# Compact MIP Models for the Resource-Constrained Project Scheduling Problem

Masterarbeit bei
Prof. Dr. Ralf Borndörfer

Zweitkorrektur: Prof. Dr. Rolf H. Möhring

vorgelegt von
Alexander Tesch
Technische Universität Berlin
Fachbereich Mathematik

Juni 2015

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Berlin, den

........................................................
Unterschrift

# Zusammenfassung

In dem *ressourcenbeschränkten Projektplanungsproblem* (RCPSP) wird ein Zeitplan für eine Menge von Jobs erstellt, wobei Jobs unterschiedliche Anforderungen an Ressourcen- und Vorrangrestriktionen aufweisen. Die gesamte Projektdauer, das heißt die Fertigstellungszeit des letzten Jobs, soll minimiert werden. Zur Lösung dieses Problems existiert eine Reihe von Ansätzen, die auf gemischt-ganzzahliger Programmierung (MIP) basieren. Die meisten aktuellen Modelle verwenden eine Diskretisierung des Planungszeitraums in Einheitsintervalle. Dabei wird jeder Job einem gültigen Zeitintervall zugeordnet. Ein Nachteil von zeitdiskreten Modellen entsteht für große Planungszeiträume oder besonders feine Diskretisierungen. Die dadurch wachsende Modelgröße macht eine effiziente Berechnung kaum möglich. In diesem Zusammenhang untersuchen wir *kompakte* MIP Modelle, deren Größe unabhängig vom Planungshorizont ist. Zusätzlich zu zwei kompakten Modellen aus der Literatur präsentieren wir zwei neue kompakte Modelle. Wir untersuchen die induzierten Polyeder und leiten eine Inklusionshierarchie ab, die auf linearen Transformationen basiert. Diese Transformationen werden kombinatorisch interpretiert. Weiterhin betrachten wir eine Klasse von Schnittebenen, sogenannte *Cover Ungleichungen*. Letztere werden durch einen effizienten Lifting-Algorithmus verstärkt, dessen Laufzeit unabhängig vom gewählten Modell ist. Darüber hinaus erforschen wir untere Schranken für das Planungsproblem, die mit linearen Programmen berechnet werden. Zwischen zwei bestehenden Modellen aus der Literatur identifizieren wir eine primal-duale Verbindung und generieren resultierende Schnittebenen für eines der Modelle. Es werden zwei Schnittebenenalgorithmen vorgestellt. Wir zeigen außerdem, dass dieselbe Charakterisierung auf Schnittebenen der kompakten Modelle übertragen werden kann. Unsere Modelle wurden auf den Testinstanzen der PSPLIB [65] implementiert, getestet und bewertet.

# Abstract

In the *Resource-Constrained Project Scheduling Problem* (RCPSP) a set of jobs is planned subject to resource- and precedence constraints. The objective is to minimize the makespan, that is the time when all jobs have been completed. There exist several Mixed-Integer-Programming (MIP) models in order to solve the problem. Most common models are based on time-discretization. In this case, the scheduling horizon is split into unit size intervals and each job gets assigned a unique starting interval. The drawback of time-discrete models is the computational intractability for large scheduling horizons or fine discretizations. In this connection, this thesis deals with *compact* MIP models where the model size is independent of the scheduling horizon. In addition to two compact models from the literature, we present two new compact models. We investigate their induced polyhedra and deduce an inclusion hierarchy via linear transformations. Moreover, we give a combinatorial interpretation of these transformations. Furthermore, we study a class of valid cutting planes for the compact models, which are known as *cover inequalities*. In order to strengthen these cutting planes we introduce a lifting algorithm that is independent of the model size. Subsequently, we examine lower bounds for the RCPSP from linear programming. Based on a linear transformation, we reveal a connection between two approaches from the literature. For one model we generate strong cutting planes that are obtained from a primal-dual relation between the models. Two cutting plane algorithms are derived. Likewise, we show that similar cutting planes can be transferred to the compact MIP models. Our models have been implemented, tested and evaluated on the instances of the PSPLIB [65] problem library.

# Danksagungen

Ich bedanke mich besonders bei Prof. Dr. Ralf Borndörfer für die Unterstützung, Hilfsbereit und das Vertrauen diese Arbeit zu einem Thema meiner Wahl anzufertigen. Diese Arbeit zeigt, dass eine effektivere Zeitplanung stets aktuellen Stellenwert besitzt.

Mein Dank gilt weiterhin Prof. Dr. Rolf Möhring, dessen Vorlesungen und Seminare mir eine sehr gute Grundlage für die Thematik geschaffen haben. Für viele nützliche Anmerkungen und Korrekturen bedanke ich mich außerdem bei Heide Hoppman, Kai Hennig und Jakob Witzig. Weiterhin danke ich dem ganzen ZIB für eine angenehme und konstruktive Arbeitsatmosphäre.

# Contents

# Chapter 1

# Introduction

Project scheduling under limited resources has applications in almost all kind of modern industry. Many recent real-world examples show that improper project planning may result in unfavorably high cost. Since the general RCPSP is very hard to compute [3], efficient methods are needed to generate valuable project schedules. In this thesis we investigate the deterministic RCPSP without preemption. That is, jobs have a constant duration and cannot be interrupted while being processed. In addition, there is a set of resources with limited capacity where each job has a certain demand of each resource. Moreover, there exist precedence relations between the jobs. The objective is to minimize the latest completion time of all jobs, which is also denoted as the *makespan* of the project. One possibility to compute a project schedule with minimum makespan is provided by Mixed-Integer Programming (MIP). Most common models for the RCPSP rely on time-discretization where the scheduling horizon is decomposed into discrete time intervals of unit length. The jobs get allocated to feasible starting times which respect the resource and precedence restrictions. Since the model size expands with increasing scheduling horizon, time-discrete models become intractable for large time horizons. This justifies our thesis to account for *compact models* whose size is independent of the planning period.

My motivation for this topic originates from my participation at the *AIMMS-MOPTA Modeling Competition 2013* which was awarded the first prize [1]. The task was to develop optimization tools for efficient operation room scheduling under limited resources. The straightforward problem structure but the hardness of their computation inspired me to consider the more general RCPSP.

## 1.1 Related Work and Outline

One of the first but commonly used integer programming models for the RCPSP is due to Pritsker et al. [12]. They introduce a time-discrete model (DT) that assigns jobs to feasible starting times. Based on their model, Christofides et al. [7] propose an extended model with disaggregated precedence constraints (DDT) which yields a stronger LP-relaxation. Further extensions of Pritsker's model can be found in Patterson et al. [15] and Stinson et al. [16]. Many of the former and current solving approaches use variants of the time-discrete model. Möhring et al. [18] exploit integral substructures of the model when the resource constraints are relaxed. The resulting integer program can be computed efficiently by minimum-cut detections in a problem-defining network. The initial problem is then solved by Lagrangean Relaxation. Other approaches focus on strengthening the original formulation. For example, Calvacante et al. [59] introduce GUB cover inequalities and construct an LP-based Branching algorithm. Similarly, Hardin et al. [60] suggest minimal cover inequalities for the uniform resource case in connection with fast lifting methods.

A different non-time-discrete MIP model was introduced by Alvarez-Valdés and Goerlich [14]. It is based on cuts for incompatible job subsets and large constants. The authors study polyhedral aspects and lifting theorems for a class of valid inequalities. Artigues et al. [17] replaced the incompatible job cuts by a flow extension (FCT). This yields a compact representation but large constants remain in the model. Most recent work on compact formulations can be found in Koné et al. [8, 9]. They introduce two compact models which decompose the scheduling horizon into intervals of variable length. The authors provide an experimental study concerning the solvability compared to DDT and FCT. Their testings reveal that their compact models can solve small problem instances but they are considerably weaker than DDT on common problem instances of the PSPLIB [65]. Apart from that, they showed that DDT is outperformed by the compact models if the job durations are scaled by a sufficiently large factor. But in general, the compact models provide a weak alternative to DDT in terms of practical solvability.

In addition to exact solving approaches, the literature studies lower bounds for the RCPSP. Mostly, lower bounds are computed by so-called destructive improvement, see Klein et al. [30] and Dorndorf et al. [29]. This aims to solve the decision problem for fixed makespan. If we can prove that the considered makespan is infeasible, the lower bound is increased. The procedure is repeated either incrementally or by binary search until no infeasibilities can be found. A powerful lower bound was introduced by Mingozzi et al. [13]. They propose an LP model that is based feasible job subsets and a preemptive relaxation. Brucker and Knust [22] extended their model by time-windows and solved it by column generation. Baptiste and Demassey [24] added further cutting planes to the model. Up to now, this combined approach still provides the best lower bounds on a considerable number of PSPLIB instances. Another lower bound from linear programming is presented by Carlier and Néron [23]. Their model uses a compact representation of the schedule which is strengthened by additional inequalities obtained from different methods like *energetic reasoning* and *redundant functions*, see Carlier and Néron [34]. More valid cutting planes were introduced by Haouari et al. [25]. Advanced reduction tests that are combined with MIP are given in Haouari et al. [26].

Apart from linear programming techniques, Constraint Programming (CP) has a growing impact on scheduling problems including the RCPSP. The algorithms of Schütt et al. [36, 37, 38] combine Branch-and-Bound with SAT and CP solving in an integrated framework. Further efficient reduction tests has been proposed by Vilim [40]. By efficient pruning of the solution space, their algorithms were able to close and improve the lower bound for a significant number of PSPLIB instances. Further studies in combination with CP, IP and SAT can be found in Baptiste et al. [39] and Schulz [27].

### Outline of the Thesis

The thesis is structured as follows. Section 1.2 gives an overview of fundamental definitions from polyhedral theory and Mixed-Integer Programming. In Section 1.3 we formally define the RCPSP and related notation. Chapter 2 introduces the considered MIP models of the thesis. This includes the time-indexed model of Pritsker et al. [12] and two compact models of Koné et al. [8]. Additionally, we present stronger inequalities for the existing compact models. Moreover, we add two new compact formulations. In Chapter 3 we explore their polyhedral relations and deduce an inclusion hierarchy based on linear transformations. A class of so-called *lifted cover inequalities* for the compact models, is addressed in Chapter 4. We give an introduction to the formal method and construct a sequential lifting algorithm. Chapter 5 deals with lower bounds of the RCPSP that are obtained from linear programming. We relate two current models via a primal-dual connection.

From this characterization, we derive two cutting plane algorithms. In addition, we show that the same characterization can be implemented in the compact MIP setting in order to generate strong cutting planes. According to test instances of the PSPLIB, we discuss our computational results in Chapter 6. Finally, we draw a conclusion between compact MIP models and the applicability for solving the RCPSP in Chapter 7.

## 1.2 Preliminaries

In the following we give a brief introduction to basic definitions of polyhedral theory and and Mixed-Integer Programming. For a complete survey we refer to Schrijver [4, Chapter 5].

### Polyhedra

A *halfspace* in $\mathbb{R}^d$ is given by a set $H = \{x \in \mathbb{R}^d \mid \pi^T x \leq \pi_0\}$ with $\pi \in \mathbb{R}^d$ and $\pi_0 \in \mathbb{R}$. A *polyhedron* $P = \{x \in \mathbb{R}^d \mid Ax \leq b\}$ is the intersection of finitely many halfspaces given by a matrix $A \in \mathbb{R}^{m \times d}$ and vector $b \in \mathbb{R}^m$. A *polytope* is a bounded polyhedron. An inequality $\pi^T x \leq \pi_0$ is valid for $P$, if $P$ is contained in its defining halfspace. The dimension of a polyhedron $P$

$$dim(P) = \max\{|V| - 1 \mid V \subseteq P \text{ set of affinely independent points }\}$$

is one less than the maximum number of affinely independent points in $P$. A *face* of $P$ is given by a set $F = \{x \in P \mid \pi^T x = \pi_0\}$ where $\pi^T x \leq \pi_0$ is a valid inequality of $P$. A face $F$ of $P$ is called *facet*, if and only if $dim(F) = dim(P) - 1$. Moreover, $F$ is a *vertex* if and only if $dim(F) = 0$.

The *polar* of $P$ is defined as $P^* = \{\pi \in \mathbb{R}^d \mid x^T \pi \leq 1 \; \forall x \in P\}$. In particular, $P^*$ is again a polyhedron and the vertices of $P^*$ correspond to facets of $P$, see Ziegler [5].

### Linear Programs

A *linear program* (LP) consists of a linear function that is maximized over a polyhedron. We focus on non-negative linear programs. This yields a system

$$\begin{aligned} \max c^T x& \\ Ax &\leq b \\ x &\in \mathbb{R}^d_{\geq 0} \end{aligned}$$

with constraint matrix $A \in \mathbb{R}^{m\times d}$, right-hand side $b \in \mathbb{R}^m$ and objective coefficients $c \in \mathbb{R}^d$. We call this linear program the *primal* LP. The corresponding *dual* LP is given by the system

$$\begin{aligned} \min b^T\pi& \\ A^T\pi &\geq c \\ \pi &\in \mathbb{R}^m_{\geq 0} \end{aligned}$$

where $\pi$ is the vector of dual variables. Given feasible solutions $x$ and $\pi$ for the primal and dual LP, it holds $c^T x \leq b^T\pi$ by weak duality. If $x^*$ and $\pi^*$ are both optimal solutions to the respective linear programs, it holds $c^T x^* = b^T\pi^*$ by strong duality, see [4, Chapter 5.5].

The matrix $A \in \mathbb{R}^{m\times d}$ is *totally unimodular*, if every square non-singular submatrix $A'$ satisfies $\det(A') \in \{-1, 1\}$. If $A$ is a totally unimodular matrix and $b$ is integral then the extremal solutions of the corresponding polyhedron are integral, provided the solution space is non-empty, see [4, Chapter 5.16].

### Mixed-Integer Programs

As before, we restrict to non-negative space. For given matrices $A \in \mathbb{R}^{m\times d}$, $B \in \mathbb{R}^{m\times e}$ and right-hand side $b \in \mathbb{R}^m$ we define the integer hull as the set

$$P_I = conv\{(x, y) \in \mathbb{Z}^d_{\geq 0} \times \mathbb{R}^e_{\geq 0} \mid Ax + By \leq b\}.$$

A *Mixed-Integer Program* (MIP) considers a linear function with coefficients $(c_1, c_2) \in \mathbb{R}^{d+e}$ that is maximized over an integer hull which yields a system

$$\begin{aligned} \max c_1^T x + c_2^T y& \\ Ax + By &\leq b \\ x &\in \mathbb{Z}^d_{\geq 0} \\ y &\in \mathbb{R}^e_{\geq 0}. \end{aligned}$$

The polyhedron of the *linear programming relaxation* of this MIP is given by

$$P = \{(x, y) \in \mathbb{R}^{d+e}_{\geq 0} \mid Ax + By \leq b\}$$

which relaxes the integrality conditions. In particular, it holds $P_I \subseteq P$.

## 1.3 The Resource-Constrained Project Scheduling Problem

In the *Resource-Constrained Project Scheduling Problem* (RCPSP) we are given a set of jobs $I = \{i_1, ..., i_n\}$ with processing times $p_i \in \mathbb{Z}_{>0}$ for all $i \in I$. The jobs must be scheduled *non-preemtively*, that is they cannot be interrupted while being in process. Moreover, we are given a set of resources $R$ with limited capacities $D_r \in \mathbb{Z}_{>0}$ for all $r \in R$. Each job $i \in I$ has a demand of $d_{ir} \in \mathbb{Z}_{\geq 0}$ units of resource $r \in R$. Naturally, we assume $d_{ir} \leq D_r$ for all $i \in I$ and $r \in R$. In addition, there are precedence relations $E \subseteq I \times I$ with $(i, j) \in E$ if $i$ has to complete before $j$ starts. These relations form the precedence graph $G = (I, E)$ which must not contain any directed cycles.

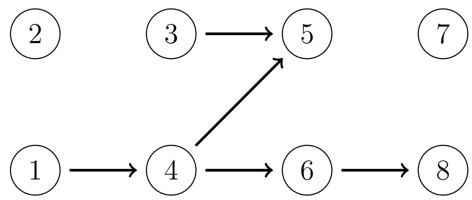
A *schedule* is defined as vector $s = (s_i)_{i \in I} \in \mathbb{R}^n_{\geq 0}$ which consists of the job starting times. Let $\mathcal{T} = [0, \infty)$ be the considered time horizon and $I(t) = \{i \in I \mid s_i \leq t < s_i + p_i\}$ the set of jobs that are active at time $t \in \mathcal{T}$. The makespan is defined as $C_{max} = \max\{s_i + p_i \mid i \in I\}$, which corresponds to the project completion time. The objective of the RCPSP is to minimize the makespan. More formally, the RCPSP can be stated as follows:

$$\begin{aligned} \min\ & C_{max} \\ & s_i + p_i \leq C_{max} && \forall i \in I && (1.1) \\ & \sum_{i \in I(t)} d_{ir} \leq D_r && \forall t \in \mathcal{T}, r \in R && (1.2) \\ & s_i + p_i \leq s_j && \forall (i,j) \in E && (1.3) \\ & s_i \geq 0 && \forall i \in I \end{aligned}$$

A schedule $s \in \mathbb{R}^n_{\geq 0}$ is *feasible*, if it satisfies conditions (1.1)-(1.3). The schedule is *minimal*, if $C_{max}$ is minimal. Consequently, the RCPSP looks for a minimal feasible schedule. The RCPSP includes a number of classical optimization problems such as *knapsack* and *partition*. Blazewicz et al. [3] have shown that RCPSP is a generalization of the classical *job shop problem*, thus solving is $\mathcal{NP}$-hard in the strong sense.

### 1.3.1 Example

| job | $p_i$ | $D_1$ | $D_2$ | $D_3$ |
|---|---|---|---|---|
| | | 5 | 2 | 3 |
| job | $p_i$ | $d_{i1}$ | $d_{i2}$ | $d_{i3}$ |
| 1 | 3 | 2 | 2 | - |
| 2 | 4 | 2 | - | 1 |
| 3 | 3 | - | - | 2 |
| 4 | 1 | 3 | 2 | 2 |
| 5 | 2 | 1 | 1 | 1 |
| 6 | 2 | 1 | - | 2 |
| 7 | 3 | 2 | 1 | - |
| 8 | 1 | 2 | 1 | 2 |



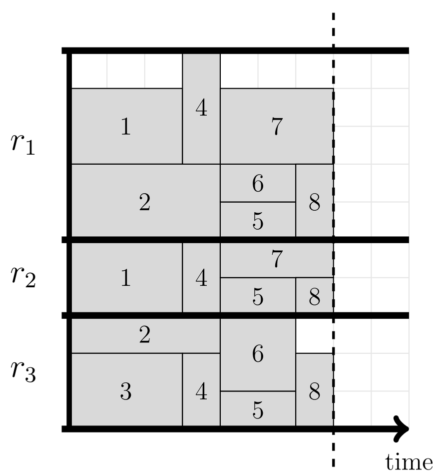




**Figure 1.1:** Above: Instance of the RCPSP with jobs $I = \{i_1, ..., i_8\}$, resources $R = \{r_1, r_2, r_3\}$ and the underlying precedence graph. Below: feasible schedule of makespan 7 in Gantt-chart representation.

# Chapter 2

# MIP Models for RCPSP

The literature proposes different MIP models for the RCPSP. In this thesis we distinguish between two types: *time-indexed* and *compact* models. Time-indexed models subdivide the scheduling horizon into unit size discrete time intervals. Each job gets assigned a feasible starting time. Therefore, the model size is depends on the required time horizon. For large scheduling horizons and fine discretizations the models size becomes intractable. In this case, we prefer compact models whose size is polynomial in the number of jobs. First, we state the common time-indexed model DT of Pritsker et al. [12] and its extension DDT of Christofides et al. [7]. Subsequently, we introduce two compact models OOE and SEE of Koné et al. [8] for which we propose stronger inequalities. Furthermore, we add two new compact models HPF and DP which are based on common modeling ideas. Structurally however, the models reveal significant differences. Chapter 3 deals with a detailed polyhedral analysis.

## 2.1 Time-Indexed Model (DDT)

The most common time-discrete model is due to Pritsker et al. [12]. Let $\mathcal{T} = \{1, ..., T\}$ be the discrete time horizon where $T$ denotes an upper bound on the minimum makespan. For the makespan we introduce a dummy job $n+1$ with $p_{n+1} = 0$ and extend the set of precedence relations by $E' = E \cup \{(i, n+1) \mid i \in I\}$. Let $x_{it} \in \{0, 1\}$ be binary variables that are equal to one, if job $i \in I$ starts at time $t \in \mathcal{T}$ and zero otherwise. To abuse notation in the model, let $x_{it} = 0$ for all $i \in I$ and $t \leq 0$.

The *discrete-time model* (DT) states as follows:

$$\min \sum_{t \in \mathcal{T}} (t-1) \cdot x_{n+1t}$$

$$\sum_{t \in \mathcal{T}} x_{it} = 1 \qquad \forall i \in I \cup \{n+1\} \tag{2.1}$$

$$\sum_{i \in I} \sum_{t'=t-p_i+1}^{t} d_{ir} x_{it'} \leq D_r \qquad \forall t \in \mathcal{T}, r \in R \tag{2.2}$$

$$\sum_{t \in \mathcal{T}} t \cdot x_{jt} - \sum_{t \in \mathcal{T}} t \cdot x_{it} \geq p_i \qquad \forall (i,j) \in E' \tag{2.3}$$

$$x_{it} \in \{0,1\} \qquad \forall i \in I \cup \{n+1\}, t \in \mathcal{T}$$

The objective minimizes the makespan that is attained at the starting time of job $n+1$. All jobs including $n+1$ must start at a time slot $t \in \mathcal{T}$ (2.1). Inequalities (2.2) ensure that the resource demands are within the capacity limits for each resource $r \in R$ at any time slot $t \in \mathcal{T}$. In addition, job $i$ must end before job $j$ starts for all $(i,j) \in E'$ (2.3).

Christofides et al. [7] replace inequalities (2.3) by the *disaggregated precedence constraints*:

$$\sum_{t'=1}^{t+p_i-1} x_{jt'} + \sum_{t'=t}^{T} x_{it'} \leq 1 \qquad \forall (i,j) \in E', t \in \mathcal{T} \tag{2.4}$$

which yields the *disaggregated discrete-time model* (DDT). Since (2.3) can be obtained from (2.1) and (2.4), DDT provides a stronger LP-relaxation than DT. In a few cases, DT is preferred by computational reasons, see [7].

Next, we inspect strengths and weaknesses of the LP-relaxation of DDT. Consider $B = \max_{r \in R} \sum_{i \in I} \frac{d_{ir} p_i}{D_r}$ as a natural lower bound to an instance of the RCPSP.

**Proposition 2.1.** *Assume an instance of RCPSP with $E = \emptyset$ and define $B = \max_{r \in R} \sum_{i \in I} \frac{d_{ir} p_i}{D_r}$. Furthermore, let $p_i \leq \frac{B-2}{2}$ for all $i \in I$. For the optimal value of the LP-relaxation of DDT it holds $OPT_{LP}(DDT) \leq B$.*

*Proof.* We construct a feasible LP solution. Set $x_{it} = \frac{1}{B}$ for all $i \in I$ and $t \in \{1, ..., \lfloor B \rfloor\}$. Furthermore, set $x_{it} = \frac{B - \lfloor B \rfloor}{B}$ for all $i \in I$ and $t = \lfloor B \rfloor + 1$.

First, we show that the inequalities of DDT are satisfied for all $i \in I$. For (2.1) we obtain

$$\sum_{t=1}^{T} x_{it} = \sum_{t=1}^{\lfloor B \rfloor + 1} x_{it} = \frac{\lfloor B \rfloor}{B} + \frac{B - \lfloor B \rfloor}{B} = 1 \qquad \forall i \in I \cup \{n+1\}$$

which is valid. The resource constraints (2.2) are implied by

$$\sum_{i \in I} \sum_{t'=t-p_i+1}^{t} d_{ir} x_{it'} \leq \sum_{i \in I} \sum_{t'=t-p_i+1}^{t} \frac{d_{ir}}{B} = \sum_{i \in I} \frac{d_{ir} p_i}{B} \leq D_r \quad \forall r \in R, t \in \mathcal{T}$$

that is also valid. Since $E = \emptyset$ we have $E' = \{(i, n+1) \mid i \in I\}$. Therefore, each precedence constraint satisfies

$$\sum_{t'=1}^{t+p_i-1} x_{n+1t'} \leq 1 - \sum_{t'=t}^{T} x_{it'} \overset{(2.1)}{=} \sum_{t'=1}^{t-1} x_{it'} \qquad \forall (i, n+1) \in E', t \in \mathcal{T}$$

which by substitution is equivalent to

$$\sum_{t'=1}^{t} x_{n+1t'} \leq \sum_{t'=1}^{t-p_i} x_{it'} = \max\left\{0, \min\left\{1, \frac{t-p_i}{B}\right\}\right\} \quad \forall (i, n+1) \in E', t \in \mathcal{T}$$

In an optimal LP solution the values $x_{n+1t}$ are maximal with respect to the earliest possible time slots $t \in \mathcal{T}$. Let $p^* = \max_{i \in I}(p_i)$, by the above formula we deduce $x_{n+1t} = \frac{1}{B}$ for all $t \in \{p^* + 1, ..., p^* + \lfloor B \rfloor\}$ and $x_{n+1t} = \frac{B - \lfloor B \rfloor}{B}$ for $t = p^* + \lfloor B \rfloor + 1$.

Evaluating the objective yields:

$$
\begin{aligned}
\sum_{t=1}^{T}(t-1)\cdot x_{n+1t} &= \sum_{t=p^*+1}^{\lfloor B\rfloor+p^*}\frac{(t-1)}{B} + \frac{(p^*+\lfloor B\rfloor)\cdot(B-\lfloor B\rfloor)}{B} \\
&= \sum_{t=1}^{\lfloor B\rfloor}\frac{(t+p^*-1)}{B} + \frac{(p^*+\lfloor B\rfloor)\cdot(B-\lfloor B\rfloor)}{B} \\
&= \frac{\lfloor B\rfloor\cdot(\lfloor B\rfloor+1)}{2B} + \frac{\lfloor B\rfloor\cdot(p^*-1)}{B} + \frac{(p^*+\lfloor B\rfloor)\cdot(B-\lfloor B\rfloor)}{B} \\
&= \frac{\lfloor B\rfloor\cdot(\lfloor B\rfloor+1+2p^*-2+2B-2\lfloor B\rfloor)}{2B} + \frac{p^*\cdot(B-\lfloor B\rfloor)}{B} \\
&= \frac{\lfloor B\rfloor\cdot(2p^*-1+2B-\lfloor B\rfloor)}{2B} + \frac{p^*\cdot(B-\lfloor B\rfloor)}{B} \\
&\leq \frac{\lfloor B\rfloor\cdot(2p^*+B)}{2B} + \frac{p^*\cdot(B-\lfloor B\rfloor)}{B} \\
&\leq p^*+\frac{B}{2}+\frac{p^*\cdot(B-\lfloor B\rfloor)}{B} = \frac{B}{2}+p^*\cdot\left(2-\frac{\lfloor B\rfloor}{B}\right) \\
&\leq \frac{B}{2}+B-2-\frac{(B-2)\cdot\lfloor B\rfloor}{2B} = B+\underbrace{\frac{B}{2}-\frac{\lfloor B\rfloor}{2}+\frac{\lfloor B\rfloor}{B}}_{\leq 2}-2 \leq B
\end{aligned}
$$

Consequently, we have $OPT_{LP}(DDT) \leq B$. □

**Proposition 2.2.** *Let $L$ denote the length of the longest path in the precedence graph $G=(I,E)$. For the optimal value of the LP-relaxation of DDT it holds $L \leq OPT_{LP}(DDT)$.*

*Proof.* Relaxing the resource constraints (2.2) yields a constraint matrix with integral solutions, see [19, 20]. Since the precedence constraints (2.4) hold in the relaxation, the integral solution has a lower bound at least $L$. The optimal LP value is larger than the relaxed value, therefore $L \leq OPT_{LP}(DDT)$. $\square$

Propositions (2.1) and (2.2) give a rough idea of the strength and weakness of the DDT model. On instances where the longest precedence path is dominating, the LP bound gets strong. Contrary, the bound appears to be weak for cumulative instances that are independent of the precedence graph. A study of the LP-relaxation based on real data is given in Chapter 6.

## 2.2 On/Off Event-based Model (OOE)

The following compact model was introduced by Koné et al. [8]. In contrast to DDT, the scheduling horizon is subdivided into intervals of variable length. The start of an interval marks an *event*, that is a new job starts at the beginning of the interval. Each job starts at a unique event and the starting time of the job equals the starting time of the assigned event. Since events correspond to positions in which jobs are added to the processing sequence, we denote them by *positions* or by their associated *interval* throughout the thesis. Let $K = \{1, ..., n\}$ be the set of starting positions. Note that each job may start at its own position, for example in the one machine case. Furthermore, let $K' = \{2, ..., n + 1\}$ be the set of completing positions and $A = \{(k, l) \in K \times K' \mid k < l\}$ be the set of possible start/end positions for each job. A job $i \in I$ is *active* at position $k \in K$, if it is processed right after the starting time of position $k$. A Job $i \in I$ starts at $k \in K$, if and only if it is active at position $k$ but not at position $k - 1$. Similarly, a job $i \in I$ completes at position $k \in K'$ if and only if it is active at position $k - 1$ but not at position $k$. Let $u_{ik} \in \{0, 1\}$ be the binary variable that equals one, if job $i \in I$ is active at position $k \in K$ and zero otherwise. In addition, let $s_k \geq 0$ be the starting time of position $k \in K$. The makespan is denoted by $s_{n+1}$. To abuse notation in the model, consider $u_{i0} = 0$ and $u_{in+1} = 0$ for all $i \in I$.

The *on/off event-based model* of Koné et al. [8] is defined as follows:

$$\min \; s_{n+1}$$

$$\sum_{k\in K} u_{ik} \geq 1 \qquad \forall i \in I \tag{2.5}$$

$$\sum_{i\in I} d_{ir} u_{ik} \leq D_r \qquad \forall k \in K, r \in R \tag{2.6}$$

$$s_k - s_{k+1} \leq 0 \qquad \forall k \in K \tag{2.7}$$

$$s_k + p_i(u_{ik} - u_{ik-1} + u_{il-1} - u_{il} - 1) \leq s_l \qquad \forall i \in I, (k,l) \in A \tag{2.8}$$

$$\sum_{k'=1}^{k} u_{ik'} - k \cdot (1 + u_{ik} - u_{ik+1}) \leq 0 \qquad \forall i \in I, 1 < k < n \tag{2.9}$$

$$\sum_{k'=k}^{n} u_{ik'} - (n-k+1) \cdot (1 + u_{ik} - u_{ik-1}) \leq 0 \qquad \forall i \in I, 1 < k < n \tag{2.10}$$

$$\sum_{k'=1}^{k} u_{jk'} - k \cdot (1 - u_{ik}) \leq 0 \qquad \forall (i,j) \in E, k \in K \tag{2.11}$$

$$u_{ik} \in \{0,1\} \qquad \forall i \in I, k \in K$$

$$s_k \geq 0 \qquad \forall k \in K \cup \{n+1\}.$$

Inequality (2.5) requires each job to be active in at least one position interval. The consumption of resource $r \in R$ of all active jobs at position $k$ must not exceed the capacity $D_r$ (2.6). In addition, subsequent positions have increasing starting times (2.7).

If a jobs $i \in I$ starts at position $k \in K$ and completes at position $l \in K'$ then it must hold $s_k + p_i \leq s_l$ which is expressed by (2.8). Constraints (2.9) and (2.10) are the *contiguity constraints* that require the sequence $(u_{i1}, ..., u_{in})$ to be unimodal for each $i \in I$ since the jobs are scheduled non-preemptively. In particular, if some job starts at position $k \in K$ then it cannot be active at a position $k' < k$ (2.10). Analogously, if some job completes at position $k \in K'$ then it cannot be active at a position $k' > k$ (2.9). The precedence constraints (2.11) are modeled similarly: if job $i$ is active at position $k \in K$ then job $j$ cannot be active at positions $k' \in \{1, ..., k\}$ for all $(i,j) \in E$. Note that constraints (2.9)-(2.10) use big-M constants that linearly depend on $n$.

In the following we present stronger inequalities for the OOE model.

**Proposition 2.3.** *The inequalities*

$$s_k - s_l + p_i(u_{ik} - u_{ik-1} - u_{il}) \leq 0 \qquad \forall i \in I, (k,l) \in A \tag{2.12}$$

*are valid for OOE and they dominate inequalities* (2.8)*.*

*Proof.* The inequalities state that if job $i \in I$ starts at position $k \in K$ but it is not active at position $l \in K'$ then $s_k + p_i \leq s_l$. Since in this case, job $i$ has to complete before $l$ the inequality is valid. Obviously, (2.12) is stronger because adding $p_i(u_{il-1} - 1) \leq 0$ yields (2.8). □

**Proposition 2.4.** *The inequalities*

$$u_{ih} - u_{ik} + u_{il} \leq 1 \qquad \forall i \in I, (h,k,l) \in K^3 : h < k < l \tag{2.13}$$

*are valid for OOE and they dominate inequalities* (2.9) *and* (2.10)*.*

*Proof.* The constraint indicates that whenever a job is active at positions $h, l \in K$ with $h < k < l$, then it must also be active at position $k$. Certainly, the inequalities are valid for OOE. Define $k' \in K$ with $1 < k' < n$. Summing up (2.13) over all $h \in \{1, ..., k'-1\}$ with $k = k'$ and $l = k'+1$, and adding the trivial inequality $u_{ik'+1} - 1 \leq 0$ yields inequality (2.9):

$$\begin{aligned}
&\sum_{h=1}^{k'-1} (u_{ih} - u_{ik'} + u_{ik'+1} - 1) + u_{ik'+1} - 1 \leq 0 \\
\iff &\sum_{h=1}^{k'-1} u_{ih} + (k'-1) \cdot (u_{ik'+1} - u_{ik'} - 1) + u_{ik'+1} - 1 \leq 0 \\
\iff &\sum_{h=1}^{k'} u_{ih} - k' \cdot (1 + u_{ik'} - u_{ik'+1}) \leq 0 \qquad \forall i \in I, 1 < k' < n
\end{aligned}$$

Equally, fix $h = k'-1$ and $k = k'$. Summing up (2.13) for all $l \in \{k'+1, ..., n\}$

and adding $u_{ik'-1} - 1 \leq 0$ yields (2.10):

$$\begin{aligned}
&\sum_{l=k'+1}^{n} (u_{ik'-1} - u_{ik'} + u_{il} - 1) + u_{ik'-1} - 1 \leq 0 \\
\iff &\sum_{l=k'+1}^{n} u_{il} + (n-k') \cdot (u_{ik'-1} - u_{ik'} - 1) + u_{ik'-1} - 1 \leq 0 \\
\iff &\sum_{l=k'+1}^{n} u_{il} - (n-k') \cdot (u_{ik'} - u_{ik'-1} + 1) + u_{ik'-1} - 1 + u_{ik'} - u_{ik'} \leq 0 \\
\iff &\sum_{l=k'}^{n} u_{il} - (n-k'+1) \cdot (1 + u_{ik'} - u_{ik'-1}) \leq 0 \qquad \forall i \in I, 1 < k' < n.
\end{aligned}$$

Consequently, (2.13) dominate (2.9) and (2.10). □

**Proposition 2.5.** *The stable set inequalities*

$$u_{jk} + u_{il} \leq 1 \qquad \forall (i,j) \in E, (k,l) \in K^2 : k \leq l \tag{2.14}$$

*are valid for OOE and they dominate inequalities* (2.11)*.*

*Proof.* For $(i,j) \in E$ inequalities (2.14) forbid that job $j$ is active before job $i$, hence it is valid for OOE. Summing up (2.14) for all $k \in \{1, ..., l\}$ yields inequality (2.11):

$$\sum_{k=1}^{l} (u_{jk} + u_{il} - 1) \leq 0 \iff \sum_{k=1}^{l} u_{jk} - l \cdot (1 - u_{il}) \leq 0 \quad \forall (i,j) \in E, l \in K$$

Consequently, (2.11) is dominated by (2.14). □

Note that inequalities (2.14) constitute that feasible solutions correspond to stable sets in the graph consisting of vertices $v_{ik}$ and edges $(v_{il}, v_{jk})$ with $(i,j) \in E$, $k \leq l$.

**Proposition 2.6.** *In the stated OOE model, the contiguity constraints* (2.9)*,* (2.10) *and* (2.13) *can be omitted.*

*Proof.* Assume that neither constraints (2.9), (2.10) or (2.13) are added to the model. Suppose an optimal solution with $u_{ih} = u_{il} = 1$ and $u_{ik} = 0$ for some $i \in I$ and $(h,k,l) \in K^3$ with $h < k < l$. This indicates a job is active for more than one sequence. Since such a solution strengthens any other constraint of the OOE model, the objective value can only get larger. Therefore, an optimal solution either satisfies the contiguity constraints or it does not affect the optimal solution. In the latter case, the active sequence can be chosen arbitrarily. □

**Proposition 2.7.** *Consider the OOE model including the stronger inequalities* (2.12), (2.13) *and* (2.14)*. For the optimal value of the LP-relaxation of OOE it holds* $OPT_{LP}(OOE) = 0$*.*

*Proof.* We construct an optimal LP solution. Let $x_{ik} = \frac{1}{n}$ for all $i \in I, k \in K$ and $s_k = 0$ for all $k \in K \cup \{n+1\}$. From (2.5) we get

$$\sum_{k \in K} u_{ik} = \sum_{k \in K} \frac{1}{n} = 1 \geq 1 \qquad \forall i \in I$$

Inequalities (2.6) satisfy

$$\sum_{i \in I} d_{ir} u_{ik} = \frac{1}{n} \sum_{i \in I} d_{ir} \leq \max_{i \in I}(d_{ir}) \leq D_r \qquad \forall k \in K, r \in R$$

For (2.12) we get

$$s_k - s_l + p_i(u_{ik} - u_{ik-1} - u_{il}) = -\frac{p_i}{n} \leq 0 \qquad \forall i \in I, (k,l) \in A$$

Constraints (2.13) and (2.14) are trivially satisfied. Consequently, all inequalities hold with LP value equal zero. □

Propositions 2.3 to 2.7 reveal structural weaknesses of the OOE model. First, modeling non-preemptivity requires $\mathcal{O}(n^3)$ constraints, since we only decide if a job is active at some position. Second, the start and completion of a job $i \in I$ at some position $k \in K$ has to be build via clauses that involve more than one variable. In particular, for the starting time constraints this leads to weak inequalities and therefore poor LP bounds.

The SEE model of the next section tackles this problem by expanding the variable space.

## 2.3 Start/End Event-based Model (SEE)

As in the previous model, the SEE model of Koné et al. [8] considers positions $K = \{1, ..., n\}$. We use the same prerequisites as for the OOE model. The idea of this model is to determine a starting and completing position for each job. Therefore, the binary variable $x_{ik} \in \{0, 1\}$ equals one, if job $i \in I$ starts at position $k \in K$. Similarly, the binary variable $y_{il} \in \{0, 1\}$ equals one, if $i \in I$ completes at position $l \in K'$. The starting time of position $k \in K$ is denoted by $s_k \geq 0$ and $s_{n+1}$ is the makespan. In contrast to originally stated in [8], we substitute out variables which yields an equivalent model. The *start/end event-based model* reads as follows:

$$
\begin{aligned}
\min\ & s_{n+1} \\
\sum_{k \in K} x_{ik} &= 1 && \forall i \in I && (2.15)\\
\sum_{k \in K'} y_{ik} &= 1 && \forall i \in I && (2.16)\\
s_k - s_l + p_i(x_{ik} + y_{il} - 1) &\leq 0 && \forall i \in I, (k, l) \in A && (2.17)\\
s_k - s_{k+1} &\leq 0 && \forall k \in K && (2.18)\\
\sum_{i \in I} d_{ir} \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \right) &\leq D_r && \forall k \in K, r \in R && (2.19)\\
\sum_{k'=2}^{k} y_{ik'} + \sum_{k'=k}^{n} x_{ik'} &\leq 1 && \forall i \in I, k \in K && (2.20)\\
\sum_{k'=1}^{k} x_{jk'} + \sum_{k'=k+1}^{n+1} y_{ik'} &\leq 1 && \forall (i, j) \in E, k \in K && (2.21)\\
x_{ik} &\in \{0, 1\} && \forall i \in I, k \in K \\
y_{ik} &\in \{0, 1\} && \forall i \in I, k \in K' \\
s_k &\geq 0 && \forall k \in K \cup \{n + 1\}
\end{aligned}
$$

Constraints (2.15) and (2.16) indicate that each job starts and ends at a unique position. Certainly, each job has to start before it ends (2.20). If job $i \in I$ starts at position $k \in K$ and finishes at $l \in K'$, then the condition $s_k + p_i \leq s_l$ must hold (2.17). In addition, consecutive positions must have increasing starting times (2.18). Constraints (2.19) require that the total resource consumption of a resource $r \in R$ at position $k \in K$ does not exceed the capacity $D_r$. The disaggregated precedence constraints (2.21) are adopted from DDT which imply for all $(i, j) \in E$ that job $i$ cannot complete after $j$ has started.

**Proposition 2.8.** *The inequalities*

$$s_k - s_l + p_i \left( \sum_{k'=k}^{n} x_{ik'} + \sum_{k'=2}^{l} y_{ik'} - 1 \right) \leq 0 \qquad \forall i \in I, (k, l) \in A \tag{2.22}$$

*are valid for SEE and they dominate inequalities (2.17).*

*Proof.* The constraint describes that if job $i \in I$ does not start before $k \in K$ and does not end after $l \in K'$, then it holds $s_k + p_i \leq s_l$. Since in this case, job $i$ must entirely be processed between $k$ and $l$ the inequality is valid. Obviously, (2.22) is stronger than (2.17). □

**Proposition 2.9.** *Assume the SEE model with inequalities* (2.22)*. The optimal value of the LP-relaxation satisfies* $\max_{i \in I}(p_i) \leq OPT_{LP}(SEE)$.

*Proof.* Choose $j \in I$ with $p_j = \max_{i \in I}(p_i)$. Then consider (2.22) for $j \in I$ and $(1, n+1) \in A$. By (2.15) and (2.16) we get $p_j \leq s_{n+1}$. □

The bound of the strengthened SEE is stronger than OOE. However, the bound is still weak for practical purposes. The computational study in [8] showed that OOE performs better than SEE on a various set of problem instances. One could argue that the smaller number of variables makes OOE more convenient to solve on small instances. However, we show the opposite throughout the thesis, namely the SEE model yields a stronger LP-relaxation and better computational results. This also stems from a non-singular transformation of SEE that is presented in the next section.

### 2.3.1 Revised SEE Model

Modern MIP solvers work more efficiently when the constraint matrix is sparse. Some models can be converted into a sparse model via non-singular transformations, see Bianco et al. [21] and Artigues [10] for DDT. In particular, non-singular transformations yield equivalent models. In most cases they simply give a new interpretation for the existing variables.

We apply such a transformation to SEE. To simplify notation we use the same variable names. Consider the transformation given by $x_{ik} \leftarrow \sum_{k'=1}^{k} x_{ik'}$ and $y_{il} \leftarrow \sum_{l'=2}^{k} y_{il'}$ for all $i \in I$, $(k,l) \in A$. That is, $x_{ik}$ is one if job $i \in I$ starts at some position $k' \leq k$ and $y_{il}$ is one if job $i \in I$ completes at some position $l' \leq l$. We apply this conversion directly to the strengthened SEE model. The *revised SEE model* states as follows:

$$
\begin{aligned}
\min s_{n+1} & & & \\
x_{in} &= 1 & \forall i \in I & \quad (2.23)\\
y_{in+1} &= 1 & \forall i \in I & \quad (2.24)\\
s_k - s_l + p_i(y_{il} - x_{ik-1}) &\leq 0 & \forall i \in I, (k,l) \in A & \quad (2.25)\\
s_k - s_{k+1} &\leq 0 & \forall k \in K & \quad (2.26)\\
\sum_{i \in I} d_{ir}(x_{ik} - y_{ik}) &\leq D_r & \forall k \in K, r \in R & \quad (2.27)\\
y_{ik+1} &\leq x_{ik} & \forall i \in I, k \in K & \quad (2.28)\\
x_{jk} &\leq y_{ik} & \forall (i,j) \in E, k \in K : 1 < k & \quad (2.29)\\
x_{ik} &\leq x_{ik+1} & \forall i \in I, k \in K : k < n & \quad (2.30)\\
y_{ik} &\leq y_{ik+1} & \forall i \in I, k \in K' : k < n+1 & \quad (2.31)\\
x_{ik} &\in \{0,1\} & \forall i \in I, k \in K &\\
y_{ik} &\in \{0,1\} & \forall i \in I, k \in K' &
\end{aligned}
$$

Inequalities (2.23) to (2.31) correspond to the transformed inequalities (2.15) to (2.22). Constraints (2.30) and (2.31) emerge from $x_{ik} \geq 0$ and $y_{ik} \geq 0$ in the original SEE model. We directly observe that the constraint matrix is more sparse than in the first version, since many rows involve only one or two variables. In particular, it reveals further structural properties.

**Proposition 2.10.** *Consider the MIP obtained from inequalities* (2.28) *to* (2.31)*. The corresponding constraint matrix it totally unimodular.*

*Proof.* Each row contains exactly two entries with coefficients 1 and -1, hence the constraint matrix is a network matrix. Network matrices are known to be totally unimodular. □

Proposition 2.10 gives rise to construct the associated network graph. It shows that the revised SEE model includes structures that are well-suited for constraint propagation. But also a Lagrangean Relaxation approach similar

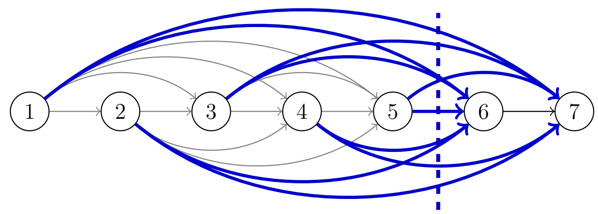



**Figure 2.1:** Sequence graph $G = (K, A)$ with sequence cut $A_5$

to Möhring et al. [18] for DDT would be interesting. However, this will not be part of the thesis and we leave this open for future research. Our computational results were made using the revised SEE model, since the performance impact on the MIP solver was remarkable.

## 2.4 Disaggregated Position Model (DP)

In the following we present a novel compact MIP formulation for the RCPSP, which is denoted the *disaggregated position model* (DP). The basic idea is to incorporate multiple decisions of the previous models into one decision. More precisely, instead of making two decisions at which position a jobs starts and ends, we couple this by one decision. Assume the notation of the previous models. Additionally, define the *position graph* by $G_K = (K, A)$ and the *sequence cut* $A_h = \{(k, l) \in A \mid k \leq h < l\} \subseteq A$ at position $h \in K$. The set $A_h$ can be seen as the network cut in $G_K$ that separates the nodes $\{1, ..., h\}$ and $\{h+1, ..., n+1\}$, see Figure 2.1. We introduce binary variables $z_{ikl} \in \{0, 1\}$ that are one, if job $i \in I$ starts at position $k$ and ends at position $l$ with $(k, l) \in A$. As before, $s_k \geq 0$ denotes the starting time of position $k \in K$ and $s_{n+1}$ is the makespan.

The *disaggregated position model* states as follows:

$$\min \quad s_{n+1}$$

$$\sum_{(k,l)\in A} z_{ikl} = 1 \qquad \forall i \in I \tag{2.32}$$

$$s_k + p_i \sum_{k'=k}^{l-1} \sum_{l'=k'+1}^{l} z_{ik'l'} \leq s_l \qquad \forall i \in I, (k,l) \in A \tag{2.33}$$

$$\sum_{i\in I} \sum_{(k,l)\in A_h} d_{ir} z_{ikl} \leq D_r \qquad \forall h \in K, r \in R \tag{2.34}$$

$$\sum_{k=1}^{h} \sum_{l=k+1}^{n+1} z_{jkl} + \sum_{l=h+1}^{n+1} \sum_{k=1}^{l-1} z_{ikl} \leq 1 \qquad \forall (i,j) \in E, h \in K \tag{2.35}$$

$$z_{ikl} \in \{0,1\} \qquad \forall i \in I, (k,l) \in A$$

$$s_k \geq 0 \qquad \forall k \in K \cup \{n+1\}$$

By inequalities (2.32) each job is assigned to an arc $(k,l) \in A$, that is each job receives a starting and completing position. In (2.33) it holds $s_k + p_i \leq s_l$ if and only if job $i \in I$ is scheduled between positions $k \in K$ and $l \in K'$. Inequalities (2.34) represent the resource constraints. By construction, a job $i \in I$ is active at position $h \in K$ if and only if job $i$ is assigned to an arc $(k,l) \in A_h$. Consequently, the total resource consumption at position $h \in K$ is the total resource demand of jobs on the sequence cut $A_h$. Constraints (2.35) are the usual disaggregated precedence constraints translated to this setting.

Note that in contrast to the previous compact models, DP does not depend on big-M constants to restrict the $s_k$ variables. In exchange, the variables space is increased by an additional factor of $n$ which gives $\mathcal{O}(n^3)$ variables in general.

## 2.5 Hybrid Position Flow Model (HPF)

In this section we introduce a further new MIP model that is based on *resource flows*. We use the same notation of the previous models. In contrast to the other compact models, HPF requires that exactly one job starts at each position and vice versa. According to this assignment a resource flow is induced in $G_K$, as defined in the previous section. Let $x_{ik} \in \{0,1\}$ be a binary variable that is one if job $i \in I$ starts at position $k \in K$. Furthermore, let $w_{kl} \in \{0,1\}$ be a binary variable that is one, if the unique job at position $k$ completes at $l \in K'$. Moreover, let $f_{klr} \geq 0$ be the resource demand of resource $r \in R$ that is allocated during positions $k \in K$ and $l \in K'$ and let $\bar{d}_r = \max_{i \in I} d_{ir}$. As usual, $s_k \geq 0$ is the starting time of position $k \in K$ and $s_{n+1}$ is the makespan. The *hybrid position flow model* states as follows:

$$\min \; s_{n+1}$$

$$\sum_{k \in K} x_{ik} = 1 \qquad \forall i \in I \tag{2.36}$$

$$\sum_{i \in I} x_{ik} = 1 \qquad \forall k \in K \tag{2.37}$$

$$\sum_{l'=k+1}^{n+1} w_{kl'} = 1 \qquad \forall k \in K \tag{2.38}$$

$$s_k - s_l + p_i \left( x_{ik} + \sum_{l'=k+1}^{l} w_{kl'} - 1 \right) \leq 0 \qquad \forall i \in I, (k,l) \in A \tag{2.39}$$

$$s_k - s_{k+1} \leq 0 \qquad \forall k \in K \tag{2.40}$$

$$\sum_{i \in I} d_{ir} x_{ik} - \sum_{l=k+1}^{n+1} f_{klr} = 0 \qquad \forall k \in K, r \in R \tag{2.41}$$

$$\sum_{(k,l) \in A_h} f_{klr} \leq D_r \qquad \forall h \in K, r \in R \tag{2.42}$$

$$f_{klr} - \bar{d}_r w_{kl} \leq 0 \qquad \forall (k,l) \in A, r \in R \tag{2.43}$$

$$x_{ik} - \sum_{l'=k+1}^{l-1} w_{kl'} + \sum_{h=l}^{n} x_{ih} + \sum_{h=1}^{l-1} x_{jh} \leq 1 \qquad \forall (i,j) \in E, (k,l) \in A \tag{2.44}$$

$$x_{ik} \in \{0,1\} \qquad \forall i \in I, k \in K$$

$$w_{kl} \in \{0,1\} \qquad \forall (k,l) \in A$$

$$f_{klr} \geq 0 \qquad \forall (k,l) \in A, r \in R$$

$$s_k \geq 0 \qquad \forall k \in K \cup \{n+1\}$$

Constraints (2.36) and (2.37) model the assignment between $I$ and $K$. In (2.38) any job that is assigned to position $k \in K$ must complete until position $n+1$. If a job $i \in I$ starts at position $k \in K$ and if the job assigned to position $k$ completes at position $l \in K'$ then it holds $s_k + p_i \leq s_l$ (2.39). Normally, the starting times must be increasing (2.40). Equalities (2.41) indicate that the resource demand of the job assigned to position $k \in K$ must be released until position $n+1$. Inequalities (2.43) limit the total active resource flow by the available capacity on every sequence cut. Furthermore, if the job at position $k \in K$ does not complete at position $l \in K'$ then the corresponding resource flow cannot be released at position $l$. Constraints (2.44) are the precedence constraints. Given jobs $(i,j) \in E$, if job $i$ starts at position $k \in K$ and does not end until $l-1 \in K$, then job $j$ must not start at some position $h \leq l-1$. This also holds, if job $i$ starts after position $l-1$.

Similar to SEE, the HPF model admits a non-singular transformation. This yields an equivalent model but reduces the number of non-zero entries in the constraint matrix. With respect to the original HPF model we applied the transformations $w_{kl} \leftarrow \sum_{l'=k+1}^{l} w_{kl'}$ for all $(k,l) \in A$ and $f_{klr} \leftarrow \sum_{l'=k+1}^{l} f_{kl'r}$ for all $(k,l) \in A, r \in R$. The *revised HPF model* is presented in the appendix.

## 2.6 Preprocessing Feasible Job Positions

Due to the precedence constraints, we can reduce the set of feasible starting and completing positions for each job, see Koné et al. [8]. We associate a boolean to the expression $i \rightsquigarrow j$ that is true, if there is a path from $i$ to $j$ in the precedence graph $G = (I, E)$. Then define $P_i = \{j \in I \mid j \rightsquigarrow i\}$ as the set of predecessors and $S_i = \{j \in I \mid i \rightsquigarrow j\}$ as the set of successors of each job $i \in I$. Certainly, all predecessors of job $i \in I$ must end at a position smaller or equal to the starting position of job $i$. Analogously, all successors of job $i \in I$ must start at a position greater or equal to the completing position of job $i$. Therefore, define

$$K_i = \{k \in K \mid |P_i| + 1 \leq k \leq n - |S_i|\} \qquad \text{and}$$
$$K'_i = \{k \in K \mid |P_i| + 2 \leq k \leq n - |S_i| + 1\}$$

as the set of possible starting and completing positions of job $i \in I$. The sets $P_i$ and $S_i$ can be computed, for example, by the *Floyd-Warshall Algorithm* in $\mathcal{O}(n^3)$. Finally, in each MIP model the corresponding variables are restricted to $K_i$ and $K'_i$ respectively.

# Chapter 3

# Polyhedral Study

The last chapter introduced the time-indexed model DDT and two compact models OOE and SEE from the literature. In addition, we presented two new compact models DP and HPF. This section investigates polyhedral aspects of these models with particular regard to the induced LP-relaxations. In fact, we show that OOE, SEE, DP and HPF, DP are equivalent up to linear transformations. We further explore the LP-relaxations under these transformations. Moreover, we introduce an indicator that measures the tightness of the transformation between different classes of inequalities. In addition, we examine the underlying transformation matrix and derive some properties.

## 3.1 Characterization of Integer Solutions

In the following we present affine transformations between the models OOE, SEE, HPF, DP and give a combinatorial interpretation. Furthermore, we show that the transformations yield bijections between the integer solutions of the models. Therefore, recall the variables of OOE, SEE, HPF and DP. Consider the following linear transformation between the binary variables of the models SEE, HPF and DP:

$$x_{ik} = \sum_{l=k+1}^{n+1} z_{ikl} \qquad \forall i \in I, k \in K \tag{3.1}$$

$$y_{il} = \sum_{k=1}^{l-1} z_{ikl} \qquad \forall i \in I, l \in K' \tag{3.2}$$

$$w_{kl} = \sum_{i \in I} z_{ikl} \qquad \forall (k,l) \in A. \tag{3.3}$$

Note that the transformation preserves integral solutions. First, we deduce some properties of the transformation.

**Lemma 3.1.** *The transformation matrix of* (3.1) *and* (3.2) *is totally unimodular.*

*Proof.* Let $B \in \{-1,0,1\}^{2n^2 \times n \cdot |A|}$ be the associated matrix with $Bz = \binom{x}{y}$. Then $B$ has contains exactly two ones in each column, therefore $B$ is a graph incidence matrix. Such matrices are known to be totally unimodular. □

**Lemma 3.2.** *The transformation matrix of* (3.1) *and* (3.3) *is totally unimodular.*

*Proof.* Let $B \in \{-1,0,1\}^{n^2+|A| \times n \cdot |A|}$ be the associated matrix with $Bz = \binom{x}{w}$. Then $B$ contains exactly two ones in each column, so $B$ is a graph incidence matrix. These are known to be totally unimodular. □

Lemmas 3.1 and 3.2 reveal a certain flow structure of the transformation matrix. In particular, recall the definition of a 3-way transportation problem, see [64]. Another implied characterization is given by the following.

**Observation 3.3.** *The transformation matrix of equalities* (3.1)-(3.3) *corresponds to a restricted 3-way transportation problem.*

In this context, 'restricted' means the flow on edges $(k,l)$ with $l \leq k$ is restricted to zero. This interpretation reveals a natural characterization of feasible integer solutions. Therefore, we use the strong connection of 3-way transportation flows to *multicommodity flows*, see Schrijver [4, Chapter 70]. Let $G_M = (V_M, A_M)$ be a graph with

$$V_M = K \cup K' \cup \{v_s, v_t\} \text{ and}$$
$$A_M = \{(v_s,k) \mid k \in K\} \cup A \cup \{(l,v_t) \mid l \in K'\}.$$

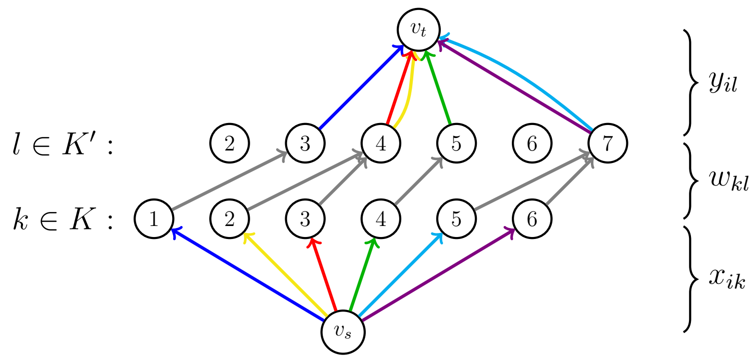



**Figure 3.1:** Positional multicommodity flow in $G$

Furthermore, let $I$ be a set of commodities. We define a *positional multicommodity flow* in $G_M$ as a multicommodity flow that sends one flow unit of each commodity $i \in I$ from $v_s$ to $v_t$, see Figure 3.1.

**Lemma 3.4.** *Feasible solutions of DP correspond to a positional multicommodity flow in $G_M$.*

*Proof.* Let $z \in \{0,1\}^{n \cdot |A|}$ be a feasible solution to DP and define $z_{ikl}$ as the flow value of commodity $i$ on the unique path $(v_s, k, l, v_t)$. By equality (2.32) it holds $\sum_{(k,l)\in A} z_{ikl} = 1$ for all $i \in I$. Consequently, every commodity has flow value one. □

In addition, the transformation (3.1)-(3.3) yields a bijection between the integer solutions of SEE, HPF and DP. Therefore, consider the example in Figure 3.1. There is shown a positional multicommodity flow of five commodities (colored) from $v_s$ to $v_t$. As in the proof, $z_{ikl}$ is the flow of commodity $i$ on the path $(v_s, k, l, v_t)$. Then the $x_{ik}$ variables correspond to the flow of commodity $i$ on the edge $(v_s, k)$. Similarly, $y_{il}$ is the flow of commodity $i \in I$ on the edge $(l, v_t)$. The variables $w_{kl}$ indicate the total 'anonymous' flow on edge $(k,l)$. By constraints (2.32) there is a one-to-one correspondence between the variables $z_{ikl}$ and $x_{ik}$, $y_{il}$. we need a further restriction because the model expects exactly one job entering each $k \in K$. For HPF, we additionally need to restrict DP such that each $k \in K$ gets assigned exactly one commodity. Then there is also a one-to-one correspondence between $z_{ikl}$ and $x_{ik}$, $w_{kl}$. We conclude it as follows.

**Corollary 3.5.** *From transformations* (3.1) *and* (3.2) *it follows*

$$z_{ikl} = 1 \iff x_{ik} = y_{il} = 1 \qquad \forall i \in I, (k,l) \in A$$

**Corollary 3.6.** *Assume* $\sum_{i \in I} x_{ik} = 1$ *for all* $k \in K$*. From transformations* (3.1) *and* (3.3) *it follows*

$$z_{ikl} = 1 \iff x_{ik} = w_{kl} = 1 \qquad \forall i \in I, (k,l) \in A$$

Moreover, we can construct a linear transformation between OOE and SEE, given by

$$u_{ih} = \sum_{k=1}^{h} x_{ik} - \sum_{l=2}^{h} y_{il} = \sum_{l=h+1}^{n+1} y_{il} - \sum_{k=h+1}^{n} x_{ik} \qquad \forall i \in I, h \in K \tag{3.4}$$

This equation states that $u_{ih} = 1$ if $x_{ik} = 1$ for some $k \leq h$ and $y_{il} = 0$ for all $l \leq k$. That means job $i$ is active at position $h$ if and only if $i$ has started before $h$ ans has not completed until $h$. Alternatively, the second equation states that job $i$ is active at position $h$ if and only if job $i$ completes, but does not start after position $h$. The second equation can also be received from equations (2.15) and (2.16), which yields for all $i \in I$ and $h \in K$:

$$\sum_{k=1}^{n} x_{ik} = \sum_{l=2}^{n+1} y_{il} = 1 \implies \sum_{k=1}^{h} x_{ik} - \sum_{l=2}^{h} y_{il} = \sum_{l=h+1}^{n+1} y_{il} - \sum_{k=h+1}^{n} x_{ik}.$$

Now we combine the transformations $OOE \to SEE$ and $SEE \to DP$ in order to receive the transformation $OOE \to DP$.

**Lemma 3.7.** *From OOE to DP, combining* (3.4) *with* (3.1) *and* (3.2) *yields*

$$u_{ih} = \sum_{(k,l) \in A_h} z_{ikl} \qquad \forall i \in I, h \in K \tag{3.5}$$

*Proof.* By using the definition we get

$$\begin{aligned}
u_{ih} &\overset{(3.4)}{=} \sum_{k=1}^{h} x_{ik} - \sum_{l=2}^{h} y_{il} \\
&\overset{(3.1),(3.2)}{=} \sum_{k=1}^{h} \sum_{l=k+1}^{n+1} z_{ikl} - \sum_{l=2}^{h} \sum_{k=1}^{l-1} z_{ikl} \\
&= \sum_{k=1}^{h} \sum_{l=k+1}^{n+1} z_{ikl} - \sum_{k=1}^{h-1} \sum_{l=k+1}^{h} z_{ikl} \\
&= \sum_{k=1}^{h} \sum_{l=h+1}^{n+1} z_{ikl} = \sum_{(k,l) \in A_h} z_{ikl} \qquad \forall i \in I, h \in K.
\end{aligned}$$

□

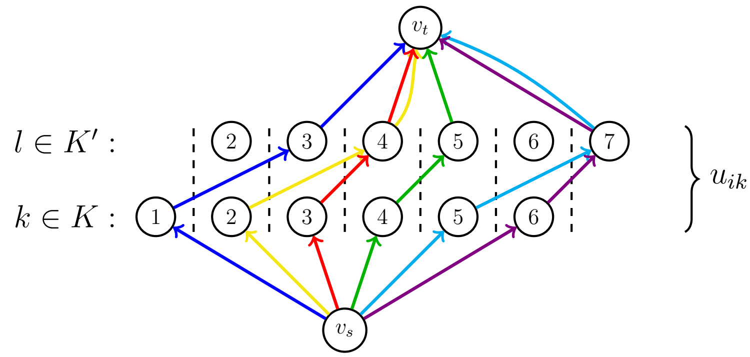



**Figure 3.2:** $u_{ik}$ is the flow of $i$ over the sequence cut $A_k$

**Corollary 3.8.** *From the transformation* (3.5) *it follows*

$$z_{ikl} = 1 \iff u_{ik} = \ldots = u_{il-1} = 1 \qquad \forall i \in I, (k,l) \in A$$

Interpreting Lemma 3.7 in $G_M$, we get $u_{ik}$ as the flow of commodity $i$ on the sequence cut at $k \in K$. These transformations have important structural consequences for the induced integer hulls of each model. In the following we omit variables $s_k \geq 0$, because they are mapped under identity. We also omit variables $f_{klr} \geq 0$ of the HPF model, since $z_{ikl} = x_{ik} = w_{kl} = 1$ implies $f_{klr} = d_{ir}$ for all $r \in R$ and zero otherwise in any integral solution of HPF by (2.38) and (2.43). Define the integer hulls of OOE, SEE, HPF and DP by

$$\begin{aligned}
P_I(OOE) &= conv\{u \in \{0,1\}^{n^2} \mid u \text{ satisfies (2.5)-(2.14) } \}\\
P_I(SEE) &= conv\{(x,y) \in \{0,1\}^{2n^2} \mid (x,y) \text{ satisfies (2.15)-(2.22) } \}\\
P_I(HPF) &= conv\{(x,w) \in \{0,1\}^{n^2+|A|} \mid (x,w) \mid \text{ satisfies (2.36)-(2.44) } \}\\
P_I(DP) &= conv\{z \in \{0,1\}^{n\cdot|A|} \mid z \text{ satisfies (2.32)-(2.35) } \}.
\end{aligned}$$

In addition, let $P_I(DPK)$ be the integer hull of $P_I(DP)$, if we add the constraints $\sum_{i\in I}\sum_{l=k+1}^{n+1} z_{ikl} = 1$ for all $k \in K$. Then define $\Phi(z) \mapsto (u,x,y,w)$ as the overall transformation that is composed of equations (3.1)-(3.3) and (3.5). Similarly, let $\tilde{\Phi}(x,y) \mapsto u$ denote the transformation (3.4). Let $\Phi_x(z)$ be the restriction of $\Phi(z)$ to variables $x$ (or $u, y, w$ respectively). From our previous observations we deduce that the integer hulls are equivalent under the transformations.

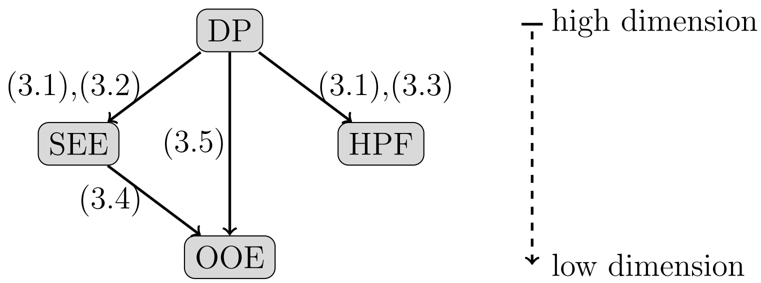



**Figure 3.3:** Linear transformations between the models

**Corollary 3.9.** $\Phi_u(P_I(DP)) = P_I(OOE)$

**Corollary 3.10.** $\Phi_{x,y}(P_I(DP)) = P_I(SEE)$

**Corollary 3.11.** $\Phi_{x,w}(P_I(DPK)) = P_I(HPF)$

**Corollary 3.12.** $\tilde{\Phi}(P_I(SEE)) = P_I(OOE)$

The transformation does not only state equality of the associated integer polytopes. Moreover, we get a full characterization of integrality in DP. That means in order to compute integral solutions for DP, it suffices to demand integrality on the substituted terms given by (3.1)-(3.3) and (3.5). The transformation scheme is illustrated in Figure 3.3. The given models yield a small variety of possible decisions that define solutions to the RCPSP in a compact representation. The next section compares the polyhedra of the LP-relaxation under the given transformations.

## 3.2 Comparison of LP-Relaxations

In Section 3.1 we showed that the models OOE, SEE, DP and DPK, HPF are equivalent under the linear transformations (3.1)-(3.3) and (3.5). It is of particular interest how the LP-relaxations are related under the same transformations. We show that the models get stronger with increasing variable space. That is, DP has the strongest LP-relaxation followed by SEE and OOE. Furthermore, DPK has a stronger LP-relaxation than HPF.

Before we show the pure results, we propose a function that measures the tightness of the transformations according to the different classes of inequalities. In short, we measure the distance between the normal vectors of the transformed and the dominating inequality. For our models we evaluate this indicator for all inequalities which yields polynomials in index-space.

### 3.2.1 Tightness Indicator

In the context of polyhedra, affine transformations are often used to construct, so called *aggregation schemes*, see Borndörfer and Weismantel [6]. The idea is to construct a combinatorial relaxation of the image space of a polyhedron with respect to an affine transformation. If the relaxation is well-separable then the associated cutting planes are translated back to the original polyhedron.

This section deals with linear transformations of polyhedra. Specifically, we want to measure the tightness of a transformed polyhedron compared to a polyhedron that is located in the image space. More formally, define a linear transformation $\Phi : \mathbb{R}^m \to \mathbb{R}^n, y \mapsto \Phi(y) = By$ with transformation matrix $B \in \mathbb{R}^{n\times m}$. Assume we are given two full-dimensional polyhedra $P_x \subset \mathbb{R}^n_{\geq 0}$ and $P_y \subset \mathbb{R}^m_{\geq 0}$. Let $a_x^T x \leq b$ and $a_y^T y \leq b$ be non-trivial valid inequalities for $P_x$ and $P_y$ respectively. We assume that both inequalities share the same right-hand side $b$, because we intend to compare inequalities that express the same constraint in different models. In other words, we require the cutting plane to be normalized.

Given vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ with $x = \Phi(y)$ the transformation $\Phi$ yields an inequality for $P_y$:

$$a_x^T x \leq b \iff a_x^T By \leq b \iff \bar{a}_x^T y \leq b$$

Now assume $\bar{a}_x^T y \leq b$ dominates $a_y^T y \leq b$, then it holds

$$\bar{a}_x^T y \leq a_y^T y \leq b. \tag{3.6}$$

Moreover, assume for each facet-defining inequality $a_x^T x \leq b$ of $P_x$ its transformed inequality $\bar{a}_x^T x \leq b$ is dominated by some facet-defining inequality $a_y^T y \leq b$ of $P_y$. Then, by construction, it holds $P_y \subseteq \Phi^{-1}(P_x)$ which is equivalent to $\Phi(P_y) \subseteq P_x$. It follows that $P_y$ represents the stronger formulation. In such constellations we want to measure the tightness between $\Phi(P_y)$ and $P_x$. We propose to evaluate the normal distance between their competing inequalities. That is, for each inequality $a_x^T x \leq b$ of $P_x$ we look for a dominating inequality $a_y^T y \leq b$ of $P_y$ according to (3.6) and compute

$$\Delta(a_x, a_y, \Phi) = \|\bar{a}_x - a_y\| = \|B^T a_x - a_y\|$$

where $\|\cdot\|$ denotes an arbitrary norm. For example, if we use the standard p-norm then $\Delta(a_x, a_y, \Phi)^p$ usually corresponds to a polynomial of degree $p$

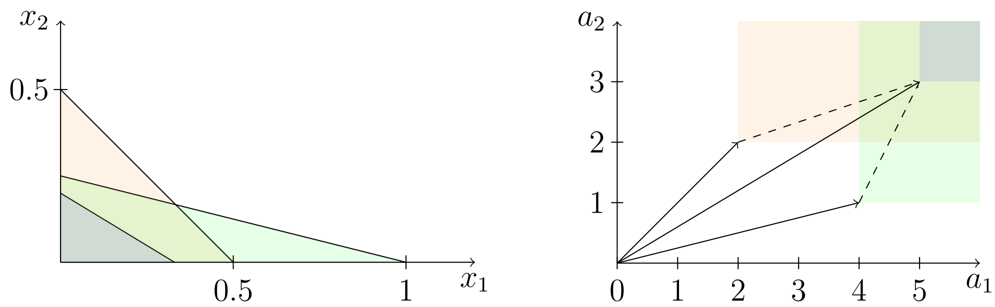



**Figure 3.4:** Three knapsack constraints: (a) $2x_1 + 2x_2 \leq 1$ (orange), (b) $4x_1 + x_2 \leq 1$ (green) and (c) $5x_1 + 3x_2 \leq 1$ (blue) and their dominating normal space. For $\|\cdot\|_1$ we have $\Delta(a, c) = 4$ and $\Delta(b, c) = 3$. For $\|\cdot\|_2$ we get $\Delta(a, c) \approx 3.162$ and $\Delta(b, c) \approx 2.236$. That means (c) dominates (a) 'more' than (b).

in the variable and constraint-index domain. In order to get the tightest representation we look for vectors $a_y$ with largest distance to $\bar{a}_x$. This can be found by enumeration of all inequalities of $P_y$. But mostly, we already get it from the context. In the following sections we evaluate the tightness indicator $\Delta$ to the linear transformations of our models. We use the standard 1-norm, that is $\|a\|_1 = \sum_{i=1}^{n} a_i$ for some vector $a \in \mathbb{R}^n$.

### 3.2.2 SEE to OOE

In this section we compare the SEE and OOE formulation in terms of their relaxed polyhedra. We will show that the LP-relaxation of SEE is tighter than OOE, up to a linear transformation. First, we will prove the general statement. In addition, we apply the indicator of Section 3.2.1 to identify the weak inequalities OOE compared to SEE. We consider OOE and SEE with the stronger inequalities that were introduced in Sections 2.2 and 2.3. Define the polyhedra of the linear relaxations of OOE and SEE as follows:

$$P(OOE) = \{(s, u) \in \mathbb{R}^{n+1}_{\geq 0} \times [0, 1]^{n^2} \mid (s, u) \text{ satisfies (2.5)-(2.14) } \}$$
$$P(SEE) = \{(s, x, y) \in \mathbb{R}^{n+1}_{\geq 0} \times [0, 1]^{2n^2} \mid (s, x, y) \text{ satisfies (2.15)-(2.22)}\}.$$

**Theorem 3.13.** *There exists a linear transformation $\Phi$ such that*

$$\Phi(P(SEE)) \subseteq P(OOE)$$

*Proof.* We define $\Phi$ as the transformation given by (3.4) and the $s_k$ variables are mapped under identity. We show that each inequality of OOE is implied by SEE. This is checked for each constraint separately. First, by (2.15) it holds $\sum_{k=h}^{n} x_{ik} = 1 - \sum_{k=1}^{h-1} x_{ik}$ and substitution in (2.20) yields:

$$\sum_{l=2}^{h} y_{il} + \sum_{k=h}^{n} x_{ik} \leq 1 \iff \sum_{k=1}^{h-1} x_{ik} - \sum_{l=2}^{h} y_{il} \geq 0 \qquad \forall i \in I, h \in K. \tag{3.7}$$

Then inequality (2.5) of OOE is implied by SEE, since

$$\begin{aligned}
\sum_{h\in K} u_{ih} &= \sum_{h\in K} \left( \sum_{k=1}^{h} x_{ik} - \sum_{l=2}^{h} y_{il} \right) \\
&= \sum_{h\in K} \left( \sum_{k=1}^{h-1} x_{ik} - \sum_{l=2}^{h} y_{il} \right) + \sum_{h\in K} x_{ik} \\
&\overset{(3.7)}{\geq} \sum_{h\in K} x_{ik} \overset{(2.15)}{=} 1 \qquad\qquad \forall i \in I
\end{aligned}$$

Constraints (2.6) are obtained directly:

$$\sum_{i\in I} d_{ir} u_{ik} = \sum_{i\in I} d_{ir} \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \right) \overset{(2.34)}{\leq} D_r \qquad \forall k \in K, r \in R.$$

Inequalities (2.7) are equal to (2.18). The stronger starting time constraints (2.12) are implied by

$$\begin{aligned}
& s_k - s_l + p_i(u_{ik} - u_{ik-1} - u_{il}) \\
= \;& s_k - s_l + p_i \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} - \sum_{k'=1}^{k-1} x_{ik'} + \sum_{k'=2}^{k-1} y_{ik'} - \sum_{k'=1}^{l} x_{ik'} + \sum_{k'=2}^{l} y_{ik'} \right) \\
\leq \;& s_k - s_l + p_i \left( \sum_{k'=2}^{l} y_{ik'} - \sum_{k'=1}^{k-1} x_{ik'} \right) \\
\overset{(2.15)}{=} \;& s_k - s_l + p_i \left( \sum_{k'=k}^{n} x_{ik'} + \sum_{k'=2}^{l} y_{ik'} - 1 \right) \overset{(2.22)}{\leq} 0 \qquad \forall i \in I, (k,l) \in A
\end{aligned}$$

The stronger contiguity constraints (2.13) are obtained as follows:

$$\begin{aligned}
u_{ih} - u_{ik} + u_{il} &= \sum_{k'=1}^{h} x_{ik'} - \sum_{k'=2}^{h} y_{ik'} - \sum_{k'=1}^{k} x_{ik'} + \sum_{k'=2}^{k} y_{ik'} + \sum_{k'=1}^{l} x_{ik'} - \sum_{k'=2}^{l} y_{ik'} \\
&= \sum_{k'=1}^{h} x_{ik'} - \sum_{k'=2}^{h} y_{ik'} + \sum_{k'=k+1}^{l} x_{ik'} - \sum_{k'=k+1}^{l} y_{ik'} \leq \sum_{k'=1}^{n} x_{ik'} \overset{(2.15)}{=} 1 \\
&\quad \forall i \in I, (h,k,l) \in K^3 : h < k < l
\end{aligned}$$

Similarly, the stronger precedence constraints (2.14) are implied by

$$\begin{aligned}
u_{jk} + u_{il} &= \sum_{k'=1}^{k} x_{jk'} - \sum_{k'=2}^{k} y_{jk'} + \sum_{k'=1}^{l} x_{ik'} - \sum_{k'=2}^{l} y_{ik'} \\
&\leq \sum_{k'=1}^{k} x_{jk'} + \sum_{k'=1}^{l} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \\
&\overset{(2.16)}{=} \sum_{k'=1}^{k} x_{jk'} + \sum_{k'=1}^{l} x_{ik'} + \sum_{k'=k+1}^{n+1} y_{ik'} - 1 \\
&\overset{(2.21)}{\leq} \sum_{k'=1}^{l} x_{ik'} \leq \sum_{k'=1}^{n} x_{ik'} \overset{(2.15)}{=} 1 \qquad \forall (i,j) \in E, (k,l) \in A
\end{aligned}$$

Finally, all inequalities of OOE are implied by SEE using the linear transformation (3.4). Consequently, $\Phi(P(SEE)) \subseteq P(OOE)$. □

From the proof we deduce the tightness parameters of each constraint presented in Table 3.1. It reveals that constraints (2.5)-(2.7) are tight under the transformation. The weakness of the OOE model is mainly due to (2.12), (2.13) and (2.14). In (2.13) and (2.14) the tightness parameter is constant and does not depend on the inequality. The starting time constraints (2.12) get weaker for OOE when $p_i$ and $l-k$ is large. Since the start time constraints mainly influence the bound of the LP-relaxation, the SEE model is supposed to have stronger bounds.

### 3.2.3 DP to SEE

Recall the models DP and the strengthened SEE model of Section 2.3. We show that the LP-relaxation of DP is tighter than the LP-relaxation of SEE

| No. | constraint | $\Delta(\Phi)$ | domain $\forall$ |
|---|---|---|---|
| (2.5) | activity | 0 | $i \in I$ |
| (2.6) | resource | 0 | $k \in K, r \in R$ |
| (2.7) | incr. start times | 0 | $k \in K$ |
| (2.12) | start time | $p_i \cdot (l - k + 1)$ | $i \in I, (k, l) \in A$ |
| (2.13) | contiguity | $n - 1$ | $i \in I, h < k < l$ |
| (2.14) | precedence | $n - 1$ | $(i, j) \in E, (k, l) \in A$ |

**Table 3.1:** Tightness values of OOE compared to SEE

with respect to a linear transformation. Therefore, consider the polyhedra of the LP-relaxations of SEE and DP:

$$P(SEE) = \{(s, x, y) \in \mathbb{R}^{n+1}_{\geq 0} \times [0, 1]^{2n^2} \mid (s, x, y) \text{ satisfies (2.15)-(2.22)}\}$$
$$P(DP) = \{(s, z) \in \mathbb{R}^{n+1}_{\geq 0} \times [0, 1]^{n \cdot |A|} \mid (s, z) \text{ satisfies (2.32)-(2.35)}\}$$

**Theorem 3.14.** *There exists a linear transformation $\Phi$ such that*

$$\Phi(P(DP)) \subseteq P(SEE)$$

*Proof.* Let $\Phi$ be the linear transformation given by equations (3.1), (3.2) and assume $s_k$ is mapped under identity. Inequalities (2.15) and (2.16) are obtained directly by

$$\sum_{k \in K} x_{ik} = \sum_{k \in K} \sum_{l=k+1}^{n+1} z_{ikl} = \sum_{(k,l) \in A} z_{ikl} \overset{(2.32)}{=} 1 \qquad \forall i \in I$$
$$\sum_{l \in K'} y_{il} = \sum_{l \in K'} \sum_{k=1}^{l-1} z_{ikl} = \sum_{(k,l) \in A} z_{ikl} \overset{(2.32)}{=} 1 \qquad \forall i \in I.$$

The stronger starting time constraints (2.22) we get by

$$\begin{aligned}
& s_k - s_l + p_i \left( \sum_{k'=k}^{n} x_{ik'} + \sum_{l'=2}^{l} y_{il'} - 1 \right) \\
\overset{(2.32)}{=} \;& s_k - s_l + p_i \left( \sum_{k'=k}^{n} \sum_{l'=k'+1}^{n+1} z_{ik'l'} + \sum_{l'=2}^{l} \sum_{k'=1}^{l'-1} z_{ik'l'} - \sum_{k'=1}^{n} \sum_{l'=k'+1}^{n+1} z_{ik'l'} \right) \\
= \;& s_k - s_l + p_i \left( \sum_{k'=k}^{l-1} \sum_{l'=k'+1}^{l} z_{ik'l'} - \sum_{k'=1}^{k-1} \sum_{l'=l+1}^{n+1} z_{ik'l'} \right) \\
\leq \;& s_k - s_l + p_i \sum_{k'=k}^{l-1} \sum_{l'=k'+1}^{l} z_{ik'l'} \overset{(2.33)}{\leq} 0 \qquad\qquad \forall i \in I, (k,l) \in A.
\end{aligned}$$

Constraints (2.18) are implied by (2.33). For the resource constraints (2.19) we obtain

$$\begin{aligned}
\sum_{i \in I} d_{ir} \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{l'=2}^{k} y_{il'} \right) &= \sum_{i \in I} d_{ir} \left( \sum_{k'=1}^{k} \sum_{l'=k'+1}^{n+1} z_{ik'l'} - \sum_{l'=2}^{k} \sum_{k'=1}^{l'-1} z_{ik'l'} \right) \\
&= \sum_{i \in I} d_{ir} \sum_{k'=1}^{k} \sum_{l'=k+1}^{n+1} z_{ik'l'} \\
&= \sum_{i \in I} d_{ir} \sum_{(k',l') \in A_k} z_{ik'l'} \overset{(2.34)}{\leq} D_r \qquad \forall k \in K, r \in R
\end{aligned}$$

In addition, inequalities (2.20) yield

$$\begin{aligned}
\sum_{k'=k}^{n} x_{ik'} + \sum_{l'=2}^{k} y_{il'} &= \sum_{k'=k}^{n} \sum_{l'=k'+1}^{n+1} z_{ik'l'} + \sum_{l'=2}^{k} \sum_{k'=1}^{l'-1} z_{ik'l'} \\
&= \sum_{k'=k}^{n} \sum_{l'=k+1}^{n+1} z_{ik'l'} + \sum_{k'=1}^{k-1} \sum_{l'=k'+1}^{k} z_{ik'l'} \\
&\leq \sum_{(k,l) \in A} z_{ik'l'} \overset{(2.32)}{=} 1 \qquad\qquad \forall i \in I, k \in K
\end{aligned}$$

The precedence constraints (2.21) are implied directly

$$\begin{aligned} & \sum_{k'=1}^{k} x_{jk'} + \sum_{l'=k+1}^{n+1} y_{il'} \\ = \quad & \sum_{k'=1}^{k} \sum_{l'=k'+1}^{n+1} z_{jk'l'} + \sum_{l'=k+1}^{n+1} \sum_{k'=1}^{l'-1} z_{ik'l'} \overset{(2.35)}{\leq} 1 \qquad \forall (i,j) \in E, k \in K \end{aligned}$$

Finally, all inequalities of SEE are implied by DP under the linear transformation $\Phi$, so $\Phi(P(DP)) \subseteq P(SEE)$. □

From the proof we deduce the tightness parameters of Table 3.2. The values reveal that the SEE model is weak at the starting time constraints, in particular for large values of $p_i$. Furthermore, constraints (2.20) which indicate that a job has to start before it completes is weak compared to DP. This seems natural because in DP this condition is already implied by the variables.

| No. | constraint | $\Delta(\Phi)$ | range $\forall$ |
|---|---|---|---|
| (2.15) | start | 0 | $i \in I$ |
| (2.16) | completion | 0 | $i \in I$ |
| (2.22) | start times | $p_i \cdot (k-1) \cdot (n-l+1)$ | $i \in I, (k,l) \in A$ |
| (2.19) | resource | 0 | $k \in K, r \in R$ |
| (2.20) | start $<$ end | $(k-1) \cdot (n-k+1)$ | $i \in I, k \in K$ |
| (2.21) | precedence | 0 | $(i,j) \in E, k \in K$ |

**Table 3.2:** Tightness values of SEE compared to DP

### 3.2.4 DPK to HPF

Consider the HPF model as defined by constraints (2.36)-(2.44). For DP we assume the additional constraint

$$\sum_{i \in I} \sum_{l=k+1}^{n+1} z_{ikl} = 1 \qquad \forall k \in K \tag{3.8}$$

which yields the model DPK. We show that the LP-relaxation of DPK is stronger than the LP-relaxation of HPF. Define the polyhedra of the LP-relaxations of HPF and DPK by

$$P(HPF) = \{(s, x, w, f) \in \mathbb{R}_{\geq 0}^{n+1} \times [0,1]^{n^2+|A|} \times \mathbb{R}_{\geq 0}^{|A|\cdot|R|} \mid \\ (s, x, w, f) \text{ satisfies } (2.36)\text{-}(2.44)\}$$

$$P(DPK) = \{(s, z) \in \mathbb{R}_{\geq 0}^{n+1} \times [0,1]^{n\cdot|A|} \mid (s, z) \text{ satisfies } (2.32)\text{-}(2.35),(3.8)\}.$$

**Theorem 3.15.** *There exists a linear transformation $\Phi$ such that*

$$\Phi(P(DPK)) \subseteq P(HPF)$$

*Proof.* Let $\Phi$ be defined by equations (3.1),(3.3), the identity map for variables $s_k$ and by the additional equations

$$f_{klr} = \sum_{i \in I} d_{ir} z_{ikl} \qquad \forall (k,l) \in A, r \in R \tag{3.9}$$

We show that under the transformation $\Phi$, DPK implies all inequalities of HPF. Constraints (2.36), (2.37) and (2.38) are obtained by

$$\sum_{k \in K} x_{ik} = \sum_{k \in K} \sum_{l=k+1}^{n+1} z_{ikl} = \sum_{(k,l) \in A} z_{ikl} \overset{(2.32)}{=} 1 \qquad \forall i \in I$$

$$\sum_{i \in I} x_{ik} = \sum_{i \in I} \sum_{l=k+1}^{n+1} z_{ikl} \overset{(3.8)}{=} 1 \qquad \forall k \in K$$

$$\sum_{l=k+1}^{n+1} w_{kl} = \sum_{l=k+1}^{n+1} \sum_{i \in I} z_{ikl} \overset{(3.8)}{=} 1 \qquad \forall k \in K.$$

For the starting time constraints (2.39) we get

$$\begin{aligned}
& s_k - s_l + p_i \left( x_{ik} + \sum_{l'=k+1}^{l} w_{kl'} - 1 \right) \\
= \quad & s_k - s_l + p_i \left( \sum_{l'=k+1}^{n+1} z_{ikl'} + \sum_{l'=k+1}^{l} \sum_{j \in I} z_{jkl'} - 1 \right) \\
= \quad & s_k - s_l + p_i \left( \sum_{l'=k+1}^{n+1} z_{ikl'} + \sum_{l'=k+1}^{l} z_{ikl'} + \sum_{j \in I \setminus \{i\}} \sum_{l'=k+1}^{l} z_{jkl'} - 1 \right) \\
\overset{(3.8)}{\leq} \quad & s_k - s_l + p_i \sum_{l'=k+1}^{l} z_{ikl'} \\
\leq \quad & s_k - s_l + p_i \sum_{k'=k}^{l-1} \sum_{l'=k'+1}^{l} z_{ik'l'} \overset{(2.33)}{\leq} 0 \qquad \forall i \in I, (k,l) \in A.
\end{aligned}$$

Constraints (2.40) are the same in both models and inequalities (2.41) we get directly from the transformation:

$$\sum_{i \in I} d_{ir} x_{ik} - \sum_{l=k+1}^{n+1} f_{klr} \overset{(3.9)}{=} \sum_{i \in I} \sum_{l=k+1}^{n+1} d_{ir} z_{ikl} - \sum_{l=k+1}^{n+1} \sum_{i \in I} d_{ir} z_{ikl} = 0 \quad \forall k \in K, r \in R.$$

The resource constraints (2.42) are induced directly by

$$\sum_{(k,l) \in A_h} f_{klr} \overset{(3.9)}{=} \sum_{(k,l) \in A_h} \sum_{i \in I} d_{ir} z_{ikl} \overset{(2.34)}{\leq} D_r \qquad \forall h \in K, r \in R.$$

For the capacity restrictions (2.43) it holds

$$f_{klr} - \bar{d}_r w_{kl} \overset{(3.9)}{=} \sum_{i \in I} d_{ir} z_{ikl} - \bar{d}_r \sum_{i \in I} z_{ikl} \leq 0 \qquad \forall (k,l) \in A, r \in R$$

because $d_{ir} \leq \bar{d}_r$ for all $r \in R$. Finally, the precedence constraints (2.44) are implied by

$$
\begin{aligned}
& x_{ik} + \sum_{k'=l}^{n} x_{ik'} + \sum_{k'=1}^{l-1} x_{jk'} - \sum_{l'=k+1}^{l-1} w_{kl'} \\
= \quad & \sum_{l'=k+1}^{n+1} z_{ikl'} + \sum_{k'=l}^{n} \sum_{l'=k'+1}^{n+1} z_{ik'l'} + \sum_{k'=1}^{l-1} \sum_{l'=k'+1}^{n+1} z_{jk'l'} - \sum_{l'=k+1}^{l-1} \sum_{i \in I} z_{ikl'} \\
= \quad & \sum_{l'=l}^{n+1} z_{ikl'} + \sum_{k'=l}^{n} \sum_{l'=k'+1}^{n+1} z_{ik'l'} + \sum_{k'=1}^{l-1} \sum_{l'=k'+1}^{n+1} z_{jk'l'} - \sum_{l'=k+1} \sum_{i' \in I \setminus \{i\}} z_{i'kl'} \\
\leq \quad & \sum_{l'=l}^{n+1} z_{ikl'} + \sum_{k'=l}^{n} \sum_{l'=k'+1}^{n+1} z_{ik'l'} + \sum_{k'=1}^{l-1} \sum_{l'=k'+1}^{n+1} z_{jk'l'} \\
\leq \quad & \sum_{l'=l}^{n+1} \sum_{k'=1}^{l'-1} z_{ik'l'} + \sum_{k'=1}^{l-1} \sum_{l'=k'+1}^{n+1} z_{jk'l'} \overset{(2.35)}{\leq} 1 \qquad \forall (i,j) \in E, l \in K'
\end{aligned}
$$

Finally, all inequalities of HPF are dominated by DP under the transformation $\Phi$, therefore $\Phi(P(DP)) \subseteq P(HPF)$. $\square$

From the proof we deduce the tightness parameters illustrated in Table 3.3. It reveals that the start time, ordering and precedence constraints are the weak inequalities with respect to $\Phi$. Especially the starting time constraints appear are expected to yield poor bounds. But we also see that the ordering constraints appear to be weak, if the maximum resource demand of some resource $r \in R$ is large compared to the average demand of all $i \in I$.

| No. | constraint | $\Delta(\Phi)$ | range $\forall$ |
|---|---|---|---|
| (2.47) | start | 0 | $i \in I$ |
| (2.48) | position | 0 | $i \in I$ |
| (2.49) | end | 0 | $i \in I, (k,l) \in A$ |
| (2.50) | start time | $p_i \cdot \left(n \cdot \binom{n}{2} - (n-1) \cdot (l-k) - n + k - 1 + \binom{l-k}{2}\right)$ | $i \in I, (k,l) \in A$ |
| (2.52) | fix flow | 0 | $k \in K, r \in R$ |
| (2.53) | resource | 0 | $h \in K, r \in R$ |
| (2.54) | ordering | $n \cdot \bar{d}_r - \sum_{i \in I} d_{ir}$ | $(k,l) \in A, r \in R$ |
| (2.55) | precedence | $(l-k-1) \cdot (n-1) + (l-2) \cdot (n-l+2)$ | $(i,j) \in E, (k,l) \in A$ |

**Table 3.3:** Tightness values of HPF compared to DP

### 3.2.5 Relation to DDT

This section compares the polyhedra of the compact models SEE and DP to the time-indexed model DDT. We show that by extensive expansion and, in turn, restriction of the solutions space it is possible to convert SEE and DP into the time-discrete model DDT. Based on the level of discretization, these operations can be seen as an approximate scheme to DDT.

#### SEE to DDT

Recall the SEE model and set $K = \mathcal{T} = \{1, ..., T\}$, that means we expand the position set to a larger set that represents the discrete time horizon. Since the starting times of the positions are implied by the discretization, we set $s_t = t$ for all $t \in \mathcal{T}$. The starting time constraints change to

$$p_i(\sum_{\tau=t}^{T} x_{i\tau} + \sum_{\tau=2}^{t'} y_{i\tau} - 1) \leq s_{t'} - s_t = t' - t \qquad \forall i \in I, (t, t') \in A$$

Consequently, it must hold $\sum_{\tau=t}^{T} x_{i\tau} + \sum_{\tau=2}^{t'} y_{i\tau} \leq 1$ for all $i \in I$ and $(t, t') \in A$ with $p_i - 1 \geq t' - t$. Without loss of generality we might also restrict $\sum_{\tau=t}^{T} x_{i\tau} + \sum_{\tau=2}^{t'} y_{i\tau} \leq 1$ for all $i \in I$ and $(t, t') \in A$ with $t' - t \geq p_i + 1$ since early jobs are preferred. We call the resulting model 'modified' SEE model.

**Lemma 3.16.** *Any integral solution to the modified SEE model satisfies* $x_{it} = \ y_{it+p_i}$ *for all* $i \in I, t \in \mathcal{T}$.

*Proof.* By constraint 2.15 there is exactly one $t \in \mathcal{T}$ and one $t' \in \mathcal{T}$ with $x_{it} = y_{it'} = 1$. From the assumptions made, this is only possible for $t' = t+p_i$. Otherwise it holds $x_{it} = y_{it+p_i} = 0$. □

Lemma 3.16 restricts the variable space to the $x_{it}$ variables. This leads to the following theorem.

**Theorem 3.17.** *Assume the SEE model with* $K = \mathcal{T} = \{1, ..., T\}$ *and set* $x_{it} = y_{it+p_i}$ *for all* $i \in I, t \in \mathcal{T}$. *The resulting model is equivalent to DDT.*

*Proof.* We show that all constraints of DDT are equally implied by the modified modified SEE model. The assignment constraints are implied directly:

$$\sum_{t\in\mathcal{T}} x_{it} \overset{(2.15)}{=} 1 \qquad \forall t \in \mathcal{T}.$$

By construction it holds $\sum_{t'=2}^{p_i} y_{it'} = 0$. For the resource constraints it follows

$$\begin{aligned}\sum_{i\in I}\sum_{t'=t-p_i+1}^{t} d_{ir}x_{it'} &= \sum_{i\in I} d_{ir}\left(\sum_{t'=1}^{t} x_{it'} - \sum_{t'=1}^{t-p_i} x_{it'}\right)\\ &= \sum_{i\in I} d_{ir}\left(\sum_{t'=1}^{t} x_{it'} - \sum_{t'=p_i+1}^{t} x_{it'-p_i}\right)\\ &= \sum_{i\in I} d_{ir}\left(\sum_{t'=1}^{t} x_{it'} - \sum_{t'=p_i+1}^{t} y_{it'}\right)\\ &= \sum_{i\in I} d_{ir}\left(\sum_{t'=1}^{t} x_{it'} - \sum_{t'=2}^{t} y_{it'}\right) \overset{(2.19)}{\leq} D_r \quad \forall r\in R, t\in\mathcal{T}\end{aligned}$$

For the precedence constraints it holds

$$\begin{aligned}\sum_{t'=1}^{t+p_i-1} x_{jt'} + \sum_{t'=t}^{T} x_{it'} &= \sum_{t'=1}^{t+p_i-1} x_{jt'} + \sum_{t'=t}^{T-p_i+1} y_{it'+p_i}\\ &= \sum_{t'=1}^{t+p_i-1} x_{jt'} + \sum_{t'=t+p_i}^{T+1} y_{it'} \overset{(2.21)}{\leq} 1 \quad \forall (i,j)\in E, t\in\mathcal{T}.\end{aligned}$$

Therefore, the modified SEE model is equivalent to DDT. □

### DP to DDT

For the DP model we use a similar approach as for SEE. Therefore, consider the expanded position set $K = \mathcal{T} = \{1, ..., T\}$ and set $s_t = t$ for all $t \in \mathcal{T}$. This implies the inequality

$$p_i \sum_{t\leq\tau<\tau'\leq t'} z_{i\tau\tau'} \leq s_{t'} - s_t = t' - t \qquad \forall i\in I, (t,t')\in A.$$

From that, it follows $\sum_{t\leq\tau<\tau'\leq t'} z_{i\tau\tau'} = 0$ for all $(t',t)\in A$ with $p_i - 1 \geq t' - t$. Again, we can further assume that $\sum_{t\leq\tau<\tau'\leq t'} z_{i\tau\tau'} = 0$ for all $(t',t)\in A$ with $t' - t \geq p_i + 1$ because jobs are supposed to finish as early as possible. The created model is denoted as 'modified' DP model.

**Lemma 3.18.** *If $z_{itt'} = 1$ in any integral solution to the modified DP model then $t' = t + p_i$.*

*Proof.* From our assumptions $z_{itt'} = 0$ for all $(t,t')$ with $t' - t \neq p_i$. Since $\sum_{(t,t')\in A} z_{itt'} = 1$ by (2.32) $z_{itt'} = 1$ is possible if and only if $t' = t + p_i$. □

Lemma 3.18 and the condition (2.32) allows us restrict to variables $z_{itt+p_i}$ since all other variables are zero.

**Theorem 3.19.** *Let $K = \mathcal{T}$ and assume the DP model with $s_t = t$ for all $t \in \mathcal{T}$. In addition, set $z_{itt'} = 0$ for all $t' - t \neq p_i$ and substitute $\bar{z}_{it} = z_{itt+p_i}$. The modified DP model is equivalent to DDT.*

*Proof.* We show that each constraint of DDT is equivalently implied by the modified DP model. The assignment condition is satisfied, since

$$\sum_{t\in\mathcal{T}} \bar{z}_{it} = \sum_{(t,t')\in A} z_{itt'} \overset{(2.32)}{=} 1 \qquad \forall i \in I$$

The resource constraints are implied by

$$\begin{aligned}
\sum_{i\in I} \sum_{t'=t-p_i+1}^{t} d_{ir}\bar{z}_{it'} &= \sum_{i\in I} \sum_{t'=t-p_i+1}^{t} d_{ir} z_{it't'+p_i} \\
&= \sum_{i\in I} \sum_{t'=1}^{t} \sum_{t''=t+1}^{T} d_{ir} z_{it't''} \\
&= \sum_{i\in I} \sum_{(t',t'')\in A_t} d_{ir} z_{it't''} \overset{(2.34)}{\leq} D_r \qquad \forall t \in \mathcal{T}, r \in R
\end{aligned}$$

and from (2.35) we obtain

$$\begin{aligned}
\sum_{t'=1}^{t+p_i-1} \bar{z}_{jt'} + \sum_{t'=t}^{T} \bar{z}_{it'} &= \sum_{t'=1}^{t+p_i-1} \sum_{t''=t'+1}^{T+1} z_{jt't''} + \sum_{t'=t}^{T} \sum_{t''=t'+1}^{T+1} z_{it't''} \\
&= \sum_{t'=1}^{t} \sum_{t''=t'+1}^{T+1} z_{jt't''} + \sum_{t''=t+1}^{T+1} \sum_{t'=1}^{t'-1} z_{it't''} \overset{(2.35)}{\leq} 1
\end{aligned}$$

Consequently, all constraints are equivalent under the substitution. Therefore, DDT and the modified DP model are equivalent. □

**Further Remarks**

The previous sections show that the models of DDT and SEE, DP are closely related. We showed that the compact models SEE and DP can be transformed into DDT by expansions and restrictions of the variable space. These operations depend on the discretization level, that is $\mathcal{T}$ for DDT (fine) and $K = \{1, ..., n\}$ for the compact models (coarse). This gives an approximative flavor between these models. According to this, we argue that the LP-relaxations of the compact models and DDT are incomparable in general.

Another significant difference between DDT and the compact models is the objective function. In the compact models the makespan variable $s_{n+1}$ is minimized which becomes constant in the modified version. In DDT, the implicit assumption that the intervals $t \in \mathcal{T}$ have of unit length allows the alternative makespan representation of $\sum_{t\in\mathcal{T}}(t-1) \cdot x_{n+1t}$. This definition is not applicable in the compact case, since the position-index gives no statement concerning the starting time of a job. Consequently, even if we equalize the models by the mentioned operations, the models consider a completely different objective function. Hence, it is hard to construct generally valid statements that compare the solution quality of the related models.

# Chapter 4

# Lifted Cover Inequalities

In this section we introduce a specific class of valid inequalities for the compact models OOE, SEE and DP which are known as *lifted cover inequalities.* A *minimal cover* is defined as a minimal subsets of jobs that cannot be scheduled simultaneously. On the basis of minimal covers it is possible to construct strong cutting planes of the underlying polytopes. These can be strengthened further by the so-called *lifting* method. If one can show that a cover inequality defines a facets of a certain subpolytope, then lifting extends this inequality to a facet-inducing inequality of the original polytope. Usually, cover inequalities and lifting are used in the context of knapsack inequalities. In our compact MIP models we apply them to knapsack subpolytopes induced by resource constraints.

First contributions to this method are due to Balas [50] in order to obtain facets for the classical knapsack polytope. Balas and Zemel [51] investigated upper and lower bounds on the lifting coefficients in order to improve computational aspects. The bounds are shown to be close to the exact coefficients. Similar results have been provided for set packing polyhedra by Padberg [54] and also Nemhauser and Trotter [55]. Their approaches were extended and generalized to arbitrary 0-1 programs by Padberg [53], Wolsey [56, 57] and also Zemel [52]. In terms of MIP formulations for the RCPSP, cover inequalities are rarely studied. Hardin et al. [60] propose cover inequalities for the DDT model and the uniform resources case, where $|R| = 1$. They present facet-inducing inequalities on specific sub-polytopes depending in minimal covers and propose a number of fast lifting techniques. Since minimal covers yield an exact representation on the set of jobs that can simultaneously be active, Stork and Uetz [61] dealt with the exact computation and complexity of all minimal covers. They showed that already deciding whether a given job is contained in some minimal cover is $\mathcal{N}P$-hard. Furthermore, they computed the number of all minimal covers on practical instances that turns out to grow rapidly with increasing job number.

Our work focuses on cover inequalities for the compact models OOE, SEE and DP. We prove that cover inequalities define facets of certain subpolytopes. Afterwards, the cover inequalities can be extended to facets by the general lifting method. We show that lifting can be performed more efficiently, if the precedence constraints are relaxed. This assumption yields lower bounds on the exact lifting coefficients. Moreover, the relaxed lifting runs in polynomial time and it is independent of the model size.

## 4.1 Minimal Cover Inequalities

A subset $C \subseteq I$ is called a *cover*, if $\sum_{i \in C} d_{ir} > D_r$ for some $r \in R$ and for all $i, j \in C$ it holds $(i,j), (j,i) \notin E$. A cover $C$ is a *minimal cover*, if $\sum_{i \in C \setminus \{j\}} d_{ir} \leq D_r$ for all $j \in C$ and $r \in R$. In any feasible schedule, the number of simultaneously scheduled jobs of $C$ must not exceed $|C| - 1$. Let $\mathcal{C} \subseteq 2^I$ denote the set of all minimal covers. More generally, the set $\mathcal{C}$ corresponds to the minimal dependent sets of the independence system $(I, \mathcal{F})$ where

$$\mathcal{F} = \left\{ I' \subseteq I \mid \sum_{i \in I'} d_{ir} \leq D_r \ \forall r \in R \wedge (i,j) \notin E \ \forall i,j \in I' \right\}.$$

In this case, the dependent sets are denoted as feasible job subsets $F \in \mathcal{F}$ where $\mathcal{F}$ includes all feasible job subsets.

### 4.1.1 OOE Model

First, we introduce the minimal cover inequalities for the OOE model. To make our construction more convenient we omit the $s_k$ variables, since the cover inequalities affect only the 0-1 variables. In addition, we allow that not every job has to be scheduled. Consider the resulting 0-1 program defined by the inequalities

$$\begin{aligned}
\sum_{i\in I} d_{ir}u_{ik} &\leq D_r && \forall k\in K, r\in R && (4.1)\\
u_{ih} - u_{ik} + u_{il} &\leq 1 && \forall i\in I, (h,k,l)\in K^3 : h<k<l && (4.2)\\
u_{jk} + u_{il} &\leq 1 && \forall (i,j)\in E, (k,l)\in A && (4.3)\\
u_{ik} &\in \{0,1\} && \forall i\in I, k\in K
\end{aligned}$$

We define the induced 0-1 polytope and its linear relaxation by

$$\begin{aligned}
P_I(OOE) &= conv\{u\in\{0,1\}^{n^2} \mid u \text{ satisfies } (4.1)\text{-}(4.3)\}\\
P(OOE) &= \{u\in[0,1]^{n^2} \mid u \text{ satisfies } (4.1)\text{-}(4.3)\}.
\end{aligned}$$

The polytope $P_I(OOE)$ has full dimension $n^2$ since it includes all unit vectors. Given a minimal cover $C$ and position $k\in K$ define the corresponding cover inequality as

$$\sum_{i\in C} u_{ik} \leq |C| - 1 \tag{4.4}$$

which states that at most $|C|-1$ jobs can be active at position $k$. Define the restricted cover polytope

$$P_C(OOE) = P_I(OOE)\cap\{u\in[0,1]^{n^2} \mid u_{ik'} = 0 \;\; \forall i\in I\setminus C, k'\in K\}$$

as the polytope which is restricted to the cover $C$. Due to the restrictions, $P_C(OOE)$ has dimension $n^2 - n\cdot(n-|C|) = |C|\cdot n$.

**Theorem 4.1.** *Let $C$ be a minimal cover and $k\in K$. The corresponding cover inequality* (4.4) *is facet-defining for $P_C(OOE)$.*

*Proof.* We construct $|C|\cdot n$ linearly independent vectors in $P_C(OOE)$ that satisfy (4.4) with equality. Consider an arbitrary ordering $C=\{i_1,...,i_{|C|}\}$. For each pair $(i_q,l)$ with $1\leq q\leq|C|$ and $l\in K$ define a vector $u^{(i_q,l)}\in\{0,1\}^{n^2}$ with entries $u_{ih}^{(i_q,l)}$ as follows:

$$u_{ih}^{(i_q,l)} = \begin{cases} 1 & \text{if } (i=i_q \wedge h=l \wedge h\neq k)\vee(i\in C\setminus\{i_q\}\wedge h=k)\\ 0 & \text{else}\end{cases}$$

Any such vector satisfies (4.1)-(4.3) since $i_q$ is the only job that is active at some position $h \neq k$ and the $|C|-1$ jobs of $C \setminus \{i_q\}$ are active at position $k$. In particular, (4.4) is satisfied with equality. Now order the vectors by jobs first and then by positions. This yields the vector sequence

$$\{u^{(i_1,1)}, ..., u^{(i_1,n)}, u^{(i_2,1)}, ..., u^{(i_2,n)}, ..., u^{(i_{|C|},n)}\}.$$

We apply the same index-ordering for the vector entries. If we omit the zero rows of all $i \in I \setminus C$ we obtain a matrix $B \in \{0,1\}^{|C|\cdot n \times |C| \cdot n}$ of the form

$$B = \begin{pmatrix} \mathbb{1}_k & M_k & M_k & \dots & M_k \\ M_k & \mathbb{1}_k & M_k & \dots & M_k \\ M_k & M_k & \mathbb{1}_k & \dots & M_k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M_k & M_k & M_k & \dots & \mathbb{1}_k \end{pmatrix}$$

The submatrix $\mathbb{1}_k \in \{0,1\}^{n \times n}$ has ones on the diagonal except for the $k$-th row and column. The matrix $M_k \in \{0,1\}^{n \times n}$ has the $k$-th row containing only ones. Therefore, any row that belongs to some position $h \neq k$ has exactly one non-zero entry. We perform Laplace expansion on these rows and obtain a matrix $B' \in \{0,1\}^{|C| \times |C|}$ of the form

$$B' = \begin{pmatrix} 0 & 1 & 1 & \dots & 1 \\ 1 & 0 & 1 & \dots & 1 \\ 1 & 1 & 0 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 0 \end{pmatrix}$$

Obviously, $B'$ has full rank, so $|\det B| = |\det B'| \neq 0$. That means the vector sequence consists of $|C| \cdot n$ linearly independent vectors which satisfy (4.4) with equality. Consequently, the cover inequality is facet-defining for $P_C(OOE)$. □

An illustration to minimal cover inequalities on a subpolytope is given in Figure 4.1.

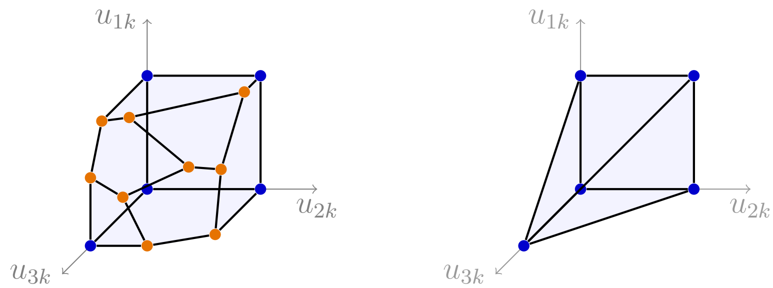



**Figure 4.1:** LP-relaxation of a MDKC subpolytope (left) and after adding two minimal cover cuts (right).

### 4.1.2 SEE Model

As before, we focus on the contained 0-1 polytope of the SEE model by omitting the $s_k$ variables. Furthermore, we assume that not every job has to be scheduled. Then SEE is given by the inequalities

$$\sum_{k \in K} x_{ik} \leq 1 \qquad \forall i \in I \tag{4.5}$$

$$\sum_{l \in K'} y_{il} \leq 1 \qquad \forall i \in I \tag{4.6}$$

$$\sum_{k'=1}^{k} d_{ir}(x_{ik'} - y_{ik'}) \leq D_r \qquad \forall k \in K, r \in R \tag{4.7}$$

$$\sum_{l'=2}^{k} y_{il'} - \sum_{k'=1}^{k-1} x_{ik'} \leq 0 \qquad \forall k \in K \tag{4.8}$$

$$\sum_{k'=1}^{k} x_{jk'} + \sum_{l'=k+1}^{n+1} y_{il'} \leq 1 \qquad \forall (i,j) \in E, k \in K \tag{4.9}$$

$$x_{ik} \in \{0,1\} \qquad \forall i \in I, k \in K$$

$$y_{il} \in \{0,1\} \qquad \forall i \in I, l \in K'$$

Note that constraints (4.8) are the same as (2.20) under the assumption that not every job has to complete. Define the induced integer hull and its LP-relaxation by

$$P_I(SEE) = conv\{(x,y) \in \{0,1\}^{2n^2} \mid (x,y) \text{ satisfy (4.5)-(4.9)}\}$$

$$P(SEE) = \{(x,y) \in [0,1]^{2n^2} \mid (x,y) \text{ satisfy (4.5)-(4.9)}\}$$

**Lemma 4.2.** *The polytope $P_I(SEE)$ has full dimension $2n^2$.*

*Proof.* We construct $2n^2$ linearly independent vectors in $P_I(SEE)$. For this, take an arbitrary ordering $I = \{i_1, ..., i_n\}$ of the jobs. Then define two types of tuples $(i_q, k)$ with with $1 \leq q \leq n$ and $k \in K$ and $(i_q, k)'$ with $1 \leq q \leq n$ and $k \in K'$. For each $(i_q, k)$ define a vector $(x, y)^{(i_q,k)} \in \{0, 1\}^{2n^2}$ as follows

$$x_{ih}^{(i_q,k)} = \begin{cases} 1 & \text{if } i = i_q \wedge k = h \\ 0 & \text{else} \end{cases} \qquad \forall i \in I, h \in K$$

$$y_{ih}^{(i_q,k)} = 0 \qquad \forall i \in I, h \in K'$$

In addition, for each tuple $(i_q, k)'$ define a vector $(x, y)^{(i_q,k)'} \in \{0, 1\}^{2n^2}$ with

$$x_{ih}^{(i_q,k)'} = \begin{cases} 1 & \text{if } i = i_q \wedge k = h + 1 \\ 0 & \text{else} \end{cases} \qquad \forall i \in I, h \in K$$

$$y_{ih}^{(i_q,k)'} = \begin{cases} 1 & \text{if } i = i_q \wedge k = h \\ 0 & \text{else} \end{cases} \qquad \forall i \in I, h \in K'$$

Any vector has at most one job active at each position $k \in K$, consequently (4.5)-(4.9) is satisfied. Then sort the tuples $(i_q, k)$ and $(i_q, k)'$ by

$$\{(i_1, 1), ..., (i_1, n), (i_2, 1), ..., (i_2, n), ..., (i_n, n)\} \text{ and}$$
$$\{(i_1, 2)', ..., (i_1, n + 1)', (i_2, 2)', ..., (i_2, n + 1)', ..., (i_n, n + 1)'\}.$$

We construct a matrix by first taking all vectors $(x, y)^{(i_q,k)}$ and then all $(x, y)^{(i_q,k)'}$ with respect to the sorting. The vector entries are sorted equally. The resulting matrix $M \in \{0, 1\}^{2n^2 \times 2n^2}$ has the form

$$M = \left( \begin{array}{c|c} I & I \\ \hline 0 & I \end{array} \right)$$

where $I \in \{0, 1\}^{n^2 \times n^2}$ is the identity matrix having ones on the diagonal. It follows that $M$ has full rank. Therefore, the $2n^2$ vectors are linearly independent and they are contained in $P_I(SEE)$. Consequently, $P_I(SEE)$ has full dimension $2n^2$.

$\square$

Let $C$ be a minimal cover and $k \in K$. The corresponding cover inequality of the SEE model is defined as

$$\sum_{i \in C} \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \right) \leq |C| - 1 \tag{4.10}$$

For given minimal cover $C$ define the restricted cover polytope by

$$P_C(SEE) = P_I(SEE) \cap \{(x,y) \in [0,1]^{2n^2} \mid x_{ik'} = y_{ik'+1} = 0 \ \forall i \in I \setminus C, k' \in K\}$$

**Lemma 4.3.** *The polytope $P_C(SEE)$ has dimension $2n \cdot |C|$.*

*Proof.* By Lemma 4.2, $P_I(SEE)$ has full dimension $2n^2$. In $P_C(SEE)$ we fix $2n \cdot (n - |C|)$ variables of $P_I(SEE)$, therefore $P_C(SEE)$ has dimension $2n \cdot |C|$. $\square$

**Theorem 4.4.** *Let $C$ be a minimal cover and $k \in K$. The corresponding cover inequality* (4.10) *defines a facet of $P_C(SEE)$.*

*Proof.* We construct $2n \cdot |C|$ linearly independent vectors in $P_C(SEE)$ that satisfy (4.10) with equality. Consider an arbitrary ordering $C = \{i_1, ..., i_{|C|}\}$. Define two types of tuples $(i_q, l)$ with $1 \leq q \leq n$, $l \in K$ and $(i_q, l)'$ with $1 \leq q \leq n$, $l \in K'$. For each tuple $(i_q, l)$ define a vector $(x,y)^{(i_q,l)} \in \{0,1\}^{2n^2}$ that satisfies

$$x_{ih}^{(i_q,l)} = \begin{cases} 1 & \text{if } i = i_q \wedge h = l \wedge h \neq k \\ 1 & \text{if } i \in C \setminus \{i_q\} \wedge h = k \\ 0 & \text{else} \end{cases} \qquad \forall i \in I, h \in K$$

$$y_{ih}^{(i_q,l)} = 0 \qquad \forall i \in I, h \in K'.$$

For each tuple $(i_q, l)'$ define a vector $(x,y)^{(i_q,l)'} \in \{0,1\}^{2n^2}$ with

$$x_{ih}^{(i_q,l)'} = \begin{cases} 1 & \text{if } i = i_q \wedge h + 1 = l \wedge h \neq k \\ 1 & \text{if } i \in C \setminus \{i_q\} \wedge h + 1 = k \\ 0 & \text{else} \end{cases} \qquad \forall i \in I, h \in K$$

$$y_{ih}^{(i_q,l)'} = \begin{cases} 1 & \text{if } i = i_q \wedge h = l \wedge h \neq k + 1 \\ 1 & \text{if } i \in C \setminus \{i_q\} \wedge h = k \\ 0 & \text{else} \end{cases} \qquad \forall i \in I, h \in K'$$

We observe that all vectors are valid for $P_C(SEE)$, because $i_q$ is the only job that is active at some position $h \neq k$. Moreover, each vector satisfies (4.10) with equality since all jobs $i \in C \setminus \{i_q\}$ are active at position $k$. Define an ordering of the tuples $(i_q, l)$ and $(i_q, l)'$ by

$$\{(i_1, 1), ..., (i_1, n), (i_2, 1), ...., (i_2, n), ..., (i_n, n)\} \text{ and}$$
$$\{(i_1, 2)', ..., (i_1, n+1)', (i_2, 2)', ...., (i_2, n+1)', ..., (i_n, n+1)'\}.$$

We create a matrix by first taking all vectors $(x, y)^{(i_q,l)}$ and then all $(x, y)^{(i_q,l)'}$ with respect to the ordering. The vector entries are sorted analogously. If we delete all rows that correspond to jobs $i \in I \setminus C$ we obtain a matrix $M \in \{0, 1\}^{2n\cdot|C|\times 2n\cdot|C|}$ of the form

$$M = \left(\begin{array}{c|c} M' & M' \\ \hline 0 & M' \end{array}\right)$$

where $M' \in \{0, 1\}^{n\cdot|C|\times n\cdot|C|}$ has the form

$$M' = \begin{pmatrix} \mathbb{1}_k & M_k & M_k & \dots & M_k \\ M_k & \mathbb{1}_k & M_k & \dots & M_k \\ M_k & M_k & \mathbb{1}_k & \dots & M_k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M_k & M_k & M_k & \dots & \mathbb{1}_k \end{pmatrix}$$

The matrix $\mathbb{1}_k \in \{0, 1\}^{n\times n}$ has ones on the diagonal except for the $k$-th row and column. The matrix $M_k \in \{0, 1\}^{n\times n}$ has the $k$-th row containing only ones. Any row that corresponds to a position $h \neq k$ contains exactly one non-zero entry. We perform Laplace expansion on these rows and obtain a matrix $M'' \in \{0, 1\}^{|C|\times|C|}$ with

$$M'' = \begin{pmatrix} 0 & 1 & 1 & \dots & 1 \\ 1 & 0 & 1 & \dots & 1 \\ 1 & 1 & 0 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 0 \end{pmatrix}$$

Obviously, $\det M'' \neq 0$, therefore $|\det M| = |\det M'| = |\det M''| \neq 0$. This implies that the generated vectors are linearly independent and satisfy (4.10) with equality. Consequently, (4.10) is facet defining for $P_C(SEE)$.

□

### 4.1.3 DP Model

As for the previous models, we omit the $s_k$ variables and focus on the induced 0-1 polytope. For convenience in the construction, we assume that not every job has to be scheduled. Then the DP model is given by the inequalities

$$\sum_{(k,l)\in A} z_{ikl} \le 1 \qquad \forall i \in I \tag{4.11}$$

$$\sum_{i\in I}\sum_{(k,l)\in A_h} d_{ir} z_{ikl} \le D_r \qquad \forall h \in K, r \in R \tag{4.12}$$

$$\sum_{k=1}^{h}\sum_{l=k+1}^{n+1} z_{jkl} + \sum_{l=h+1}^{n+1}\sum_{k=1}^{l-1} z_{ikl} \le 1 \qquad \forall (i,j) \in E, h \in K \tag{4.13}$$

$$z_{ikl} \in \{0,1\} \qquad \forall i \in I, (k,l) \in A$$

Let $|A| = m$ and denote the polytopes of the integer hull and its linear relaxation by

$$P_I(DP) = conv\{z \in \{0,1\}^{n\cdot m} \mid z \text{ satisfies } (4.11)-(4.13)\}$$
$$P(DP) = \{z \in [0,1]^{n\cdot m} \mid z \text{ satisfies } (4.11)-(4.13)\}.$$

The polytope $P_I(DP)$ has full dimension $n \cdot m$ because it contains all unit vectors. Let $C$ be a minimal cover and $h \in K$. Define the corresponding cover inequality by

$$\sum_{i\in C}\sum_{(k,l)\in A_h} z_{ikl} \le |C| - 1 \tag{4.14}$$

For given minimal cover $C$ the restricted cover polytope of DP is defined by

$$P_C(DP) = P_I(DP) \cap \{z \in [0,1]^{n\cdot m} \mid z_{ikl} = 0 \ \forall i \in I \setminus C, (k,l) \in A\}$$

As before, $P_C(DP)$ contains all unit vectors, so its dimension equals $|C| \cdot m$.

**Theorem 4.5.** *Let $C$ be a minimal cover and $h \in K$. The corresponding cover inequality* (4.14) *defines a facet of $P_C(DP)$.*

*Proof.* It suffices to construct $|C| \cdot m$ linearly independent vectors. Consider an arbitrary orderings of $C = \{i_1, ..., i_{|C|}\}$ and $A = \{a_1, ..., a_m\}$. Then define an arbitrary but fixed 'special' arc $a^* \in A_h$. For simplicity, let $i_{|C|+1} = i_1$.

For each tuple $(i_q, a)$ with $1 \leq q \leq |C|$ and $a \in A_h$ we define a vector $z^{(i_q,a)} \in \{0,1\}^{n \cdot m}$ as follows:

$$z_{ikl}^{(i_q,a)} = \begin{cases} 1 & \text{if } i = i_{q+1} \wedge (k,l) \in A_h \wedge (k,l) = a \\ 1 & \text{if } i \in C \setminus \{i_q, i_{q+1}\} \wedge (k,l) = a^* \\ 0 & \text{else} \end{cases}$$

In addition, for each tuple $(i_q, a)$ with $1 \leq q \leq |C|$ and $a \in A \setminus A_h$ create a vector $z^{(i_q,a)} \in \{0,1\}^{n \cdot m}$ with

$$z_{ikl}^{(i_q,a)} = \begin{cases} 1 & \text{if } i = i_q \wedge (k,l) \notin A_h \wedge (k,l) = a \\ 1 & \text{if } i \in C \setminus \{i_q\} \wedge (k,l) = a^* \\ 0 & \text{else} \end{cases}$$

Each such vector satisfies (4.11)-(4.13), since $C \setminus \{i_q\}$ contains the active jobs on $A_h$. Therefore, equality holds for (4.14). We partition the tuples by $(C, A_h)$ and $(C, A \setminus A_h)$. Then we sort the corresponding tuples of $(C, A_h)$ and $(C, A \setminus A_h)$ by jobs first and then by arcs with respect to the predefined ordering. Then apply this sorting to the created vectors and their entries. Further delete all zero rows associated to $i \in I \setminus C$. This yields a matrix $M \in \{0,1\}^{|C| \cdot m \times |C| \cdot m}$ of the form

$$M = \left( \begin{array}{c|c} M_1 & M_2 \\ \hline 0 & I \end{array} \right)$$

where $I \in \{0,1\}^{|C| \times |A \setminus A_h|}$ is the identity matrix and $M_1 \in \{0,1\}^{|C| \times |A_h|}$ has the shape

$$M_1 = \begin{pmatrix} 0 & \mathbb{1}_{a^*} & \mathbb{1}_{a^*} & \dots & \mathbb{1}_{a^*} & I' \\ I' & 0 & \mathbb{1}_{a^*} & \dots & \mathbb{1}_{a^*} & \mathbb{1}_{a^*} \\ \mathbb{1}_{a^*} & I' & 0 & \dots & \mathbb{1}_{a^*} & \mathbb{1}_{a^*} \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 & \mathbb{1}_{a^*} \\ \mathbb{1}_{a^*} & \mathbb{1}_{a^*} & \mathbb{1}_{a^*} & \dots & I' & 0 \end{pmatrix}$$

In this case $I' \in \{0,1\}^{|A_h| \times |A_h|}$ is the identity matrix and $\mathbb{1}_{a^*} \in \{0,1\}^{|A_h| \times |A_h|}$ contains only ones in the row corresponding to $a^*$. Therefore, any row that is associated to some $a \in A_h \setminus \{a^*\}$ has only one non-zero entry. We perform Laplace expansion on these rows and get a matrix $M' \in \{0,1\}^{|C|\times|C|}$ which looks like:

$$M' = \begin{pmatrix} 0 & 1 & 1 & \dots & 1 \\ 1 & 0 & 1 & \dots & 1 \\ 1 & 1 & 0 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 0 \end{pmatrix}$$

Hence, $M'$ has full rank which implies $|\det M| = |\det M_1| = |\det M'| \neq 0$. It follows that the generated vectors are linearly independent and satisfy (4.14) with equality. Therefore, (4.14) defines a facet of $P_C(DP)$. □

## 4.2 Sequential Lifting

In the last section we confirmed that, for given minimal cover $C$ and $k \in K$, the corresponding minimal cover inequality is facet-defining for the restricted cover polytope $P_C \subseteq P_I$. In general, the minimal cover inequalities do not define facets of the initial polytope $P_I$. A common technique to strengthen such inequalities is *lifting*. The idea is to compute maximal valid coefficients for a subset of fixed variables by solving an optimization problem on $P_C$, see Zemel [52]. If the variables are released after coefficient strengthening, we receive a polytope of higher dimension where the extended inequality still defines a facet. This method is repeated in order to obtain a facet of $P_I$.

In this section we first give an introduction to the general method. In particular, we focus on *sequential lifting* where the variables are lifted sequentially according to a specific lifting sequence. In addition, we show that the minimal cover inequalities of any compact model can be lifted efficiently when the precedence constraints are relaxed. The resulting lifting problem reduced to a sequence of multidimensional knapsack problems which gives lower bounds on the exact lifting coefficients. In particular, the complexity of lifting is model-independent.

### 4.2.1 Integer 0-1 Programs

In this section we temporarily forget about the introduced notation to allow easier access to the following aspects. Suppose we are given an integer 0-1 program with constraint matrix $A \in \mathbb{R}^{m \times n}$ consisting of columns $a_i \in \mathbb{R}^m$ with $i \in \{1, ..., n\}$, right-hand side $b \in \mathbb{R}^m$ and cost vector $c \in \mathbb{R}^n$ as follows:

$$\begin{aligned} &\max c^T x \\ &\sum_{i \in I} a_i x_i \leq b \\ &\qquad x_i \in \{0, 1\} \qquad\qquad \forall i \in \{1, ..., n\}. \end{aligned}$$

Let $I = \{1, ..., n\}$ be the variable domain. Further, let

$$P_I = conv\{x \in \{0, 1\}^n \mid \sum_{i \in I} a_i x_i \leq b\}$$

be the induced integer hull. For given $C \subseteq I$ define the subpolytope

$$P_C = P_I \cap \{x_i = 0 \mid i \in I \setminus C\}$$

Assume an inequality $\pi^T x \leq \pi_0$ is facet-defining for $P_C$ and w.l.o.g. $\pi_i = 0$ for all $i \in I \setminus C$. For given $j \in I \setminus C$ we are interested in the maximum valid coefficient $\pi_j$ such that $\sum_{i \in C} \pi_i x_i + \pi_j x_j \leq \pi_0$ is valid for $P_I$ and $P_{C \cup \{j\}}$ respectively. The direct way to compute this is by assuming $x_j = 1$ which yields $\pi_j \leq \pi_0 - \sum_{i \in C} \pi_i x_i$ and then to compute the smallest possible right-hand side of the inequality. This leads to an optimization problem $H_j$ that is defined as follows:

$$\begin{aligned} \max \ \phi_j = &\sum_{i \in C} \pi_i x_i \\ &\sum_{i \in C} a_i x_i \leq b - a_j \\ &\qquad x_i \in \{0, 1\} \qquad\qquad \forall i \in C. \end{aligned}$$

The lifting coefficient of $x_j$ becomes $\pi_j = \pi_0 - \phi_j$ and $j$ is added to $C$. This method is repeated iteratively with all remaining variables. The crucial theorem according to this method was developed simultaneously by Padberg [53] and Wolsey [56], but see also Nemhauser and Trotter [55].

**Theorem 4.6** (**Padberg [53]**). *Let $C \subset N$ and $\pi^T x \leq \pi_0$ be a non-trivial facet of $P_C$. Define an ordering $I \setminus C = \{j_1, ..., j_t\}$ and compute the lifting coefficients $\pi_j$ by sequentially solving the subproblems $H_j$ for $j = j_1, ..., j_t$. The resulting inequality $\sum_{i \in C} \pi_i x_i + \sum_{j \in I \setminus C} \pi_j x_j \leq \pi_0$ is facet-defining for $P_I$.*

This means, given a facet of a subpolytope, it is possible to compute facets of the initial polytope by successively solving optimization problems on $P_C$. The sequence $I \setminus C = \{j_1, ..., j_t\}$ is denoted as the *lifting sequence*. In general, different lifting sequences may yield different facets and there may exist facets of $P_I$ that are not obtainable by sequential lifting, see Gu et al. [58]. For arbitrary constraint matrices $A$, it is further important that the lifting sequence is valid, i.e. the subproblems $H_j$ must admit a feasible solution in each lifting step. In the next section we consider lifting according to the multidimensional knapsack problem. Later, this is used as subroutine to rapidly generate strong lifting coefficients for OOE, SEE and DP respectively.

### 4.2.2 Multidimensional Knapsack Problem

In the compact models OOE, SEE and DP the resource constraints imply multidimensional knapsack inequalities at each position $k \in K$. According to that, we consider the *multidimensional knapsack problem*, and even more specific, *with conflict constraints* (MDKC). There we are given items $i \in I$ with resource demands $d_{ir}$ and a knapsack with capacities $D_r$ for all $r \in R$. In addition, there are item conflicts $(i, j) \in E$, where $i$ and $j$ must not appear in the knapsack simultaneously. Given profits $v_i$ for all $i \in I$ the objective is to find a subset $I' \subseteq I$ of maximal profit such that $\sum_{i \in I'} d_{ir} \leq D_r$ for all $r \in R$ and $(i, j) \notin E$ for all $i, j \in I'$.

For the one-dimensional case, several heuristic, exact and approximative approaches have been developed, see for example Yamada et al. [45], Pferschy and Schauer [43], Hifi et al. [46, 47, 48, 49] and Battinelli et al.[44]. In the context of lifted cover inequalities, the general multidimensional knapsack problem has been studied by Kaparis et al. [42] and Gabrel et al. [41]. The MDKC can be stated by an integer program as follows:

$$\begin{aligned} \max \sum_{i \in I} v_i x_i & & \\ \sum_{i \in I} d_{ir} x_i \leq D_r & & \forall r \in R \\ x_i + x_j \leq 1 & & \forall (i, j) \in E \\ x_i \in \{0, 1\} & & \forall i \in I \end{aligned}$$

From our definition of a minimal cover $C$, we have $|\{i, j\} \cap C| \leq 1$ for all

$(i, j) \in E$. Therefore, the cover inequality

$$\sum_{i \in C} x_i \leq |C| - 1$$

is valid for MDKC. With the notion of $P_I$ and $P_C$ from the last section, one can easily verify that the cover inequality defines a facet of $P_C$. In order to lift this cover inequality, we first determine an arbitrary lifting sequence $I \setminus C = \{i_1, ..., i_N\}$. Define $C_0 = \emptyset$ and $C_t = \{i_1, ..., i_t\}$ for all $t \in \{1, ..., N\}$. As stated in Section 4.2.1, for the computation of the lifting coefficients $\pi_{i_t}$ we recursively solve the subproblem

$$\begin{aligned} \max \ \phi_t = \sum_{i \in C} x_i + \sum_{i \in C_{t-1}} \pi_i x_i & \\ \sum_{i \in C} d_{ir} x_i + \sum_{i \in C_{t-1}} d_{ir} x_i \leq D_r - d_{i_t r} & \quad \forall r \in R \\ x_i + x_j \leq 1 & \quad \forall (i, j) \in E \cap (C \cup C_{t-1})^2 \\ x_i \leq 0 & \quad \forall (i, i_t) \in E : i \in C \cup C_{t-1} \\ x_i \in \{0, 1\} & \quad \forall i \in C \cup C_{t-1} \end{aligned}$$

The lifting coefficients of job $i_t$ becomes $\pi_{i_t} = |C| - 1 - \phi_t$. Throughout the next sections we use the notion $MDK(C, i_t) := \phi_t$ which denotes the $t$-th optimal solution of the above knapsack problem according to the initial cover $C$ and the partial lifting sequence $C_t$.

### 4.2.3 Compact MIP

As mentioned, each compact model induces a minimal cover inequality at each position. Hence, Theorem 4.6 in combination with the results of Section 4.1 states that exact lifting of a cover inequality of OOE, SEE or DP yields facets of $P_I(OOE)$, $P_I(SEE)$ or $P_I(DP)$ respectively.

**Corollary 4.7.** *Given a cover inequality of OOE, exact lifting yields a facet of $P_I(OOE)$.*

**Corollary 4.8.** *Given a cover inequality of SEE, exact lifting yields a facet of $P_I(SEE)$.*

**Corollary 4.9.** *Given a cover inequality of DP, exact lifting yields a facet of $P_I(DP)$.*

Any lifting step requires to solve an optimization problem on the currently lifted subpolytope. Especially, if the dedicated optimization problem is hard then lifting might get intractable. In this case, it is possible to take a relaxation of the problem. In particular, lifting an inequality on a relaxed polytope yields a valid inequality for the original polytope, see the following observation.

**Observation 4.10.** *Let $P_I$ and $P'_I$ be integer polytopes with $P_I \subseteq P'_I$. Furthermore, let $\pi^T x \leq \pi_0$ be an inequality of $P'_I$ that is obtained by sequential lifting. Then $\pi^T x \leq \pi_0$ is valid for $P_I$.*

In the following sections we assume the relaxation of the precedence constraints. This is because it is hard to maintain a generally valid problem structure during lifting that both includes precedences and allows fast computation. The next sections show that, without precedence constraints, we can restrict to multidimensional knapsack computations in each lifting step. Since all lifting coefficients are positive, we get lower bounds on the exact lifting coefficients.

### OOE Model

Recall the definitions of Section 4.1.1. For a minimal cover $C$ and $k \in K$ the cover inequality of the OOE model is given by

$$\sum_{i \in C} u_{ik} \leq |C| - 1$$

By Theorem 4.1, the cover inequality is facet-defining for $P_C(OOE)$. We compute the lifting coefficients for the remaining restricted variables $u_{jl}$ with $j \in I \setminus C$, $l \in K$. Therefore, take an arbitrary ordering $I \setminus C = \{i_1, ..., i_N\}$ and define the lifting sequence of variable indices as follows:

$$S = \{(i_1, 1), ..., (i_1, n), (i_2, 1), ..., (i_2, n), ..., (i_N, n)\}.$$

**Proposition 4.11.** *Let $C$ be a minimal cover and $k \in K$. Sequentially lifting the corresponding cover inequality with respect to $S$ yields the lifting coefficients*

$$\pi_{i_t l} = \begin{cases} |C| - 1 - MDK(C, i_t) & l = k \\ 0 & l \neq k \end{cases}$$

*for all variables $u_{i_t l}$ with $1 \leq t \leq N$ and $l \in K$.*

*Proof.* We perform induction on the number of currently lifted variables. Let $u_{i_1 1} = 1$ be the first lifted variable. The lifting problem is to maximize the number of jobs at position $k$. For $k = 1$ we solve a MKDC at position 1 which gives

$$\pi_{i_1 1} = |C| - 1 - MDK(C, i_1).$$

If $k = 2$ then choose $|C| - 1$ jobs of $C$ that are active at position $k$ which yields

$$\pi_{i_1 1} = |C| - 1 - (|C| - 1) = 0.$$

In the inductive step let $u_{i_t l}$ be the currently lifted variable. By induction hypothesis, the lifting problem is to find a subset $I' \subseteq C \cup C_{t-1}$ where $\sum_{i \in I'} \pi_{ik}$ is maximal. First, assume $l \neq k$ then choose $|C| - 1$ jobs of $C$ to be active at position $k$ which yields

$$\pi_{i_t l} = |C| - 1 - (|C| - 1) = 0.$$

and for $k = l$ we solve a MKDC at position $k$ which gives

$$\pi_{i_t l} = |C| - 1 - MDK(C, i_t).$$

Consequently, we get the desired result. $\square$

**SEE Model**

Let $C$ be a minimal cover and $k \in K$. Consider the cover inequality for SEE:

$$\sum_{i \in C} \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \right) \leq |C| - 1.$$

By Theorem 4.4, the cover inequality is facet-defining for $P_C(SEE)$. Define an arbitrary ordering $I \setminus C = \{i_1, ..., i_N\}$. Let $(i_t, l)$ denote an index tuples that belongs to variables $x_{i_t l}$ and $(i_t, l)'$ the tuples that belong to $y_{i_t l}$. For convenience, write

$$(i_t, K) = (i_t, 1), ..., (i_t, n) \text{ and } (i_t, K')' = (i_t, 2)', ..., (i_t, n+1)'.$$

Then define the lifting sequence

$$S = \{(i_1, K), (i_1, K')', (i_2, K), (i_2, K')', ..., (i_N, K), (i_N, K')'\}.$$

**Proposition 4.12.** *Let $C$ be a minimal cover and $k \in K$. Sequentially lifting the corresponding cover inequality with respect to $S$ yields the lifting coefficients*

$$\pi_{i_t l} = \begin{cases} |C| - 1 - MDK(C, i_t) & l \le k \\ 0 & l > k \end{cases}$$

*for all variables $x_{i_t l}$ with $1 \le t \le N$, $l \in K$ and*

$$\pi'_{i_t l} = \begin{cases} MDK(C, i_t) - |C| + 1 & l \le k \\ 0 & l > k \end{cases}$$

*for all variables $y_{i_t l}$ with $1 \le t \le N$, $l \in K'$.*

*Proof.* We perform induction on the number of lifted variables. Lifting the first variable $x_{i_1 1} = 1$ yields that job $i_1$ is active at position $k = 1$ since all $y_{i_1 l} = 0$ with $l \in K'$ are unlifted yet. The first lifting problem is to maximize the number of active jobs at position $k = 1$. Since job $i_1$ is active at position $k = 1$ we solve a MDKC with respect to the already lifted variables, which belong to $C$. Therefore, it holds $\pi_{i_1 1} = |C| - 1 - MDK(C, i_1)$. If we lift $x_{i_1 2} = 1$ then we set $|C| - 1$ jobs of $C$ active at $k = 1$ which gives $\pi_{i_1 1} = |C| - 1 - (|C| - 1) = 0$. Furthermore, lifting $y_{i_1 2} = 1$ yields $x_{i_1 1} = 1$ by (4.8). Hence, job $i_1$ is active at position 1 and we solve the same MDKC:

$$\pi'_{i_1 2} = |C| - 1 - \pi_{i_1 1} - MDK(C, i_1) = 0.$$

Otherwise, if $k = 2$ then job $i_1$ is not active at position $k$ and we choose $|C| - 1$ jobs of $C$ to be active at position $k$ which gives

$$\pi'_{i_1 2} = |C| - 1 - \pi_{i_1 1} - (|C| - 1) = MDK(C, i_1) - |C| + 1.$$

Now we proceed with the inductive step. By induction hypothesis, the lifting problem is to find a subset $I' \subseteq C \cup C_{t-1}$ where $\sum_{i \in I'} \sum_{k'=1}^{k} (\pi_{ik'} - \pi'_{ik'})$ is maximal. Let $x_{i_t l} = 1$ be the current lifting variable. If $l \le k$ then job $i_t$ is active at position $k$ because $y_{i_t l'} = 0$ for all $l' \in K'$ appears later in the lifting sequence. We receive the coefficient

$$\pi_{i_t l} = |C| - 1 - MDK(C, i_t).$$

Otherwise, if $l > k$ then job $i_t$ is not active at position $k$ and we choose $|C| - 1$ jobs of $C$ to be active at position $k$. This yields

$$\pi_{i_t l} = |C| - 1 - (|C| - 1) = 0$$

which shows the first part. Next, let $y_{i_t l} = 1$ be the current lifting variable. By (4.8) there exists $l' < l$ such that $x_{i_t l'} = 1$. The case $l \leq k$ implies that job $i_t$ is not active at position $k$ and therefore we can choose $|C| - 1$ jobs of $C$ to be active at position $k$ which yields

$$\pi'_{i_t l} = |C| - 1 - \pi_{i_t l'} - (|C| - 1) = MDK(C, i_t) - |C| + 1.$$

For the case $l > k$ job $i_t$ is either active at position $k$ ($l' \leq k$) or not ($l' > k$). In the first case, we have to solve a MDKC which gives

$$\pi'_{i_t l} = |C| - 1 - \pi_{i_t l'} - MDK(C, i_t) = 0$$

and in the second case we place $|C| - 1$ jobs of $C$ at position $k$ and get

$$\pi'_{i_t l} = |C| - 1 - (|C| - 1) = 0.$$

Finally, the statement is validated for every case.

□

### DP Model

For a minimal cover $C$ and $h \in K$ the DP model admits the cover inequality:

$$\sum_{i \in C} \sum_{(k,l) \in A_h} z_{ikl} \leq |C| - 1.$$

By Theorem (4.5), the cover inequality is facet-inducing for $P_C(DP)$. Define an arbitrary ordering of $I \setminus C = \{i_1, ..., i_N\}$ and consider the lifting sequence of variable indices

$$S = \{(i_1, A_h), (i_1, A \setminus A_h), ..., (i_N, A_h), (i_N, A \setminus A_h)\}$$

where $(i_t, A')$ contains all variable indices $(i_t, k, l)$ with $(k, l) \in A' \subseteq A$ in an arbitrary order.

**Proposition 4.13.** *Let $C$ be a minimal cover and $h \in K$. Sequentially lifting the corresponding cover inequality with respect to $S$ yields the lifting coefficients*

$$\pi_{i_t kl} = \begin{cases} |C| - 1 - MDK(C, i_t) & (k,l) \in A_h \\ 0 & (k,l) \in A \setminus A_h \end{cases}$$

*for all variables $z_{i_t kl}$ with $1 \leq t \leq N$ and $(k,l) \in A$.*

*Proof.* We show the statement by induction on the number of lifted variables. Let $z_{i_1 kl} = 1$ with $(k,l) \in A_h$ be the first lifted variable in $S$. The lifting problem is to maximize the number of active arcs in $A_h$. Since job $i_1$ is assigned to $(k,l) \in A_h$ we solve a MDKC which yields the lifting coefficient

$$\pi_{i_1 kl} = |C| - 1 - MDK(C, i_1).$$

If $(k,l) \in A \setminus A_h$ we pick $|C| - 1$ jobs of $C$ that have an arc in $A_h$. This gives

$$\pi_{i_1 kl} = |C| - 1 - (|C| - 1) = 0.$$

In the inductive step let $z_{i_t kl} = 1$ be the currently lifted variable. By induction hypothesis, the lifting problem is to find a subset $I' \subseteq C \cup C_{t-1}$ where $\sum_{i \in I'} \sum_{(k,l) \in A_h} \pi_{ikl}$ is maximal. First, if $(k,l) \in A_h$ we have to solve a MDKC which yields

$$\pi_{i_t kl} = |C| - 1 - MDK(C, i_t)$$

and otherwise we assign $|C| - 1$ jobs of $C$ to $A \setminus A_h$, then

$$\pi_{i_t kl} = |C| - 1 - (|C| - 1) = 0.$$

Consequently, the statement is valid. □

## 4.3 Algorithms

In this section we introduce the algorithmic view on the lifted cover inequalities. It consists of two parts. First, given a solution to the LP-relaxation, find the maximum violated cover inequality. Second, given a cover inequality, find a valid lifting sequence and execute the lifting method. In the latter case we refer to our observations made in Section 4.2.

### 4.3.1 Separation

Given an optimal LP solution to any of the models OOE, SEE or DP, we want to determine a minimal cover inequality at some $k \in K$ that violates the LP solution most. Let $u^*$ (OOE), $x^*$, $y^*$ (SEE) and $z^*$ (DP) be the optimal values of the LP-relaxations. Since cover inequalities consider only the active part of each job $i$ at position $k$ we take the substitutions according to equations (3.4) and (3.5):

$$u^*_{ik} = \sum_{k'=1}^{k}(x^*_{ik'} - y^*_{ik'}) = \sum_{(k',l')\in A_k} z^*_{ik'l'}.$$

Consequently, the separation problem is to find a position $k \in K$ and a minimal cover $C$ such that $\sum_{i\in C} u^*_{ik} > |C| - 1$. This problem can be solved in polynomial time by dynamic programming, see Algorithm 1.

**Proposition 4.14.** *Let $k \in K$ and compute $C_r = minCover(k, I, \emptyset, D_r, r)$ for each $r \in R$. The cover $C_r$ with maximum $\sum_{i\in C_r} u^*_{ik}$ for all $r \in R$ yields the most violated cover inequality at position $k$.*

*Proof.* Assume $C_r$ is the returned job set. We have to show that $C_r$ is a minimal cover. Line 1 achieves that no complete pair $(i, j) \in E$ is contained in $C_r$. Inductively, we can assume that the algorithm expects a partial cover $C'$. Then $C'$ is returned if and only if $D' < 0$ and $D' + \min_{i\in C'} d_{ir} \geq 0$ which is equivalent to $D_r < \sum_{i\in C'} d_{ir}$ and $D_r \geq \sum_{i\in C'\setminus\{j\}}$ with $j \in C'$ such that $d_{jr} \leq d_{ir}$ for all $i \in C'$. This condition is equivalent to the definition of a minimal cover. By the natural recursion, the algorithm returns the cover of maximum weight. Hence, the algorithm is correct. □

**Corollary 4.15.** *Given $k \in K$, the maximum violated cover inequality can be computed in $\mathcal{O}(|R| \cdot D_{max} \cdot n)$.*

*Proof.* By Proposition (4.14), Algorithm 1 is called for each $r \in R$. It corresponds to the filling of a $D_r \times n$ matrix where each entry $(D', i_q)$ contains an optimal minimal cover for capacity $D'$ when only jobs $i \in \{i_1, ..., i_q\}$ are used. This yields a maximum running time of $\mathcal{O}(|R| \cdot D_{max} \cdot n)$. □

Note that even if the computed cover inequality is not violated by the current LP solution, it may get violated after lifting the inequality, see the next section.

### 4.3.2 Lifting

In the following we condense our results from Section 4.2 into an algorithmic approach. Assume we are given a position $k \in K$, a minimal cover $C$ and the associated cover inequality. For the compact models, we take the lifting sequences of Propositions 4.11, 4.12 and 4.13 respectively. In particular, all of the lifting sequences depend on an arbitrary ordering of $I \setminus C$. We showed that any of the lifting problems reduces to a lifting problem of a MDKC. This can be solved in polynomial time. In particular, the lifting coefficients of the MDKC uniquely determine the lifting coefficients of the compact models, as given in Section 4.2. Therefore, the computational effort for this lifting approach, see Algorithm 2 is independent of the model size.

**Proposition 4.16.** *Given a minimal cover $C$, calling $lift(C)$ yields valid lifting coefficients.*

*Proof.* In line 1 we determine an arbitrary ordering of $I \setminus C$. As in Section 4.2, this ordering defines the lifting sequence $S$. The MDKC subproblems are solved for each job $i_t$ with $1 \leq t \leq N$. According to Section 4.2, the values $MDK(C, i_t)$ suffice to define each lifting coefficient $\pi_q$ with $q \in S$, which is returned by the algorithm. □

**Proposition 4.17.** *Lifting one cover inequality runs in $\mathcal{O}(n^2 \cdot D_{max}^{|R|})$.*

*Proof.* An ordering of $I \setminus C$ can be found in $\mathcal{O}(n)$. An optimal solution to MDKC corresponds to the is found in $\mathcal{O}(n \cdot D_{max}^{|R|})$ with dynamic programming. Thus, iterating for each $t \in \{1, ..., N\}$ yields a total running time of $\mathcal{O}(n^2 \cdot D_{max}^{|R|})$. □

In conclusion, we introduced cover inequalities for the models OOE, SEE and DP. We showed that they define facets of the restricted cover subpolytopes. In addition, we introduced the lifting method in order to obtain facets for the integer hull of the original problem. We relaxed the precedence constraints, in order to reduce the lifting process to a sequence of MKDC problems. This gives lower bounds on the exact lifting coefficients.

The introduced lifting method was not implemented and tested practically. However, we believe that the computation of lifted cover cuts is fast in practice. It remains to analyze the running time and the bounds of this method compared to exact lifting coefficients. In the next section we present a further type of cutting plane. In particular, its general construction includes the lifted cover inequalities. However, the computation is more involving and it is based on randomization. Computational results to this method are presented in the sequel.

**Algorithm 1:** $minCover(k, I', C', D', r)$

**Input**: position $k \in K$, remaining jobs $I' = \{i_1, i_2, ..., i_q\}$, current (partial) minimal cover $C'$, remaining capacity $D'$, resource $r \in R$

**Output**: minimal cover of with respect to $k \in K$, $I' \subseteq I$, $D' \in \mathbb{Z}$, $r \in R$

**if** $\exists\ i, j \in C' : (i, j) \in E \vee (j, i) \in E$ **then**
  **return** $\emptyset$
**else if** $D' < 0$ **then**
  $\underline{d}_r \leftarrow \min_{i \in C'}\ d_{ir}$
  **if** $D' + \underline{d}_r \geq 0$ **then**
    **return** $C'$
  **else**
    **return** $\emptyset$
**else if** $I' = \emptyset$ **then**
  **return** $\emptyset$
**else**
  $C_1 = minCover(I' \setminus \{i_q\}, C' \cup \{i_q\}, D' - d_{i_q r})$
  $C_2 = minCover(I' \setminus \{i_q\}, C', D')$
  **if** $\sum_{i \in C_1} u^*_{ik} \geq \sum_{i \in C_2} u^*_{ik}$ **then**
    **return** $C_1$
  **else**
    **return** $C_2$

**Algorithm 2:** $lift(C)$

**Input**: minimal cover $C$

**Output**: lifting coefficients $\pi_q$, $q \in S$

define ordering $I \setminus C = \{i_1, ..., i_N\}$
**for** $t \in \{1, ..., N\}$ **do**
  $\phi_t \leftarrow MDK(C, i_t)$
set the lifting coefficients $(\pi_q)_{q \in S}$ according to $\phi$ see Section 4.2
**return** lifted inequality with coefficients $\pi = (\pi_q)_{q \in S}$

# Chapter 5

# Linear Programming Lower Bounds

In addition to exact solving methods, the literature studies lower bounds for the RCPSP. Most commonly, lower bounds are obtained by so-called *destructive improvement*, see [30]. The idea is to prove the non-existence of a schedule for given makespan. If such a schedule does not exist, the lower bound is increased by one. This iterative process is repeated either incrementally or by binary search. One possibility to detect the infeasibility of a schedule yields linear programming. In the following we introduce two conventional LP models from the literature. We show that they are related by a linear transformation and deduce strong inequalities for one model from a primal-dual characterization. From that, we derive two cutting plane algorithms. Finally, we observe that the same cuts can be transferred to the compact MIP models by linear extension. In order to preserve the compactness of the models we propose a randomized cut generation scheme.

## 5.1 Two Linear Programming Models

In order to compute lower bounds for the RCPSP, the literature mainly provides two linear programming models. The first is based on feasible jobs subsets and the computation of a non-preemptive relaxation of the schedule. In particular, it has exponentially many variables. The second model considers partial assignments of jobs to given intervals. In contrast, it has polynomially many variables and various cutting planes are proposed to strengthen the obtained lower bound.

Both models consider a fixed makespan $C_{max}$ and decompose the scheduling horizon into discrete intervals that are build as follows. According to the precedence network $G$ and $C_{max}$ the earliest starting time $e_i$ and latest completion time $l_i$ is computed for each job $i \in I$. These values are sorted in increasing order $\bar{s}_1 < ... < \bar{s}_{|K|} < \bar{s}_{|K|+1}$ and form the interval set $K$ where $\bar{s}_{|K|+1} = C_{max}$ denotes the constant makespan. In particular, $[\bar{s}_k, \bar{s}_{k+1}]$ is the associated interval to $k \in K$. Furthermore, define $K_i = \{k \in K \mid e_i \leq \bar{s}_k < \bar{s}_{k+1} \leq l_i\}$ as the set of feasible intervals for job $i \in I$. Similarly, let $I_k = \{i \in I \mid k \in K_i\}$ be the set of jobs that can be scheduled in interval $k \in K$.

### 5.1.1 The Model of Brucker and Knust (1998)

The first model of Brucker and Knust [22] is an extension of the model of Mingozzi et al. [13] which introduces feasible job subsets $F \in \mathcal{F}$ that satisfy $\sum_{i \in F} d_{ir} \leq D_r$ for all $r \in R$ and where $\mathcal{F}$ is the set of all feasible subsets. Moreover, $\mathcal{F}_k = \{F \in \mathcal{F} \mid F \subseteq I_k\}$ are the feasible subsets that are valid for the interval $k \in K$. The idea is to each $F \in \mathcal{F}$ a variable duration such that every $i \in I$ is covered for at least $p_i$ time units by subsets that contain job $i$. Since in any integral schedule the job subsets occur in sequence, the makespan equals the total length of all feasible subsets. The relaxation is caused by the non-preemtivity of the subsets and the relaxation of precedence constraints. Let $\xi_{Fk} \geq 0$ be the duration of $F \in \mathcal{F}$ in interval $k \in K$. Since we intend to solve the decision problem, let $\alpha_k \geq 0$ be the time exceeding of interval $k \in K$. In particular, if the optimal solution satisfies $\sum_{k \in K} \alpha_k > 0$ then the problem is infeasible for the given makespan. The linear program of Brucker and Knust [22] reads as follows:

$$\min \sum_{k \in K} \alpha_k$$

$$\sum_{k \in K} \sum_{F \in \mathcal{F}_k : i \in F} \xi_{Fk} \geq p_i \qquad \forall i \in I \tag{5.1}$$

$$\sum_{F \in \mathcal{F}_k} \xi_{Fk} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K \tag{5.2}$$

$$\xi_{Fk} \geq 0 \qquad \forall k \in K, F \in \mathcal{F}_k$$

$$\alpha_k \geq 0 \qquad \forall k \in K$$

In (5.1) each job $i \in I$ must be covered by feasible subsets that include $i$ for a total duration of $p_i$. The interval restrictions are expressed by (5.2).

Baptiste and Demassey [24] improved the above LP by additional cutting planes. Currently, this combined approach still achieves the best lower bounds on a considerable number of PSPLIB instances [65]. Further note the strong relation to the well-known *cutting stock problem* [63]. In this case, the set $\mathcal{F}$ corresponds to feasible roll patterns and there is a demand of $p_i$ units of each roll $i \in I$. The total pattern length (makespan) is minimized. However, the general problem statement does not consider non-preemptivity, precedences or feasible intervals.

### 5.1.2 The Model of Carlier and Néron (2003)

The second linear programming approach was introduced by Carlier and Néron [23] who state a polynomial LP which is extended by different cutting planes. As before, there are intervals $k \in K$ in which all jobs have to be processed. Define $\mu_{ik} \geq 0$ as the partial duration of job $i \in I$ that is assigned to interval $k \in K_i$. Again, $\alpha_k \geq 0$ denotes the time exceeding of interval $k \in K$. The basic LP model of Carlier and Néron [23] states as follows:

$$\min \sum_{k \in K} \alpha_k$$

$$\sum_{i \in I} \sum_{k \in K_i} \mu_{ik} \geq p_i \qquad \forall i \in I \tag{5.3}$$

$$\mu_{ik} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall i \in I, k \in K_i \tag{5.4}$$

$$\mu_{ik} \geq 0 \qquad \forall i \in I, k \in K_i$$

$$\alpha_k \geq 0 \qquad \forall k \in K$$

Inequalities (5.3) indicate that for each job $i \in I$ the total duration $p_i$ must be assigned to feasible intervals. In addition, the partial duration $\mu_{ik}$ must not exceed the interval length of $k \in K$ for all $i \in I$.

Note that the basic model is weak since there is no consideration of resource constraints. Therefore, the basic model is extended by valid cutting planes which are mainly due to [23, 34, 25]. In the following we mention the most important ones.

### Core Time

The concept of *core times* was used by Baptiste [39] and Klein et al. [30] in order to estimate the minimum induced time of each job in a given interval. In particular, it counts the overlap time in interval $k \in K$, if job $i \in I_k$ is scheduled either at the earliest or latest possible starting time. Therefore, let $e_i$ denote the earliest starting time and $l_i$ the latest completion time of job $i \in I$. The core time $p(i,k)$ of job $i \in I$ in interval $k \in K_i$ is given by

$$p(i,k) = \min\{\bar{s}_{k+1} - \bar{s}_k, p_i, \max\{0, e_i + p_i - \bar{s}_k\}, \max\{0, \bar{s}_{k+1} - l_i + p_i\}\}.$$

We derive the valid inequalities

$$\mu_{ik} \geq p(i,k) \qquad \forall i \in I, k \in K_i. \tag{5.5}$$

### Energetic Reasoning Cuts

The term *energetic reasoning* is widely used in project scheduling and has occurrences in various solving approaches such as MIP and constraint programming, see for example Carlier and Néron [23], Haouari et al.[26] and Schulz [27]. It states that the total induced volume (or energy) of some resource $r \in R$ in interval $k \in K$ must not exceed the available volume. This is expressed by the *energetic reasoning cuts*

$$\sum_{i \in I_k} d_{ir}\mu_{ik} - \alpha_k \leq D_r(\bar{s}_{k+1} - \bar{s}_k) \qquad \forall k \in K, r \in R \tag{5.6}$$

These inequalities combine time- and resource allocation in a one-dimensional knapsack constraint. The following lemma shows that energetic reasoning cuts already yield reasonable lower bounds for given makespan $C_{max} = \bar{s}_{|K|+1}$.

**Lemma 5.1.** *Let $\mu$ denote a feasible solution to* (5.3)*,* (5.4) *and* (5.6)*. If $C_{max} < \max_{r\in R} \sum_{i \in I} \frac{d_{ir}p_i}{D_r}$ then it holds $\sum_{k\in K} \alpha_k > 0$.*

*Proof.* Summing up (5.6) over all $k \in K$ yields for all $r \in R$

$$\begin{aligned}
\sum_{i \in I} d_{ir}p_i &\overset{(5.3)}{\leq} \sum_{k \in K}\sum_{i \in I_k} d_{ir}\mu_{ik} \\
&\overset{(5.6)}{\leq} D_r \sum_{k \in K}(\bar{s}_{k+1} - \bar{s}_k + \alpha_k) \leq D_r(C_{max} + \sum_{k \in K} \alpha_k) \\
&\iff \sum_{i \in I} \frac{d_{ir}p_i}{D_r} - C_{max} \leq \sum_{k \in K} \alpha_k
\end{aligned}$$

The last inequality implies $0 < \sum_{k \in K} \alpha_k$. Consequently, the makespan $C_{max}$ is invalid. $\square$

### Cover Cuts

As defined in Section 4, consider minimal covers $C \in \mathcal{C}_k$ for position $k \in K$ where $\mathcal{C}_k = \{C \in \mathcal{C} \mid C \subseteq I_k\}$ is the set of possible minimal covers at position $k$. Since only $|C| - 1$ jobs of $C$ can be scheduled in parallel, the *cover cuts* are given by

$$\sum_{i \in C} \mu_{ik} - \alpha_k \leq (|C| - 1)(\bar{s}_{k+1} - \bar{s}_k) \qquad \forall k \in K, C \in \mathcal{C}_k \tag{5.7}$$

This can be seen as energetic reasoning cuts where the resource demand of each $i \in C$ is one and the capacity is $|C| - 1$. This illustration motivates our later generalization in Section 5.2.

### Clique Cuts

Let $\mathcal{C}^*$ denote the set of cliques in the disjunctive graph $G^* = (I, E^*)$ with $(i, j) \in E^*$ if jobs $i$ and $j$ cannot be scheduled in parallel, especially if $(i, j) \in E$. Similarly, let $\mathcal{C}^*_k = \{C^* \in \mathcal{C}^* \mid C^* \subseteq I_k\}$ denote the set of cliques at interval $k \in K$. The *clique cuts* are given by the inequalities

$$\sum_{i \in C^*} \mu_{ik} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K, C^* \in \mathcal{C}^*_k \tag{5.8}$$

which counts the total duration of all $i \in C^*$ at interval $k \in K$ because all jobs in $\mathcal{C}^*$ must appear in sequence.

### Redundant Function Cuts

Redundant functions are strongly related to *dual-feasible functions*, see Fekete and Schepers [33] and Clautiaux et al. [35] for a survey. It was brought into the context of the RCPSP by Carlier and Néron [23] and studied extensively in Carlier and Néron [34]. Formally, a dual-feasible function is a function $f : [0, D] \mapsto [0, D']$ that satisfies

$$\sum_{i \in I'} \delta_i \leq D \implies \sum_{i \in I'} f(\delta_i) \leq f(D)$$

for a subset $I' \subseteq I$ and $D, D' \in \mathbb{N}$. A redundant function is a discrete dual-feasible function. If we consider redundant functions in terms of an energetic reasoning constraint (5.6) for some resource $r \in R$, we get new inequalities of the form

$$\begin{aligned} & \sum_{i \in I'} d_{ir} \mu_{ik} - \alpha_k \leq D_r(\bar{s}_{k+1} - \bar{s}_k) \\ \Longrightarrow & \sum_{i \in I'} f(d_{ir}) \mu_{ik} - \alpha_k \leq f(D_r)(\bar{s}_{k+1} - \bar{s}_k) \qquad \forall k \in K \end{aligned} \tag{5.9}$$

Note that redundant functions are *value-dependent*, that is they neither depend on the jobs or the resources. But for a proper redundant function $f$ the scalarization of the coefficients may yield stronger inequalities. Since redundant functions are discrete mappings, there are finitely many of them. In [34] they presented a full characterization and a method to compute all maximal and non-dominated redundant functions for small values of $D'$. In particular, not all valid cuts can be generated by maximal redundant functions.

### Precedence Cut 1

Similar to the disaggregated precedence constraints of the previous sections we derive precedence cuts as follows:

$$\sum_{k'=1}^{k} \frac{\mu_{jk'}}{p_j} + \sum_{k'=k+1}^{|K|} \frac{\mu_{ik'}}{p_i} \leq 1 \qquad \forall (i,j) \in E, k \in K \tag{5.10}$$

In contrast to the MIP modeling, $(i,j) \in E$ can both be active in the same interval.

### Precedence Cut 2

The following precedence-based cuts were originally introduced by Baptiste and Demassey [24] for the model of Brucker and Knust. But due to the later transformation (5.14) it is also valid for this model. Let $m_i \geq 0$ denote the midpoint when job $i \in I$ is active. It must hold

$$\frac{p_j + p_i}{2} \leq m_j - m_i \qquad \forall (i,j) \in E. \tag{5.11}$$

If job $i \in I$ starts in interval $k$ then $m_i$ is at least $\bar{s}_k + \frac{1}{2}$ and at most $\bar{s}_k - \frac{1}{2}$. Therefore, the following cuts are valid:

$$\sum_{k \in K_i} \left( \bar{s}_k + \frac{1}{2} \right) \cdot \frac{\mu_{ik}}{p_i} \leq m_i \leq \sum_{k \in K_i} \left( \bar{s}_{k+1} - \frac{1}{2} \right) \cdot \frac{\mu_{ik}}{p_i} \qquad \forall i \in I. \tag{5.12}$$

It measures the 'average' starting times with respect to the lower and upper time bounds $\bar{s}_k$ and $\bar{s}_{k+1}$.

### Incompatible Intervals

The next cuts are based on the observation that a job $i \in I$ cannot simultaneously be active in intervals $[\bar{s}_k, \bar{s}_{k+1}]$ and $[\bar{s}_l, \bar{s}_{l+1}]$ with $\bar{s}_{k+1} + p_i - 1 \leq \bar{s}_l$. Let $\mathcal{K}_i^{inc} \subseteq 2^K$ be a set of incompatible intervals for job $i \in I$. Then the inequalities

$$\sum_{k \in K'} \frac{\mu_{ik}}{\bar{s}_{k+1} - \bar{s}_k} \leq 1 \qquad \forall i \in I, K' \in \mathcal{K}_i^{inc} \tag{5.13}$$

restrict the total portion of job $i$ to one on these intervals. Since the general model has a polynomial number of variables, maximum violated incompatible intervals are well-suited for separation. Haouari et al. [25] identified the separating sets $K'$ as maximum weighted cliques in the position graph with edges between any two positions $(k, l)$ with $\bar{s}_{k+1} + p_i - 1 \leq \bar{s}_l$.

## 5.2 Primal-Dual Cutting Planes

In this section we compare the two LP models that of last section in terms of their induced polyhedra. A first apparent but important observation is the following linear transformation between the models:

$$\mu_{ik} = \sum_{F \in \mathcal{F}_k : i \in F} \xi_{Fk} \qquad \forall i \in I, k \in K_i. \tag{5.14}$$

In other words, the duration of job $i \in I$ in interval $k \in K_i$ is equal to the total duration of all job subsets $F \in \mathcal{F}_k$ that contain job $i$. This transformation reveals that the model of Brucker and Knust has the stronger relaxation since any inequality involving $\mu_{ik}$ variables can be expressed by the $\xi_{Fk}$ variables. However, the model has exponentially many variables. We deduce the following corollary.

**Corollary 5.2.** *The model of Brucker and Knust has the stronger LP-relaxation if all cuts* (5.5)-(5.13) *are added.*

But more than that. In the following we show that the cutting planes (5.6)-(5.9) are implied by the feasible subset model. In exchange, we are interested in the type of cuts that we would have to add to the model of Carlier and Néron in order to get the same bound as in the exponential model. For this, we propose a general characterization of cutting planes of the form

$$\sum_{i \in I_k} \delta_{ik}\mu_{ik} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K$$

with valid coefficients $\tilde{d}_{ik} \geq 0$ for all $i \in I$ and $k \in K_i$. Note that inequalities (5.6)-(5.9) are of this type. It turns out that such cuts originate from the dual LP of Brucker and Knust. From that, we deduce a separation algorithm that approximates the bound of Brucker and Knust.

The basic idea is to incorporate the $\mu_{ik}$ variables into the model of Brucker and Knust while preserving the transformation. Therefore, we rewrite their model by decomposing inequality (5.1) which yields:

$$\min \sum_{k \in K} \alpha_k$$

$$\sum_{F \in \mathcal{F}_k : i \in F} \xi_{Fk} \geq \mu_{ik} \qquad \forall i \in I, k \in K_i \tag{5.15}$$

$$\sum_{k \in K_i} \mu_{ik} \geq p_i \qquad \forall i \in I \tag{5.16}$$

$$\sum_{F \in \mathcal{F}_k} \xi_{Fk} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K \tag{5.17}$$

$$\xi_{Fk} \geq 0 \qquad \forall k \in K, F \in \mathcal{F}_k$$

$$\mu_{ik} \geq 0 \qquad \forall i \in I, k \in K_i.$$

By summing up (5.15) over all $k \in K_i$ we get

$$\sum_{k \in K_i} \sum_{F \in \mathcal{F}_k : i \in F} \xi_{Fk} \geq \sum_{k \in K_i} \mu_{ik} \geq p_i \qquad \forall i \in I$$

which again reproduces the original inequality. Note that in (5.15) we can assume an inequality since any optimal LP solution expects equality which maintains the transformation (5.14). Consequently, the model is equivalent. In the following we construct a cutting plane scheme. For this, introduce dual variables $\beta_i, \gamma_k, \delta_{ik} \geq 0$ and state the dual LP as follows:

$$\begin{aligned}
\max \sum_{i\in I} \beta_i p_i - \sum_{k\in K} \gamma_k(\bar{s}_{k+1} - \bar{s}_k) \\
\sum_{i\in F} \delta_{ik} - \gamma_k \leq 0 &\qquad \forall k \in K, F \in \mathcal{F}_k &\qquad (5.18)\\
\gamma_k \leq 1 &\qquad \forall k \in K &\qquad (5.19)\\
\beta_i - \delta_{ik} \leq 0 &\qquad \forall i \in I, k \in K_i &\qquad (5.20)\\
\delta_{ik} \geq 0 &\qquad \forall i \in I, k \in K_i\\
\gamma_k \geq 0 &\qquad k \in K\\
\beta_i \geq 0 &\qquad i \in I.
\end{aligned}$$

Note that the dual can be decomposed for all $k \in K$. The main theorem states as follows.

**Theorem 5.3.** *Given solution vectors $\xi, \mu, \alpha \geq 0$ and $\delta, \gamma, \beta \geq 0$ to the primal and dual LP, the inequality $\sum_{i\in I_k} \delta_{ik}\mu_{ik} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k$ is valid for every $k \in K$.*

*Proof.* Assume $\xi, \mu, \alpha$ and $\delta, \gamma, \beta$ are feasible solutions to the primal and dual LP. Then it holds

$$\begin{aligned}
\sum_{i\in I_k} \delta_{ik}\mu_{ik} &\overset{(5.15)}{\leq} \sum_{i\in I_k} \delta_{ik} \sum_{F\in\mathcal{F}_k : i\in F} \xi_{Fk} = \sum_{F\in\mathcal{F}_k} \xi_{Fk} \sum_{i\in F} \delta_{ik}\\
&\overset{(5.18)}{\leq} \sum_{F\in\mathcal{F}_k} \xi_{Fk}\gamma_k \overset{(5.19)}{\leq} \sum_{F\in\mathcal{F}_k} \xi_{Fk} \overset{(5.17)}{\leq} \bar{s}_{k+1} - \bar{s}_k + \alpha_k
\end{aligned}$$

which shows the statement. □

**Theorem 5.4.** *Given optimal solution vectors $\xi^*, \mu^*, \alpha^* \geq 0$ and $\delta^*, \gamma^*, \beta^* \geq 0$ to the primal and dual LP, it holds $\sum_{i\in I_k} \delta^*_{ik}\mu^*_{ik} - \alpha^*_k = \gamma^*_k(\bar{s}_{k+1} - \bar{s}_k)$ for each $k \in K$.*

*Proof.* By complementary slackness it must hold

$$\delta^*_{ik}\Big(\sum_{F\in\mathcal{F}_k:i\in F}\xi^*_{Fk}-\mu^*_{ik}\Big)=0 \qquad \forall i\in I, k\in K_i \tag{5.21}$$

$$\xi^*_{Fk}\Big(\gamma^*_k-\sum_{i\in F}\delta^*_{ik}\Big)=0 \qquad \forall k\in K, F\in\mathcal{F}_k \tag{5.22}$$

$$\gamma^*_k\Big(\bar{s}_{k+1}-\bar{s}_k+\alpha^*_k-\sum_{F\in\mathcal{F}_k}\xi^*_{Fk}\Big)=0 \qquad \forall k\in K \tag{5.23}$$

$$\alpha^*_k(1-\gamma^*_k)=0 \qquad \forall k\in K \tag{5.24}$$

This implies

$$\begin{aligned}\sum_{i\in I_k}\delta^*_{ik}\mu^*_{ik} &\overset{(5.21)}{=} \sum_{i\in I_k}\delta^*_{ik}\sum_{F\in\mathcal{F}_k:i\in F}\xi^*_{Fk}=\sum_{F\in\mathcal{F}_k}\xi^*_{Fk}\sum_{i\in F}\delta^*_{ik}\\ &\overset{(5.22)}{=} \sum_{F\in\mathcal{F}_k}\xi^*_{Fk}\gamma^*_k\\ &\overset{(5.23)}{=} \gamma^*_k(\bar{s}_{k+1}-\bar{s}_k+\alpha^*_k)\\ &\overset{(5.24)}{=} \gamma^*_k(\bar{s}_{k+1}-\bar{s}_k)+\alpha^*_k\end{aligned}$$

which yields the result. □

**Theorem 5.5.** *Let $k\in K$ and $\sum_{i\in F}\delta_{ik}\le 1$ for all $F\in\mathcal{F}_k$ with $\delta_{ik}\ge 0$ for all $i\in I_k$. Then the inequality $\sum_{i\in I_k}\delta_{ik}\mu_{ik}-\alpha_k\le\bar{s}_{k+1}-\bar{s}_k$ is valid.*

*Proof.* In the dual constraints (5.18) and (5.19) we set $\gamma_k=1$ without loss of generality. Therefore, the values $\delta_{ik}$ are dual-feasible by (5.18) and by Theorem 5.3 the inequality is valid. □

Theorem 5.5 gives a characterization of valid cuts for the LP of Carlier and Néron from a dual interpretation of the LP of Brucker and Knust, see Figure 5.1. In the next section we derive a separation algorithm in order to strengthen the LP bound of Carlier and Néron. Furthermore, note that the model still relaxes the precedence and non-preemptivity conditions. The next example shows where generated cuts are not maximal with respect to the general RCPSP.

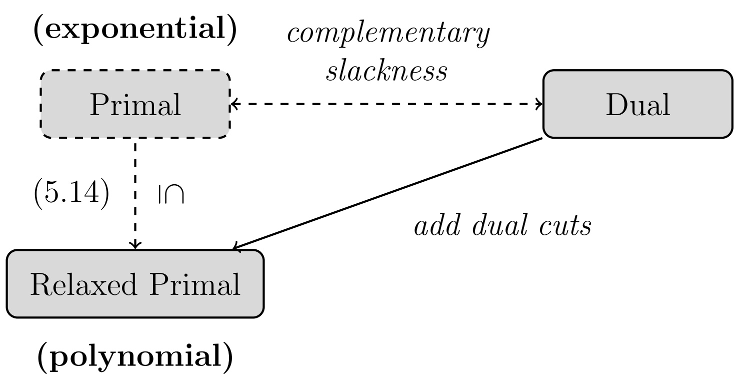



**Figure 5.1:** Adding dual cuts from separation

**Example 5.6.** *Assume intervals $K = \{1, 2, 3\}$ and resources $|R| = 1$ with $D = 2$. Furthermore, let $I = \{i_1, i_2, i_3\}$ with $p_i = 2$ and $d_i = 1$ for all $i \in I$ and define interval starting times $s_1 = 0, s_2 = 1, s_3 = 2, s_4 = 3$ where $s_4$ is the considered makespan. The feasible subsets are $\{1, 2\}$, $\{1, 3\}, \{2, 3\}$ which yields the maximum valid inequalities*

$$\begin{array}{rrrrl}
\mu_{1k} & & & -\alpha_k \leq & \bar{s}_{k+1} - \bar{s}_k \\
 & \mu_{2k} & & -\alpha_k \leq & \bar{s}_{k+1} - \bar{s}_k \\
 & & \mu_{3k} & -\alpha_k \leq & \bar{s}_{k+1} - \bar{s}_k \\
0.5\mu_{1k} & +0.5\mu_{2k} & +0.5\mu_{3k} & -\alpha_k \leq & \bar{s}_{k+1} - \bar{s}_k
\end{array}$$

*for each $k \in K$ and $\mu_{ik} \geq 0$ for all $i \in I, k \in K$. The corresponding LP has the optimal solution $\mu_{11} = \mu_{12} = \mu_{22} = \mu_{23} = \mu_{31} = \mu_{33} = 1$ and zero otherwise, see Figure 5.2. This yields an objective value of $\sum_{k \in K} \alpha_k = 0$. But in the general RCPSP there is no feasible solution to this instance, since one job must be split. The considered cutting planes use only local information at each $k \in K$ and do not take into account non-preemtivity or precedence constraints on subsets $K' \subseteq K$. This is the remaining weakness of these cuts.*

Example 5.6 shows that for the general RCPSP these cuts are not sufficient. From our experience it seems hard to find stronger cutting planes without computing real subinstances of the RCPSP which makes a separation approach hard to deal with. Discretization may help here, but the ultimate goal would be to integrate non-preemtivity and precedences into cutting planes that can be computed efficiently.

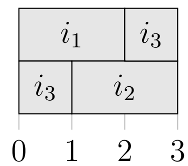



**Figure 5.2:** Example where the dual cuts are not sufficient to prove infeasibility.

Theorem 5.5 gives rise to use a separation routine for the LP model of Carlier and Néron. But first we show that a number of proposed cutting planes of Section 5.1.2 are implied by Theorem 5.5.

### Implied Inequalities

We show that the stated inequalities (5.6)-(5.9) are dominated by inequalities obtained from Theorem 5.5. In particular, these inequalities are of the form

$$\sum_{i \in I_k} \delta_{ik} \mu_{ik} \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K.$$

Therefore, we have to guarantee that the coefficients satisfy

$$\sum_{i \in F} \delta_{ik} \leq 1 \qquad \forall k \in K, F \in \mathcal{F}_k. \tag{5.25}$$

In the following we show this for each of the stated inequalities.

**Corollary 5.7.** *The coefficients of the trivial job cuts* (5.4) *satisfy* (5.25).

*Proof.* For given $i \in I$ set $\delta_{ik} = 1$ for all $k \in K$ and zero otherwise. Then

$$\sum_{i \in F} \delta_{ik} \leq 1 \qquad \forall k \in K, F \in \mathcal{F}_k$$

which proves the statement. □

**Corollary 5.8.** *The coefficients of the energetic cuts* (5.6) *satisfy* (5.25).

*Proof.* Let $r \in R$ and set $\delta_{ik} = \frac{d_{ir}}{D_r}$ for all $i \in I, k \in K$. Then it holds

$$\sum_{i \in F} \delta_{ik} = \sum_{i \in F} \frac{d_{ir}}{D_r} \leq 1 \qquad \forall k \in K, F \in \mathcal{F}_k$$

by definition of a feasible subset. □

**Corollary 5.9.** *The coefficients of the cover cuts* (5.7) *satisfy* (5.25)*.*

*Proof.* For given $k \in K$ and minimal cover $C \in \mathcal{C}_k$ set $\delta_{ik} = \frac{1}{|C|-1}$ for all $i \in C$ and $\delta_{jk} = 0$ for all $j \in I \setminus C$. Then for any feasible subset $F \in \mathcal{F}_k$ it holds

$$\sum_{i \in F} \delta_{ik} \leq \sum_{i \in F} \frac{1}{|C|-1} \leq 1$$

since $|F| \leq |C| - 1$ by definition of a minimal cover. □

**Corollary 5.10.** *The coefficients of the clique cuts* (5.8) *satisfy* (5.25)*.*

*Proof.* For given $k \in K$ and clique $C^* \in \mathcal{C}_k^*$ in the disjunctive graph set $\delta_{ik} = 1$ for all $i \in C^*$ and $\delta_{jk} = 0$ for all $j \in I \setminus C^*$. Since for each feasible subset $F \in \mathcal{F}_k$ and $i, j \in C^*$ with $i \neq j$ it holds $|\{i, j\} \cap F| \leq 1$, we get

$$\sum_{i \in F} \delta_{ik} \leq 1.$$

□

**Corollary 5.11.** *The coefficients of the redundant function cuts* (5.9) *satisfy* (5.25)*.*

*Proof.* Let $r \in R$ be a resource and $f$ be a redundant function as defined in (5.9). Setting $\delta_{ik} = \frac{f(d_{ir})}{f(D_r)}$ for all $i \in I, k \in K$ yields

$$\sum_{i \in F} \delta_{ik} = \sum_{i \in F} \frac{f(d_{ir})}{f(D_r)} \leq 1 \qquad \forall k \in K, F \in \mathcal{F}_k$$

by definition of a redundant function. □

In the following we give an example that shows the strengths of the inequality coefficients compared to the maximum valid cuts of Theorem 5.5.

**Example 5.12.** *Given jobs $I = \{1, 2, 3, 4, 5\}$ with resource demands $d = (3, 5, 8, 9, 11)$ and capacity $D = 15$. The minimal covers are $\{1, 2, 3\}$, $\{1, 2, 4\}$, $\{2, 5\}$, $\{3, 4\}$, $\{3, 5\}$, $\{4, 5\}$. The non-dominated cliques are $\{3, 4, 5\}$ and $\{2, 5\}$. As redundant function we take, for example, $f(3) = 0$, $f(5) = 1$, $f(8) = f(9) = 2$, $f(11) = 3$ and $f(15) = 3$. For every $k \in K$ we get the following inequalities:*

*Energetic cut*

$$0.2\mu_{1k} \quad +0.33\mu_{2k} \quad +0.53\mu_{3k} \quad +0.6\mu_{4k} \quad +0.73\mu_{5k} \quad \leq \quad \bar{s}_{k+1} - \bar{s}_k$$

*Cover cuts*

$$\begin{array}{ccccccc}
0.5\mu_{1k} & +0.5\mu_{2k} & +0.5\mu_{3k} & & & \leq & \bar{s}_{k+1} - \bar{s}_k \\
0.5\mu_{1k} & +0.5\mu_{2k} & & +0.5\mu_{4k} & & \leq & \bar{s}_{k+1} - \bar{s}_k \\
& \mu_{2k} & & & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k \\
& & \mu_{3k} & +\mu_{4k} & & \leq & \bar{s}_{k+1} - \bar{s}_k \\
& & \mu_{3k} & & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k \\
& & & \mu_{4k} & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k
\end{array}$$

*Clique cuts*

$$\begin{array}{cccccc}
& \mu_{3k} & +\mu_{4k} & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k \\
\mu_{2k} & & & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k
\end{array}$$

*Redundant function cut*

$$0.33\mu_{2k} \quad +0.66\mu_{3k} \quad +0.66\mu_{4k} \quad +\mu_{5k} \quad \leq \quad \bar{s}_{k+1} - \bar{s}_k$$

*Maximum valid cuts from* (5.25)

$$\begin{array}{ccccccc}
0.5\mu_{1k} & +0.5\mu_{2k} & +0.5\mu_{3k} & +0.5\mu_{4k} & +0.5\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k \\
& & \mu_{3k} & +\mu_{4k} & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k \\
& \mu_{2k} & & & +\mu_{5k} & \leq & \bar{s}_{k+1} - \bar{s}_k \\
\mu_{1k} & & & & & \leq & \bar{s}_{k+1} - \bar{s}_k
\end{array}$$

In Example 5.12 every cut can be generated from maximum valid cuts by taking a proper convex combination. Since maximal redundant functions are monotonically increasing, see [34], the third inequality of the maximum valid cuts would not be obtainable by a maximal redundant function. However, maximal redundant functions may be computed more efficiently. Figure 5.3 gives a further polyhedral view on the different cutting planes.

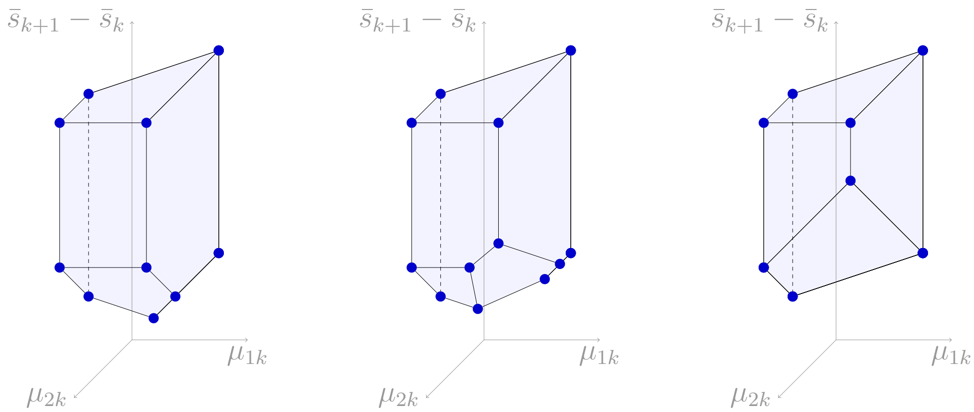



**Figure 5.3:** Further example for two jobs with $p = (3, 5)$, $d = (5, 4)$ and $D = 6$. From left to right: (a) trivial cuts, (b) energetic cut, (c) with cover/clique/maximal valid cut (same in this example)

## 5.3 Two Cutting Plane Algorithms

In the following we introduce two approximative approaches to compute lower bounds for the RCPSP. The first is based on the model of Carlier and Néron, as presented in Section 5.1.2. We use our knowledge of Section 5.2 in order to compute maximum violated cuts at each interval $k \in K$. The second algorithm is based on the transformation equalities (5.14). We take the cuts (5.5)-(5.12) originally provided for the model of Carlier and Néron and integrate them into the model of Brucker and Knust. The resulting linear program is solved by column generation or, equivalently, by separation in the dual linear program.

### 5.3.1 Dual Cut Approximation Algorithm

Consider the LP model of Carlier and Néron as introduced in Section 5.1.2:

$$\begin{aligned}
\min \sum_{k \in K} \alpha_k & & \\
\sum_{k \in K_i} \mu_{ik} &\geq p_i & \forall i \in I \\
\mu_{ik} - \alpha_k &\leq \bar{s}_{k+1} - \bar{s}_k & \forall i \in I, k \in K_i \\
\mu_{ik} &\geq 0 & \forall i \in I, k \in K_i \\
\alpha_k &\geq 0 & k \in K
\end{aligned}$$

Due to Theorem 5.5 we construct separating cutting planes of the form

$$\sum_{i \in I_k} \delta_{ik} \mu_{ik} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K.$$

In particular, given an optimal solution $\mu^*$ to the above LP we look for coefficients $\delta_{ik}$ such that $\sum_{i \in I_k} \delta_{ik} \mu^*_{ik} - \alpha_k > \bar{s}_{k+1} - \bar{s}_k$. By Theorem 5.5 we solve the following separation problem which we simply call the 'subproblem':

$$\begin{aligned}
\max \sum_{k \in K} \sum_{i \in I_k} \delta_{ik} \mu^*_{ik} & & \\
\sum_{i \in F} \delta_{ik} &\leq 1 & \forall k \in K, F \in \mathcal{F}_k \\
\delta_{ik} &\geq 0 & \forall k \in K, i \in I_k.
\end{aligned}$$

**Algorithm 3:** Solve relaxed LP

**Input**: instance of RCPSP, fixed $C_{max}$
**Output**: Is there a feasible LP-solution?

$N_k \leftarrow \emptyset \quad \forall k \in K$
$N \leftarrow \bigcup_{k \in K} N_k$
$\beta \leftarrow true$
**while** $\beta = true$ **do** `// solve LP and look for cuts`
  $\beta \leftarrow false$
  $(\mu^*, \alpha^*) \leftarrow solveLP(N)$
  **for** $k \in K$ **do** `// solve subproblem for all` $k \in K$
    $\delta_k \leftarrow solveSub(\mu^*, k)$
    **if** $\sum_{i \in I} \delta_{ik} \mu^*_{ik} - \alpha^*_k > \bar{s}_{k+1} - \bar{s}_k$ **then** `// add inequality`
      $N_k \leftarrow N_k \cup \{\delta\}$
      $\beta \leftarrow true$
  $\beta \leftarrow \beta \wedge \sum_{k \in K} \alpha_k = 0$
**return** $\phi^* = 0$

Note that this LP decomposes for all $k \in K$. If the optimal solution $\delta^*$ of the LP satisfies $\sum_{i \in I_k} \delta^*_{ik} \mu^*_{ik} > \bar{s}_{k+1} - \bar{s}_k$ for some position $k \in K$ the cut $\sum_{i \in I_k} \delta^*_{ik} \mu_{ik} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k$ is added to the LP. This procedure is repeated until either $\sum_{k \in K} \alpha_k > 0$ or no further violated cutting plane is found, compare Algorithm 3 for a complete description.

In particular, the separation subproblem has exponentially many constraints, hence it is also solved by separation. For given $k \in K$ and current solution $\delta_{ik}$, we look for a feasible subset $F \in F_k$ with $\sum_{i \in F_k} \delta_{ik} > 1$. Therefore, we solve a MDKC for each $k \in K$ by the following MIP:

$$
\begin{aligned}
\max \ & \sum_{i \in I} \delta_{ik} x_i & \\
& \sum_{i \in I_k} d_{ir} x_i \leq D_r & \forall r \in R \\
& x_i + x_j \leq 1 & \forall (i,j) \in I_k^2 \cap E \\
& x_i \in \{0,1\} & \forall i \in I_k
\end{aligned}
$$

If the optimal solution $x^*$ satisfies $\sum_{i \in I} \delta_{ik} x^*_i > 1$, the cut $\sum_{i \in I} \delta_{ik} x^*_i \leq 1$ is added to the subproblem. Otherwise, if there exists no improving cut for the

**Algorithm 4:** *solveSub*

**Input**: LP solution $\mu^*$, position $k \in K$
**Output**: maximal feasible coefficients $\delta_i, i \in I$

$\mathcal{F}' \leftarrow \bigcup_{i \in I}\{i\}$
$\phi \leftarrow \infty$
**while** $\phi > 1$ **do** `// look for separating cut`
  $\delta \leftarrow solveSubLP(\mu^*, \mathcal{F}')$
  $(\phi, x) \leftarrow solveMDKC(\delta, k)$
  **if** $\phi > 1$ **then** `// if violated, add cut`
    $\mathcal{F}' \leftarrow \mathcal{F}' \cup \{i \in I \mid x_i = 1\}$
**return** $\delta$

subproblem then the current solution of the subproblem is optimal, compare Algorithm 4. The entire approach relies on double-separation. At first glance, this seems computationally intractable. But since knapsack problems are routine for modern MIP solvers, their computation is fast in practice.

### 5.3.2 Model Extension

Basically, this section extends the model of Brucker and Knust [22] by additional cuts that were introduced in Section 5.1.2 originally for the model of Carlier an Néron. The transformation (5.14) allows us to integrate them into the exponential model. Therefore, recall the basic model of Brucker and Knust as introduced in Section 5.1.1 and the cutting planes of Section 5.1.2.

The extended model of Brucker and Knust states as follows:

$$\min \sum_{k \in K} \alpha_k$$

$$\sum_{\substack{F \in \mathcal{F}_k \\ i \in F}} \xi_{Fk} \geq p_i \qquad \forall i \in I \tag{5.26}$$

$$\sum_{F \in \mathcal{F}_k} \xi_{Fk} - \alpha_k \leq \bar{s}_{k+1} - \bar{s}_k \qquad \forall k \in K \tag{5.27}$$

$$\sum_{\substack{F \in \mathcal{F}_k \\ i \in F}} \xi_{Fk} \geq p(i,k) \qquad \forall i \in I, k \in K_i \tag{5.28}$$

$$\sum_{k'=1}^{k} \left( \sum_{\substack{F \in \mathcal{F}_{k'} \\ i \in F}} \frac{\xi_{Fk'}}{p_i} - \sum_{\substack{F \in \mathcal{F}_{k'} \\ j \in F}} \frac{\xi_{Fk'}}{p_j} \right) \geq 0 \qquad \forall (i,j) \in E, k \in K \tag{5.29}$$

$$m_i - \sum_{k \in K} \sum_{\substack{F \in \mathcal{F}_k \\ i \in F}} \left( s_k + \frac{1}{2} \right) \frac{\xi_{Fk}}{p_i} \geq 0 \qquad \forall i \in I \tag{5.30}$$

$$\sum_{k \in K} \sum_{\substack{F \in \mathcal{F}_k \\ i \in F}} \left( s_{k+1} - \frac{1}{2} \right) \frac{\xi_{Fk}}{p_i} - m_i \geq 0 \qquad \forall i \in I \tag{5.31}$$

$$m_j - m_i \geq \frac{p_i + p_j}{2} \qquad \forall (i,j) \in E \tag{5.32}$$

$$\sum_{k \in K'} \sum_{\substack{F \in \mathcal{F}_k \\ i \in F}} \frac{\xi_{Fk}}{\bar{s}_{k+1} - \bar{s}_k} \leq 1 \qquad \forall i \in I, K' \in \mathcal{K}_i^{inc} \tag{5.33}$$

$$\xi_{Fk} \geq 0 \qquad \forall k \in K, F \in \mathcal{F}_k$$
$$\alpha_k \geq 0 \qquad \forall k \in K$$
$$m_i \geq 0 \qquad \forall i \in I$$

Constraints (5.26) and (5.27) are adopted from the standard model. Inequalities (5.28) correspond to the energetic reasoning constraints and (5.29) are precedence cuts type 1. Additionally, (5.30)-(5.32) are the precedence constraints type 2. Finally, (5.33) are the incompatible position subsets cuts.

For the generation of the sets $\mathcal{K}_i^{inc}$ we adopted the heuristic of Baptiste and Demassey [24] and extended it by consideration of additional step sizes.

In order compare this approach with the algorithm of Section 5.3.1 in terms of lower bounds, we formulated the dual of the above linear program and solved it by extensive separation. As given in Brucker and Knust [22], the pricing/separation problem consists of solving a MDKC. In our implementation we solved a MDKC at each position $k \in K$ and, if violated, added the corresponding cut to the LP. This allows us to add more than one violated cutting plane and hence to speed-up the computation.

## 5.4 Linear Extension for MIP

In Section 5.1.2 the linear programming model of Carlier and Néron was introduced to compute lower bounds for the RCPSP. This linear program has strong structural similarities to the compact MIP models of Chapter 2. In the following we transfer the same bounds to our compact MIP setting. Therefore, recall the MIP models OOE, SEE and DP. In contrast to previous sections of this chapter, we define the set $K = \{1, ..., n\}$ as originally introduced in Chapter 2. Hence, also the interval starting times $s_k \geq 0$ with $k \in K$ are variables again.

The MIP models OOE, SEE and DP include the constraints (2.8), (2.17) and (2.33) which couple the starting times $s_k$ with the positional assignments of the jobs. In this approach, we replace each of these constraints by the following inequalities that were taken from Section 5.1.2:

$$\sum_{k \in K} \mu_{ik} \geq p_i \qquad \forall i \in I \tag{5.34}$$

$$\mu_{ik} \leq s_{k+1} - s_k \qquad \forall i \in I, k \in K \tag{5.35}$$

$$\mu_{ik} \geq 0 \qquad \forall i \in I, k \in K.$$

As before, the variables $\mu_{ik}$ denote the partial duration of job $i \in I$ in interval $k \in K$. Now the $\mu$ variables are coupled with the decision variables each compact MIP model.

### Coupling Inequalities

In each of the models OOE, SEE and DP the coupling is based on the same correlation: if job $i \in I$ is partially assigned to interval $k \in K$, then it must be active in interval $k$. Therefore, the coupling inequalities are obtained analogous to the transformations of Chapter 3:

$$\mu_{ik} \leq p_i u_{ik} \qquad \forall i \in I, k \in K \tag{OOE}$$

$$\mu_{ik} \leq p_i \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \right) \qquad \forall i \in I, k \in K \tag{SEE}$$

$$\mu_{ik} \leq p_i \sum_{(k',l') \in A_k} z_{ik'l'} \qquad \forall i \in I, k \in K \tag{DP}$$

Since every job $i \in I$ has its duration $p_i$ distributed to intervals, the fractional $\mu_{ik}$ variables imply binary expressions. From a polyhedral point of view this seems weak at first glance. But due to the new correlation to the $s_k$ variables, even the current formulation yields decent bounds. However, the main benefit of this linear extension emerges from strong valid cutting planes involving the $\mu$ variables. A first example is given by the energetic reasoning cuts as introduced in Section 5.1.2:

$$\sum_{i \in I} \frac{d_{ir}}{D_r} \mu_{ik} \leq s_{k+1} - s_k \qquad \forall k \in K, r \in R. \tag{5.36}$$

**Theorem 5.13.** *Let $B = \max_{r \in R} \frac{d_{ir} p_i}{D_r}$ and assume an instance of RCPSP with $E = \emptyset$ and $p_i \leq \frac{B-2}{2}$ for all $i \in I$. Furthermore, assume the linear extension of the models OOE, SEE and DP given by* (5.34)-(5.36). *For the optimal values of the LP-relaxation it holds:*

$$OPT_{LP}(DDT) \leq OPT_{LP}(OOE) \leq OPT_{LP}(SEE) \leq OPT_{LP}(DP).$$

*Proof.* Summing up (5.36) for all $k \in K$ gives $\sum_{i \in I} \frac{d_{ir} p_i}{D_r} \leq s_{n+1}$ for all $r \in R$, thus $B \leq s_{n+1}$ in each of the models OOE, SEE and DP. By Proposition 2.1 we have $OPT_{LP}(DDT) \leq B$. Moreover, Theorem 3.13 and Theorem 3.14 yield $B \leq OPT_{LP}(OOE) \leq OPT_{LP}(SEE) \leq OPT_{LP}(DP)$ since the coupling inequalities preserve the used transformations between the models. Consequently, the statement is valid. □

The proposition reveals that without precedence constraints the compact models have provably better LP bounds than DDT. The condition $p_i \leq \frac{B-2}{2}$ for all $i \in I$ is satisfied in almost every practical instance. Moreover, energetic reasoning cuts are by far not the strongest cuts, see the implied inequalities of Section 5.2. For more sophisticated tightening we use Theorem 5.5 and compute the corresponding cuts with respect to random search directions. This is furthermore helpful to keep track of the model size and to maintain the compactness property of the models.

### Randomized Cut Generation

We follow the scheme of Section 5.2 and look for cutting planes of the form

$$\sum_{i \in I} \delta_{ik} \mu_{ik} \leq s_{k+1} - s_k \qquad \forall k \in K$$

for valid coefficients $\delta_{ik}$ with $i \in I$ and $k \in K$. that are not yet determined. It suggests to compute maximum valid coefficients $\delta_{ik}$ by solving the LP

$$\begin{aligned} \max \sum_{i \in I} \sum_{k \in K} \delta_{ik} \lambda_{ik} & & \\ \sum_{i \in F} \delta_{ik} \leq 1 & \qquad \forall k \in K, F \in \mathcal{F}_k \\ \delta_{ik} \geq 0 & \qquad \forall i \in I, k \in K \end{aligned} \tag{5.37}$$

for objective coefficients $\lambda_{ik} \in \mathbb{R}_{\geq 0}$. Note that the LP can be decomposed for all $k \in K$. In particular, we generate the values $\lambda_{ik} \in [0, 1]$ randomly with respect to a uniform distribution, see Algorithm 5. This has several reasons.

First, an exact separation approach, as in Section 5.3.1, requires too many cuts in order to solve the MIP efficiently. Moreover, a small number of cuts suffices to generate very reasonable lower bounds. Furthermore, we intend to maintain the compactness of the models, so the number of random cuts is chosen constant. Finally, randomization appears to be a valuable approach in this context. On average, the probability is high that an optimal solution has high coefficients for mutually exclusive jobs. This includes, for example, jobs that belong to a precedence path. On the contrary, there is a small probability that a cut has high coefficients, for instance, for two jobs that cannot be scheduled in parallel according to resource restrictions. If many jobs are involved, it is indeed unlikely that the LP-relaxation violates such a cut. Consequently, the randomized cuts have high coefficients for job subsets that certainly induce a conflict when assigned to the same interval.

**Algorithm 5:** Generate randomized cutting planes

**Input**: $H \in \mathbb{Z}_{\geq 0}$ number of cutting planes

**Output**: random cuts $N_k$, $k \in K$

$N_k \leftarrow \emptyset \quad \forall k \in K$

**for** $h = 1, ..., H$ **do** `// generate random cut`

  $\lambda = (\lambda_i)_{i \in I} \leftarrow \text{uniform}(0, 1)$

  $\delta \leftarrow solveSub(\lambda)$

  **for** $k \in K$ **do** `// add cut to all` $k \in K$

    $N_k \leftarrow N_k \cup \{\delta\}$

**return** $N \leftarrow \bigcup_{k \in K} N_k$

Let $\delta^*$ is an optimal solution to the stated linear program. Then the cut

$$\sum_{i \in I} \delta^*_{ik} \mu_{ik} \leq s_{k+1} - s_k \qquad \forall k \in K$$

is added to the MIP model.

The cut generation approach provides a synergy effect. The coefficients $\delta_{ik}$ from (5.37) also strengthen the integer part of the model. In particular, they correspond to feasible solutions of the polar, or anti-blocking respectively, polytope of the induced knapsack sub-polytope at every position $k \in K$. Therefore, the coefficients are valid for the binary activity variables at each $k \in K$. That is, given values $\delta_{ik} \geq 0$ for all $i \in I$, $k \in K$ that satisfy (5.37) the following inequalities are valid:

$$\sum_{i \in I} \delta_{ik} u_{ik} \leq 1 \tag{OOE}$$

$$\sum_{i \in I} \delta_{ik} \left( \sum_{k'=1}^{k} x_{ik'} - \sum_{k'=2}^{k} y_{ik'} \right) \leq 1 \tag{SEE}$$

$$\sum_{i \in I} \sum_{(k',l') \in A_k} \delta_{ik} z_{ik'l'} \leq 1 \tag{DP}$$

Note that, in combination with these cuts, the coefficients from (5.37) generalize the minimal cover inequalities of Section 4. We consider the same knapsack subpolytope at each position $k \in K$. Since (5.37) includes the coefficients of all valid inequalities of the subpolytope, it also contains the coefficients of the lifted cover cuts. Consequently, the randomization approach may yield stronger inequalities than the lifted cover inequalities of Section 4.

# Chapter 6

# Computational Results

This chapter presents the results of our computational study concerning the different models of this thesis. As testing environment we used the problem instances of the PSPLIB [65]. It includes RCPSP instances that involve 30-120 jobs and 4 resources with different scales of precedence and resource strength. First, we evaluate the DDT model and the compact MIP models and compare them with regard to the LP-relaxation, as well as lower and upper bounds after MIP solving. Subsequently, we discuss the two linear programming algorithms of Section 5.3.

Our computations were executed on a 3.50GHz Intel Xeon CPU with 16GB RAM. For LP and MIP solving we used the commercial solver CPLEX version 12.6.

## 6.1 MIP Models

The time-indexed model DDT and the compact models OOE, SEE, HPF and DP were implemented. For SEE and HPF we used the revised version, as introduced in Chapter 2. The performance of the models was tested on the J30 test set which contains 480 instances, each including 30 jobs.

Our first attempts showed that the HPF model is not suited for the J30 instances. We neither get acceptable lower bounds or integral solutions in reasonable time. It is rather applicable for instances with 10-15 jobs. Because of the weak performance, we only refer to OOE, SEE and DP in the following.

### LP-Relaxation

If we consider the compact models in their standard version of Chapter 2 the optimal value of the LP-relaxation is weak in all compact models, especially in OOE where it is close to zero. It is not exactly zero, as stated in Proposition 2.7, because of the preprocessing step of Section 2.6. On average over all J30 instances, OOE attains 1.2%, SEE 38.72% and DP 38.72% of the LP-relaxation value of DDT. In other words, the LP-bound of DDT is more than twice the bound of SEE and DP. On most instances, the LP-relaxation value of SEE and DP is equal. This shows that the linear transformation of Chapter 3 is tight in practice. Moreover, MIP solving suffers from the weak bounds in the standard models. Our testings showed that the strengthening approaches of OOE and SEE could not contribute to the solving performance on the J30 instances. At this point, the linear extension and cutting plane approach of Section 5.4 is much more valuable.

According to Section 5.4, we computed 30 random cuts which are added to all positions $k \in K$ by decomposition. This approach drastically increases the value of the LP-relaxation, see Table 6.1. On average, the LP-relaxation value of SEE and DP gets larger by a factor of 2.78 compared to the previous bound. Naturally, the factor becomes very large for OOE since the standard LP-value is close to zero.

| | OOE | SEE | DP |
|---|---|---|---|
| min | 20.90 | 1.71 | 1.78 |
| max | 636.16 | 6.17 | 5.91 |
| average | 129.51 | 2.786 | 2.79 |

**Table 6.1:** Randomized cuts: average increase of the LP-relaxation value by factor

In the following we only refer to the extended compact models that include the randomized cutting planes. In this case, the compact models yield LP-bounds that are on average 5.6% larger than DDT. But we fairly have to distinguish between the different instances. On instances where DDT performs weak the LP-relaxation value of the compact models is about 27% larger, see Table C.1 in the appendix. This is explained by the weakness of DDT on cumulative instances that are rather independent of the precedence relations, see Theorem 5.13. In this context, the compact models yield considerably stronger LP-bounds. Note that in Table C.1 the optimal LP-values can deviate, since randomized cutting planes generate slightly different bounds. This does not violate the inclusions shown in Section 3.

| | OOE | | SEE | | DP | |
|---|---|---|---|---|---|---|
| | #inst. | %dev. | #inst. | %dev. | #inst. | %dev. |
| $>$ | 140 | 19.41% | 136 | 20.16% | 139 | 19.76% |
| $=$ | 273 | - | 266 | - | 270 | - |
| $<$ | 67 | -3.05% | 78 | -3.91% | 71 | -3.98% |

**Table 6.2:** First column: Number of instances of J30 with greater/equal/smaller LP-relaxation value than DDT. Second column: average % deviation from DDT.

### MIP Solving

The stronger LP bounds have a significant impact on MIP solving. The DDT model and the compact models have been tested on all 480 instances of J30 with a 300 second time limit. The scheduling horizon for DDT was set to $\mathcal{T} = \{1, ..., T\}$ with $T = \sum_{i \in I} p_i$. Table 6.3 shows the average size of the models without presolving. The number of binary variables for OOE and SEE is much less than for DDT, because it does not depend on $T$. Additionally, the number of constraints of the compact models is larger because additional relations concerning starting and completing a job have to be modeled. Since in DP, starting and completing a job is implied by the variables, the number of constraints is potentially smaller. In exchange, the number of variables increases. Note that if the $p_i$ values are scaled by a sufficiently large number, then the size of DDT becomes intractable for MIP solving. However, on the J30 instances this is not the case.

| | DDT | OOE | SEE | DP |
|---|---|---|---|---|
| # 0/1 variables | 9368 | 900 | 1800 | 13050 |
| # constraints | 4980 | 6240 | 8423 | 5670 |

**Table 6.3:** Average number of binary variables and constraints on the J30 instances

After MIP solving, OOE and SEE found the optimal solution nearly as often as DDT, but without optimality proof (remaining gap), see Table 6.4. In case of proving optimality, all compact models underlie DDT. The DP model has weaker results than OOE and SEE which comes mainly from the large number of variables, that is hard to deal with, already for $n = 30$.

| | DDT | OOE | SEE | DP |
|---|---|---|---|---|
| optimal + proof | 427 | 272 | 303 | 234 |
| optimal + no proof | 428 | 392 | 407 | 324 |

**Table 6.4:** Number of optimal instances of J30 (480 instances in total)

Tables 6.5 and 6.6 give an overview of the obtained lower and upper bounds after MIP solving compared to DDT. On most instances, the lower bounds of the compact models are weaker or equal to DDT. In case the lower bounds of the compact models are larger than DDT, then by 12-13% on average. This huge deviation is verified on the same hard cumulative instances of Table C.1. Otherwise, if the lower bounds are smaller, then about 7% on average. In addition, DDT is slightly better in finding solutions. If the solutions found by DDT are better then about 4-5% for OOE and SEE and 8-9% for DP. Otherwise, if the compact models provide better solutions, then their objective value is about 8% smaller. Again, this is observed on the hard instances of Table C.1. Similarly, the weakness of DP is due to the large number of variables. A complete statistic of MIP solving on the J30 instances is given in Table B.1 in the appendix.

| | OOE | | SEE | | DP | |
|---|---|---|---|---|---|---|
| | #inst. | %dev. | #inst. | %dev. | #inst. | %dev. |
| $>$ | 43 | 13.1% | 47 | 13.16% | 44 | 12.1% |
| $=$ | 266 | - | 294 | - | 231 | - |
| $<$ | 171 | -7.49% | 139 | -6.81% | 205 | -7.59% |

**Table 6.5:** First column: Number of instances of J30 with greater/equal/smaller lower bound than DDT after MIP solving. Second column: average % deviation to DDT.

Finally, the randomized cutting plane has been shown to have a significant impact on the lower bounds and the solving performance of the compact models. On average, DDT performs better on the comparatively easier instances which are stronger correlated to precedence constraints. On such instances, the compact models miss to close the remaining minor gap in order to prove optimality. Conversely, the compact models strictly outperform the DDT

| | OOE | | SEE | | DP | |
|---|---|---|---|---|---|---|
| | #inst. | %dev. | #inst. | %dev. | #inst. | %dev. |
| $<$ | 43 | -8.54% | 40 | -8.66% | 36 | -7.68% |
| $=$ | 375 | - | 387 | - | 310 | - |
| $>$ | 62 | 4.46% | 53 | 4.74% | 134 | 8.75% |

**Table 6.6:** First column: Number of instances of J30 with smaller/equal/greater upper bound than DDT after MIP solving. Second column: average % deviation to DDT.

model on highly cumulative instances. This originates from the strong cutting planes of Section 5.4. Among the compact models, SEE shows the best results followed by OOE and DP. We believe that this mainly comes from the sparse structure of the revised model, which is well-suited for modern MIP solvers. The OOE model benefits from the small number of variables, while the large number of variables is the drawback of DP. Since the cut generation is based on simple randomization there remains an unexplored potential in compact models to solve even harder instances.

## 6.2 Linear Programming Lower Bounds

In the following we analyze the two linear programming approaches of Section 5.3.1 and Section 5.3.2 to compute lower bounds for the RCPSP. Both approaches were tested on the J60 and J120 instances, each involving 60 or 120 jobs respectively. It turns out that the first approximative scheme of Section 5.3.1 cannot compete with the exponential LP of Section 5.3.2. On most of the J60 and J120 instances, the approximative dual-cut scheme of Section 5.3.1 does not terminate in a reasonable time limit. We strictly emphasize that the bottleneck is not the time needed to generate one violated cutting plane via double separation, which is quite decent. The difficulty of the approximation originates from polyhedral properties. In order to terminate the approximation there are far too many cuts required. This is caused by the polynomial representation of the model and the kind of separating cuts. They restrict the polytope on certain knapsack sub-structures at each interval $k \in K$. In contrast, the exponential model of Section 5.3.2 implies these cuts by taking feasible subset variables. This is also the reason why the exponential LP is favorable, since one pricing step in the exponential model includes various cuts of the weaker model. In addition, the pricing problem consists of solving only a MDKC in contrast to an exponential LP in the polynomial model. However, there might be one aspect where the first algorithm dominates. As it is a polynomial model it is suited for additional separation methods. In the exponential model it is hard to integrate both column generation and separation. For example, the incompatible interval cuts of Section 5.1.2 can be separated exactly in the first model, while the exponential model uses a heuristic which is less effective. But at the current state, the computational effort is too high to prefer it in terms of different separation techniques.

Finally, the exponential model was applicable for the J60 and J120 instances. The additional precedence cuts and the extended heuristic was already able to improve some of the best known lower bounds for the RCPSP, see Tables 6.7 and 6.8. As reference we considered the provided lower bounds of [25, 40, 65]. However, we have not added all possible cuts provided in [25]. Hence, we presumably receive even better lower bounds. As possible, we submit additional results in a postpone paper.

**Table 6.7:** New lower bounds of the J60 instances

| Inst. | new LB | increase |
|---|---|---|
| J6013_1 | 105 | +1 |
| J6029_3 | 115 | +1 |
| J6045_6 | 133 | +1 |
| J6045_9 | 115 | +1 |
| J6045_10 | 104 | +2 |

**Table 6.8:** New lower bounds of the J120 instances

| Inst. | new LB | increase |
|---|---|---|
| X34_3 | 99 | +1 |
| X52_10 | 131 | +1 |
| X53_2 | 109 | +1 |
| X53_10 | 124 | +1 |

# Chapter 7

# Conclusion

This thesis studied compact MIP models for the *Resource-Constrained Project Scheduling Problem* whose size is independent of the scheduling horizon. In addition to two compact models OOE and SEE of Koné et al. [8] we presented two novel compact MIP models DP and HPF. We showed that the models OOE, SEE and DP are equivalent up to linear transformations. The same holds for HPF and DPK, where DPK emerges from DP by adding a single constraint. With respect to linear transformations, we proved the following inclusions for the corresponding LP-relaxations:

$$P(DP) \xrightarrow{\subseteq} P(SEE) \xrightarrow{\subseteq} P(OOE) \quad \text{and} \quad P(DPK) \xrightarrow{\subseteq} P(HPF).$$

Furthermore, we examined *lifted cover inequalities* for the compact models and showed that general cover cuts are facet-defining for certain knapsack sub-polytopes of OOE, SEE and DP respectively. Sequential lifting was proposed to extend cover inequalities to facets of the initial polytope. A closer investigation revealed that sequential lifting can be done more efficiently, if the precedence constraints are relaxed. The resulting lifting algorithm runs in polynomial time and it is model-independent. Moreover, we addressed lower bounds of the RCPSP and compared two recent linear programming models. We identified a linear relation between these models and discovered a primal-dual characterization of valid cutting planes for one of the models. Two cutting plane algorithms are derived. Similar inequalities can be used for the compact MIP models. We recommend randomization for their computation.

Our computational study demonstrates that, in the standard formulation, the compact models are not competitive with DDT in any setting. In contrast, adding the randomized cuts yields a very significant increase of the LP bounds. In MIP solving, the DDT models are preferred on comparatively easier disjunctive instances that are majorly determined by the precedence graph. On harder cumulative instances the compact models are highly preferred. Eventually, the compact models are also preferred when the scheduling horizon is large. Concerning the two linear programming algorithms, the polynomial model cannot compete with the exponential model because the number of cuts needed to achieve the same bound is too large. Minor modifications on the exponential model increased the best known lower bounds on a few problem instances. Finally, the exponential LP model asks for stronger cutting planes. In particular, it is searched for an efficient way to exactly integrate non-preemtivity and precedences into the exponential model by means of cutting planes or other approaches.

Finally, we make a statement regarding the long-term suitability of compact MIP models for exact solving the RCPSP. This thesis highlighted strong properties of the compact models. Especially on more difficult cumulative instances our approach suggests more sophisticated cut generation within an LP environment. But finally, we guess that MIP in general is not the best alternative for solving the RCPSP in a compact manner. In LP terms, compact modeling admits weak correlations between resource, time and decisional aspects of the RCPSP, even if the LP bounds is strong. In the future, MIP is not expected to fill this space. In fact, a very promising direction would be to incorporate the compact setting into a constraint programming framework. Since some current CP approaches use time-discretization the search space appears to be much smaller in a non-redundant compact representation of the problem. In combination with strong conflict analysis, compact CP models may become an alternative of growing importance to solve the RCPSP.

# Bibliography


[1] MOPTA 2013 competition website: http://coral.ie.lehigh.edu/ mopta2013/competition

[2] AIMMS competition website: http://www.aimms.com/team-oropt-has-won-2013-aimms-mopta-modeling-competition/

[3] Blazewicz, J., Lenstra, J. K., Kan, A. R. (1983). Scheduling subject to resource constraints: classification and complexity. Discrete Applied Mathematics, 5(1), 11-24.

[4] Schrijver, A. (2003). Combinatorial optimization: polyhedra and efficiency (Vol. 24). Springer Science & Business Media.

[5] Ziegler, G. M. (1995). Lectures on polytopes (Vol. 152). Springer Science & Business Media.

[6] Borndörfer, R., Weismantel, R. (2001). Discrete relaxations of combinatorial programs. Discrete applied mathematics, 112(1), 11-26.

[7] Christofides, N., Alvarez-Valdés, R., Tamarit, J. M. (1987). Project scheduling with resource constraints: A branch and bound approach. European Journal of Operational Research, 29(3), 262-273.

[8] Koné, O., Artigues, C., Lopez, P., Mongeau, M. (2011). Event-based MILP models for resource-constrained project scheduling problems. Computers & Operations Research, 38(1), 3-13.

[9] Artigues, C., Brucker, P., Knust, S., Koné, O., Lopez, P., Mongeau, M. (2013). A note on 'event-based MILP models for resource-constrained project scheduling problems'. Computers Operations Research, 40(4), 1060-1063.

[10] Artigues, C. (2013). A note on time-indexed formulations for the resource-constrained project scheduling problem.

[11] J. C. Zapata, B. M. Hodge, and G. V. Reklaitis, The multimode resource constrained multiproject scheduling problem: Alternative formulations", AIChE Journal, 54(8):2101-2119, 2008.

[12] Pritsker A, Watters L (1968) A zero-one programming approach to scheduling with limited resourcse. The RAND Corporation, RM-5561-PR.

[13] Mingozzi, A., Maniezzo, V., Ricciardelli, S., Bianco, L. (1998). An exact algorithm for the resource-constrained project scheduling problem based on a new mathematical formulation. Management Science, 44(5), 714-729.

[14] Olaguibel, R. A. V., Goerlich, J. T. (1993). The project scheduling polyhedron: dimension, facets and lifting theorems. European Journal of Operational Research, 67(2), 204-220.

[15] Patterson, J. H., Roth, G. W. (1976). Scheduling a project under multiple resource constraints: a zero-one programming approach. AIIE transactions, 8(4), 449-455.

[16] Stinson, J. P., Davis, E. W., Khumawala, B. M. (1978). Multiple resource–constrained scheduling using branch and bound. AIIE Transactions, 10(3), 252-259.

[17] Artigues, C., Michelon, P., Reusser, S. (2003). Insertion techniques for static and dynamic resource-constrained project scheduling. European Journal of Operational Research, 149(2), 249-267.

[18] Möhring, R. H., Schulz, A. S., Stork, F., Uetz, M. (2003). Solving project scheduling problems by minimum cut computations. Management Science, 49(3), 330-350.

[19] Möhring, R. H., Schulz, A. S., Stork, F., Uetz, M. (2001). On project scheduling with irregular starting time costs. Operations Research Letters, 28(4), 149-154.

[20] Chaudhuri, S., Walker, R. A., Mitchell, J. E. (1994). Analyzing and exploiting the structure of the constraints in the ILP approach to the scheduling problem. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, 2(4), 456-471.

[21] Bianco, L., Caramia, M. (2013). A new formulation for the project scheduling problem under limited resources. Flexible Services and Manufacturing Journal, 25(1-2), 6-24.

[22] Brucker, P., Knust, S. (2000). A linear programming and constraint propagation-based lower bound for the RCPSP. European Journal of Operational Research, 127(2), 355-362.

[23] Carlier, J., Néron, E. (2003). On linear lower bounds for the resource constrained project scheduling problem. European Journal of Operational Research, 149(2), 314-324.

[24] Baptiste, P., Demassey, S. (2004). Tight LP bounds for resource constrained project scheduling. OR Spectrum, 26(2), 251-262.

[25] Haouari, M., Kooli, A., Néron, E., Carlier, J. (2014). A preemptive bound for the resource constrained project scheduling problem. Journal of Scheduling, 17(3), 237-248.

[26] Haouari, M., Kooli, A., Néron, E. (2012). Enhanced energetic reasoning-based lower bounds for the resource constrained project scheduling problem. Computers Operations Research, 39(5), 1187-1194.

[27] Schulz, J. (2012). Hybrid Solving Techniques for Project Scheduling Problems (Doctoral dissertation, Universitätsbibliothek der Technischen Universität Berlin).

[28] Demeulemeester, E., Herroelen, W. (1992). A branch-and-bound procedure for the multiple resource-constrained project scheduling problem. Management science, 38(12), 1803-1818.

[29] Dorndorf, U., Huy, T. P., Pesch, E. (1999). A survey of interval capacity consistency tests for time-and resource-constrained scheduling. In Project Scheduling (pp. 213-238). Springer US.

[30] Klein, R., Scholl, A. (1999). Computing lower bounds by destructive improvement: An application to resource-constrained project scheduling. European Journal of Operational Research, 112(2), 322-346.

[31] Baptiste, P., Le Pape, C., Nuijten, W. (1999). Satisfiability tests and time-bound adjustments for cumulative scheduling problems. Annals of Operations research, 92, 305-333.

[32] Östergard, P. R. (2001). A new algorithm for the maximum-weight clique problem. Nordic Journal of Computing, 8(4), 424-436.

[33] Fekete, S. P., Schepers, J. (2001). New classes of fast lower bounds for bin packing problems. Mathematical programming, 91(1), 11-31.

[34] Carlier, J., Neron, E. (2007). Computing redundant resources for the resource constrained project scheduling problem. European Journal of Operational Research, 176(3), 1452-1463.

[35] Clautiaux, F., Alves, C., de Carvalho, J. V. (2010). A survey of dual-feasible and superadditive functions. Annals of Operations Research, 179(1), 317-342.

[36] Schutt, A., Feydy, T., Stuckey, P. J., Wallace, M. G. (2009). Why cumulative decomposition is not as bad as it sounds. In Principles and Practice of Constraint Programming-CP 2009 (pp. 746-761). Springer Berlin Heidelberg.

[37] Schutt, A., Feydy, T., Stuckey, P. J., Wallace, M. G. (2013). Solving RCPSP/max by lazy clause generation. Journal of Scheduling, 16(3), 273-289.

[38] Schutt, A., Feydy, T., Stuckey, P. J., Wallace, M. G. (2011). Explaining the cumulative propagator. Constraints, 16(3), 250-282.

[39] Baptiste, P., Le Pape, C. (1997). Constraint propagation and decomposition techniques for highly disjunctive and highly cumulative project scheduling problems. In Principles and Practice of Constraint Programming-CP97 (pp. 375-389). Springer Berlin Heidelberg.

[40] Vilim, P. (2011). Timetable edge finding filtering algorithm for discrete cumulative resources. In Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (pp. 230-245). Springer Berlin Heidelberg.

[41] Gabrel, V., Minoux, M. (2002). A scheme for exact separation of extended cover inequalities and application to multidimensional knapsack problems. Operations Research Letters, 30(4), 252-264.

[42] Kaparis, K., Letchford, A. N. (2008). Local and global lifted cover inequalities for the 0–1 multidimensional knapsack problem. European journal of operational research, 186(1), 91-103..

[43] Pferschy, U., Schauer, J. (2009). The Knapsack Problem with Conflict Graphs. J. Graph Algorithms Appl., 13(2), 233-249.

[44] Bettinelli, A., Cacchiani, V., Malaguti, E. (2014). Bounds and algorithms for the knapsack problem with conflict graph. Technical Report OR-14-16, DEIS–University of Bologna, Bologna, Italy.

[45] Yamada, T., Kataoka, S., Watanabe, K. (2002). Heuristic and exact algorithms for the disjunctively constrained knapsack problem. Information Processing Society of Japan Journal, 43(9).

[46] Hifi, M. (2014). An iterative rounding search-based algorithm for the disjunctively constrained knapsack problem. Engineering Optimization, 46(8), 1109-1122.

[47] Hifi, M., Michrafy, M. (2006). A reactive local search-based algorithm for the disjunctively constrained knapsack problem. Journal of the Operational Research Society, 57(6), 718-726.

[48] Hifi, M., Michrafy, M. (2007). Reduction strategies and exact algorithms for the disjunctively constrained knapsack problem. Computers & operations research, 34(9), 2657-2673.

[49] Hifi, M., Otmani, N. (2012). An algorithm for the disjunctively constrained knapsack problem. International Journal of Operational Research, 13(1), 22-43.

[50] Balas, E. (1975). Facets of the knapsack polytope. Mathematical Programming, 8(1), 146-164.

[51] Balas, E., Zemel, E. (1978). Facets of the knapsack polytope from minimal covers. SIAM Journal on Applied Mathematics, 34(1), 119-148.

[52] Zemel, E. (1978). Lifting the facets of zero-one polytopes. Mathematical Programming, 15(1), 268-277.

[53] Padberg, M. W. (1975). Technical Note: A Note on Zero-One Programming. Operations Research, 23(4), 833-837.

[54] Padberg, M. W. (1973). On the facial structure of set packing polyhedra. Mathematical programming, 5(1), 199-215.

[55] Nemhauser, G. L., Trotter Jr, L. E. (1974). Properties of vertex packing and independence system polyhedra. Mathematical Programming, 6(1), 48-61.

[56] Wolsey, L. A. (1976). Technical Note: Facets and Strong Valid Inequalities for Integer Programs. Operations Research, 24(2), 367-372.

[57] Wolsey, L. A. (1975). Faces for a linear inequality in 0–1 variables. Mathematical Programming, 8(1), 165-178.

[58] Gu, Z., Nemhauser, G. L., Savelsbergh, M. W. (1999). Lifted cover inequalities for 0-1 integer programs: Complexity. INFORMS Journal on Computing, 11(1), 117-123.

[59] Cavalcante, C. C., De Souza, C. C., Savelsbergh, M. W., Wang, Y., Wolsey, L. A. (2001). Scheduling projects with labor constraints. Discrete Applied Mathematics, 112(1), 27-52.

[60] Hardin, J. R., Nemhauser, G. L., Savelsbergh, M. W. (2008). Strong valid inequalities for the resource-constrained scheduling problem with uniform resource requirements. Discrete Optimization, 5(1), 19-35.

[61] Stork, F., Uetz, M. (2005). On the generation of circuits and minimal forbidden sets. Mathematical programming, 102(1), 185-203.

[62] Belov, G., Kartak, V. M., Rohling, H., Scheithauer, G. (2013). Conservative scales in packing problems. OR spectrum, 35(2), 505-542.

[63] de Carvalho, J. V. (2002). LP models for bin packing and cutting stock problems. European Journal of Operational Research, 141(2), 253-273.

[64] De Loera, J. A., Onn, S. (2006). All linear and integer programs are slim 3-way transportation programs. SIAM Journal on Optimization, 17(3), 806-821.

[65] Kolisch, R., Sprecher, A. (1997). PSPLIB-a project scheduling problem library: OR software-ORSEP operations research software exchange program. European journal of operational research, 96(1), 205-216.

# Appendix A

# Revised HPF Model

$$\min \; s_{n+1}$$

$$\sum_{k \in K} x_{ik} = 1 \qquad \forall i \in I \tag{A.1}$$

$$\sum_{i \in I} x_{ik} = 1 \qquad \forall k \in K \tag{A.2}$$

$$w_{kn+1} = 1 \qquad \forall k \in K \tag{A.3}$$

$$s_k - s_l + p_i(x_{ik} + w_{kl} - 1) \leq 0 \qquad \forall i \in I, (k,l) \in A \tag{A.4}$$

$$s_k - s_{k+1} \leq 0 \qquad \forall k \in K \tag{A.5}$$

$$\sum_{i \in I} d_{ir} x_{ik} - f_{kn+1r} = 0 \qquad \forall k \in K, r \in R \tag{A.6}$$

$$\sum_{(k,l) \in A_h} (f_{kl+1r} - f_{klr}) \leq D_r \qquad \forall h \in K, r \in R \tag{A.7}$$

$$f_{klr} - \bar{d}_r w_{kl} \leq 0 \qquad \forall (k,l) \in A, r \in R \tag{A.8}$$

$$w_{kl} - w_{kl+1} \leq 0 \qquad \forall (k,l) \in A : l < n+1 \tag{A.9}$$

$$f_{klr} - f_{kl+1r} \leq 0 \qquad \forall r \in R, (k,l) \in A : l < n+1 \tag{A.10}$$

$$x_{ik} - w_{kl-1} + \sum_{h=l}^{n} x_{ih} + \sum_{h=1}^{l-1} x_{jh} \leq 1 \qquad \forall (i,j) \in E, (k,l) \in A \tag{A.11}$$

$$x_{ik} \in \{0,1\} \qquad \forall i \in I, k \in K$$

$$w_{kl} \in \{0,1\} \qquad \forall (k,l) \in A$$

$$f_{klr} \geq 0 \qquad \forall (k,l) \in A, r \in R$$

$$s_k \geq 0 \qquad \forall k \in K \cup \{n+1\}$$

# Appendix B

# MIP Solutions

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J301_1 | 43 | 43 | 43 | 5 | 38 | 43 | 300 | 43 | 43 | 71 | 38 | 43 | 300 |
| J301_2 | 47 | 47 | 47 | 5 | 42 | 47 | 300 | 45 | 47 | 300 | 42 | 47 | 300 |
| J301_3 | 47 | 47 | 47 | 3 | 43 | 47 | 300 | 47 | 47 | 67 | 43 | 47 | 300 |
| J301_4 | 62 | 62 | 62 | 17 | 56 | 62 | 300 | 56.5 | 62 | 300 | 56 | 62 | 300 |
| J301_5 | 39 | 39 | 39 | 5 | 36.14 | 39 | 300 | 37 | 39 | 300 | 35 | 40 | 300 |
| J301_6 | 48 | 48 | 48 | 8 | 42 | 48 | 300 | 42 | 48 | 300 | 42 | 49 | 300 |
| J301_7 | 60 | 60 | 60 | 3 | 60 | 60 | 1 | 60 | 60 | 2 | 60 | 60 | 6 |
| J301_8 | 53 | 53 | 53 | 4 | 53 | 53 | 5 | 53 | 53 | 4 | 53 | 53 | 32 |
| J301_9 | 49 | 49 | 49 | 8 | 45 | 50 | 300 | 46.5 | 49 | 300 | 45 | 49 | 300 |
| J301_10 | 45 | 45 | 45 | 5 | 41 | 46 | 300 | 44 | 45 | 300 | 41 | 47 | 300 |
| J302_1 | 38 | 38 | 38 | 4 | 34 | 38 | 300 | 37 | 38 | 300 | 34 | 38 | 300 |
| J302_2 | 51 | 51 | 51 | 5 | 46 | 51 | 300 | 48 | 51 | 300 | 46 | 51 | 300 |
| J302_3 | 43 | 43 | 43 | 4 | 43 | 43 | 3 | 43 | 43 | 13 | 43 | 43 | 205 |
| J302_4 | 43 | 42 | 43 | 3 | 43 | 43 | 3 | 43 | 43 | 3 | 43 | 43 | 23 |
| J302_5 | 51 | 51 | 51 | 4 | 51 | 51 | 2 | 51 | 51 | 3 | 51 | 51 | 15 |
| J302_6 | 47 | 47 | 47 | 3 | 47 | 47 | 1 | 47 | 47 | 3 | 47 | 47 | 14 |
| J302_7 | 47 | 47 | 47 | 4 | 47 | 47 | 2 | 47 | 47 | 2 | 47 | 47 | 54 |
| J302_8 | 54 | 54 | 54 | 5 | 50 | 54 | 300 | 54 | 54 | 37 | 50 | 54 | 300 |
| J302_9 | 54 | 54 | 54 | 5 | 51 | 54 | 300 | 51 | 54 | 300 | 51 | 54 | 300 |
| J302_10 | 43 | 43 | 43 | 5 | 42.5 | 43 | 300 | 42.5 | 43 | 300 | 40.5 | 45 | 300 |
| J303_1 | 72 | 72 | 72 | 5 | 72 | 72 | 2 | 72 | 72 | 1 | 72 | 72 | 3 |
| J303_2 | 40 | 40 | 40 | 3 | 40 | 40 | 242 | 40 | 40 | 55 | 39 | 40 | 300 |
| J303_3 | 57 | 57 | 57 | 3 | 57 | 57 | 2 | 57 | 57 | 2 | 57 | 57 | 4 |
| J303_4 | 98 | 98 | 98 | 4 | 98 | 98 | 0 | 98 | 98 | 2 | 98 | 98 | 4 |
| J303_5 | 53 | 53 | 53 | 6 | 46 | 53 | 300 | 53 | 53 | 189 | 43 | 53 | 300 |
| J303_6 | 54 | 54 | 54 | 4 | 54 | 54 | 15 | 54 | 54 | 2 | 53 | 54 | 300 |
| J303_7 | 48 | 48 | 48 | 3 | 48 | 48 | 2 | 48 | 48 | 3 | 48 | 48 | 3 |
| J303_8 | 54 | 54 | 54 | 3 | 54 | 54 | 1 | 54 | 54 | 1 | 54 | 54 | 6 |
| J303_9 | 65 | 65 | 65 | 3 | 60 | 65 | 300 | 65 | 65 | 109 | 60 | 65 | 300 |
| J303_10 | 59 | 59 | 59 | 4 | 59 | 59 | 181 | 59 | 59 | 30 | 57 | 59 | 300 |
| J304_1 | 49 | 49 | 49 | 3 | 49 | 49 | 2 | 49 | 49 | 1 | 49 | 49 | 90 |
| J304_2 | 60 | 60 | 60 | 4 | 60 | 60 | 1 | 60 | 60 | 3 | 60 | 60 | 5 |
| J304_3 | 47 | 47 | 47 | 3 | 47 | 47 | 3 | 47 | 47 | 3 | 47 | 47 | 19 |
| J304_4 | 57 | 57 | 57 | 4 | 57 | 57 | 1 | 57 | 57 | 1 | 57 | 57 | 4 |
| J304_5 | 59 | 59 | 59 | 4 | 59 | 59 | 2 | 59 | 59 | 2 | 59 | 59 | 4 |
| J304_6 | 45 | 45 | 45 | 4 | 41 | 45 | 300 | 45 | 45 | 102 | 41 | 45 | 300 |
| J304_7 | 56 | 56 | 56 | 6 | 56 | 56 | 2 | 56 | 56 | 2 | 56 | 56 | 9 |
| J304_8 | 55 | 55 | 55 | 4 | 55 | 55 | 1 | 55 | 55 | 2 | 55 | 55 | 2 |
| J304_9 | 38 | 38 | 38 | 3 | 38 | 38 | 10 | 38 | 38 | 5 | 38 | 38 | 300 |
| J304_10 | 48 | 48 | 48 | 3 | 48 | 48 | 2 | 48 | 48 | 2 | 48 | 48 | 6 |
| J305_1 | 53 | 53 | 53 | 16 | 44 | 53 | 300 | 51 | 53 | 300 | 44 | 59 | 300 |
| J305_2 | 82 | 82 | 82 | 253 | 63 | 83 | 300 | 68 | 83 | 300 | 62 | 83 | 300 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J305_3 | 76 | 76 | 76 | 126 | 72 | 76 | 300 | 74 | 76 | 300 | 72 | 85 | 300 |
| J305_4 | 63 | 63 | 63 | 197 | 52 | 63 | 300 | 52 | 65 | 300 | 52 | 66 | 300 |
| J305_5 | 76 | 76 | 76 | 39 | 64 | 76 | 300 | 75 | 76 | 300 | 65 | 76 | 300 |
| J305_6 | 64 | 64 | 64 | 45 | 61 | 70 | 300 | 64 | 67 | 300 | 61.49 | 73 | 300 |
| J305_7 | 76 | 76 | 76 | 207 | 73 | 76 | 300 | 73 | 78 | 300 | 73 | 79 | 300 |
| J305_8 | 67 | 67 | 67 | 38 | 57 | 67 | 300 | 57 | 68 | 300 | 55.33 | 77 | 300 |
| J305_9 | 49 | 49 | 49 | 8 | 36.5 | 50 | 300 | 37.52 | 50 | 300 | 35.5 | 74 | 300 |
| J305_10 | 70 | 70 | 70 | 66 | 59 | 71 | 300 | 59 | 70 | 300 | 59 | 76 | 300 |
| J306_1 | 59 | 59 | 59 | 10 | 54 | 60 | 300 | 54 | 59 | 300 | 54 | 63 | 300 |
| J306_2 | 51 | 51 | 51 | 7 | 46 | 52 | 300 | 50 | 51 | 300 | 46 | 52 | 300 |
| J306_3 | 48 | 48 | 48 | 5 | 45 | 49 | 300 | 45 | 49 | 300 | 45 | 49 | 300 |
| J306_4 | 42 | 42 | 42 | 9 | 36 | 42 | 300 | 36 | 42 | 300 | 36 | 48 | 300 |
| J306_5 | 67 | 67 | 67 | 10 | 57 | 67 | 300 | 59 | 67 | 300 | 57 | 67 | 300 |
| J306_6 | 37 | 37 | 37 | 4 | 35 | 37 | 300 | 35 | 37 | 300 | 35 | 38 | 300 |
| J306_7 | 46 | 46 | 46 | 4 | 43 | 46 | 300 | 44 | 46 | 300 | 43 | 47 | 300 |
| J306_8 | 39 | 39 | 39 | 4 | 39 | 41 | 300 | 39 | 39 | 293 | 39 | 44 | 300 |
| J306_9 | 51 | 51 | 51 | 3 | 51 | 51 | 131 | 51 | 51 | 9 | 48 | 51 | 300 |
| J306_10 | 61 | 61 | 61 | 10 | 58 | 61 | 300 | 58 | 61 | 300 | 58 | 67 | 300 |
| J307_1 | 55 | 55 | 55 | 3 | 55 | 55 | 2 | 55 | 55 | 2 | 55 | 55 | 7 |
| J307_2 | 42 | 42 | 42 | 3 | 42 | 42 | 6 | 42 | 42 | 5 | 42 | 42 | 94 |
| J307_3 | 42 | 42 | 42 | 5 | 39 | 42 | 300 | 40 | 42 | 300 | 39 | 42 | 300 |
| J307_4 | 44 | 44 | 44 | 3 | 44 | 44 | 109 | 44 | 44 | 6 | 44 | 44 | 225 |
| J307_5 | 44 | 44 | 44 | 7 | 41 | 44 | 300 | 42 | 44 | 300 | 41 | 45 | 300 |
| J307_6 | 35 | 35 | 35 | 3 | 35 | 35 | 2 | 35 | 35 | 2 | 35 | 35 | 25 |
| J307_7 | 50 | 50 | 50 | 4 | 50 | 50 | 24 | 50 | 50 | 16 | 50 | 51 | 300 |
| J307_8 | 44 | 44 | 44 | 5 | 43 | 44 | 300 | 43 | 44 | 300 | 43 | 44 | 300 |
| J307_9 | 60 | 60 | 60 | 3 | 56 | 60 | 300 | 60 | 60 | 86 | 56 | 60 | 300 |
| J307_10 | 49 | 49 | 49 | 4 | 49 | 49 | 127 | 49 | 49 | 38 | 45 | 49 | 300 |
| J308_1 | 44 | 44 | 44 | 3 | 44 | 44 | 1 | 44 | 44 | 2 | 44 | 44 | 11 |
| J308_2 | 51 | 51 | 51 | 4 | 51 | 51 | 1 | 51 | 51 | 4 | 51 | 51 | 10 |
| J308_3 | 53 | 53 | 53 | 4 | 53 | 53 | 3 | 53 | 53 | 1 | 53 | 53 | 5 |
| J308_4 | 48 | 48 | 48 | 3 | 48 | 48 | 2 | 48 | 48 | 2 | 48 | 48 | 15 |
| J308_5 | 58 | 58 | 58 | 4 | 58 | 58 | 2 | 58 | 58 | 2 | 58 | 58 | 7 |
| J308_6 | 47 | 47 | 47 | 5 | 47 | 47 | 4 | 47 | 47 | 2 | 47 | 47 | 23 |
| J308_7 | 41 | 41 | 41 | 3 | 41 | 41 | 3 | 41 | 41 | 2 | 41 | 41 | 18 |
| J308_8 | 51 | 51 | 51 | 3 | 51 | 51 | 2 | 51 | 51 | 2 | 51 | 51 | 23 |
| J308_9 | 39 | 39 | 39 | 4 | 39 | 39 | 4 | 39 | 39 | 3 | 39 | 39 | 21 |
| J308_10 | 67 | 67 | 67 | 3 | 67 | 67 | 1 | 67 | 67 | 1 | 67 | 67 | 5 |
| J309_1 | 83 | 83 | 83 | 155 | 80 | 83 | 300 | 81 | 83 | 300 | 80 | 84 | 300 |
| J309_2 | 92 | 70.58 | 98 | 300 | 90 | 92 | 300 | 92 | 92 | 45 | 90 | 92 | 300 |
| J309_3 | 68 | 63.72 | 74 | 300 | 63 | 71 | 300 | 65 | 74 | 300 | 63 | 78 | 300 |
| J309_4 | 71 | 68 | 72 | 300 | 62.25 | 75 | 300 | 61.33 | 76 | 300 | 63.25 | 79 | 300 |
| J309_5 | 70 | 70 | 70 | 51 | 62 | 70 | 300 | 66 | 70 | 300 | 62 | 73 | 300 |
| J309_6 | 59 | 59 | 59 | 159 | 48.67 | 64 | 300 | 49.71 | 69 | 300 | 49.12 | 63 | 300 |
| J309_7 | 63 | 56 | 65 | 300 | 51 | 71 | 300 | 49 | 72 | 300 | 50.5 | 74 | 300 |
| J309_8 | 91 | 81 | 97 | 300 | 81 | 92 | 300 | 81 | 93 | 300 | 81 | 93 | 300 |
| J309_9 | 63 | 54 | 74 | 300 | 50.5 | 67 | 300 | 50.5 | 69 | 300 | 51 | 77 | 300 |
| J309_10 | 88 | 83 | 88 | 300 | 82 | 88 | 300 | 88 | 88 | 230 | 76.09 | 92 | 300 |
| J3010_1 | 42 | 42 | 42 | 5 | 41 | 42 | 300 | 41 | 42 | 300 | 41 | 50 | 300 |
| J3010_2 | 56 | 56 | 56 | 28 | 52 | 57 | 300 | 52 | 57 | 300 | 52 | 61 | 300 |
| J3010_3 | 62 | 62 | 62 | 18 | 61 | 63 | 300 | 61 | 64 | 300 | 61 | 69 | 300 |
| J3010_4 | 58 | 58 | 58 | 24 | 53 | 59 | 300 | 53 | 59 | 300 | 53 | 63 | 300 |
| J3010_5 | 41 | 41 | 41 | 6 | 35 | 45 | 300 | 41 | 41 | 230 | 41 | 46 | 300 |
| J3010_6 | 44 | 44 | 44 | 8 | 40 | 44 | 300 | 40 | 45 | 300 | 40 | 53 | 300 |
| J3010_7 | 49 | 49 | 49 | 7 | 47 | 49 | 300 | 47 | 49 | 300 | 47 | 53 | 300 |
| J3010_8 | 54 | 54 | 54 | 9 | 50 | 54 | 300 | 50 | 55 | 300 | 50 | 62 | 300 |
| J3010_9 | 49 | 49 | 49 | 3 | 49 | 49 | 1 | 49 | 49 | 4 | 49 | 49 | 10 |
| J3010_10 | 41 | 40 | 41 | 8 | 37 | 41 | 300 | 37 | 43 | 300 | 37 | 51 | 300 |
| J3011_1 | 54 | 54 | 54 | 6 | 52 | 54 | 300 | 52 | 54 | 300 | 52 | 59 | 300 |
| J3011_2 | 56 | 56 | 56 | 7 | 56 | 58 | 300 | 56 | 56 | 66 | 56 | 58 | 300 |
| J3011_3 | 81 | 81 | 81 | 4 | 81 | 81 | 1 | 81 | 81 | 1 | 81 | 81 | 3 |
| J3011_4 | 63 | 63 | 63 | 6 | 63 | 63 | 220 | 60 | 63 | 300 | 60 | 63 | 300 |
| J3011_5 | 49 | 49 | 49 | 7 | 48 | 52 | 300 | 48 | 51 | 300 | 48 | 53 | 300 |
| J3011_6 | 44 | 44 | 44 | 5 | 44 | 44 | 7 | 44 | 44 | 30 | 44 | 44 | 184 |
| J3011_7 | 36 | 36 | 36 | 7 | 35 | 36 | 300 | 35 | 36 | 300 | 35 | 41 | 300 |
| J3011_8 | 62 | 62 | 62 | 5 | 62 | 62 | 5 | 62 | 62 | 22 | 62 | 62 | 21 |
| J3011_9 | 67 | 67 | 67 | 3 | 67 | 67 | 1 | 67 | 67 | 2 | 67 | 67 | 6 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J3011_10 | 38 | 38 | 38 | 4 | 38 | 38 | 8 | 38 | 38 | 6 | 38 | 38 | 132 |
| J3012_1 | 47 | 47 | 47 | 4 | 47 | 47 | 21 | 47 | 47 | 4 | 46 | 47 | 300 |
| J3012_2 | 46 | 46 | 46 | 4 | 46 | 46 | 6 | 46 | 46 | 8 | 46 | 46 | 195 |
| J3012_3 | 37 | 37 | 37 | 2 | 37 | 37 | 244 | 37 | 37 | 37 | 35 | 37 | 300 |
| J3012_4 | 63 | 63 | 63 | 6 | 63 | 63 | 2 | 63 | 63 | 1 | 63 | 63 | 13 |
| J3012_5 | 47 | 47 | 47 | 3 | 47 | 47 | 2 | 47 | 47 | 1 | 47 | 47 | 3 |
| J3012_6 | 53 | 53 | 53 | 3 | 53 | 53 | 2 | 53 | 53 | 6 | 53 | 53 | 27 |
| J3012_7 | 55 | 55 | 55 | 5 | 55 | 55 | 3 | 55 | 55 | 6 | 55 | 55 | 19 |
| J3012_8 | 35 | 35 | 35 | 4 | 35 | 35 | 12 | 35 | 35 | 66 | 35 | 35 | 24 |
| J3012_9 | 52 | 52 | 52 | 4 | 52 | 52 | 3 | 52 | 52 | 5 | 52 | 52 | 22 |
| J3012_10 | 57 | 57 | 57 | 5 | 57 | 57 | 1 | 57 | 57 | 2 | 57 | 57 | 18 |
| J3013_1 | 58 | 51.07 | 61 | 300 | 52.25 | 61 | 300 | 52 | 68 | 300 | 52.25 | 85 | 300 |
| J3013_2 | 62 | 56 | 70 | 300 | 56 | 68 | 300 | 57 | 72 | 300 | 56.2 | 80 | 300 |
| J3013_3 | 76 | 62 | 87 | 300 | 65.9 | 85 | 300 | 65.11 | 85 | 300 | 66 | 109 | 300 |
| J3013_4 | 72 | 65 | 73 | 300 | 62.67 | 73 | 300 | 62.67 | 76 | 300 | 62.5 | 79 | 300 |
| J3013_5 | 67 | 59.04 | 72 | 300 | 59 | 74 | 300 | 59.17 | 83 | 300 | 59 | 97 | 300 |
| J3013_6 | 64 | 59 | 65 | 300 | 56.56 | 72 | 300 | 57.4 | 73 | 300 | 57.6 | 99 | 300 |
| J3013_7 | 77 | 64 | 85 | 300 | 70 | 79 | 300 | 70.5 | 79 | 300 | 72 | 84 | 300 |
| J3013_8 | 106 | 76.32 | 123 | 300 | 101 | 106 | 300 | 101 | 110 | 300 | 100.5 | 107 | 300 |
| J3013_9 | 71 | 69 | 72 | 300 | 64 | 77 | 300 | 63 | 81 | 300 | 63 | 84 | 300 |
| J3013_10 | 64 | 64 | 64 | 212 | 57.5 | 67 | 300 | 57.5 | 64 | 300 | 55 | 70 | 300 |
| J3014_1 | 50 | 50 | 50 | 22 | 43 | 50 | 300 | 43 | 50 | 300 | 43 | 53 | 300 |
| J3014_2 | 53 | 53 | 53 | 131 | 46.8 | 63 | 300 | 46.81 | 59 | 300 | 46.92 | 73 | 300 |
| J3014_3 | 58 | 58 | 58 | 13 | 58 | 61 | 300 | 58 | 61 | 300 | 58 | 67 | 300 |
| J3014_4 | 50 | 50 | 50 | 10 | 45 | 52 | 300 | 45 | 56 | 300 | 45 | 56 | 300 |
| J3014_5 | 52 | 52 | 52 | 6 | 50 | 53 | 300 | 50 | 52 | 300 | 50 | 56 | 300 |
| J3014_6 | 35 | 35 | 35 | 8 | 34 | 35 | 300 | 34 | 36 | 300 | 34 | 39 | 300 |
| J3014_7 | 50 | 50 | 50 | 16 | 47 | 53 | 300 | 47 | 52 | 300 | 47 | 59 | 300 |
| J3014_8 | 54 | 54 | 54 | 4 | 54 | 54 | 7 | 54 | 54 | 6 | 54 | 55 | 300 |
| J3014_9 | 46 | 46 | 46 | 48 | 44 | 48 | 300 | 44 | 48 | 300 | 44 | 51 | 300 |
| J3014_10 | 61 | 61 | 61 | 7 | 58 | 61 | 300 | 58 | 61 | 300 | 58 | 64 | 300 |
| J3015_1 | 46 | 46 | 46 | 3 | 46 | 46 | 12 | 46 | 46 | 5 | 46 | 46 | 38 |
| J3015_2 | 47 | 47 | 47 | 5 | 47 | 47 | 4 | 47 | 47 | 4 | 47 | 47 | 18 |
| J3015_3 | 48 | 48 | 48 | 3 | 48 | 48 | 3 | 48 | 48 | 3 | 48 | 48 | 46 |
| J3015_4 | 48 | 48 | 48 | 3 | 48 | 48 | 1 | 48 | 48 | 2 | 48 | 48 | 5 |
| J3015_5 | 58 | 58 | 58 | 11 | 56 | 61 | 300 | 56 | 58 | 300 | 56 | 62 | 300 |
| J3015_6 | 67 | 67 | 67 | 4 | 67 | 67 | 5 | 67 | 67 | 3 | 67 | 67 | 50 |
| J3015_7 | 47 | 47 | 47 | 4 | 47 | 47 | 5 | 47 | 47 | 6 | 47 | 48 | 300 |
| J3015_8 | 50 | 50 | 50 | 6 | 50 | 50 | 52 | 50 | 50 | 197 | 48 | 52 | 300 |
| J3015_9 | 54 | 54 | 54 | 4 | 54 | 54 | 3 | 54 | 54 | 8 | 52 | 54 | 300 |
| J3015_10 | 65 | 65 | 65 | 4 | 65 | 65 | 3 | 65 | 65 | 7 | 65 | 65 | 83 |
| J3016_1 | 51 | 51 | 51 | 3 | 51 | 51 | 3 | 51 | 51 | 6 | 51 | 51 | 30 |
| J3016_2 | 48 | 48 | 48 | 4 | 48 | 48 | 6 | 48 | 48 | 4 | 48 | 48 | 85 |
| J3016_3 | 36 | 36 | 36 | 3 | 36 | 36 | 6 | 36 | 36 | 3 | 36 | 36 | 212 |
| J3016_4 | 47 | 47 | 47 | 3 | 47 | 47 | 233 | 47 | 47 | 10 | 47 | 47 | 6 |
| J3016_5 | 51 | 51 | 51 | 4 | 51 | 51 | 1 | 51 | 51 | 2 | 51 | 51 | 11 |
| J3016_6 | 51 | 51 | 51 | 3 | 51 | 51 | 2 | 51 | 51 | 1 | 51 | 51 | 6 |
| J3016_7 | 34 | 34 | 34 | 3 | 34 | 34 | 4 | 34 | 34 | 2 | 31 | 34 | 300 |
| J3016_8 | 44 | 44 | 44 | 3 | 44 | 44 | 3 | 44 | 44 | 2 | 44 | 44 | 7 |
| J3016_9 | 44 | 44 | 44 | 3 | 44 | 44 | 2 | 44 | 44 | 2 | 44 | 44 | 9 |
| J3016_10 | 51 | 51 | 51 | 4 | 51 | 51 | 2 | 51 | 51 | 2 | 51 | 51 | 5 |
| J3017_1 | 64 | 64 | 64 | 35 | 58 | 64 | 300 | 61 | 64 | 300 | 57 | 65 | 300 |
| J3017_2 | 68 | 68 | 68 | 7 | 68 | 68 | 57 | 68 | 68 | 14 | 59 | 68 | 300 |
| J3017_3 | 60 | 60 | 60 | 4 | 60 | 60 | 1 | 60 | 60 | 2 | 60 | 60 | 7 |
| J3017_4 | 49 | 49 | 49 | 5 | 49 | 49 | 4 | 49 | 49 | 10 | 49 | 49 | 45 |
| J3017_5 | 47 | 47 | 47 | 11 | 40 | 47 | 300 | 40.44 | 47 | 300 | 39 | 47 | 300 |
| J3017_6 | 63 | 63 | 63 | 5 | 63 | 63 | 1 | 63 | 63 | 3 | 63 | 63 | 4 |
| J3017_7 | 57 | 57 | 57 | 13 | 51 | 57 | 300 | 57 | 57 | 116 | 51 | 57 | 300 |
| J3017_8 | 61 | 61 | 61 | 7 | 61 | 61 | 116 | 61 | 61 | 249 | 50 | 62 | 300 |
| J3017_9 | 48 | 48 | 48 | 4 | 45 | 48 | 300 | 48 | 48 | 39 | 45 | 49 | 300 |
| J3017_10 | 66 | 66 | 66 | 7 | 66 | 66 | 2 | 66 | 66 | 2 | 66 | 66 | 11 |
| J3018_1 | 53 | 53 | 53 | 6 | 53 | 53 | 88 | 53 | 53 | 6 | 51 | 53 | 300 |
| J3018_2 | 55 | 55 | 55 | 4 | 55 | 55 | 22 | 55 | 55 | 4 | 49 | 55 | 300 |
| J3018_3 | 56 | 56 | 56 | 6 | 56 | 56 | 41 | 56 | 56 | 4 | 56 | 56 | 209 |
| J3018_4 | 70 | 70 | 70 | 6 | 70 | 70 | 1 | 70 | 70 | 2 | 70 | 70 | 6 |
| J3018_5 | 52 | 52 | 52 | 5 | 52 | 52 | 31 | 52 | 52 | 4 | 52 | 52 | 156 |
| J3018_6 | 62 | 62 | 62 | 11 | 55 | 62 | 300 | 62 | 62 | 183 | 55 | 62 | 300 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J3018_7 | 48 | 48 | 48 | 4 | 48 | 48 | 70 | 48 | 48 | 5 | 48 | 48 | 242 |
| J3018_8 | 52 | 52 | 52 | 5 | 52 | 52 | 42 | 52 | 52 | 3 | 49 | 52 | 300 |
| J3018_9 | 47 | 47 | 47 | 5 | 47 | 47 | 28 | 47 | 47 | 25 | 44 | 47 | 300 |
| J3018_10 | 49 | 49 | 49 | 7 | 45 | 49 | 300 | 47 | 49 | 300 | 42 | 49 | 300 |
| J3019_1 | 40 | 40 | 40 | 4 | 40 | 40 | 186 | 40 | 40 | 43 | 39 | 40 | 300 |
| J3019_2 | 58 | 58 | 58 | 6 | 58 | 58 | 1 | 58 | 58 | 2 | 58 | 58 | 3 |
| J3019_3 | 83 | 83 | 83 | 8 | 83 | 83 | 1 | 83 | 83 | 2 | 83 | 83 | 2 |
| J3019_4 | 39 | 39 | 39 | 4 | 39 | 39 | 4 | 39 | 39 | 4 | 39 | 39 | 21 |
| J3019_5 | 48 | 48 | 48 | 5 | 47 | 48 | 300 | 48 | 48 | 136 | 47 | 48 | 300 |
| J3019_6 | 49 | 49 | 49 | 4 | 44 | 49 | 300 | 49 | 49 | 31 | 49 | 49 | 5 |
| J3019_7 | 57 | 57 | 57 | 6 | 54 | 57 | 300 | 57 | 57 | 46 | 54 | 57 | 300 |
| J3019_8 | 55 | 55 | 55 | 7 | 55 | 55 | 5 | 55 | 55 | 44 | 55 | 55 | 36 |
| J3019_9 | 38 | 38 | 38 | 4 | 38 | 38 | 8 | 38 | 38 | 3 | 38 | 38 | 16 |
| J3019_10 | 47 | 47 | 47 | 5 | 47 | 47 | 11 | 47 | 47 | 5 | 47 | 47 | 37 |
| J3020_1 | 57 | 57 | 57 | 6 | 57 | 57 | 1 | 57 | 57 | 1 | 57 | 57 | 3 |
| J3020_2 | 70 | 70 | 70 | 5 | 70 | 70 | 2 | 70 | 70 | 2 | 70 | 70 | 8 |
| J3020_3 | 49 | 49 | 49 | 6 | 48 | 49 | 300 | 49 | 49 | 2 | 49 | 49 | 6 |
| J3020_4 | 43 | 43 | 43 | 4 | 43 | 43 | 8 | 43 | 43 | 2 | 43 | 43 | 16 |
| J3020_5 | 61 | 61 | 61 | 4 | 61 | 61 | 1 | 61 | 61 | 1 | 61 | 61 | 5 |
| J3020_6 | 51 | 51 | 51 | 5 | 51 | 51 | 2 | 51 | 51 | 1 | 51 | 51 | 7 |
| J3020_7 | 42 | 42 | 42 | 4 | 42 | 42 | 3 | 42 | 42 | 2 | 42 | 42 | 7 |
| J3020_8 | 51 | 51 | 51 | 4 | 51 | 51 | 2 | 51 | 51 | 1 | 51 | 51 | 4 |
| J3020_9 | 41 | 41 | 41 | 3 | 41 | 41 | 2 | 41 | 41 | 1 | 41 | 41 | 5 |
| J3020_10 | 37 | 37 | 37 | 3 | 37 | 37 | 2 | 37 | 37 | 2 | 37 | 37 | 9 |
| J3021_1 | 84 | 84 | 84 | 68 | 77.71 | 84 | 300 | 84 | 84 | 249 | 76 | 84 | 300 |
| J3021_2 | 59 | 59 | 59 | 117 | 52 | 59 | 300 | 54 | 59 | 300 | 52 | 62 | 300 |
| J3021_3 | 76 | 76 | 76 | 34 | 74 | 76 | 300 | 75 | 76 | 300 | 71 | 76 | 300 |
| J3021_4 | 70 | 70 | 70 | 58 | 62 | 70 | 300 | 63.5 | 70 | 300 | 60.5 | 74 | 300 |
| J3021_5 | 55 | 55 | 55 | 18 | 52 | 55 | 300 | 53 | 55 | 300 | 52 | 57 | 300 |
| J3021_6 | 76 | 76 | 76 | 75 | 68 | 76 | 300 | 74 | 76 | 300 | 68 | 76 | 300 |
| J3021_7 | 65 | 65 | 65 | 50 | 57.5 | 66 | 300 | 59 | 65 | 300 | 58 | 66 | 300 |
| J3021_8 | 62 | 62 | 62 | 54 | 54.09 | 66 | 300 | 55 | 63 | 300 | 55.5 | 65 | 300 |
| J3021_9 | 69 | 69 | 69 | 84 | 68 | 69 | 300 | 68 | 69 | 300 | 68 | 69 | 300 |
| J3021_10 | 69 | 69 | 69 | 35 | 59 | 71 | 300 | 59 | 69 | 300 | 59 | 74 | 300 |
| J3022_1 | 42 | 42 | 42 | 6 | 40 | 42 | 300 | 39 | 42 | 300 | 37 | 42 | 300 |
| J3022_2 | 45 | 45 | 45 | 5 | 45 | 45 | 166 | 45 | 45 | 55 | 42 | 45 | 300 |
| J3022_3 | 63 | 63 | 63 | 5 | 63 | 63 | 220 | 63 | 63 | 93 | 61 | 63 | 300 |
| J3022_4 | 42 | 42 | 42 | 7 | 40 | 42 | 300 | 40 | 42 | 300 | 37 | 42 | 300 |
| J3022_5 | 52 | 52 | 52 | 5 | 46 | 52 | 300 | 46.25 | 52 | 300 | 46 | 55 | 300 |
| J3022_6 | 52 | 52 | 52 | 17 | 45 | 52 | 300 | 45 | 52 | 300 | 45 | 52 | 300 |
| J3022_7 | 60 | 60 | 60 | 11 | 55 | 60 | 300 | 55 | 60 | 300 | 55 | 63 | 300 |
| J3022_8 | 55 | 55 | 55 | 11 | 50 | 56 | 300 | 50 | 55 | 300 | 50 | 57 | 300 |
| J3022_9 | 76 | 76 | 76 | 6 | 76 | 76 | 9 | 76 | 76 | 6 | 73 | 76 | 300 |
| J3022_10 | 55 | 55 | 55 | 8 | 52 | 55 | 300 | 52 | 55 | 300 | 52 | 55 | 300 |
| J3023_1 | 63 | 63 | 63 | 4 | 63 | 63 | 1 | 63 | 63 | 4 | 63 | 63 | 26 |
| J3023_2 | 53 | 53 | 53 | 4 | 53 | 53 | 3 | 53 | 53 | 2 | 53 | 53 | 8 |
| J3023_3 | 46 | 46 | 46 | 4 | 46 | 46 | 2 | 46 | 46 | 2 | 46 | 46 | 11 |
| J3023_4 | 65 | 65 | 65 | 6 | 65 | 65 | 7 | 65 | 65 | 2 | 65 | 65 | 9 |
| J3023_5 | 52 | 52 | 52 | 3 | 52 | 52 | 51 | 52 | 52 | 33 | 48 | 52 | 300 |
| J3023_6 | 48 | 48 | 48 | 4 | 48 | 48 | 49 | 48 | 48 | 3 | 48 | 48 | 186 |
| J3023_7 | 60 | 60 | 60 | 5 | 60 | 60 | 158 | 60 | 60 | 28 | 58 | 60 | 300 |
| J3023_8 | 48 | 48 | 48 | 4 | 48 | 48 | 20 | 48 | 48 | 6 | 48 | 49 | 300 |
| J3023_9 | 63 | 63 | 63 | 9 | 63 | 63 | 97 | 63 | 63 | 4 | 58 | 63 | 300 |
| J3023_10 | 61 | 61 | 61 | 4 | 61 | 61 | 1 | 61 | 61 | 3 | 61 | 61 | 6 |
| J3024_1 | 53 | 53 | 53 | 5 | 53 | 53 | 2 | 53 | 53 | 1 | 53 | 53 | 3 |
| J3024_2 | 58 | 58 | 58 | 3 | 58 | 58 | 2 | 58 | 58 | 2 | 58 | 58 | 3 |
| J3024_3 | 69 | 69 | 69 | 7 | 69 | 69 | 3 | 69 | 69 | 3 | 69 | 69 | 8 |
| J3024_4 | 53 | 53 | 53 | 4 | 53 | 53 | 1 | 53 | 53 | 1 | 53 | 53 | 5 |
| J3024_5 | 51 | 51 | 51 | 5 | 51 | 51 | 1 | 51 | 51 | 2 | 51 | 51 | 9 |
| J3024_6 | 56 | 56 | 56 | 5 | 56 | 56 | 2 | 56 | 56 | 1 | 56 | 56 | 5 |
| J3024_7 | 44 | 44 | 44 | 4 | 44 | 44 | 3 | 44 | 44 | 1 | 44 | 44 | 4 |
| J3024_8 | 38 | 38 | 38 | 4 | 38 | 38 | 4 | 38 | 38 | 2 | 37 | 38 | 300 |
| J3024_9 | 43 | 43 | 43 | 6 | 43 | 43 | 3 | 43 | 43 | 2 | 43 | 43 | 16 |
| J3024_10 | 53 | 53 | 53 | 4 | 53 | 53 | 2 | 53 | 53 | 1 | 53 | 53 | 7 |
| J3025_1 | 93 | 81 | 106 | 300 | 85 | 94 | 300 | 84.33 | 96 | 300 | 82.67 | 98 | 300 |
| J3025_2 | 75 | 64.03 | 82 | 300 | 68.43 | 79 | 300 | 72.5 | 75 | 300 | 68.29 | 78 | 300 |
| J3025_3 | 76 | 68.28 | 81 | 300 | 68.5 | 79 | 300 | 69.83 | 84 | 300 | 72 | 80 | 300 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J3025_4 | 81 | 63.26 | 97 | 300 | 73 | 83 | 300 | 74.5 | 83 | 300 | 73 | 89 | 300 |
| J3025_5 | 72 | 66 | 73 | 300 | 64.5 | 75 | 300 | 64.5 | 72 | 300 | 64.5 | 77 | 300 |
| J3025_6 | 58 | 58 | 58 | 254 | 50 | 65 | 300 | 48.87 | 64 | 300 | 49.5 | 73 | 300 |
| J3025_7 | 95 | 75.84 | 106 | 300 | 90.53 | 95 | 300 | 89.5 | 96 | 300 | 90.09 | 97 | 300 |
| J3025_8 | 69 | 58 | 81 | 300 | 62.5 | 72 | 300 | 62.5 | 75 | 300 | 63.8 | 82 | 300 |
| J3025_9 | 84 | 71.67 | 100 | 300 | 75 | 85 | 300 | 81 | 84 | 300 | 75 | 85 | 300 |
| J3025_10 | 58 | 58 | 58 | 42 | 49.35 | 63 | 300 | 47.74 | 60 | 300 | 47.52 | 62 | 300 |
| J3026_1 | 59 | 59 | 59 | 4 | 59 | 59 | 97 | 59 | 59 | 35 | 57 | 60 | 300 |
| J3026_2 | 40 | 40 | 40 | 4 | 40 | 40 | 10 | 40 | 40 | 22 | 40 | 40 | 42 |
| J3026_3 | 58 | 58 | 58 | 5 | 58 | 58 | 3 | 58 | 58 | 8 | 58 | 58 | 92 |
| J3026_4 | 62 | 62 | 62 | 9 | 62 | 62 | 9 | 62 | 62 | 28 | 62 | 63 | 300 |
| J3026_5 | 74 | 74 | 74 | 10 | 72 | 74 | 300 | 72 | 74 | 300 | 71 | 75 | 300 |
| J3026_6 | 53 | 53 | 53 | 6 | 46 | 55 | 300 | 49 | 54 | 300 | 49 | 55 | 300 |
| J3026_7 | 56 | 56 | 56 | 6 | 56 | 56 | 19 | 56 | 56 | 11 | 56 | 58 | 300 |
| J3026_8 | 66 | 66 | 66 | 6 | 66 | 66 | 3 | 66 | 66 | 2 | 66 | 66 | 18 |
| J3026_9 | 43 | 43 | 43 | 8 | 39.33 | 44 | 300 | 39.25 | 44 | 300 | 41 | 48 | 300 |
| J3026_10 | 49 | 49 | 49 | 7 | 47 | 49 | 300 | 47 | 49 | 300 | 47 | 53 | 300 |
| J3027_1 | 43 | 43 | 43 | 5 | 43 | 43 | 15 | 43 | 43 | 6 | 43 | 43 | 114 |
| J3027_2 | 58 | 58 | 58 | 6 | 58 | 58 | 3 | 58 | 58 | 2 | 58 | 58 | 32 |
| J3027_3 | 60 | 60 | 60 | 6 | 60 | 60 | 1 | 60 | 60 | 2 | 60 | 60 | 4 |
| J3027_4 | 64 | 64 | 64 | 4 | 64 | 64 | 1 | 64 | 64 | 4 | 64 | 64 | 5 |
| J3027_5 | 49 | 49 | 49 | 6 | 49 | 49 | 12 | 49 | 49 | 7 | 49 | 50 | 300 |
| J3027_6 | 59 | 59 | 59 | 8 | 59 | 59 | 9 | 59 | 59 | 3 | 59 | 59 | 54 |
| J3027_7 | 49 | 49 | 49 | 8 | 46 | 49 | 300 | 45 | 49 | 300 | 45 | 51 | 300 |
| J3027_8 | 66 | 66 | 66 | 6 | 66 | 66 | 2 | 66 | 66 | 5 | 66 | 66 | 21 |
| J3027_9 | 55 | 55 | 55 | 6 | 55 | 55 | 14 | 55 | 55 | 3 | 55 | 55 | 62 |
| J3027_10 | 62 | 62 | 62 | 5 | 62 | 62 | 1 | 62 | 62 | 1 | 62 | 62 | 5 |
| J3028_1 | 69 | 69 | 69 | 4 | 69 | 69 | 1 | 69 | 69 | 3 | 69 | 69 | 6 |
| J3028_2 | 57 | 57 | 57 | 5 | 57 | 57 | 1 | 57 | 57 | 2 | 57 | 57 | 5 |
| J3028_3 | 40 | 40 | 40 | 3 | 40 | 40 | 4 | 40 | 40 | 4 | 40 | 40 | 40 |
| J3028_4 | 49 | 49 | 49 | 4 | 49 | 49 | 2 | 49 | 49 | 9 | 49 | 49 | 6 |
| J3028_5 | 73 | 73 | 73 | 5 | 73 | 73 | 1 | 73 | 73 | 1 | 73 | 73 | 3 |
| J3028_6 | 55 | 55 | 55 | 4 | 55 | 55 | 1 | 55 | 55 | 2 | 55 | 55 | 5 |
| J3028_7 | 48 | 48 | 48 | 6 | 48 | 48 | 2 | 48 | 48 | 1 | 48 | 48 | 4 |
| J3028_8 | 53 | 53 | 53 | 4 | 53 | 53 | 3 | 53 | 53 | 2 | 53 | 53 | 6 |
| J3028_9 | 62 | 62 | 62 | 6 | 62 | 62 | 12 | 62 | 62 | 4 | 62 | 62 | 17 |
| J3028_10 | 59 | 59 | 59 | 5 | 59 | 59 | 2 | 59 | 59 | 1 | 59 | 59 | 4 |
| J3029_1 | 85 | 74 | 93 | 300 | 83 | 89 | 300 | 83 | 91 | 300 | 83 | 94 | 300 |
| J3029_2 | 90 | 73.05 | 97 | 300 | 81.5 | 95 | 300 | 81.5 | 97 | 300 | 80.5 | 109 | 300 |
| J3029_3 | 78 | 66.13 | 83 | 300 | 75.5 | 79 | 300 | 75.5 | 79 | 300 | 75.5 | 78 | 300 |
| J3029_4 | 103 | 78.55 | 116 | 300 | 99.5 | 105 | 300 | 100 | 105 | 300 | 99.5 | 107 | 300 |
| J3029_5 | 98 | 77.13 | 117 | 300 | 90.33 | 102 | 300 | 90.67 | 101 | 300 | 90.67 | 112 | 300 |
| J3029_6 | 92 | 68.36 | 111 | 300 | 79 | 97 | 300 | 80 | 99 | 300 | 77.5 | 106 | 300 |
| J3029_7 | 73 | 62 | 77 | 300 | 68.5 | 73 | 300 | 68.5 | 73 | 300 | 68.5 | 75 | 300 |
| J3029_8 | 80 | 71 | 89 | 300 | 71.8 | 90 | 300 | 72.96 | 90 | 300 | 71.87 | 93 | 300 |
| J3029_9 | 97 | 75.54 | 117 | 300 | 89 | 99 | 300 | 89 | 100 | 300 | 89 | 102 | 300 |
| J3029_10 | 76 | 76 | 76 | 176 | 68.33 | 78 | 300 | 70.33 | 78 | 300 | 68 | 79 | 300 |
| J3030_1 | 47 | 47 | 47 | 59 | 43.75 | 56 | 300 | 43 | 49 | 300 | 43.75 | 53 | 300 |
| J3030_2 | 68 | 68 | 68 | 107 | 65 | 69 | 300 | 65 | 70 | 300 | 65 | 70 | 300 |
| J3030_3 | 55 | 55 | 55 | 9 | 53 | 55 | 300 | 53 | 55 | 300 | 53 | 55 | 300 |
| J3030_4 | 53 | 53 | 53 | 13 | 50 | 53 | 300 | 50 | 54 | 300 | 50 | 53 | 300 |
| J3030_5 | 54 | 54 | 54 | 16 | 52 | 56 | 300 | 52 | 54 | 300 | 52 | 57 | 300 |
| J3030_6 | 62 | 62 | 62 | 103 | 54 | 67 | 300 | 55 | 66 | 300 | 54.26 | 80 | 300 |
| J3030_7 | 68 | 68 | 68 | 21 | 62.83 | 71 | 300 | 63 | 69 | 300 | 63 | 78 | 300 |
| J3030_8 | 46 | 45 | 46 | 9 | 40 | 47 | 300 | 40 | 46 | 300 | 40 | 50 | 300 |
| J3030_9 | 46 | 46 | 46 | 26 | 44 | 48 | 300 | 44 | 47 | 300 | 44 | 51 | 300 |
| J3030_10 | 53 | 53 | 53 | 17 | 47.33 | 55 | 300 | 47 | 56 | 300 | 46 | 62 | 300 |
| J3031_1 | 43 | 43 | 43 | 3 | 43 | 43 | 7 | 43 | 43 | 4 | 43 | 43 | 99 |
| J3031_2 | 63 | 63 | 63 | 4 | 63 | 63 | 1 | 63 | 63 | 2 | 63 | 63 | 6 |
| J3031_3 | 58 | 58 | 58 | 8 | 58 | 58 | 2 | 58 | 58 | 4 | 58 | 58 | 19 |
| J3031_4 | 50 | 50 | 50 | 3 | 50 | 50 | 2 | 50 | 50 | 3 | 50 | 50 | 17 |
| J3031_5 | 52 | 52 | 52 | 7 | 48 | 53 | 300 | 50 | 52 | 300 | 48 | 57 | 300 |
| J3031_6 | 53 | 53 | 53 | 4 | 53 | 53 | 2 | 53 | 53 | 1 | 53 | 53 | 23 |
| J3031_7 | 61 | 61 | 61 | 7 | 61 | 61 | 14 | 61 | 61 | 20 | 61 | 62 | 300 |
| J3031_8 | 58 | 58 | 58 | 6 | 58 | 58 | 8 | 58 | 58 | 189 | 58 | 58 | 181 |
| J3031_9 | 50 | 50 | 50 | 11 | 46 | 50 | 300 | 47.33 | 51 | 300 | 46 | 55 | 300 |
| J3031_10 | 55 | 55 | 55 | 20 | 48 | 58 | 300 | 49 | 56 | 300 | 48 | 62 | 300 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J3032_1 | 61 | 61 | 61 | 5 | 61 | 61 | 1 | 61 | 61 | 2 | 61 | 61 | 5 |
| J3032_2 | 60 | 60 | 60 | 5 | 60 | 60 | 1 | 60 | 60 | 2 | 60 | 60 | 8 |
| J3032_3 | 57 | 57 | 57 | 8 | 57 | 57 | 4 | 57 | 57 | 6 | 57 | 57 | 16 |
| J3032_4 | 68 | 68 | 68 | 6 | 68 | 68 | 3 | 68 | 68 | 5 | 68 | 68 | 16 |
| J3032_5 | 54 | 54 | 54 | 5 | 54 | 54 | 3 | 54 | 54 | 3 | 54 | 54 | 11 |
| J3032_6 | 44 | 44 | 44 | 3 | 44 | 44 | 3 | 44 | 44 | 4 | 44 | 44 | 17 |
| J3032_7 | 35 | 35 | 35 | 4 | 35 | 35 | 4 | 35 | 35 | 3 | 35 | 35 | 5 |
| J3032_8 | 54 | 54 | 54 | 4 | 54 | 54 | 2 | 54 | 54 | 2 | 54 | 54 | 11 |
| J3032_9 | 65 | 65 | 65 | 5 | 65 | 65 | 1 | 65 | 65 | 1 | 65 | 65 | 4 |
| J3032_10 | 51 | 51 | 51 | 6 | 51 | 51 | 4 | 51 | 51 | 2 | 51 | 51 | 11 |
| J3033_1 | 65 | 65 | 65 | 11 | 65 | 65 | 6 | 65 | 65 | 3 | 65 | 65 | 32 |
| J3033_2 | 60 | 60 | 60 | 9 | 55 | 60 | 300 | 60 | 60 | 57 | 53 | 60 | 300 |
| J3033_3 | 55 | 55 | 55 | 23 | 53.07 | 55 | 300 | 55 | 55 | 17 | 47 | 55 | 300 |
| J3033_4 | 77 | 77 | 77 | 9 | 77 | 77 | 2 | 77 | 77 | 2 | 77 | 77 | 5 |
| J3033_5 | 53 | 53 | 53 | 7 | 53 | 53 | 2 | 53 | 53 | 2 | 53 | 53 | 4 |
| J3033_6 | 59 | 59 | 59 | 7 | 59 | 59 | 33 | 59 | 59 | 4 | 59 | 59 | 64 |
| J3033_7 | 58 | 58 | 58 | 6 | 58 | 58 | 1 | 58 | 58 | 4 | 58 | 58 | 5 |
| J3033_8 | 61 | 61 | 61 | 26 | 59 | 61 | 300 | 61 | 61 | 95 | 59 | 61 | 300 |
| J3033_9 | 65 | 65 | 65 | 19 | 58 | 65 | 300 | 62 | 65 | 300 | 57 | 65 | 300 |
| J3033_10 | 53 | 53 | 53 | 7 | 53 | 53 | 183 | 53 | 53 | 22 | 51 | 53 | 300 |
| J3034_1 | 68 | 68 | 68 | 9 | 68 | 68 | 2 | 68 | 68 | 3 | 68 | 68 | 5 |
| J3034_2 | 44 | 44 | 44 | 4 | 44 | 44 | 11 | 44 | 44 | 16 | 39.5 | 44 | 300 |
| J3034_3 | 69 | 69 | 69 | 6 | 69 | 69 | 1 | 69 | 69 | 3 | 69 | 69 | 5 |
| J3034_4 | 67 | 67 | 67 | 6 | 67 | 67 | 1 | 67 | 67 | 1 | 67 | 67 | 2 |
| J3034_5 | 63 | 63 | 63 | 7 | 63 | 63 | 1 | 63 | 63 | 1 | 63 | 63 | 3 |
| J3034_6 | 52 | 52 | 52 | 6 | 52 | 52 | 167 | 52 | 52 | 32 | 51 | 52 | 300 |
| J3034_7 | 58 | 58 | 58 | 8 | 58 | 58 | 2 | 58 | 58 | 1 | 58 | 58 | 3 |
| J3034_8 | 58 | 58 | 58 | 6 | 57 | 58 | 300 | 58 | 58 | 32 | 51 | 58 | 300 |
| J3034_9 | 60 | 60 | 60 | 4 | 60 | 60 | 1 | 60 | 60 | 1 | 60 | 60 | 2 |
| J3034_10 | 47 | 47 | 47 | 6 | 47 | 47 | 2 | 47 | 47 | 2 | 45 | 47 | 300 |
| J3035_1 | 57 | 57 | 57 | 6 | 57 | 57 | 9 | 57 | 57 | 2 | 57 | 57 | 8 |
| J3035_2 | 53 | 53 | 53 | 6 | 53 | 53 | 1 | 53 | 53 | 2 | 53 | 53 | 5 |
| J3035_3 | 60 | 60 | 60 | 4 | 60 | 60 | 1 | 60 | 60 | 1 | 60 | 60 | 3 |
| J3035_4 | 50 | 50 | 50 | 6 | 47 | 50 | 300 | 50 | 50 | 24 | 48 | 50 | 300 |
| J3035_5 | 60 | 60 | 60 | 5 | 60 | 60 | 16 | 60 | 60 | 1 | 60 | 60 | 31 |
| J3035_6 | 58 | 58 | 58 | 6 | 58 | 58 | 4 | 58 | 58 | 3 | 58 | 58 | 4 |
| J3035_7 | 61 | 61 | 61 | 5 | 61 | 61 | 1 | 61 | 61 | 2 | 61 | 61 | 2 |
| J3035_8 | 63 | 63 | 63 | 6 | 63 | 63 | 1 | 63 | 63 | 1 | 63 | 63 | 4 |
| J3035_9 | 59 | 59 | 59 | 8 | 58 | 59 | 300 | 59 | 59 | 26 | 58 | 59 | 300 |
| J3035_10 | 59 | 59 | 59 | 7 | 59 | 59 | 5 | 59 | 59 | 18 | 59 | 59 | 54 |
| J3036_1 | 66 | 66 | 66 | 6 | 66 | 66 | 1 | 66 | 66 | 1 | 66 | 66 | 3 |
| J3036_2 | 44 | 44 | 44 | 5 | 44 | 44 | 2 | 44 | 44 | 1 | 44 | 44 | 6 |
| J3036_3 | 61 | 61 | 61 | 5 | 61 | 61 | 1 | 61 | 61 | 2 | 61 | 61 | 3 |
| J3036_4 | 59 | 59 | 59 | 5 | 59 | 59 | 1 | 59 | 59 | 2 | 59 | 59 | 2 |
| J3036_5 | 64 | 64 | 64 | 6 | 64 | 64 | 2 | 64 | 64 | 1 | 64 | 64 | 4 |
| J3036_6 | 46 | 46 | 46 | 5 | 46 | 46 | 1 | 46 | 46 | 2 | 46 | 46 | 6 |
| J3036_7 | 56 | 56 | 56 | 6 | 56 | 56 | 2 | 56 | 56 | 1 | 56 | 56 | 6 |
| J3036_8 | 63 | 63 | 63 | 4 | 63 | 63 | 1 | 63 | 63 | 1 | 63 | 63 | 2 |
| J3036_9 | 59 | 59 | 59 | 6 | 59 | 59 | 12 | 59 | 59 | 1 | 59 | 59 | 3 |
| J3036_10 | 59 | 59 | 59 | 6 | 59 | 59 | 2 | 59 | 59 | 2 | 59 | 59 | 2 |
| J3037_1 | 79 | 79 | 79 | 190 | 79 | 79 | 21 | 79 | 79 | 21 | 77 | 79 | 300 |
| J3037_2 | 69 | 69 | 69 | 21 | 59 | 69 | 300 | 59 | 69 | 300 | 59 | 69 | 300 |
| J3037_3 | 81 | 81 | 81 | 273 | 81 | 81 | 3 | 81 | 81 | 7 | 81 | 81 | 65 |
| J3037_4 | 83 | 83 | 83 | 65 | 78 | 83 | 300 | 82 | 83 | 300 | 74.19 | 83 | 300 |
| J3037_5 | 80 | 80 | 80 | 192 | 72.43 | 80 | 300 | 74.38 | 80 | 300 | 72 | 82 | 300 |
| J3037_6 | 73 | 73 | 73 | 110 | 65.5 | 73 | 300 | 66.76 | 73 | 300 | 65.5 | 73 | 300 |
| J3037_7 | 92 | 78.5 | 99 | 300 | 92 | 92 | 77 | 92 | 92 | 70 | 85.13 | 92 | 300 |
| J3037_8 | 72 | 72 | 72 | 57 | 67.32 | 72 | 300 | 68 | 72 | 300 | 66.5 | 72 | 300 |
| J3037_9 | 57 | 57 | 57 | 14 | 53 | 57 | 300 | 55 | 57 | 300 | 49.8 | 57 | 300 |
| J3037_10 | 81 | 81 | 81 | 58 | 80.25 | 81 | 300 | 81 | 81 | 36 | 79 | 81 | 300 |
| J3038_1 | 48 | 48 | 48 | 10 | 46 | 48 | 300 | 48 | 48 | 217 | 45 | 48 | 300 |
| J3038_2 | 54 | 54 | 54 | 7 | 53 | 54 | 300 | 54 | 54 | 26 | 50 | 54 | 300 |
| J3038_3 | 59 | 59 | 59 | 6 | 53 | 59 | 300 | 53.25 | 59 | 300 | 53 | 59 | 300 |
| J3038_4 | 59 | 59 | 59 | 8 | 59 | 59 | 95 | 59 | 59 | 30 | 56 | 59 | 300 |
| J3038_5 | 71 | 71 | 71 | 37 | 67 | 71 | 300 | 68.5 | 71 | 300 | 67 | 71 | 300 |
| J3038_6 | 63 | 63 | 63 | 7 | 63 | 63 | 10 | 63 | 63 | 11 | 63 | 63 | 49 |
| J3038_7 | 65 | 65 | 65 | 6 | 63 | 65 | 300 | 64.33 | 65 | 300 | 63 | 65 | 300 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J3038_8 | 61 | 61 | 61 | 9 | 57.39 | 61 | 300 | 60 | 61 | 300 | 55.51 | 61 | 300 |
| J3038_9 | 63 | 63 | 63 | 20 | 63 | 63 | 17 | 63 | 63 | 12 | 60 | 63 | 300 |
| J3038_10 | 60 | 60 | 60 | 7 | 58.5 | 60 | 300 | 58 | 60 | 300 | 58 | 60 | 300 |
| J3039_1 | 55 | 55 | 55 | 6 | 55 | 55 | 2 | 55 | 55 | 1 | 55 | 55 | 4 |
| J3039_2 | 54 | 54 | 54 | 5 | 54 | 54 | 4 | 54 | 54 | 1 | 54 | 54 | 7 |
| J3039_3 | 54 | 54 | 54 | 9 | 53 | 54 | 300 | 54 | 54 | 24 | 53.5 | 54 | 300 |
| J3039_4 | 53 | 53 | 53 | 8 | 53 | 53 | 3 | 53 | 53 | 2 | 53 | 53 | 12 |
| J3039_5 | 55 | 55 | 55 | 6 | 55 | 55 | 3 | 55 | 55 | 2 | 55 | 55 | 6 |
| J3039_6 | 69 | 69 | 69 | 9 | 69 | 69 | 14 | 69 | 69 | 10 | 69 | 69 | 23 |
| J3039_7 | 56 | 56 | 56 | 4 | 56 | 56 | 2 | 56 | 56 | 1 | 56 | 56 | 3 |
| J3039_8 | 67 | 67 | 67 | 7 | 67 | 67 | 1 | 67 | 67 | 1 | 67 | 67 | 4 |
| J3039_9 | 64 | 64 | 64 | 8 | 63 | 64 | 300 | 64 | 64 | 25 | 60 | 64 | 300 |
| J3039_10 | 60 | 59 | 60 | 5 | 60 | 60 | 4 | 60 | 60 | 3 | 60 | 60 | 37 |
| J3040_1 | 51 | 51 | 51 | 7 | 51 | 51 | 2 | 51 | 51 | 2 | 51 | 51 | 6 |
| J3040_2 | 56 | 56 | 56 | 6 | 56 | 56 | 1 | 56 | 56 | 9 | 56 | 56 | 3 |
| J3040_3 | 57 | 57 | 57 | 5 | 57 | 57 | 2 | 57 | 57 | 1 | 57 | 57 | 41 |
| J3040_4 | 57 | 57 | 57 | 7 | 57 | 57 | 7 | 57 | 57 | 2 | 57 | 57 | 8 |
| J3040_5 | 65 | 65 | 65 | 7 | 65 | 65 | 1 | 65 | 65 | 1 | 65 | 65 | 5 |
| J3040_6 | 60 | 60 | 60 | 5 | 60 | 60 | 2 | 60 | 60 | 1 | 60 | 60 | 4 |
| J3040_7 | 46 | 46 | 46 | 7 | 43 | 46 | 300 | 46 | 46 | 41 | 46 | 46 | 69 |
| J3040_8 | 57 | 57 | 57 | 6 | 57 | 57 | 2 | 57 | 57 | 1 | 57 | 57 | 5 |
| J3040_9 | 64 | 64 | 64 | 7 | 64 | 64 | 2 | 64 | 64 | 1 | 64 | 64 | 7 |
| J3040_10 | 51 | 51 | 51 | 7 | 51 | 51 | 1 | 51 | 51 | 1 | 51 | 51 | 5 |
| J3041_1 | 86 | 75 | 89 | 300 | 80 | 86 | 300 | 84.5 | 86 | 300 | 75.5 | 88 | 300 |
| J3041_2 | 89 | 71 | 97 | 300 | 79.24 | 90 | 300 | 83 | 89 | 300 | 79.93 | 91 | 300 |
| J3041_3 | 85 | 85 | 85 | 273 | 85 | 85 | 253 | 85 | 85 | 19 | 82.13 | 85 | 300 |
| J3041_4 | 78 | 63.57 | 93 | 300 | 71.5 | 78 | 300 | 74 | 78 | 300 | 71.5 | 79 | 300 |
| J3041_5 | 99 | 88 | 108 | 300 | 91 | 99 | 300 | 90.92 | 99 | 300 | 90.5 | 100 | 300 |
| J3041_6 | 103 | 90 | 108 | 300 | 102 | 103 | 300 | 101 | 103 | 300 | 100 | 103 | 300 |
| J3041_7 | 92 | 76.17 | 103 | 300 | 90 | 92 | 300 | 90 | 92 | 300 | 90 | 92 | 300 |
| J3041_8 | 88 | 73 | 98 | 300 | 77.5 | 89 | 300 | 77 | 89 | 300 | 77 | 88 | 300 |
| J3041_9 | 92 | 84 | 95 | 300 | 86 | 92 | 300 | 89 | 92 | 300 | 86 | 92 | 300 |
| J3041_10 | 99 | 78.8 | 118 | 300 | 99 | 99 | 46 | 99 | 99 | 14 | 97 | 99 | 300 |
| J3042_1 | 58 | 58 | 58 | 7 | 58 | 58 | 9 | 58 | 58 | 6 | 58 | 59 | 300 |
| J3042_2 | 50 | 50 | 50 | 11 | 46 | 50 | 300 | 47 | 50 | 300 | 47 | 52 | 300 |
| J3042_3 | 60 | 60 | 60 | 23 | 52 | 62 | 300 | 52 | 60 | 300 | 51.45 | 64 | 300 |
| J3042_4 | 49 | 49 | 49 | 13 | 44.5 | 53 | 300 | 43.68 | 50 | 300 | 43.38 | 52 | 300 |
| J3042_5 | 52 | 52 | 52 | 6 | 52 | 52 | 116 | 52 | 52 | 10 | 46 | 55 | 300 |
| J3042_6 | 66 | 66 | 66 | 27 | 62 | 67 | 300 | 62.5 | 67 | 300 | 62 | 67 | 300 |
| J3042_7 | 66 | 66 | 66 | 8 | 66 | 66 | 3 | 66 | 66 | 4 | 65 | 66 | 300 |
| J3042_8 | 82 | 82 | 82 | 42 | 82 | 82 | 19 | 82 | 82 | 3 | 82 | 82 | 85 |
| J3042_9 | 60 | 60 | 60 | 13 | 59 | 60 | 300 | 51.54 | 63 | 300 | 59 | 64 | 300 |
| J3042_10 | 75 | 75 | 75 | 8 | 75 | 75 | 2 | 75 | 75 | 2 | 75 | 75 | 36 |
| J3043_1 | 55 | 55 | 55 | 9 | 52 | 56 | 300 | 53 | 55 | 300 | 53 | 57 | 300 |
| J3043_2 | 43 | 43 | 43 | 5 | 43 | 43 | 18 | 43 | 43 | 2 | 43 | 43 | 30 |
| J3043_3 | 57 | 56 | 57 | 14 | 55 | 58 | 300 | 55 | 57 | 300 | 55 | 58 | 300 |
| J3043_4 | 67 | 67 | 67 | 5 | 67 | 67 | 2 | 67 | 67 | 4 | 64.5 | 67 | 300 |
| J3043_5 | 64 | 64 | 64 | 17 | 62 | 64 | 300 | 62 | 64 | 300 | 62 | 66 | 300 |
| J3043_6 | 58 | 58 | 58 | 9 | 56 | 58 | 300 | 58 | 58 | 77 | 56 | 59 | 300 |
| J3043_7 | 52 | 52 | 52 | 7 | 52 | 52 | 90 | 52 | 52 | 39 | 52 | 54 | 300 |
| J3043_8 | 62 | 62 | 62 | 8 | 62 | 62 | 17 | 62 | 62 | 9 | 62 | 63 | 300 |
| J3043_9 | 57 | 57 | 57 | 7 | 56 | 57 | 300 | 56 | 57 | 300 | 56 | 57 | 300 |
| J3043_10 | 60 | 60 | 60 | 6 | 59 | 60 | 300 | 59 | 60 | 300 | 53.31 | 63 | 300 |
| J3044_1 | 50 | 50 | 50 | 4 | 50 | 50 | 2 | 50 | 50 | 2 | 50 | 50 | 24 |
| J3044_2 | 54 | 54 | 54 | 7 | 54 | 54 | 4 | 54 | 54 | 2 | 54 | 54 | 28 |
| J3044_3 | 51 | 51 | 51 | 6 | 51 | 51 | 1 | 51 | 51 | 2 | 51 | 51 | 8 |
| J3044_4 | 57 | 57 | 57 | 5 | 57 | 57 | 1 | 57 | 57 | 1 | 57 | 57 | 9 |
| J3044_5 | 55 | 55 | 55 | 8 | 55 | 55 | 2 | 55 | 55 | 2 | 55 | 55 | 44 |
| J3044_6 | 56 | 56 | 56 | 5 | 56 | 56 | 1 | 56 | 56 | 1 | 56 | 56 | 2 |
| J3044_7 | 42 | 42 | 42 | 6 | 42 | 42 | 2 | 42 | 42 | 1 | 42 | 42 | 4 |
| J3044_8 | 49 | 49 | 49 | 5 | 49 | 49 | 2 | 49 | 49 | 2 | 49 | 49 | 14 |
| J3044_9 | 64 | 64 | 64 | 5 | 64 | 64 | 1 | 64 | 64 | 1 | 64 | 64 | 4 |
| J3044_10 | 63 | 63 | 63 | 6 | 63 | 63 | 1 | 63 | 63 | 2 | 61 | 63 | 300 |
| J3045_1 | 82 | 68 | 94 | 300 | 82 | 82 | 10 | 82 | 82 | 6 | 80 | 82 | 300 |
| J3045_2 | 125 | 92.5 | 135 | 300 | 125 | 125 | 1 | 125 | 125 | 2 | 125 | 125 | 6 |
| J3045_3 | 92 | 81.79 | 99 | 300 | 92 | 92 | 3 | 92 | 92 | 31 | 86.5 | 92 | 300 |
| J3045_4 | 84 | 69.85 | 94 | 300 | 79 | 85 | 300 | 78 | 84 | 300 | 79 | 88 | 300 |

**Table B.1:** Results for the J30 Instances

| | | DDT | | | OOE | | | SEE | | | DP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inst. | OPT | LB | UB | time | LB | UB | time | LB | UB | time | LB | UB | time |
| J3045_5 | 86 | 75 | 97 | 300 | 79 | 86 | 300 | 81 | 86 | 300 | 77.49 | 88 | 300 |
| J3045_6 | 129 | 91.22 | 166 | 300 | 129 | 129 | 2 | 129 | 129 | 3 | 129 | 129 | 12 |
| J3045_7 | 101 | 88 | 111 | 300 | 100 | 101 | 300 | 100.68 | 101 | 300 | 98.96 | 101 | 300 |
| J3045_8 | 94 | 80 | 96 | 300 | 93 | 94 | 300 | 93 | 94 | 300 | 91 | 94 | 300 |
| J3045_9 | 82 | 65.47 | 93 | 300 | 77 | 84 | 300 | 77.25 | 85 | 300 | 77 | 87 | 300 |
| J3045_10 | 90 | 79.32 | 95 | 300 | 81.67 | 91 | 300 | 84 | 91 | 300 | 82.07 | 91 | 300 |
| J3046_1 | 59 | 59 | 59 | 15 | 56 | 61 | 300 | 55.54 | 61 | 300 | 58 | 62 | 300 |
| J3046_2 | 67 | 67 | 67 | 24 | 64 | 69 | 300 | 64 | 70 | 300 | 64 | 70 | 300 |
| J3046_3 | 65 | 65 | 65 | 21 | 64 | 65 | 300 | 64 | 65 | 300 | 64 | 69 | 300 |
| J3046_4 | 64 | 64 | 64 | 22 | 61 | 64 | 300 | 60 | 64 | 300 | 61 | 66 | 300 |
| J3046_5 | 57 | 57 | 57 | 15 | 56 | 57 | 300 | 53 | 57 | 300 | 53.67 | 62 | 300 |
| J3046_6 | 59 | 59 | 59 | 54 | 53 | 60 | 300 | 56 | 59 | 300 | 56 | 61 | 300 |
| J3046_7 | 59 | 59 | 59 | 93 | 54.67 | 60 | 300 | 54.5 | 62 | 300 | 53.63 | 68 | 300 |
| J3046_8 | 58 | 58 | 58 | 57 | 55 | 58 | 300 | 55 | 58 | 300 | 55 | 61 | 300 |
| J3046_9 | 49 | 49 | 49 | 16 | 45.25 | 52 | 300 | 47 | 49 | 300 | 47.33 | 56 | 300 |
| J3046_10 | 55 | 55 | 55 | 25 | 49 | 59 | 300 | 49.27 | 58 | 300 | 49.67 | 63 | 300 |
| J3047_1 | 58 | 58 | 58 | 6 | 58 | 58 | 3 | 58 | 58 | 3 | 58 | 58 | 40 |
| J3047_2 | 59 | 59 | 59 | 7 | 59 | 59 | 4 | 59 | 59 | 3 | 59 | 59 | 27 |
| J3047_3 | 55 | 55 | 55 | 9 | 55 | 55 | 7 | 55 | 55 | 4 | 55 | 55 | 16 |
| J3047_4 | 49 | 49 | 49 | 7 | 48 | 49 | 300 | 48 | 49 | 300 | 48 | 49 | 300 |
| J3047_5 | 47 | 47 | 47 | 6 | 40.35 | 47 | 300 | 42.5 | 47 | 300 | 43 | 47 | 300 |
| J3047_6 | 53 | 53 | 53 | 13 | 50 | 53 | 300 | 50 | 53 | 300 | 50 | 55 | 300 |
| J3047_7 | 66 | 66 | 66 | 13 | 60 | 66 | 300 | 60 | 66 | 300 | 60 | 66 | 300 |
| J3047_8 | 48 | 48 | 48 | 7 | 48 | 48 | 2 | 48 | 48 | 14 | 48 | 48 | 10 |
| J3047_9 | 65 | 65 | 65 | 7 | 65 | 65 | 2 | 65 | 65 | 2 | 65 | 65 | 34 |
| J3047_10 | 60 | 60 | 60 | 19 | 57 | 60 | 300 | 57 | 60 | 300 | 57 | 62 | 300 |
| J3048_1 | 63 | 63 | 63 | 3 | 63 | 63 | 1 | 63 | 63 | 1 | 63 | 63 | 3 |
| J3048_2 | 54 | 54 | 54 | 4 | 54 | 54 | 1 | 54 | 54 | 1 | 54 | 54 | 3 |
| J3048_3 | 50 | 50 | 50 | 4 | 50 | 50 | 1 | 50 | 50 | 1 | 50 | 50 | 8 |
| J3048_4 | 57 | 57 | 57 | 6 | 57 | 57 | 1 | 57 | 57 | 2 | 57 | 57 | 5 |
| J3048_5 | 58 | 58 | 58 | 6 | 58 | 58 | 3 | 58 | 58 | 2 | 58 | 58 | 25 |
| J3048_6 | 58 | 58 | 58 | 5 | 58 | 58 | 1 | 58 | 58 | 1 | 58 | 58 | 2 |
| J3048_7 | 55 | 55 | 55 | 7 | 55 | 55 | 2 | 55 | 55 | 3 | 55 | 55 | 7 |
| J3048_8 | 44 | 44 | 44 | 5 | 44 | 44 | 16 | 44 | 44 | 3 | 44 | 44 | 42 |
| J3048_9 | 59 | 59 | 59 | 9 | 59 | 59 | 2 | 59 | 59 | 1 | 59 | 59 | 5 |
| J3048_10 | 54 | 54 | 54 | 6 | 54 | 54 | 2 | 54 | 54 | 2 | 54 | 54 | 57 |

# Appendix C

# LP-Relaxation Values

**Table C.1:** LP-relaxation values of the extended compact models compared to DDT on the hard problem instances

| Instance | DDT | OOE | dev.% | SEE | dev.% | DP | dev.% |
|---|---|---|---|---|---|---|---|
| J305_1 | 41.4 | 44 | 6.28% | 44 | 6.28% | 44 | 6.28% |
| J305_2 | 58.82 | 63 | 7.11% | 63 | 7.11% | 63 | 7.11% |
| J305_3 | 54.78 | 72 | 31.43% | 72 | 31.43% | 72 | 31.43% |
| J305_4 | 45.56 | 52 | 14.14% | 52 | 14.14% | 52 | 14.14% |
| J305_5 | 58.31 | 65 | 11.47% | 65 | 11.47% | 64 | 9.76% |
| J305_6 | 51.18 | 61 | 19.19% | 61 | 19.19% | 61 | 19.19% |
| J305_7 | 51.11 | 73 | 42.83% | 73 | 42.83% | 73 | 42.83% |
| J305_8 | 53 | 57 | 7.55% | 56 | 5.66% | 57 | 7.55% |
| J305_9 | 38.19 | 36.25 | -5.08% | 36.5 | -4.43% | 36.5 | -4.43% |
| J305_10 | 56 | 58.67 | 4.77% | 59 | 5.36% | 57 | 1.79% |
| J309_1 | 58.27 | 80 | 37.29% | 80 | 37.29% | 80 | 37.29% |
| J309_2 | 53.03 | 90 | 69.72% | 90 | 69.72% | 90 | 69.72% |
| J309_3 | 52.85 | 63 | 19.21% | 63 | 19.21% | 63 | 19.21% |
| J309_4 | 54.61 | 62 | 13.53% | 63.25 | 15.82% | 62.25 | 13.99% |
| J309_5 | 53 | 59 | 11.32% | 62 | 16.98% | 62 | 16.98% |
| J309_6 | 47 | 51.67 | 9.94% | 51.67 | 9.94% | 51.67 | 9.94% |
| J309_7 | 45.6 | 49 | 7.46% | 52 | 14.04% | 54 | 18.42% |
| J309_8 | 60.97 | 81 | 32.85% | 81 | 32.85% | 78.33 | 28.47% |
| J309_9 | 42.73 | 49.75 | 16.43% | 50.75 | 18.77% | 50.75 | 18.77% |
| J309_10 | 63.81 | 79 | 23.81% | 79 | 23.81% | 76 | 19.10% |
| J3013_1 | 42.7 | 51.1 | 19.67% | 52.25 | 22.37% | 52.25 | 22.37% |
| J3013_2 | 42.24 | 56.43 | 33.59% | 56.29 | 33.26% | 56.43 | 33.59% |
| J3013_3 | 54.04 | 67 | 23.98% | 67 | 23.98% | 67 | 23.98% |
| J3013_4 | 51.75 | 62.5 | 20.77% | 62.67 | 21.10% | 62.67 | 21.10% |
| J3013_5 | 48.69 | 59 | 21.17% | 59 | 21.17% | 59 | 21.17% |
| J3013_6 | 47.84 | 58 | 21.24% | 57.8 | 20.82% | 57.29 | 19.75% |
| J3013_7 | 54.25 | 71.25 | 31.34% | 72 | 32.72% | 68.5 | 26.27% |
| J3013_8 | 63.46 | 97 | 52.85% | 98 | 54.43% | 100.5 | 58.37% |
| J3013_9 | 54.69 | 63 | 15.19% | 64 | 17.02% | 63.2 | 15.56% |
| J3013_10 | 52 | 57.5 | 10.58% | 57.5 | 10.58% | 57.5 | 10.58% |

**Table C.1:** LP-relaxation values of the extended compact models compared to DDT on the hard problem instances

| Instance | DDT | OOE | dev.% | SEE | dev.% | DP | dev.% |
|---|---|---|---|---|---|---|---|
| J3021_1 | 62.46 | 76 | 21.68% | 76.5 | 22.48% | 76 | 21.68% |
| J3021_2 | 43.9 | 52 | 18.45% | 52 | 18.45% | 52 | 18.45% |
| J3021_3 | 62.03 | 72 | 16.07% | 72 | 16.07% | 72 | 16.07% |
| J3021_4 | 52.75 | 63 | 19.43% | 60 | 13.74% | 60.5 | 14.69% |
| J3021_5 | 44.8 | 52 | 16.07% | 52 | 16.07% | 52 | 16.07% |
| J3021_6 | 55.35 | 68 | 22.85% | 68 | 22.85% | 68 | 22.85% |
| J3021_7 | 53 | 58 | 9.43% | 57.4 | 8.30% | 58 | 9.43% |
| J3021_8 | 49.73 | 53 | 6.58% | 54 | 8.59% | 56.5 | 13.61% |
| J3021_9 | 47.73 | 68 | 42.47% | 68 | 42.47% | 68 | 42.47% |
| J3021_10 | 55.93 | 56.83 | 1.61% | 59 | 5.49% | 57.5 | 2.81% |
| J3025_1 | 67.25 | 82.5 | 22.68% | 83.75 | 24.54% | 83.67 | 24.42% |
| J3025_2 | 50.95 | 68 | 33.46% | 69.33 | 36.07% | 67.49 | 32.46% |
| J3025_3 | 53.2 | 68.5 | 28.76% | 69.5 | 30.64% | 72 | 35.34% |
| J3025_4 | 52.84 | 73 | 38.15% | 73 | 38.15% | 74 | 40.05% |
| J3025_5 | 49.62 | 64 | 28.98% | 64.5 | 29.99% | 64.5 | 29.99% |
| J3025_6 | 43.93 | 50.31 | 14.52% | 49.67 | 13.07% | 49.86 | 13.50% |
| J3025_7 | 64.5 | 89 | 37.98% | 89 | 37.98% | 89.5 | 38.76% |
| J3025_8 | 45.95 | 63.67 | 38.56% | 63.67 | 38.56% | 63.67 | 38.56% |
| J3025_9 | 57.16 | 75 | 31.21% | 75 | 31.21% | 75 | 31.21% |
| J3025_10 | 47 | 47.15 | 0.32% | 47 | 0.00% | 49 | 4.26% |
| J3029_1 | 64.74 | 83 | 28.21% | 83 | 28.21% | 83 | 28.21% |
| J3029_2 | 62.02 | 81.5 | 31.41% | 80.5 | 29.80% | 81.5 | 31.41% |
| J3029_3 | 54.06 | 75.5 | 39.66% | 75.5 | 39.66% | 75.5 | 39.66% |
| J3029_4 | 66.62 | 99.5 | 49.35% | 98 | 47.10% | 99.5 | 49.35% |
| J3029_5 | 65.7 | 90.33 | 37.49% | 90.33 | 37.49% | 90 | 36.99% |
| J3029_6 | 58.46 | 77.2 | 32.06% | 76 | 30.00% | 77 | 31.71% |
| J3029_7 | 50.22 | 68.5 | 36.40% | 68.5 | 36.40% | 68.5 | 36.40% |
| J3029_8 | 59.97 | 72.2 | 20.39% | 73 | 21.73% | 72.75 | 21.31% |
| J3029_9 | 64.34 | 89 | 38.33% | 89 | 38.33% | 89 | 38.33% |
| J3029_10 | 54.99 | 68 | 23.66% | 68 | 23.66% | 68.33 | 24.26% |
| J3037_1 | 49.67 | 77 | 55.02% | 77 | 55.02% | 77 | 55.02% |
| J3037_2 | 51.16 | 59 | 15.32% | 59 | 15.32% | 59 | 15.32% |
| J3037_3 | 55.94 | 81 | 44.80% | 81 | 44.80% | 81 | 44.80% |
| J3037_4 | 62.69 | 73 | 16.45% | 73 | 16.45% | 73 | 16.45% |
| J3037_5 | 61.62 | 72 | 16.85% | 72 | 16.85% | 72 | 16.85% |
| J3037_6 | 52.85 | 65.5 | 23.94% | 65.5 | 23.94% | 65.5 | 23.94% |
| J3037_7 | 57.89 | 83 | 43.38% | 81.05 | 40.01% | 79 | 36.47% |
| J3037_8 | 57 | 66.5 | 16.67% | 66 | 15.79% | 66.5 | 16.67% |
| J3037_9 | 46.2 | 49.5 | 7.14% | 50.25 | 8.77% | 48.92 | 5.89% |
| J3037_10 | 61.43 | 79 | 28.60% | 79 | 28.60% | 79 | 28.60% |
| J3041_1 | 56.05 | 78 | 39.16% | 77 | 37.38% | 78 | 39.16% |
| J3041_2 | 56.58 | 80.33 | 41.98% | 79.5 | 40.51% | 78 | 37.86% |
| J3041_3 | 62.55 | 82 | 31.10% | 82 | 31.10% | 82 | 31.10% |
| J3041_4 | 51 | 71.5 | 40.20% | 71.5 | 40.20% | 71.5 | 40.20% |
| J3041_5 | 72.41 | 90 | 24.29% | 91 | 25.67% | 90.5 | 24.98% |
| J3041_6 | 69.51 | 100 | 43.86% | 100 | 43.86% | 100 | 43.86% |
| J3041_7 | 62.4 | 90 | 44.23% | 90 | 44.23% | 90 | 44.23% |
| J3041_8 | 62.47 | 78 | 24.86% | 77 | 23.26% | 77 | 23.26% |

**Table C.1:** LP-relaxation values of the extended compact models compared to DDT on the hard problem instances

| Instance | DDT | OOE | dev.% | SEE | dev.% | DP | dev.% |
|---|---|---|---|---|---|---|---|
| J3041_9 | 69.81 | 86 | 23.19% | 86 | 23.19% | 86 | 23.19% |
| J3041_10 | 65.85 | 97 | 47.30% | 97 | 47.30% | 97 | 47.30% |
| J3045_1 | 58.76 | 82 | 39.55% | 82 | 39.55% | 82 | 39.55% |
| J3045_2 | 75.57 | 125 | 65.41% | 125 | 65.41% | 125 | 65.41% |
| J3045_3 | 63.86 | 85.71 | 34.22% | 87 | 36.24% | 87.5 | 37.02% |
| J3045_4 | 63.6 | 79 | 24.21% | 79 | 24.21% | 79 | 24.21% |
| J3045_5 | 63.75 | 78 | 22.35% | 78 | 22.35% | 78 | 22.35% |
| J3045_6 | 79.95 | 129 | 61.35% | 129 | 61.35% | 129 | 61.35% |
| J3045_7 | 75.91 | 98.5 | 29.76% | 98.5 | 29.76% | 98.5 | 29.76% |
| J3045_8 | 65.3 | 91 | 39.36% | 91 | 39.36% | 91 | 39.36% |
| J3045_9 | 57.31 | 77 | 34.36% | 77 | 34.36% | 77 | 34.36% |
| J3045_10 | 63.93 | 83.5 | 30.61% | 80.5 | 25.92% | 81.25 | 27.09% |