Lagrangian-informed reformulation for the Two-level Uncapacitated Facility Location Problem with Single Sourcing (TUFLP-S), a variant of the classical Two-level Uncapacitated Facility Location Problem (TUFLP). The TUFLP consists of determining how to connect a set of customer locations to depot nodes via satellite (transshipment) nodes, while minimising the total of the transportation costs to every customer and of the fixed costs for satellite and depot nodes [1]. The TUFLP-S adds the constraint that each open satellite must be linked to exactly one depot.

We chose the TUFLP-S for three reasons: it forms the basis of the industrial location problem tackled in Chapter 5; we can compare our formulations with prior work thanks to TUFLP instances that we can also solve as TUFLP-S; and last but crucial, a straightforward mixed integer formulation for the TUFLP-S is amenable to Lagrangian-informed reformulation.

There are three sections to this chapter. Section 3.1 provides further details on the TUFLP and introduces $(\mathrm{PC_{path}})$ and $(\mathrm{PG_{path}})$, mixed integer programming formulations for, respectively, the TUFLP and a generalisation of the TUFLP. Section 3.2 adapts $(\mathrm{PG_{path}})$ into $(\mathrm{PS_{weak}})$, our initial mixed integer formulation for TUFLP-S, and improves $(\mathrm{PS_{weak}})$ with a Lagrangian-informed reformulation; the result is formulation $(\mathrm{PS_{path}})$. Finally, Section 3.3 compares $(\mathrm{PS_{path}})$ and close variants, $(\mathrm{PS'_{path}})$ and $(\mathrm{PS^0_{path}})$, with each other, with $(\mathrm{PC_{path}})$, and with a more recent mixed integer formulation for the TUFLP.

3.1 Two-level uncapacitated facility location problems

The TUFLP extends the single-level Uncapacitated Facility Location Problem [57] (UFLP)—between depots and customers—by replacing the set of depot locations with two sets: one of depots and another of intermediate satellite nodes. The task is to decide which depots and satellites to open and which depot–satellite pair to assign to each customer, in order to satisfy customer demands at minimum cost [1] [1].

A general two-level uncapacitated facility location problem, TUFLP-G, is introduced in Barros and Labbé [8]. In addition to transportation costs for each path from a depot to a satellite to a customer and fixed costs on the use of depots and satellites, the problem includes fixed costs for arcs from depots to satellites. Figure 3.1 outlines a solution to an instance of the TUFLP-G: the solution incurs fixed costs for the use of i_1, i_2, j_2, and j_3, and for the arcs $i_1 \to j_2$, $i_2 \to j_2$ and $i_2 \to j_3$. Barros and Labbé [8] propose to solve this problem with mixed integer formulation (PG_{path}).

Let I be the set of potential depot locations, J the set of potential satellite locations, and K the set of customer locations, and let

$$y_i = \begin{cases} 1, & \text{if depot } i \text{ is open,} \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i \in I,$$

$$z_j = \begin{cases} 1, & \text{if satellite } j \text{ is open,} \\ 0, & \text{otherwise,} \end{cases} \qquad \forall j \in J,$$

$$t_{ij} = \begin{cases} 1, & \text{if depot } i \text{ and satellite } j \text{ are operating together,} \\ 0, & \text{otherwise,} \end{cases} \qquad \forall (i,j) \in I \times J,$$

$$x_{ijk} = \begin{cases} 1, & \text{if customer } k \text{ is served through pair } (i,j), \\ 0, & \text{otherwise,} \end{cases} \qquad \forall\, (i,j,k) \in I \times J \times K.$$

In addition, let f_i, g_j and h_{ij} be the fixed costs for, respectively, each depot $i \in I$, each satellite $j \in J$ and each pair of depot–satellite $(i,j) \in I \times J$, and let c_{ijk} be the total transportation cost of each path from a depot i to a satellite j to a customer k. Formulation (PG_{path}) [8] is

1. This section is heavily inspired by a prior collaboration with Bernard Gendron and Frédéric Semet [35] submitted to Transportation Science.

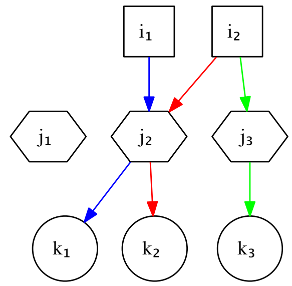


Figure 3.1: Solution to a TUFLP-G that violates single sourcing

$$(\mathrm{PG_{path}}) \qquad \min_{y,z,t,x} \sum_{i \in I} f_i y_i + \sum_{j \in J} g_j z_j + \sum_{(i,j) \in I \times J} h_{ij} t_{ij} + \sum_{(i,j,k) \in I \times J \times K} c_{ijk} x_{ijk}$$

subject to

$$\sum_{(i,j) \in I \times J} x_{ijk} = 1, \qquad \forall k \in K, \tag{3.1}$$

$$x_{ijk} \leq t_{ij}, \qquad \forall (i,j,k) \in I \times J \times K, \tag{3.2}$$

$$\sum_{j \in J} x_{ijk} \leq y_i, \qquad \forall (i,k) \in I \times K, \tag{3.3}$$

$$\sum_{i \in I} x_{ijk} \leq z_j, \qquad \forall (j,k) \in J \times K, \tag{3.4}$$

$$0 \leq x_{ijk} \leq 1, \qquad \forall (i,j,k) \in I \times J \times K,$$

$$y_i \in \{0,1\}, \qquad \forall i \in I,$$

$$z_j \in \{0,1\}, \qquad \forall j \in J,$$

$$t_{ij} \in \{0,1\}, \qquad \forall (i,j) \in I \times J.$$

Constraints (3.1) guarantee that the demand for each customer is satisfied exactly, and constraints (3.2) to (3.4) ensure that fixed costs are incurred for the use of depots, satellites and depot–satellite pairs. Variables x_{ijk} are continuous: there are no capacity limit so ($\mathrm{PG}_{\mathrm{path}}$) will never split the flow for a single customer across multiple paths.

Work on two-level uncapacitated facility location problems mostly studies a simplification of the general model: there are no fixed cost on links between depots and satellites ($h_{ij} = 0$). Two seminal papers [53, 63] marked early research on this classical TUFLP (TUFLP-C); they introduced formulations and specialised lower bounding methods, and exploited them in branch-and-bound algorithms. More recent approaches [1, 7, 58] are based on the mixed integer formulation that eliminates variables t_{ij} and constraints (3.2) from ($\mathrm{PG}_{\mathrm{path}}$). This does not affect linear programming bounds: forcing every $t_{ij} = 1$ costs nothing. The resulting formulation is $(\mathrm{PC}_{\mathrm{path}})$:

$$(\mathrm{PC}_{\mathrm{path}}) \qquad \min_{y,z,t,x} \sum_{i\in I} f_i y_i + \sum_{j\in J} g_j z_j + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}$$

subject to

$$\sum_{(i,j)\in I\times J} x_{ijk} = 1, \qquad \forall k \in K,$$

$$\sum_{j\in J} x_{ijk} \leq y_i, \qquad \forall (i,k) \in I \times K,$$

$$\sum_{i\in I} x_{ijk} \leq z_j, \qquad \forall (j,k) \in J \times K,$$

$$0 \leq x_{ijk} \leq 1, \qquad \forall (i,j,k) \in I \times J \times K,$$

$$y_i \in \{0,1\}, \qquad \forall i \in I,$$

$$z_j \in \{0,1\}, \qquad \forall j \in J.$$

$(\mathrm{PC}_{\mathrm{path}})$ is simple, but inherits the strength of ($\mathrm{PG}_{\mathrm{path}}$). In particular, constraints (3.3) and (3.4) define facets of the feasible polytope for $(\mathrm{PC}_{\mathrm{path}})$ [7]. The next section shows how a Lagrangian relaxation leads to a reformulation for TUFLP-S that dominates $(\mathrm{PC}_{\mathrm{path}})$ when both are applicable.

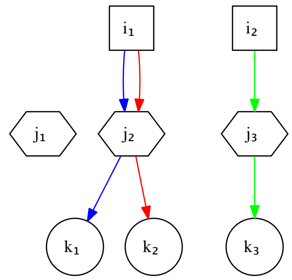


Figure 3.2: A TUFLP-S solution.

3.2 MIP formulations for the TUFLP-S

We wish to solve a variant of the TUFLP-G that forces each satellite to be linked with at most one depot; Fig. 3.2 shows a solution that satisfies this constraint. For some instances, solutions to (PG_{path}) will naturally satisfy single sourcing. In Chardaire, Lutton, and Sutter [14], this lead to a trivial reduction to TUFLP. We instead exploit the single sourcing constraint to derive a novel formulation for TUFLP-S.

Formulation (PG_{path}) includes variables t_{ij} to determine whether each arc from depot $i \in I$ to satellite $j \in J$ is open. We force each satellite to take at most one arc with

$$\sum_{i \in I} t_{ij} \leq 1, \quad \forall j \in J,$$

and obtain formulation (PS_{weak}).

$$(\text{PS}_{\text{weak}}) \qquad \min_{y,z,t,x} \sum_{i \in I} f_i y_i + \sum_{j \in J} g_j z_j + \sum_{(i,j) \in I \times J} h_{ij} t_{ij} + \sum_{(i,j,k) \in I \times J \times K} c_{ijk} x_{ijk}$$

subject to

$$\sum_{(i,j) \in I \times J} x_{ijk} = 1, \qquad \forall k \in K,$$

$$\sum_{i \in I} t_{ij} \leq 1, \qquad \forall j \in J, \tag{3.5}$$

$$x_{ijk} \leq t_{ij}, \qquad \forall (i,j,k) \in I \times J \times K, \tag{3.6}$$

$$\sum_{j \in J} x_{ijk} \leq y_i, \qquad \forall (i,k) \in I \times K,$$

$$\sum_{i \in I} x_{ijk} \leq z_j, \qquad \forall (j,k) \in J \times K, \tag{3.7}$$

$$0 \leq x_{ijk} \leq 1, \qquad \forall (i,j,k) \in I \times J \times K,$$

$$y_i \in \{0,1\}, \qquad \forall i \in I,$$

$$z_j \in \{0,1\}, \qquad \forall j \in J,$$

$$t_{ij} \in \{0,1\}, \qquad \forall (i,j) \in I \times J.$$

We improve (PS_{weak}) with the reformulation guided by relaxing all but the domain constraints and constraints (3.5), (3.6), and (3.7). The subproblem for that relaxation decomposes into one subproblem for each satellite $j \in J$:

$$
\begin{aligned}
\sum_{i \in I} t_{ij} &\leq 1, && \\
x_{ijk} &\leq t_{ij}, && \forall (i,k) \in I \times K, \\
\sum_{i \in I} x_{ijk} &\leq z_j, && \forall k \in K, \\
0 \leq x_{ijk} &\leq 1, && \forall (i,k) \in I \times K, \\
z_j &\in \{0,1\}, && \\
t_{ij} &\in \{0,1\}, && \forall i \in I.
\end{aligned}
$$

There are few feasible assignments for t_{ij}: at most one can be equal to 1. Moreover, given nonnegative fixed costs, z_j is positive iff some t_{ij} is. Either every variable is fixed to 0, or z_j and exactly one t_{ij} are fixed to 1, and the remaining variables to 0.

Once we fix that assignment, the restricted (decomposed) subproblem is a trivial linear program. When all binary variables are 0, so are path variables x_{ijk}. Otherwise, let $i^* \in I$ be the depot such that $t_{i^* j} = 1$; after simplification, the restricted subproblem is

$$
\begin{aligned}
0 \leq x^{i^*}_{i^* jk} &\leq 1, && \forall k \in K, \\
x^{i^*}_{ijk} &= 0, && \forall (i,k) \in I \times K, i \neq i^*.
\end{aligned}
$$

Lagrangian-informed reformulation combines these restricted subproblems into

$$
\begin{aligned}
\theta^0 + \sum_{i \in I} \theta^i &= 1, && && (3.8) \\
0 \leq x_{ijk} &\leq \theta^i, && \forall (i,k) \in I \times K, && \\
z_j &= \sum_{i \in I} \theta^i, && && (3.9) \\
t_{ij} &= \theta^i, && \forall i \in I, && (3.10) \\
\theta^i &\in \{0,1\}, && \forall i \in I \cup \{0\}. &&
\end{aligned}
$$

This reformulation does not duplicate continuous decision variables: each path variable x_{ijk} appears in exactly one restricted subproblem.

We simplify the reformulated subproblems further. Weight variable θ^0 serves no purpose except to allow not to choose any other θ^i; we express that possibility by converting constraint (3.8) into an inequality. Constraints (3.10) now show that the remaining weight variables θ^i correspond directly to t_{ij}. We insert the simplified reformulation in (PS_{weak}) to obtain

$$(\text{PS}^0_{\text{path}}) \qquad \min_{y,z,t,x} \sum_{i\in I} f_i y_i + \sum_{j\in J} g_j z_j + \sum_{(i,j)\in I\times J} h_{ij} t_{ij} + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}$$

subject to

$$\begin{aligned}
\sum_{(i,j)\in I\times J} x_{ijk} &= 1, && \forall k \in K,\\
\sum_{i\in I} t_{ij} &\le 1, && \forall j \in J,\\
x_{ijk} &\le t_{ij}, && \forall (i,j,k) \in I\times J\times K,\\
z_j &= \sum_{i\in I} t_{ij}, && \forall j \in J, \qquad (3.11)\\
\sum_{j\in J} x_{ijk} &\le y_i, && \forall (i,k) \in I\times K,\\
0 \le x_{ijk} &\le 1, && \forall (i,j,k) \in I\times J\times K,\\
y_i &\in \{0,1\}, && \forall i \in I,\\
z_j &\in \{0,1\}, && \forall j \in J,\\
t_{ij} &\in \{0,1\}, && \forall (i,j) \in I\times J.
\end{aligned}$$

We can eliminate variables z_j: constraints (3.11) mean that we can attribute the fixed costs g_j for variables z_j to every weight variable θ^i, or, equivalently, to every link variable t_{ij}. This yields the even simpler formulation (PS_{path}).

$$
(\mathrm{PS}_{\mathrm{path}}) \qquad \min_{y,t,x} \sum_{i\in I} f_i y_i + \sum_{(i,j)\in I\times J} (g_j + h_{ij}) t_{ij} + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}
$$

subject to

$$
\begin{aligned}
\sum_{(i,j)\in I\times J} x_{ijk} &= 1, & \forall k \in K,\\
\sum_{i\in I} t_{ij} &\leq 1, & \forall j \in J,\\
x_{ijk} &\leq t_{ij}, & \forall (i,j,k) \in I\times J\times K,\\
\sum_{j\in J} x_{ijk} &\leq y_i, & \forall (i,k) \in I\times K,\\
0 \leq x_{ijk} &\leq 1, & \forall (i,j,k) \in I\times J\times K,\\
y_i &\in \{0,1\}, & \forall i \in I\\
t_{ij} &\in \{0,1\}, & \forall (i,j) \in I\times J.
\end{aligned}
$$

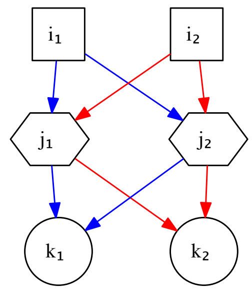
The linear relaxation of $(\mathrm{PS}_{\mathrm{path}})$ approximates the feasible set more tightly than that of $(\mathrm{PS}_{\mathrm{weak}})$, while comprising fewer constraints and variables. Intuitively, the effect of the reformulation is to attribute the fixed cost of each satellite $j \in J$ to every link to that satellite; single sourcing guarantees that at most one link will be open, and nonnegative fixed costs mean that we only open a satellite if we also open a link to that satellite.

In other words, we can obtain $(\mathrm{PS}_{\mathrm{path}})$ from $(\mathrm{PS}_{\mathrm{weak}})$ by adding the valid constraints

$$
z_j = \sum_{i\in I} t_{ij}, \quad \forall j \in J,
$$

and substituting z away. As shown in Fig. 3.3, optimal solutions to the linear relaxation of $(\mathrm{PS}_{\mathrm{weak}})$ may be such that

$$
z_j \lneqq \sum_{i\in I} t_{ij}
$$



We have origin–destination pairs $i_1 \to k_1$ and $i_2 \to k_2$. Let all fixed and transportation costs be 0, except for the use of satellites j_1 and j_2, with a fixed cost of 1.
One solution is to let a flow of $\frac{1}{2}$ pass through each possible path. In formulation (PS_{weak}), only half the fixed cost for j_1 and j_2 will be incurred, for a total lower bound of 1. Formulation (PS_{path}) instead assigns half the fixed cost of satellite j_1 on arc $i_1 \to j_1$ and half on $i_2 \to j_1$, and similarly for satellite j_2 and arcs $i_1 \to j_2$ and $i_2 \to j_2$; the bound is then 2. Optimal integer solutions are also optimal in the linear relaxation.

Figure 3.3: A TUFLP-S instance for which (PS_{path}) rules out a fractional solution.

for some $j \in J$, and this simple modification strengthens the linear relaxations.

The Lagrangian-informed reformulation is less intuitive, but helps characterise the improvement to the linear relaxation: it stems from the exact description of the convex hulls for its integer Lagrangian subproblems.

When solving formulation (PS_{path}) to optimality, constraints

$$t_{ij} \leq y_i, \quad \forall (i, j) \in I \times J$$

may be useful. They do not affect the linear relaxation bound but let us relax the integrality constraint on variables y_i. Adding these constraints yields (PS'_{path}), a formulation with more constraints but fewer binary variables than (PS_{path}).

$$
(\text{PS}'_{\text{path}}) \qquad \min_{y,t,x} \sum_{i \in I} f_i y_i + \sum_{(i,j) \in I \times J} (g_j + h_{ij}) t_{ij} + \sum_{(i,j,k) \in I \times J \times K} c_{ijk} x_{ijk}
$$

subject to

$$
\begin{aligned}
\sum_{(i,j) \in I \times J} x_{ijk} &= 1, && \forall k \in K, \\
\sum_{i \in I} t_{ij} &\leq 1, && \forall j \in J, && (3.12) \\
x_{ijk} &\leq t_{ij}, && \forall (i,j,k) \in I \times J \times K, && (3.13) \\
\sum_{j \in J} x_{ijk} &\leq y_i, && \forall (i,k) \in I \times K, \\
t_{ij} &\leq y_i, && \forall (i,j) \in I \times J, \\
0 \leq x_{ijk} &\leq 1, && \forall (i,j,k) \in I \times J \times K, \\
0 \leq y_i &\leq 1, && \forall i \in I, \\
t_{ij} &\in \{0,1\}, && \forall (i,j) \in I \times J.
\end{aligned}
$$

These three formulations ($(\text{PS}^0_{\text{path}})$, $(\text{PS}_{\text{path}})$, $(\text{PS}'_{\text{path}})$) are all equivalent when relaxed into linear programs. They differ in the binary variables that remain: $(\text{PS}^0_{\text{path}})$ preserves all the decision variables in $(\text{PS}_{\text{weak}})$, $(\text{PS}_{\text{path}})$ eliminates variables z_j, and $(\text{PS}'_{\text{path}})$ further relaxes the integrality of variables y_i.

Their performance when solved as mixed integer programs may thus vary: is it better to mimic the initial formulation closely (*i.e.*, $(\text{PS}^0_{\text{path}})$), to minimise the number of binary variables (*i.e.*, $(\text{PS}'_{\text{path}})$), or to strike a compromise between these two extremes and only eliminate variables that appear solely in subproblems (*i.e.*, $(\text{PS}_{\text{path}})$)?

3.3 Numerical results

This chapter describes two mixed integer formulations for TUFLP-C ((PG$_{\text{path}}$) and (PC$_{\text{path}}$)) and four novel ones for TUFLP-S ((PS$_{\text{weak}}$) (PS$_{\text{path}}$), (PS$^0_{\text{path}}$) and (PS$'_{\text{path}}$)). We compare all six formulations thanks to standard TUFLP-C instances that we can also solve as TUFLP-S.

First, we streamline the comparison. (PC$_{\text{path}}$) dominates both (PG$_{\text{path}}$) and (PS$_{\text{weak}}$) on instances that we can solve as both TUFLP-S and TUFLP-C: their linear relaxations are equivalent, but (PC$_{\text{path}}$) comprises fewer constraints and variables.

In the general case, we can only solve TUFLP-S with formulations (PS$_{\text{weak}}$), (PS$_{\text{path}}$), (PS$^0_{\text{path}}$) or (PS$'_{\text{path}}$). On these instances, (PS$^0_{\text{path}}$) dominates (PS$_{\text{weak}}$): the linear relaxation of the former is provably stronger and they have the same integer variables.

We will thus only compare four formulations: (PC$_{\text{path}}$), (PS$_{\text{path}}$), (PS$^0_{\text{path}}$) and (PS$'_{\text{path}}$). We mentioned another formulation: Landete and Marín [58] adds facet-defining inequalities to (PC$_{\text{path}}$) to obtain (PC$_{\text{hole}}$). This last formulation provides another comparison basis for instances that we solve as both TUFLP-C and TUFLP-S. However, the constraints are complicated, and, rather than re-implementing the constraint generator, we solved the same instances as in Landete and Marín [58] and report integrality gaps at the root, exactly as they appear in Tables 1 to 3 in Landete and Marín [58].

Section 3.3.1 studies the performance of all five formulations on small artificial instances of the TUFLP-C that we also solve as TUFLP-S. Formulations (PS$_{\text{path}}$), (PS$^0_{\text{path}}$) and (PS$'_{\text{path}}$) will yield tighter lower bounds at the root than (PC$_{\text{path}}$); it is interesting to compare the improvement in lower bound with that achieved by (PC$_{\text{hole}}$).

Numerical experiments also help evaluate the reformulations as integer programs. Comparing formulations (PS$^0_{\text{path}}$), (PS$_{\text{path}}$), and (PS$'_{\text{path}}$) is especially useful: they all share equivalent linear relaxations, but (PS$^0_{\text{path}}$) preserves all binary variables in the initial formulation (PS$_{\text{weak}}$), (PS$_{\text{path}}$) eliminates variables that appear only in the relaxed subproblem, and (PS$'_{\text{path}}$) reduces binary variables to a bare minimum.

Section 3.3.2 compares (PS$^0_{\text{path}}$), (PS$_{\text{path}}$), and (PS$'_{\text{path}}$), the three novel formulations, when solving industrial instances of the TUFLP-S. These instances appeared as subproblems when solving Lagrangian decomposition for the more complex industrial problem of Chapter 5. Their integrality gap at the root is small, but the instances are larger than the ones in Section 3.3.1, and solving them is challenging.

All the tables report computations performed on a 2.9 GHz E5-4617 with 128 GB of DDR3-1600 RAM and with CPLEX 12.5 [2] in single-threaded mode, but otherwise default settings. The only exceptions are the results for (PC_{hole}): we copied them from Landete and Marín [58], and the results correspond to a less powerful computer and an older version of CPLEX. We only report linear relaxation bounds at the root for this formulation.

When solving integer programs, we provided as initial upper bounds the optimal values in order to control the influence of primal heuristics. We also exploited the similarity between (PS^0_{path}) and (PC_{path}) by giving branching priority to variables y_i and z_j. For TUFLP-C instances, (PS^0_{path}) is a variant of (PC_{path}) with improved lower bounds.

We only present synthetic results and relegate complete tables of results to Appendices I to V.

3.3.1 Two-level Gap instances

In addition to characterising new facets of the (PC_{path}) polytope, Landete and Marín [58] describes a procedure to convert a set of classical UFLP instances, Gap [56], into TUFLP-C instances with transportation costs that are decomposable by arc. This results in two-level Gap instances that comprise 50 depot locations, 50 satellite locations and 50 customers. Fixed costs for depots and satellites are constant (3000), and only paths and transportation costs vary (between 0 and 8). The cost of each path ijk is the sum of the costs for the arcs from depot i to satellite j and from j to customer k; this cost structure guarantees that at least one optimal solution to (PC_{path}) satisfies single sourcing.

There are minor (apparently unintended) interactions between the conversion proce-

2. IBM CPLEX (http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/) is currently available under a free academic license through the IBM Academic Initiative (http://www-03.ibm.com/ibm/university/academic/pub/page/ban_ilog_programming).

dure and asymmetry in the Gap instances [3], but only one issue seems important. Gap instances are sparse and assign artificial "big-M" costs to inexistant arcs. However, the conversion procedure disregards that sparsity and the artificial costs are low enough that some optimal TUFLP solutions use forbidden arcs. This affects 26 of the 90 instances, and we eliminated them from consideration. We solved the rest on sparse graphs with $(\text{PC}_{\text{path}})$, $(\text{PS}^0_{\text{path}})$, $(\text{PS}_{\text{path}})$ and $(\text{PS}'_{\text{path}})$.

There are three sets of Gap instances for the UFLP (A, B and C), each generated randomly with different constraints on the distribution of degrees for depots and customers. In general, Gap B instances seem easier (the integrality gap is lower at the root), and Gap C instances more difficult (the gap is higher at the root) [56]. These sets give rise to three sets of TUFLP instances (A, B and C) that exhibit the same difficulty pattern.

Table 3.I summarises the strength and efficiency of formulations $(\text{PC}_{\text{path}})$, $(\text{PS}^0_{\text{path}})$, $(\text{PS}_{\text{path}})$, $(\text{PS}'_{\text{path}})$ and $(\text{PC}_{\text{hole}})$. At the root, $(\text{PS}^0_{\text{path}})$, $(\text{PS}_{\text{path}})$ and $(\text{PS}'_{\text{path}})$ are completely equivalent; we only report a single gap value.

For all formulations, root computation times are negligible (on the order of a tenth of a second or less), and we report none. In contrast, when solved as integer programs, only runtimes matter (every method solves all instances to optimality), and we report runtimes to optimality for all mixed integer formulations, except for $(\text{PC}_{\text{hole}})$: Landete and Marín [58] report disappointing runtimes. We believe the reason is that they represent every 50^3 path in the formulation, regardless of forbidden arcs; executing an earlier version of CPLEX on older hardware only worsens matters.

We computed integrality gaps as the difference between the lower bound bound and the optimal (up to a relative error of 10^{-4}) integer value, divided by the latter. Branch-and-bound times correspond to the default branch-and-cut search of CPLEX 12.5 with the default stopping criterion: the solution must be optimal up to a relative error of 10^{-4} or less.

These figures are coarse, but telling: formulations of the $(\text{PS}_{\text{path}})$ family results in more accurate lower bounds than $(\text{PC}_{\text{path}})$, particularly on more difficult instances (Gap

3. The algorithm in Kochetov and Ivanenko [56] generates specific sparsity patterns without guaranteeing that their distribution is independent of the index of depots and customers. An odd/even partition (instead of high/low) would likely better preserve the characteristics of Gap instances.

Instance set	n	Root integrality gap (average %) $(\text{PC}_{\text{path}})$	$(\text{PS}_{\text{path}})$	$(\text{PC}_{\text{hole}})$
Gap A	26	11.39	10.85	10.55
Gap B	13	7.63	7.45	7.36
Gap C	25	13.20	12.59	12.28

Instance set	n	Branch-and-bound time (average sec) $(\text{PC}_{\text{path}})$	$(\text{PS}_{\text{path}})$	$(\text{PS}'_{\text{path}})$	$(\text{PS}^0_{\text{path}})$
Gap A	26	1.27	5.35	3.98	2.19
Gap B	13	0.86	3.13	2.15	1.37
Gap C	25	2.43	28.42	11.01	4.60

Table 3.I: Summary of formulation performances for Gap A, B, and C TUFLP-C instances

A and C), and $(\text{PC}_{\text{hole}})$ does even better. The performance profiles [24] in Fig. 3.4 confirm this impression. In practice, formulation $(\text{PS}_{\text{path}})$ is reliably stronger than $(\text{PC}_{\text{path}})$ and seems competitive with $(\text{PC}_{\text{hole}})$, especially for Gap B instances. Overall, $(\text{PS}_{\text{path}})$ and related formulations strike a practical balance between the size and complexity of the formulations and the strength of their linear relaxation bound.

Our new formulations fail to improve runtimes for branch-and-bound searches: Gap instances for the TUFLP are small, and the integrality gap at the root is wide, at least 5% to 10%. In such circumstances, the best strategy will be to branch rapidly and explore the search tree as quickly as possible. That is exactly what solving $(\text{PC}_{\text{path}})$ achieves.

In the same vein, runtimes for $(\text{PS}_{\text{path}})$ and $(\text{PS}'_{\text{path}})$ show that reducing the number of binary variables helps improve runtimes in such situations. However, formulation $(\text{PS}^0_{\text{path}})$ performs even better by instead preserving all discrete decision variables in the initial formulation: its runtimes are much closer to those of $(\text{PC}_{\text{path}})$.

Again, the performance profiles in Fig. 3.5 confirm these impressions: $(\text{PC}_{\text{path}})$ always leads to the quickest branch-and-bound times, solving $(\text{PS}'_{\text{path}})$ take about half as much time as $(\text{PS}_{\text{path}})$, and $(\text{PS}^0_{\text{path}})$ comes even closer to formulation $(\text{PC}_{\text{path}})$.

The results on Gap instances show that $(\text{PS}^0_{\text{path}})$ improves the linear relaxation bound compared to $(\text{PC}_{\text{path}})$ (which is equivalent to the base formulation $(\text{PG}_{\text{path}})$) in practice. However, the reformulation is larger, and the trade-off is unproductive when solving small artificial instances. Larger practical instances will be more affected by the lower

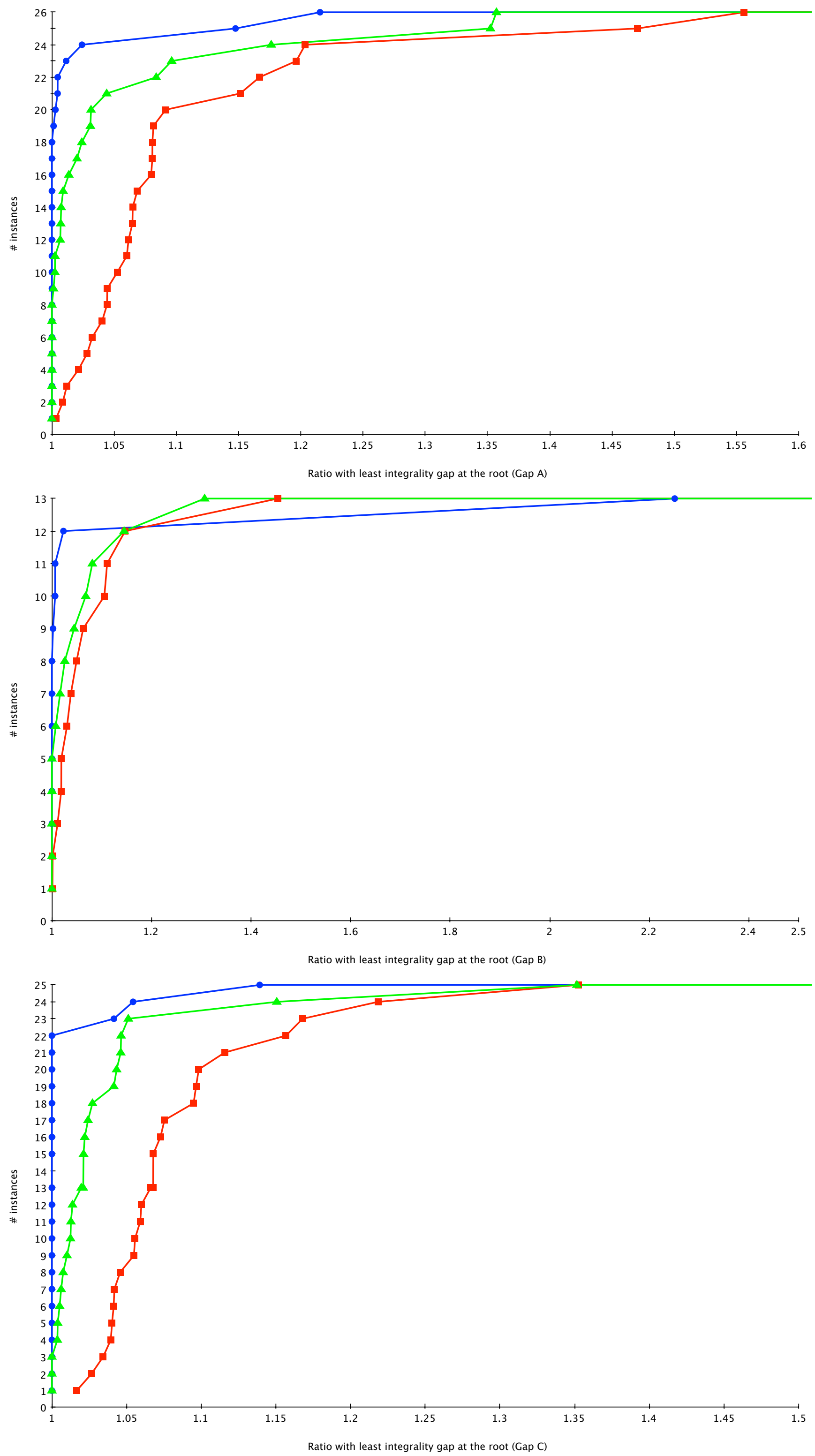


Figure 3.4: Performance profiles for integrality gaps at the root of TUFLP-S instances Gap A, B, and C (blue circles: $(\mathrm{PC_{hole}})$; green triangles: $(\mathrm{PS_{path}})$; red squares: $(\mathrm{PC_{path}})$).

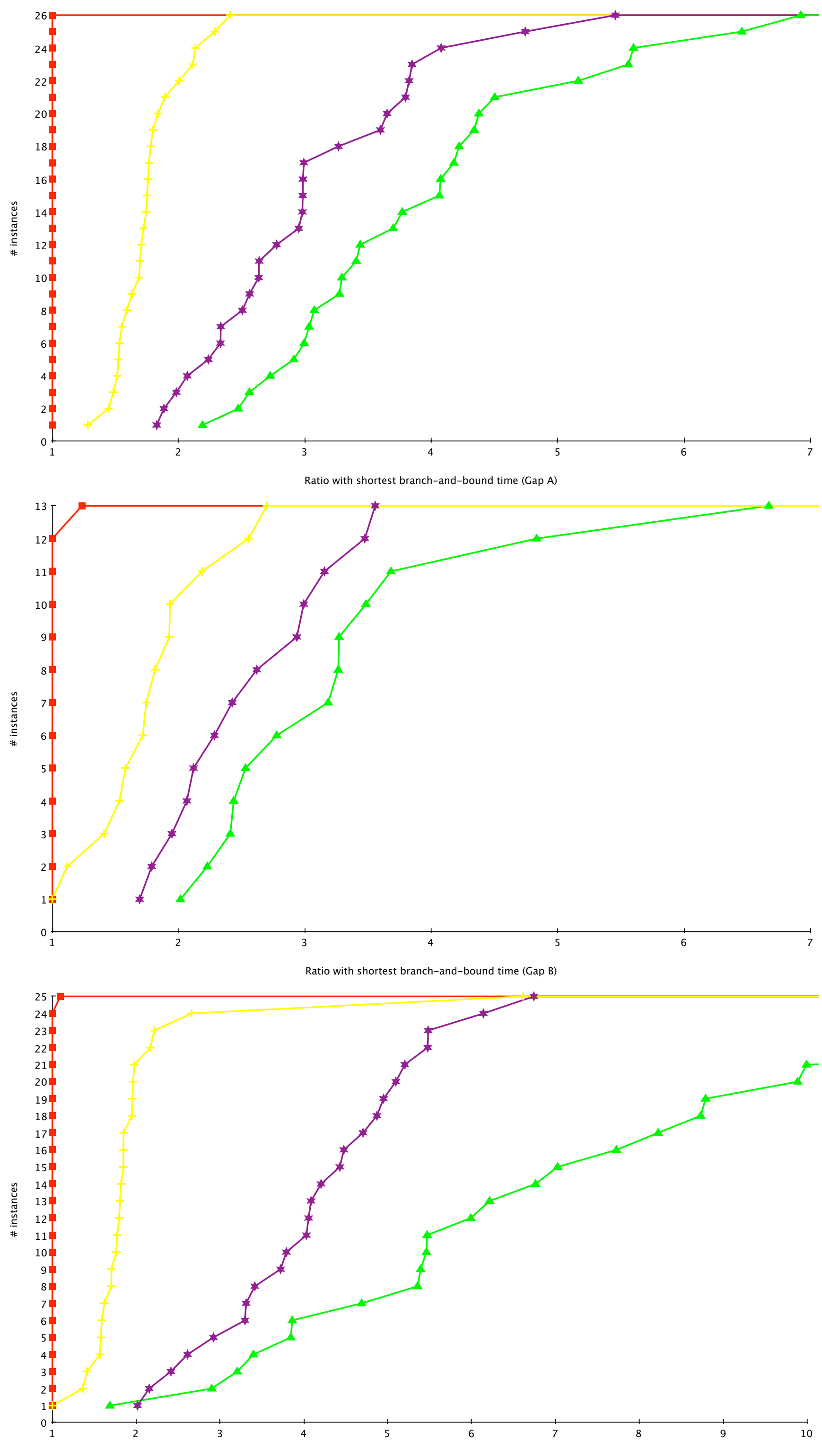


Figure 3.5: Performance profiles for runtimes to solve TUFLP-S instances Gap A, B, and C to optimality (red squares: (PC_{path}); yellow crosses: (PS^0_{path}); purple stars: (PS'_{path}); green triangles: (PS_{path})).

Name	n	# depots	# satellites	# depot–satellite arcs	# customers	# paths
$\frac{1}{4}$	100	23	80	175	134	3366
$\frac{1}{2}$	100	47	160	351	592	28496
$\frac{3}{4}$	100	70	240	526	1236	92554
Full	100	93	320	701	2250	222308

Table 3.II: Characteristics of the four industrial TUFLP-S instance sets

bound than by the size of the reformulation and should favour ($\text{PS}^0_{\text{path}}$): branch-and-bound for that reformulation is only twice as slow as for $(\text{PC}_{\text{path}})$ on small artificial instances.

3.3.2 Industrial TUFLP-S instances

We designed formulations ($\text{PS}^0_{\text{path}}$), ($\text{PS}_{\text{path}}$) and ($\text{PS}'_{\text{path}}$) to solve instances of the TUFLP-S, and this subsection reports their performance on four sets of 100 instances derived from the industrial problem of Chapter 5.

The instances appeared as subproblems during preliminary development of a Lagrangian decomposition. Each set corresponds to an industrial instance from which 100 TUFLP-S instances inherit their network structure. We list their characteristics in Table 3.II; instances in set $\frac{1}{4}$ are roughly one fourth the size of instances in set Full, those in $\frac{1}{2}$ half the size, and those in $\frac{3}{4}$ three quarters. These TUFLP-S are defined on more realistic graphs than Gap instances, but present a cost structure that makes them particularly difficult: depots all incur the same large fixed cost, while satellites and depot–satellite arcs incur none. Moreover, transportation costs vary greatly depending on the locations on the path, and the single sourcing constraints are not redundant.

All three formulations are equivalent when solved as linear programs; we only report integrality gaps at the root for (PS_{path}) in Table 3.III. However, their runtimes may differ, particularly in branch-and-bound, and we report all runtimes for root relaxations and branch-and-bound.

The linear relaxation bounds at the root are overall close to the optimal integer values; Fig. 3.6 shows that integrality gaps at the root are always lower than 1%.

The linear relaxations of ($\text{PS}^0_{\text{path}}$) and ($\text{PS}'_{\text{path}}$) are solved slightly more quickly than

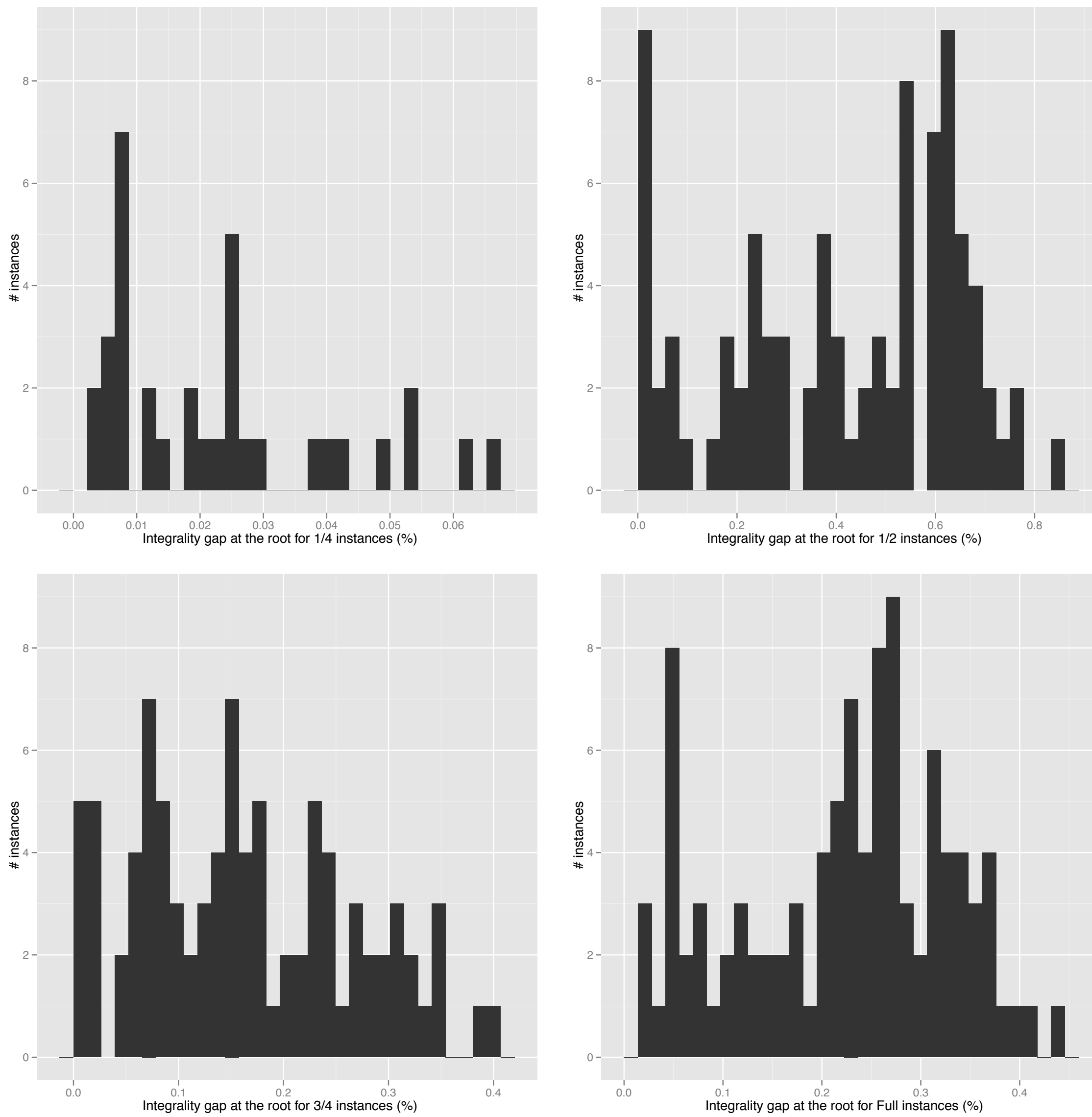


Figure 3.6: Distribution of integrality gaps at the root for industrial TUFLP-S instances

Set	n	Root LP time (average sec) (PS_{path})	(PS'_{path})	(PS^0_{path})	Root gap (average %)
$\frac{1}{4}$	100	0.03	0.03	0.03	0.01
$\frac{1}{2}$	100	0.53	0.54	0.47	0.43
$\frac{3}{4}$	100	3.71	3.50	2.88	0.15
Full	100	23.00	20.22	21.61	0.22

Set	n	Total MIP time (average sec) (PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{4}$	100	0.06	0.06	0.06
$\frac{1}{2}$	100	3.66	4.44	2.96
$\frac{3}{4}$	100	16.83	17.63	15.25
Full	100	178.32	234.65	139.49

Table 3.III: Summary of formulations performance for industrial TUFLP-S instances

that of (PS_{path}), despite the additional constraints (Fig. 3.7).

The advantage of formulation (PS^0_{path}), which preserves all the binary decision variables from the natural formulation (PS_{weak}), grows wider when solved to optimality (Fig. 3.8). For industrial TUFLP-S instances as well, it is preferable to preserve redundant binary decision variables.

3.4 Summary

Our experiments on hard artificial instances and on more approachable practical ones lead to two results:

1. Lagrangian-informed reformulation can improve linear relaxations in practice with a small increase in size.
2. It is preferable to preserve all integer decision variables in the initial formulation.

The second conclusion is common knowledge for branch-and-price methods [6, 23, 61]. We only confirm that it is still true for Lagrangian-informed reformulations.

The experiments also reveal a first novel formulation: (PS^0_{path}). (PS^0_{path}) improves the linear bound of the classical (PC_{path}) and preserves its decision variables. It is already comparable to (PC_{path}) on small artificial instances, and we hope that it will be useful for larger but less difficult instances.

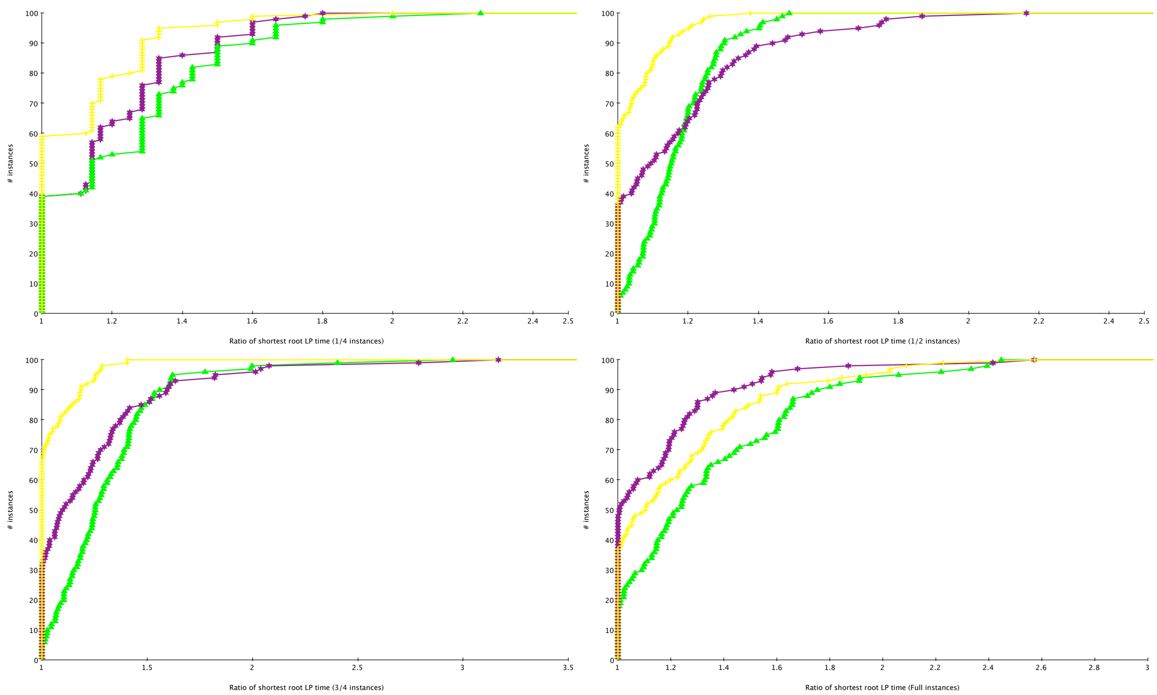


Figure 3.7: Performance profiles of root relaxation solving times for industrial TUFLP-S instances (yellow crosses: $(\mathrm{PS}^0_{\mathrm{path}})$; purple stars: $(\mathrm{PS}'_{\mathrm{path}})$; green triangles: $(\mathrm{PS}_{\mathrm{path}})$).

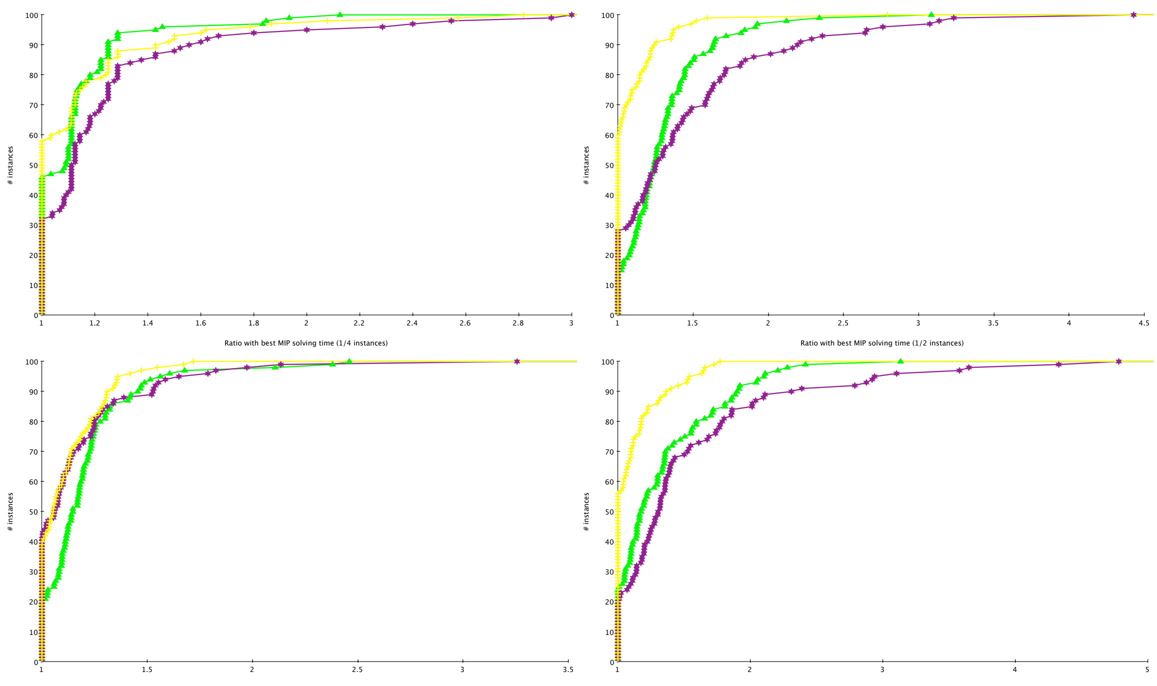


Figure 3.8: Performance profiles for runtimes to solve industrial TUFLP-S instances to optimality (yellow crosses: (PS^0_{path}); green triangles: (PS_{path}); purple stars: (PS'_{path})).

CHAPTER 4

A SECOND LAGRANGIAN-INFORMED REFORMULATION FOR TUFLP-S

This chapter presents a second novel formulation for the TUFLP-S. The formulation is a Lagrangian-informed reformulation that is well adapted to our larger industrial instances. We use the Lagrangian relaxation proposed in Gendron et al [35], which dualises the constraints that link depot–satellite arcs and customers. This preserves the core of our industrial instances: they only impose fixed costs on the use of depots. Section 4.2 shows that the reformulation leaves no integrality gap on these industrial instances.

The next section describes the reformulation in details and Section 4.2 summarises its performance. For industrial TUFLP-S instances, our novel reformulation leads to a Lagrangian heuristic that is comparable to branch-and-cut on ($\mathrm{PS}^0_{\mathrm{path}}$).

4.1 A Lagrangian relaxation for TUFLP-S

Let λ be the Lagrange multipliers associated with constraints (3.13). Relaxing the latter constraints mechanically in formulation ($\mathrm{PS}'_{\mathrm{path}}$) yields the following subproblem.

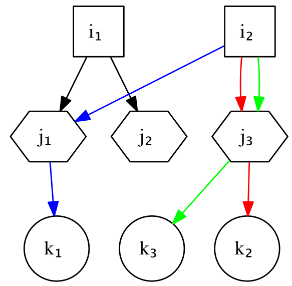


Figure 4.1: Solution to a TUFLP-S instance without constraints (3.13)

$$\min_{y,t,x} \sum_{i\in I} f_i y_i + \sum_{(i,j)\in I\times J} \left(g_j + h_{ij} - \sum_{k\in K} \lambda_{ijk} \right) t_{ij} + \sum_{(i,j,k)\in I\times J\times K} (c_{ijk} + \lambda_{ijk}) x_{ijk}$$

subject to

$$\sum_{(i,j)\in I\times J} x_{ijk} = 1, \qquad \forall k \in K,$$
$$\sum_{i\in I} t_{ij} \leq 1, \qquad \forall j \in J,$$
$$\sum_{j\in J} x_{ijk} \leq y_i, \qquad \forall (i,k) \in I\times K,$$
$$t_{ij} \leq y_i, \qquad \forall (i,j) \in I\times J,$$
$$0 \leq x_{ijk} \leq 1, \qquad \forall (i,j,k) \in I\times J\times K,$$
$$0 \leq y_i \leq 1, \qquad \forall i \in I,$$
$$t_{ij} \in \{0,1\}, \qquad \forall (i,j) \in I\times J.$$

This Lagrangian subproblem is a variation of the classical UFLP: the subproblem includes satellites whose demands may be satisfied partially. Figure 4.1 sketches a solution to one such subproblem.

We can simplify the previous formulation: for any pair of depot $i \in I$ and customer

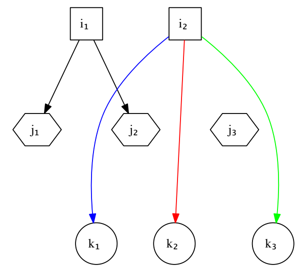


Figure 4.2: Solution to a simplified TUFLP-S instance without constraints (3.13)

$k \in K$, all paths in $\{(i,j,k) \mid j \in J\}$ are equivalent, except for their cost. We exploit this by only considering a single least-cost representative for each of these sets. This simplification yields formulation (SUBP). Figure 4.2 outlines the simplified solution of Fig. 4.1.

$$\text{(SUBP)}\quad \min_{y,t,\tilde{x}} \sum_{i \in I} f_i y_i + \sum_{(i,j) \in I \times J} \left(g_j + h_{ij} - \sum_{k \in K} \lambda_{ijk} \right) t_{ij} + \sum_{(i,k) \in I \times K} \left(\min_{j \in J} c_{ijk} + \lambda_{ijk} \right) \tilde{x}_{ik}$$

subject to

$$\begin{aligned}
\sum_{i \in I} \tilde{x}_{ik} &= 1, && \forall k \in K, \\
\sum_{i \in I} t_{ij} &\leq 1, && \forall j \in J, \\
\tilde{x}_{ik} &\leq y_i, && \forall (i,k) \in I \times K, \\
t_{ij} &\leq y_i, && \forall (i,j) \in I \times J, \\
0 \leq \tilde{x}_{ik} &\leq 1, && \forall (i,k) \in I \times K, \\
y_i &\in \{0,1\}, && \forall i \in I, \\
0 \leq t_{ij} &\leq 1, && \forall (i,j) \in I \times J,
\end{aligned}$$

where λ are the Lagrange multipliers for constraints (3.13), and each $\tilde{x}_{ik}$ is a representative for path variables $\{x_{ijk} \mid j \in J\}$.

This second formulation for the Lagrangian subproblem is strictly equivalent to the previous one: if $\tilde{x}_{i^*k^*} = 1$ in the compact formulation, we set $x_{i^*j^*k^*}$ to 1 in the original formulation, with

$$j^* = \arg\min_{j \in J} c_{i^*jk^*} + \lambda_{i^*jk^*}$$

and arbitrary tiebreaking. The only difference is that (SUBP) is more compact: the number of variables is quadratic, rather than cubic, in the number of locations.

Even after simplification, the Lagrangian subproblem is NP-hard; the reductions to and from the NP-hard UFLP [57] are trivial [1]. However, branch-and-bound trees are usually small for path-based formulations like (SUBP) [57].

The relaxation is also well suited to Lagrangian-informed reformulation: once we restrict the Lagrangian subproblem to $y = s$, it becomes

$$\begin{aligned}
\sum_{i \in I} x_{ijk} &= 1, & \forall (j,k) \in J \times K, \\
\sum_{i \in I} t_{ij} &\leq 1, & \forall j \in J, \\
0 \leq x_{ijk} &\leq s_i, & \forall (i,j,k) \in I \times J \times K, \\
0 \leq t_{ij} &\leq s_i, & \forall (i,j) \in I \times J.
\end{aligned}$$

This Lagrangian relaxation defines formulation (PS_{hull}).

1. We associate satellites to an artificial depot at no cost instead of leaving them unconnected.

$$(\text{PS}_{\text{hull}}) \qquad \min_{y,t,x} \sum_{i\in I} f_i y_i + \sum_{(i,j)\in I\times J} (g_j + h_{ij}) t_{ij} + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}$$

subject to

$$
\begin{aligned}
x_{ijk} &\le t_{ij}, && \forall (i,j,k) \in I\times J\times K,\\
y_i &= \sum_{s\in Y} s_i \theta^s, && \forall i \in I,\\
t_{ij} &= \sum_{s\in Y} t_{ij}^s && \forall (i,j) \in I\times J,\\
x_{ijk} &= \sum_{s\in Y} x_{ijk}^s && \forall (i,j,k) \in I\times J\times K,\\
\sum_{i\in I} x_{ijk}^s &= \theta^s && \forall (j,k) \in J\times K, \forall s \in Y,\\
\sum_{i\in I} t_{ij}^s &\le \theta^s && \forall j \in J, \forall s \in Y,\\
0 \le x_{ijk}^s &\le s_i \theta^s && \forall (i,j,k) \in I\times J\times K, \forall s \in Y,\\
0 \le t_{ij}^s &\le s_i \theta^s && \forall (i,j) \in I\times J, \forall s \in Y,\\
y_i &\in \{0,1\}, && \forall i \in I,\\
0 \le t_{ij} &\le 1, && \forall (i,j) \in I\times J,\\
0 \le x_{ijk} &\le 1, && \forall (i,j,k) \in I\times J\times K,\\
\theta^s &\in \{0,1\}, && \forall s \in Y,
\end{aligned}
$$

where $Y = \mathbb{B}^{|I|}$, the set of potential value assignments for y.

(PS_{hull}) is large, and we propose to optimise only its linear relaxation with Algorithms 3 and 4 (Section 2.3) and to solve each Lagrangian subproblem with (SUBP).

We can also turn the TUFLP-S into a single-level facility location problem by determining which depots are closed and open ahead of time: restricting $y = s$ leaves a location problem that we solve directly with the default branch-and-cut solver of CPLEX.

Each iteration of our structured Dantzig-Wolfe decomposition (Algorithms 3 and 4) generate such an assignment; we implement a Lagrangian heuristic by solving a restricted TUFLP-S subproblem (with $y = \bar{y}$) whenever the bundle $\mathscr{B}$ grows.

4.2 Numerical results

This section reports numerical results for structured (stabilised) Dantzig-Wolfe decomposition (S$^{(2)}$DW) of our novel formulation for the TUFLP-S. We compare them with bundle and volume algorithms on the Lagrangian dual defined by (SUBP); we also compare the S$^{(2)}$DW Lagrangian heuristic with a branch-and-cut solver for ($\mathrm{PS}^0_{\mathrm{path}}$).

The first subsection describes the performance of the Lagrangian heuristics and of their components on the Gap instances described in Section 3.3.1, small and difficult artificial TUFLP-C instances. We dedicate the second subsection to the larger and more realistic TUFLP-S instances introduced in Section 3.3.2.

4.2.1 Two-level Gap instances

There are two interesting questions when attempting to understand the lower bounding algorithms described in this chapter: how strong the resulting bounds are in practice, and how much computational effort is necessary to reach these bounds.

Table 4.I reports average integrality gaps for instance sets Gap A, B and C for six bounding methods.

($\mathrm{PC}_{\mathrm{hole}}$) corresponds to the linear relaxation of the strengthened formulation in Landete and Marín [58], ($\mathrm{PS}'_{\mathrm{path}}$) to the linear relaxation of ($\mathrm{PS}'_{\mathrm{path}}$) and (SUBP) to solving the Lagrangian subproblem once, with multipliers from ($\mathrm{PS}'_{\mathrm{path}}$).

The next three columns report gaps for the quadratically stabilised bundle method [48] (BTT 2.12[2] [28] by Antonio Frangioni), for the volume algorithm [5] (version 1.4 of the COIN-OR implementation[3]), and for our unstabilised structured Dantzig-Wolfe decomposition (Algorithm 3). All three methods maximised the same Lagrangian

2. An older version is available at `http://sorsa.unica.it/it/software.php`; we obtained BTT 2.12 by personal email.
3. The project homepage at `https://projects.coin-or.org/Vol` trails development; we checked out version 1.4 from the repository (`svn co https://projects.coin-or.org/svn/Vol/stable/1.4 coin-Vol`).

Set	n	Average relative gap with optimum (%)						
		(PC_{hole})	(PS'_{path})	(SUBP)	Bundle	Volume	(PS_{hull})	Primal
Gap A	26	10.55	10.85	10.84	6.25	6.40	6.25	0.01
Gap B	13	7.36	7.45	7.44	3.52	3.63	3.51	0.01
Gap C	25	12.28	12.59	12.59	7.51	7.60	7.45	0.01

Table 4.I: Average integrality gaps for formulations (PC_{hole}), (PS'_{path}), the warm-started Lagrangian solution (SUBP), the Lagrangian dual solved with the bundle and volume algorithms, (PS_{hull}) solved as a structured decomposition, and the primal Lagrangian heuristic (Primal) on Gap A, B and C instances.

duals for up to two hours, after a warm start with multipliers from (PS'_{path}). In theory, they compute the same bound, but only the structured Dantzig-Wolfe decomposition of (PS_{hull}) always reaches optimal solutions within two hours.

The last column reports the average relative optimality gap for our primal Lagrangian heuristic guided by the unstabilised structured Dantzig-Wolfe decomposition (the choice of stabilisation term does not affect the heuristic).

The linear relaxation of (PS_{hull}) is stronger than that of (PC_{hole}) and (PS'_{path}) (Chapter 3), and the Lagrangian heuristic always computes nearly optimal solutions: the worst difference from optimum is 0.09%.

Disappointingly, detailed results in Appendices II to V show that the warm-started Lagrangian lower bound is always equal to the linear relaxation bound of (PS'_{path}). Nevertheless, these multipliers are useful initialisers.

The performance profiles in Fig. 4.3 confirm the general impressions above: the linear relaxation of (PS_{hull}) dominates those of (PS'_{path}) (as predicted by theory) and (PC_{hole}), some bundle and volume bounds are suboptimal after two hours, and the volume algorithm computes particularly weak bounds because it stops too early.

If we only consider lower bounds, this second Lagrangian-informed reformulation is a resounding success on Gap instances. However, the runtimes in Table 4.II are prohibitive: CPLEX 12.5 solved each of these instances in seconds with formulation (PC_{path}), as reported in Section 3.3.1.

Table 4.III reports runtimes for (stabilised) Dantzig-Wolfe structured decomposition and for the bundle and volume algorithms. Structured Dantzig-Wolfe decomposition

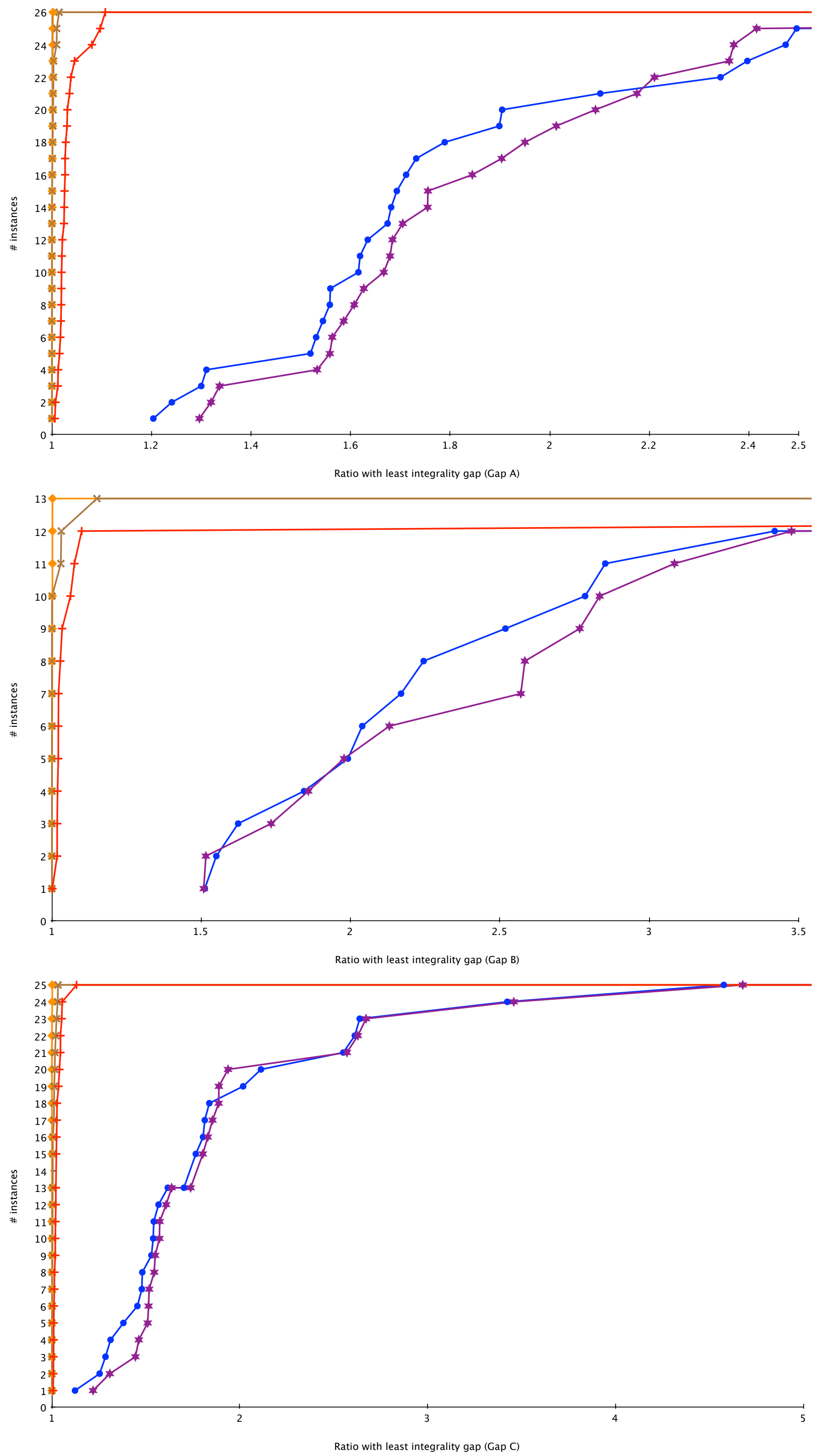


Figure 4.3: Performance profiles for the gap between lower bounds and the optimum value for TUFLP-C instances Gap A, B, and C (orange diamonds: (PS_{hull}); brown crosses: bundle; red crosses: volume; purple stars; blue circles: (PC_{hole}); purple stars:

Set	n	Average amount of time spent in each phase (sec)			
		(PS'_{path})	(SUBP)	(PS_{hull})	Primal
Gap A	26	0.10	0.08	1348.85	2.15
Gap B	13	0.09	0.08	871.48	1.70
Gap C	25	0.13	0.12	1529.54	2.22

Table 4.II: Average amount of computation time expended on the linear relaxation of (PS'_{path}), the initial Lagrangian subproblem ((SUBP)), the iterative algorithm for (PS_{hull}), and on the primal heuristic, for Gap A, B and C instances.

Set	n	Average amount of time to optimality (sec)					
				Stabilisation term			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
Gap A	26	6208.22	1078.97	1349.03	262.75	2585.13	396.12
Gap B	13	5562.59	781.59	871.65	246.69	1904.13	353.48
Gap C	25	6868.59	1903.46	1529.79	500.76	3883.15	753.73

Table 4.III: Average amount of computation time to optimality (up to 2 hours) for the bundle and volume master problems and for the structured Dantzig-Wolfe decomposition with and without stabilisation, on Gap A, B and C instances.

with an ℓ_1 constraint reached optimal solutions more quickly than all other methods, but was still hundreds of times slower than solving (PC_{path}) to optimality.

Results for the bundle and volume algorithms show another issue with generic methods for nondifferentiable optimisation: numerical convergence to nearly optimal solutions is not only slow, but also difficult to detect. Runtimes for the bundle algorithm are inflated by iterations that only serve to prove it has maximised the Lagrangian dual. The volume algorithm instead stops when too many iterations fail to improve the lower bound; this improves runtimes, but results in suboptimal bounds. In contrast, structured Dantzig-Wolfe decomposition implements a simple and practical criterion that reliably stops with optimal solutions.

4.2.2 Industrial TUFLP-S instances

Again, we present the results in two phases: first lower and upper bounds, then computational requirements.

Table 4.IV summarises the integrality gap for formulation (PS'_{path}), for the initial

Set	n	Average relative gap with optimum (%) (PS$'_{\text{path}}$)	(SUBP)	(PS$_{\text{hull}}$)	Primal
$\frac{1}{4}$	100	0.01	0.01	0.01	0.01
$\frac{1}{2}$	100	0.43	0.36	0.01	0.01
$\frac{3}{4}$	100	0.15	0.08	0.01	0.01
Full	100	0.22	0.11	0.01	0.01

Table 4.IV: Average integrality gaps for formulation (PS$'_{\text{path}}$), the warm-start Lagrangian solution ((SUBP)), (PS$_{\text{hull}}$), and the primal Lagrangian heuristic (Primal) on industrial TUFLP-S instances.

(warm-started) Lagrangian subproblem, and for formulation (PS$_{\text{hull}}$) solved with Algorithm 3, along with the optimality gap for the primal Lagrangian heuristic. Most bounds are close to the optimum. We only computed optimal values up to a relative error of 10^{-4}, so we processed gaps as though they always were at least 10^{-4} (*i.e.*, 0.01%).

On these instances, the primal heuristic always converges within 0.01% of optimum, and the linear relaxation of (PS$_{\text{hull}}$) exceeds that limit for two instances (with gaps of 0.02%). Together, these two methods virtually form an exact method for our industrial instances. We achieved comparable lower and upper bounds with stabilised Dantzig-Wolfe decomposition (Algorithm 4) and bundle or volume algorithms.

Section 3.3.2 showed that the linear relaxation of (PS$'_{\text{path}}$) is strong on industrial instances; the performance profiles in Fig. 4.4 show that the linear relaxation of (PS$_{\text{hull}}$) is even tighter, especially on large instances.

We already noted that the Lagrangian heuristic is virtually exact on industrial instances. Table 4.V further shows that, in terms of runtime, the Lagrangian heuristic is comparable to solving (PS$^0_{\text{path}}$) with CPLEX, the best formulation for industrial instances (Section 3.3.2).

The performance profiles in Fig. 4.5 confirm this conclusion. The unstabilised Lagrangian heuristic is competitive with a state-of-the-art branch-and-cut algorithm on all but the smallest instances. Moreover, as Table 4.VI shows, Algorithms 3 and 4 compute their strong bounds more quickly than bundle and volume algorithms for an equivalent Lagrangian dual.

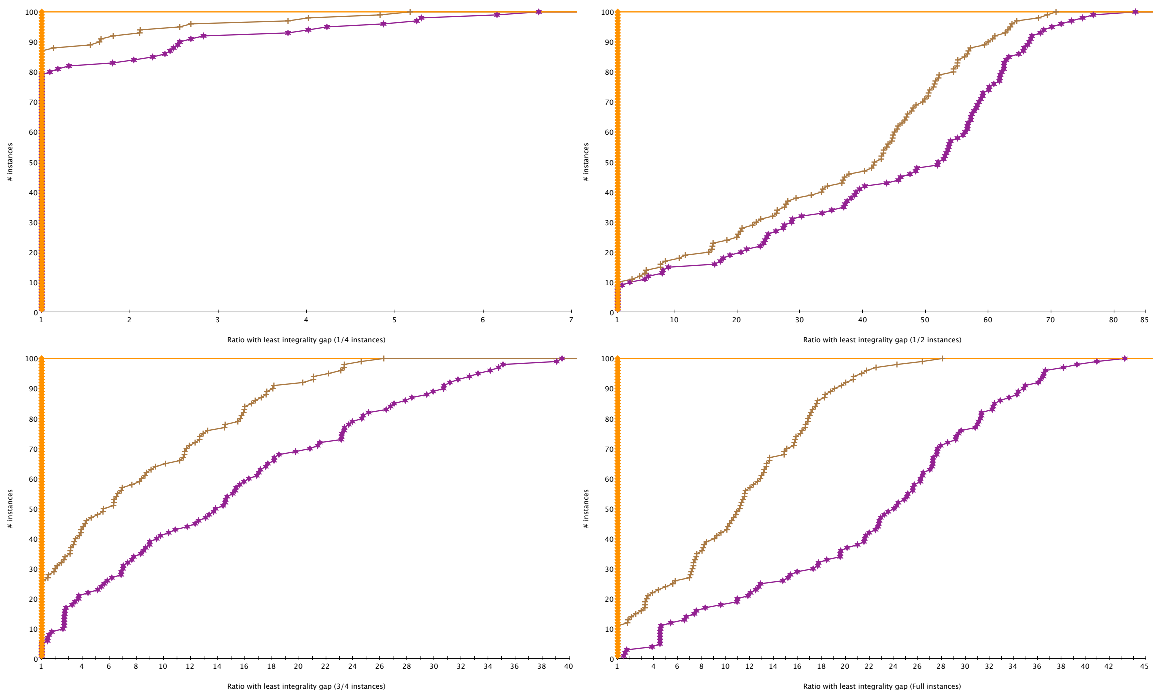


Figure 4.4: Performance profiles for the gap between lower bounds and the optimum value for industrial TUFLP-S instances (purple stars: (PS'_{path}); brown crosses: initial Lagrangian subproblem; orange diamonds: (PS_{hull})).

Set	MIP time (sec) for (PS^0_{path})	Average amount of time spent in each phase (sec) (PS'_{path})	(SUBP)	(PS_{hull})	Primal
$\frac{1}{4}$	0.06	0.02	0.01	0.13	0.02
$\frac{1}{2}$	2.96	0.44	0.20	2.57	0.29
$\frac{3}{4}$	15.25	3.20	0.92	13.61	0.91
Full	139.49	23.15	9.23	108.78	2.58

Table 4.V: Average amount of time to solve formulation (PS_{path}) to optimality, and for each step of the Lagrangian heuristic— the linear relaxation of (PS'_{path}), the initial Lagrangian subproblem (SUBP), the iterative algorithm for (PS_{hull}), and the primal heuristic—on industrial TUFLP-S instances.

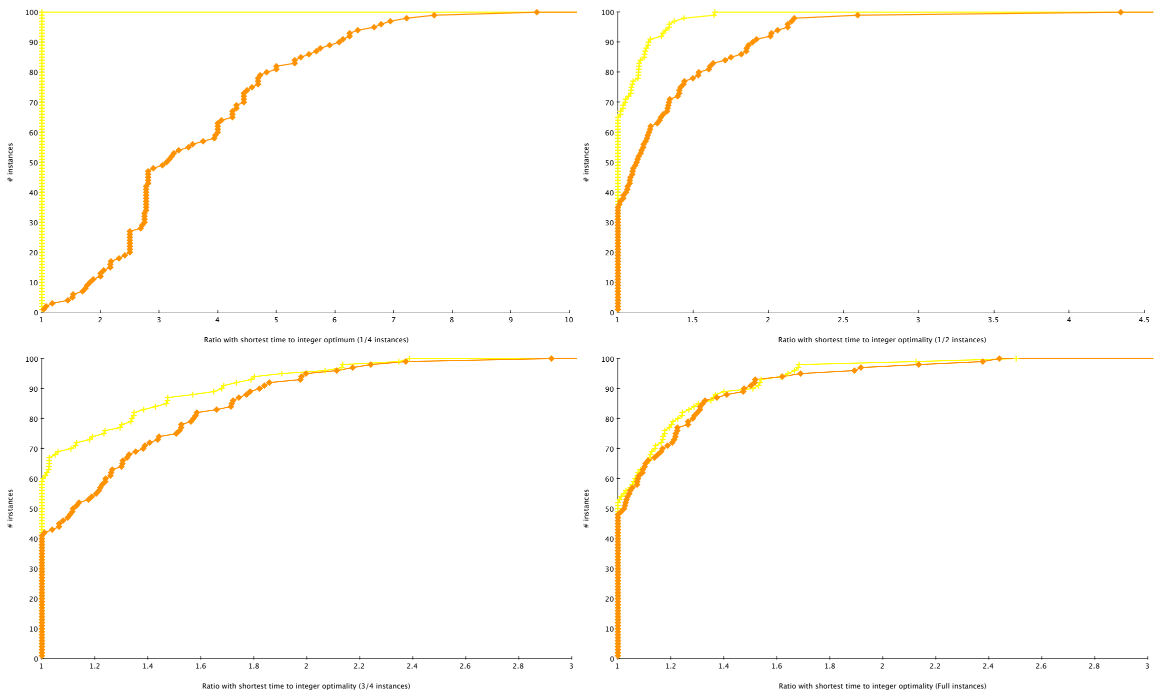


Figure 4.5: Performance profiles for runtimes to solve industrial TUFLP-S instances to optimality (yellow crosses: (PS^0_{path}); orange diamonds: non-stabilised structured Dantzig-Wolfe decomposition for (PS_{hull})).

		Average amount of time to optimality (sec)					
				Stabilisation term			
Set	n	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{1}{4}$	100	0.11	0.59	0.18	0.19	0.21	0.21
$\frac{1}{2}$	100	47.52	154.50	3.49	3.57	3.72	3.58
$\frac{3}{4}$	100	67.64	318.62	18.64	53.07	31.48	52.07
Full	100	1713.64	3737.23	143.73	233.17	180.12	232.70

Table 4.VI: Average amount of computation time to optimality (up to 2 hours) for the bundle and volume master problems and the structured Dantzig-Wolfe decomposition with and without stabilisation, for industrial TUFLP-S instances.

4.3 Summary

We exploited the cost structure of our industrial TUFLP-S instances with a Lagrangian relaxation that lead to a new Lagrangian-informed reformulation.

The reformulation eliminates the integrality gap for industrial instances and, compared to prior formulations, reduces it for small artificial instances. We solved the linear relaxation of the reformulation with Algorithms 3 and 4 and combined these structured Dantzig-Wolfe decomposition algorithms with a mathematical heuristic; the result is a Lagrangian heuristic that solves our industrial instances exactly.

This chapter confirmed that linear stabilisation, particularly the ℓ_1 ball, is useful for Lagrangian-informed structured Dantzig-Wolfe decomposition.

More importantly, we described a Lagangian heuristic that is competitive with a state-of-the-art branch-and-cut method on large practical instances of the TUFLP-S: the heuristic is exact on these instances and solves large ones as quickly as branch-and-cut. We believe that the heuristic will outperform branch-and-cut on larger practical instances. The same Lagrangian-informed structured decomposition will also be better suited to re-optimisation than branch-and-cut: we can reuse the restricted master problem.

CHAPTER 5

A LAGRANGIAN-INFORMED REFORMULATION FOR A MODULAR LOCATION PROBLEM

This chapter presents novel formulations for an industrial location problem with modular capacity constraints [34]. We first introduce a stronger mixed integer programming formulation for the problem than the one proposed in Gendron and Semet [34]. We then describe a Lagrangian decomposition scheme for the formulation, and derive a novel reformulation from that decomposition. The reformulation is large and we solve its linear relaxation with the warm-starting technique (Algorithm 1) of Section 2.2.

The reformulation yields lower bounds that are competitive with those computed by a parallel branch-and-cut algorithm after four hours. We also combine the reformulation with a mathematical heuristic to tighten the integrality gap more quickly than parallel branch-and-cut.

Section 5.1 describes the industrial problem and an initial mixed integer formulation for the model; the formulation builds on (PS_{path}) (introduced in Chapter 2) to closely approximate the uncapacitated location aspect of the problem in its linear programming relaxation. Section 5.2 introduces a Lagrangian decomposition for that mixed integer formulation and guides a reformulation with that decomposition. Section 5.3 shows how we exploit fractional solutions in a primal heuristic. Finally, Section 5.4 compares the lower and upper bounding algorithms with the parallel branch-and-cut method of CPLEX 12.5.

5.1 An industrial facility location problem and its MIP formulation

The case study for this final application of Lagrangian-informed reformulation comes from the operations of a European company that manages a dedicated delivery network.

The distribution network is arranged in three tiers: large vehicles from hubs to depots, medium ones from depots to satellites, and final delivery vehicles from satellites to delivery routes. This structure exposes opportunities for economies of scale at the top of the network (between hubs, depots, and satellites), but also scales down to local demand levels (between satellites and delivery routes).

The company owns and operates its top-tier hubs, but opens all other locations temporarily in order to maximise flexibility: it rents depots for short periods, and parcels are transferred from transport vehicles to delivery vehicles in ad hoc satellite locations that do not incur any direct cost. In order to exploit this flexibility and respond to changes in demand, we must solve two-level location-distribution problems (we disregard hubs because we can always link depots to the closest hub).

Gendron and Semet [34] proposes a model that captures the location aspect of the problem precisely: decision variables determine which locations to open and how to connect hubs to preassigned delivery routes.

Most research on two-level location-distribution problems instead focus on routing decisions [27] and consider the problem as a variant of the vehicle routing problem: they assume that most, if not all, location decisions are made ahead of time.

The latter choice seems correct in classical settings: opening or closing depots and satellites are usually strategic decisions made over long planning horizons. However, the specific problem encountered by our industrial partner does not correspond to these assumptions: hubs are already in operation, and the partner makes all other decisions, including location decisions, on a short term basis (over a single planning period in our model). At country scale, location and sorting costs can drastically outweigh the cost of the final delivery routes. Short planning horizons also mean that we must find solutions quickly.

The problem is best seen as a two-level uncapacitated facility location problem with modular capacities for depots, for arcs between depots and satellites and for satellites. We must add the constraint that flows are unsplittable and that each satellite be linked to at most one depot: parcels are sorted into routes at satellites and it is impractical to coordinate departures from multiple depots so that they arrive at an intermediate satellite location simultaneously. The result is a TUFLP-S with a more complicated cost function.

Modular capacities (in terms of total parcel volume) for depots correspond to large

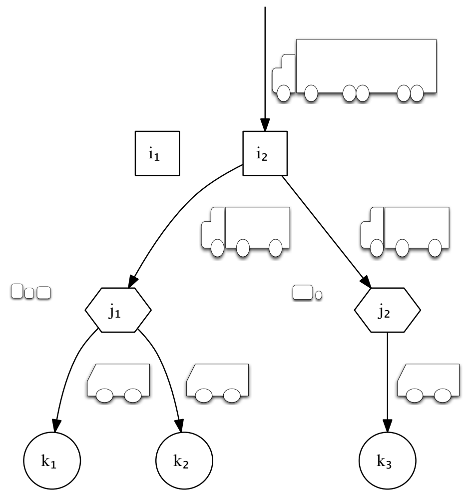


Figure 5.1: A solution for a toy instance of the modular location problem

distribution vehicles from the nearest hub to each depot and those for depot–satellite arcs to medium-size distribution vehicles from depots to satellites. Capacity modules for satellites instead reflect the cost of sorting each batch of parcels into delivery routes; they are in terms of the number of parcels.

Figure 5.1 schematises a solution for a trivial instance of our problem. One large vehicle links depot i_2 to its closest hub. Two medium-size vehicles then leave i_2 for satellites j_1 and j_2. Parcels are sorted in j_1 and j_2, and delivery vehicles depart from the satellites to execute routes k_1, k_2 and k_3.

We build on our work on the TUFLP-S to define a stronger MIP formulation than the one proposed in Gendron and Semet [34].

The formulation depends on the following data:

– I, the set of potential depot locations;

– J, the set of potential satellite locations;

– K, the set of routes;

– v_k, the total volume of all parcels delivered along route $k \in K$;

– n_k, the number of parcels delivered along route $k \in K$;

– R, the volumetric capacity of each large vehicle;

- r, the volumetric capacity of each medium vehicle;
- Q, the number of parcels in each sorting batch;
- f_i, the fixed cost of each depot $i \in I$;
- F_i, the cost of operating one large vehicle to depot $i \in I$;
- H_{ij}, the cost of operating one medium-sized vehicle from depot i to satellite j;
- G_j, the cost of sorting one batch of parcels at satellite $j \in J$;
- c_{ijk}, the total cost of delivery route k when transiting through depot i and departing from satellite j.

Given these parameters, we define formulation ($\mathrm{M}_{\mathrm{weak}}$) on decision variables

$$y_i = \begin{cases} 1, & \text{if depot } i \text{ is open,} \\ 0, & \text{otherwise,} \end{cases} \qquad \forall i \in I,$$

$$z_j = \begin{cases} 1, & \text{if satellite } j \text{ is open,} \\ 0, & \text{otherwise,} \end{cases} \qquad \forall j \in J,$$

$$t_{ij} = \begin{cases} 1, & \text{if depot } i \text{ and satellite } j \text{ are operating together,} \\ 0, & \text{otherwise,} \end{cases} \qquad \forall (i,j) \in I \times J,$$

$$x_{ijk} = \begin{cases} 1, & \text{if route } k \text{ is served through pair } (i,j), \\ 0, & \text{otherwise,} \end{cases} \qquad \forall (i,j,k) \in I \times J \times K,$$

$$U_i = \text{the number of large vehicles to } i \text{ (departing from the nearest hub)} \qquad \forall i \in I,$$

$$V_{ij} = \text{the number of medium vehicles from } i \text{ to } j, \qquad \forall (i,j) \in I \times J,$$

$$N_j = \text{the number of batches of parcels sorted at } j, \qquad \forall j \in J.$$

$$(\mathrm{M_{weak}}) \quad \min_{y,t,U,N,V,x} \sum_{i \in I} (f_i y_i + F_i U_i) + \sum_{j \in J} G_j N_j + \sum_{(i,j) \in I \times J} H_{ij} V_{ij} + \sum_{(i,j,k) \in I \times J \times K} c_{ijk} x_{ijk}$$

subject to

$$\sum_{(i,j) \in I \times J} x_{ijk} = 1, \qquad \forall k \in K, \tag{5.1}$$

$$\sum_{i \in I} t_{ij} \leq 1, \qquad \forall j \in J, \tag{5.2}$$

$$x_{ijk} \leq t_{ij}, \qquad \forall (i,j,k) \in I \times J \times K, \tag{5.3}$$

$$\sum_{j \in J} x_{ijk} \leq y_i, \qquad \forall (i,k) \in I \times K, \tag{5.4}$$

$$\sum_{(j,k) \in J \times K} v_k x_{ijk} \leq RU_i, \qquad \forall I \in I, \tag{5.5}$$

$$\sum_{k \in K} v_k x_{ijk} \leq rV_{ij}, \qquad \forall (i,j) \in I \times J, \tag{5.6}$$

$$\sum_{(i,k) \in I \times K} n_k x_{ijk} \leq QN_j, \qquad \forall j \in J, \tag{5.7}$$

$$x_{ijk} \in \{0,1\}, \qquad \forall (i,j,k) \in I \times J \times K,$$

$$y_i \in \{0,1\}, \qquad \forall i \in I,$$

$$t_{ij} \in \{0,1\}, \qquad \forall (i,j) \in I \times J,$$

$$U_i \in \mathbb{N}, \qquad \forall i \in I,$$

$$V_{ij} \in \mathbb{N}, \qquad \forall (i,j) \in I \times J,$$

$$N_j \in \mathbb{N}, \qquad \forall j \in J.$$

The objective function is the sum of fixed costs for depots, of module costs for capacity at depots, at satellites, and from depots to satellites, and of transportation costs for delivery routes.

Constraints (5.1), (5.2), (5.3), and (5.4) are directly copied from ($\mathrm{PS_{path}}$) and improve the linear relaxation bound compared with the formulation proposed in Gendron and Semet [34]: the latter replaces (5.4) with the weaker

$$t_{ij} \le y_i, \quad \forall (i,j) \in I \times J.$$

Variables x_{ijk} are forced to take binary values because there may be a cost to flowing more commodities through a location or arc, and thus an incentive to split flows.

Constraints (5.5) and (5.6) force the allocation of enough capacity from hubs to depots and from depots to satellites; integrality ensures this happens in discrete increments. Constraints (5.7) do the same for the batches of parcels sorted at each satellite.

Formulation ($\mathrm{M_{weak}}$) comprises hundreds of thousands of variables and constraints when applied to country-size instances, but contemporary linear programming solvers solve its linear relaxation in seconds or minutes. However, it is mediocre: the ($\mathrm{PS_{path}}$) component is strong, but constraints (5.5) to (5.7) amount to linearising transportation and sorting costs. For example, the cost of flowing an additional unit of volume through depot i is always F_i/R.

We improve the linear programming bound with a simple observation: if flow passes though a depot, a satellite, or a depot–satellite link, its capacity must be strictly positive. This leads to three sets of valid inequalities:

$$U_i \ge y_i, \qquad \forall i \in I, \tag{5.8}$$

$$V_{ij} \ge t_{ij}, \qquad \forall (i,j) \in I \times J, \tag{5.9}$$

$$N_j \ge \sum_{i \in I} t_{ij}, \qquad \forall j \in J, \tag{5.10}$$

where constraints (5.10) exploit the equations

$$z_j = \sum_{i \in I} t_{ij}, \quad \forall j \in J$$

of Section 3.2.

We add this trio of constraints to ($\mathrm{M_{weak}}$) to define formulation ($\mathrm{M_{strong}}$).

$$(\text{M}_{\text{strong}}) \quad \min_{y,t,U,N,V,x} \sum_{i\in I}(f_i y_i + F_i U_i) + \sum_{j\in J} G_j N_j + \sum_{(i,j)\in I\times J} H_{ij} V_{ij} + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}$$

subject to

$$\begin{aligned}
\sum_{(i,j)\in I\times J} x_{ijk} &= 1, && \forall k \in K,\\
\sum_{i\in I} t_{ij} &\le 1, && \forall j \in J,\\
x_{ijk} &\le t_{ij}, && \forall (i,j,k) \in I\times J\times K,\\
\sum_{j\in J} x_{ijk} &\le y_i, && \forall (i,k) \in I\times K,\\
\sum_{(i,k)\in I\times K} n_k x_{ijk} &\le QN_j, && \forall j \in J,\\
\sum_{k\in K} v_k x_{ijk} &\le rV_{ij}, && \forall (i,j) \in I\times J,\\
\sum_{(j,k)\in J\times K} v_k x_{ijk} &\le RU_i, && \forall I \in I,\\
U_i &\ge y_i, && \forall i \in I,\\
V_{ij} &\ge t_{ij}, && \forall (i,j) \in I\times J,\\
N_j &\ge \sum_{i\in I} t_{ij}, && \forall j \in J,\\
x_{ijk} &\in \{0,1\}, && \forall (i,j,k) \in I\times J\times K,\\
y_i &\in \{0,1\}, && \forall i \in I,\\
t_{ij} &\in \{0,1\}, && \forall (i,j) \in I\times J,\\
U_i &\in \mathbb{N}, && \forall i \in I,\\
V_{ij} &\in \mathbb{N}, && \forall (i,j) \in I\times J,\\
N_j &\in \mathbb{N}, && \forall j \in J.
\end{aligned}$$

The next section strengthens (M_{strong}) and shows how to solve the linear relaxation of its tighter reformulation efficiently.

5.2 A Lagrangian-informed reformulation for (M_{strong})

Work on a Lagrangian decomposition [45] for formulation (M_{strong}) revealed that we can improve lower bounds by separating the TUFLP-S and all modular capacity (the unsplittable arc flow problem [3] covered in Section 1.3) components.

We first derive (M'_{strong}), a decomposable formulation with explicit linking constraints, by duplicating path variables x_{ijk} that appear in UAFP (unsplittable arc flow problem) components.

Constraints (5.18) are equivalent to constraints (5.6), given constraints (5.19).

The only complication is that, thanks to constraints (5.1), some path variables are equivalent with respect to constraints (5.5) and (5.7). In constraints (5.15) and (5.21), we aggregate classes of equivalent path variables in a single representative clone, via artificial constraints (5.16) and (5.22).

$$(\mathrm{M}'_{\mathrm{strong}}) \quad \min_{y,t,U,N,V,x} \sum_{i\in I}(f_i y_i + F_i U_i) + \sum_{j\in J} G_j N_j + \sum_{(i,j)\in I\times J} H_{ij} V_{ij} + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}$$

subject to

$$\sum_{(i,j)\in I\times J} x_{ijk} = 1, \qquad \forall k \in K, \tag{5.11}$$

$$\sum_{i\in I} t_{ij} \leq 1, \qquad \forall j \in J, \tag{5.12}$$

$$x_{ijk} \leq t_{ij}, \qquad \forall (i,j,k) \in I\times J\times K, \tag{5.13}$$

$$\sum_{j\in J} x_{ijk} \leq y_i, \qquad \forall (i,k) \in I\times K, \tag{5.14}$$

$$\sum_{k\in K} n_k x_k^j \leq QN_j, \qquad \forall j \in J, \tag{5.15}$$

$$\sum_{i\in I} x_{ijk} = x_k^j, \qquad \forall (j,k) \in J\times K, \tag{5.16}$$

$$N_j \geq \sum_{i\in I} t_{ij}, \qquad \forall j \in J, \tag{5.17}$$

$$\sum_{k\in K} v_k x_k^{ij} \leq rV_{ij}, \qquad \forall (i,j) \in I\times J, \tag{5.18}$$

$$x_{ijk} = x_k^{ij}, \qquad \forall (i,j,k) \in I\times J\times K, \tag{5.19}$$

$$V_{ij} \geq t_{ij}, \qquad \forall (i,j) \in I\times J, \tag{5.20}$$

$$\sum_{k\in K} v_k x_k^i \leq RU_i, \qquad \forall I \in I, \tag{5.21}$$

$$\sum_{j\in J} x_{ijk} = x_k^i, \qquad \forall (i,k) \in I\times K, \tag{5.22}$$

$$U_i \geq y_i, \qquad \forall i \in I, \tag{5.23}$$

$$x_{ijk}, x_k^i, x_k^j, x_k^{ij} \in \{0,1\}, \qquad \forall (i,j,k) \in I\times J\times K,$$

$$y_i \in \{0,1\}, \qquad \forall i \in I,$$

$$t_{ij} \in \{0,1\}, \qquad \forall (i,j) \in I\times J,$$

$$U_i \in \mathbb{N}, \qquad \forall i \in I,$$

$$V_{ij} \in \mathbb{N}, \qquad \forall (i,j) \in I\times J,$$

$$N_j \in \mathbb{N}, \qquad \forall j \in J.$$

We define a first Lagrangian decomposition by relaxing linking constraints (5.16), (5.19), and (5.22), as well as valid inequalities (5.17), (5.20) and (5.23). Doing so relaxes the problem into independent subproblems, and we exploit that with a disaggregated (one convex hull per subproblem) reformulation.

Let λ_{jk} be the Lagrange multipliers for constraints (5.16), μ_{ijk} those for (5.19) and ν_{ik} those for (5.22), and let ξ_j be the Lagrange multipliers for constraints (5.17), o_{ij} for (5.20) and π_i for (5.23).

The largest subproblem is a TUFLP-S component over variables x_{ijk}, t_{ij} and y_i, solved with formulation (PS_{path}) of Chapter 3 (relaxing the integrality of x_{ijk} does not affect the optimal value):

$$\min_{y,t,x} \sum_{i\in I}(\pi_i + f_i)y_i + \sum_{(i,j)\in I\times J}(\xi_j + o_{ij})t_{ij} + \sum_{(i,j,k)\in I\times J\times K}(c_{ijk} + \lambda_{jk} + \mu_{ijk} + \nu_{ik})x_{ijk}$$

subject to

$$\begin{aligned}
\sum_{(i,j)\in I\times J} x_{ijk} &= 1, && \forall k \in K,\\
\sum_{i\in I} t_{ij} &\leq 1, && \forall j \in J,\\
x_{ijk} &\leq t_{ij}, && \forall (i,j,k) \in I\times J\times K,\\
\sum_{j\in J} x_{ijk} &\leq y_i, && \forall (i,k) \in I\times K,\\
0 \leq x_{ijk} &\leq 1, && \forall (i,j,k) \in I\times J\times K,\\
y_i &\in \{0,1\}, && \forall i \in I\\
t_{ij} &\in \{0,1\}, && \forall (i,j) \in I\times J.
\end{aligned}$$

The remaining components are one UAFP for each potential location. For each depot $i \in I$, we have

$$\min_{x^i,U_i}(F_i - \pi_i)U_i - \sum_{k\in K} \nu_{ik}x^i_k$$

subject to

$$\sum_{k \in K} v_k x_k^i \leq RU_i, \qquad \forall I \in I,$$
$$U_i \in \mathbb{N},$$
$$x_k^i \in \{0,1\}, \qquad \forall k \in K.$$

Similarly, the subproblem corresponding to each satellite $j \in J$ is

$$\min_{x^j, N_j} (G_j - \xi_j) N_j - \sum_{k \in K} \lambda_{jk} x_k^j$$

subject to

$$\sum_{k \in K} n_k x_k^j \leq QN_j, \qquad \forall j \in J,$$
$$N_j \in \mathbb{N},$$
$$x_k^j \in \{0,1\}, \qquad \forall k \in K,$$

and the one for each link $(i, j) \in I \times J$ is

$$\min_{x^{ij}, V_{ij}} (H_{ij} - \mathrm{o}_{ij}) V_{ij} - \sum_{k \in K} \mu_{ijk} x_k^{ij}$$

subject to

$$\sum_{k \in K} v_k x_k^{ij} \leq rV_{ij}, \qquad \forall (i, j) \in I \times J,$$
$$V_{ij} \in \mathbb{N},$$
$$x_k^i \in \{0,1\}, \qquad \forall k \in K.$$

Despite disaggregation, this reformulation is ineffective: there are many UAFP components, and their reformulation is exactly the (disaggregated) Dantzig-Wolfe master problem. However, preliminary experiments showed that we can simplify the problem with little negative effect on the bound: we relax all UAFP components into splittable arc flow problems (SAFP) [3], and solve the (PS_{path}) component as a linear program.

We showed in Section 1.3 that the SAFP is a valid relaxation of the UAFP (we relax integrality for all but one variable) and derived reformulation (SAFP) for this class of subproblems. We further reduce the reformulation with a simplification from Section 1.5: if we allocate no capacity, nothing can transit through that location or arc, and we eliminate the $s = 0$ block.

We already showed in Section 3.3.2 that the integrality gap for (PS_{path}) is low for industrial instances; it makes sense to only solve the linear relaxation of (PS_{path}).

This second Lagrangian decomposition comprises a component that is the linear relaxation of (PS_{path}). We could equivalently completely relax all constraints that appear in that component (constraints (5.11) to (5.14)). However, the Lagrangian dual problem for the latter relaxation will likely be difficult to maximise: the relaxed constraints may be violated by a wide margin. Moreover, we will show how to compute initial Lagrange multipliers efficiently by preserving this linear programming component and only relaxing artificial linking constraints introduced by the decomposition. In the end, the choice between relaxing the TUFLP-S component and separating it from the SAFP components does not affect the complete Lagrangian-informed reformulation: we represent the convex hull of a linear programming component with the original formulation for that component. The only difference is that the Lagrangian master problem is better behaved for the decomposition (i.e., separating components by relaxing artificial linking constraints) than for the relaxation.

(SAFP) depends on additional parameters: M_i is the set of potential values of U_i,

$$M_i = \left\{ 1, 2, \ldots, \left\lceil \sum_{k \in K} v_k / R \right\rceil \right\} \quad \forall i \in I;$$

similarly, for each depot–satellite link

$$M_{ij} = \left\{ 1, 2, \ldots, \left\lceil \sum_{k \in K} v_k / r \right\rceil \right\} \quad \forall (i, j) \in I \times J;$$

and for each satellite

$$M_j = \left\{ 1, 2, \ldots, \left\lceil \sum_{k \in K} n_k / Q \right\rceil \right\} \quad \forall j \in J.$$

We can now replace (5.15), (5.18) and (5.21) in ($\mathrm{M}_{\mathrm{strong}}$) with (SAFP) and obtain ($\mathrm{M}_{\mathrm{hull}}$).

$$(\text{M}_{\text{hull}}) \qquad \min_{y,t,U,N,V,x,\theta} \sum_{i\in I}(f_i y_i + F_i U_i) + \sum_{j\in J} G_j N_j + \sum_{(i,j)\in I\times J} H_{ij} V_{ij} + \sum_{(i,j,k)\in I\times J\times K} c_{ijk} x_{ijk}$$

subject to

$$\sum_{(i,j)\in I\times J} x_{ijk} = 1, \qquad \forall k \in K,$$

$$\sum_{i\in I} t_{ij} \le 1, \qquad \forall j \in J,$$

$$x_{ijk} \le t_{ij}, \qquad \forall (i,j,k) \in I\times J\times K,$$

$$\sum_{j\in J} x_{ijk} \le y_i, \qquad \forall (i,k) \in I\times K,$$

$$\sum_{k\in K} n_k x_k^{js} \le Qs\theta^{js}, \qquad \forall j \in J, \forall s \in M_j, \tag{5.24}$$

$$\sum_{s\in M_J} \theta^{js} = \sum_{i\in I} t_{ij}, \qquad \forall j \in J, \tag{5.25}$$

$$0 \le x_k^{js} \le \theta^{js}, \qquad \forall (j,k) \in J\times K, \forall s \in M_j, \tag{5.26}$$

$$\sum_{i\in I} x_{ijk} = \sum_{s\in M_j} x_k^{js}, \qquad \forall (j,k) \in J\times K, \tag{5.27}$$

$$N_j = \sum_{s\in M_J} s\theta^{js}, \qquad \forall j \in J, \tag{5.28}$$

$$\sum_{k\in K} v_k x_k^{ijs} \le rs\theta^{ijs}, \qquad \forall (i,j) \in I\times J, \forall s \in M_{ij}, \tag{5.29}$$

$$\sum_{s\in M_{ij}} \theta^{ijs} = t_{ij}, \qquad \forall (i,j) \in I\times J, \tag{5.30}$$

$$0 \le x_k^{ijs} \le \theta^{ijs}, \qquad \forall (i,j,k) \in I\times J\times K, \forall s \in M_{ij}, \tag{5.31}$$

$$x_{ijk} = \sum_{s\in M_{ij}} x_k^{ijs}, \qquad \forall (i,j,k) \in I\times J\times K, \tag{5.32}$$

$$V_{ij} = \sum_{s\in M_{ij}} s\theta^{ijs}, \qquad \forall (i,j) \in I\times J, \tag{5.33}$$

$$\sum_{k\in K} v_k x_k^{is} \le Rs\theta^{is}, \qquad \forall i \in I, \forall s \in M_i, \tag{5.34}$$

$$\sum_{s\in M_i} \theta^{is} = y_i, \qquad \forall i \in I, \tag{5.35}$$

$$0 \le x_k^{is} \le \theta^{is}, \qquad \forall (i,k) \in I\times K, \forall s \in M_i, \tag{5.36}$$

$$\sum_{j\in J} x_{ijk} = \sum_{s\in M_i} x_k^{is}, \qquad \forall (i,k) \in I\times K, \tag{5.37}$$

$$U_i = \sum_{s\in M_i} s\theta^{is}, \qquad \forall i \in I, \tag{5.38}$$

with variables x_{ijk}, y, t and θ binary, and x^s_{ijk}, U, V and N in $\mathbb{R}$.

Constraints (5.15) become constraints (5.24), (5.25) and (5.26); (5.18) become (5.29), (5.30) and (5.31); finally (5.21) become (5.34), (5.35) and (5.36).

There is one nontrivial change: valid inequalities (5.17), (5.20), and (5.23) combine with the choice constraints ($\sum_{s \in Y} \theta^s \leq 1$) to yield constraints (5.25), (5.30), and (5.35). Logically, the inequalities mean that if a location variable is 1, at least one full module of capacity must be installed at that location, and none otherwise. The reformulation represents integer variables as sets of binary variables, and we have tighter constraints for the same conditions.

This final reformulation is reasonably larger than (M_{strong}): the increase is polynomial in the input values, pseudopolynomial in initial formulation size.

Despite its benign growth in size, the reformulation is only barely tractable for full-scale industrial instances: even with the parallel barrier implementation in CPLEX 12.5, solving its linear relaxation requires more than one hour on a 24-core 2.9 GHz machine. The positive result is that the integrality gap is low.

That is why we developed Algorithm 1: we propose to solve these large linear programs approximately, but more quickly. We obtain initial dual multipliers by solving (M_{strong}) (*i.e.*, (B)) with a parallel barrier algorithm: interior point methods scale well to large programs like (M_{strong}), there is no parallel simplex algorithm, and we do not need a simplex basis. We then relax only constraints (5.27), (5.32) and (5.37) in (M_{hull}) to define (B′): this does not fully separate the problem, but suffices to make (B′) tractable. Finally, we solve (B′) with the dual simplex algorithm and use the advance basis to warm-start the dual simplex algorithm on (M_{hull}) (*i.e.*, (B″)).

5.3 A MIP-based heuristic for (M_{strong})

Formulation (M_{strong}) is strong: its linear programming lower bound is close to the optimal integer value. The problem is that it is large and solving its linear relaxation is slow.

We obtain a simple heuristic by eliminating decision variables based on information from ($\mathrm{M}_{\mathrm{hull}}$); we do not solve ($\mathrm{M}_{\mathrm{hull}}$) exactly, but hope that approximate solutions to this tight reformulation will be informative.

We restrict ($\mathrm{M}_{\mathrm{strong}}$) by only allowing each t_{ij} to be used if some flow passes through it in our fractional solution.

Let x^*_{ijk} be the values for path variables x_{ijk} in the approximate solution for the linear relaxation of ($\mathrm{M}_{\mathrm{hull}}$), and let

$$L = \{(i, j) \in I \times J \mid \exists k \in K, {x^*}_{ijk} > \varepsilon\},$$

with $\varepsilon = 10^{-4}$. We restrict ($\mathrm{M}_{\mathrm{strong}}$) by fixing

$$t_{ij} = 0 \quad \forall (i, j) \notin L.$$

In other words, if no path passes (even partially) through that depot–satellite link in the best fractional solution available, we deem the link uninteresting and let the heuristic disregard it.

We solve the restricted mixed integer program with the branch-and-cut of CPLEX, with an emphasis on primal feasibility, and solution polishing triggered when the time limit is nearly reached. A more sophisticated approach like Kernel search [2] could lead to better primal solutions. However, this simple heuristic already finds solutions that are within 2 to 4 percent of optimum on practical instances.

5.4 Numerical results

We presented three novel mixed integer programming formulations: ($\mathrm{M}_{\mathrm{weak}}$), its stronger variant ($\mathrm{M}_{\mathrm{strong}}$) and the Lagrangian-informed reformulation ($\mathrm{M}_{\mathrm{hull}}$).

This section first compares their linear relaxations with one another and with the lower bound computed by the parallel branch-and-cut of CPLEX after four hours of wallclock time.

We computed these results on the same unloaded 24-core machine, with four 2.9 GHz Xeon E5-4617 and 128 GB of DDR3-1600 RAM. Even four hours (*i.e.*, almost four CPU days) of parallel branch-and-cut do not suffice to close these instances, and the gaps for the three formulations are thus computed with respect to the inexact branch-and-cut lower bound.

Section 5.4.1 shows that the linear programming relaxation of (M_{hull}) almost eliminates the integrality gap on larger instances, but that, even with a parallel barrier (there is no effective parallel simplex algorithm) algorithm, it is impractical.

Section 5.4.2 compares parallel branch-and-cut with the lower and upper bounds computed with Algorithm 1 and the MIP-based heuristic of Section 5.3, under time limits.

It also compares the warm-started dual simplex with solving the Lagrangian decomposition that lead to the reformulation. We optimised the Lagrangian dual with the quadratically stabilised bundle method [48] (BTT 2.12 [1] [28]), and with the volume algorithm [5] (version 1.4 of the COIN implementation [2]). We initialised these methods as well with dual multipliers computed from a solution to (M_{strong}).

In both subsections, the results come from four sets of eight instances each.

We derived *F*ull instances from industrial data for the network over a complete country, while *L*arge ones keep three quarters of the nodes and arcs, *M*edium ones half and *S*mall ones one quarter. These networks correspond to the Full, $\frac{3}{4}$, $\frac{1}{2}$ and $\frac{1}{4}$ industrial instances of Chapters 3 and 4, and Table 3.II summarises their characteristics.

For each of these networks, we generated eight instances by either doubling the fixed cost of depots or leaving it as is, doubling the volumetric capacity of large vehicles (without changing their cost) or not, and halving the capacity of medium vehicles or not. We report the multipliers, in order, in the name of each instance; *e.g.*, F(1,2,.5) refers to the *F*ull instance with doubled large vehicle capacity and halved small vehicle capacity.

Unless noted otherwise, we solved linear programs and mixed integer programs with the dual simplex and branch-and-cut solvers of CPLEX 12.5 [3] with default parameters

1. An older version is available at `http://sorsa.unica.it/it/software.php`; we obtained BTT 2.12 by personal email.

2. The project homepage at `https://projects.coin-or.org/Vol` trails development; we checked out version 1.4 from the repository (`svn co https://projects.coin-or.org/svn/Vol/stable/1.4 coin-Vol`).

3. IBM CPLEX (`http://www-01.ibm.com/software/commerce/optimization/`

	(M_{strong})			Lagrangian subproblem			(M_{hull})		
Instance	# var.	# rows	# nonzeros	# var.	# rows	# nonzeros	# var.	# rows	# nonzeros
S(1,1,1)	3932	4718	25765	4176	4962	15801	18029	24449	83590
M(1,1,1)	30782	34808	209846	32243	36269	127759	196881	242686	877567
L(1,1,1)	98959	108596	681247	101908	111545	407403	811162	945114	3535831
F(1,1,1)	233918	250820	1614470	239066	255942	956405	2286559	2583731	9834020

Table 5.I: Presolved formulation sizes for four industrial instances.

and in single-threaded mode.

5.4.1 Formulation strengths

Formulation (M_{hull}) is large. Even with 24×2.9 GHz cores, the parallel barrier algorithm (including crossover [12]) took at least 80 minutes to solve each full instance.

That is not surprising considering the number of variables, constraints and nonzeros reported in Table 5.I. Thus, although we solved other formulations with the default (serial) dual simplex algorithm of CPLEX, runtimes for (M_{hull}) are wallclock times for the parallel barrier algorithm on 24 cores.

Solving (M_{strong}) as an integer program is difficult as well. We computed baseline bounds by executing the parallel branch-and-cut of CPLEX on that strong formulation for up to four hours on 24 cores. We chose this formulation because its linear relaxation is stronger than that of (M_{weak}) but is still solved rapidly (in minutes). Yet, we could only close the smallest instances and two medium ones within four hours.

Table 5.II reports average lower and upper bounds, gaps and runtimes for the mixed integer solver and the linear relaxations of the three formulations; complete results are in Appendix VI.

Most of the mixed integer programming bounds are suboptimal, so we present both upper and lower bounds, and average their differences under "MIP Gap." The remaining three columns ("Weak", "Strong" and "Hull") report the average gap between the MIP lower bound and those computed with the linear relaxations of (M_{weak}), (M_{strong}) and

`cplex-optimizer/`) is currently available under a free academic license through the IBM Academic Initiative (`http://www-03.ibm.com/ibm/university/academic/pub/page/ban_ilog_programming`).

Instances		MIP Upper	MIP Lower	MIP Gap	Weak	Strong	Hull
S(-,-,-)	Bound	$5.306 \cdot 10^7$	$5.305 \cdot 10^7$	0.01%	20.40%	1.66%	0.72%
	Time (s)	115.50	115.50		0.02	0.06	1.22
M(-,-,-)	Bound	$8.422 \cdot 10^7$	$8.411 \cdot 10^7$	0.12%	16.44%	1.18%	0.72%
	Time (s)	13109.36	13109.36		0.21	9.32	118.76
L(-,-,-)	Bound	$1.062 \cdot 10^8$	$1.050 \cdot 10^8$	1.12%	13.35%	1.12%	0.38%
	Time (s)	14400.45	14400.45		1.40	132.18	1187.38
F(-,-,-)	Bound	$1.254 \cdot 10^8$	$1.233 \cdot 10^8$	1.73%	14.47%	0.49%	0.05%
	Time (s)	14401.78	14401.78		9.97	599.99	8747.62

Table 5.II: Average bounds and runtimes under lenient computation limits

(M_{hull}). For three F instances, formulation (M_{hull}) computes stronger lower bounds than branch-and-cut; we treated these negative gaps as zeros.

We need heuristics to solve the industrial problem approximately and much more quickly: four hours is too long for short-term decisions. Moreover, we preassign delivery routes, and solving the problem faster will allow us to experiment with more route assignments.

5.4.2 Bound values under time limits

In this section, we compare two methods to compute feasible solutions and lower bounds under time limits. The first method is to solve (M_{strong}) as a mixed integer program with the branch-and-cut of CPLEX at default settings, but in opportunistic parallel mode and on 24 cores. The second combines the heuristic of Section 5.3 with warm-starting (Algorithm 1) the dual simplex algorithm on (M_{hull}) (Section 5.2).

Table 5.I shows the size of the Lagrangian subproblem in Algorithm 1 (*i.e.*, formulation (B')): the intermediate linear program is comparable in size to (M_{strong}) and solved even more quickly.

We dedicate half the wallclock time budget to the lower bounding algorithm of Section 5.2. This method also yields an approximate solution that we pass to the primal heuristic (Section 5.3). That heuristic receives the other half of the time limit.

Table 5.III breaks down lower bounding times. Column "Strong" reports the average time to solve (M_{strong}) with the parallel barrier algorithm; "Lagrangian" the average time

Instances	Lower bound times (10 min) Strong	Lagrangian	Hull	(15 min) Hull
S(-,-,-)	0.39	0.05	1.12	1.11
M(-,-,-)	3.80	1.12	225.44	300.38
L(-,-,-)	23.41	11.08	263.37	413.71
F(-,-,-)	137.59	81.46	62.14	215.69

Table 5.III: Average division of real time usage (sec) within the warm-starting heuristic

	Lower bound gaps (10 min)					Upper bound gaps (10 min)	
Instances	Strong	Hull	Bundle	Volume	MIP	Heuristic	MIP
S(-,-,-)	1.66%	0.72%	0.76%	0.76%	0.00%	0.19%	0.00%
M(-,-,-)	1.18%	0.73%	0.93%	0.82%	0.28%	0.00%	0.03%
L(-,-,-)	1.12%	0.59%	0.94%	0.96%	0.15%	0.05%	0.74%
F(-,-,-)	0.49%	0.47%	0.48%	0.48%	0.20%	0.18%	2.87%
	Lower bound gaps (15 min)					Upper bound gaps (15 min)	
Instances	Strong	Hull	Bundle	Volume	MIP	Heuristic	MIP
S(-,-,-)	1.66%	0.72%	0.75%	0.76%	0.00%	0.19%	0.00%
M(-,-,-)	1.18%	0.72%	0.91%	0.82%	0.25%	0.01%	0.02%
L(-,-,-)	1.12%	0.52%	0.92%	0.94%	0.15%	0.05%	1.00%
F(-,-,-)	0.49%	0.38%	0.48%	0.41%	0.12%	0.01%	2.62%

Table 5.IV: Average bound gaps under aggressive real time limits

to solve (B') (*i.e.*, the linear Lagrangian subproblem of (M_{hull})); finally, "Hull" reports the average time left to improve the dual feasible basis for the linear relaxation of (M_{hull}) (formulation (B'')). When we allow more time, only the final step is affected, and we only report runtimes for "Hull." Some time is also necessary to modify formulation (B') into (B'') (*i.e.*, (M_{hull})) and is not attributed to any of the three steps; that is why the averages consistently sum to less than half the total time limit. Simply solving (M_{strong}) already comes close to the time limits; the warm-started dual simplex algorithm must improve the lower bound with what little computation time remains.

We explored other methods to solve the linear relaxation of (M_{hull}) as a Lagrangian decomposition, with (PS^0_{path}) solved as a linear program and each SAFP subproblem with a $\mathcal{O}(n \log n)$-time routine. We warm-started all methods with the same multipliers obtained by solving (M_{strong}) and imposed the same time limits. Structured Dantzig-Wolfe decomposition (Section 2.3) was a failure: its restricted master problem becomes impractically large after one iteration. The quadratically-stabilised bundle [28] and the volume [5] algorithms fared better, and we will report results for both these methods.

Table 5.IV summarises the quality of the bounds computed within wallclock time limits of 10 and 15 minutes. Again, we do not have exact values and report lower bound gaps with respect to the lower bound obtained with the parallel branch-and-cut (on (M_{strong})) after four hours.

Columns “Strong” and “Hull” correspond to the linear relaxation of (M_{strong}) and to the warm-started linear relaxation of (M_{hull}). Columns “Bundle” and “Volume” report the quality of the bounds computed by the bundle and volume algorithms, for a Lagrangian dual equivalent to the linear relaxation of (M_{hull}); columns “MIP” correspond to solving (M_{strong}) as an integer program.

The results support our intuition that it is preferable to solve barely tractable linear programs with linear programming solvers: neither nondifferentiable optimisation algorithm is competitive with the warm-started dual simplex algorithm. In particular, the bundle algorithm scarcely achieves any improvement on the initial (M_{strong}) bound for Full instances.

We also report upper bound gaps relative to the upper bound computed by branch-and-cut after four hours. The *Heuristic* of Section 5.3 quickly computes better feasible solutions than branch-and-cut on (M_{strong}) under time limit; in fact, they are nearly identical to the ones the latter finds after four hours.

Overall, our warm start (Section 5.2 and Algorithm 1) is able to improve the initial dual multipliers, and seems more efficient that nondifferentiable optimisation methods. However, the branch-and-cut of CPLEX is slightly more efficient at converting computation time into bound improvements. On the other hand, our primal heuristic (Section 5.3) computes better primal solutions than the branch-and-cut under the same time limit. Combining the warm start with the primal heuristic yields a method that computes better integer solutions with tighter gaps than parallel branch-and-cut.

5.5 Summary

We described three novel formulations for the modular location problem introduced in Gendron and Semet [34]. The first, (M_{weak}), improves an older formulation by substituting in (PS_{path}), our formulation for the TUFLP-S. The second, (M_{strong}), adds three sets of valid inequalities; this improves the linear relaxation but already makes it difficult to solve. The third, (M_{hull}), is even larger; we obtained that reformulation after identifying a strong Lagrangian decomposition.

Formulation (M_{hull}) almost closes the integrality gap, but we are unable to solve its linear relaxation in reasonable time, even with a parallel interior point algorithm. Instead, we implemented Algorithm 1 to warm start the dual simplex algorithm. This quickly produces bounds that, although suboptimal, are higher than the linear programming bound of (M_{strong}).

We also described a primal heuristic (Section 5.3). Given short time limits, this heuristic computes better feasible solutions than the parallel branch-and-cut method of CPLEX for (M_{strong}). Combining this heuristic with the warm start for (M_{hull}) yields a method that, under the same total time limit (10 and 15 minutes), computes better primal solutions and proves tighter gaps than parallel branch-and-cut on (M_{strong}).

Overall, these results show that it is practical and useful to solve the linear relaxation of a large Lagrangian-informed reformulation approximately, by warm-starting the dual simplex algorithm.

CHAPTER 6

CONCLUSION AND FUTURE WORK

This dissertation explored an approach to reformulate mixed integer programs. The reformulations are guided by Lagrangian relaxations: once we identify a relaxation scheme, we construct a stronger mixed integer Lagrangian-informed reformulation mechanically. This second program captures the same integer feasible set as the original formulation, and its linear relaxation is equivalent to the guiding Lagrangian dual. We can always reach formulations through different paths; the particularity of our approach is that insights into the problem directly lead to a tighter linear relaxation. Chapter 2 describes methods to solve the linear relaxation of large reformulations; Chapters 3 to 5 show that the reformulation improves lower bounds in practice and incurs reasonable computational overhead.

Chapter 3 shows that this improvement sometimes comes with a reduction in formulation size; we even solve the reformulation as a mixed integer program. Unfortunately, off-the-shelf branch-and-cut programs are badly adapted to Lagrangian-informed reformulations. It is best to preserve all initial integer variables even if they are redundant in the reformulation, and to only branch on variables that are present in the original formulation. The result is to exploit the reformulation to strengthen the linear programming relaxation without otherwise affecting branching.

Despite this mismatch, Lagrangian-informed reformulations improve the practicality of enhancing branch-and-bound with strong Lagrangian bounds. Guiding a branch-and-bound solver with a Lagrangian relaxation is the source of myriad practical challenges.

Obtaining an optimal (or nearly enough) bound can require many iterations. When we impose a hard limit on the number of Lagrangian subproblem evaluations, search nodes may end up with a weaker bound value than their parent.

The need for primal fractional solutions only worsens the situation. Few approaches for maximising Lagrangian duals compute a feasible solution that corresponds to the lower bound. Those that do [5, 21, 22, 48, 59] take cleverly weighted convex combinations of optimal (extreme) solutions for the Lagrangian subproblem; fine tuning the weights and generating extreme solutions requires many subproblem evaluations.

Warm-starting the master problem helps, but, with or without warm-starting, it takes many iterations to reach accurate solutions. If only a lower bound is necessary, medium or low accuracy solutions may suffice; however, state-of-the-art branching techniques require nearly feasible primal solutions that are close to the lower bounds, which, in turn, calls for multiple subproblem solutions in a close neighbourhood of the optimum.

The only advantage of Lagrangian-based branch-and-bound is that its lower bounds are closer to the optimal integer value than linear relaxation bounds. Particularly elegant relaxations may also enable more computationally efficient bounding, but such cases are rare and often only outstrip the dual simplex at the root node, when no warm start is possible.

The reformulation technique presented in this dissertation is a starting point for other, hopefully more practical, ways to compute Lagrangian bounds and embed them into branch-and-bound methods. The Lagrangian-informed reformulation of Chapter 3 is simply solved as a linear program. Section 2.1 describes a heuristic for larger reformulations: solve a Lagrangian subproblem once, with multipliers set to dual values for a related but simpler linear program. Chapter 4 shows the overhead is negligible.

Some reformulations strengthen the linear relaxation enough to make branching pointless. That is the case of the reformulations in Chapters 4 and 5: they nearly eliminate the integrality gap on practical instances. However, the reformulations are so large that solving their linear relaxations to (near) optimality poses practical challenges. Chapter 4 explores one potential solution: extending a restricted master problem iteratively, as a structured Dantzig-Wolfe decomposition. Chapter 5 instead warm-starts a generic linear programming solver on a large reformulation by first solving a smaller linear relaxation. In both chapters, the procedures are augmented with simple mathematical heuristics. When computational resources are limited, the combinations are competitive with branch-and-cut with respect to lower bounds on practical instances and regularly generate higher quality, if not optimal, primal solutions.

We also maximised the Lagrangian duals underlying Chapters 4 and 5 with classical non-differentiable optimisation solvers. Numerical results show that their runtime performance is modest and that they only achieve middling bounds. This seems typical of Lagrangian relaxation in practice: only a few particularly well designed specialised programs [5, 26, 38, 39, 46, 47] are competitive with state-of-the-art generic solvers. In some ways, this situation evokes the state of cutting-plane methods in the 1980's and early 1990's [11]: the theory is well understood, some specialised solvers apply the technique to good ends, but no robust generic algorithm exists.

We hope that Lagrangian-informed reformulations will improve this situation: our approach shifts the computational burden towards robust linear programming solvers. Our hope is strengthened by the fact that, for the foreseeable future, serial computing power will stagnate. Contemporary branch-and-cut solvers dedicate enormous efforts to improving mixed integer formulations through sequential constraint generation. The reformulation approach presented in this dissertation instead mechanically improves formulations with a one-time expansion; we can then solve the resulting massive relaxations with parallel linear optimisation routines.

BIBLIOGRAPHY

[1] K. Aardal, M. Labbé, J. Leung, and M. Queyranne. On the two-level uncapacitated facility location problem. *INFORMS Journal on Computing*, 8(3):289, 1996.

[2] E. Angelelli, R. Mansini, and M. Grazia Speranza. Kernel search: A general heuristic for the multi-dimensional knapsack problem. *Computers and Operations Research*, 37(11):2017–2026, 2010.

[3] A. Atamtürk and D. Rajan. On splittable and unsplittable flow capacitated network design arc–set polyhedra. *Mathematical Programming*, 92(2):315–333, 2002.

[4] A. Balakrishnan and S. C. Graves. A composite algorithm for a concave-cost network flow problem. *Networks*, 19(2):175–202, 1989.

[5] F. Barahona. The volume algorithm: producing primal solutions with a subgradient method. *Mathematical Programming*, 87(3):385–399, 2000.

[6] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance. Branch-and-Price: Column Generation for Solving Huge Integer Programs. *Operations Research*, 46(3):316–329, May 1998.

[7] A. I. Barros. *Discrete and Fractional Programming Techniques for Location Models*. PhD thesis, Erasmus University, Rotterdam, 1995.

[8] A. I. Barros and M. Labbé. A general model for the uncapacitated facility and depot location problem. *Location Science*, 2(3):173–191, 1994.

[9] M. S. Bazaraa and H. D. Sherali. On the choice of step size in subgradient optimization. *European Journal of Operational Research*, 7(4):380–388, 1981.

[10] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.

[11] R. E. Bixby and E. Rothberg. Progress in computational mixed integer programming — A look back from the other side of the tipping point. *Annals of Operations Research*, 149(1):37–41, 2007.

[12] R. E. Bixby and M. J. Saltzman. Recovering an optimal LP basis from an interior point solution. *Operations Research Letters*, 15(4):169–178, 1994.

[13] O. Briant, C. Lemaréchal, and P. Meurdesoif. Comparison of bundle and classical column generation. *Mathematical Programming*, 113(2):299–344, 2008.

[14] P. Chardaire, J. Lutton, and A. Sutter. Upper and lower bounds for the two-level simple plant location problem. *Annals of Operations Research*, 86:117–140, 1999.

[15] E. W. Cheney and A. A. Goldstein. Newton's Method for Convex Programming and Tchebycheff Approximation. *Numerische Mathematik*, 1:253–268, 1959.

[16] V. Chvátal. *Linear Programming*. Series of books in the mathematical sciences. W. H. Freeman, 1983.

[17] A. R. Conn, N. I. M. Gould, and P. L. Toint. *Trust Region Methods*. MPS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2000.

[18] K. L. Croxton, B. Gendron, and T. L. Magnanti. A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science*, 49(9):1268–1273, 2003.

[19] K. L. Croxton, B. Gendron, and T. L. Magnanti. Variable Disaggregation in Network Flow Problems with Piecewise Linear Costs. *Operations Research*, 55(1): 146–157, 2007.

[20] G. B. Dantzig and W. Orchard-Hays. The Product Form for the Inverse in the Simplex Method. *Mathematical Tables and Other Aids to Computation*, 8(46): 64–67, Apr. 1954.

[21] G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8(1):101–111, 1960.

[22] G. B. Dantzig and P. Wolfe. The decomposition algorithm for linear programs. *Econometrica: Journal of the Econometric Society*, 29(4):767–778, 1961.

[23] J. Desrosiers and M. E. Lübbecke. A Primer in Column Generation. In *link.springer.com*, pages 1–32. Springer-Verlag, New York, 2005.

[24] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213, 2002.

[25] A. Engau, M. F. Anjos, and A. Vannelli. On Interior-Point Warmstarts for Linear and Combinatorial Optimization. *SIAM Journal on Optimization*, 20(4):1828–1861, 2010.

[26] D. Erlenkotter. A Dual-Based Procedure for Uncapacitated Facility Location. *Operations Research*, 26(6):992–1009, 1978.

[27] J. Feliu, G. Perboli, R. Tadei, and D. Vigo. The two-echelon capacitated vehicle routing problem. *Proceedings of the 22th European Conference on Operational Research*, 2007.

[28] A. Frangioni. Generalized Bundle Methods. *SIAM Journal on Optimization*, 13 (1):117–156, 2002.

[29] A. Frangioni. About Lagrangian methods in integer optimization. *Annals of Operations Research*, 139(1):163–193, 2005.

[30] A. Frangioni and B. Gendron. 0–1 reformulations of the multicommodity capacitated network design problem. *Discrete Applied Mathematics*, 157(6):1229–1241, 2009.

[31] A. Frangioni and B. Gendron. A stabilized structured Dantzig–Wolfe decomposition method. *Mathematical Programming*, 140(1):45–76, 2012.

[32] A. Frangioni and E. Gorgone. Bundle methods for sum-functions with “easy” components: applications to multicommodity network design. *Mathematical Programming*, pages 1–29, 2013.

[33] B. Gavish. Augmented Lagrangean Based Algorithms for Centralized Network Design. *Communications, IEEE Transactions on*, 33(12):1247–1257, 1985.

[34] B. Gendron and F. Semet. Formulations and relaxations for a multi-echelon capacitated location-distribution problem. *Computers and Operations Research*, 36(5): 1335–1355, 2009.

[35] B. Gendron, P. V. Khuong, and F. Semet. A Lagrangian-Based Branch-and-Bound Algorithm for the Two-Level Uncapacitated Facility Location Problem with Single-Assignment Constraints. Technical Report 2013-21, CIRRELT, 2013.

[36] A. M. Geoffrion. Duality in nonlinear programming: a simplified applications-oriented development. *SIAM Review*, 13(1):1–37, 1971.

[37] A. M. Geoffrion. Lagrangean relaxation for integer programming. In M. L. Balinski, editor, *Approaches to Integer Programming*, pages 82–114. Springer Berlin Heidelberg, 1974.

[38] P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting-stock problem. *Operations Research*, 9(6):849–859, 1961.

[39] P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting-stock problem — Part II. *Operations Research*, 11(6):863–888, 1963.

[40] J. L. Goffin. On convergence rates of subgradient optimization methods. *Mathematical Programming*, 13(1):329–347, 1977.

[41] J.-L. Goffin and J.-P. Vial. Convex nondifferentiable optimization: A survey focused on the analytic center cutting plane method. *Optimisation Methods and Software*, 17(5):805–867, 2002.

[42] J.-L. Goffin, A. Haurie, and J.-P. Vial. Decomposition and nondifferentiable optimization with the projective algorithm. *Management Science*, 38(2):284–302, 1992.

[43] J. Gondzio. Warm start of the primal-dual method applied in the cutting-plane scheme. *Mathematical Programming*, 83(1-3):125–143, 1998.

[44] L. Gouveia. A 2n Constraint Formulation for the Capacitated Minimal Spanning Tree Problem. *Operations Research*, 43(1):130–141, Jan. 1995.

[45] M. Guignard and S. Kim. Lagrangean decomposition: A model yielding stronger Lagrangean bounds. *Mathematical Programming*, 39(2):215–228–228, 1987.

[46] M. Held and R. M. Karp. The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18(6):1138–1162, 1970.

[47] M. Held and R. M. Karp. The traveling-salesman problem and minimum spanning trees: Part II. *Mathematical Programming*, 1(1):6–25, 1971.

[48] B. J. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms: Part 2: Advanced Theory and Bundle Methods*. A Series of Comprehensive Studies in Mathematics. Springer, 1996.

[49] K. Holmberg. Cross decomposition applied to integer programming problems: Duality gaps and convexification in parts. *Operations Research*, 42(4):657–668, 1994.

[50] R. G. Jeroslow and J. K. Lowe. Modelling with integer variables. In B. Korte and K. Ritter, editors, *Mathematical Programming Studies*, pages 167–184. Springer Berlin Heidelberg, Berlin, Heidelberg, 1984.

[51] E. John and E. A. Yıldırım. Implementation of warm-start strategies in interior-point methods for linear programming in fixed dimension. *Computational Optimization and Applications*, 41(2):151–183, 2007.

[52] K. Jones, I. Lustig, J. Farvolden, and W. Powell. Multicommodity network flows: The impact of formulation on decomposition. *Mathematical Programming*, 62(1-3):95–117–117, 1993.

[53] L. Kaufman, M. Eede, and P. Hansen. A plant and warehouse location problem. *Operational Research Quarterly*, 1977.

[54] J. E. Kelley, Jr. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8(4):703–712, 1960.

[55] K. C. Kiwiel. *Methods of descent for nondifferentiable optimization*. Lecture notes in mathematics. Springer-Verlag, 1985.

[56] Y. Kochetov and D. Ivanenko. Computationally difficult instances for the uncapacitated facility location problem. *Metaheuristics: Progress as real problem solvers*, pages 351–367, 2005.

[57] J. Krarup and P. Pruzan. The simple plant location problem: Survey and synthesis. *European Journal of Operational Research*, 12(1):36–81, 1983.

[58] M. Landete and A. Marín. New facets for the two-stage uncapacitated facility location polytope. *Computational Optimization and Applications*, 44(3):487–519, 2009.

[59] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov. New variants of bundle methods. *Mathematical Programming*, 69(1-3):111–147–147, 1995.

[60] C. E. Lemke. The dual method of solving the linear programming problem. *Naval Research Logistics*, 1(1):36–47, Mar. 1954.

[61] M. Lübbecke. *Generic Branch-Cut-and-Price*. PhD thesis, TU Berlin, 2010.

[62] B. T. Polyak. Minimization of unsmooth functionals. *USSR Computational Mathematics and Mathematical Physics*, 9(3):14–29, 1969.

[63] H.-B. Ro and D.-W. Tcha. A branch and bound algorithm for the two-level uncapacitated facility location problem with some side constraints. *European Journal of Operational Research*, 1984.

[64] N. Z. Shor. *Minimization Methods for Non-Differentiable Functions*, volume 3 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1985.

[65] S. Spoorendonk. *Cut and column generation*. PhD thesis, University of Copenhagen, 2008.

[66] T. J. Van Roy. Cross decomposition for mixed integer programming. *Mathematical Programming*, 25(1):46–63, 1983.

[67] J. Vielma, S. Ahmed, and G. Nemhauser. Mixed-Integer Models for Nonseparable Piecewise-Linear Optimization: Unifying Framework and Extensions. *Operations Research*, 58(2):303–315, 2010.

[68] P. Wolfe. Note on a method of conjugate subgradients for minimizing nondifferentiable functions. *Mathematical Programming*, 7(1):380–383, 1974.

[69] S. J. Wright. *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics, 1997.

[70] D. B. Yudin and A. S. Nemirovskii. Informational complexity and efficient methods for the solution of convex extremal problems. *Matekon*, 13(2):3–25, 1976.

Appendix I

Bound strengths (gap %) and computation times (s) for Gap TUFLP-C instances

In the first three tables, runtimes for the linear relaxation of (PS_{hull}) are reported for the structured Dantzig-Wolfe decomposition stabilised with an ℓ_1 term. Runtimes and lower bounds for all the methods to solve the linear relaxation (PS_{hull}) or the equivalent Lagrangian dual follow.

		LP relaxations					Reformulation		Primal	MIP times			
Instance	Exact	Landete	(PC_{path})	(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PC_{path})	(PS_{path})	(PS'_{path})	(PS^0_{path})
432GapA	45137	7.27%	7.77%	7.42%	7.42%	7.42%	7.42%	5.56%	0.00%				
			0.04	0.12	0.13	0.05	0.13	188.50	1.70	0.75	2.45	2.08	1.34
532GapA	45152	13.57%	14.45%	13.66%	13.66%	13.66%	13.66%	8.40%	0.00%				
			0.05	0.09	0.12	0.03	0.12	319.64	1.19	1.82	10.19	6.91	4.16
632GapA	48155	5.06%	7.87%	6.84%	6.84%	6.84%	6.84%	3.10%	0.00%				
			0.02	0.06	0.06	0.08	0.06	80.79	0.85	0.39	1.07	0.88	0.66
732GapA	42124	16.01%	16.46%	16.03%	16.03%	16.03%	16.03%	10.46%	0.00%				
			0.04	0.10	0.10	0.04	0.09	590.12	1.68	1.70	11.77	9.28	2.77
832GapA	48134	14.08%	14.50%	14.04%	14.04%	14.04%	14.04%	10.85%	0.00%				
			0.03	0.09	0.08	0.01	0.10	125.74	1.24	2.25	7.40	6.71	3.40
932GapA	45148	12.25%	13.05%	12.63%	12.63%	12.63%	12.63%	7.86%	0.00%				
			0.03	0.09	0.11	0.13	0.11	347.16	1.47	2.16	9.37	4.47	2.77
1032GapA	42161	6.14%	5.46%	5.05%	5.05%	5.05%	5.05%	0.92%	0.02%				
			0.04	0.09	0.10	0.06	0.09	119.55	0.97	0.34	0.86	0.78	0.59
1132GapA	45177	7.40%	10.88%	10.04%	10.04%	10.04%	10.04%	5.97%	0.00%				
			0.05	0.08	0.10	0.01	0.11	167.20	1.37	1.18	4.03	2.76	1.76
1232GapA	45140	10.17%	11.10%	10.48%	10.48%	10.48%	10.48%	5.68%	0.01%				
			0.04	0.08	0.09	0.02	0.06	346.93	1.54	1.09	3.17	2.72	1.57
1332GapA	42105	12.07%	14.44%	13.23%	13.23%	13.23%	13.23%	10.04%	0.00%				
			0.04	0.11	0.12	0.04	0.13	209.56	1.38	1.50	7.72	6.10	3.00
1432GapA	42166	10.45%	10.56%	10.44%	10.44%	10.44%	10.43%	6.22%	0.00%				
			0.03	0.08	0.09	0.06	0.12	71.11	0.75	1.07	3.68	3.50	1.97
1532GapA	42203	11.69%	11.88%	11.64%	11.64%	11.64%	11.63%	5.56%	0.00%				
			0.05	0.11	0.11	0.15	0.11	277.26	1.21	1.52	3.33	2.78	2.36
1632GapA	42175	11.72%	12.43%	11.75%	11.75%	11.75%	11.75%	6.17%	0.01%				
			0.06	0.14	0.12	0.02	0.12	289.19	1.48	1.81	6.70	3.41	3.20
1832GapA	48142	13.09%	13.20%	13.12%	13.12%	13.12%	13.12%	8.40%	0.00%				
			0.04	0.07	0.09	0.08	0.07	268.42	1.46	1.62	7.07	4.26	2.72
1932GapA	42165	8.32%	8.25%	8.23%	8.23%	8.23%	8.22%	3.48%	0.00%				
			0.03	0.11	0.09	0.07	0.09	237.99	1.13	0.64	1.92	1.69	0.98
2032GapA	42178	9.45%	9.64%	9.23%	9.23%	9.23%	9.23%	3.82%	0.00%				
			0.05	0.12	0.15	0.04	0.13	195.13	1.22	0.98	2.96	2.50	1.68
2132GapA	42149	12.21%	12.70%	12.30%	12.30%	12.30%	12.30%	5.21%	0.09%				
			0.06	0.13	0.12	0.34	0.12	618.65	2.42	1.82	10.10	6.63	3.19
2232GapA	45174	11.83%	12.77%	11.94%	11.94%	11.94%	11.94%	7.67%	0.00%				
			0.04	0.10	0.11	0.06	0.10	236.49	1.74	1.36	5.56	5.25	2.17
2432GapA	42137	11.98%	13.98%	12.98%	12.98%	12.98%	12.98%	7.40%	0.00%				
			0.04	0.10	0.12	0.04	0.12	518.62	1.90	2.08	8.45	6.22	3.17
2532GapA	42133	9.24%	9.76%	9.20%	9.20%	9.20%	9.20%	5.52%	0.00%				
			0.04	0.08	0.08	0.06	0.10	135.48	1.28	0.78	2.96	2.34	1.89
2632GapA	42168	11.14%	12.05%	11.29%	11.29%	11.29%	11.29%	6.44%	0.00%				
			0.06	0.13	0.14	0.04	0.16	233.06	1.27	1.21	5.05	3.56	2.55
2732GapA	42180	10.13%	10.58%	10.20%	10.20%	10.20%	10.20%	5.99%	0.00%				
			0.05	0.11	0.12	0.06	0.12	235.20	1.91	0.88	3.94	2.61	1.87
2832GapA	42169	7.45%	8.58%	7.63%	7.63%	7.63%	7.63%	3.91%	0.00%				
			0.04	0.10	0.11	0.08	0.13	384.67	1.78	0.66	2.03	1.31	1.25
2932GapA	45170	8.87%	10.68%	10.43%	10.43%	10.43%	10.43%	5.18%	0.00%				
			0.03	0.10	0.10	0.04	0.10	364.56	1.72	0.64	2.68	2.43	1.08
3032GapA	42146	9.61%	8.81%	8.37%	8.37%	8.37%	8.37%	3.85%	0.00%				
			0.04	0.12	0.11	0.04	0.12	153.45	1.30	0.76	1.89	2.75	1.33
3232GapA	45133	13.21%	14.28%	13.79%	13.79%	13.79%	13.79%	8.71%	0.01%				
			0.04	0.09	0.10	0.04	0.10	153.59	0.73	1.99	12.87	9.45	3.54

		LP relaxations					Reformulation		Primal	MIP times			
Instance	Exact	Landete	(PC_{path})	(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PC_{path})	(PS_{path})	(PS'_{path})	(PS^0_{path})
431GapB	51150	8.50%	8.32%	8.31%	8.31%	8.31%	8.31%	5.49%	0.02%				
			0.03	0.08	0.09	0.09	0.00	136.21	1.31	0.90	3.12	3.11	1.37
531GapB	54167	9.41%	9.88%	9.83%	9.83%	9.83%	9.83%	4.62%	0.03%				
			0.03	0.06	0.07	0.09	0.07	221.30	1.06	1.48	4.72	3.06	2.34
931GapB	52139	4.14%	1.84%	1.84%	1.84%	1.84%	1.84%	0.02%	0.00%				
			0.04	0.06	0.06	0.06	0.04	312.90	1.27	0.14	0.34	0.29	0.15
1031GapB	51157	9.58%	9.85%	9.56%	9.56%	9.56%	9.56%	6.34%	0.00%				
			0.04	0.10	0.10	0.11	0.02	347.23	1.90	1.69	11.26	5.32	2.38
1231GapB	45139	4.55%	6.62%	5.95%	5.95%	5.95%	5.95%	2.10%	0.00%				
			0.04	0.11	0.11	0.13	0.01	131.04	1.11	0.43	1.05	1.54	0.83
1731GapB	51147	9.39%	9.50%	9.33%	9.33%	9.33%	9.33%	4.72%	0.01%				
			0.03	0.11	0.09	0.10	0.02	236.88	1.30	1.85	5.52	3.64	1.50
2331GapB	48156	5.79%	5.90%	5.89%	5.89%	5.89%	5.89%	1.70%	0.00%				
			0.04	0.08	0.10	0.08	0.08	65.95	0.55	0.30	0.99	0.80	0.78
2431GapB	48162	11.20%	12.38%	11.96%	11.96%	11.96%	11.96%	6.91%	0.00%				
			0.04	0.09	0.09	0.09	0.04	143.67	0.94	1.16	5.59	3.46	2.23
2731GapB	45119	7.69%	8.17%	7.75%	7.75%	7.75%	7.73%	4.18%	0.01%				
			0.04	0.11	0.09	0.10	0.22	161.69	1.04	0.74	1.79	2.18	1.28
2831GapB	54136	6.87%	7.13%	7.05%	7.05%	7.05%	7.04%	2.73%	0.00%				
			0.05	0.07	0.07	0.09	0.10	76.85	0.66	0.71	2.33	1.27	1.56
2931GapB	48174	6.04%	6.71%	6.53%	6.53%	6.53%	6.53%	2.12%	0.00%				
			0.03	0.09	0.12	0.13	0.05	719.03	2.48	0.45	1.24	0.87	0.81
3131GapB	48196	9.96%	10.01%	9.90%	9.90%	9.90%	9.90%	3.58%	0.00%				
			0.05	0.09	0.10	0.10	0.06	634.42	1.80	1.14	2.30	1.93	1.99
3231GapB	51165	2.55%	2.93%	2.92%	2.92%	2.92%	2.92%	1.14%	0.00%				
			0.04	0.06	0.05	0.06	0.02	35.94	0.68	0.21	0.47	0.48	0.57

Instance	Exact	LP relaxations					Reformulation		Primal	MIP times			
		Landete	(PC_{path})	(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PC_{path})	(PS_{path})	(PS'_{path})	(PS^0_{path})
333GapC	42231	8.58%	8.97%	8.67%	8.67%	8.67%	8.67%	2.51%	0.00%				
			0.05	0.10	0.10	0.10	0.20	310.01	1.06	0.80	2.32	1.72	1.48
433GapC	42186	16.26%	17.23%	16.32%	16.32%	16.32%	16.32%	10.56%	0.04%				
			0.07	0.19	0.20	0.20	0.26	1528.86	2.87	5.40	36.51	21.76	9.20
533GapC	45129	13.74%	16.05%	14.34%	14.34%	14.34%	14.34%	10.96%	0.00%				
			0.04	0.13	0.16	0.17	0.05	619.30	2.00	3.55	96.44	17.28	6.46
633GapC	45162	11.67%	12.07%	11.92%	11.92%	11.92%	11.92%	6.61%	0.00%				
			0.04	0.08	0.08	0.09	0.05	171.72	0.92	1.26	4.88	4.18	2.01
733GapC	48105	14.91%	15.73%	15.60%	15.60%	15.60%	15.60%	10.80%	0.05%				
			0.05	0.10	0.13	0.13	0.22	248.24	1.15	5.29	106.32	27.51	10.30
833GapC	45154	10.68%	9.75%	9.38%	9.38%	9.38%	9.38%	5.06%	0.02%				
			0.06	0.11	0.12	0.15	0.04	350.58	1.65	1.41	7.61	5.94	2.00
933GapC	45103	12.66%	12.50%	12.01%	12.01%	12.01%	12.01%	9.87%	0.00%				
			0.04	0.07	0.08	0.09	0.01	50.84	0.68	1.02	5.44	2.97	1.73
1033GapC	48126	12.81%	12.50%	12.30%	12.30%	12.30%	12.30%	6.35%	0.02%				
			0.04	0.08	0.08	0.08	0.12	526.39	1.81	1.32	7.24	8.93	1.80
1133GapC	48106	11.98%	13.37%	12.59%	12.59%	12.59%	12.59%	7.83%	0.05%				
			0.06	0.12	0.14	0.14	0.12	352.26	1.36	4.84	30.05	19.75	7.64
1233GapC	42183	10.15%	11.11%	10.23%	10.23%	10.22%	10.23%	3.98%	0.00%				
			0.06	0.12	0.12	0.12	0.06	758.39	2.26	0.90	4.94	2.98	1.62
1433GapC	45124	15.11%	18.41%	17.38%	17.38%	17.38%	17.38%	11.52%	0.02%				
			0.08	0.16	0.14	0.16	0.16	1089.67	2.46	5.19	120.46	31.88	11.27
1533GapC	45134	13.18%	14.05%	13.79%	13.79%	13.79%	13.79%	8.90%	0.01%				
			0.04	0.09	0.12	0.12	0.02	527.52	1.72	2.06	18.02	11.30	4.05
1633GapC	42137	16.39%	18.00%	16.47%	16.47%	16.47%	16.47%	11.27%	0.00%				
			0.07	0.13	0.13	0.16	0.07	347.13	1.59	3.23	32.26	15.19	5.72
1733GapC	42129	17.19%	18.21%	17.61%	17.61%	17.60%	17.61%	11.63%	0.01%				
			0.09	0.16	0.16	0.18	0.16	1080.96	2.76	5.03	104.53	25.64	8.87
1833GapC	45143	12.89%	13.40%	13.07%	13.07%	13.07%	13.07%	7.14%	0.00%				
			0.04	0.12	0.12	0.12	0.06	328.89	1.61	1.25	10.97	6.18	2.47
1933GapC	51166	6.27%	8.48%	8.47%	8.47%	8.48%	8.47%	5.59%	0.00%				
			0.03	0.07	0.08	0.08	0.01	78.28	0.85	0.79	3.04	2.70	1.55
2133GapC	45154	10.67%	12.34%	11.11%	11.11%	11.11%	11.11%	5.88%	0.01%				
			0.04	0.08	0.09	0.11	0.10	165.59	1.01	1.31	6.13	5.30	2.42
2333GapC	42143	11.57%	12.41%	11.61%	11.61%	11.61%	11.61%	7.38%	0.00%				
			0.07	0.18	0.16	0.18	0.04	705.72	2.32	2.47	19.07	13.53	6.56
2433GapC	45151	13.79%	14.73%	14.16%	14.16%	14.16%	14.16%	7.50%	0.02%				
			0.06	0.12	0.14	0.14	0.32	641.63	1.76	2.27	13.61	8.61	5.03
2533GapC	42169	9.02%	9.26%	9.08%	9.08%	9.07%	9.08%	3.45%	0.00%				
			0.06	0.11	0.13	0.11	0.06	390.80	1.28	0.78	2.66	1.89	1.23
2733GapC	45144	15.26%	15.90%	15.45%	15.45%	15.45%	15.45%	9.44%	0.00%				
			0.06	0.11	0.14	0.19	0.06	557.84	1.43	3.96	35.79	16.03	7.96
2833GapC	45164	13.13%	14.12%	13.41%	13.41%	13.40%	13.41%	7.71%	0.00%				
			0.05	0.12	0.12	0.14	0.03	383.43	1.58	2.55	9.41	9.50	3.62
2933GapC	42162	5.38%	5.68%	5.50%	5.50%	5.50%	5.50%	1.18%	0.00%				
			0.04	0.12	0.10	0.11	0.08	215.47	1.15	0.54	0.90	1.08	0.97
3133GapC	42128	15.19%	16.22%	15.49%	15.49%	15.49%	15.49%	9.85%	0.00%				
			0.08	0.18	0.19	0.19	0.10	807.96	2.12	4.04	28.38	15.03	7.46
3233GapC	39177	8.60%	9.43%	8.71%	8.71%	8.71%	8.70%	3.26%	0.00%				
			0.06	0.15	0.15	0.14	0.28	322.17	1.31	0.98	3.14	2.56	1.59

Instance	Exact	NDO master		(PS_{hull})			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
432GapA	45237	5.77%	6.31%	5.77%	5.77%	5.77%	5.77%
		7200.89	321.62	1292.02	188.50	3659.66	293.91
532GapA	45252	8.65%	8.73%	8.60%	8.60%	8.60%	8.60%
		7200.45	1855.23	662.53	319.64	1498.77	442.18
632GapA	48255	3.30%	3.55%	3.30%	3.30%	3.30%	3.30%
		4468.34	313.58	659.36	80.79	493.41	137.43
732GapA	42224	10.80%	10.90%	10.67%	10.67%	10.83%	10.67%
		7201.58	1654.94	1520.47	590.12	7214.94	1042.66
832GapA	48234	11.04%	11.27%	11.03%	11.03%	11.03%	11.03%
		7200.44	697.17	606.29	125.74	1786.30	178.24
932GapA	45248	8.10%	8.20%	8.07%	8.07%	8.07%	8.07%
		7200.37	1747.03	1032.83	347.16	3640.59	545.76
1032GapA	42261	1.15%	1.25%	1.15%	1.15%	1.15%	1.15%
		2060.59	507.42	517.58	119.55	995.61	158.14
1132GapA	45277	6.18%	6.34%	6.18%	6.18%	6.18%	6.18%
		7200.54	1016.70	1000.62	167.20	1017.91	301.79
1232GapA	45240	5.91%	6.06%	5.89%	5.89%	5.89%	5.89%
		7201.24	1066.63	2161.65	346.93	3551.88	468.18
1332GapA	42205	10.26%	10.47%	10.26%	10.26%	10.26%	10.26%
		7200.18	1618.63	5963.93	209.56	1713.18	325.20
1432GapA	42266	6.45%	6.50%	6.44%	6.44%	6.44%	6.44%
		1357.71	321.54	310.13	71.11	449.98	96.53
1532GapA	42303	5.80%	5.90%	5.79%	5.79%	5.79%	5.79%
		7200.63	1495.27	376.08	277.26	2326.23	345.46
1632GapA	42275	6.41%	6.56%	6.39%	6.39%	6.39%	6.39%
		7200.51	1349.50	1578.42	289.19	1385.75	455.50
1832GapA	48242	8.60%	8.77%	8.59%	8.59%	8.59%	8.59%
		7200.39	1045.91	1483.44	268.42	1363.32	302.62
1932GapA	42265	3.71%	3.87%	3.70%	3.70%	3.70%	3.70%
		4815.95	646.28	3769.65	237.99	1064.47	364.08
2032GapA	42278	4.06%	4.20%	4.05%	4.05%	4.05%	4.05%
		7200.33	941.21	393.66	195.13	1191.94	229.81
2132GapA	42249	5.52%	5.61%	5.44%	5.44%	5.44%	5.44%
		7200.31	1624.28	1800.72	618.65	3747.96	928.95
2232GapA	45274	7.88%	7.97%	7.87%	7.87%	7.87%	7.87%
		6980.84	730.68	266.97	236.49	1127.14	217.25
2432GapA	42237	7.71%	7.87%	7.62%	7.62%	7.62%	7.62%
		7200.71	1685.08	1361.96	518.62	4604.04	748.76
2532GapA	42233	5.77%	5.90%	5.74%	5.74%	5.74%	5.74%
		3815.22	655.77	1102.61	135.48	1910.85	292.55
2632GapA	42268	6.67%	6.79%	6.66%	6.66%	6.66%	6.66%
		7200.66	1467.42	779.14	233.06	2384.10	604.41
2732GapA	42280	6.22%	6.38%	6.21%	6.21%	6.21%	6.21%
		7200.13	1073.57	986.38	235.20	1790.47	471.11
2832GapA	42269	4.16%	4.29%	4.14%	4.14%	4.45%	4.14%
		7200.87	1869.90	665.27	384.67	7216.00	610.06
2932GapA	45270	5.42%	5.53%	5.39%	5.39%	5.72%	5.39%
		7200.51	929.91	1732.25	364.56	7205.10	356.28
3032GapA	42246	4.09%	4.15%	4.08%	4.08%	4.08%	4.08%
		3903.21	650.33	852.98	153.45	1121.80	195.62
3232GapA	45233	8.92%	8.97%	8.92%	8.92%	8.92%	8.92%
		4401.13	767.74	2253.85	153.59	2834.47	232.80

Instance	Exact	NDO master Bundle	Volume	(PS_{hull}) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
431GapB	51250	5.68%	5.83%	5.67%	5.67%	5.67%	5.67%
		7200.11	656.91	158.08	136.21	1421.67	203.59
531GapB	54267	4.80%	4.90%	4.80%	4.80%	4.80%	4.80%
		6936.65	682.97	998.39	221.30	1998.19	291.07
931GapB	54246	0.20%	0.45%	0.20%	0.20%	0.20%	0.20%
		7200.41	683.85	788.09	312.90	666.56	376.75
1031GapB	51257	6.53%	6.67%	6.52%	6.52%	6.52%	6.52%
		7200.19	1177.24	754.79	347.23	3107.73	677.29
1231GapB	45239	2.32%	2.48%	2.32%	2.32%	2.32%	2.32%
		6165.59	588.86	261.99	131.04	1850.20	253.06
1731GapB	51247	4.91%	5.01%	4.90%	4.90%	4.90%	4.90%
		7200.52	1020.27	1102.73	236.88	907.40	318.20
2331GapB	48256	1.90%	1.93%	1.90%	1.90%	1.90%	1.90%
		625.55	459.66	238.25	65.95	119.75	102.68
2431GapB	48262	7.11%	7.23%	7.10%	7.10%	7.10%	7.10%
		7200.63	690.07	111.77	143.67	1691.93	288.59
2731GapB	45219	4.39%	4.53%	4.39%	4.39%	4.39%	4.39%
		7200.40	521.74	213.60	161.69	1073.82	161.80
2831GapB	54236	2.91%	2.92%	2.91%	2.91%	2.91%	2.91%
		554.62	427.26	1947.76	76.85	2509.20	94.97
2931GapB	48274	2.39%	2.54%	2.32%	2.32%	2.43%	2.32%
		7200.90	1868.29	4091.02	719.03	7223.32	1020.17
3131GapB	48296	3.90%	4.01%	3.78%	3.78%	3.78%	3.78%
		7200.99	1136.34	466.13	634.42	2084.33	784.45
3231GapB	51265	1.33%	1.35%	1.33%	1.33%	1.33%	1.33%
		427.08	247.18	220.97	35.94	135.02	43.53

		NDO master		(PS_{hull})			
Instance	Exact	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
333GapC	42331	2.80%	2.88%	2.74%	2.74%	2.74%	2.74%
		7200.67	2909.93	726.88	310.01	3808.36	411.82
433GapC	42286	11.14%	10.90%	10.78%	10.78%	11.12%	10.78%
		7201.11	4469.02	3229.31	1528.86	7273.97	1475.78
533GapC	45229	11.18%	11.27%	11.16%	11.16%	11.16%	11.16%
		7201.11	2392.45	4326.55	619.30	3561.74	1580.21
633GapC	45262	6.85%	6.97%	6.81%	6.81%	6.81%	6.81%
		7200.19	734.00	534.14	171.72	987.71	245.99
733GapC	48205	11.01%	11.08%	10.99%	10.99%	10.99%	10.99%
		7200.64	1060.83	447.82	248.24	985.70	479.26
833GapC	45254	5.30%	5.45%	5.27%	5.26%	5.26%	5.26%
		7200.26	1919.73	559.62	350.58	1712.09	499.95
933GapC	45203	10.07%	10.33%	10.07%	10.07%	10.07%	10.07%
		3327.71	270.19	265.74	50.84	699.85	338.58
1033GapC	48226	6.59%	6.71%	6.54%	6.54%	6.54%	6.54%
		7200.34	1030.38	1045.81	526.39	2640.41	725.70
1133GapC	48206	8.04%	8.14%	8.02%	8.02%	8.43%	8.02%
		7201.16	1778.74	989.70	352.26	7234.17	471.33
1233GapC	42283	4.34%	4.40%	4.21%	4.21%	4.21%	4.21%
		7201.53	2098.28	932.56	758.39	2959.64	1067.35
1433GapC	45224	11.88%	11.85%	11.71%	11.71%	11.71%	11.71%
		7201.19	2551.60	730.42	1089.67	5018.08	1468.61
1533GapC	45234	9.15%	9.28%	9.10%	9.10%	9.10%	9.10%
		7200.72	1782.67	1068.32	527.52	5102.60	768.66
1633GapC	42237	11.55%	11.66%	11.48%	11.48%	11.48%	11.48%
		7200.31	1577.32	604.02	347.13	2830.89	470.76
1733GapC	42229	11.99%	11.99%	11.83%	11.83%	12.07%	11.83%
		7200.48	3433.57	2565.76	1080.96	7319.31	1863.86
1833GapC	45243	7.37%	7.53%	7.35%	7.35%	7.35%	7.35%
		7200.29	1189.12	865.07	328.89	3337.70	446.04
1933GapC	51266	5.78%	5.93%	5.78%	5.78%	5.78%	5.78%
		2769.28	388.00	1280.80	78.28	1612.07	136.82
2133GapC	45254	6.12%	6.37%	6.09%	6.09%	6.09%	6.09%
		7200.45	519.89	1652.01	165.59	3199.28	242.40
2333GapC	42243	7.70%	7.76%	7.98%	7.60%	7.67%	7.60%
		7201.51	3219.42	7248.83	705.72	7217.35	1491.82
2433GapC	45251	7.88%	7.88%	7.71%	7.71%	7.71%	7.71%
		7200.83	2334.89	771.90	641.63	3960.16	960.83
2533GapC	42269	3.74%	3.87%	3.68%	3.68%	3.84%	3.68%
		7200.91	1396.43	648.31	390.80	7258.82	621.66
2733GapC	45244	9.71%	9.73%	9.64%	9.64%	9.64%	9.64%
		7201.69	3162.75	4118.37	557.84	2584.17	705.31
2833GapC	45264	7.97%	8.09%	7.92%	7.92%	7.92%	7.92%
		7200.39	1515.70	1014.08	383.43	4495.23	637.73
2933GapC	42262	1.42%	1.57%	1.41%	1.41%	1.41%	1.41%
		7200.60	1195.44	592.13	215.47	2149.64	441.86
3133GapC	42228	10.25%	10.26%	10.06%	10.06%	10.09%	10.06%
		7200.51	3101.87	1359.83	807.96	7222.93	967.32
3233GapC	39277	3.52%	3.64%	3.51%	3.51%	3.51%	3.51%
		7200.89	1554.32	722.07	322.17	2000.83	376.62

Appendix II

Bound strengths (gap %) and computation times (s) for $\frac{1}{4}$ TUFLP-S instances

Results for (PS_{hull}) correspond to the non-stabilised structured Dantzig-Wolfe decomposition in the first four tables. Full results for (PS_{hull}) follow.

		LP Relaxations			Reformulation		Primal	MIP times		
Instance	Exact	(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{4}$-0	37702672	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.02	0.03	0.01	0.11	0.01	0.03	0.03	0.03
$\frac{1}{4}$-1	37703185	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.02	0.01	0.08	0.01	0.03	0.04	0.04
$\frac{1}{4}$-2	37706200	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.00	0.11	0.01	0.03	0.03	0.04
$\frac{1}{4}$-3	37731199	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.10	0.01	0.04	0.04	0.03
$\frac{1}{4}$-4	37940175	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.03	0.00	0.10	0.01	0.04	0.04	0.03
$\frac{1}{4}$-5	38627442	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.10	0.01	0.04	0.03	0.04
$\frac{1}{4}$-6	39432961	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.02	0.01	0.15	0.03	0.04	0.04	0.03
$\frac{1}{4}$-7	40013957	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.02	0.03	0.00	0.08	0.00	0.03	0.04	0.04
$\frac{1}{4}$-8	40385357	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.10	0.00	0.04	0.04	0.04
$\frac{1}{4}$-9	40823665	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.02	0.03	0.01	0.17	0.03	0.04	0.03	0.04
$\frac{1}{4}$-10	41100067	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.02	0.00	0.10	0.01	0.04	0.04	0.04
$\frac{1}{4}$-11	41127212	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.10	0.02	0.04	0.04	0.04
$\frac{1}{4}$-12	41598942	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.01	0.10	0.01	0.03	0.04	0.04
$\frac{1}{4}$-13	41842769	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.01	0.12	0.01	0.04	0.05	0.04
$\frac{1}{4}$-14	41814212	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.00	0.07	0.01	0.04	0.04	0.04
$\frac{1}{4}$-15	42201492	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.01	0.17	0.04	0.04	0.04	0.03
$\frac{1}{4}$-16	42262066	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.00	0.12	0.01	0.04	0.04	0.04
$\frac{1}{4}$-17	42540582	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.02	0.03	0.00	0.10	0.01	0.04	0.04	0.03
$\frac{1}{4}$-18	42577767	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.04	0.01	0.17	0.02	0.04	0.04	0.04
$\frac{1}{4}$-19	42830481	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.02	0.01	0.10	0.00	0.04	0.04	0.04
$\frac{1}{4}$-20	42874155	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.15	0.03	0.04	0.04	0.04
$\frac{1}{4}$-21	43196227	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.02	0.03	0.01	0.16	0.02	0.03	0.03	0.04
$\frac{1}{4}$-22	43207567	0.02%	0.02%	0.02%	0.01%	0.00%	0.00%			
		0.04	0.02	0.04	0.01	0.13	0.01	0.11	0.11	0.10
$\frac{1}{4}$-23	43356354	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.17	0.01	0.04	0.04	0.03
$\frac{1}{4}$-24	43465444	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.04	0.00	0.13	0.01	0.04	0.04	0.04

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{4}$-25	43414860	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.02	0.03	0.01	0.10	0.00	0.04	0.04	0.04
$\frac{1}{4}$-26	43430155	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.04	0.04	0.00	0.16	0.01	0.04	0.04	0.04
$\frac{1}{4}$-27	43640192	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.04	0.00	0.15	0.01	0.04	0.04	0.04
$\frac{1}{4}$-28	43687976	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.04	0.01	0.09	0.01	0.04	0.04	0.04
$\frac{1}{4}$-29	43880477	0.01%	0.01%	0.01%	0.00%	0.00%	0.01%			
		0.04	0.03	0.03	0.01	0.26	0.02	0.12	0.14	0.06
$\frac{1}{4}$-30	44004531	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.03	0.00	0.16	0.02	0.03	0.04	0.04
$\frac{1}{4}$-31	43974501	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.00	0.09	0.01	0.04	0.04	0.04
$\frac{1}{4}$-32	44058492	0.01%	0.01%	0.01%	0.01%	0.00%	0.00%			
		0.03	0.04	0.03	0.01	0.24	0.02	0.05	0.07	0.05
$\frac{1}{4}$-33	44477513	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.01	0.13	0.01	0.04	0.04	0.04
$\frac{1}{4}$-34	44539826	0.02%	0.02%	0.02%	0.01%	0.00%	0.00%			
		0.04	0.04	0.03	0.01	0.24	0.01	0.13	0.12	0.14
$\frac{1}{4}$-35	44917006	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.05	0.04	0.04	0.01	0.22	0.02	0.04	0.04	0.04
$\frac{1}{4}$-36	44850479	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.22	0.00	0.03	0.04	0.04
$\frac{1}{4}$-37	45008608	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.03	0.01	0.17	0.01	0.04	0.06	0.04
$\frac{1}{4}$-38	45002122	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.03	0.01	0.15	0.01	0.04	0.04	0.03
$\frac{1}{4}$-39	45482319	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.04	0.04	0.00	0.25	0.00	0.06	0.13	0.10
$\frac{1}{4}$-40	45426565	0.04%	0.04%	0.04%	0.02%	0.00%	0.00%			
		0.02	0.03	0.03	0.00	0.29	0.04	0.13	0.12	0.11
$\frac{1}{4}$-41	45663313	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.04	0.03	0.01	0.14	0.01	0.14	0.06	0.06
$\frac{1}{4}$-42	45730363	0.01%	0.01%	0.01%	0.01%	0.01%	0.00%			
		0.04	0.03	0.03	0.00	0.23	0.01	0.05	0.15	0.05
$\frac{1}{4}$-43	45872928	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.03	0.01	0.16	0.01	0.04	0.04	0.04
$\frac{1}{4}$-44	45789434	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.03	0.00	0.11	0.00	0.04	0.04	0.03
$\frac{1}{4}$-45	46011542	0.04%	0.04%	0.04%	0.04%	0.00%	0.00%			
		0.02	0.03	0.03	0.01	1.41	0.08	0.14	0.12	0.13
$\frac{1}{4}$-46	46000483	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.24	0.03	0.02	0.04	0.04
$\frac{1}{4}$-47	46080513	0.02%	0.02%	0.02%	0.02%	0.02%	0.00%			
		0.03	0.03	0.04	0.01	0.16	0.02	0.08	0.11	0.14
$\frac{1}{4}$-48	46176847	0.03%	0.03%	0.03%	0.03%	0.01%	0.00%			
		0.04	0.04	0.03	0.01	0.22	0.03	0.13	0.13	0.13
$\frac{1}{4}$-49	46333968	0.03%	0.03%	0.03%	0.03%	0.00%	0.00%			
		0.04	0.04	0.02	0.01	0.28	0.04	0.10	0.09	0.10

Instance	Exact	LP Relaxations (PS_{path})	(PS'_{path})	(PS^0_{path})	Reformulation SUBP	(PS_{hull})	Primal Heuristic	MIP times (PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{4}$-50	46434919	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		0.04	0.03	0.03	0.01	0.26	0.02	0.10	0.10	0.12
$\frac{1}{4}$-51	46420582	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.04	0.03	0.01	0.16	0.01	0.04	0.03	0.04
$\frac{1}{4}$-52	46591657	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.04	0.01	0.15	0.01	0.04	0.04	0.03
$\frac{1}{4}$-53	46515284	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.02	0.03	0.01	0.14	0.02	0.02	0.03	0.03
$\frac{1}{4}$-54	46498260	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.12	0.01	0.04	0.04	0.03
$\frac{1}{4}$-55	46400001	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.04	0.00	0.11	0.01	0.02	0.03	0.04
$\frac{1}{4}$-56	46500403	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.22	0.00	0.04	0.04	0.04
$\frac{1}{4}$-57	46558247	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.04	0.01	0.16	0.02	0.04	0.04	0.04
$\frac{1}{4}$-58	46470512	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.02	0.04	0.00	0.14	0.01	0.03	0.03	0.03
$\frac{1}{4}$-59	46580267	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.02	0.01	0.12	0.01	0.04	0.04	0.03
$\frac{1}{4}$-60	46561835	0.01%	0.01%	0.01%	0.01%	0.01%	0.00%			
		0.04	0.03	0.04	0.01	0.16	0.00	0.09	0.11	0.14
$\frac{1}{4}$-61	46816829	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.02	0.02	0.00	0.16	0.01	0.04	0.04	0.04
$\frac{1}{4}$-62	46883539	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.03	0.01	0.10	0.01	0.02	0.04	0.03
$\frac{1}{4}$-63	46999121	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.04	0.00	0.15	0.01	0.04	0.04	0.04
$\frac{1}{4}$-64	46889139	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.02	0.01	0.12	0.01	0.04	0.04	0.04
$\frac{1}{4}$-65	46841965	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.17	0.02	0.05	0.04	0.04
$\frac{1}{4}$-66	46902461	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.32	0.04	0.07	0.06	0.14
$\frac{1}{4}$-67	47070665	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.03	0.00	0.27	0.04	0.04	0.04	0.04
$\frac{1}{4}$-68	47021412	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.14	0.02	0.03	0.04	0.04
$\frac{1}{4}$-69	47158705	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		0.03	0.04	0.04	0.01	0.15	0.01	0.09	0.14	0.09
$\frac{1}{4}$-70	47197245	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		0.03	0.04	0.03	0.01	0.19	0.01	0.05	0.05	0.05
$\frac{1}{4}$-71	47159434	0.01%	0.01%	0.01%	0.01%	0.00%	0.00%			
		0.04	0.04	0.03	0.01	0.32	0.04	0.03	0.10	0.04
$\frac{1}{4}$-72	47205192	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.03	0.04	0.00	0.09	0.01	0.04	0.04	0.04
$\frac{1}{4}$-73	47269714	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		0.04	0.04	0.02	0.01	0.18	0.02	0.09	0.05	0.05
$\frac{1}{4}$-74	47187123	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.02	0.02	0.01	0.16	0.02	0.04	0.04	0.04

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{4}$-75	47281403	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.04	0.03	0.00	0.09	0.01	0.04	0.04	0.04
$\frac{1}{4}$-76	47378808	0.01%	0.01%	0.01%	0.00%	0.00%	0.01%			
		0.04	0.04	0.04	0.01	0.24	0.02	0.06	0.05	0.05
$\frac{1}{4}$-77	47369887	0.03%	0.03%	0.03%	0.02%	0.01%	0.00%			
		0.03	0.03	0.04	0.00	0.25	0.04	0.14	0.15	0.12
$\frac{1}{4}$-78	47377163	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		0.03	0.04	0.02	0.01	0.16	0.01	0.04	0.05	0.12
$\frac{1}{4}$-79	47514847	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		0.02	0.03	0.03	0.01	0.18	0.01	0.05	0.11	0.04
$\frac{1}{4}$-80	47463123	0.04%	0.04%	0.04%	0.04%	0.00%	0.00%			
		0.04	0.03	0.03	0.01	0.24	0.04	0.05	0.10	0.11
$\frac{1}{4}$-81	47476491	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.25	0.02	0.06	0.06	0.04
$\frac{1}{4}$-82	47587761	0.07%	0.07%	0.07%	0.05%	0.00%	0.00%			
		0.02	0.03	0.03	0.01	0.16	0.02	0.08	0.10	0.10
$\frac{1}{4}$-83	47662881	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.04	0.03	0.01	0.28	0.02	0.05	0.05	0.04
$\frac{1}{4}$-84	47664295	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.26	0.03	0.05	0.05	0.04
$\frac{1}{4}$-85	47677928	0.01%	0.01%	0.01%	0.01%	0.00%	0.00%			
		0.04	0.03	0.04	0.01	0.11	0.00	0.06	0.10	0.11
$\frac{1}{4}$-86	47781796	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.02	0.01	0.11	0.01	0.04	0.04	0.04
$\frac{1}{4}$-87	47804756	0.02%	0.02%	0.02%	0.02%	0.00%	0.00%			
		0.04	0.03	0.03	0.01	0.28	0.04	0.12	0.13	0.11
$\frac{1}{4}$-88	47868075	0.02%	0.02%	0.02%	0.01%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.25	0.04	0.11	0.13	0.12
$\frac{1}{4}$-89	47957918	0.03%	0.03%	0.03%	0.01%	0.00%	0.00%			
		0.04	0.04	0.03	0.01	0.20	0.01	0.11	0.13	0.11
$\frac{1}{4}$-90	47942470	0.05%	0.05%	0.05%	0.02%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.18	0.02	0.11	0.14	0.12
$\frac{1}{4}$-91	48075076	0.05%	0.05%	0.05%	0.05%	0.00%	0.00%			
		0.03	0.04	0.04	0.01	0.25	0.03	0.12	0.13	0.12
$\frac{1}{4}$-92	47902648	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.02	0.03	0.01	0.10	0.01	0.03	0.04	0.03
$\frac{1}{4}$-93	47922814	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.03	0.02	0.01	0.09	0.01	0.04	0.04	0.04
$\frac{1}{4}$-94	47862006	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.02	0.01	0.11	0.01	0.03	0.04	0.04
$\frac{1}{4}$-95	47980570	0.05%	0.05%	0.05%	0.02%	0.00%	0.00%			
		0.03	0.03	0.03	0.00	0.14	0.01	0.10	0.05	0.06
$\frac{1}{4}$-96	48017505	0.06%	0.06%	0.06%	0.02%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.26	0.03	0.12	0.16	0.10
$\frac{1}{4}$-97	47955122	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.03	0.03	0.03	0.01	0.38	0.04	0.04	0.05	0.05
$\frac{1}{4}$-98	48005699	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.04	0.04	0.02	0.01	0.24	0.01	0.03	0.04	0.04
$\frac{1}{4}$-99	48092815	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.02	0.03	0.02	0.01	0.16	0.02	0.04	0.03	0.03

		NDO master		(PS_{hull})			
Instance	Exact	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{1}{4}$-0	37702672	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.07	0.10	0.09	0.10	0.10
$\frac{1}{4}$-1	37703185	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.07	0.10	0.08	0.10	0.10
$\frac{1}{4}$-2	37706200	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.06	0.09	0.10	0.10	0.09
$\frac{1}{4}$-3	37731199	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.07	0.09	0.10	0.08	0.10
$\frac{1}{4}$-4	37940175	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.07	0.10	0.09	0.09	0.10
$\frac{1}{4}$-5	38627442	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.09	0.10	0.09	0.08	0.09
$\frac{1}{4}$-6	39432961	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.06	0.17	0.16	0.19	0.18
$\frac{1}{4}$-7	40013957	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.06	0.09	0.07	0.10	0.09
$\frac{1}{4}$-8	40385357	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.08	0.10	0.10	0.09
$\frac{1}{4}$-9	40823665	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.08	0.18	0.19	0.19	0.19
$\frac{1}{4}$-10	41100067	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.08	0.10	0.09	0.09	0.10
$\frac{1}{4}$-11	41127212	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.08	0.11	0.10	0.12	0.12
$\frac{1}{4}$-12	41598942	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.12	0.11	0.08	0.12	0.11
$\frac{1}{4}$-13	41842769	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.07	0.11	0.10	0.12	0.12
$\frac{1}{4}$-14	41814212	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.07	0.08	0.06	0.09	0.08
$\frac{1}{4}$-15	42201492	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.12	0.19	0.19	0.18	0.18
$\frac{1}{4}$-16	42262066	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.12	0.10	0.12	0.11	0.12
$\frac{1}{4}$-17	42540582	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.06	0.09	0.08	0.10	0.09
$\frac{1}{4}$-18	42577767	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.06	0.16	0.16	0.17	0.17
$\frac{1}{4}$-19	42830481	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.07	0.10	0.08	0.09	0.10
$\frac{1}{4}$-20	42874155	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.11	0.18	0.17	0.19	0.17
$\frac{1}{4}$-21	43196227	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.08	0.16	0.16	0.15	0.16
$\frac{1}{4}$-22	43207567	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.10	0.09	0.15	0.13	0.14	0.15
$\frac{1}{4}$-23	43356354	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.07	0.15	0.18	0.16	0.18
$\frac{1}{4}$-24	43465444	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.08	0.13	0.13	0.14	0.14

Instance	Exact	NDO master Bundle	Volume	(PS_{hull}) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
$\frac{1}{4}$-25	43414860	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.09	0.09	0.07	0.10	0.09
$\frac{1}{4}$-26	43430155	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.10	0.16	0.17	0.16	0.17
$\frac{1}{4}$-27	43640192	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.11	0.17	0.16	0.17	0.18
$\frac{1}{4}$-28	43687976	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.07	0.09	0.08	0.10	0.10
$\frac{1}{4}$-29	43880477	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.26	0.35	0.29	0.30	0.30	0.29
$\frac{1}{4}$-30	44004531	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.07	0.17	0.16	0.16	0.17
$\frac{1}{4}$-31	43974501	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.10	0.10	0.10	0.09
$\frac{1}{4}$-32	44058492	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.16	0.38	0.26	0.26	0.25	0.27
$\frac{1}{4}$-33	44477513	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.08	0.11	0.11	0.11	0.11
$\frac{1}{4}$-34	44539826	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.31	3.55	0.28	0.28	0.29	0.30
$\frac{1}{4}$-35	44917006	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.23	0.21	0.23	0.26
$\frac{1}{4}$-36	44850479	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.11	0.23	0.23	0.24	0.26
$\frac{1}{4}$-37	45008608	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.07	0.16	0.18	0.18	0.19
$\frac{1}{4}$-38	45002122	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.08	0.15	0.16	0.18	0.17
$\frac{1}{4}$-39	45482319	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.06	0.29	0.26	0.42	0.74
$\frac{1}{4}$-40	45426565	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.66	8.34	0.36	0.34	0.39	0.38
$\frac{1}{4}$-41	45663313	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.11	0.12	0.12	0.15	0.15
$\frac{1}{4}$-42	45730363	0.01%	0.01%	0.01%	0.01%	0.01%	0.01%
		0.08	0.31	0.40	0.25	0.34	0.46
$\frac{1}{4}$-43	45872928	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.09	0.17	0.17	0.18	0.18
$\frac{1}{4}$-44	45789434	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.07	0.09	0.08	0.11	0.10
$\frac{1}{4}$-45	46011542	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		0.50	4.34	0.52	1.53	1.17	1.24
$\frac{1}{4}$-46	46000483	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.08	0.34	0.28	0.37	0.30
$\frac{1}{4}$-47	46080513	0.02%	0.02%	0.02%	0.02%	0.02%	0.02%
		0.12	2.75	0.16	0.15	0.16	0.16
$\frac{1}{4}$-48	46176847	0.01%	0.02%	0.01%	0.01%	0.01%	0.01%
		0.26	4.62	0.23	0.24	0.25	0.25
$\frac{1}{4}$-49	46333968	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.33	3.09	0.33	0.31	0.35	0.36

Instance	Exact	NDO master		(PS$_{hull}$)			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{1}{4}$-50	46434919	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.18	0.27	0.28	0.37	0.38
$\frac{1}{4}$-51	46420582	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.11	0.17	0.17	0.18	0.17
$\frac{1}{4}$-52	46591657	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.08	0.17	0.16	0.19	0.18
$\frac{1}{4}$-53	46515284	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.06	0.14	0.15	0.15	0.16
$\frac{1}{4}$-54	46498260	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.08	0.13	0.09	0.12	0.13
$\frac{1}{4}$-55	46400001	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.06	0.10	0.09	0.09	0.09
$\frac{1}{4}$-56	46500403	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.20	0.25	0.26	0.26
$\frac{1}{4}$-57	46558247	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.16	0.16	0.16	0.16	0.16
$\frac{1}{4}$-58	46470512	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.08	0.15	0.15	0.16	0.15
$\frac{1}{4}$-59	46580267	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.12	0.11	0.12	0.12
$\frac{1}{4}$-60	46561835	0.01%	0.01%	0.01%	0.01%	0.01%	0.01%
		0.08	0.06	0.14	0.15	0.14	0.14
$\frac{1}{4}$-61	46816829	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.08	0.16	0.17	0.18	0.17
$\frac{1}{4}$-62	46883539	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.09	0.09	0.07	0.10	0.09
$\frac{1}{4}$-63	46999121	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.12	0.16	0.16	0.17	0.19
$\frac{1}{4}$-64	46889139	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.10	0.11	0.10	0.11
$\frac{1}{4}$-65	46841965	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.15	0.19	0.19	0.20	0.19
$\frac{1}{4}$-66	46902461	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.10	0.39	0.36	0.77	0.71
$\frac{1}{4}$-67	47070665	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.08	0.26	0.28	0.29	0.31
$\frac{1}{4}$-68	47021412	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.10	0.14	0.15	0.17	0.17
$\frac{1}{4}$-69	47158705	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.13	0.16	0.14	0.13	0.15	0.15
$\frac{1}{4}$-70	47197245	0.01%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.15	0.26	0.22	0.21	0.24	0.20
$\frac{1}{4}$-71	47159434	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.12	0.09	0.25	0.35	0.52	0.52
$\frac{1}{4}$-72	47205192	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.07	0.10	0.09	0.10	0.09
$\frac{1}{4}$-73	47269714	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.11	0.35	0.19	0.17	0.20	0.20
$\frac{1}{4}$-74	47187123	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.12	0.16	0.16	0.17	0.17

		NDO master		(PS_{hull})			
Instance	Exact	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
$\frac{1}{4}$-75	47281403	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.11	0.10	0.09	0.09	0.10
$\frac{1}{4}$-76	47378808	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.08	0.30	0.29	0.31	0.29
$\frac{1}{4}$-77	47369887	0.03%	0.02%	0.01%	0.01%	0.01%	0.01%
		0.12	3.72	0.26	0.27	0.32	0.27
$\frac{1}{4}$-78	47377163	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.10	0.10	0.19	0.18	0.22	0.23
$\frac{1}{4}$-79	47514847	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.13	0.19	0.18	0.20	0.19
$\frac{1}{4}$-80	47463123	0.04%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.08	6.10	0.25	0.26	0.28	0.25
$\frac{1}{4}$-81	47476491	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.15	0.12	0.25	0.27	0.40	0.38
$\frac{1}{4}$-82	47587761	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.36	0.33	0.17	0.17	0.19	0.18
$\frac{1}{4}$-83	47662881	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.10	0.26	0.29	0.33	0.32
$\frac{1}{4}$-84	47664295	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.08	0.17	0.27	0.29	0.30	0.31
$\frac{1}{4}$-85	47677928	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.08	0.29	0.12	0.13	0.13	0.12
$\frac{1}{4}$-86	47781796	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.11	0.19	0.09	0.10	0.10	0.10
$\frac{1}{4}$-87	47804756	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.11	0.28	0.29	0.32	0.31	0.29
$\frac{1}{4}$-88	47868075	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.08	3.16	0.28	0.28	0.33	0.28
$\frac{1}{4}$-89	47957918	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.20	0.65	0.23	0.20	0.25	0.22
$\frac{1}{4}$-90	47942470	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.54	4.29	0.21	0.20	0.21	0.20
$\frac{1}{4}$-91	48075076	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.19	4.04	0.33	0.28	0.31	0.28
$\frac{1}{4}$-92	47902648	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.09	0.09	0.09	0.10	0.10	0.10
$\frac{1}{4}$-93	47922814	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.07	0.10	0.10	0.08	0.12	0.10
$\frac{1}{4}$-94	47862006	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.08	0.10	0.12	0.11	0.10
$\frac{1}{4}$-95	47980570	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.11	0.28	0.13	0.13	0.14	0.13
$\frac{1}{4}$-96	48017505	0.01%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.12	0.40	0.27	0.29	0.34	0.29
$\frac{1}{4}$-97	47955122	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.08	0.32	0.43	0.43	0.48
$\frac{1}{4}$-98	48005699	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.10	0.11	0.25	0.25	0.29	0.25
$\frac{1}{4}$-99	48092815	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.06	0.06	0.17	0.14	0.17	0.15

Appendix III

Bound strengths (gap %) and computation times (s) for $\frac{1}{2}$ TUFLP-S instances

Results for (PS_{hull}) correspond to the non-stabilised structured Dantzig-Wolfe decomposition in the first four tables. Full results for (PS_{hull}) follow.

		LP Relaxations			Reformulation		Primal	MIP times		
Instance	Exact	(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{2}$-0	65144759	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.30	0.27	0.28	0.03	0.75	0.06	0.35	0.39	0.38
$\frac{1}{2}$-1	65145763	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.32	0.30	0.28	0.02	0.83	0.06	0.33	0.46	0.38
$\frac{1}{2}$-2	65145137	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.31	0.36	0.26	0.03	0.83	0.05	0.40	0.40	0.40
$\frac{1}{2}$-3	65145026	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.34	0.30	0.25	0.03	0.86	0.06	0.45	0.38	0.35
$\frac{1}{2}$-4	65147513	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.30	0.35	0.28	0.02	0.80	0.06	0.36	0.42	0.39
$\frac{1}{2}$-5	65172370	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.34	0.34	0.26	0.03	0.80	0.05	0.44	0.43	0.34
$\frac{1}{2}$-6	65406245	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.29	0.49	0.32	0.03	0.90	0.05	0.51	0.59	0.43
$\frac{1}{2}$-7	66948834	0.08%	0.08%	0.08%	0.06%	0.00%	0.00%			
		0.43	0.38	0.31	0.05	1.70	0.24	1.20	1.79	1.82
$\frac{1}{2}$-8	67820933	0.09%	0.09%	0.09%	0.08%	0.00%	0.00%			
		0.38	0.30	0.33	0.04	1.30	0.13	1.88	1.30	1.50
$\frac{1}{2}$-9	68120430	0.29%	0.29%	0.29%	0.22%	0.00%	0.00%			
		0.50	0.50	0.34	0.06	1.60	0.14	2.18	1.68	1.92
$\frac{1}{2}$-10	67994192	0.24%	0.24%	0.25%	0.16%	0.00%	0.00%			
		0.36	0.32	0.30	0.05	1.28	0.13	1.84	1.44	1.62
$\frac{1}{2}$-11	67975228	0.06%	0.06%	0.06%	0.05%	0.00%	0.00%			
		0.36	0.32	0.32	0.05	1.68	0.22	1.69	1.36	1.86
$\frac{1}{2}$-12	68846428	0.27%	0.27%	0.27%	0.24%	0.00%	0.00%			
		0.41	0.36	0.33	0.09	1.82	0.20	2.05	1.36	2.03
$\frac{1}{2}$-13	68757575	0.18%	0.18%	0.18%	0.03%	0.00%	0.00%			
		0.36	0.32	0.31	0.04	1.31	0.14	1.71	1.31	1.84
$\frac{1}{2}$-14	69155968	0.05%	0.05%	0.05%	0.05%	0.00%	0.00%			
		0.44	0.37	0.40	0.04	1.37	0.15	1.73	2.10	1.74
$\frac{1}{2}$-15	69333921	0.03%	0.03%	0.03%	0.01%	0.00%	0.00%			
		0.41	0.38	0.42	0.12	1.88	0.14	1.68	1.48	1.75
$\frac{1}{2}$-16	69568996	0.37%	0.37%	0.37%	0.29%	0.00%	0.00%			
		0.53	0.41	0.42	0.09	2.41	0.31	3.40	1.98	2.71
$\frac{1}{2}$-17	69687018	0.22%	0.22%	0.22%	0.16%	0.00%	0.00%			
		0.48	0.47	0.41	0.12	2.16	0.20	2.54	2.02	2.07
$\frac{1}{2}$-18	69994151	0.34%	0.34%	0.34%	0.21%	0.00%	0.00%			
		0.50	0.43	0.42	0.11	1.63	0.15	2.47	2.00	2.48
$\frac{1}{2}$-19	70384403	0.08%	0.08%	0.08%	0.08%	0.00%	0.00%			
		0.52	0.37	0.44	0.04	1.98	0.24	2.07	0.67	1.88
$\frac{1}{2}$-20	70450169	0.28%	0.28%	0.28%	0.20%	0.00%	0.00%			
		0.48	0.39	0.37	0.07	1.58	0.14	2.40	1.71	2.14
$\frac{1}{2}$-21	70356065	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		0.40	0.50	0.41	0.05	1.62	0.08	0.47	0.56	0.47
$\frac{1}{2}$-22	70331207	0.19%	0.19%	0.19%	0.12%	0.00%	0.00%			
		0.38	0.37	0.34	0.05	1.86	0.13	2.02	1.52	2.06
$\frac{1}{2}$-23	71146525	0.38%	0.38%	0.38%	0.34%	0.00%	0.00%			
		0.38	0.37	0.43	0.11	2.63	0.19	2.34	1.96	2.39
$\frac{1}{2}$-24	71218806	0.39%	0.39%	0.39%	0.33%	0.00%	0.00%			
		0.58	0.47	0.57	0.12	1.71	0.17	3.03	3.45	2.42

		LP Relaxations			Reformulation		Primal	MIP times		
Instance	Exact	$(\mathrm{PS}_{\mathrm{path}})$	$(\mathrm{PS}'_{\mathrm{path}})$	$(\mathrm{PS}^0_{\mathrm{path}})$	SUBP	$(\mathrm{PS}_{\mathrm{hull}})$	Heuristic	$(\mathrm{PS}_{\mathrm{path}})$	$(\mathrm{PS}'_{\mathrm{path}})$	$(\mathrm{PS}^0_{\mathrm{path}})$
$\frac{1}{2}$-25	71254809	0.26%	0.26%	0.26%	0.21%	0.00%	0.00%			
		0.53	0.46	0.54	0.08	2.68	0.38	2.84	2.99	2.38
$\frac{1}{2}$-26	71240359	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		0.39	0.56	0.38	0.06	1.20	0.06	2.18	1.03	1.64
$\frac{1}{2}$-27	71598985	0.24%	0.24%	0.24%	0.16%	0.00%	0.00%			
		0.50	0.60	0.44	0.05	6.02	0.35	3.00	1.57	1.88
$\frac{1}{2}$-28	71896041	0.25%	0.25%	0.25%	0.20%	0.00%	0.00%			
		0.62	0.47	0.47	0.06	2.25	0.22	2.61	1.92	2.20
$\frac{1}{2}$-29	71968215	0.24%	0.24%	0.24%	0.18%	0.00%	0.00%			
		0.50	0.78	0.42	0.12	2.13	0.27	3.20	3.46	2.54
$\frac{1}{2}$-30	71795909	0.25%	0.25%	0.25%	0.23%	0.00%	0.00%			
		0.49	0.54	0.39	0.12	2.56	0.28	2.82	2.08	2.55
$\frac{1}{2}$-31	72036365	0.30%	0.30%	0.30%	0.26%	0.00%	0.00%			
		0.52	0.49	0.40	0.15	2.23	0.24	2.92	3.02	2.69
$\frac{1}{2}$-32	72097464	0.35%	0.35%	0.35%	0.28%	0.00%	0.00%			
		0.47	0.44	0.45	0.14	2.56	0.30	2.99	3.10	2.78
$\frac{1}{2}$-33	72132498	0.38%	0.38%	0.38%	0.26%	0.00%	0.00%			
		0.55	0.59	0.50	0.16	2.91	0.37	3.06	3.21	2.44
$\frac{1}{2}$-34	72161229	0.29%	0.29%	0.29%	0.28%	0.00%	0.00%			
		0.43	0.47	0.36	0.05	1.80	0.22	2.43	1.61	2.18
$\frac{1}{2}$-35	72461817	0.39%	0.39%	0.39%	0.38%	0.00%	0.00%			
		0.64	0.50	0.53	0.09	4.43	0.55	3.25	2.85	2.44
$\frac{1}{2}$-36	72505749	0.16%	0.16%	0.16%	0.09%	0.00%	0.00%			
		0.49	0.61	0.44	0.10	2.01	0.18	3.02	2.15	2.22
$\frac{1}{2}$-37	72559408	0.17%	0.17%	0.17%	0.11%	0.00%	0.00%			
		0.48	0.53	0.46	0.04	2.10	0.26	1.96	1.77	1.99
$\frac{1}{2}$-38	72774159	0.21%	0.21%	0.21%	0.16%	0.00%	0.00%			
		0.45	0.49	0.42	0.14	3.01	0.35	2.82	2.15	2.61
$\frac{1}{2}$-39	73017224	0.44%	0.44%	0.44%	0.37%	0.01%	0.01%			
		0.54	0.51	0.47	0.14	3.26	0.33	3.13	3.28	2.52
$\frac{1}{2}$-40	72981934	0.48%	0.48%	0.49%	0.41%	0.00%	0.00%			
		0.51	0.58	0.48	0.13	2.25	0.21	3.03	3.38	2.60
$\frac{1}{2}$-41	73124728	0.54%	0.54%	0.54%	0.37%	0.00%	0.00%			
		0.60	0.49	0.60	0.16	2.41	0.22	3.89	3.28	3.98
$\frac{1}{2}$-42	73005050	0.52%	0.52%	0.52%	0.44%	0.00%	0.00%			
		0.42	0.38	0.45	0.18	2.29	0.20	2.81	2.48	2.71
$\frac{1}{2}$-43	73184610	0.49%	0.49%	0.49%	0.43%	0.00%	0.00%			
		0.66	0.61	0.63	0.17	3.27	0.34	4.03	3.59	3.09
$\frac{1}{2}$-44	73290266	0.57%	0.57%	0.57%	0.55%	0.00%	0.00%			
		0.56	0.53	0.62	0.08	3.76	0.37	3.70	4.93	2.94
$\frac{1}{2}$-45	73416844	0.46%	0.46%	0.46%	0.42%	0.00%	0.00%			
		0.44	0.44	0.51	0.12	2.20	0.22	2.11	2.86	2.20
$\frac{1}{2}$-46	73539630	0.55%	0.55%	0.55%	0.46%	0.00%	0.00%			
		0.62	0.50	0.54	0.16	4.02	0.42	4.53	4.99	3.14
$\frac{1}{2}$-47	73590399	0.56%	0.56%	0.56%	0.50%	0.00%	0.00%			
		0.60	0.51	0.53	0.15	3.74	0.48	5.61	5.62	3.04
$\frac{1}{2}$-48	73592122	0.53%	0.53%	0.53%	0.43%	0.00%	0.00%			
		0.53	0.47	0.51	0.18	2.73	0.27	2.91	2.41	2.61
$\frac{1}{2}$-49	73883808	0.40%	0.40%	0.40%	0.34%	0.00%	0.00%			
		0.64	0.60	0.56	0.16	2.50	0.22	4.11	3.80	3.94

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{1}{2}$-50	73973073	0.53%	0.53%	0.53%	0.45%	0.00%	0.00%			
		0.58	0.70	0.57	0.23	3.10	0.20	3.59	4.26	3.62
$\frac{1}{2}$-51	73982838	0.59%	0.59%	0.59%	0.48%	0.00%	0.00%			
		0.63	0.53	0.54	0.17	2.40	0.19	3.22	4.70	3.36
$\frac{1}{2}$-52	73946741	0.57%	0.57%	0.58%	0.40%	0.00%	0.00%			
		0.63	0.70	0.56	0.21	1.91	0.13	3.41	5.38	4.00
$\frac{1}{2}$-53	73876522	0.59%	0.59%	0.59%	0.51%	0.00%	0.00%			
		0.47	0.44	0.48	0.06	3.47	0.39	5.24	3.58	2.88
$\frac{1}{2}$-54	74095422	0.67%	0.67%	0.67%	0.61%	0.00%	0.00%			
		0.54	0.61	0.54	0.26	4.14	0.37	4.93	11.37	3.70
$\frac{1}{2}$-55	74119744	0.72%	0.72%	0.72%	0.63%	0.00%	0.00%			
		0.50	0.82	0.47	0.22	5.36	0.58	4.47	6.85	3.17
$\frac{1}{2}$-56	74266880	0.37%	0.37%	0.37%	0.32%	0.00%	0.00%			
		0.58	0.51	0.52	0.13	2.77	0.35	5.20	4.02	3.54
$\frac{1}{2}$-57	74247682	0.40%	0.40%	0.40%	0.28%	0.00%	0.00%			
		0.66	0.52	0.57	0.22	3.70	0.24	4.04	6.22	3.94
$\frac{1}{2}$-58	74333478	0.58%	0.58%	0.58%	0.50%	0.00%	0.00%			
		0.68	0.96	0.55	0.17	5.95	0.53	4.38	7.57	3.31
$\frac{1}{2}$-59	74290126	0.67%	0.67%	0.67%	0.49%	0.01%	0.00%			
		0.48	0.53	0.46	0.14	10.61	0.87	3.96	6.42	4.18
$\frac{1}{2}$-60	74514411	0.62%	0.62%	0.62%	0.45%	0.00%	0.00%			
		0.50	0.51	0.53	0.14	4.03	0.49	3.70	5.62	2.56
$\frac{1}{2}$-61	74561984	0.63%	0.63%	0.63%	0.54%	0.00%	0.00%			
		0.62	0.55	0.63	0.10	1.86	0.16	4.11	7.10	3.38
$\frac{1}{2}$-62	74539329	0.54%	0.54%	0.54%	0.47%	0.00%	0.00%			
		0.53	0.61	0.48	0.09	2.60	0.24	5.70	10.61	4.01
$\frac{1}{2}$-63	74841349	0.57%	0.57%	0.57%	0.47%	0.00%	0.00%			
		0.56	0.46	0.63	0.10	3.08	0.30	5.43	4.46	3.66
$\frac{1}{2}$-64	74796372	0.48%	0.48%	0.48%	0.43%	0.00%	0.00%			
		0.57	0.55	0.55	0.14	4.38	0.34	4.50	5.52	4.03
$\frac{1}{2}$-65	75029390	0.52%	0.52%	0.52%	0.45%	0.00%	0.00%			
		0.69	0.76	0.58	0.12	2.69	0.24	5.94	5.62	3.79
$\frac{1}{2}$-66	75056224	0.62%	0.62%	0.62%	0.57%	0.00%	0.00%			
		0.84	0.82	0.57	0.13	3.90	0.41	4.86	6.83	3.59
$\frac{1}{2}$-67	75078289	0.62%	0.62%	0.62%	0.47%	0.00%	0.00%			
		0.67	0.53	0.60	0.13	2.63	0.23	4.62	9.52	4.04
$\frac{1}{2}$-68	75090225	0.63%	0.63%	0.63%	0.56%	0.00%	0.00%			
		0.69	0.72	0.54	0.09	3.40	0.33	4.94	5.78	4.84
$\frac{1}{2}$-69	75041538	0.60%	0.60%	0.60%	0.57%	0.00%	0.00%			
		0.68	0.86	0.55	0.08	2.88	0.35	4.75	4.16	2.90
$\frac{1}{2}$-70	75141753	0.57%	0.57%	0.57%	0.50%	0.00%	0.00%			
		0.48	0.54	0.45	0.22	3.17	0.30	4.19	5.06	4.05
$\frac{1}{2}$-71	75222131	0.65%	0.65%	0.65%	0.59%	0.00%	0.00%			
		0.55	0.74	0.57	0.14	4.33	0.38	7.28	6.39	4.47
$\frac{1}{2}$-72	75187115	0.56%	0.56%	0.56%	0.45%	0.00%	0.00%			
		0.60	0.74	0.49	0.33	4.00	0.38	5.58	5.12	4.11
$\frac{1}{2}$-73	75240905	0.63%	0.63%	0.63%	0.51%	0.00%	0.00%			
		0.64	0.70	0.58	0.13	3.26	0.28	5.65	13.37	5.04
$\frac{1}{2}$-74	75303717	0.63%	0.63%	0.63%	0.51%	0.00%	0.00%			
		0.56	0.55	0.48	0.20	3.32	0.30	5.19	8.58	4.74

Instance	Exact	LP Relaxations (PS_{path})	LP Relaxations (PS'_{path})	LP Relaxations (PS^0_{path})	Reformulation SUBP	Reformulation (PS_{hull})	Primal Heuristic	MIP times (PS_{path})	MIP times (PS'_{path})	MIP times (PS^0_{path})
$\frac{1}{2}$-75	75192972	0.57%	0.57%	0.57%	0.52%	0.00%	0.00%			
		0.63	0.52	0.50	0.16	3.14	0.27	4.70	5.76	3.38
$\frac{1}{2}$-76	75427333	0.59%	0.59%	0.59%	0.54%	0.00%	0.00%			
		0.52	0.48	0.52	0.10	3.07	0.30	4.47	4.86	4.54
$\frac{1}{2}$-77	75485131	0.68%	0.68%	0.68%	0.65%	0.00%	0.00%			
		0.62	0.75	0.42	0.14	5.02	0.54	6.11	4.98	5.92
$\frac{1}{2}$-78	75508478	0.73%	0.73%	0.73%	0.64%	0.00%	0.00%			
		0.62	0.62	0.56	0.15	8.28	0.57	8.45	16.00	3.61
$\frac{1}{2}$-79	75613523	0.83%	0.83%	0.84%	0.71%	0.00%	0.00%			
		0.51	0.58	0.49	0.10	5.54	0.48	7.76	8.24	6.80
$\frac{1}{2}$-80	75681223	0.75%	0.75%	0.75%	0.69%	0.00%	0.00%			
		0.68	0.53	0.58	0.12	5.93	0.59	7.81	8.16	4.05
$\frac{1}{2}$-81	75701267	0.77%	0.77%	0.77%	0.68%	0.00%	0.00%			
		0.72	0.52	0.65	0.09	7.12	0.56	8.94	17.52	5.42
$\frac{1}{2}$-82	75515872	0.66%	0.66%	0.66%	0.63%	0.00%	0.00%			
		0.55	0.60	0.46	0.09	3.72	0.43	3.92	10.13	3.23
$\frac{1}{2}$-83	75550592	0.69%	0.69%	0.69%	0.64%	0.00%	0.00%			
		0.58	0.61	0.48	0.09	4.41	0.48	5.25	4.02	3.65
$\frac{1}{2}$-84	75525669	0.46%	0.46%	0.46%	0.37%	0.00%	0.00%			
		0.65	0.54	0.60	0.20	3.40	0.32	3.90	4.29	3.30
$\frac{1}{2}$-85	75707486	0.60%	0.60%	0.60%	0.55%	0.00%	0.00%			
		0.84	0.69	0.69	0.13	3.54	0.32	5.74	9.71	3.52
$\frac{1}{2}$-86	75622388	0.61%	0.61%	0.61%	0.44%	0.00%	0.00%			
		0.60	0.56	0.51	0.12	3.20	0.31	3.18	5.44	3.44
$\frac{1}{2}$-87	75678850	0.62%	0.62%	0.62%	0.45%	0.00%	0.00%			
		0.62	0.65	0.54	0.11	3.50	0.28	5.68	7.05	4.32
$\frac{1}{2}$-88	75783345	0.54%	0.54%	0.54%	0.26%	0.00%	0.00%			
		0.74	0.76	0.60	0.16	5.58	0.18	4.46	5.80	3.45
$\frac{1}{2}$-89	75877234	0.70%	0.70%	0.70%	0.60%	0.00%	0.00%			
		0.50	0.54	0.43	0.14	6.82	0.28	3.89	5.96	3.47
$\frac{1}{2}$-90	75816488	0.54%	0.54%	0.54%	0.43%	0.01%	0.00%			
		0.58	0.55	0.53	0.14	2.43	0.23	4.39	5.80	3.63
$\frac{1}{2}$-91	75823903	0.58%	0.58%	0.58%	0.51%	0.00%	0.00%			
		0.71	0.50	0.64	0.18	2.63	0.22	3.98	4.11	3.84
$\frac{1}{2}$-92	75985392	0.59%	0.59%	0.59%	0.52%	0.00%	0.00%			
		0.62	0.56	0.53	0.12	3.20	0.22	4.16	4.97	3.56
$\frac{1}{2}$-93	75968366	0.53%	0.53%	0.53%	0.48%	0.00%	0.00%			
		0.68	0.62	0.68	0.10	3.31	0.30	5.04	5.82	4.27
$\frac{1}{2}$-94	76028460	0.66%	0.66%	0.66%	0.55%	0.00%	0.00%			
		0.64	0.66	0.53	0.16	3.56	0.31	4.84	7.33	4.56
$\frac{1}{2}$-95	75903771	0.57%	0.57%	0.57%	0.46%	0.00%	0.00%			
		0.47	0.46	0.52	0.23	3.03	0.28	3.94	3.75	3.57
$\frac{1}{2}$-96	75988719	0.66%	0.66%	0.66%	0.42%	0.00%	0.00%			
		0.53	1.11	0.51	0.18	4.08	0.39	4.66	8.34	3.76
$\frac{1}{2}$-97	75984046	0.63%	0.63%	0.63%	0.57%	0.00%	0.00%			
		0.53	0.61	0.47	0.17	2.92	0.27	4.56	4.76	4.02
$\frac{1}{2}$-98	76025585	0.67%	0.67%	0.67%	0.61%	0.00%	0.00%			
		0.51	0.56	0.46	0.14	3.32	0.31	4.06	5.64	3.44
$\frac{1}{2}$-99	75986166	0.58%	0.58%	0.58%	0.52%	0.00%	0.00%			
		0.56	0.56	0.50	0.10	3.86	0.37	4.57	5.17	2.84

Instance	Exact	NDO master		(PS_{hull})			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{1}{2}$-0	65144759	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.55	0.49	0.71	0.65	1.09	0.84
$\frac{1}{2}$-1	65145763	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.51	0.48	0.73	0.69	0.79	0.76
$\frac{1}{2}$-2	65145137	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.51	0.51	0.86	0.76	0.84	0.81
$\frac{1}{2}$-3	65145026	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.65	0.58	0.75	0.77	0.83	0.74
$\frac{1}{2}$-4	65147513	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.55	0.55	0.72	0.72	0.77	0.72
$\frac{1}{2}$-5	65172370	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.57	0.59	0.73	0.71	0.78	0.76
$\frac{1}{2}$-6	65406245	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.58	0.60	0.74	0.85	0.92	0.75
$\frac{1}{2}$-7	66948834	0.05%	0.02%	0.00%	0.00%	0.00%	0.00%
		1.00	22.73	2.10	1.89	2.25	2.07
$\frac{1}{2}$-8	67820933	0.05%	0.03%	0.00%	0.00%	0.00%	0.00%
		1.98	42.91	1.38	1.38	1.54	1.40
$\frac{1}{2}$-9	68120430	0.10%	0.01%	0.00%	0.00%	0.00%	0.00%
		11.71	50.56	1.46	1.70	1.64	1.59
$\frac{1}{2}$-10	67994192	0.22%	0.06%	0.00%	0.00%	0.00%	0.00%
		0.70	79.04	1.42	1.35	1.58	1.48
$\frac{1}{2}$-11	67975228	0.05%	0.03%	0.00%	0.00%	0.00%	0.00%
		0.88	38.79	2.07	1.91	2.12	2.26
$\frac{1}{2}$-12	68846428	0.17%	0.02%	0.00%	0.00%	0.00%	0.00%
		2.10	81.18	2.10	2.04	2.20	2.31
$\frac{1}{2}$-13	68757575	0.08%	0.03%	0.00%	0.00%	0.00%	0.00%
		1.66	58.15	1.37	1.38	1.66	1.63
$\frac{1}{2}$-14	69155968	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		0.85	0.73	1.60	1.50	1.69	1.66
$\frac{1}{2}$-15	69333921	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		2.00	6.41	1.89	2.43	2.12	1.92
$\frac{1}{2}$-16	69568996	0.00%	0.03%	0.00%	0.00%	0.00%	0.00%
		44.70	59.83	3.17	2.79	3.11	3.03
$\frac{1}{2}$-17	69687018	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		13.40	25.43	2.51	2.60	2.59	2.93
$\frac{1}{2}$-18	69994151	0.13%	0.03%	0.00%	0.00%	0.00%	0.00%
		6.90	98.34	1.92	1.91	2.18	2.65
$\frac{1}{2}$-19	70384403	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		4.03	36.37	2.17	2.29	2.55	2.36
$\frac{1}{2}$-20	70450169	0.19%	0.03%	0.00%	0.00%	0.00%	0.00%
		3.76	96.01	1.88	1.82	1.87	2.52
$\frac{1}{2}$-21	70356065	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		1.03	1.08	2.05	2.02	1.99	2.09
$\frac{1}{2}$-22	70331207	0.19%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.87	33.33	1.92	2.07	2.06	2.59
$\frac{1}{2}$-23	71146525	0.26%	0.01%	0.00%	0.00%	0.00%	0.00%
		10.08	127.53	3.07	3.06	3.05	3.17
$\frac{1}{2}$-24	71218806	0.27%	0.01%	0.00%	0.00%	0.00%	0.00%
		12.53	86.52	2.14	2.06	1.98	2.24

		NDO master		(PS_{hull})			
Instance	Exact	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
$\frac{1}{2}$-25	71254809	0.14%	0.01%	0.00%	0.00%	0.00%	0.00%
		5.53	66.35	3.20	3.20	3.28	3.33
$\frac{1}{2}$-26	71240359	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.90	3.56	1.14	1.49	1.20	1.29
$\frac{1}{2}$-27	71598985	0.15%	0.03%	0.00%	0.00%	0.00%	0.00%
		1.49	75.66	3.30	6.53	5.15	6.40
$\frac{1}{2}$-28	71896041	0.03%	0.02%	0.00%	0.00%	0.00%	0.00%
		31.78	51.34	2.42	2.65	2.45	2.44
$\frac{1}{2}$-29	71968215	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		8.02	68.94	2.70	2.58	2.93	3.08
$\frac{1}{2}$-30	71795909	0.21%	0.02%	0.00%	0.00%	0.00%	0.00%
		1.67	79.47	3.08	3.05	3.13	3.30
$\frac{1}{2}$-31	72036365	0.04%	0.03%	0.00%	0.00%	0.00%	0.00%
		41.42	151.14	2.86	2.68	2.84	3.00
$\frac{1}{2}$-32	72097464	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		51.52	197.55	3.05	3.03	3.28	3.14
$\frac{1}{2}$-33	72132498	0.09%	0.02%	0.00%	0.00%	0.00%	0.00%
		48.18	146.34	3.65	3.62	3.75	3.81
$\frac{1}{2}$-34	72161229	0.14%	0.02%	0.00%	0.00%	0.00%	0.00%
		8.67	63.08	2.11	2.07	2.16	2.15
$\frac{1}{2}$-35	72461817	0.17%	0.04%	0.00%	0.00%	0.00%	0.00%
		23.30	222.92	4.93	5.05	5.08	5.47
$\frac{1}{2}$-36	72505749	0.11%	0.01%	0.00%	0.00%	0.00%	0.00%
		0.91	32.87	2.34	2.34	2.50	2.55
$\frac{1}{2}$-37	72559408	0.16%	0.02%	0.00%	0.00%	0.00%	0.00%
		1.20	61.82	2.65	2.56	2.79	2.74
$\frac{1}{2}$-38	72774159	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		14.69	55.70	4.01	3.61	3.71	3.70
$\frac{1}{2}$-39	73017224	0.26%	0.04%	0.01%	0.01%	0.01%	0.01%
		14.47	152.71	3.86	3.87	3.96	4.20
$\frac{1}{2}$-40	72981934	0.23%	0.04%	0.00%	0.00%	0.00%	0.00%
		18.54	218.31	3.12	2.79	2.79	2.82
$\frac{1}{2}$-41	73124728	0.10%	0.02%	0.00%	0.00%	0.00%	0.00%
		126.23	235.18	2.97	2.98	3.23	3.28
$\frac{1}{2}$-42	73005050	0.22%	0.02%	0.00%	0.00%	0.00%	0.00%
		39.59	94.31	2.58	2.75	2.84	2.78
$\frac{1}{2}$-43	73184610	0.09%	0.01%	0.00%	0.00%	0.00%	0.00%
		97.01	285.81	4.44	3.95	4.18	4.24
$\frac{1}{2}$-44	73290266	0.23%	0.03%	0.00%	0.00%	0.00%	0.00%
		87.60	204.24	4.24	4.33	4.93	5.03
$\frac{1}{2}$-45	73416844	0.21%	0.03%	0.00%	0.00%	0.00%	0.00%
		24.22	157.34	2.63	2.59	2.84	2.95
$\frac{1}{2}$-46	73539630	0.09%	0.04%	0.00%	0.00%	0.00%	0.00%
		77.23	221.20	5.12	4.64	5.05	5.02
$\frac{1}{2}$-47	73590399	0.19%	0.01%	0.00%	0.00%	0.00%	0.00%
		39.73	231.61	4.91	4.45	5.46	4.83
$\frac{1}{2}$-48	73592122	0.39%	0.01%	0.00%	0.00%	0.00%	0.00%
		3.39	92.86	3.29	3.25	3.81	4.16
$\frac{1}{2}$-49	73883808	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		151.37	112.30	3.03	3.05	3.34	3.33

Instance	Exact	NDO master Bundle	Volume	(PS$_{hull}$) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
$\frac{1}{2}$-50	73973073	0.20%	0.00%	0.00%	0.00%	0.00%	0.00%
		31.28	299.60	3.50	3.93	4.32	3.54
$\frac{1}{2}$-51	73982838	0.23%	0.03%	0.00%	0.00%	0.00%	0.00%
		51.31	310.54	2.86	2.84	2.89	3.01
$\frac{1}{2}$-52	73946741	0.25%	0.01%	0.00%	0.00%	0.00%	0.00%
		34.61	171.68	2.43	2.38	2.88	2.64
$\frac{1}{2}$-53	73876522	0.34%	0.02%	0.00%	0.00%	0.00%	0.00%
		31.54	180.91	4.08	4.07	4.16	4.21
$\frac{1}{2}$-54	74095422	0.32%	0.02%	0.00%	0.00%	0.00%	0.00%
		47.12	274.62	4.94	4.99	5.26	5.16
$\frac{1}{2}$-55	74119744	0.40%	0.02%	0.00%	0.00%	0.00%	0.00%
		44.93	328.02	6.38	6.30	6.68	6.46
$\frac{1}{2}$-56	74266880	0.17%	0.03%	0.00%	0.00%	0.00%	0.00%
		20.27	178.42	3.26	3.22	3.48	3.40
$\frac{1}{2}$-57	74247682	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		155.30	304.14	4.32	4.66	4.48	4.16
$\frac{1}{2}$-58	74333478	0.07%	0.01%	0.00%	0.00%	0.00%	0.00%
		162.75	276.44	6.82	6.82	7.16	7.22
$\frac{1}{2}$-59	74290126	0.28%	0.03%	0.01%	0.01%	0.01%	0.01%
		22.17	270.86	8.89	11.71	9.73	10.02
$\frac{1}{2}$-60	74514411	0.12%	0.04%	0.00%	0.00%	0.00%	0.00%
		35.57	191.26	4.86	4.78	5.33	4.92
$\frac{1}{2}$-61	74561984	0.22%	0.03%	0.00%	0.00%	0.00%	0.00%
		21.67	231.93	2.06	2.22	2.25	2.29
$\frac{1}{2}$-62	74539329	0.10%	0.01%	0.00%	0.00%	0.00%	0.00%
		159.47	251.78	3.49	3.14	3.44	3.26
$\frac{1}{2}$-63	74841349	0.16%	0.01%	0.00%	0.00%	0.00%	0.00%
		64.74	101.07	3.70	3.73	3.93	3.50
$\frac{1}{2}$-64	74796372	0.15%	0.04%	0.00%	0.00%	0.00%	0.00%
		27.43	195.71	5.15	5.16	5.18	5.24
$\frac{1}{2}$-65	75029390	0.20%	0.00%	0.00%	0.00%	0.00%	0.00%
		60.98	104.17	3.62	3.32	3.40	3.75
$\frac{1}{2}$-66	75056224	0.38%	0.01%	0.00%	0.00%	0.00%	0.00%
		21.64	193.85	4.67	4.64	4.88	4.77
$\frac{1}{2}$-67	75078289	0.24%	0.01%	0.00%	0.00%	0.00%	0.00%
		30.85	105.49	3.55	3.18	3.32	3.51
$\frac{1}{2}$-68	75090225	0.47%	0.01%	0.00%	0.00%	0.00%	0.00%
		4.52	94.23	4.03	4.06	3.98	4.11
$\frac{1}{2}$-69	75041538	0.15%	0.03%	0.00%	0.00%	0.00%	0.00%
		77.73	183.91	3.43	3.34	3.89	3.63
$\frac{1}{2}$-70	75141753	0.24%	0.05%	0.00%	0.00%	0.00%	0.00%
		28.59	224.75	3.98	3.70	4.03	3.74
$\frac{1}{2}$-71	75222131	0.21%	0.02%	0.00%	0.00%	0.00%	0.00%
		70.89	315.70	4.82	4.99	4.79	4.58
$\frac{1}{2}$-72	75187115	0.22%	0.02%	0.00%	0.00%	0.00%	0.00%
		98.87	275.83	4.80	4.71	6.37	5.14
$\frac{1}{2}$-73	75240905	0.23%	0.03%	0.00%	0.00%	0.00%	0.00%
		89.18	285.51	4.19	4.04	4.09	3.96
$\frac{1}{2}$-74	75303717	0.12%	0.01%	0.00%	0.00%	0.00%	0.00%
		124.48	211.01	4.30	4.02	4.48	4.21

Instance	Exact	NDO master Bundle	Volume	(PS_{hull}) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
$\frac{1}{2}$-75	75192972	0.42%	0.03%	0.00%	0.00%	0.00%	0.00%
		4.49	213.51	3.80	3.69	3.85	3.70
$\frac{1}{2}$-76	75427333	0.39%	0.03%	0.00%	0.00%	0.00%	0.00%
		6.33	282.63	4.00	3.53	3.66	3.78
$\frac{1}{2}$-77	75485131	0.39%	0.01%	0.00%	0.00%	0.00%	0.00%
		21.01	177.46	5.03	5.81	5.15	5.39
$\frac{1}{2}$-78	75508478	0.37%	0.05%	0.00%	0.00%	0.00%	0.00%
		73.46	263.81	9.37	9.34	10.25	9.50
$\frac{1}{2}$-79	75613523	0.53%	0.02%	0.00%	0.00%	0.00%	0.00%
		11.59	225.13	6.21	6.32	6.20	6.34
$\frac{1}{2}$-80	75681223	0.31%	0.03%	0.00%	0.00%	0.00%	0.00%
		50.45	165.31	7.51	6.78	7.96	6.76
$\frac{1}{2}$-81	75701267	0.37%	0.04%	0.00%	0.00%	0.00%	0.00%
		41.26	417.51	7.63	8.11	7.89	7.69
$\frac{1}{2}$-82	75515872	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		193.88	159.29	4.33	4.28	4.37	4.35
$\frac{1}{2}$-83	75550592	0.02%	0.03%	0.00%	0.00%	0.00%	0.00%
		180.40	163.45	5.14	5.06	5.62	4.92
$\frac{1}{2}$-84	75525669	0.01%	0.04%	0.00%	0.00%	0.00%	0.00%
		130.79	218.78	3.76	4.07	4.36	3.96
$\frac{1}{2}$-85	75707486	0.02%	0.03%	0.00%	0.00%	0.00%	0.00%
		144.44	137.88	4.28	4.37	4.28	4.41
$\frac{1}{2}$-86	75622388	0.24%	0.04%	0.00%	0.00%	0.00%	0.00%
		64.22	226.82	3.89	3.82	3.74	3.78
$\frac{1}{2}$-87	75678850	0.04%	0.04%	0.00%	0.00%	0.00%	0.00%
		147.86	223.31	4.68	4.15	4.12	4.39
$\frac{1}{2}$-88	75783345	0.17%	0.02%	0.00%	0.00%	0.00%	0.00%
		80.65	185.32	4.58	6.53	8.20	7.66
$\frac{1}{2}$-89	75877234	0.20%	0.02%	0.00%	0.00%	0.00%	0.00%
		79.19	224.57	6.32	7.56	9.06	8.04
$\frac{1}{2}$-90	75816488	0.01%	0.04%	0.01%	0.01%	0.01%	0.01%
		107.25	268.16	2.64	2.98	3.01	3.01
$\frac{1}{2}$-91	75823903	0.21%	0.01%	0.00%	0.00%	0.00%	0.00%
		50.62	244.54	3.23	3.13	3.32	3.15
$\frac{1}{2}$-92	75985392	0.32%	0.01%	0.00%	0.00%	0.00%	0.00%
		36.85	280.74	3.62	3.73	4.08	3.90
$\frac{1}{2}$-93	75968366	0.03%	0.04%	0.00%	0.00%	0.00%	0.00%
		154.75	237.51	3.51	4.06	4.11	4.32
$\frac{1}{2}$-94	76028460	0.08%	0.03%	0.00%	0.00%	0.00%	0.00%
		161.83	259.50	4.01	4.35	4.58	4.36
$\frac{1}{2}$-95	75903771	0.08%	0.03%	0.00%	0.00%	0.00%	0.00%
		110.73	235.09	3.51	3.61	3.58	3.69
$\frac{1}{2}$-96	75988719	0.37%	0.01%	0.00%	0.00%	0.00%	0.00%
		65.16	235.67	5.26	4.98	4.94	4.79
$\frac{1}{2}$-97	75984046	0.07%	0.01%	0.00%	0.00%	0.00%	0.00%
		91.83	77.85	3.40	3.45	3.36	3.46
$\frac{1}{2}$-98	76025585	0.22%	0.01%	0.00%	0.00%	0.00%	0.00%
		83.15	301.07	3.57	3.99	3.75	3.64
$\frac{1}{2}$-99	75986166	0.12%	0.02%	0.00%	0.00%	0.00%	0.00%
		99.13	231.99	4.55	4.39	4.00	4.01

Appendix IV

Bound strengths (gap %) and computation times (s) for $\frac{3}{4}$ TUFLP-S instances

Results for (PS_{hull}) correspond to the non-stabilised structured Dantzig-Wolfe decomposition in the first four tables. Full results for (PS_{hull}) follow.

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{3}{4}$-0	85603163	0.03%	0.03%	0.03%	0.00%	0.00%	0.00%			
		1.34	1.40	1.32	0.18	3.45	0.17	7.33	7.02	6.88
$\frac{3}{4}$-1	85604667	0.03%	0.03%	0.03%	0.00%	0.00%	0.00%			
		1.80	1.15	1.46	0.15	3.23	0.18	6.84	5.14	2.87
$\frac{3}{4}$-2	85603518	0.03%	0.03%	0.03%	0.01%	0.00%	0.00%			
		1.21	1.29	1.40	0.24	3.68	0.15	6.63	5.37	7.03
$\frac{3}{4}$-3	85603388	0.03%	0.03%	0.03%	0.01%	0.00%	0.00%			
		1.24	1.16	1.12	0.30	3.50	0.14	9.96	4.05	6.76
$\frac{3}{4}$-4	85603358	0.03%	0.03%	0.03%	0.01%	0.00%	0.00%			
		1.57	1.32	1.32	0.30	3.34	0.15	7.79	6.79	7.16
$\frac{3}{4}$-5	85605570	0.03%	0.03%	0.03%	0.00%	0.00%	0.00%			
		1.56	1.10	1.24	0.13	7.60	0.39	7.17	6.38	7.03
$\frac{3}{4}$-6	85628527	0.03%	0.03%	0.03%	0.00%	0.00%	0.00%			
		1.74	1.29	1.14	0.12	6.56	0.38	7.90	7.59	6.60
$\frac{3}{4}$-7	85851985	0.09%	0.09%	0.09%	0.03%	0.00%	0.00%			
		1.33	1.17	1.14	0.16	3.50	0.16	6.92	6.76	8.29
$\frac{3}{4}$-8	87033703	0.07%	0.07%	0.07%	0.01%	0.00%	0.00%			
		1.71	1.52	1.43	0.35	5.20	0.40	7.21	7.75	8.13
$\frac{3}{4}$-9	87479687	0.06%	0.06%	0.06%	0.03%	0.00%	0.00%			
		1.76	1.33	1.25	0.21	3.95	0.15	7.32	6.65	9.04
$\frac{3}{4}$-10	87879203	0.11%	0.11%	0.11%	0.01%	0.00%	0.00%			
		2.46	1.96	1.96	0.40	5.57	0.39	10.90	8.54	10.41
$\frac{3}{4}$-11	87710171	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		1.91	1.36	1.61	0.09	2.75	0.16	1.90	1.70	1.52
$\frac{3}{4}$-12	88038766	0.17%	0.17%	0.17%	0.07%	0.00%	0.00%			
		2.13	1.85	1.72	0.30	26.88	0.97	8.94	7.35	8.33
$\frac{3}{4}$-13	88460145	0.04%	0.04%	0.04%	0.00%	0.00%	0.00%			
		2.34	2.45	2.29	0.34	5.04	0.11	10.60	8.57	9.76
$\frac{3}{4}$-14	88275858	0.07%	0.07%	0.07%	0.01%	0.00%	0.00%			
		2.21	1.53	1.92	0.22	3.90	0.16	7.41	7.94	7.20
$\frac{3}{4}$-15	88789018	0.17%	0.17%	0.17%	0.06%	0.00%	0.00%			
		2.34	2.39	2.18	0.35	8.34	0.45	8.57	7.03	8.49
$\frac{3}{4}$-16	88375534	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		1.63	1.60	1.44	0.11	3.43	0.20	1.82	1.55	1.66
$\frac{3}{4}$-17	89021721	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		2.75	2.52	2.52	0.38	4.89	0.15	10.33	8.84	10.45
$\frac{3}{4}$-18	89266492	0.03%	0.03%	0.03%	0.00%	0.00%	0.00%			
		2.71	2.27	2.28	0.42	4.56	0.13	11.03	11.35	11.38
$\frac{3}{4}$-19	89515379	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		2.58	1.97	2.45	0.25	8.56	0.53	10.21	7.63	10.26
$\frac{3}{4}$-20	89597792	0.07%	0.07%	0.07%	0.02%	0.00%	0.00%			
		2.08	1.66	1.78	0.24	4.37	0.16	6.57	7.48	8.35
$\frac{3}{4}$-21	89826449	0.16%	0.16%	0.16%	0.07%	0.00%	0.00%			
		2.62	2.09	2.26	0.47	7.82	0.64	11.68	11.40	10.27
$\frac{3}{4}$-22	89764952	0.04%	0.04%	0.04%	0.00%	0.00%	0.00%			
		1.82	2.35	1.77	0.26	5.05	0.16	8.11	6.22	7.64
$\frac{3}{4}$-23	90180424	0.15%	0.15%	0.15%	0.03%	0.00%	0.00%			
		2.47	2.16	2.76	0.56	9.45	0.41	11.45	9.70	11.97
$\frac{3}{4}$-24	90403561	0.09%	0.09%	0.09%	0.04%	0.00%	0.00%			
		2.59	2.19	2.66	0.73	12.72	0.76	11.32	10.33	11.74

Instance	Exact	LP Relaxations (PS_{path})	(PS'_{path})	(PS^0_{path})	Reformulation SUBP	(PS_{hull})	Primal Heuristic	MIP times (PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{3}{4}$-25	90532364	0.09%	0.09%	0.09%	0.04%	0.00%	0.00%			
		4.67	3.76	4.09	1.02	12.63	0.90	20.82	16.56	13.34
$\frac{3}{4}$-26	90375563	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%			
		2.86	2.92	2.31	0.46	7.66	0.16	6.92	7.44	9.81
$\frac{3}{4}$-27	90637373	0.08%	0.08%	0.08%	0.02%	0.00%	0.00%			
		6.08	3.42	3.78	0.99	11.01	0.74	18.19	13.75	12.51
$\frac{3}{4}$-28	91013405	0.05%	0.05%	0.05%	0.01%	0.00%	0.00%			
		2.71	2.75	2.37	0.31	5.84	0.33	12.04	9.79	9.26
$\frac{3}{4}$-29	91136339	0.15%	0.15%	0.15%	0.08%	0.00%	0.00%			
		2.81	2.40	2.68	0.80	14.31	1.18	18.52	18.00	15.00
$\frac{3}{4}$-30	91017870	0.07%	0.07%	0.07%	0.03%	0.00%	0.00%			
		3.54	4.87	2.34	0.48	10.04	0.64	14.80	14.13	12.34
$\frac{3}{4}$-31	91032203	0.03%	0.03%	0.03%	0.00%	0.00%	0.00%			
		2.04	2.33	1.80	0.54	5.07	0.18	13.98	10.45	9.50
$\frac{3}{4}$-32	91211223	0.09%	0.09%	0.10%	0.03%	0.00%	0.00%			
		4.51	3.33	2.82	0.89	19.04	0.90	18.36	16.55	15.57
$\frac{3}{4}$-33	91108897	0.07%	0.07%	0.07%	0.05%	0.00%	0.00%			
		3.46	8.17	2.58	0.58	11.88	0.54	14.41	9.69	11.57
$\frac{3}{4}$-34	91177033	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		6.36	4.24	3.18	0.85	7.42	0.16	10.24	9.51	10.13
$\frac{3}{4}$-35	91553211	0.08%	0.08%	0.08%	0.00%	0.00%	0.01%			
		3.72	4.48	3.50	0.84	10.64	0.35	19.23	18.15	15.48
$\frac{3}{4}$-36	91300669	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		2.45	2.14	1.92	0.12	4.24	0.16	2.10	4.14	2.36
$\frac{3}{4}$-37	91571240	0.12%	0.12%	0.12%	0.04%	0.00%	0.00%			
		5.19	3.20	3.78	0.84	27.79	0.75	18.58	14.93	16.86
$\frac{3}{4}$-38	91605661	0.10%	0.10%	0.10%	0.01%	0.00%	0.00%			
		3.82	3.67	3.54	0.75	15.02	0.18	13.36	11.40	14.31
$\frac{3}{4}$-39	91633421	0.13%	0.13%	0.13%	0.03%	0.00%	0.00%			
		3.09	4.08	2.62	1.09	11.13	0.61	15.25	13.73	15.03
$\frac{3}{4}$-40	90602972	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		1.74	1.46	1.44	0.08	2.76	0.16	1.66	1.86	1.74
$\frac{3}{4}$-41	91756196	0.14%	0.14%	0.14%	0.06%	0.00%	0.00%			
		3.45	3.24	3.00	0.74	9.50	0.68	15.52	16.02	14.16
$\frac{3}{4}$-42	91811361	0.12%	0.12%	0.12%	0.09%	0.00%	0.00%			
		3.06	3.01	1.90	0.38	31.64	1.26	11.94	13.08	10.48
$\frac{3}{4}$-43	91894879	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%			
		2.24	2.05	1.65	0.12	5.75	0.11	2.80	2.27	2.24
$\frac{3}{4}$-44	92411537	0.06%	0.06%	0.06%	0.01%	0.00%	0.00%			
		4.02	4.21	3.07	0.69	15.52	0.68	14.49	9.57	11.14
$\frac{3}{4}$-45	92366919	0.14%	0.14%	0.14%	0.12%	0.00%	0.00%			
		2.65	2.75	1.95	0.20	13.26	0.89	7.86	7.89	6.60
$\frac{3}{4}$-46	92585190	0.03%	0.03%	0.03%	0.02%	0.00%	0.00%			
		3.56	3.68	2.77	0.37	9.00	0.37	11.57	10.60	10.57
$\frac{3}{4}$-47	92841543	0.13%	0.13%	0.13%	0.04%	0.00%	0.00%			
		4.04	3.54	2.90	0.84	40.50	1.43	12.27	13.66	10.88
$\frac{3}{4}$-48	93094878	0.25%	0.25%	0.25%	0.11%	0.00%	0.00%			
		3.28	2.66	2.62	0.70	69.02	1.78	20.68	17.36	18.24
$\frac{3}{4}$-49	93080488	0.27%	0.27%	0.27%	0.16%	0.00%	0.00%			
		4.95	5.20	3.23	1.40	36.77	1.48	17.52	16.59	15.39

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{3}{4}$-50	93222453	0.32%	0.32%	0.32%	0.25%	0.00%	0.00%			
		3.50	3.51	2.38	0.65	27.17	1.83	16.82	21.00	17.38
$\frac{3}{4}$-51	93242553	0.06%	0.06%	0.06%	0.02%	0.00%	0.00%			
		5.24	4.84	3.42	0.82	30.97	1.39	17.96	16.67	19.02
$\frac{3}{4}$-52	92969190	0.04%	0.04%	0.04%	0.04%	0.00%	0.00%			
		3.61	3.98	2.48	0.42	8.57	0.37	13.19	9.27	12.57
$\frac{3}{4}$-53	93117276	0.15%	0.15%	0.15%	0.03%	0.00%	0.00%			
		3.96	2.80	2.90	0.59	21.25	1.06	12.72	10.43	11.49
$\frac{3}{4}$-54	93381476	0.19%	0.19%	0.19%	0.15%	0.00%	0.00%			
		5.20	7.50	3.68	0.78	13.75	0.85	18.90	14.30	15.94
$\frac{3}{4}$-55	93397731	0.34%	0.34%	0.34%	0.23%	0.01%	0.01%			
		3.77	4.85	2.66	1.51	27.88	1.75	19.75	30.23	20.50
$\frac{3}{4}$-56	93126498	0.10%	0.10%	0.10%	0.05%	0.00%	0.00%			
		2.30	3.67	2.42	0.51	16.21	0.86	11.82	9.09	11.69
$\frac{3}{4}$-57	93358394	0.15%	0.15%	0.15%	0.12%	0.00%	0.00%			
		3.21	3.58	2.36	0.56	11.93	0.61	14.81	14.61	10.52
$\frac{3}{4}$-58	93448411	0.14%	0.14%	0.14%	0.09%	0.00%	0.00%			
		10.59	4.40	5.18	1.10	34.68	1.71	22.09	23.71	23.56
$\frac{3}{4}$-59	93552042	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		5.39	4.42	4.80	0.65	7.86	0.15	14.32	13.25	16.91
$\frac{3}{4}$-60	93802040	0.30%	0.30%	0.30%	0.23%	0.00%	0.00%			
		3.82	3.85	3.87	0.55	154.60	1.17	18.07	17.61	20.77
$\frac{3}{4}$-61	93652804	0.28%	0.28%	0.28%	0.22%	0.00%	0.00%			
		3.64	2.57	2.56	0.55	137.89	2.13	18.93	18.17	14.32
$\frac{3}{4}$-62	93209783	0.23%	0.23%	0.23%	0.16%	0.00%	0.00%			
		2.98	2.74	2.39	0.54	43.96	0.82	13.96	11.96	12.05
$\frac{3}{4}$-63	93350899	0.08%	0.08%	0.08%	0.04%	0.00%	0.00%			
		3.34	3.83	2.74	0.55	11.54	0.95	14.19	13.39	11.86
$\frac{3}{4}$-64	93822033	0.18%	0.18%	0.18%	0.17%	0.00%	0.00%			
		3.02	3.22	2.98	0.43	33.19	2.05	17.01	12.56	11.59
$\frac{3}{4}$-65	93824675	0.33%	0.33%	0.33%	0.21%	0.00%	0.00%			
		4.44	4.10	3.06	0.50	106.40	1.47	20.02	28.25	17.12
$\frac{3}{4}$-66	94119719	0.21%	0.21%	0.21%	0.16%	0.00%	0.00%			
		3.96	3.67	3.06	0.95	31.59	1.56	18.39	28.30	19.37
$\frac{3}{4}$-67	93990418	0.31%	0.31%	0.31%	0.23%	0.00%	0.00%			
		3.95	4.02	3.06	0.67	230.49	1.60	34.45	41.33	31.53
$\frac{3}{4}$-68	93996826	0.23%	0.23%	0.23%	0.13%	0.00%	0.00%			
		4.22	4.70	3.82	0.84	19.65	0.99	16.79	18.91	15.35
$\frac{3}{4}$-69	94168667	0.21%	0.21%	0.21%	0.10%	0.00%	0.00%			
		4.00	2.85	2.88	1.01	180.29	1.93	23.74	43.34	24.09
$\frac{3}{4}$-70	94275201	0.20%	0.20%	0.20%	0.06%	0.00%	0.00%			
		3.64	3.62	2.74	1.09	50.18	0.86	20.16	12.55	16.98
$\frac{3}{4}$-71	94294624	0.13%	0.13%	0.13%	0.09%	0.00%	0.00%			
		4.32	5.35	4.44	0.65	36.09	0.88	19.87	18.78	27.63
$\frac{3}{4}$-72	94518242	0.18%	0.18%	0.18%	0.12%	0.00%	0.00%			
		5.95	5.10	4.69	2.39	65.04	0.98	32.25	28.92	44.76
$\frac{3}{4}$-73	94455182	0.25%	0.25%	0.25%	0.06%	0.00%	0.00%			
		4.36	4.82	3.91	1.40	137.19	0.86	24.29	28.59	41.75
$\frac{3}{4}$-74	94499399	0.16%	0.16%	0.16%	0.08%	0.00%	0.00%			
		7.38	4.68	4.59	1.27	151.33	1.51	23.62	21.88	20.08

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
$\frac{3}{4}$-75	94380310	0.16%	0.16%	0.16%	0.04%	0.00%	0.00%			
		4.80	5.71	4.50	1.90	24.41	0.74	29.76	45.29	38.48
$\frac{3}{4}$-76	94557454	0.33%	0.33%	0.33%	0.18%	0.00%	0.00%			
		3.46	6.03	3.31	1.51	43.34	0.83	31.03	30.57	35.35
$\frac{3}{4}$-77	94443801	0.15%	0.15%	0.15%	0.06%	0.00%	0.00%			
		4.15	4.91	3.01	1.30	10.28	0.48	16.75	19.45	15.09
$\frac{3}{4}$-78	94553612	0.31%	0.31%	0.31%	0.13%	0.00%	0.00%			
		5.30	5.63	4.70	2.37	50.84	2.04	37.85	26.72	34.93
$\frac{3}{4}$-79	94310061	0.28%	0.28%	0.28%	0.11%	0.00%	0.00%			
		3.28	3.19	2.72	1.72	17.12	1.12	16.58	21.07	15.69
$\frac{3}{4}$-80	94495817	0.23%	0.23%	0.23%	0.18%	0.00%	0.00%			
		5.26	4.10	5.77	1.18	135.81	1.42	20.18	17.03	17.63
$\frac{3}{4}$-81	94393094	0.29%	0.29%	0.30%	0.17%	0.00%	0.00%			
		5.86	4.26	5.02	1.41	48.01	1.76	26.86	41.03	28.41
$\frac{3}{4}$-82	94468418	0.18%	0.18%	0.18%	0.13%	0.00%	0.00%			
		4.93	3.35	4.20	0.84	89.10	1.20	18.58	19.60	19.74
$\frac{3}{4}$-83	94733235	0.24%	0.24%	0.24%	0.21%	0.00%	0.00%			
		5.52	4.53	3.71	0.70	113.71	1.78	20.14	25.77	22.07
$\frac{3}{4}$-84	95190701	0.35%	0.35%	0.35%	0.16%	0.00%	0.00%			
		5.32	5.87	4.29	2.41	418.57	2.28	39.65	105.25	32.32
$\frac{3}{4}$-85	94820933	0.22%	0.22%	0.22%	0.20%	0.00%	0.00%			
		5.35	4.84	5.09	0.48	11.13	0.62	30.89	14.66	19.13
$\frac{3}{4}$-86	94905457	0.24%	0.24%	0.24%	0.16%	0.00%	0.00%			
		4.16	5.09	3.77	1.29	15.72	0.70	26.35	25.75	23.81
$\frac{3}{4}$-87	94977880	0.35%	0.35%	0.35%	0.17%	0.00%	0.00%			
		3.51	3.38	2.44	1.21	12.05	0.48	25.11	24.65	22.36
$\frac{3}{4}$-88	94630460	0.18%	0.18%	0.18%	0.15%	0.00%	0.00%			
		4.22	2.12	2.72	0.29	77.32	1.86	9.99	8.95	11.96
$\frac{3}{4}$-89	94797993	0.09%	0.09%	0.09%	0.07%	0.00%	0.00%			
		5.56	3.57	3.45	1.64	21.17	1.48	18.82	23.17	20.40
$\frac{3}{4}$-90	95153081	0.39%	0.39%	0.40%	0.26%	0.00%	0.00%			
		2.70	3.50	2.32	1.61	516.00	2.26	47.41	60.34	28.27
$\frac{3}{4}$-91	94723310	0.26%	0.26%	0.27%	0.08%	0.00%	0.00%			
		4.28	9.03	3.24	1.17	211.81	1.12	20.03	16.32	16.26
$\frac{3}{4}$-92	95144033	0.25%	0.25%	0.25%	0.13%	0.00%	0.00%			
		4.48	6.50	3.22	1.14	98.84	1.06	23.17	25.19	20.21
$\frac{3}{4}$-93	95213966	0.31%	0.31%	0.31%	0.18%	0.00%	0.00%			
		11.96	4.05	4.14	1.35	465.45	1.64	28.93	40.96	25.83
$\frac{3}{4}$-94	94971542	0.15%	0.15%	0.15%	0.07%	0.00%	0.00%			
		3.26	3.73	3.36	1.04	12.71	0.87	13.37	13.64	13.39
$\frac{3}{4}$-95	95374759	0.39%	0.39%	0.39%	0.18%	0.00%	0.00%			
		4.12	3.86	3.88	2.95	20.33	0.96	39.97	42.35	51.96
$\frac{3}{4}$-96	95423167	0.27%	0.27%	0.27%	0.12%	0.00%	0.00%			
		3.54	3.40	3.88	1.60	79.22	1.35	29.55	45.92	31.67
$\frac{3}{4}$-97	95256555	0.23%	0.23%	0.23%	0.12%	0.00%	0.00%			
		3.88	3.80	5.35	0.81	46.88	1.11	23.96	22.15	24.71
$\frac{3}{4}$-98	95352920	0.17%	0.17%	0.17%	0.09%	0.00%	0.00%			
		4.21	3.96	4.70	1.22	50.24	2.02	24.06	25.63	19.16
$\frac{3}{4}$-99	95535145	0.23%	0.23%	0.23%	0.16%	0.00%	0.00%			
		7.60	7.82	6.18	1.87	138.89	1.92	46.44	38.49	39.19

Instance	Exact	NDO master		(PS_{hull})			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{3}{4}$-0	85603163	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		2.85	87.13	4.08	3.83	4.55	4.72
$\frac{3}{4}$-1	85604667	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		2.38	3.21	3.46	3.41	4.71	4.04
$\frac{3}{4}$-2	85603518	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.13	3.99	3.69	4.07	3.69	4.09
$\frac{3}{4}$-3	85603388	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		2.67	4.61	3.78	3.40	3.68	3.72
$\frac{3}{4}$-4	85603358	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		2.72	3.49	3.36	3.57	4.52	3.55
$\frac{3}{4}$-5	85605570	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.20	7.46	6.84	8.37	11.61	13.83
$\frac{3}{4}$-6	85628527	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.58	3.22	6.29	7.10	7.85	7.89
$\frac{3}{4}$-7	85851985	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		3.32	112.62	3.47	3.59	3.97	3.64
$\frac{3}{4}$-8	87033703	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.25	7.01	5.18	6.09	6.35	6.97
$\frac{3}{4}$-9	87479687	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		3.69	90.86	3.85	4.21	4.10	4.75
$\frac{3}{4}$-10	87879203	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		8.03	4.35	7.06	6.87	7.39	7.34
$\frac{3}{4}$-11	87710171	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		2.14	2.37	2.61	2.64	3.36	2.80
$\frac{3}{4}$-12	88038766	0.03%	0.03%	0.00%	0.00%	0.00%	0.00%
		3.29	103.75	8.41	29.06	18.13	21.09
$\frac{3}{4}$-13	88460145	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		4.19	4.15	6.83	6.89	7.18	7.44
$\frac{3}{4}$-14	88275858	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.59	8.21	4.37	4.04	4.02	5.42
$\frac{3}{4}$-15	88789018	0.06%	0.02%	0.00%	0.00%	0.00%	0.00%
		4.04	199.37	12.20	10.64	11.08	10.00
$\frac{3}{4}$-16	88375534	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		2.24	2.05	3.95	3.38	3.67	3.50
$\frac{3}{4}$-17	89021721	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		3.46	10.61	7.11	6.35	6.72	6.13
$\frac{3}{4}$-18	89266492	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		4.30	4.25	5.50	5.84	6.49	6.00
$\frac{3}{4}$-19	89515379	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		5.16	8.26	7.66	9.62	10.34	10.64
$\frac{3}{4}$-20	89597792	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.12	5.79	4.63	4.97	4.93	5.07
$\frac{3}{4}$-21	89826449	0.05%	0.01%	0.00%	0.00%	0.00%	0.00%
		4.13	190.51	8.62	9.65	10.91	10.27
$\frac{3}{4}$-22	89764952	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		2.88	173.65	5.18	5.68	5.67	5.94
$\frac{3}{4}$-23	90180424	0.08%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.41	218.39	10.62	11.80	12.44	11.88
$\frac{3}{4}$-24	90403561	0.06%	0.01%	0.00%	0.00%	0.00%	0.00%
		5.30	329.54	14.85	15.47	14.75	14.78

Instance	Exact	NDO master Bundle	Volume	(PS$_{\text{hull}}$) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{3}{4}$-25	90532364	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		14.15	179.84	15.68	16.31	16.00	16.47
$\frac{3}{4}$-26	90375563	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		7.47	6.80	10.19	10.50	9.35	9.78
$\frac{3}{4}$-27	90637373	0.05%	0.03%	0.00%	0.00%	0.00%	0.00%
		6.99	98.35	11.79	15.09	13.83	12.37
$\frac{3}{4}$-28	91013405	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.28	242.54	7.15	7.48	8.40	7.85
$\frac{3}{4}$-29	91136339	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		43.46	190.54	16.72	18.15	18.17	18.39
$\frac{3}{4}$-30	91017870	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.24	125.16	13.54	13.42	14.29	13.80
$\frac{3}{4}$-31	91032203	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		4.10	15.54	5.66	6.63	6.77	6.66
$\frac{3}{4}$-32	91211223	0.03%	0.03%	0.00%	0.00%	0.00%	0.00%
		6.16	157.23	15.19	23.66	33.47	22.50
$\frac{3}{4}$-33	91108897	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		4.94	6.41	13.74	17.49	16.12	16.65
$\frac{3}{4}$-34	91177033	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.06	18.54	9.87	11.06	10.45	11.00
$\frac{3}{4}$-35	91553211	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.22	22.39	15.19	14.56	14.63	15.67
$\frac{3}{4}$-36	91300669	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.57	4.58	4.68	5.35	9.42	5.11
$\frac{3}{4}$-37	91571240	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		32.47	248.59	16.41	31.73	24.86	76.66
$\frac{3}{4}$-38	91605661	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		24.89	455.48	15.24	18.89	27.16	29.27
$\frac{3}{4}$-39	91633421	0.05%	0.02%	0.00%	0.00%	0.00%	0.00%
		7.88	232.55	18.44	14.46	17.58	16.76
$\frac{3}{4}$-40	90602972	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		2.21	2.25	2.98	2.61	3.30	3.29
$\frac{3}{4}$-41	91756196	0.04%	0.04%	0.00%	0.00%	0.00%	0.00%
		7.39	164.29	12.52	12.75	14.02	14.70
$\frac{3}{4}$-42	91811361	0.09%	0.01%	0.00%	0.00%	0.00%	0.00%
		4.92	251.68	17.96	35.22	22.85	47.19
$\frac{3}{4}$-43	91894879	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		3.41	4.30	5.03	7.22	6.50	7.28
$\frac{3}{4}$-44	92411537	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.69	28.07	20.70	20.37	25.71	21.37
$\frac{3}{4}$-45	92366919	0.14%	0.01%	0.00%	0.00%	0.00%	0.00%
		4.46	143.34	11.50	19.12	14.22	15.37
$\frac{3}{4}$-46	92585190	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		7.68	129.38	13.78	12.44	14.79	14.00
$\frac{3}{4}$-47	92841543	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		14.72	486.47	21.73	45.94	43.22	70.06
$\frac{3}{4}$-48	93094878	0.13%	0.01%	0.00%	0.00%	0.00%	0.00%
		41.73	546.57	22.96	74.61	78.78	77.33
$\frac{3}{4}$-49	93080488	0.06%	0.01%	0.00%	0.00%	0.00%	0.00%
		105.54	279.02	24.05	42.65	42.36	46.63

		NDO master		(PS_{hull})			
Instance	Exact	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
$\frac{3}{4}$-50	93222453	0.07%	0.04%	0.00%	0.00%	0.00%	0.00%
		207.81	552.61	31.66	31.30	33.15	32.83
$\frac{3}{4}$-51	93242553	0.03%	0.02%	0.00%	0.00%	0.00%	0.00%
		7.66	203.99	26.29	37.18	74.84	89.73
$\frac{3}{4}$-52	92969190	0.04%	0.02%	0.00%	0.00%	0.00%	0.00%
		7.14	125.92	12.52	11.82	12.74	12.17
$\frac{3}{4}$-53	93117276	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		40.16	286.41	14.02	25.12	21.98	28.54
$\frac{3}{4}$-54	93381476	0.14%	0.06%	0.00%	0.00%	0.00%	0.00%
		9.62	497.30	19.37	18.59	28.42	25.76
$\frac{3}{4}$-55	93397731	0.17%	0.03%	0.01%	0.01%	0.01%	0.01%
		62.91	625.46	31.14	33.92	36.80	37.53
$\frac{3}{4}$-56	93126498	0.03%	0.02%	0.00%	0.00%	0.00%	0.00%
		7.03	386.47	12.44	19.23	33.86	42.37
$\frac{3}{4}$-57	93358394	0.11%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.85	209.37	14.61	17.36	17.22	15.93
$\frac{3}{4}$-58	93448411	0.05%	0.02%	0.00%	0.00%	0.00%	0.00%
		49.22	457.12	41.76	39.70	39.06	42.17
$\frac{3}{4}$-59	93552042	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		6.30	7.30	12.56	11.89	13.77	13.20
$\frac{3}{4}$-60	93802040	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		416.64	638.13	32.81	161.48	88.81	145.22
$\frac{3}{4}$-61	93652804	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		57.44	285.14	30.26	144.56	53.31	172.42
$\frac{3}{4}$-62	93209783	0.06%	0.01%	0.00%	0.00%	0.00%	0.00%
		98.92	460.05	13.46	47.33	47.96	84.89
$\frac{3}{4}$-63	93350899	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		31.15	169.60	12.80	15.37	14.87	17.04
$\frac{3}{4}$-64	93822033	0.07%	0.01%	0.00%	0.00%	0.00%	0.00%
		32.31	314.68	21.33	37.33	33.52	36.61
$\frac{3}{4}$-65	93824675	0.08%	0.01%	0.00%	0.00%	0.00%	0.00%
		101.57	301.43	22.62	112.75	64.27	207.86
$\frac{3}{4}$-66	94119719	0.06%	0.02%	0.00%	0.00%	0.00%	0.00%
		88.14	420.12	38.27	36.99	42.96	43.87
$\frac{3}{4}$-67	93990418	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		320.99	968.80	39.11	239.17	62.25	118.02
$\frac{3}{4}$-68	93996826	0.00%	0.04%	0.00%	0.00%	0.00%	0.00%
		57.03	468.71	19.02	24.78	22.32	23.10
$\frac{3}{4}$-69	94168667	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		261.87	410.45	38.18	188.08	58.25	143.75
$\frac{3}{4}$-70	94275201	0.01%	0.04%	0.00%	0.00%	0.00%	0.00%
		47.84	222.97	22.07	55.60	46.03	43.82
$\frac{3}{4}$-71	94294624	0.11%	0.01%	0.00%	0.00%	0.00%	0.00%
		6.84	295.39	19.96	41.39	29.60	66.36
$\frac{3}{4}$-72	94518242	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		223.79	631.98	36.14	73.01	72.59	93.00
$\frac{3}{4}$-73	94455182	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		86.15	647.88	24.07	147.04	43.80	68.85
$\frac{3}{4}$-74	94499399	0.09%	0.02%	0.00%	0.00%	0.00%	0.00%
		15.77	433.54	58.71	159.63	89.30	193.76

Instance	Exact	NDO master Bundle	NDO master Volume	(PS_{hull}) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
$\frac{3}{4}$-75	94380310	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		133.34	756.59	37.46	30.58	29.75	33.20
$\frac{3}{4}$-76	94557454	0.13%	0.01%	0.00%	0.00%	0.00%	0.00%
		46.03	463.49	26.22	50.67	45.53	65.25
$\frac{3}{4}$-77	94443801	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		87.73	426.94	12.79	15.16	14.21	14.89
$\frac{3}{4}$-78	94553612	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		314.04	1787.23	50.37	58.99	109.44	137.41
$\frac{3}{4}$-79	94310061	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		202.65	675.12	21.24	21.10	21.68	25.59
$\frac{3}{4}$-80	94495817	0.07%	0.02%	0.00%	0.00%	0.00%	0.00%
		43.54	386.21	19.93	142.98	56.85	189.17
$\frac{3}{4}$-81	94393094	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		378.83	684.45	31.42	53.73	59.92	63.70
$\frac{3}{4}$-82	94468418	0.04%	0.02%	0.00%	0.00%	0.00%	0.00%
		102.98	420.65	24.89	95.65	31.79	88.05
$\frac{3}{4}$-83	94733235	0.10%	0.01%	0.00%	0.00%	0.00%	0.00%
		60.47	668.61	28.80	119.90	61.91	187.97
$\frac{3}{4}$-84	95190701	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		417.88	990.87	48.71	431.74	71.07	226.30
$\frac{3}{4}$-85	94820933	0.00%	0.04%	0.00%	0.00%	0.00%	0.00%
		50.98	275.19	14.70	15.22	17.17	18.97
$\frac{3}{4}$-86	94905457	0.13%	0.01%	0.00%	0.00%	0.00%	0.00%
		31.17	964.41	19.29	21.59	21.93	27.18
$\frac{3}{4}$-87	94977880	0.04%	0.01%	0.00%	0.00%	0.00%	0.00%
		177.06	269.08	16.74	15.72	16.81	19.28
$\frac{3}{4}$-88	94630460	0.06%	0.01%	0.00%	0.00%	0.00%	0.00%
		23.30	149.96	21.36	82.04	41.15	57.95
$\frac{3}{4}$-89	94797993	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		123.83	624.86	27.11	26.61	28.81	31.51
$\frac{3}{4}$-90	95153081	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		518.31	1049.09	46.91	525.78	154.60	509.08
$\frac{3}{4}$-91	94723310	0.00%	0.02%	0.00%	0.00%	0.00%	0.00%
		121.93	441.98	35.34	220.99	49.36	64.17
$\frac{3}{4}$-92	95144033	0.14%	0.01%	0.00%	0.00%	0.00%	0.00%
		17.32	821.29	30.84	107.18	67.54	109.49
$\frac{3}{4}$-93	95213966	0.03%	0.03%	0.00%	0.00%	0.00%	0.00%
		231.79	558.67	36.33	476.72	102.71	248.85
$\frac{3}{4}$-94	94971542	0.00%	0.02%	0.00%	0.00%	0.00%	0.00%
		64.14	279.70	20.42	16.62	18.27	21.37
$\frac{3}{4}$-95	95374759	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		446.10	591.00	24.33	25.81	25.79	30.97
$\frac{3}{4}$-96	95423167	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		114.91	379.33	28.55	85.62	76.04	126.97
$\frac{3}{4}$-97	95256555	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		101.09	393.26	24.40	52.26	55.20	84.43
$\frac{3}{4}$-98	95352920	0.00%	0.02%	0.00%	0.00%	0.00%	0.00%
		120.79	733.28	30.11	55.40	55.27	114.56
$\frac{3}{4}$-99	95535145	0.03%	0.02%	0.00%	0.00%	0.00%	0.00%
		107.23	1421.29	44.66	149.91	144.66	130.26

Appendix V

Bound strengths (gap %) and computation times (s) for Full TUFLP-S instances

Results for (PS_{hull}) correspond to the non-stabilised structured Dantzig-Wolfe decomposition in the first four tables. Full results for (PS_{hull}) follow.

Instance	Exact	LP Relaxations (PS_{path})	LP Relaxations (PS'_{path})	LP Relaxations (PS^0_{path})	Reformulation SUBP	Reformulation (PS_{hull})	Primal Heuristic	MIP times (PS_{path})	MIP times (PS'_{path})	MIP times (PS^0_{path})
Full-0	99950119	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		14.53	9.95	12.66	0.68	18.23	0.54	26.81	27.98	20.65
Full-1	99952100	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		14.70	11.02	13.85	0.60	18.69	0.42	19.93	20.23	27.03
Full-2	99950464	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		21.79	9.81	12.76	0.88	15.85	0.42	24.35	21.46	20.97
Full-3	99950324	0.05%	0.05%	0.05%	0.03%	0.00%	0.00%			
		8.81	9.54	8.28	0.99	23.80	1.15	20.56	19.17	20.98
Full-4	99950319	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		15.02	9.11	23.45	1.38	29.87	0.48	21.69	26.79	35.92
Full-5	99950320	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		7.39	13.81	9.45	1.57	20.21	0.40	22.20	20.39	26.68
Full-6	99952791	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		14.75	13.45	13.88	0.97	19.86	0.55	25.70	21.12	28.76
Full-7	99978511	0.05%	0.05%	0.05%	0.00%	0.00%	0.00%			
		12.16	7.82	9.62	0.91	15.67	0.51	24.81	26.13	23.55
Full-8	100210784	0.12%	0.12%	0.12%	0.03%	0.00%	0.00%			
		16.24	15.22	13.06	1.49	32.47	1.79	38.94	59.78	46.36
Full-9	100948603	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		22.27	16.15	15.49	1.71	24.23	0.43	34.11	45.82	37.50
Full-10	101230435	0.23%	0.23%	0.23%	0.15%	0.00%	0.00%			
		12.67	9.03	14.80	1.47	25.56	2.45	43.22	57.37	45.72
Full-11	101382113	0.30%	0.30%	0.30%	0.11%	0.00%	0.00%			
		18.06	11.26	15.71	3.61	43.19	3.11	53.41	80.13	87.99
Full-12	102105878	0.20%	0.20%	0.20%	0.08%	0.00%	0.00%			
		10.49	6.42	9.02	2.72	20.01	1.17	36.06	31.03	33.65
Full-13	101635061	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		10.81	7.21	13.30	0.72	12.89	0.48	22.67	26.50	11.10
Full-14	102874117	0.25%	0.25%	0.25%	0.10%	0.00%	0.00%			
		22.61	16.95	24.01	3.54	38.88	1.06	77.15	85.82	81.83
Full-15	102626556	0.22%	0.22%	0.22%	0.11%	0.00%	0.00%			
		12.91	8.03	12.33	2.79	24.24	1.08	34.90	41.40	33.28
Full-16	102658223	0.13%	0.13%	0.13%	0.05%	0.00%	0.00%			
		15.57	16.73	19.49	1.85	44.98	1.78	41.66	32.74	53.41
Full-17	103588498	0.28%	0.28%	0.28%	0.16%	0.00%	0.00%			
		16.70	17.40	14.57	4.00	47.34	1.86	78.29	92.50	81.53
Full-18	103107136	0.11%	0.11%	0.11%	0.07%	0.00%	0.00%			
		23.05	9.42	20.99	1.03	29.68	1.94	31.77	31.54	54.36
Full-19	103615765	0.08%	0.08%	0.08%	0.04%	0.00%	0.00%			
		12.33	10.33	9.65	1.02	25.41	1.43	41.87	35.67	37.95
Full-20	103382324	0.15%	0.15%	0.15%	0.07%	0.00%	0.00%			
		10.74	10.52	11.62	1.68	26.18	2.01	34.56	31.04	34.56
Full-21	103531568	0.04%	0.04%	0.04%	0.03%	0.00%	0.00%			
		9.21	8.70	12.57	1.73	26.85	1.95	25.63	27.39	28.06
Full-22	103989391	0.24%	0.24%	0.24%	0.15%	0.00%	0.00%			
		17.61	13.15	18.91	3.43	37.07	1.87	52.01	72.16	53.22
Full-23	103842546	0.07%	0.07%	0.07%	0.02%	0.00%	0.00%			
		17.26	11.06	14.83	1.83	23.55	0.91	33.36	42.78	40.99
Full-24	104727560	0.18%	0.18%	0.18%	0.08%	0.00%	0.00%			
		16.06	18.31	15.53	3.86	33.91	1.17	80.04	51.22	53.28

Instance	Exact	LP Relaxations			Reformulation		Primal	MIP times		
		(PS_{path})	(PS'_{path})	(PS^0_{path})	SUBP	(PS_{hull})	Heuristic	(PS_{path})	(PS'_{path})	(PS^0_{path})
Full-25	104419774	0.21%	0.21%	0.21%	0.07%	0.00%	0.00%			
		12.29	9.85	13.23	4.62	24.95	0.93	67.93	43.82	53.62
Full-26	104695072	0.28%	0.28%	0.28%	0.15%	0.00%	0.00%			
		11.99	11.98	9.66	2.50	44.00	2.01	105.03	157.97	56.64
Full-27	104571556	0.07%	0.07%	0.07%	0.00%	0.00%	0.00%			
		15.26	19.10	15.47	1.67	29.04	1.27	33.62	34.53	29.43
Full-28	104832909	0.27%	0.27%	0.27%	0.18%	0.00%	0.00%			
		32.02	13.72	20.50	4.48	35.64	0.97	138.57	74.69	57.37
Full-29	104733344	0.07%	0.07%	0.07%	0.03%	0.00%	0.00%			
		18.00	16.46	13.57	2.26	33.56	1.97	45.55	34.03	39.86
Full-30	105063205	0.20%	0.20%	0.20%	0.10%	0.00%	0.00%			
		13.76	14.77	11.82	3.94	46.07	1.45	86.33	85.25	61.30
Full-31	105108478	0.08%	0.08%	0.08%	0.02%	0.00%	0.00%			
		23.17	14.45	17.31	2.30	55.00	1.78	52.56	44.03	49.75
Full-32	105425737	0.23%	0.23%	0.23%	0.17%	0.00%	0.00%			
		27.73	18.41	17.41	3.30	80.43	2.29	101.74	102.75	109.27
Full-33	105298999	0.23%	0.23%	0.23%	0.13%	0.00%	0.00%			
		24.33	10.08	20.42	3.22	72.93	3.88	76.12	100.21	79.63
Full-34	105177415	0.18%	0.18%	0.18%	0.06%	0.00%	0.00%			
		24.47	13.97	17.85	2.32	34.14	1.37	57.34	64.56	54.62
Full-35	105399665	0.15%	0.15%	0.15%	0.08%	0.00%	0.00%			
		12.83	15.31	14.91	2.74	41.20	2.26	51.98	38.43	58.39
Full-36	105636810	0.26%	0.26%	0.26%	0.14%	0.00%	0.00%			
		13.54	11.90	12.36	4.59	52.44	2.67	84.43	107.71	77.21
Full-37	105879289	0.28%	0.28%	0.28%	0.13%	0.00%	0.00%			
		20.77	18.05	18.29	3.96	83.25	3.06	150.68	170.90	129.62
Full-38	105779654	0.13%	0.13%	0.13%	0.11%	0.00%	0.00%			
		17.30	18.65	16.44	0.74	34.51	1.16	60.52	58.60	67.94
Full-39	105971166	0.13%	0.13%	0.13%	0.03%	0.00%	0.00%			
		13.68	19.66	12.44	5.08	109.91	2.45	67.71	64.60	72.00
Full-40	106222815	0.32%	0.32%	0.32%	0.09%	0.00%	0.00%			
		22.39	16.97	13.05	7.32	50.41	2.37	100.46	162.83	87.53
Full-41	106561264	0.25%	0.25%	0.25%	0.11%	0.00%	0.00%			
		21.43	19.26	19.01	4.02	63.89	2.20	161.45	114.84	101.50
Full-42	106746217	0.36%	0.36%	0.37%	0.17%	0.00%	0.00%			
		18.48	15.29	17.28	6.56	97.83	3.27	104.75	181.96	138.49
Full-43	106512301	0.17%	0.17%	0.17%	0.08%	0.00%	0.00%			
		26.19	23.38	20.89	5.48	52.89	1.58	109.96	104.86	91.88
Full-44	106408488	0.28%	0.28%	0.28%	0.11%	0.00%	0.00%			
		22.25	20.19	13.39	7.68	70.98	1.86	259.04	112.42	82.65
Full-45	106668481	0.31%	0.31%	0.31%	0.18%	0.00%	0.00%			
		19.97	17.54	16.91	4.87	87.81	3.30	96.10	124.07	87.41
Full-46	106899314	0.26%	0.26%	0.26%	0.12%	0.00%	0.00%			
		16.64	20.02	24.65	8.92	44.23	1.01	90.75	123.98	66.86
Full-47	106945452	0.26%	0.26%	0.26%	0.11%	0.00%	0.00%			
		19.04	15.99	21.29	4.41	65.76	2.99	138.61	136.74	73.53
Full-48	107319900	0.11%	0.11%	0.11%	0.03%	0.00%	0.00%			
		22.55	28.71	22.07	5.38	46.09	0.98	70.32	75.73	66.47
Full-49	107499600	0.22%	0.22%	0.23%	0.14%	0.00%	0.00%			
		29.36	28.63	28.61	11.14	178.78	4.12	152.75	187.06	110.91
Full-50	107599670	0.32%	0.32%	0.32%	0.20%	0.00%	0.00%			
		21.31	15.98	17.75	5.15	207.35	3.14	286.86	219.75	136.46

Instance	Exact	LP Relaxations (PS_{path})	(PS'_{path})	(PS^0_{path})	Reformulation SUBP	(PS_{hull})	Primal Heuristic	MIP times (PS_{path})	(PS'_{path})	(PS^0_{path})
Full-51	107654237	0.41%	0.41%	0.41%	0.28%	0.00%	0.00%			
		20.11	16.85	18.61	8.75	135.74	3.43	364.47	848.16	177.37
Full-52	107347315	0.15%	0.15%	0.15%	0.04%	0.00%	0.00%			
		27.61	15.95	24.44	7.44	57.51	1.30	92.27	89.31	81.34
Full-53	107629820	0.37%	0.37%	0.37%	0.19%	0.00%	0.00%			
		15.93	23.52	16.20	5.69	110.37	3.90	182.35	238.78	135.18
Full-54	107709781	0.31%	0.31%	0.31%	0.18%	0.00%	0.00%			
		15.98	21.27	13.49	6.56	128.94	2.58	294.03	214.38	158.05
Full-55	107603587	0.18%	0.18%	0.18%	0.06%	0.00%	0.00%			
		28.34	20.97	24.15	5.10	45.53	1.08	127.97	70.81	125.35
Full-56	107835481	0.23%	0.23%	0.23%	0.12%	0.00%	0.00%			
		24.98	20.16	22.95	8.46	92.31	2.55	170.13	150.26	125.50
Full-57	108204009	0.10%	0.10%	0.10%	0.04%	0.00%	0.00%			
		22.33	29.05	25.76	2.96	48.08	1.01	88.15	100.57	103.69
Full-58	108260976	0.22%	0.22%	0.22%	0.08%	0.00%	0.00%			
		15.11	36.18	14.98	12.09	143.50	3.32	147.45	176.82	225.19
Full-59	108344093	0.26%	0.26%	0.26%	0.22%	0.00%	0.00%			
		35.89	19.53	31.37	3.98	133.47	2.34	188.15	195.27	153.12
Full-60	108359891	0.24%	0.24%	0.24%	0.13%	0.00%	0.00%			
		21.39	18.67	19.66	7.48	86.34	2.44	202.00	176.64	190.04
Full-61	108470640	0.24%	0.24%	0.24%	0.10%	0.00%	0.00%			
		31.03	20.97	18.71	10.00	67.30	1.06	187.59	191.13	159.22
Full-62	108091600	0.20%	0.20%	0.20%	0.13%	0.00%	0.00%			
		23.57	19.13	18.70	1.95	38.36	0.96	110.48	127.15	101.16
Full-63	107854681	0.02%	0.02%	0.02%	0.00%	0.00%	0.00%			
		35.00	18.30	35.44	3.57	30.61	0.41	34.88	47.25	38.29
Full-64	108355427	0.36%	0.36%	0.36%	0.24%	0.00%	0.01%			
		18.11	18.73	23.95	4.18	282.09	4.54	312.12	389.79	404.38
Full-65	108301483	0.31%	0.31%	0.31%	0.11%	0.00%	0.00%			
		19.05	20.12	16.89	10.72	72.95	1.96	206.06	276.88	119.87
Full-66	108562860	0.34%	0.34%	0.34%	0.16%	0.00%	0.00%			
		20.35	23.88	29.16	14.23	3731.53	3.76	347.33	758.50	259.85
Full-67	109027223	0.29%	0.29%	0.29%	0.12%	0.00%	0.00%			
		18.58	21.91	19.61	11.98	252.41	2.51	487.78	442.68	254.07
Full-68	109071187	0.38%	0.38%	0.38%	0.21%	0.00%	0.00%			
		33.72	20.73	28.88	9.91	473.03	5.34	614.66	593.91	291.12
Full-69	108966429	0.39%	0.39%	0.39%	0.23%	0.00%	0.00%			
		20.25	20.26	24.82	12.28	304.54	4.45	316.37	816.52	263.09
Full-70	108739438	0.16%	0.16%	0.16%	0.07%	0.00%	0.00%			
		26.09	22.93	17.13	7.15	49.94	1.03	150.52	158.58	116.37
Full-71	109017680	0.34%	0.34%	0.34%	0.17%	0.00%	0.00%			
		27.25	20.32	19.18	9.16	256.12	4.49	351.77	624.90	212.68
Full-72	108811565	0.22%	0.22%	0.22%	0.09%	0.00%	0.00%			
		18.50	28.57	21.20	7.77	90.43	2.03	193.18	231.04	270.32
Full-73	109039618	0.27%	0.27%	0.27%	0.16%	0.00%	0.00%			
		21.11	21.21	37.86	6.24	290.76	3.90	291.24	283.17	198.10
Full-74	109482028	0.35%	0.35%	0.35%	0.17%	0.00%	0.00%			
		21.26	32.47	21.07	18.51	2127.40	5.69	566.12	857.78	298.23

Instance	Exact	LP Relaxations (PS_{path})	(PS'_{path})	(PS^0_{path})	Reformulation SUBP	(PS_{hull})	Primal Heuristic	MIP times (PS_{path})	(PS'_{path})	(PS^0_{path})
Full-75	109394833	0.34%	0.34%	0.34%	0.21%	0.01%	0.00%			
		27.18	29.19	23.13	14.54	3105.10	4.65	379.52	526.28	292.15
Full-76	109481987	0.35%	0.35%	0.35%	0.21%	0.00%	0.00%			
		19.89	24.77	20.55	11.46	270.43	4.93	405.20	815.75	470.84
Full-77	109349348	0.31%	0.31%	0.31%	0.16%	0.00%	0.00%			
		24.88	31.48	24.77	15.32	246.06	4.89	342.79	246.16	201.51
Full-78	109242830	0.20%	0.20%	0.20%	0.08%	0.00%	0.00%			
		29.00	22.89	27.01	4.57	140.34	4.02	105.50	160.89	117.84
Full-79	109292026	0.29%	0.29%	0.29%	0.17%	0.00%	0.00%			
		31.77	27.28	29.68	11.95	418.03	4.74	280.39	434.40	290.95
Full-80	109418679	0.27%	0.27%	0.27%	0.10%	0.00%	0.00%			
		37.69	19.72	39.96	13.03	223.92	4.48	299.36	322.55	230.31
Full-81	109418342	0.37%	0.37%	0.37%	0.16%	0.00%	0.00%			
		33.40	18.55	38.76	9.93	133.38	2.71	513.41	797.99	450.52
Full-82	109596968	0.24%	0.24%	0.24%	0.17%	0.00%	0.00%			
		24.40	40.94	37.53	9.54	194.70	3.22	452.82	267.30	205.22
Full-83	109572186	0.29%	0.29%	0.29%	0.13%	0.00%	0.00%			
		41.54	25.86	41.47	11.87	174.66	2.52	314.49	202.34	311.38
Full-84	109948400	0.36%	0.36%	0.36%	0.17%	0.00%	0.00%			
		35.30	39.78	30.85	27.68	383.13	4.12	340.40	518.10	309.58
Full-85	109898382	0.31%	0.31%	0.31%	0.13%	0.00%	0.00%			
		35.33	25.64	40.99	15.73	254.18	3.22	307.52	269.08	316.34
Full-86	109789890	0.26%	0.26%	0.26%	0.17%	0.00%	0.00%			
		25.92	37.27	35.07	9.47	140.57	3.32	286.93	261.31	212.46
Full-87	109734474	0.25%	0.25%	0.25%	0.07%	0.00%	0.00%			
		54.54	31.56	26.48	16.23	122.86	3.32	215.23	270.19	151.11
Full-88	109862389	0.27%	0.27%	0.27%	0.12%	0.00%	0.00%			
		34.28	21.03	27.91	23.20	208.85	2.53	179.54	247.42	210.51
Full-89	109738695	0.22%	0.22%	0.22%	0.07%	0.00%	0.00%			
		28.45	35.70	26.11	8.98	87.78	1.90	301.96	213.51	166.99
Full-90	109715464	0.12%	0.12%	0.12%	0.02%	0.00%	0.00%			
		42.89	29.64	39.00	10.10	51.19	0.41	177.39	92.67	112.82
Full-91	110010434	0.34%	0.34%	0.34%	0.12%	0.01%	0.00%			
		27.39	22.82	22.69	23.19	491.25	4.63	300.46	1300.56	437.20
Full-92	109950201	0.23%	0.23%	0.23%	0.07%	0.00%	0.00%			
		65.44	27.34	28.78	19.47	72.47	1.09	199.71	252.84	125.55
Full-93	110133703	0.43%	0.43%	0.43%	0.26%	0.00%	0.00%			
		25.51	39.53	15.37	17.00	365.04	4.10	567.82	1558.91	427.12
Full-94	110201145	0.27%	0.27%	0.27%	0.20%	0.00%	0.00%			
		35.97	43.58	36.79	8.19	295.27	3.35	443.57	341.93	258.93
Full-95	110312649	0.33%	0.33%	0.33%	0.21%	0.00%	0.00%			
		44.33	33.50	41.33	12.77	487.20	3.25	442.68	1053.44	294.32
Full-96	110287755	0.31%	0.31%	0.31%	0.16%	0.00%	0.00%			
		29.86	24.61	24.42	10.20	434.69	3.92	276.49	531.46	251.87
Full-97	110306919	0.26%	0.26%	0.27%	0.19%	0.00%	0.00%			
		26.45	29.61	25.37	11.92	127.74	2.72	402.09	233.20	176.43
Full-98	110303293	0.26%	0.26%	0.26%	0.11%	0.00%	0.00%			
		32.33	42.89	31.62	25.88	165.83	2.16	233.62	401.99	191.50
Full-99	110370538	0.32%	0.32%	0.32%	0.13%	0.00%	0.00%			
		30.26	27.25	29.00	13.92	198.68	2.02	338.91	330.48	305.54

		NDO master		(PS_{hull})			
Instance	Exact	Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
Full-0	99950119	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		12.79	13.46	19.38	26.20	26.72	30.85
Full-1	99952100	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		11.57	11.66	21.00	28.70	29.19	27.81
Full-2	99950464	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		23.73	17.87	25.27	22.96	23.09	22.66
Full-3	99950324	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		11.95	14.77	26.53	34.88	32.62	27.34
Full-4	99950319	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		18.47	14.64	28.33	50.61	33.64	24.74
Full-5	99950320	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		16.23	13.90	32.34	32.00	35.30	39.82
Full-6	99952791	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		14.90	17.48	31.46	30.32	26.44	35.69
Full-7	99978511	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		12.63	11.75	38.16	22.38	26.45	26.68
Full-8	100210784	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		17.98	16.86	63.67	47.55	52.31	46.53
Full-9	100948603	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		15.40	16.21	55.36	37.64	50.92	33.78
Full-10	101230435	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		38.39	544.92	33.85	34.22	44.61	45.68
Full-11	101382113	0.13%	0.02%	0.00%	0.00%	0.00%	0.00%
		65.72	2087.14	52.37	58.57	67.21	80.05
Full-12	102105878	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		120.96	1774.08	30.39	27.85	30.87	45.38
Full-13	101635061	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		14.78	11.95	16.67	17.99	16.74	23.98
Full-14	102874117	0.04%	0.01%	0.00%	0.00%	0.00%	0.00%
		254.95	1027.00	50.64	61.38	82.95	65.69
Full-15	102626556	0.07%	0.03%	0.00%	0.00%	0.00%	0.00%
		26.06	1446.98	30.67	35.34	36.66	34.85
Full-16	102658223	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		44.38	66.01	52.78	73.50	57.24	53.22
Full-17	103588498	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		586.30	2264.28	79.12	68.18	61.04	64.09
Full-18	103107136	0.04%	0.01%	0.00%	0.00%	0.00%	0.00%
		42.60	717.67	38.83	41.84	42.52	44.75
Full-19	103615765	0.03%	0.01%	0.00%	0.00%	0.00%	0.00%
		18.07	1788.92	53.55	36.89	41.24	38.81
Full-20	103382324	0.08%	0.02%	0.00%	0.00%	0.00%	0.00%
		24.36	729.66	43.71	36.64	41.91	67.58
Full-21	103531568	0.03%	0.03%	0.00%	0.00%	0.00%	0.00%
		13.67	300.00	41.33	38.46	38.92	41.36
Full-22	103989391	0.22%	0.01%	0.00%	0.00%	0.00%	0.00%
		29.75	1742.55	54.44	52.27	51.87	56.22
Full-23	103842546	0.04%	0.03%	0.00%	0.00%	0.00%	0.00%
		17.17	572.28	32.97	34.32	31.26	35.22
Full-24	104727560	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		122.45	1127.84	46.75	52.36	60.74	49.27

Instance	Exact	NDO master		(PS_{hull})			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
Full-25	104419774	0.09%	0.03%	0.00%	0.00%	0.00%	0.00%
		41.98	1645.45	39.41	37.67	44.30	46.19
Full-26	104695072	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		542.49	1712.39	64.45	63.55	62.02	54.79
Full-27	104571556	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		39.56	25.13	55.70	45.21	55.81	44.68
Full-28	104832909	0.18%	0.01%	0.00%	0.00%	0.00%	0.00%
		26.03	1789.38	58.89	53.46	56.80	54.92
Full-29	104733344	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		26.90	509.12	76.42	51.36	59.03	58.43
Full-30	105063205	0.01%	0.05%	0.00%	0.00%	0.00%	0.00%
		115.09	1078.33	80.92	68.10	67.48	65.95
Full-31	105108478	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		21.94	396.33	51.55	80.43	105.21	82.76
Full-32	105425737	0.07%	0.01%	0.00%	0.00%	0.00%	0.00%
		215.03	2875.04	90.49	102.37	79.97	91.77
Full-33	105298999	0.13%	0.02%	0.00%	0.00%	0.00%	0.00%
		27.23	3213.59	81.93	88.73	85.43	89.49
Full-34	105177415	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		95.94	706.19	48.65	51.89	53.37	57.05
Full-35	105399665	0.11%	0.03%	0.00%	0.00%	0.00%	0.00%
		35.20	2119.91	58.01	60.66	58.55	61.49
Full-36	105636810	0.06%	0.01%	0.00%	0.00%	0.00%	0.00%
		301.36	4587.69	65.98	71.17	70.32	63.58
Full-37	105879289	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		804.58	3208.55	104.39	108.28	115.77	113.89
Full-38	105779654	0.09%	0.01%	0.00%	0.00%	0.00%	0.00%
		36.70	1267.60	71.08	53.33	65.19	59.12
Full-39	105971166	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		247.63	819.93	93.60	150.25	132.10	126.43
Full-40	106222815	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		414.72	4483.03	81.36	73.07	82.67	73.70
Full-41	106561264	0.00%	0.02%	0.00%	0.00%	0.00%	0.00%
		412.67	4886.97	86.27	90.31	95.74	85.76
Full-42	106746217	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		3447.71	7247.01	146.03	123.25	129.94	127.06
Full-43	106512301	0.07%	0.03%	0.00%	0.00%	0.00%	0.00%
		27.92	1249.11	76.41	78.90	113.04	80.31
Full-44	106408488	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		687.46	3452.16	106.00	108.12	87.84	95.67
Full-45	106668481	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		1496.07	3580.22	106.38	116.97	99.97	116.64
Full-46	106899314	0.05%	0.01%	0.00%	0.00%	0.00%	0.00%
		193.64	4971.98	71.72	68.04	84.02	85.31
Full-47	106945452	0.05%	0.01%	0.00%	0.00%	0.00%	0.00%
		691.75	4797.06	96.36	92.07	93.96	99.68
Full-48	107319900	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		47.71	586.24	72.63	76.17	83.94	83.48
Full-49	107499600	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		483.52	4061.86	263.68	232.19	236.13	225.81

Instance	Exact	NDO master		(PS_{hull})			
		Bundle	Volume	$\emptyset$	ℓ_1	ℓ_∞	$\ell_1 + \ell_\infty$
Full-50	107599670	0.06%	0.05%	0.00%	0.00%	0.00%	0.00%
		3215.40	7206.65	291.44	243.89	371.49	268.01
Full-51	107654237	0.07%	0.04%	0.00%	0.00%	0.00%	0.00%
		2985.16	7200.59	158.63	174.93	162.16	180.13
Full-52	107347315	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		106.08	1659.50	87.49	89.91	116.80	82.55
Full-53	107629820	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		2979.72	7254.48	148.82	136.31	173.78	151.74
Full-54	107709781	0.04%	0.02%	0.00%	0.00%	0.00%	0.00%
		1929.38	7244.02	159.98	161.71	141.87	164.23
Full-55	107603587	0.04%	0.03%	0.00%	0.00%	0.00%	0.00%
		223.57	2287.77	81.90	71.38	98.11	80.14
Full-56	107835481	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		2427.45	4714.15	130.91	127.43	134.57	119.77
Full-57	108204009	0.04%	0.02%	0.00%	0.00%	0.00%	0.00%
		39.05	1723.92	61.58	77.32	77.65	79.70
Full-58	108260976	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		520.70	4044.33	215.01	185.51	186.31	216.30
Full-59	108344093	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		3408.55	7211.28	128.24	171.61	167.43	144.61
Full-60	108359891	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		919.51	4532.35	114.05	109.99	110.62	125.45
Full-61	108470640	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		882.54	3383.40	103.34	101.91	110.21	125.08
Full-62	108091600	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		282.37	4580.11	65.86	57.77	58.62	62.41
Full-63	107854681	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		23.86	27.94	50.15	47.58	55.17	47.70
Full-64	108355427	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		5558.77	7272.35	310.23	311.62	344.46	342.85
Full-65	108301483	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		302.10	4383.85	111.68	103.35	114.80	101.18
Full-66	108562860	0.02%	0.04%	0.00%	0.00%	0.00%	0.00%
		7206.39	7228.66	303.57	3784.73	439.34	1122.10
Full-67	109027223	0.01%	0.08%	0.00%	0.00%	0.00%	0.00%
		4152.86	7230.01	222.42	295.29	319.19	265.82
Full-68	109071187	0.01%	0.06%	0.00%	0.00%	0.00%	0.00%
		7270.69	7277.82	324.19	507.28	628.58	506.00
Full-69	108966429	0.01%	0.08%	0.00%	0.00%	0.00%	0.00%
		2450.80	7252.83	322.06	359.92	470.47	406.88
Full-70	108739438	0.02%	0.02%	0.00%	0.00%	0.00%	0.00%
		52.77	2295.97	113.74	76.05	70.32	70.94
Full-71	109017680	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%
		7221.35	5640.09	247.33	290.73	447.50	385.16
Full-72	108811565	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		1069.37	7253.35	127.16	122.72	173.70	139.00
Full-73	109039618	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		2726.24	7279.50	175.68	320.34	388.82	313.90
Full-74	109482028	0.02%	0.09%	0.00%	0.00%	0.00%	0.00%
		7206.59	7270.35	727.72	2183.64	1455.99	3246.13

Instance	Exact	NDO master Bundle	Volume	(PS_{hull}) $\emptyset$	ℓ_1	ℓ_∞	$\ell_1+\ell_\infty$
Full-75	109394833	0.02%	0.06%	0.01%	0.01%	0.01%	0.01%
		5794.65	7254.26	322.45	3166.46	673.15	4502.90
Full-76	109481987	0.02%	0.11%	0.00%	0.00%	0.00%	0.00%
		7161.66	7273.16	311.91	314.32	315.19	363.39
Full-77	109349348	0.01%	0.04%	0.00%	0.00%	0.00%	0.00%
		5819.68	7236.55	340.47	299.72	316.75	302.88
Full-78	109242830	0.02%	0.01%	0.00%	0.00%	0.00%	0.00%
		231.11	3401.80	151.89	167.21	186.00	139.84
Full-79	109292026	0.01%	0.08%	0.00%	0.00%	0.00%	0.00%
		7223.10	7229.98	314.15	470.42	742.49	588.76
Full-80	109418679	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		3240.60	7240.92	218.27	257.24	217.41	239.79
Full-81	109418342	0.05%	0.05%	0.00%	0.00%	0.00%	0.00%
		1774.88	7239.52	179.95	174.21	148.29	182.19
Full-82	109596968	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		3707.80	7186.73	236.10	248.59	223.39	235.51
Full-83	109572186	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		1762.61	4908.27	189.62	231.54	204.21	187.87
Full-84	109948400	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		7223.71	6276.61	469.33	471.93	472.19	487.73
Full-85	109898382	0.01%	0.09%	0.00%	0.00%	0.00%	0.00%
		7291.31	7213.35	271.89	304.75	289.23	281.20
Full-86	109789890	0.01%	0.04%	0.00%	0.00%	0.00%	0.00%
		5072.46	7217.27	180.53	181.57	191.14	173.04
Full-87	109734474	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		659.91	5753.04	184.08	163.81	186.18	177.35
Full-88	109862389	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		449.76	6181.58	217.36	241.82	235.24	198.58
Full-89	109738695	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		516.71	5004.32	157.49	133.81	169.94	138.34
Full-90	109715464	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		145.02	1007.22	82.47	88.62	100.64	119.43
Full-91	110010434	0.03%	0.12%	0.01%	0.01%	0.01%	0.01%
		7270.26	7214.58	397.32	548.29	649.41	550.48
Full-92	109950201	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		851.99	5096.60	112.06	110.51	116.57	101.81
Full-93	110133703	0.13%	0.03%	0.00%	0.00%	0.00%	0.00%
		1629.07	7287.18	348.38	420.14	379.42	337.47
Full-94	110201145	0.00%	0.03%	0.00%	0.00%	0.00%	0.00%
		3775.72	7249.07	277.64	338.46	539.20	393.42
Full-95	110312649	0.04%	0.04%	0.00%	0.00%	0.00%	0.00%
		7288.89	7255.17	391.13	551.73	523.54	509.22
Full-96	110287755	0.01%	0.04%	0.00%	0.00%	0.00%	0.00%
		7203.71	7286.30	382.43	474.71	406.77	477.71
Full-97	110306919	0.01%	0.01%	0.00%	0.00%	0.00%	0.00%
		2309.40	5100.76	209.47	166.75	168.75	155.91
Full-98	110303293	0.01%	0.03%	0.00%	0.00%	0.00%	0.00%
		2142.36	7300.24	234.50	236.77	222.91	202.21
Full-99	110370538	0.01%	0.02%	0.00%	0.00%	0.00%	0.00%
		3600.26	7232.89	261.29	292.36	255.41	269.53

Appendix VI

Bound strengths (gap %) and computation times (s) for industrial instances

Negative gaps indicate improvement on the lower or upper bound computed by the parallel branch-and-cut of CPLEX after 4 hours. Times are wallclock (real) times with 24 cores available. Results for (M_{hull}) correspond to the warm-started simplex. Tables of results for the two Lagrangian decomposition duals follow.

	MIP (4h)		Lower bounds (M)			Lagrangian heuristic (10 min)			MIP (10 min)		Lagrangian heuristic (15 min)			MIP (15 min)	
Instances	Lower	Upper	weak	strong	hull	Subproblem	Warm start	Primal	Lower	Upper	Subproblem	Warm start	Primal	Lower	Upper
S(1,1,$\frac{1}{2}$)	48644591	48649454	16.26%	2.34%	1.08%	2.34%	1.08%	0.77%	-0.00%	0.00%	2.34%	1.08%	0.77%	0.03%	0.00%
			0.02	0.06	2.33	0.06	2.23	300.02	31.75	31.75	0.06	2.23	450.01	900.02	900.02
S(1,1,1)	48389175	48394013	22.08%	2.36%	1.14%	2.36%	1.14%	0.62%	0.03%	0.00%	2.36%	1.14%	0.62%	0.00%	0.00%
			0.02	0.06	1.18	0.04	1.44	300.01	600.01	600.01	0.05	1.45	450.01	623.07	623.07
S(1,2,$\frac{1}{2}$)	47803225	47807847	21.72%	1.37%	0.51%	1.37%	0.51%	0.00%	0.00%	0.00%	1.37%	0.51%	0.00%	-0.00%	0.00%
			0.02	0.05	0.98	0.05	0.86	0.17	0.92	0.92	0.05	0.85	2.01	0.92	0.92
S(1,2,1)	47378693	47382376	27.39%	1.21%	0.42%	1.21%	0.42%	0.00%	0.00%	0.00%	1.21%	0.42%	0.00%	0.00%	0.00%
			0.02	0.04	0.38	0.05	0.39	0.11	0.79	0.79	0.05	0.37	0.16	0.85	0.85
S(2,1,$\frac{1}{2}$)	58643592	58649454	13.99%	1.94%	0.89%	1.94%	0.89%	0.15%	-0.00%	0.00%	1.94%	0.89%	0.15%	0.00%	0.00%
			0.01	0.06	2.40	0.05	1.91	300.01	29.36	29.36	0.05	1.89	450.02	59.57	59.57
S(2,1,1)	58388177	58394013	19.26%	1.96%	0.94%	1.96%	0.94%	0.00%	0.00%	0.00%	1.96%	0.94%	0.00%	0.00%	0.00%
			0.02	0.06	1.15	0.05	1.13	0.40	89.44	89.44	0.05	1.10	0.69	183.91	183.91
S(2,2,$\frac{1}{2}$)	57804594	57807847	18.66%	1.14%	0.43%	1.14%	0.43%	0.00%	-0.00%	0.00%	1.14%	0.43%	0.00%	0.00%	0.00%
			0.02	0.06	0.89	0.05	0.55	0.18	1.08	1.08	0.05	0.53	0.16	0.96	0.96
S(2,2,1)	57377611	57382376	23.81%	1.00%	0.34%	1.00%	0.34%	0.00%	-0.01%	0.00%	1.00%	0.34%	0.00%	-0.01%	0.00%
			0.01	0.04	0.43	0.05	0.48	0.10	1.01	1.01	0.05	0.48	0.09	2.25	2.25

	MIP (4h)		Lower bounds (M)			Lagrangian heuristic (10 min)			MIP (10 min)		Lagrangian heuristic (15 min)			MIP (15 min)	
Instances	Lower	Upper	weak	strong	hull	Subproblem	Warm start	Primal	Lower	Upper	Subproblem	Warm start	Primal	Lower	Upper
M(1,1,$\frac{1}{2}$)	81612004	81874987	12.85%	1.28%	0.74%	1.28%	0.76%	0.00%	0.44%	0.08%	1.28%	0.75%	0.01%	0.40%	0.02%
			0.22	20.06	144.80	0.93	295.09	300.06	600.07	600.07	0.93	445.26	450.05	900.06	900.06
M(1,1,1)	79917612	79969202	18.49%	1.08%	0.65%	1.08%	0.65%	0.00%	0.16%	0.00%	1.08%	0.65%	0.00%	0.17%	0.02%
			0.19	6.50	93.55	0.84	48.55	300.04	600.06	600.06	0.84	48.60	450.04	900.05	900.05
M(1,2,$\frac{1}{2}$)	77286934	77416633	16.59%	1.14%	0.70%	1.14%	0.70%	0.00%	0.39%	0.08%	1.14%	0.70%	0.03%	0.26%	0.04%
			0.23	13.61	116.50	1.24	294.15	300.05	600.04	600.04	1.25	386.08	450.04	900.06	900.06
M(1,2,1)	75977348	75984944	22.27%	1.04%	0.61%	1.04%	0.61%	0.00%	0.07%	0.00%	1.04%	0.61%	0.00%	0.09%	0.00%
			0.19	3.04	100.05	1.04	142.36	300.20	600.06	600.06	1.04	197.94	450.04	900.08	900.08
M(2,1,$\frac{1}{2}$)	92595876	92874987	11.29%	1.11%	0.62%	1.11%	0.64%	0.00%	0.26%	0.03%	1.11%	0.63%	0.01%	0.12%	0.00%
			0.21	12.36	160.00	1.36	294.96	300.07	600.08	600.08	1.37	445.16	450.06	900.07	900.07
M(2,1,1)	90960107	90969202	16.29%	0.99%	0.57%	0.99%	0.57%	0.00%	0.13%	0.00%	0.99%	0.57%	0.00%	0.19%	0.01%
			0.20	5.16	116.58	1.11	137.77	300.05	600.07	600.07	1.12	140.56	450.04	900.08	900.08
M(2,2,$\frac{1}{2}$)	87990196	88064952	14.24%	1.43%	0.96%	1.43%	0.99%	0.02%	0.60%	0.03%	1.43%	0.97%	0.00%	0.51%	0.03%
			0.24	11.65	128.96	1.33	295.20	300.10	600.06	600.06	1.34	444.75	450.05	900.05	900.05
M(2,2,1)	86560526	86573008	19.47%	1.34%	0.90%	1.34%	0.90%	0.00%	0.19%	0.05%	1.34%	0.90%	0.00%	0.22%	0.01%
			0.20	2.20	89.62	1.14	295.47	300.04	600.05	600.05	1.13	294.64	450.04	900.06	900.06

	MIP (4h)		Lower bounds (M)			Lagrangian heuristic (10 min)			MIP (10 min)		Lagrangian heuristic (15 min)			MIP (15 min)	
Instances	Lower	Upper	weak	strong	hull	Subproblem	Warm start	Primal	Lower	Upper	Subproblem	Warm start	Primal	Lower	Upper
L(1,1,$\frac{1}{2}$)	104294318	106288388	10.39%	1.21%	0.39%	1.21%	0.62%	0.00%	0.07%	1.70%	1.21%	0.54%	0.00%	0.11%	2.14%
			0.79	173.68	1596.76	8.63	266.36	300.16	600.60	600.60	8.58	415.86	450.18	900.19	900.19
L(1,1,1)	102083120	102740054	16.17%	1.35%	0.46%	1.35%	0.75%	0.25%	0.41%	0.67%	1.35%	0.57%	0.45%	0.26%	0.63%
			0.99	115.26	998.33	4.77	276.43	300.13	600.23	600.23	4.70	424.93	450.13	900.23	900.23
L(1,2,$\frac{1}{2}$)	95960111	97148978	11.99%	1.11%	0.51%	1.11%	0.64%	0.07%	0.08%	0.54%	1.11%	0.63%	0.05%	0.08%	0.55%
			0.88	128.00	1363.33	6.18	264.20	300.21	600.35	600.35	9.93	411.85	450.14	900.23	900.23
L(1,2,1)	94218226	94912559	18.54%	1.29%	0.56%	1.29%	0.73%	0.00%	0.15%	0.30%	1.29%	0.73%	0.00%	0.13%	0.20%
			1.56	125.73	877.76	9.34	264.15	300.16	600.31	600.31	8.64	416.81	450.13	900.13	900.13
L(2,1,$\frac{1}{2}$)	116154927	117959416	9.02%	0.97%	0.21%	0.97%	0.39%	0.01%	0.05%	0.22%	0.97%	0.33%	0.00%	-0.00%	0.27%
			0.93	194.35	1411.73	6.55	266.99	300.18	602.37	602.37	6.65	416.23	450.18	901.84	901.84
L(2,1,1)	114146561	114740054	14.49%	1.26%	0.50%	1.26%	0.65%	0.10%	0.35%	1.12%	1.26%	0.60%	0.00%	0.34%	0.64%
			1.04	49.91	1109.20	8.75	268.72	300.15	600.12	600.12	8.58	417.78	450.14	900.23	900.23
L(2,2,$\frac{1}{2}$)	107586189	109218948	10.19%	0.80%	0.16%	0.80%	0.46%	0.25%	-0.02%	0.74%	0.80%	0.41%	0.08%	0.08%	2.18%
			1.77	195.08	1223.82	12.57	257.90	300.16	600.91	600.91	26.21	394.08	450.16	900.31	900.31
L(2,2,1)	105459522	106440038	16.04%	0.97%	0.24%	0.97%	0.46%	0.12%	0.11%	0.67%	0.97%	0.38%	0.08%	0.20%	1.38%
			3.26	75.42	918.10	31.84	242.23	300.12	600.30	600.30	11.17	412.18	450.15	900.14	900.14

	MIP (4h)		Lower bounds (M)			Lagrangian heuristic (10 min)			MIP (10 min)		Lagrangian heuristic (15 min)			MIP (15 min)	
Instances	Lower	Upper	weak	strong	hull	Subproblem	Warm start	Primal	Lower	Upper	Subproblem	Warm start	Primal	Lower	Upper
F(1,1,$\frac{1}{2}$)	124630445	127415589	11.48%	0.53%	0.04%	0.53%	0.50%	0.00%	0.12%	3.72%	0.53%	0.39%	0.10%	0.10%	2.52%
			6.81	771.48	12875.34	57.96	102.94	300.26	601.57	601.57	70.90	255.21	450.55	900.35	900.35
F(1,1,1)	121237220	123044513	17.60%	0.49%	0.08%	0.49%	0.40%	0.37%	0.16%	1.66%	0.49%	0.35%	0.00%	0.09%	1.40%
			11.37	461.46	5777.52	53.56	94.50	300.29	600.53	600.53	53.29	249.79	450.37	900.36	900.36
F(1,2,$\frac{1}{2}$)	112693766	114986432	13.51%	0.50%	0.02%	0.50%	0.49%	0.84%	0.23%	4.95%	0.50%	0.45%	0.15%	0.23%	4.68%
			5.68	766.29	9763.50	47.48	84.82	300.34	601.09	601.09	47.00	237.54	450.30	902.40	902.40
F(1,2,1)	110448574	112450491	20.21%	0.45%	-0.07%	0.45%	0.43%	0.00%	0.17%	2.33%	0.45%	0.39%	0.00%	0.02%	2.63%
			10.58	503.51	5513.47	71.40	75.97	300.25	600.30	600.30	71.80	231.19	450.34	900.47	900.47
F(2,1,$\frac{1}{2}$)	137192665	140015862	9.73%	0.44%	0.04%	0.49%	0.49%	0.47%	0.29%	4.84%	0.44%	0.34%	0.27%	0.07%	4.12%
			11.01	650.43	14418.69	133.86	0.00	300.28	601.07	601.07	140.43	153.25	450.31	900.49	900.49
F(2,1,1)	133113038	134475487	15.07%	0.63%	0.21%	0.63%	0.57%	0.00%	0.22%	1.12%	0.63%	0.46%	0.00%	0.16%	1.87%
			6.81	290.15	5585.90	89.12	74.80	300.33	601.33	601.33	89.16	227.51	450.39	900.40	900.40
F(2,2,$\frac{1}{2}$)	124970860	127606690	10.99%	0.39%	-0.10%	0.39%	0.39%	0.10%	0.16%	2.52%	0.39%	0.34%	0.00%	0.08%	2.72%
			8.69	901.65	11254.22	107.89	20.15	300.21	600.54	600.54	110.19	172.64	450.35	900.61	900.61
F(2,2,1)	121957537	123546643	17.15%	0.48%	-0.06%	0.48%	0.46%	0.00%	0.22%	1.81%	0.48%	0.36%	0.00%	0.23%	1.01%
			18.81	454.92	4792.30	90.42	43.91	300.31	600.47	600.47	90.80	198.35	450.71	900.55	900.55

Instance	MIP (4h)	Lower bounds (300 sec)			Lower bounds (450 sec)		
	(Lower bound)	Bundle	Volume	Warm start	Bundle	Volume	Warm start
S(1,1,$\frac{1}{2}$)	48644591	1.12%	1.14%	1.08%	1.11%	1.14%	1.08%
		300.00	5.06	2.79	450.00	5.10	2.60
S(1,1,1)	48389175	1.17%	1.21%	1.14%	1.16%	1.21%	1.14%
		300.01	5.63	1.82	450.01	5.58	1.85
S(1,2,$\frac{1}{2}$)	47803226	0.55%	0.52%	0.51%	0.55%	0.52%	0.51%
		66.37	4.94	1.24	67.08	4.91	1.44
S(1,2,1)	47378694	0.50%	0.46%	0.42%	0.50%	0.46%	0.42%
		11.48	3.99	0.83	11.41	3.97	0.82
S(2,1,$\frac{1}{2}$)	58643592	0.92%	0.94%	0.89%	0.91%	0.94%	0.89%
		300.01	5.55	2.35	450.00	5.65	2.27
S(2,1,1)	58388178	0.96%	1.00%	0.94%	0.95%	1.00%	0.94%
		300.01	4.60	1.51	344.18	4.58	1.58
S(2,2,$\frac{1}{2}$)	57804595	0.45%	0.45%	0.43%	0.45%	0.45%	0.43%
		71.91	4.54	0.98	71.66	4.56	1.02
S(2,2,1)	57377611	0.41%	0.37%	0.34%	0.41%	0.37%	0.34%
		12.16	3.99	1.01	12.35	3.88	1.01

Instance	MIP (4h)	Lower bounds (300 sec)			Lower bounds (450 sec)		
	(Lower bound)	Bundle	Volume	Warm start	Bundle	Volume	Warm start
M(1,1,$\frac{1}{2}$)	81612005	0.93%	0.86%	0.76%	0.92%	0.86%	0.75%
		300.08	277.89	300.07	450.04	293.48	449.97
M(1,1,1)	79917613	0.78%	0.68%	0.65%	0.76%	0.68%	0.65%
		300.03	224.38	52.76	450.02	212.67	52.69
M(1,2,$\frac{1}{2}$)	77286935	1.12%	0.88%	0.70%	1.11%	0.85%	0.70%
		300.02	301.77	299.95	450.06	450.16	391.84
M(1,2,1)	75977349	0.73%	0.71%	0.61%	0.72%	0.71%	0.61%
		300.03	205.47	147.33	450.03	199.81	202.96
M(2,1,$\frac{1}{2}$)	92595877	0.76%	0.72%	0.64%	0.76%	0.72%	0.63%
		300.07	212.06	300.04	450.06	205.43	450.25
M(2,1,1)	90960108	0.63%	0.61%	0.57%	0.62%	0.61%	0.57%
		300.05	270.56	142.22	450.01	268.74	145.09
M(2,2,$\frac{1}{2}$)	87990196	1.42%	1.11%	0.99%	1.42%	1.11%	0.97%
		300.04	297.01	300.41	450.05	288.45	450.01
M(2,2,1)	86560526	1.04%	1.01%	0.90%	1.02%	1.01%	0.90%
		300.03	243.61	300.14	450.03	255.51	299.30

Instance	MIP (4h)	Lower bounds (300 sec)			Lower bounds (450 sec)		
	(Lower bound)	Bundle	Volume	Warm start	Bundle	Volume	Warm start
L(1,1,$\frac{1}{2}$)	104294319	1.21%	1.08%	0.62%	1.21%	1.00%	0.54%
		300.18	305.47	297.04	450.16	464.60	446.93
L(1,1,1)	102083121	0.78%	1.19%	0.75%	0.73%	1.19%	0.57%
		300.20	320.30	299.45	450.14	455.70	447.97
L(1,2,$\frac{1}{2}$)	95960112	1.11%	0.99%	0.64%	1.11%	0.98%	0.63%
		300.18	300.46	297.24	450.19	459.64	448.28
L(1,2,1)	94218226	0.88%	1.09%	0.73%	0.83%	1.10%	0.73%
		300.18	305.92	297.98	450.12	450.71	450.26
L(2,1,$\frac{1}{2}$)	116154928	0.97%	0.80%	0.39%	0.97%	0.76%	0.33%
		300.21	301.09	296.72	450.15	451.19	446.53
L(2,1,1)	114146562	0.77%	1.00%	0.65%	0.73%	1.00%	0.60%
		300.04	309.83	298.28	450.16	458.75	447.17
L(2,2,$\frac{1}{2}$)	107586189	0.80%	0.67%	0.46%	0.80%	0.66%	0.41%
		300.06	301.04	296.56	450.06	455.34	446.50
L(2,2,1)	105459523	0.97%	0.90%	0.46%	0.96%	0.81%	0.38%
		300.23	300.67	299.65	450.16	459.81	448.43

Instance	MIP (4h)	Lower bounds (300 sec)			Lower bounds (450 sec)		
	(Lower bound)	Bundle	Volume	Warm start	Bundle	Volume	Warm start
F(1,1,$\frac{1}{2}$)	124630446	0.53%	0.52%	0.50%	0.52%	0.46%	0.39%
		300.32	300.54	298.66	450.32	450.10	464.29
F(1,1,1)	121237220	0.49%	0.48%	0.40%	0.49%	0.41%	0.35%
		300.16	301.02	299.50	450.24	462.84	455.21
F(1,2,$\frac{1}{2}$)	112693767	0.50%	0.50%	0.49%	0.50%	0.47%	0.45%
		300.25	300.36	297.93	450.46	451.96	450.71
F(1,2,1)	110448575	0.45%	0.45%	0.43%	0.45%	0.43%	0.39%
		300.35	301.17	296.20	450.49	450.37	452.73
F(2,1,$\frac{1}{2}$)	137192665	0.44%	0.43%	0.49%	0.44%	0.32%	0.34%
		300.09	300.18	286.74	450.15	452.52	447.16
F(2,1,1)	133113039	0.58%	0.56%	0.57%	0.55%	0.53%	0.46%
		300.14	300.70	299.43	450.23	450.40	453.66
F(2,2,$\frac{1}{2}$)	124970861	0.39%	0.39%	0.39%	0.39%	0.30%	0.34%
		300.78	300.37	295.26	450.27	452.41	449.97
F(2,2,1)	121957538	0.48%	0.47%	0.46%	0.48%	0.32%	0.36%
		300.36	300.11	295.80	450.14	452.50	450.58