A BRANCH AND CUT ALGORITHM FOR THE INVENTORY ROUTING PROBLEM

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

INDUSTRIAL ENGINEERING

By

Özlem Mahmutoğulları

July 2019

A BRANCH AND CUT ALGORITHM FOR THE INVENTORY ROUTING PROBLEM
By Özlem Mahmutoğulları
July 2019

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Hande Yaman Paternotte(Advisor)

Bahar Yetiş

Haldun Süral

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

A BRANCH AND CUT ALGORITHM FOR THE INVENTORY ROUTING PROBLEM

Özlem Mahmutoğulları
M.S. in Industrial Engineering
Advisor: Hande Yaman Paternotte
July 2019

The inventory routing problem arises in vendor managed systems where products are distributed from a supplier to a set of retailers by a homogeneous fleet of capacitated vehicles. The routes of the vehicles and the quantities of products sent to each retailer in each time period are determined in such a way that no stockouts occur and total costs arising from inventory holding and transportation are minimized. Different inventory replenishment policies can be used while managing the inventories at retailers. We consider the problem with the maximum level inventory replenishment policy. We present a mixed integer linear programming model and derive valid inequalities using several structured relaxations. We relate our valid inequalities to those in the previous studies. We also propose new valid inequalities, implement a branch and cut algorithm and present computational results on benchmark instances from the literature as well as new randomly generated instances.

Keywords: Vendor Managed Inventory, Lot-Sizing, Vehicle Routing, Valid Inequalities, Branch and Cut.

ÖZET

ENVANTER ROTALAMA PROBLEMİ İÇİN DAL KESİ ALGORİTMASI

Özlem Mahmutoğulları
Endüstri Mühendisliği, Yüksek Lisans
Tez Danışmanı: Hande Yaman Paternotte
Temmuz 2019

Envanter rotalama problemi, ürünlerin aynı kapasitedeki araçlardan oluşan bir filo tarafından bir tedarikçiden bir dizi perakendeciye dağıtıldığı bir problemdir. Araçların güzergahları ve her perakendeciye her bir zaman periyodu için gönderilen ürün miktarları, envanter ve nakliye maliyetlerinin en aza inmesi amaçlanarak ve mevcut talepler her zaman karşılanarak hesaplanır. Perakendecilerdeki envanterler yönetilirken farklı envanter ikmal politikaları kullanılabilir. Biz problemi maksimum seviyede stok yenileme politikası ile birlikte değerlendiriyoruz. Maksimum stok yenileme politikası altında envanter rotalama problemi için karma bir tamsayılı doğrusal programlama modeli sunuyoruz ve problemin gevşetmelerinden yola çıkarak geçerli eşitsizlikler türetiyoruz. Geçerli eşitsizliklerimizi önceki çalışmalardakilerle ilişkilendirerek açıklıyoruz. Ayrıca yeni geçerli eşitsizlikler ve bir dal kesi algoritması öneriyoruz. Literatürdeki referans örnekler ve rastgele oluşturulmuş yeni örnekler üzerindeki hesaplama sonuçlarını sunuyoruz.

Anahtar sözcükler: Tedarikçi Yönetimli Envanter, Parti Büyüklüğü Belirleme, Araç Rotalama, Geçerli Eşitsizlikler, Dal Kesi.

Acknowledgement

I would like to express my gratitude to Prof. Dr. Hande Yaman for her guidance and support during my study. Without her excellent supervision and valuable ideas, this thesis would not be possible.

I am also grateful to Prof. Dr. Bahar Yetiş Kara and Prof. Dr. Haldun Süral for accepting to read and review this thesis. Their valuable comments have been very important to me.

I would like to thank Gizem Özbaygın for helping me every time I needed her experiences.

I also would like to thank Nazlıcan Arslan, Çağın Ürü, Pelin Keşrit, Damla Akoluk, Parinaz Toufani, Gül Çulhan, Deniz Emre, Yücel Naz Yetimoğlu, İrem Gürsesli, Halil İbrahim Bayrak, Kamyar Kargar and Nihal Berktaş for their precious friendships and all enjoyable moments.

I am thankful to Cansu Gülcan, Cemal İlhan, Nur Timurlenk, Haşim Özlü and Başak Yazar. They have always seen me as their sister and supported me during my graduate study.

Finally, I am deeply grateful to my family. I believe that my father would support me and be proud of me if he lived. I would like to thank my sister Halenur Şahin Mahmutoğulları for her friendship, love and support. I would like to thank my brother İrfan Mahmutoğulları. He is not only my brother but also my best teacher and friend all my life. I would like to thank my mother Zehra Mahmutoğulları for her endless love, support and trust. Feeling their love and support always give me courage and strength.

Contents

1 Introduction **1**

2 Problem Definition **7**

2.1 Notation 7

2.2 Mathematical Formulation 9

3 Relaxations and Valid Inequalities **11**

3.1 Lot Sizing Relaxations 12

3.2 Single Node Flow Set Relaxation 17

3.3 Vehicle Routing Relaxations 17

4 Branch and Cut Algorithm **23**

4.1 Separation Algorithms 25

4.1.1 Separation of (l, S)-like Inequalities with Stock Bounds 26

4.1.2 Separation of Capacity Constraints 27

4.1.3 Separation of Connectivity Constraints 28

4.1.4 Separation of Rounded Capacity Constraints 28

4.1.5 Separation of Flow Cover Inequalities 30

4.1.6 Separation of Simple DR Inequalities 31

4.1.7 One Vehicle Case 32

5 Computational Results 34

6 Conclusion and Future Research 49

A Data 54

B Results 56

C Changes in the Results with the Inequalities 86

D Average Improvements in the Results with the Inequalities 108

List of Figures

5.1 Results for Three Periods and One Vehicle 36

5.2 Results for Six Periods and One Vehicle 37

5.3 Results for Larger Instances with Three Periods and One Vehicle . . . 38

5.4 Results for Three Periods and Two Vehicles 39

5.5 Results for Six Periods and Two Vehicles 40

5.6 Results for Three Periods and Three Vehicles 41

5.7 Results for Six Periods and Three Vehicles 42

5.8 Improvements for One Vehicle 45

5.9 Improvements for Two Vehicles 46

5.10 Improvements for Three Vehicles 47

List of Tables

1.1 Some of the Studies on the PRP and IRP in the Literature 6

4.1 Separation Methods for the Inequalities 26

5.1 Percentages of The Number of Instances Solved to Optimality 43

B.1 *high*50-3 Results 57

B.2 *high*50-3 with Inequalities (3.6), (3.7) & (3.8) Results 58

B.3 *high*50-3 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results . . . 59

B.4 *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results 60

B.5 *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results 61

B.6 *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) Results 62

B.7 *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) Results 63

B.8 *high*30-6 Results 64

B.9 *high*30-6 with Inequalities (3.6), (3.7) & (3.8) Results 65

B.10 *high*30-6 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results . . . 66

B.11 *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results 67

B.12 *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results 68

B.13 *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) Results 69

B.14 *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) Results 70

B.15 *low*50-3 Results 71

B.16 *low*50-3 with Inequalities (3.6), (3.7) & (3.8) Results 72

B.17 *low*50-3 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results . . . 73

B.18 *low*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results 74

B.19 *low*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results 75

B.20 *low*50-3 with Inequalities (3.6), (3.7) and (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) Results 76

B.21 *low*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) and (3.19) Results 77

B.22 *low*30-6 Results 78

B.23 *low*30-6 with Inequalities (3.6), (3.7) & (3.8) Results 79

B.24 *low*30-6 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results . . . 80

B.25 *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results 81

B.26 *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results 82

B.27 *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) Results 83

B.28 *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) Results 84

B.29 Large Instances with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17) and (3.15) Results 85

C.1 Changes in Root LB for Three Periods and One Vehicle 87

C.2 Changes in Root LB for Six Periods and One Vehicle 88

C.3 Changes in CPU for Three Periods and One Vehicle 89

C.4 Changes in CPU for Six Periods and One Vehicle 90

C.5 Changes in Opt Gap One Vehicle 91

C.6 Changes in BLB One Vehicle 92
C.7 Changes in Root LB for Three Periods and Two Vehicles 93
C.8 Changes in Root LB for Six Periods and Two Vehicles 94
C.9 Changes in CPU for Three Periods and Two Vehicles 95
C.10 Changes in CPU for Six Periods and Two Vehicles 96
C.11 Changes in Opt Gap for Three Periods and Two Vehicles 97
C.12 Changes in Opt Gap for Six Periods and Two Vehicles 98
C.13 Changes in BLB for Three Periods and Two Vehicles 99
C.14 Changes in BLB for Six Periods and Two Vehicles 100
C.15 Changes in Root LB for Three Periods and Three Vehicles 101
C.16 Changes in Root LB for Six Periods and Three Vehicles 102
C.17 Changes in CPU for Three Vehicles 103
C.18 Changes in Opt Gap for Three Periods and Three Vehicles 104
C.19 Changes in Opt Gap for Six Periods and Three Vehicles 105
C.20 Changes in BLB for Three Periods and Three Vehicles 106
C.21 Changes in BLB for Six Periods and Three Vehicles 107

D.1 Percentage Improvements in the Results for One Vehicle 109

D.2 Percentage Improvements in the Results for Two Vehicles 111

D.3 Percentage Improvements in the Results for Three Vehicles 113

Chapter 1

Introduction

Vendor managed inventory replenishment (VMI) is a practice in which a central decision maker (the supplier) is responsible for the replenishment of inventories at a set of retailers and the coordination of a fleet of vehicles. In conventional inventory management, each retailer decides when and how many products to order, and then the required amount of products are sent from the supplier to each retailer by the vehicles. In VMI, the supplier monitors the inventory level of each retailer, and then decides when and how many products to deliver to each retailer. Decentralized management can result in larger inventory holding and transportation costs because of the uncertainty regarding the time and amount of the orders (Kleywegt et al. [1]). Therefore, centralized management provides better utilization of products and vehicles.

Inventory routing problem (IRP) is an integrated supply chain problem in which transportation management and inventory control decisions are made. Products are distributed from a supplier to the retailers by a set of identical capacitated vehicles according to VMI. Distribution of the products to retailers are realized at each discrete time period in a way that no stockouts occur. The problem is to minimize the total inventory holding and transportation costs. Therefore, the IRP considers inventory, distribution and routing decisions simultaneously. Integrated structure of the IRP provides more efficiency rather than considering inventory and transportation decisions separately.

VMI is getting much attention from different industries because the cost of technology decreases and thus information can be shared between the supplier and retailers in less time and cost (Campbell et al. [2]). Therefore, the IRP is widely used in several areas. Applications of the IRP are mostly encountered in petrochemical, gas, food and automotive industries.

Order-up-to-level (OU) and maximum level (ML) policies are usually used as inventory replenishment policies for the retailers with capacitated inventory level. OU policy stipulates that delivered quantity for each retailer equals to the difference between maximum inventory level and current inventory level of the retailer at that time. Under ML policy, delivered quantity for each retailer can be any positive number as long as the inventory level does not exceed the maximum level for the retailer after delivery.

In the literature, there are many studies on IRP and other integrated supply chain planning problems related to IRP. All these problems are NP-hard because they include vehicle routing problems. The studies propose exact solution methods by introducing stronger formulations and valid inequalities or heuristic algorithms to find better solutions for large instances.

Chandra [3] considers the dynamic distribution problem that includes replenishment decisions at the supplier and retailers with uncapacitated inventory and fixed vehicle and fixed product ordering costs. A heuristic algorithm is suggested to solve the problem and it is applied on randomly generated instances. Chandra [3] and

Chandra and Fisher [4] show that coordination of production and distribution reduces inefficiencies arising from individual operations and enables to make better decisions for more complex situations.

Fumero and Vercellis [5] propose a model that simultaneously plans production and distribution of multiple items, inventory and routing decisions. They present a Lagrangian heuristic to solve the model. Moreover, they compare the results of synchronized and decoupled approaches for the decisions to be made and they demonstrate that using the synchronized approach is more advantageous than using the decoupled approach.

Lei et al. [6] study the integrated production, inventory and distribution routing problem where there are a set of suppliers and a heterogeneous fleet of vehicles. They present a two phase solution approach. In the first phase, a mixed integer model is solved by considering direct shipments for the routes. A heuristic algorithm is applied to solve a consolidation problem for routing and delivery decisions in the second phase.

Adulyasak et al. [10] consider the production routing problem (PRP). PRP is a problem that combines the production decisions with the inventory, distribution and routing decisions. They propose an optimization based adaptive large neighborhood search heuristic to solve the PRP. The heuristic is based on decomposing the problem as production-distribution-inventory and routing problems. Ruokokoski et al. [11] present different formulations for PRP. They propose two families of valid inequalities and a branch and cut algorithm for the problem. They adapt the priori tour based heuristic for the IRP in Solyalı and Süral [12] to find upper bound for the PRP. Shiguemoto and Armentano [13] consider two integrated problems PRP and IRP. They propose tabu search methodologies to solve the PRP and IRP with single and multiple product types. Moreover, the PRP and IRP are studied by Adulyasak et al. [14]. They present PRP and IRP formulations with or without vehicle index for ML and OU policies and define valid inequalities for these formulations. They propose branch and cut algorithms and a heuristic to set initial upper bounds for the problems.

Archetti et al. [7] propose a hybrid heuristic that consists of three steps to solve the PRP. They study the problem that considers production, distribution and routing decisions of the single item under ML policy. According to these steps, firstly, it is assumed that production is infinite at the supplier. The product quantities delivered to each retailer are determined by considering the total of routing cost and holding costs at the retailers. Then, the amount of production with respect to the given quantities is determined by considering production and holding costs at the supplier. In the last step, the solutions obtained from the first two steps are improved. Moreover, they present an exact solution method to solve PRP under ML policy for single vehicle. They propose valid inequalities and a branch and cut algorithm for the problem. Armentano et al. [8] propose two tabu search approaches to solve the problem with multi items. The first one consists of two steps. These steps are setting initial solutions without considering the capacity constraints, and using the tabu search with respect to the cost function based on the route length and violation amounts of the production and the vehicle capacity. The second one consists of the same construction phase with the first one, and the tabu search that has longer term memory combined with path relinking strategy. This strategy evolves the solutions by composing them with respect to their quality and diversity. Another heuristic solution for the problem is proposed by Bertazzi et al. [9]. The problem is decomposed into two subproblems as production and distribution subproblems. The distribution problem is solved by using a heuristic and then the production problem is solved to optimality according to the results obtained in the first problem.

Archetti et al. [15] study the single vehicle IRP with OU, ML and no stock level policies. They define new valid inequalities to strengthen formulations. They compare the strengths of the valid inequalities and propose a branch and cut algorithm for the IRP under OU policy for one vehicle. This branch and cut algorithm is the first exact approach proposed for the IRP in the literature.

Solyalı and Süral [12] propose a stronger network formulation of IRP for the single vehicle and OU policy. In the formulation, two-index vehicle flow representation is used for routing decisions and a shortest-path network representation is used for the retailers' inventory replenishment decisions. Moreover, they propose a priori tour based heuristic to compute an initial upper bound. Desaulniers et al. [16] propose a new formulation of the IRP and a branch price and cut algorithm based on the formulation for multi vehicle.

Avella et al. [17] present tight formulations for the lot sizing subproblems for each retailer and derive two new cutting plane families by using the interaction between the substructures of lot sizing and routing. They exploit the particular feature of benchmark instances in Archetti et al. [15] while proposing formulations and inequalities. In benchmark instances, there are time invariant demands for the retailers and stock capacities are small integer multiples of the demands.

Coelho and Laporte [18] and Avella et al. [19] study IRP and present new valid inequalities to strengthen the formulation with ML policy. They propose valid inequalities that relate the demands and the capacities for multi periods, furthermore, Avella et al. [19] propose valid inequalities based on arc disjointness of the routes for single periods.

The production and inventory routing problems are the integrated supply chain problems mostly studied in the literature. Some of the studies on these problems can be summarized as shown in the following table.

Table 1.1: Some of the Studies on the PRP and IRP in the Literature

	Heuristics	Exact Methods
Production Routing Problem	Archetti et al. [7] Armentano et al. [8] Bertazzi et al. [9] Adulyasak et al. [10] Shiguemoto and Armentano [13]	Archetti et al. [7] Ruokokoski et al. [11] Adulyasak et al. [14]
Inventory Routing Problem	Solyalı and Süral [12] Shiguemoto and Armentano [13] Adulyasak et al. [14]	Solyalı and Süral [12] Archetti et al. [15] Desaulniers et al. [16] Avella et al. [17] Coelho and Laporte [18]

In this study, we consider the IRP for the distribution of a single product type from a supplier by a homogeneous fleet of vehicles. We propose an exact solution method for the problem. We analyze relaxations of the problem and derive valid inequalities which we incorporate in a branch and cut framework.

The rest of this thesis includes the study on stronger formulation and solution method of the IRP under ML replenishment policy. In Chapter 2, the problem is defined and an initial mathematical formulation with its notation is given. In Chapter 3, relaxations of the problem and valid inequalities in the previous studies are introduced. Then, new valid inequalities are proposed. In Chapter 4, separation algorithms for the valid inequalities are proposed and a branch and cut algorithm is described. In Chapter 5, computational results for benchmark instances and randomly generated instances are presented and discussed. In Chapter 6, a general discussion and possible extensions about the study are presented.

Chapter 2

Problem Definition

IRP combines the inventory planning decisions at the supplier and retailers, and transportation schedules of a homogeneous fleet of vehicles over a planning horizon. The problem aims to minimize the total inventory and routing costs arise from this integrated structure. In this chapter, we present the notation used for the rest of the thesis and give a mixed integer mathematical formulation for the problem.

2.1 Notation

The problem is defined on a complete graph $G = (V_0, E)$ where $V_0 = \{0, \ldots, v\}$ is the vertex set and $E = \{\{i, j\} : i, j \in V_0, i < j\}$ is the edge set. Vertex 0 represents the supplier and the other vertices $V = V_0 \setminus \{0\}$ represent the retailers. Planning horizon T consists of n periods. There are nonnegative traveling costs c_e for each $e \in E$ and inventory holding costs h_{it} for each site $i \in V_0$ in each period $t \in T$. Inventory holding costs occur for the stock levels at the end of the periods. Each site $i \in V_0$ has initial stock level $\bar{s}_{i0}$. The amount of product r_t is delivered to the supplier at the beginning of the period $t \in T$. Each retailer $i \in V$ has demand d_{it} in period $t \in T$. The storage capacity at site $i \in V$ is u_i. We assume that $\bar{s}_{i0} \leq u_i$ and $d_{it} \leq u_i$ for all $t \in T$. The deliveries from the supplier to the retailers are performed by m identical vehicles with capacity Q.

Decision variable x_e^t represents the number of times edge $e \in E$ is traversed in period $t \in T$ and y_{it} represents the number of vehicles that visit site $i \in V_0$ in period $t \in T$. The amount of inventory at site $i \in V_0$ at the end of period $t \in T$ is denoted by s_{it} and the amount shipped from the supplier to retailer $i \in V$ in period $t \in T$ is denoted by q_{it}.

We use the following additional notation. For $S \subset V_0$, $\delta(S)$ is the set of edges with exactly one endpoint in set S. For singletons, we use $\delta(i) = \delta(\{i\})$.

We also use

$$x^t(E') = \sum_{e \in E'} x_e^t \text{ for } E' \subseteq E, t \in T,$$

$$q_{i,kt} = \sum_{j=k}^{t} q_{ij}, \quad y_{i,kt} = \sum_{j=k}^{t} y_{ij}, \quad r_{kt} = \sum_{j=k}^{t} r_j, \quad d_{i,kt} = \sum_{j=k}^{t} d_{ij},$$

$$\bar{d}_{i,kt} = (d_{i,1t} - \bar{s}_{i0})^+ \text{ if } k = 1,$$

$$\bar{d}_{i,kt} = (d_{i,k-1,t} - u_i)^+ \text{ if } k \geq 2,$$

for $i \in V$ and $1 \leq k \leq t \leq n$.

Let $V' = \{i \in V : \bar{s}_{i0} \leq d_{i,1n},\ h_{it} \geq h_{0t}\ \forall t \in T\}$. We know that there exists an optimal solution in which the ending stock at all retailers in V' is zero. For others, it may be cheaper to finish with positive stock.

2.2 Mathematical Formulation

The inventory routing problem is formulated as follows:

$$\min \sum_{i \in V_0} \sum_{t \in T} h_{it} s_{it} + \sum_{t \in T} \sum_{e \in E} c_e x_e^t \tag{2.1}$$

$$\text{s.t. } s_{0,t-1} + r_t = \sum_{i \in V} q_{it} + s_{0t} \qquad t \in T \tag{2.2}$$

$$s_{i,t-1} + q_{it} = d_{it} + s_{it} \qquad i \in V, t \in T \tag{2.3}$$

$$s_{i0} = \bar{s}_{i0} \qquad i \in V_0 \tag{2.4}$$

$$s_{in} = 0 \qquad i \in V' \tag{2.5}$$

$$s_{it} + d_{it} \leq u_i \qquad i \in V, t \in T \tag{2.6}$$

$$x^t(\delta(i)) = 2y_{it} \qquad i \in V_0, t \in T \tag{2.7}$$

$$Qx^t(\delta(S)) \geq 2 \sum_{i \in S} q_{it} \qquad S \subseteq V, t \in T \tag{2.8}$$

$$0 \leq x_e^t \leq 2 \qquad e \in \delta(0), t \in T \tag{2.9}$$

$$0 \leq x_e^t \leq 1 \qquad e \in E \setminus \delta(0), t \in T \tag{2.10}$$

$$s_{it} \geq 0 \qquad i \in V_0, t \in T \tag{2.11}$$

$$q_{it} \geq 0,\ 0 \leq y_{it} \leq 1 \qquad i \in V, t \in T \tag{2.12}$$

$$0 \leq y_{0t} \leq m \qquad t \in T \tag{2.13}$$

$$x \text{ and } y \text{ integer} \tag{2.14}$$

Objective (2.1) minimizes the total inventory holding and transportation costs during the planning horizon. Constraints (2.2) and (2.3) are inventory balance equations for the supplier and the retailers, respectively. Constraints (2.4) and (2.5) set the values of initial and ending inventories. Constraints (2.6) ensure that inventory amounts do not exceed the maximum levels at the retailers. Constraints (2.7) are the degree constraints. They require two adjacent edges of a node to be traversed if the node is visited in a given period. Constraints (2.8) are capacity constraints and they eliminate subtours (it is possible to have a subtour on nodes in a set S for which $q_{it} = 0$ and $y_{it} = 1$ for $i \in S$, however there exists an optimal solution where this does not occur). The remaining constraints are variable restrictions.

Chapter 3

Relaxations and Valid Inequalities

Let X be the set of feasible solutions of the inventory routing problem. In this chapter, we study the lot sizing, single node flow set and vehicle routing relaxations of the inventory routing problem. We first discuss the valid inequalities from the literature for the relaxations and generalize them. Then, we strengthen some of these valid inequalities. We also propose new families of valid inequalities for set X.

3.1 Lot Sizing Relaxations

IRP has the following multi-item capacitated lot sizing relaxation with stock bounds and cumulative big bucket capacity constraints:

$$s_{i,t-1} + q_{it} = d_{it} + s_{it} \qquad i \in V, t \in T$$
$$s_{i0} = \bar{s}_{i0} \qquad i \in V$$
$$s_{in} = 0 \qquad i \in V'$$
$$s_{it} + d_{it} \leq u_i \qquad i \in V, t \in T$$
$$q_{it} \leq Q y_{it} \qquad i \in V, t \in T \tag{3.1}$$
$$\sum_{i \in V} q_{i,1t} \leq \bar{s}_{00} + r_{1t} \qquad t \in T \tag{3.2}$$
$$s_{it}, q_{it} \geq 0, y_{it} \in \{0,1\} \qquad i \in V, t \in T.$$

Constraints (3.1) are special cases of (2.8) for $S = \{i\}$ since $x^t(\delta(i)) = 2y_{it}$ for $i \in V$ and $t \in T$. We obtain constraints (3.2) after rewriting the balance equations at the supplier using $s_{0t} = \bar{s}_{00} + r_{1t} - \sum_{i \in V} q_{i,1t}$ for $t \in T$ and projecting out these stock variables. If $m = n$, then the set of solutions to the lot sizing problem is the projection of X on the space y, s and q (given any solution that satisfies the above system, one can make a dedicated route for each customer node visited in each period to obtain a solution in set X). Clearly, any valid inequality for the lot sizing set is also valid for X.

We first strengthen the variable upper bound constraints (3.1).

Proposition 1 *The following variable upper bound constraints are valid for set X:*

$$q_{i1} \leq \min\{Q, \bar{d}_{i,1n}, u_i - \bar{s}_{i0}, \bar{s}_{00} + r_1 - \sum_{j \in V \setminus \{i\}} \bar{d}_{j11}, \bar{s}_{00} + r_{1n} - \sum_{j \in V \setminus \{i\}} \bar{d}_{j1n}\} y_{i1} \quad i \in V'$$

$$q_{i1} \leq \min\{Q, u_i - \bar{s}_{i0}, \bar{s}_{00} + r_1 - \sum_{j \in V \setminus \{i\}} \bar{d}_{j11}, \bar{s}_{00} + r_{1n} - \sum_{j \in V \setminus \{i\}} \bar{d}_{j1n}\} y_{i1} \quad i \in V \setminus V'$$

$$q_{it} \leq \min\{Q, u_i, \bar{d}_{i,1n} - \bar{d}_{i1,t-1}, d_{i,tn}, d_{i,1,t-1} + u_i - \bar{s}_{i0}, \bar{s}_{00} + r_{1t} - \sum_{j \in V \setminus \{i\}} \bar{d}_{j1t} - \bar{d}_{i1,t-1},$$
$$\bar{s}_{00} + r_{1n} - \sum_{j \in V \setminus \{i\}} \bar{d}_{j1n} - \bar{d}_{i,1,t-1}\} y_{it} \quad i \in V', 2 \leq t \leq n$$

$$q_{it} \leq \min\{Q, u_i, d_{i,1,t-1} + u_i - \bar{s}_{i0}, \bar{s}_{00} + r_{1t} - \sum_{j \in V \setminus \{i\}} \bar{d}_{j1t} - \bar{d}_{i1,t-1},$$
$$\bar{s}_{00} + r_{1n} - \sum_{j \in V \setminus \{i\}} \bar{d}_{j1n} - \bar{d}_{i,1,t-1}\} y_{it} \quad i \in V \setminus V', 2 \leq t \leq n.$$

Proof. One can see that these are valid since $q_{i,1t} \leq d_{i,1,t-1} + u_i - \bar{s}_{i0}$ is implied by $s_{it} = \bar{s}_{i0} + q_{i,1t} - d_{i,1t} \leq u_i - d_{it}$. The explanations for the other terms are obvious and thus omitted. □

Let a_{it} be the coefficient of y_{it} in the above inequalities. The lot sizing problem has several relaxations for which strong valid inequalities are studied. For each $i \in V$, the set of solutions to

$$\begin{aligned}
& s_{i,t-1} + q_{it} = d_{it} + s_{it} && t \in T \\
& s_{i0} = \bar{s}_{i0} \\
& s_{it} + d_{it} \leq u_i && t \in T \\
& q_{it} \leq a_{it} y_{it} && t \in T \\
& s_{it}, q_{it} \geq 0, y_{it} \in \{0,1\} && t \in T
\end{aligned} \tag{3.3}$$

is a lot sizing set with stock upper bounds and capacities. Pochet and Wolsey [20] provide the convex hull description of the special case with Wagner Whitin costs (can be modelled in the space of s and y) and constant capacities where the initial inventory is not fixed. They allow the stock bounds to be time variant, which is also our case since $s_{it} + d_{it} \leq u_i$ is equivalent to $s_{it} \leq u_{it} = u_i - d_{it}$ and $u_{i0} = \bar{s}_{i0}$. The nontrivial inequalities required in the description can be adapted to our case as follows:

$$y_{i,kl} \geq \left\lceil \frac{\bar{d}_{i,kl}}{\min_{j=k,\ldots,l} a_{ij}} \right\rceil \qquad i \in V, 1 \leq k \leq l \leq n : \bar{d}_{i,kl} > 0 \tag{3.4}$$

When the stock bounds and capacities are relaxed, we have an uncapacitated lot sizing set, for which Barany et al. [21] show that the only nontrivial facet defining inequalities of the convex hull are (l, S) inequalities:

$$\sum_{j \in \{1,\ldots,l\} \setminus S} q_{ij} + \sum_{j \in S} d_{i,jl} y_{ij} \geq \bar{d}_{i,1l}$$
$$i \in V, l \in T : \bar{d}_{i,1l} > 0, S \subseteq \{1, \ldots, l\}, S \neq \emptyset. \tag{3.5}$$

Atamtürk and Küçükyavuz [22] study the convex hull of the lot sizing set with stock upper bounds where initial and final stocks are not fixed and there are no capacities on the deliveries and they present families of facet defining inequalities. We can modify their inequalities by incorporating the delivery capacities to obtain the following families of valid inequalities:

$$\sum_{j \in \{1,\ldots,l\} \setminus S} q_{ij} + \sum_{j \in S} \min\{a_{ij}, \bar{d}_{i,1l}, d_{i,jl}\} y_{ij} \geq \bar{d}_{i,1l}$$
$$i \in V, l \in T : \bar{d}_{i,1l} > 0, S \subseteq \{1, \ldots, l\}, S \neq \emptyset \tag{3.6}$$

$$\sum_{j \in \{k,\ldots,l\} \setminus S} q_{ij} + \sum_{j \in S} \min\{a_{ij}, d_{i,k-1,j-1}, \bar{d}_{i,kl}, d_{i,jl}\} y_{ij} \geq \bar{d}_{i,kl}$$

$$i \in V, 2 \leq k \leq l \leq n : \bar{d}_{i,kl} > 0, S \subseteq \{k,\ldots,l\}, S \neq \emptyset \qquad (3.7)$$

$$\sum_{j \in \{k,\ldots,n\} \setminus S} q_{ij} + \sum_{j \in S} \min\{a_{ij}, d_{i,k-1,j-1}\} y_{ij} \geq d_{i,k-1,n} - u_i + s_{in}$$

$$i \in V \setminus V', 2 \leq k \leq n, S \subseteq \{k,\ldots,n\} \qquad (3.8)$$

Note that inequality (3.6) is at least as strong as (3.5). Also if $s_{in} = 0$, then (3.8) is dominated by (3.7). This is why we present (3.8) only for $i \in V \setminus V'$.

The lot sizing based inequalities in the literature are either equivalent to or are dominated by the inequalities presented above. In particular, let $k = min\{i \in S\}$. Then using $q_{i,1k-1} = s_{i,k-1} + d_{i,1k-1} - \bar{s}_{i0}$, the (l, S) inequality (3.5) can be written as

$$s_{i,k-1} + \sum_{j \in \{k,\ldots,l\} \setminus S} q_{ij} + \sum_{j \in S} d_{i,jl} y_{ij} \geq d_{i,kl}. \qquad (3.9)$$

Archetti et al. [15] propose valid inequalities for the problem with a single vehicle. Their stock variables are defined for the beginning of periods. Below, in presenting their valid inequalities, we modify them to use the end of period inventories. The authors prove that inequalities

$$s_{i,k-1} \geq (1 - y_{ik}) d_{ik} \qquad i \in V, k \in T \qquad (3.10)$$

are valid inequalities for X. These are (l, S) inequalities with $k = l$ and $S = \{k\}$.

They also prove that inequalities

$$s_{i,l-t-1} \geq \left(\sum_{j=0}^{t} d_{i,l-j}\right)\left(1-\sum_{j=0}^{t} y_{i,l-j}\right) \qquad i \in V, l \in T, t = 0,1,\ldots,l-1 \tag{3.11}$$

are valid for X. These inequalities can also be written as

$$s_{i,k-1} \geq d_{i,kl} - \sum_{j=k}^{l} d_{i,kl} y_{ij} \qquad i \in V, l \in T, k = 1,\ldots,l \tag{3.12}$$

by letting $k = l - t$. Now one can see easily that this is dominated by the (l,S)-inequality (3.9) for $S = \{k,\ldots,l\}$, which reads $s_{i,k-1} \geq d_{i,kl} - \sum_{j=k}^{l} d_{i,jl} y_{ij}$.

Another family of valid inequalities proposed by Archetti et al. [15] is

$$y_{i,1l} \geq \left\lceil \frac{d_{i,1l} - \bar{s}_{i0}}{u_i} \right\rceil \qquad i \in V, l \in T. \tag{3.13}$$

These are dominated by (3.4).

A similar family of inequalities are given in Coelho and Laporte [18]:

$$y_{i,kl} \geq \left\lceil \frac{d_{i,kl} - u_i}{min\{Q, u_i\}} \right\rceil \qquad i \in V, k, l \in T : k \leq l \tag{3.14}$$

These are also dominated by (3.4).

3.2 Single Node Flow Set Relaxation

For each $1 \leq k \leq l \leq n$, the set of solutions to

$$\begin{aligned} & q_{ij} \leq a_{ij} y_{ij} \qquad (i,j) \in N^{kl} \\ & \sum_{(i,j)\in N^{kl}} q_{ij} \leq b_{kl} \\ & q_{ij} \leq 0, y_{ij} \in \{0,1\} \qquad (i,j) \in N^{kl} \end{aligned}$$

where $N^{kl} = \{(i,j) : i \in V, j \in \{k, \ldots, l\}\}$, $b_{kl} = \min\{\bar{s}_{00} + r_{1l} - \sum_{i\in V} \bar{d}_{i1,k-1}, Qm(l-k+1)\}$, is a single node flow set with only inflow arcs. Suppose that $b_{kl} + a_{ij} < \sum_{(i',j')\in N^{kl}} a_{i'j'}$ and $a_{ij} \leq b_{kl}$ for all $(i,j) \in N^{kl}$. Padberg et al. [23] prove that the following flow cover inequalities are valid for this set

$$\sum_{(i,j)\in N_S \cup N_L} q_{ij} \leq b_{kl} - \sum_{(i,j)\in N_S'} (a_{ij} - \lambda)(1 - y_{ij}) + \sum_{(i,j)\in N_L} (\bar{a}_{ij} - \lambda) y_{ij} \tag{3.15}$$

where $N_S \subset N^{kl}$ with $\lambda = \sum_{(i,j)\in N_S} a_{ij} - b_{kl} > 0$, $N_S' = \{(i,j) \in N_S : a_{ij} > \lambda\}$, $\tilde{a} = max_{(i,j)\in N_S} a_{ij}$, $N_L \subset N^{kl} \setminus N_S$ and $\bar{a}_{ij} = max\{\tilde{a}, a_{ij}\}$ for $(i,j) \in N_L$.

3.3 Vehicle Routing Relaxations

We consider two relaxations related with vehicle routing.

1. Routing for periods k to l: Let $1 \leq k \leq l \leq n$. Any valid inequality for the set

$$
\begin{aligned}
& q_{i,kl} \geq \bar{d}_{i,kl} \qquad i \in V \\
& Q \sum_{t=k}^{l} x^t(\delta(S)) \geq 2 \sum_{i \in S} q_{i,kl} \qquad S \subseteq V \\
& \sum_{t=k}^{l} x_e^t \in Z_+ \qquad e \in E
\end{aligned} \tag{3.16}
$$

   is also valid for X.

   Rounded capacity constraints

$$
\sum_{t=k}^{l} x^t(\delta(S)) \geq 2 \left\lceil \frac{\sum_{i \in S} \bar{d}_{ikl}}{Q} \right\rceil \qquad S \subseteq V, 1 \leq k \leq l \leq n \tag{3.17}
$$

   for $k = 1$ are proposed by Adulyasak et al. [14] and for $k \geq 2$ by Avella et al. [19] as valid inequalities. Inequality (3.17) is dominated by (3.4) for singletons, i.e., $S = \{i\}$ for some $i \in V$.

2. Routing in a single period: If we remove the constraints that link different periods, we end up with a vehicle routing relaxation for each period:

$$
\begin{aligned}
& x^t(\delta(i)) = 2y_{it} \qquad i \in V_0 \\
& q_{it} \leq a_{it} y_{it} \qquad i \in V \\
& Q x^t(\delta(S)) \geq 2 \sum_{i \in S} q_{it} \qquad S \subseteq V \\
& x_e^t \in \{0, 1, 2\} \qquad e \in \delta(0) \\
& x_e^t \in \{0, 1\} \qquad e \in E \setminus \delta(0) \\
& q_{it} \geq 0, \ y_{it} \in \{0, 1\} \qquad i \in V \\
& y_{0t} \in \{0, \ldots, m\}
\end{aligned}
$$

Constraints (2.8) can be strengthened as:

$$\min\{Q, \sum_{i \in S} a_{it}\} x^t(\delta(S)) \geq 2 \sum_{i \in S} q_{it} \qquad S \subseteq V. \tag{3.18}$$

Avella et al. [19] propose the following Simple DR inequalities as valid inequalities:

$$\sum_{\{i,j\} \in E} \mu_{ij} x_{ij}^t \geq 2 \Big(\sum_{i \in S_0 \cup S_1} q_{it} + \sum_{i \in S_2} (q_{it} - s_{i,t+1}) + \sum_{i \in S_3} (q_{it} - s_{it}) + \sum_{i \in S_4} (s_{it} - (u_i - d_{i,t-1,t})) \Big) \qquad i \in V, t \in T \tag{3.19}$$

where $(S_0, S_1, S_2, S_3, S_4)$ are partition of $S \subseteq V$ and $\mu_{ij} = max\{min\{Q - v_i, v_j\}, min\{Q - v_j, v_i\}\}$ such that $v_i = Q$ for $i \in S_0$, $v_i = u_i$ for $i \in S_1$, $v_i = d_{i,t,t+1}$ for $i \in S_2$, $v_i = d_{it}$ for $i \in S_3$, $v_i = d_{i,t-1}$ for $i \in S_4$ and $v_i = 0$ for $i \in V_0 \setminus S$.

The following connectivity constraints

$$x^t(\delta(S)) \geq 2y_{it} \qquad S \subseteq V, i \in S, t \in T \tag{3.20}$$

are well known. There exists an optimal solution to the IRP that satisfies these constraints.

Archetti et al. [15] propose the following valid inequalities:

$$y_{it} \leq y_{0t} \qquad i \in V, t \in T \tag{3.21}$$

$$x_{0i}^t \leq 2y_{it} \qquad i \in V, t \in T \tag{3.22}$$

$$x_{ij}^t \leq y_{it} \qquad i \in V, j \in V, t \in T \tag{3.23}$$

As Solyalı and Süral [12] also state, inequality (3.22) is implied by $x^t(\delta(i)) = 2y_{it}$ and nonnegativity of x_e for all $e \in E$. Inequality (3.23) is a special case of inequality (3.20). To see this, note that inequality (3.20) can be rewritten as

$$x^t(E(S)) \leq \sum_{k\in S} y_{kt} - y_{it} \qquad S \subseteq V, i \in S, t \in T$$

using $x^t(\delta(S)) = \sum_{k\in S} x^t(\delta(k)) - 2x^t(E(S)) = \sum_{k\in S} 2y_{kt} - 2x^t(E(S))$. For $S = \{i, j\}$ and $i \in S$, we obtain inequality (3.23). Inequality (3.21) is not implied in general, however its use may be limited as one may expect $y_{0t} \geq 1$ in many cases.

In the remainder of this section, we propose two families of valid inequalities that are called lifted flow cover inequalities for the single period relaxation when there is a single vehicle.

Proposition 2 *Let* $S \subseteq V$ *with* $\lambda = \sum_{i\in S} a_{it} - Q > 0$ *and* $S' = \{i \in S : a_{it} \geq \lambda\}$*. Let* $E' \subseteq E(S')$ *such that* (S', E') *does not contain any cycles. The inequality*

$$\sum_{i\in S} q_{it} + \lambda \sum_{\{i,j\}\in E'} (x^t_{\{i,j\}} + 1 - y_{it} - y_{jt}) + \sum_{i\in S'} (a_{it} - \lambda)(1 - y_{it}) \leq Q \qquad (3.24)$$

is a valid inequality for X *when* $m = 1$.

Proof. Given a feasible solution, define $S^0 = \{i \in S' : y_{it} = 0\}$, $E^k = \{e \in E' : |e \cap S^0| = k\}$ for $k = 0, 1, 2$. Then

$$\begin{aligned}
&\sum_{i\in S} q_{it} + \lambda \sum_{\{i,j\}\in E'} (x^t_{\{i,j\}} + 1 - y_{it} - y_{jt}) + \sum_{i\in S'} (a_{it} - \lambda)(1 - y_{it}) \\
&= \sum_{i\in S\setminus S^0} q_{it} + \lambda \left(\sum_{\{i,j\}\in E'} x^t_{\{i,j\}} + |E'| - 2|E^0| - |E^1| - |S^0| \right) + \sum_{i\in S^0} a_{it} \\
&= \sum_{i\in S\setminus S^0} q_{it} + \lambda \left(\sum_{\{i,j\}\in E'} x^t_{\{i,j\}} + |E^2| - |E^0| - |S^0| \right) + \sum_{i\in S^0} a_{it}
\end{aligned}$$

Now, using $\sum_{i \in S \setminus S^0} q_{it} + \sum_{i \in S^0} a_{it} \leq \sum_{i \in S} a_{it} = Q + \lambda$, $\sum_{\{i,j\} \in E'} x^t_{\{i,j\}} \leq |E^0|$ (since for an edge to be used, both of its endpoints must be visited) and $|E^2| \leq |S^0| - 1$ (since the subgraph is acyclic), we can see that

$$\sum_{i \in S \setminus S^0} q_{it} + \lambda \left(\sum_{\{i,j\} \in E'} x^t_{\{i,j\}} + |E^2| - |E^0| - |S^0| \right) + \sum_{i \in S^0} a_{it} \leq Q$$

□

Proposition 3 *Let $S \subseteq V$ and $E' \subseteq E(S)$ such that (S, E') does not contain any cycles and for each $\{i,j\} \in E'$, we have $\sum_{i' \in S \setminus \{i,j\}} a_{i't} < Q$, $\sum_{i' \in S \setminus \{i\}} a_{i't} \geq Q$ and $\sum_{i' \in S \setminus \{j\}} a_{i't} \geq Q$. The inequality*

$$\sum_{i \in S} q_{it} + \sum_{\{i,j\} \in E'} (Q - \sum_{i' \in S \setminus \{i,j\}} a_{i't})(x^t_{\{i,j\}} + 1 - y_{it} - y_{jt}) \leq Q \tag{3.25}$$

is a valid inequality for X when $m = 1$.

Proof. We define $S^0 = \{i \in S : y_{it} = 0\}$ and $E^2 = \{e \in E' : |e \cap S^0| = 2\}$. Note that $x^t_{\{i,j\}} + 1 - y_{it} - y_{jt}$ can be positive at a feasible solution only if $\{i,j\} \in E^2$. Hence

$$\begin{aligned}
&\sum_{i \in S} q_{it} + \sum_{\{i,j\} \in E'} (Q - \sum_{i' \in S \setminus \{i,j\}} a_{i't})(x^t_{\{i,j\}} + 1 - y_{it} - y_{jt}) \\
&\leq \sum_{i \in S \setminus S^0} q_{it} + \sum_{e \in E^2} (Q - \sum_{i' \in S \setminus e} a_{i't})
\end{aligned}$$

If $E^2 = \emptyset$, then as $\sum_{i \in S \setminus S^0} q_{it} \leq Q$, inequality (3.24) is satisfied.

Now suppose that E^2 is not empty. Let j^* be a vertex in set S^0 with the largest a_{jt} value. Now as the subgraph (S^0, E^2) is acyclic, we root it at vertex j^* and orient its edges in such a way that each vertex in S^0 has at most one incoming arc. In this orientation, if edge $e = \{i,j\}$ has been oriented from i

to j, then we let $\hat{i}(e) = i$ and $\hat{j}(e) = j$. Then for $e \in E^2$, $Q - \sum_{i' \in S \setminus e} a_{i't} = Q - \sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't} + a_{\hat{j}(e)t}$. Hence

$$\sum_{i \in S \setminus S^0} q_{it} + \sum_{\{i,j\} \in E^2} (Q - \sum_{i' \in S \setminus \{i,j\}} a_{i't}) = \sum_{i \in S \setminus S^0} q_{it} + \sum_{e \in E^2} (Q - \sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't}) + \sum_{e \in E^2} a_{\hat{j}(e)}$$

We know that each node in S^0 has indegree of at most one and j^* has no incoming arc. Hence $\sum_{e \in E^2} a_{\hat{j}(e)} \leq \sum_{i \in S^0 \setminus \{j^*\}} a_{it}$. Combining this with $\sum_{i \in S \setminus S^0} q_{it} \leq \sum_{i \in S \setminus S^0} a_{it}$, we get that

$$\sum_{i \in S \setminus S^0} q_{it} + \sum_{e \in E^2} (Q - \sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't}) + \sum_{e \in E^2} a_{\hat{j}(e)} \leq \sum_{i \in S \setminus \{j^*\}} a_{it} + \sum_{e \in E^2} (Q - \sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't}).$$

To prove the validity of inequality (3.24), it remains to show that

$$\sum_{i \in S \setminus \{j^*\}} a_{it} + \sum_{e \in E^2} (Q - \sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't}) \leq Q$$

or equivalently,

$$\sum_{i \in S \setminus \{j^*\}} a_{it} - Q \leq \sum_{e \in E^2} \left(\sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't} - Q \right).$$

Since $E^2 \neq \emptyset$, $a_{j^*t} \geq a_{\hat{i}(e)t}$ and $\sum_{i' \in S \setminus \{\hat{i}(e)\}} a_{i't} - Q \geq 0$ for all $e \in E^2$, the inequality is satisfied. $\square$

Chapter 4

Branch and Cut Algorithm

We propose a branch and cut algorithm in order to solve the problem. The current formulation of the problem is not compact since it has exponential number of constraints because of the capacity constraints. Hence, we start with the relaxation of the formulation, solve this relaxation and then add the constraints to the relaxation if they are violated by the current solution of the relaxation. This process is repeated until the optimal solution for the IRP is obtained.

We implement the initial mathematical model introduced in Chapter 2 excluding capacity constraints (3.20) and including some of the valid inequalities described in Chapter 3. Capacity constraints (3.20) and valid inequalities whose numbers are exponential in the size of the problem are dynamically added to the relaxation with respect to their specific fashions. Other valid inequalities are directly added to the relaxation.

In the sequel, we assume that the costs are such that zero cost subtours do not arise.

We start with the following relaxation:

$$\min \sum_{i\in V_0}\sum_{t\in T} h_{it}s_{it} + \sum_{t\in T}\sum_{e\in E} c_e x_e^t$$

s.t. (2.2)-(2.7), (2.9)-(2.13)

$$q_{it} \leq a_{it}y_{it} \qquad i\in V, t\in T$$

$$x_e^t \leq y_{it} \qquad e\in E, i\in e, t\in T$$

$$y_{i,kl} \geq \left\lceil \frac{\bar{d}_{i,kl}}{\min_{j=k,\dots,l} a_{ij}} \right\rceil \qquad i\in V, 1\leq k\leq l\leq n : l = min\{t\in T : \bar{d}_{i,kl} > 0\}$$

and $y_{it} \leq y_{0t}$ for all $i\in V, t\in T$ if $m=1$.

Let $(\bar{x},\bar{y},\bar{q},\bar{s})$ be an optimal solution of the current relaxation.

For each period $t\in T$, we construct the support graph $\bar{G}^t = (\bar{V}^t, \bar{E}^t)$ where $\bar{V}^t = \{i\in V_0 : \bar{y}_{it} > 0\}\cup\{v+1\}$ and $\bar{E}^t = \{e\in E : \bar{x}_e^t > 0\}\cup\{\{i, v+1\} : i\in \bar{V}^t\setminus\{v+1\}\}$ and let S_k^t, $k = 0,1,...,k^t$ be the node sets of connected components of $\bar{G}^t$ with $0\in S_0^t$.

We consider two cases

1. *The solution is integral;*

   If $k^t = 0$ for each period $t\in T$ and vehicle capacity is not exceeded, then the solution is optimal for the IRP.

   If $k^t = 0$, the routes of vehicles are well defined. For vehicle $j\in\{1,\dots,m\}$, let S_{0j}^t be the set of the retailers on its route in period t. If $Q < \sum_{i\in S_{0j}^t\setminus\{0\}} q_{it}$ then the capacity constraint is violated and we add

$$\min\{Q, \sum_{i\in S_{0j}^t\setminus\{0\}} a_{it}\} x^t(\delta(S_{0j}^t\setminus\{0\})) \geq 2\sum_{i\in S_{0j}^t\setminus\{0\}} q_{it}$$

to the relaxation.

If there exists a period $t \in T$ with $k^t \geq 1$, then capacity constraints (2.8) and connectivity constraints (3.20) are violated in that period. Therefore, for each $t \in T$ with $k^t \geq 1$ and $k = 1, ..., k^t$, cuts

$$\min\{Q, \sum_{i \in S_k^t} a_{it}\} x^t(\delta(S_k^t)) \geq 2 \sum_{i \in S_k^t} q_{it}$$

and then randomly selected 15 of

$$x^t(\delta(S_k^t)) \geq 2y_{it} \quad i \in S_k^t$$

are added to the relaxation. The number is selected as 15 based on preliminary analysis. When the numbers higher than 15 are selected, it is observed that the execution slows down in general.

2. *The solution is fractional;*

   If the current node is root node, separation algorithms described in the next sections are used. Separation of the inequalities are performed only at the root node in order to prevent the execution from slowing down. All of the violated inequalities are added to the relaxation until no more violated cut is found for each type of inequality.

4.1 Separation Algorithms

The valid inequalities (3.6), (3.7) & (3.8) ((l, S)-like inequalities), (3.18) (capacity constraints), (3.20) (connectivity constraints), (3.17) (rounded capacity constraints), (3.15) (flow cover inequalities), (3.24) & (3.25) (lifted flow cover inequalities) and

(3.19) (simple DR inequalities) are added to the relaxation if they are violated. Separation algorithms are used in order to detect the violated cuts. Different separation methods are used for each inequality type as shown in the following table.

Table 4.1: Separation Methods for the Inequalities

Heuristic	Exact	Inspection	MIP
Flow Cover	Capacity	(l, S)-like	Rounded Capacity
Lifted Flow Cover	Connectivity		Simple DR

4.1.1 Separation of (l, S)-like Inequalities with Stock Bounds

- For each $i \in V$ and for each $k, l \in T$ such that $1 \leq k \leq l \leq n$, following algorithms are implemented

  If $k = 1$ and $\bar{d}_{i,1l} > 0$,
  Set $S = \{j \in \{1, \ldots, l\} : \min\{a_{ij}, \bar{d}_{i,1l}, d_{i,jl}\}\bar{y}_{ij} < \bar{q}_{ij}\}$

  If $\sum_{j \in \{1,\ldots,l\} \setminus S} \bar{q}_{ij} + \sum_{j \in S} \min\{a_{ij}, \bar{d}_{i,1l}, d_{i,jl}\}\bar{y}_{ij} < \bar{d}_{i,1l}$, cuts

$$\sum_{j \in \{1,\ldots,l\} \setminus S} q_{ij} + \sum_{j \in S} \min\{a_{ij}, \bar{d}_{i,1l}, d_{i,jl}\} y_{ij} \geq \bar{d}_{i,1l}$$

  are added to the relaxation.

  If $k > 1$ and $\bar{d}_{i,kl} > 0$,
  Set $S = \{j \in \{k, \ldots, l\} : \min\{a_{ij}, d_{i,k-1,j-1}, \bar{d}_{i,kl}, d_{i,jl}\}\bar{y}_{ij} < \bar{q}_{ij}\}$

  If $\sum_{j \in \{k,\ldots,l\} \setminus S} \bar{q}_{ij} + \sum_{j \in S} \min\{a_{ij}, d_{i,k-1,j-1}, \bar{d}_{i,kl}, d_{i,jl}\}\bar{y}_{ij} < \bar{d}_{i,kl}$, cuts

$$\sum_{j \in \{k,\ldots,l\} \setminus S} q_{ij} + \sum_{j \in S} \min\{a_{ij}, d_{i,k-1,j-1}, \bar{d}_{i,kl}, d_{i,jl}\} y_{ij} \geq \bar{d}_{i,kl}$$

  are added to the relaxation.

- For each $i \in V \setminus V'$ and for each $k \in T$ such that $2 \leq k \leq n$, following algorithm is implemented

  Set $S = \{j \in \{k, \ldots, n\} : \min\{a_{i,j}, d_{i,k-1,j-1}\}\bar{y}_{ij} < \bar{q}_{ij}\}$

  If $\sum_{j\in\{k,\ldots,n\}\setminus S} \bar{q}_{ij} + \sum_{j\in S} \min\{a_{ij}, d_{i,k-1,j-1}\}\bar{y}_{ij} < d_{i,k-1,n} - u_i + \bar{s}_{in}$, cuts

$$\sum_{j\in\{k,\ldots,n\}\setminus S} q_{ij} + \sum_{j\in S} \min\{a_{ij}, d_{i,k-1,j-1}\}y_{ij} \geq d_{i,k-1,n} - u_i + s_{in}$$

  are added to the relaxation.

4.1.2 Separation of Capacity Constraints

If $k^t \geq 1$ for the corresponding support graph $\bar{G}^t$, it means that constraints (2.8) are violated in that period. Hence, for each $t \in T$ with $k^t \geq 1$ and $k = 1, ..., k^t$, cuts

$$\min\{Q, \sum_{i\in S_k^t} a_{it}\}x^t(\delta(S_k^t)) \geq 2\sum_{i\in S_k^t} q_{it}$$

are added to the relaxation.

If $k^t = 0$ for the corresponding support graph $\bar{G}^t$, i.e., $\bar{G}^t$ is connected, separation of constraint (2.8) is performed by solving a minimum cut problem on the graph $\bar{G}^t$. *MinSourceSinkCut* procedure from the Java graph theory(jgrapht) library is invoked to find a minimum cut between specified source and sink nodes for a given graph where capacities of arcs in the set $\{e \in E : \bar{x}_e^t > 0\}$ are $Q\bar{x}_e^t$ and arcs in the set $\{\{i, v+1\} : i \in \bar{V}^t \setminus \{v+1\}\}$ are $2\bar{q}_{it}$. Minimum cut problem with source node $v+1$ and sink node 0 is solved. If the capacity of the minimum cut is less than $2\sum_{i\in V} \bar{q}_{it}$, cut

$$\min\{Q, \sum_{i\in S\setminus\{v+1\}} a_{it}\}x^t(\delta(S \setminus \{v+1\})) \geq 2\sum_{i\in S\setminus\{v+1\}} q_{it}$$

is added to the relaxation with respect to the vertex partition $[S, \bar{V}^t \backslash S]$ corresponding to the solution of minimum cut problem.

4.1.3 Separation of Connectivity Constraints

If $k^t \geq 1$ for the corresponding support graph $\bar{G}^t$, it means that constraints (3.20) are violated in that period. Therefore, for each $t \in T$ with $k^t \geq 1$ and $k = 1, ..., k^t$, cuts

$$x^t(\delta(S_k^t)) \geq 2y_{it} \quad i \in S_k^t$$

are added to the relaxation.

If $k^t = 0$ for the corresponding support graph $\bar{G}^t$ i.e. $\bar{G}^t$ is connected, separation of constraints (3.20) is performed by solving a minimum cut problem on the graph $\bar{G}^t$. *StoerWagnerMinimumCut* procedure from the Java graph theory(jgrapht) library is invoked to find a global minimum cut of a given graph where capacities of the arcs are $\bar{x}_e^t$. After global minimum cut is found, according to the vertex partition $[S, \bar{V}^t \setminus S]$ where $\{0\} \in \bar{V}^t \setminus S$ corresponding to the solution of global minimum cut problem, the cuts are added. For each $i \in S$ such that global min cut value is less than $2\bar{y}_{it}$, cut

$$x^t(\delta(S)) \geq 2y_{it}$$

is added to the relaxation.

4.1.4 Separation of Rounded Capacity Constraints

The idea of the separation is based on Avella et al. [19]. The following mathematical model with a quadratic objective function is solved for all k and l such that $1 \leq k \leq l \leq n$ to detect whether the current solution violates any rounded capacity constraints.

$$z^*_{rcc} = \min \sum_{t=k}^{l} \sum_{i=1}^{v} \Big(\sum_{j:\{i,j\} \in E} \bar{x}_{ij,t} \alpha_i (1-\alpha_j) + \bar{x}_{0i,t} \alpha_i \Big) - 2\gamma \tag{4.1}$$

$$\text{s.t. } \gamma \leq \frac{\sum_{i \in V} \bar{d}_{i,kl} \alpha_i}{Q} + 1 - \epsilon \tag{4.2}$$

$$\frac{\sum_{i \in V} \bar{d}_{i,kl} \alpha_i}{Q} \leq \gamma \tag{4.3}$$

$$\alpha_i \in \{0,1\} \qquad i \in V \tag{4.4}$$

$$\gamma \geq 0 \text{ and integer} \tag{4.5}$$

where α_i is 1 if $i \in S$; 0 otherwise for all $i \in V$, γ represents the smallest integer greater than or equal to $\frac{\sum_{i \in S} \bar{d}_{ikl}}{Q}$ and ϵ is very small number. Constraints (4.2) and (4.3) represent the linearization of γ. The remaining constraints are variable restrictions.

The objective (4.1) represents the minimum value that can be obtained for the difference of left and right hand sizes of inequality (3.17) with respect to the current solution. If $z^*_{rcc} \geq 0$, then the current solution satisfies all rounded capacity constraints. If $z^*_{rcc} < 0$, then there exists at least one $S \subseteq V$ that violates inequality (3.17) for the current solution because inequality (3.17) is violated by $S \subseteq V$ that corresponds to the solution of above model.

In order to strengthen the relaxation, the cut

$$\sum_{t=k}^{l} x^t(\delta(S)) \geq 2 \left\lceil \frac{\sum_{i \in S} \bar{d}_{ikl}}{Q} \right\rceil$$

is added to the relaxation for the solution $S \subseteq V$ of above model and the corresponding periods k and l.

4.1.5 Separation of Flow Cover Inequalities

For each $k, l \in T$ such that $1 \leq k \leq l \leq n$, LP relaxation of the following problem is solved.

$$z^*_{fc} = \max \sum_{(i,j)\in N^{kl}} \alpha_{ij}\bar{q}_{ij} + \beta_{ij}a_{ij}(1-\bar{y}_{ij}) - \theta_{ij}(1-\bar{y}_{ij}) \tag{4.6}$$

$$\text{s.t.} \sum_{(i,j)\in N^{kl}} a_{ij}\alpha_{ij} - \lambda = b_{kl} \tag{4.7}$$

$$\alpha_{ij} - \beta_{ij} \geq 0 \qquad (i,j) \in N^{kl} \tag{4.8}$$

$$\lambda - \theta_{ij} \leq M(1-\beta_{ij}) \qquad (i,j) \in N^{kl} \tag{4.9}$$

$$\alpha_{ij}, \beta_{ij} \in \{0,1\} \qquad (i,j) \in N^{kl} \tag{4.10}$$

$$\theta_{ij} \geq 0 \qquad (i,j) \in N^{kl} \tag{4.11}$$

$$a_{max} \geq \lambda \geq \epsilon \tag{4.12}$$

where α_{ij} is 1 if $(i,j) \in N_S$ and 0 otherwise, β_{ij} is 1 if $(i,j) \in N'_S$ and 0 otherwise, a_{max} is $max\{a_{ij} : (i,j) \in N^{kl}\}$, θ_{ij} represents $\lambda\beta ij$ for $(i,j) \in N^{kl}$, M is a big number and ϵ is a small number. Constraints (4.7) assign the value of λ. Constraints (4.8) guarantee that N'_S is the subset of N_S. Constraints (4.9) assure the representation of θ. The remaining constraints are variable restrictions.

If there exists a solution, set $N_S = \{(i,j) \in N^{kl} : \alpha_{ij} \geq 0.5\}$ with respect to the corresponding solution. If $\tilde{a} \geq \lambda$, then it is checked that whether there is any violated flow cover inequality. λ becomes $\sum_{(i,j)\in N_S} a_{ij} - b_{kl}$ and N'_S is determined accordingly.

Let $N_L = \{(i,j) \in N \setminus N_S : \ (\bar{a}_{ij} - \lambda)\bar{y}_{ij} < \bar{q}_{ij}\}$.

If

$$\sum_{(i,j)\in N_S\cup N_L} \bar{q}_{ij} + \sum_{(i,j)\in N'_S} (a_{ij}-\lambda)(1-\bar{y}_{ij}) - \sum_{(i,j)\in N_L} (\bar{a}_{ij}-\lambda)\bar{y}_{ij} > b_{kl}$$

then add

$$\sum_{(i,j)\in N_S\cup N_L} q_{ij} \le b_{kl} - \sum_{(i,j)\in N'_S} (a_{ij}-\lambda)(1-y_{ij}) + \sum_{(i,j)\in N_L} (\bar{a}_{ij}-\lambda)y_{ij}$$

to the relaxation.

4.1.6 Separation of Simple DR Inequalities

The idea of the separation is based on Avella et al. [19].

Following model is solved for all subtours $S \subseteq V$ in the current solution if $Q \geq u_i \geq d_{it,t+1}$ for $i \in V$ and $t \in T \setminus \{n\}$. S_5 is set to $V_0 \setminus S$ with $v_i = 0$, $i \in S_5$.

$$z^*_{sdr} = \min \sum_{\{i,j\}\in E} \sum_{k=0}^{5} \sum_{l=0}^{5} \mu_{ij}\bar{x}^t_{ij}\alpha_{ik}\alpha_{jl} - 2\Big(\sum_{i\in V} \bar{q}_{it}(\alpha_{i0}+\alpha_{i1}) + \sum_{i\in V} (\bar{q}_{it}-\bar{s}_{i,t+1})\alpha_{i2} + \sum_{i\in V} (\bar{q}_{it}-\bar{s}_{it})\alpha_{i3} + \sum_{i\in V} (\bar{s}_{it}-(u_i-d_{i,t-1,t}))\alpha_{i4} \Big) \tag{4.13}$$

$$\text{s.t.} \sum_{k=0}^{5} \alpha_{ik} = 1 \qquad i \in V_0 \tag{4.14}$$

$$\alpha_{i5} = 1 \qquad i \in V_0 \setminus S \tag{4.15}$$

$$\alpha_{ik} \in \{0,1\} \qquad i \in V_0, k \in \{0,1,2,3,4,5\} \tag{4.16}$$

where α_{ik} is 1 if $i \in V_0$ is in S_k for $k \in \{0,\ldots,5\}$ and 0 otherwise. Constraints (4.14) imply the partition and constraints (4.15) define set S_5. The remaining constraints are variable restrictions.

If $z^*_{sdr} < 0$ and $\sum_{i \in \{1,2,3,4\}} |S_i| \neq 0$, corresponding inequality (3.19) is added to the relaxation. The inequality is dominated by capacity constraint when S_1, S_2, S_3 and S_4 are empty sets. Hence, inequality (3.19) is not added in that case.

4.1.7 One Vehicle Case

If the problem is solved for one vehicle, Inequalities (3.24) and (3.25) as well as other inequalities whose separation algorithms are presented above are used to strengthen the formulation.

4.1.7.1 Separation of Lifted Flow Cover Inequalities

Given a fractional solution and set S, to find a violated inequality (3.24) for this choice of S, we let $S' = \{i \in S : a_{it} \geq \lambda\}$ and $w_{\{i,j\}} = \bar{x}^t_{\{i,j\}} + 1 - \bar{y}_{it} - \bar{y}_{jt}$ for $\{i, j\} \in E(S')$. To find a violated inequality (3.25), we let $S' = \{i \in S : \sum_{i' \in S \setminus \{i\}} a_{i't} \geq Q\}$ and $w_{\{i,j\}} = (Q - \sum_{i' \in S \setminus \{i,j\}} a_{i't})(\bar{x}^t_{\{i,j\}} + 1 - \bar{y}_{it} - \bar{y}_{jt})$ for $\{i, j\} \in E(S')$. Then in both cases, we would like to find a subset $E' \subseteq E(S')$ such that (S', E') does not contain any cycles and $w(E')$ is maximized.

In our implementation, we use the following heuristic algorithm for each $t \in T$. We first add all nodes i with $\bar{y}_{it} = 1$ to set S. If $\sum_{i \in S} a_{it} \leq Q$, then among the remaining nodes, we find a node i with the smallest $\frac{1-\bar{y}_{it}}{a_{it}}$ value and add it to set S. We repeat this until $\sum_{i \in S} a_{it} > Q$.

If we can find an S such that $\lambda = \sum_{i \in S} a_{it} - Q > 0$ and $\max_{i \in S} a_{it} > \lambda$ then we let $S' = \{i \in S : a_{it} \geq \lambda\}$ and choose the edges of $E(S')$ using a greedy algorithm with weight $w_{\{i,j\}} = \bar{x}^t_{\{i,j\}} + 1 - \bar{y}_{it} - \bar{y}_{jt}$ such that $w_{\{i,j\}} \geq 0$ for $\{i, j\} \in E(S')$. In the greedy algorithm, we sort the edges in nonincreasing order of weights and add them to set E' as long as they do not form a cycle. If we add at least one edge, then we check the violation of inequality (3.24).

To separate inequalities (3.25), we let S to be the set found above and $S' = \{i \in S : \sum_{i' \in S \setminus \{i\}} a_{i't} \geq Q\}$. If S' contains at least two elements, then we set the weight $w_{\{i,j\}} = (Q - \sum_{i' \in S \setminus \{i,j\}} a_{i't})(\bar{x}^t_{\{i,j\}} + 1 - \bar{y}_{it} - \bar{y}_{jt})$ such that $w_{\{i,j\}} \geq 0$ for $\{i,j\} \in E(S')$ and apply the greedy algorithm. If not, then we add node i with the largest $\bar{q}_{it}$ value to set S and repeat. If we add at least one edge, then we check the violation of inequality (3.25).

Additionally, for each S for which violation of inequalities (3.24) or (3.25) is tested, N_S is set to S and it is checked whether inequality (3.15) is violated. Moreover, for each N_S for which violation of inequality (3.15) is tested, S is set to N_S and it is checked whether inequalities (3.24) or (3.25) are violated.

Chapter 5

Computational Results

We performed our computational experiments on test instances that consist of benchmark instances introduced in Archetti et al. [15] and randomly generated instances. Randomly generated instances have time variant retailer demands and delivered quantities to the supplier unlike benchmark instances. Detailed description of randomly generated instances can be found in Appendix A. Moreover, the instances have two groups as *low* and *high* according to the type of inventory holding costs. Inventory holding costs are in the interval $[0.01, 0.05]$ for *low* type instances, whereas they are in the interval $[0.1, 0.5]$ for *high* type instances. Hence, in the low type of instances, routing decisions have more impact on the results than replenishment decisions.

When the number of vehicles is more than one, capacity of each vehicle is taken as the nearest integer number to Q/m. The cost values in test instances satisfy triangle inequality.

Each instance is labeled in the way that represents *low* or *high*, number of retailers, number of periods and benchmark (*bench*) or generated (*gen*), for example, *high*30-6-*bench*. There are five instance sets for each type.

Computational experiments were performed to evaluate the effect of the valid inequalities on the solution times and gaps. The experiments were carried on 64-bit machine with Intel Xeon 2.60 GHz and 96 GB of RAM. The branch and cut algorithm was coded in Java by using CPLEX 12.7. Default strategies provided by CPLEX are used for variable selection and branching. Time limit is set to 1800 s. Detailed results for test instances can be found in Appendix B.

Figures 5.1-5.7 represent the status of the solver at the time limit when no inequalities and inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) are cumulatively used for the problem with different number of vehicles. The order of adding the inequalities was determined based on preliminary analysis. The number of instances that can be solved to optimality, cannot be solved but a feasible solution can be found, and no feasible solution can be found in the time limit are represented by *optimal*, *feasible* and *unknown*, respectively. Vertical axis shows the total number of the instances with *optimal*, *feasible* and *unknown* status for each instance type and horizontal axis shows the inequality that is used as well as previous inequalities.

Figures 5.1-5.2 demonstrate the status of the solver at the time limit when the inequalities are cumulatively used for the problem with one vehicle. Figure 5.3 demonstrates the status of the solver at the time limit when inequalities (3.24) & (3.25) are used as well as inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17) and (3.15) for larger instances with one vehicle. These larger instances are randomly generated with using the structure of benchmark instances. Figures 5.4-5.5 and 5.6-5.7 demonstrate the status of the solver at the time limit for the problem with two and three vehicles, respectively.

Figure 5.1: Results for Three Periods and One Vehicle

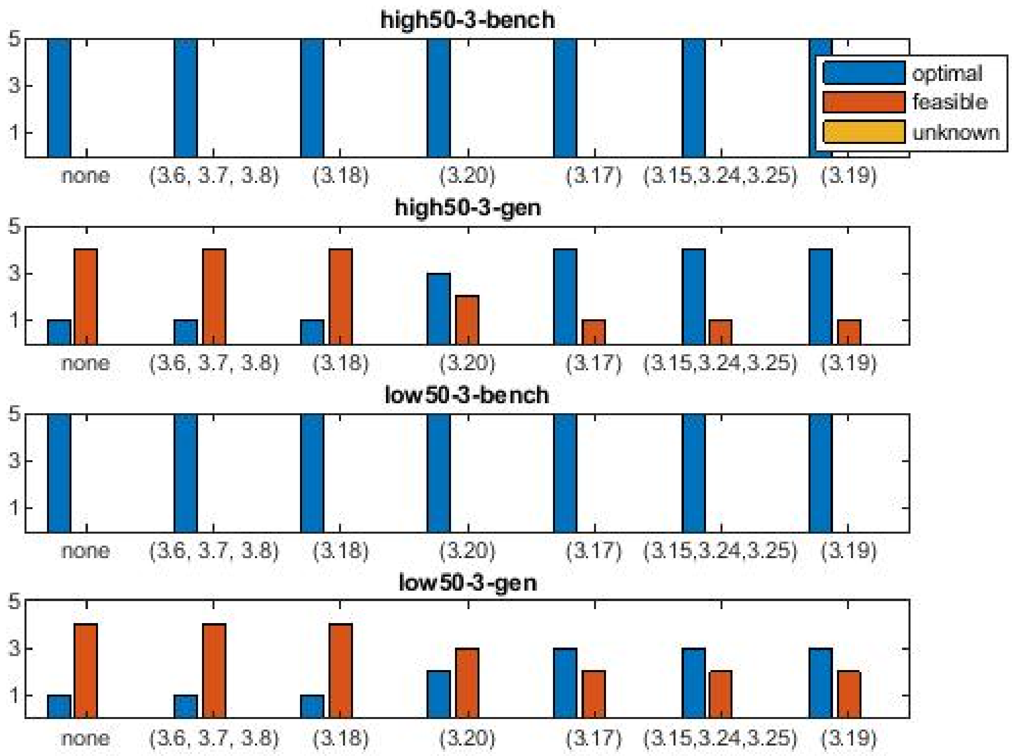


Figure 5.2: Results for Six Periods and One Vehicle

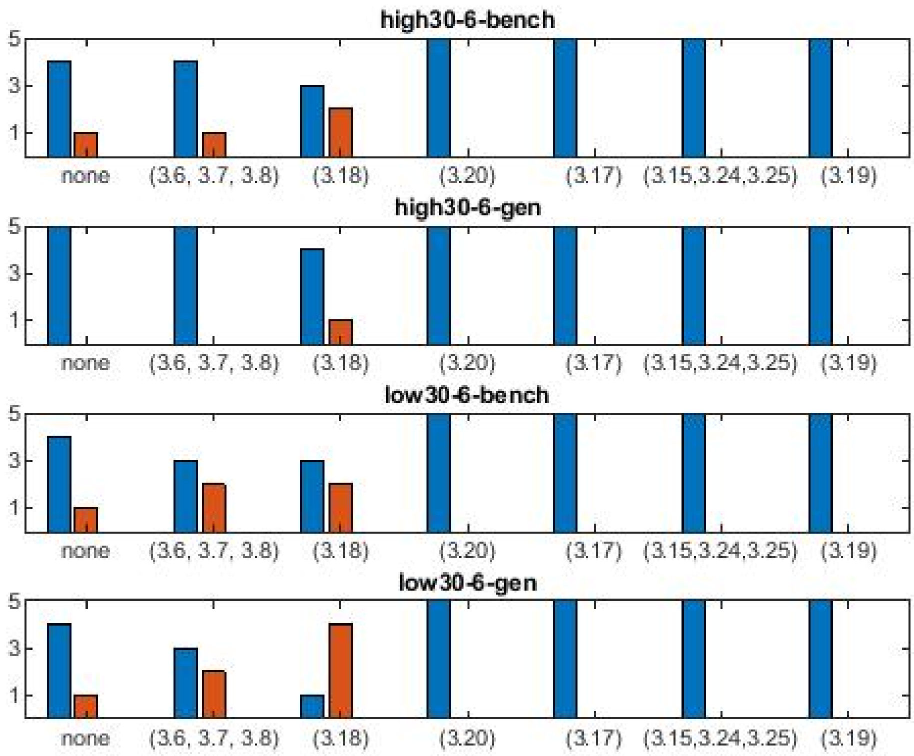
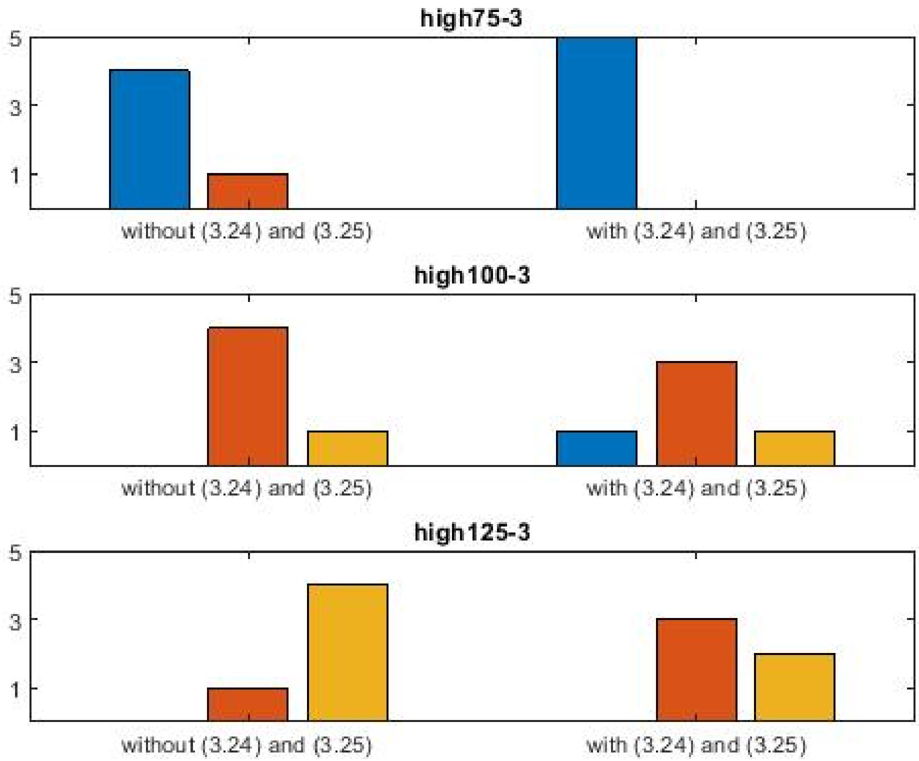


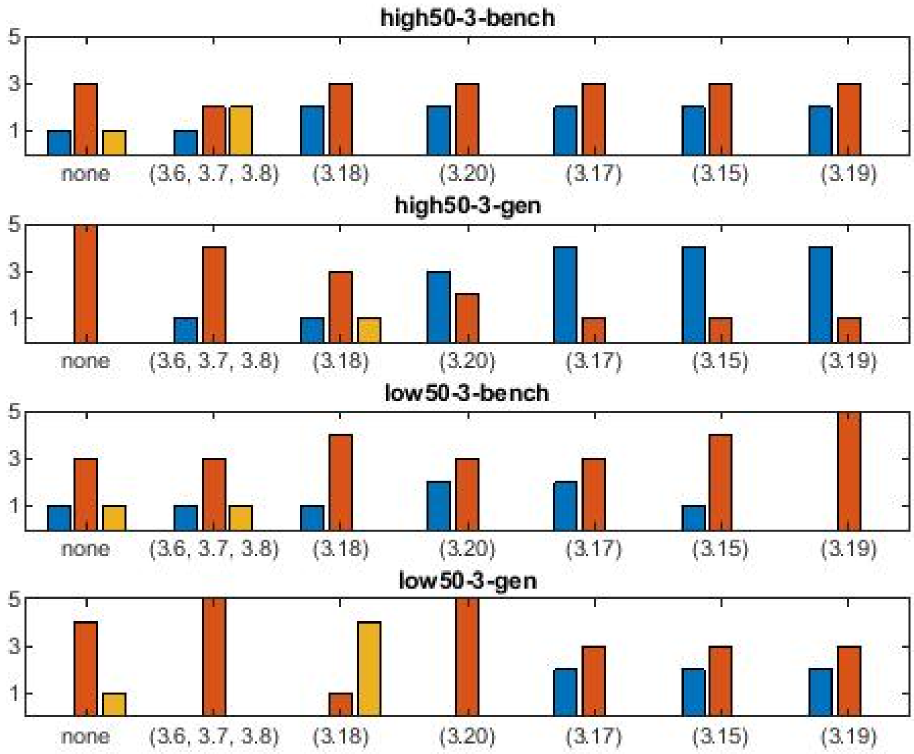
Figure 5.3: Results for Larger Instances with Three Periods and One Vehicle



According to Figures 5.1-5.2, using inequalities improves the status of the instances because some of the *feasible* instances can be solved to optimality after adding the inequalities. It can be said that the most effective inequality on the status is inequality (3.20) for one vehicle.

Since the status of the instances are stable after adding (3.20) or (3.17), the effect of (3.24) & (3.25) on the status cannot be clearly determined from Figures 5.1-5.2. Figure 5.3 shows that these inequalities improve the status of the large instances that are not solved to optimality or for which no feasible solution is found.

Figure 5.4: Results for Three Periods and Two Vehicles



Figures 5.4-5.7 show that using each valid inequality usually leads to improvements in the status of the instances for multi vehicle. It can be seen that the most effective inequalities are generally inequality (3.20) for time variant (generated) instances and inequality (3.17) for *low* instances.

Figure 5.5: Results for Six Periods and Two Vehicles

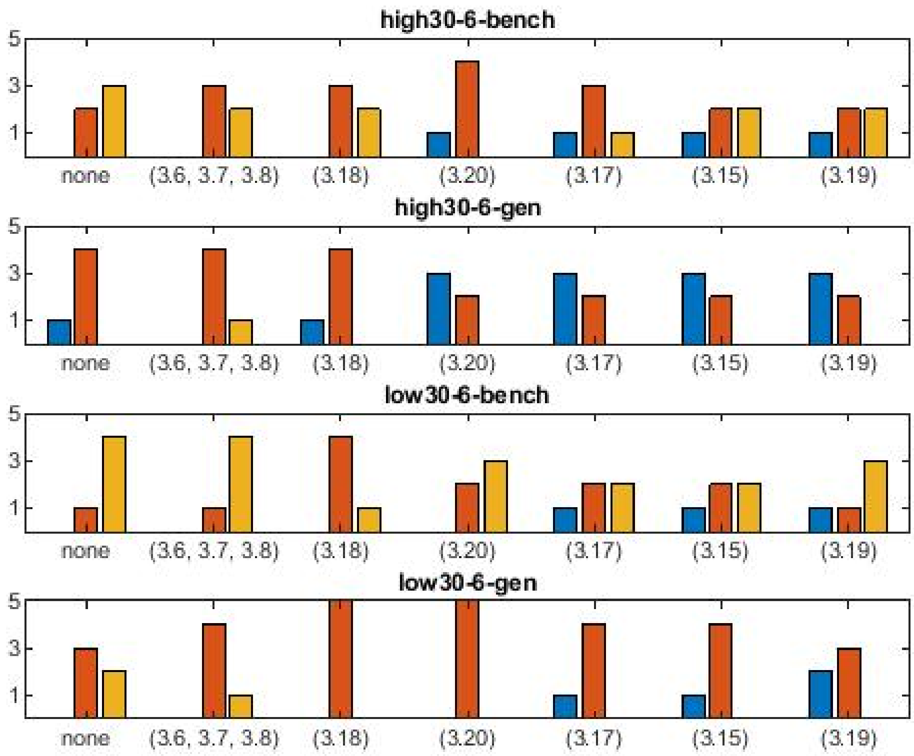


Figure 5.6: Results for Three Periods and Three Vehicles

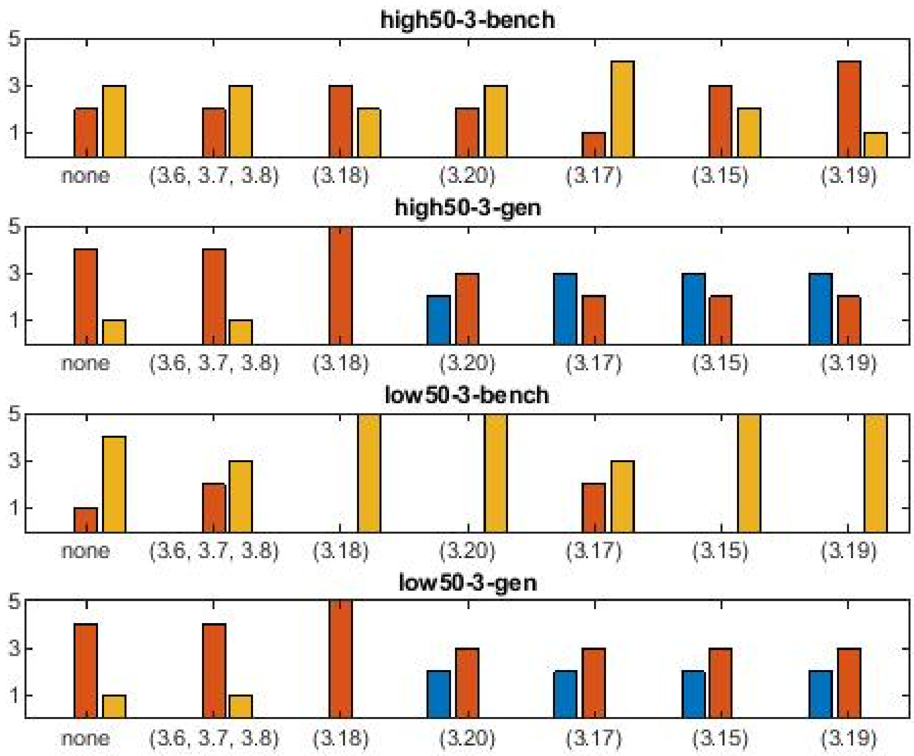
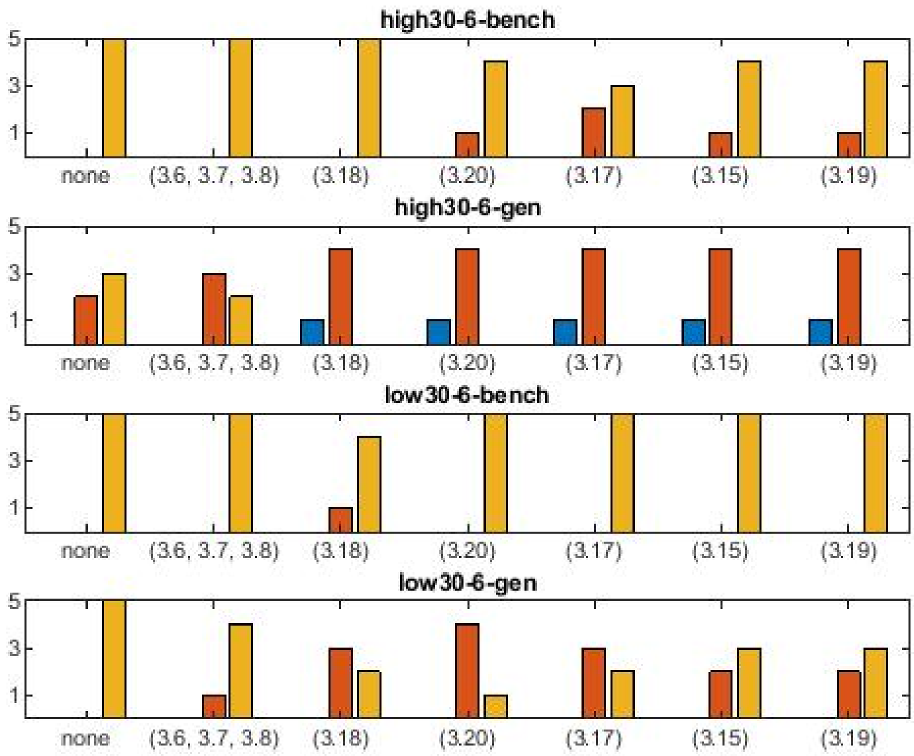


Figure 5.7: Results for Six Periods and Three Vehicles



If all inequalities are implemented, the total number of instances solved to optimality increases from 2 to 8 and 1 to 7 for three and six periods with two vehicles, respectively. None of the instances is solved to optimality if the inequalities are not used for three vehicles. After using all valid inequalities, the total number of instances with optimal status are 5 and 1 for three and six periods with three vehicles, respectively. The percentages of the number of instances solved to optimality can be seen in Table 5.1 if no inequalities and all inequalities are used for all instances.

Table 5.1: Percentages of The Number of Instances Solved to Optimality

m	Instance	n	*low/high*	w/o Ineq.	with Ineq.	w/o Ineq.	with Ineq.
1	B	3	*high*	100%	100%	77.5%	92.5%
			low	100%	100%		
		6	*high*	80%	100%		
			low	80%	100%		
	G	3	*high*	20%	80%		
			low	20%	60%		
		6	*high*	100%	100%		
			low	80%	100%		
2	B	3	*high*	20%	40%	7.5%	37.5%
			low	20%	0%		
		6	*high*	0%	20%		
			low	0%	20%		
	G	3	*high*	0%	80%		
			low	0%	40%		
		6	*high*	20%	60%		
			low	0%	40%		
3	B	3	*high*	0%	0%	0%	15%
			low	0%	0%		
		6	*high*	0%	0%		
			low	0%	0%		
	G	3	*high*	0%	20%		
			low	0%	0%		
		6	*high*	0%	60%		
			low	0%	40%		

Figures 5.8-5.10 represent the averages of percentage improvements in the optimality gap (Opt Gap), best lower bound (BLB) found in the time limit, the lower bound at root node (Root LB) and total time spent (CPU) while valid inequalities are cumulatively used for different number of vehicles. Vertical axis shows the improvements in percentages and horizontal axis shows the labels of the instances. Detailed results on the improvements can be found in Appendix C and D.

The percentages of the improvements for one vehicle are shown in Figure 5.8. The figure shows that inequalities (3.18), (3.20) and (3.17) are very effective to increase the Root LB for three periods while inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) have considerable effects and inequalities (3.15), (3.24) & (3.25) and (3.19) have small effects to increase the Root LB for six periods. Using inequalities (3.20) and (3.17) and these inequalities with inequality (3.18) are quite advantageous to decrease the CPU times for six and three periods, respectively. Inequality (3.20) is the most effective inequality to improve the Opt Gap and the BLB.

Figure 5.9 demonstrates the improvements for two vehicles. It is seen that using inequalities (3.20) and (3.17) decreases the CPU times and using these inequalities with inequality (3.18) improves the Root LB, Opt Gap and BLB. Inequality (3.15) and (3.19) are useful to improve Root LB and Opt Gap for three and six periods, respectively. Moreover, inequalities (3.6), (3.7) & (3.8) are effective to increase the Root LB for six periods.

The average improvements for three vehicles are represented in Figure 5.10. Using all inequalities is effective to increase BLB, but the impacts of inequalities (3.18), (3.20) and (3.17) are greater than impacts of other inequalities according to the figure. Using inequalities (3.18), (3.20) and (3.17) are advantageous to increase the Root LB for three periods while using these inequalities with inequalities (3.6), (3.7) & (3.8) is helpful to increase the Root LB for six periods. Inequalities (3.18), (3.20) and (3.17) are useful to decrease the CPU times. Using inequalities (3.18), (3.20)

Figure 5.8: Improvements for One Vehicle

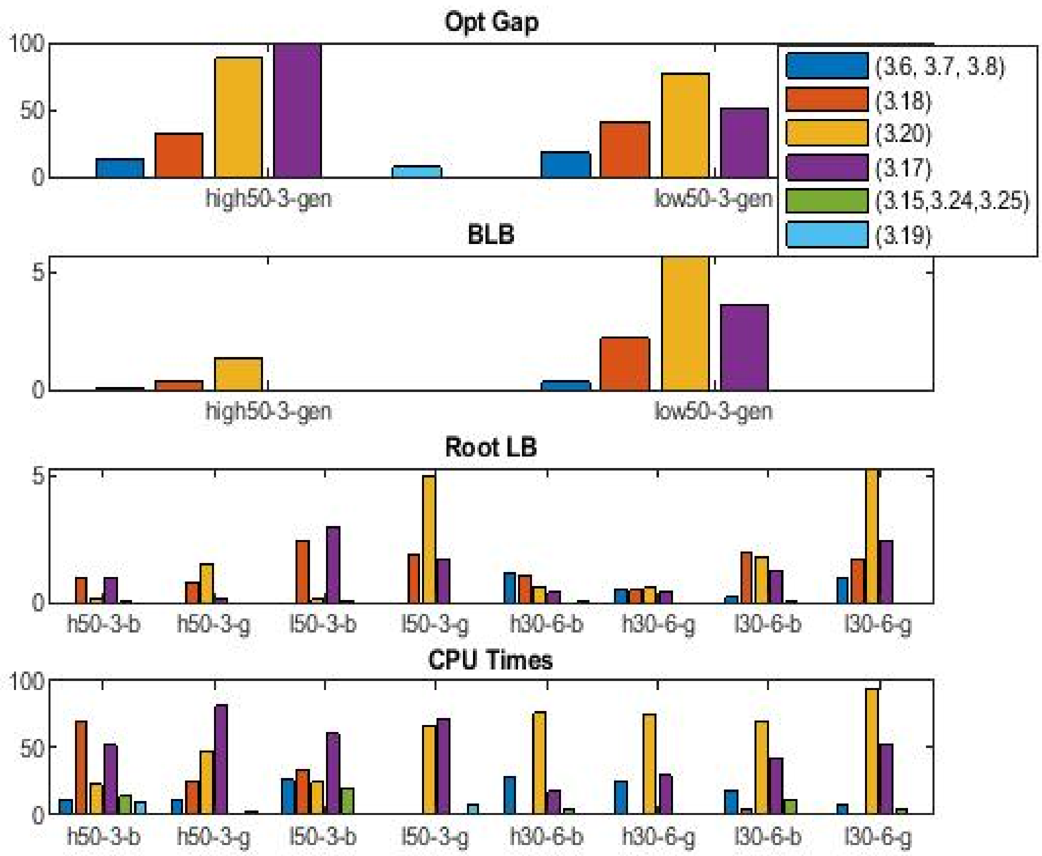


Figure 5.9: Improvements for Two Vehicles

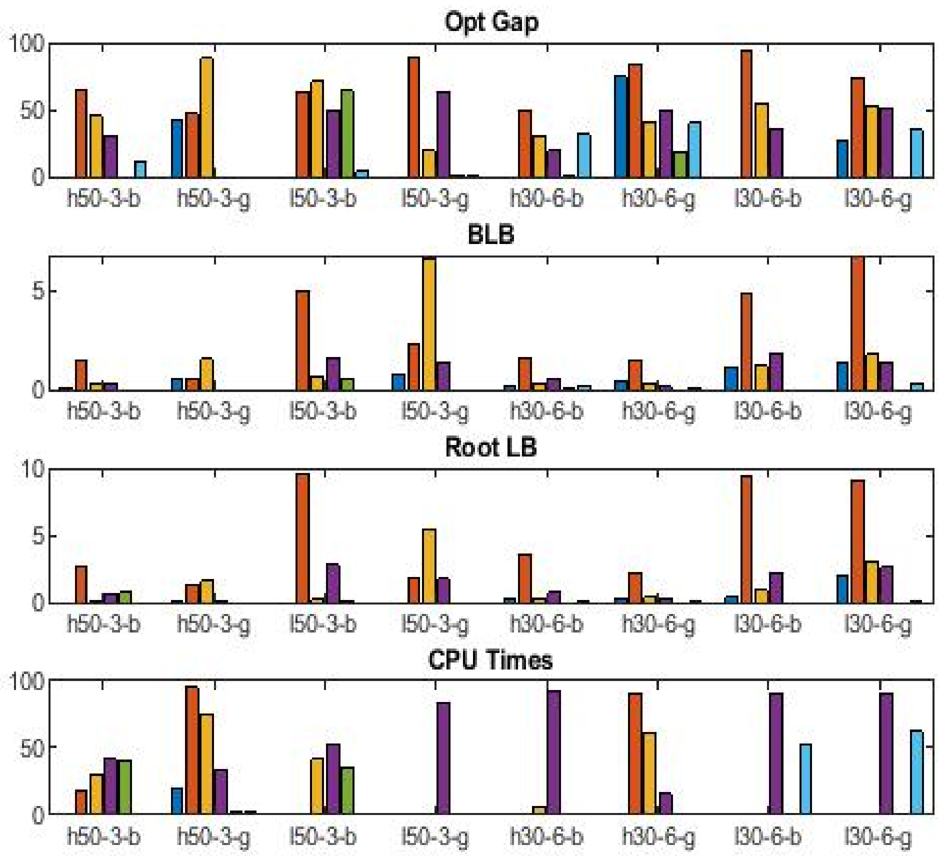
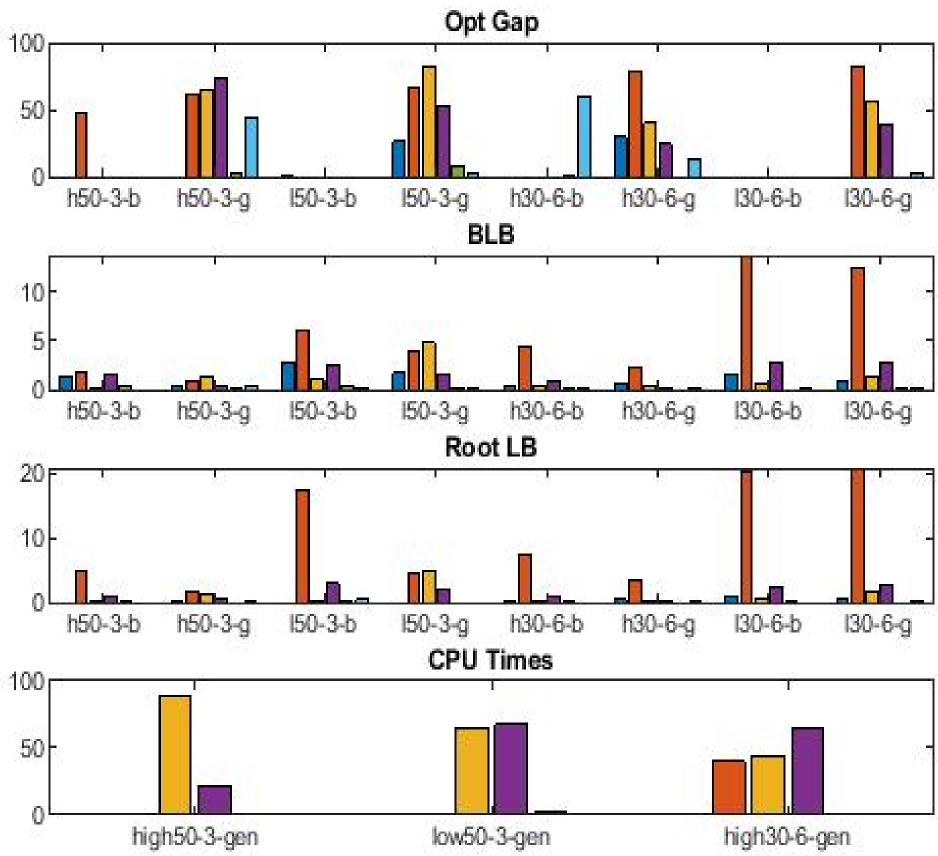


Figure 5.10: Improvements for Three Vehicles



and using these inequalities with inequality (3.17) and (3.19) improve the Opt Gap for three and six periods, respectively.

Since the frequency of finding violated inequalities (3.6), (3.7) & (3.8) for instances with three periods is rare, these inequalities usually have insignificant effects for instances with three periods. The effects of inequalities (3.15) and (3.24) & (3.25) are small for some of the instances because these inequalities are rarely added to the relaxations for these instances.

In general, inequalities (3.18), (3.20) and (3.17) have considerably greater effects than other inequalities for all categories. Specifically, (3.18) is more effective on time invariant (benchmark) instances, (3.20) is more effective on time variant (generated) instances and (3.17) is more effective on *low* instances.

Chapter 6

Conclusion and Future Research

In this thesis, we study the IRP which is one of the problems that arises in VMI. Combining inventory holding, distribution and transportation decisions into a problem and considering the system as a whole eliminate the potential costs that arise from independent decisions. This integrated structure that aims to minimize the total cost of the system provides more efficiency, however the problem becomes more complex.

We propose a mixed integer mathematical model for the IRP under ML replenishment policy. The model is given for a single product and a single supplier. We analyze the lot sizing, single node flow set and vehicle routing relaxations of the IRP. We present the valid inequalities for these relaxations in the literature and strengthen these inequalities. We also propose new valid inequalities for the IRP. We strengthen the initial mathematical model and propose a branch and cut algorithm. Different separation algorithms are implemented for the inequalities whose number is exponential in the size of the problem. While exact separation methods are proposed for inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17) and (3.19), heuristic methods are proposed for the separation of inequalities (3.15) and (3.24) & (3.25).

We generate test instances that have time variant demands and supplies. We use generated instances as well as benchmark instances, which are generated by Archetti et al. [15] and usually used in the literature for the IRP, to perform computational experiments. The branch and cut algorithm is tested on these instances. The impacts of the valid inequalities on the problem are stated and discussed in terms of the status of the solver, lower bound at the root node, cpu time, optimality gap and best lower bound obtained in the time limit.

This study can be extended in several ways. Heuristic methods for the separation of inequalities (3.15) and (3.24) & (3.25) can be improved. A heuristic method for the IRP can be designed to find good upper bounds for larger instances that cannot be solved to optimality. Production decisions can be added to the mathematical model and a branch and cut algorithm can be proposed for the PRP by performing some modifications on the current branch and cut algorithm for the IRP.

Bibliography

[1] A. J. Kleywegt, V. S. Nori, and M. W. Savelsbergh, "The stochastic inventory routing problem with direct deliveries," *Transportation Science*, vol. 36, no. 1, pp. 94–118, 2002.

[2] A. Campbell, L. Clarke, A. Kleywegt, and M. Savelsbergh, "The inventory routing problem," in *Fleet management and logistics*, pp. 95–113, Springer, 1998.

[3] P. Chandra, "A dynamic distribution model with warehouse and customer replenishment requirements," *Journal of the Operational Research Society*, vol. 44, no. 7, pp. 681–692, 1993.

[4] P. Chandra and M. L. Fisher, "Coordination of production and distribution planning," *European Journal of Operational Research*, vol. 72, no. 3, pp. 503–517, 1994.

[5] F. Fumero and C. Vercellis, "Synchronized development of production, inventory, and distribution schedules," *Transportation science*, vol. 33, no. 3, pp. 330–340, 1999.

[6] L. Lei, S. Liu, A. Ruszczynski, and S. Park, "On the integrated production, inventory, and distribution routing problem," *IIE Transactions*, vol. 38, no. 11, pp. 955–970, 2006.

[7] C. Archetti, L. Bertazzi, G. Paletta, and M. G. Speranza, "Analysis of the maximum level policy in a production-distribution system," *Computers & Operations Research*, vol. 38, no. 12, pp. 1731–1746, 2011.

[8] V. A. Armentano, A. L. Shiguemoto, and A. Løkketangen, "Tabu search with path relinking for an integrated production–distribution problem," *Computers & Operations Research*, vol. 38, no. 8, pp. 1199–1209, 2011.

[9] L. Bertazzi, G. Paletta, and M. G. Speranza, "Minimizing the total cost in an integrated vendor—managed inventory system," *Journal of heuristics*, vol. 11, no. 5-6, pp. 393–419, 2005.

[10] Y. Adulyasak, J.-F. Cordeau, and R. Jans, "Optimization-based adaptive large neighborhood search for the production routing problem," *Transportation Science*, vol. 48, no. 1, pp. 20–45, 2012.

[11] M. Ruokokoski, O. Solyali, J.-F. Cordeau, R. Jans, and H. Süral, "Efficient formulations and a branch-and-cut algorithm for a production-routing problem," *GERAD Technical Report G-2010-66*, 2010.

[12] O. Solyalı and H. Süral, "A branch-and-cut algorithm using a strong formulation and an a priori tour-based heuristic for an inventory-routing problem," *Transportation Science*, vol. 45, no. 3, pp. 335–345, 2011.

[13] A. L. Shiguemoto and V. A. Armentano, "A tabu search procedure for coordinating production, inventory and distribution routing problems," *International Transactions in Operational Research*, vol. 17, no. 2, pp. 179–195, 2010.

[14] Y. Adulyasak, J.-F. Cordeau, and R. Jans, "Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems," *INFORMS Journal on Computing*, vol. 26, no. 1, pp. 103–120, 2013.

[15] C. Archetti, L. Bertazzi, G. Laporte, and M. G. Speranza, "A branch-and-cut algorithm for a vendor-managed inventory-routing problem," *Transportation Science*, vol. 41, no. 3, pp. 382–391, 2007.

[16] G. Desaulniers, J. G. Rakke, and L. C. Coelho, "A branch-price-and-cut algorithm for the inventory-routing problem," *Transportation Science*, vol. 50, no. 3, pp. 1060–1076, 2015.

[17] P. Avella, M. Boccia, and L. A. Wolsey, "Single-item reformulations for a vendor managed inventory routing problem: Computational experience with benchmark instances," *Networks*, vol. 65, no. 2, pp. 129–138, 2015.

[18] L. C. Coelho and G. Laporte, "Improved solutions for inventory-routing problems through valid inequalities and input ordering," *International Journal of Production Economics*, vol. 155, pp. 391–397, 2014.

[19] P. Avella, M. Boccia, and L. A. Wolsey, "Single-period cutting planes for inventory routing problems," *Transportation Science*, vol. 52, no. 3, pp. 497–508, 2017.

[20] Y. Pochet and L. A. Wolsey, "Polyhedra for lot-sizing with wagner—whitin costs," *Mathematical Programming*, vol. 67, no. 1-3, pp. 297–323, 1994.

[21] I. Barany, T. J. Van Roy, and L. A. Wolsey, "Strong formulations for multi-item capacitated lot sizing," *Management Science*, vol. 30, no. 10, pp. 1255–1261, 1984.

[22] A. Atamtürk and S. Küçükyavuz, "Lot sizing with inventory bounds and fixed costs: Polyhedral study and computation," *Operations Research*, vol. 53, no. 4, pp. 711–730, 2005.

[23] M. W. Padberg, T. J. Van Roy, and L. A. Wolsey, "Valid linear inequalities for fixed charge problems," *Operations Research*, vol. 33, no. 4, pp. 842–861, 1985.

Appendix A

Data

Randomly Generated Instances

Time variant instances are generated as follows:

$n = 3, 6$, $v = 5k$ for $k = 1, \ldots, 10$ when $n = 3$; $v = 5k$ for $k = 1, \ldots, 6$ when $n = 6$,

d_{it} for $i \in V, t \in T$ is generated as an integer number in $[10, 100]$,

r_t for $t \in T$ is equal to $\sum_{i \in V} d_{it}$,

u_i for $i \in V$ is equal to $m_i g_i$ where m_i is the maximum amount of demand at one period throughout the planning horizon and g_i is a randomly selected number from $\{2, 3\}$,

$\bar{s}_{00}$ is equal to $\sum_{i \in V} u_i$; $\bar{s}_{i0}$ for $i \in V$ is equal to $u_i - a_i$ where a_i is average amount of demand throughout the planning horizon,

h_0 is equal to 0.3 (high) and 0.03 (low); h_i for $i \in V$ is generated in $[0.1, 0.5]$ (high) and $[0.01, 0.05]$ (low), and rounded to nearest number with two decimals,

Q is equal to nearest number to $\dfrac{3}{2}R$ where R is the average amount of delivered quantity to supplier throughout the planning horizon,

c_{ij} for $(i,j) \in E$ is equal to the $\left\lfloor \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \right\rfloor$ where (X_i, Y_i) and (X_j, Y_j) is generated as integer numbers in $[0, 500]$.

Time invariant instances are generated as follows (structure of benchmark instances):

$n = 3, 6$, $v = 5k$ for $k = 1, \ldots, 10$ when $n = 3$; $v = 5k$ for $k = 1, \ldots, 6$ when $n = 6$,

$d_{it} = d_i$ for $i \in V, t \in T$ is generated as an integer number in $[10, 100]$,

r_t for $t \in T$ is equal to $\sum_{i \in V} d_i$,

u_i for $i \in V$ is equal to $d_i g_i$ where g_i is a randomly selected number from $\{2, 3\}$,

$\bar{s}_{00}$ is equal to $\sum_{i \in V} u_i$; $\bar{s}_{i0}$ for $i \in V$ is equal to $u_i - d_i$,

h_0 is equal to 0.3 (high) and 0.03 (low); h_i for $i \in V$ is generated in $[0.1, 0.5]$ (high) and $[0.01, 0.05]$ (low), and rounded to nearest number with two decimals,

Q is equal to nearest number to $\dfrac{3}{2}R$ where R is the average amount of delivered quantity to supplier throughout the planning horizon,

c_{ij} for $(i,j) \in E$ is equal to the $\left\lfloor \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \right\rfloor$ where (X_i, Y_i) and (X_j, Y_j) is generated as integer numbers in $[0, 500]$.

Appendix B

Results

In Tables B.1-B.28, we report the number of vehicles, type of the instances (benchmark or generated), status of the solver, objective (or best upper bound) value of the instances, total number of nodes at the branch tree, lower bound obtained at the root node, optimality gap, best upper bound, best lower bound, total number of lazy (for integral solutions) cuts added, total number of user (for fractional solutions) cuts added and total time spent for the solution of the instances, respectively. Table B.29 presents whether inequalities (3.24) & (3.25) are used as well.

Table B.1: $high50$-3 Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	548	14224.62	-	-	-	3642	0	135.70 s
		2	optimal	15001.64	356	14501.3	-	-	-	853	0	43.50 s
		3	optimal	15279.49	1216	14917.16	-	-	-	4277	0	277.31 s
		4	optimal	16517	316	16359.94				256	0	19.76 s
		5	optimal	15678.67	6534	15516.08	-	-	-	15227	0	1618.86 s
	G	1	feasible	19383.13	5276	18546.74	1.69%	19383.13	19056.48	18260	0	1800 s
		2	feasible	17337.61	6500	16361.88	2.25%	17337.61	16947.19	20278	0	1800 s
		3	feasible	18313.12	25868	17810.14	0.52%	18313.12	18217.4	8385	0	1800 s
		4	feasible	17097.02	7626	16404.56	3.25%	17097.02	16541.94	16211	0	1800 s
		5	optimal	17708.67	1194	17504.89	-	-	-	566	0	26.27 s
2	B	1	feasible	17328.66	34433	14382.36	15.51%	17328.66	14640.28	12960	0	1800 s
		2	unknown	-	22002	14412.8	-	-	14972.55	15148	0	1800 s
		3	feasible	16112.13	24114	14957.57	4.81%	16112.13	15336.41	9875	0	1800 s
		4	optimal	16775.12	23016	16287.79	-	-	-	974	0	285.55 s
		5	feasible	17050.71	48513	15581.82	7.35%	17050.71	15796.88	7771	0	1800 s
	G	1	feasible	20711.18	13562	18439.34	9.18%	20711.18	18809.4	19513	0	1800 s
		2	feasible	17553.34	4843	16651.14	3.58%	17553.34	16925.22	21835	0	1800 s
		3	feasible	19066.47	7511	17695.23	6.45%	19066.47	17837.28	23578	0	1800 s
		4	feasibe	17761.47	7671	16262.27	7.09%	17761.47	16501.54	20973	0	1800 s
		5	feasible	18058.91	7344	17396.18	2.56%	18058.91	17596.78	24417	0	1800 s
3	B	1	unknown	-	25402	14434.55	-	-	14808.2	10576	0	1800 s
		2	unknown	-	61673	14412.8	-	-	15192.56	8121	0	1800 s
		3	feasible	17533.69	92822	14957.57	11.38%	17533.69	15538.64	6798	0	1800 s
		4	feasible	19206.7	32400	16287.79	12.66%	19206.7	16775.5	12238	0	1800 s
		5	unknown	-	48442	15526.63	-	-	15970.78	6368	0	1800 s
	G	1	feasible	19939.04	31906	18439.34	5.30%	19939.04	18881.34	14375	0	1800 s
		2	feasible	17799.26	14814	16651.14	4.83%	17799.26	16939.23	13664	0	1800 s
		3	unknown	-	9080	17678.72	-	-	18001.34	24469	0	1800 s
		4	feasible	17555.54	15646	16262.27	5.82%	17555.54	16534.2	11955	0	1800 s
		5	feasible	18075.63	41635	17396.18	0.73%	18075.63	17944.28	4718	0	1800 s

Table B.2: $high50$-3 with Inequalities (3.6), (3.7) & (3.8) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	548	14224.62	-	-	-	3642	0	136.64 s
		2	optimal	15001.64	356	14501.3	-	-	-	853	0	43.43 s
		3	optimal	15279.49	1216	14917.16	-	-	-	4277	0	269.08 s
		4	optimal	16517	316	16359.94				256	0	20.79 s
		5	optimal	15678.67	12102	15515.2	-	-	-	9266	10	1121.57 s
	G	1	feasible	19378.78	7063	18546.11	2.04%	19378.78	18984.1	16069	8	1800 s
		2	feasible	17298.39	6625	16361.88	2.00%	17298.39	16952.3	22197	0	1800 s
		3	feasible	18316.5	16960	17810.14	0.52%	18316.5	18220.68	11205	4	1800 s
		4	feasible	17008.72	7084	16404.78	2.74%	17008.72	16543.31	16839	26	1800 s
		5	optimal	17708.67	1194	17504.89	-	-	-	566	0	23.40 s
2	B	1	feasible	17328.66	34400	14382.36	15.51%	17328.66	14640.28	12960	1	1800 s
		2	unknown	-	22013	14412.8	-	-	14972.59	15148	0	1800 s
		3	feasible	17385.64	30100	14957.57	12.02%	17385.64	15296.73	11001	4	1800 s
		4	optimal	16775.12	23016	16287.79	-	-	-	974	0	290.79 s
		5	unknown	-	26781	15581.88	-	-	15806.91	9718	6	1800 s
	G	1	feasible	19823.79	7455	18438.24	4.93%	19823.79	18845.73	16795	6	1800 s
		2	feasible	17595.84	8800	16671.31	3.86%	17595.84	16917.37	18242	22	1800 s
		3	feasible	18824.39	6404	17695.23	4.45%	18824.39	17986.76	22944	17	1800 s
		4	feasible	16592.58	14146	16262.27	3.67%	17224.74	16592.27	12987	12	1800 s
		5	optimal	17831.11	87282	17396.18	-	-	-	2289	8	1435.06 s
3	B	1	feasible	19271.78	68716	14421.11	21.23%	19271.78	15179.17	5370	2	1800 s
		2	unknown	-	59421	14412.8	-	-	15190.33	7844	0	1800 s
		3	unknown	-	64971	14957.57	-	-	15570.33	5566	4	1800 s
		4	feasible	19206.7	27818	16287.79	12.66%	19206.7	16774.88	11150	0	1800 s
		5	unknown	-	38897	15526.63	-	-	15890.35	5494	6	1800 s
	G	1	unknown	-	40687	18438.24	-	-	18943.35	13951	6	1800 s
		2	feasible	17947.9	12362	16671.31	5.66%	17947.9	16931.69	16743	22	1800 s
		3	feasible	26158.59	29411	17678.72	31.45%	26158.59	17932.94	13667	11	1800 s
		4	feasible	20046.24	18642	16262.27	17.55%	20046.24	16528.46	14521	12	1800 s
		5	feasible	18076.63	38240	17396.18	0.97%	18076.63	17901.78	5088	8	1800 s

Table B.3: *high*50-3 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	95	14455.41	-	-	-	136	69	20.46 s
		2	optimal	15001.64	579	14610.8	-	-	-	786	8	47.38 s
		3	optimal	15279.49	2737	15148.79	-	-	-	108	51	45.76 s
		4	optimal	16517	86	16466.52				116	63	16.69 s
		5	optimal	15678.67	5153	15557.57	-	-	-	90	99	94.92 s
	G	1	feasible	19359.2	35860	18615.13	2.52%	19359.2	18871.81	7520	222	1800 s
		2	feasible	17322.76	6174	16675.01	2.36%	17322.76	16914.56	22028	63	1800 s
		3	feasible	18334.2	52279	17875.3	1.32%	18334.2	18091.32	5464	99	1800 s
		4	feasible	16918.14	34070	16461.07	1.87%	16918.14	16601.38	8460	173	1800 s
		5	optimal	17708.67	423	17653.33	-	-	-	282	11	17.55 s
2	B	1	feasible	16310	105761	14780.07	8.53%	16310	14918.06	3390	140	1800 s
		2	optimal	15342.94	62207	15005.11	-	-	-	1190	149	1475.51 s
		3	feasible	15704.92	21632	15289.73	1.82%	15704.92	15418.98	7343	136	1800 s
		4	optimal	16775.12	31622	16637.07	-	-	-	548	153	638.82 s
		5	feasible	16085.42	121214	15781.5	1.12%	16085.42	15906.12	886	172	1800 s
	G	1	feasible	19428.85	63600	18703.18	2.28%	19428.85	18986.25	54156	329	1800 s
		2	feasible	17458.94	72242	16647.47	3.24%	17458.94	16893.02	2732	393	1800 s
		3	feasible	18342.21	63389	17800.46	1.29%	18342.21	18106.38	1394	392	1800 s
		4	unknown	-	28710	16505.48	-	-	16638.17	9015	337	1800 s
		5	optimal	17831.11	1764	17665.26	-	-	-	596	144	71.17 s
3	B	1	feasible	17004.7	128920	15231.38	9.60%	17004.7	15372.96	1487	174	1800 s
		2	unknown	-	61482	15343.92	-	-	15614.62	4519	178	1800 s
		3	feasible	16621.16	94043	15524.53	5.67%	16621.16	15678.67	2495	159	1800 s
		4	feasible	18444.28	65232	16965.58	7.62%	18444.28	17038.84	2931	183	1800 s
		5	unknown	-	119001	16148.54	-	-	16241.73	2243	210	1800 s
	G	1	feasible	19434.67	20500	18799.69	2.24%	19434.67	18999.73	10542	365	1800 s
		2	feasible	17558.19	107000	16755.11	3.07%	17558.19	17020.01	2630	395	1800 s
		3	feasible	18461.26	49270	17938.37	1.36%	18461.26	18209.26	997	438	1800 s
		4	feasible	17989.32	31360	16680.39	6.72%	17989.32	16780.9	7454	298	1800 s
		5	feasible	18065.63	81319	17817.43	0.55%	18065.63	17966.47	1236	408	1800 s

Table B.4: *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	24	14471.95	-	-	-	67	576	18.94 s
		2	optimal	15001.64	486	14610.8	-	-	-	766	57	44.64 s
		3	optimal	15279.49	141	15161.44	-	-	-	5	384	22.16 s
		4	optimal	16517	142	16477.83				147	883	29.16 s
		5	optimal	15678.67	2209	15575.94	-	-	-	235	924	70.06 s
	G	1	optimal	19317.62	888	19188.07	-	-	-	311	1504	66.55 s
		2	feasible	17243.73	56800	17019.78	0.19%	17243.73	17211.79	6407	1308	1800 s
		3	optimal	18313.12	8839	18154.74	-	-	-	11203	664	1249.13 s
		4	feasible	16861.58	18231	16630.16	0.75%	16861.58	16735.64	9832	1057	1800 s
		5	optimal	17708.67	358	17657.69	-	-	-	117	407	14.71 s
2	B	1	feasible	15200.44	163694	14802.36	1.40%	15200.44	14987.84	857	1694	1800 s
		2	feasible	15391.62	17908	15025.61	0.66%	15391.62	15290.39	7474	1349	1800 s
		3	optimal	15520.95	84976	15305	-	-	-	1167	1282	1222.71 s
		4	optimal	16775.12	21290	16657.53	-	-	-	490	1406	456.13 s
		5	feasible	16097.75	98428	15800.28	1.04%	16097.75	15931.12	1571	1674	1800 s
		5	feasible	16085.42	121214	15781.5	1.12%	16085.42	15906.12	886	172	1800 s
	G	1	optimal	19318.53	2041	19179.41	-	-	-	497	1922	140.95 s
		2	feasible	17285.98	81300	16985.83	0.39%	17285.98	17218.69	1708	1257	1800 s
		3	optimal	18330.56	4121	18150.95	-	-	-	10945	2362	782.02 s
		4	feasible	17231.88	44706	16679.28	2.37%	17231.88	16822.82	5362	1295	1800 s
		5	optimal	17831.11	2034	17707.71	-	-	-	567	586	90.20 s
3	B	1	unknown	-	155024	15242.29	-	-	15423.44	1384	2085	1800 s
		2	unknown	-	91883	15361.97	-	-	15632.94	1573	1865	1800 s
		3	feasible	17060.75	107681	15537.94	7.97%	17060.75	15701.38	2820	1526	1800 s
		4	unknown	-	47618	16966.38	-	-	17055.99	3316	2166	1800 s
		5	feasible	27573.67	118154	16156.29	41.11%	27573.67	16237.83	2938	1848	1800 s
	G	1	optimal	19375.71	1719	19252.32	-	-	-	493	2076	109.82 s
		2	feasible	17417.22	96600	17084.61	0.46%	17417.22	17336.46	1181	1587	1800 s
		3	optimal	18418.22	1278	18236.62	-	-	-	3033	2295	276.35 s
		4	feasible	17543.05	13405	16800.16	3.66%	17543.05	16901.82	9232	1389	1800 s
		5	feasible	18134.03	15435	17828.3	1.04%	18134.03	17945.94	5538	905	1800 s

Table B.5: *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	0	14577.3	-	-	-	1	183	8.16 s
		2	optimal	15001.64	132	14947.44	-	-	-	59	233	30.47 s
		3	optimal	15279.49	37	15235.22	-	-	-	94	439	30.02 s
		4	optimal	16517	131	16474.2				46	745	38.24 s
		5	optimal	15678.67	9	15651.32	-	-	-	0	532	23.00 s
	G	1	optimal	19317.62	1919	19170.06	-	-	-	945	1344	165.97 s
		2	feasible	17259.36	13147	17014.02	0.42%	17259.36	17187.57	17611	960	1800 s
		3	optimal	18313.12	8071	18169.48	-	-	-	1949	670	307.71 s
		4	optimal	16822.21	5492	16661.8	-	-	-	1515	1175	228.50 s
		5	optimal	17708.67	154	17667.14	-	-	-	307	198	19.89 s
2	B	1	feasible	15248.86	90414	14940.15	1.39%	15248.86	15037.05	2218	639	1800 s
		2	feasible	15378.62	36423	14984.42	0.67%	15378.62	15276.02	4100	72	1800 s
		3	optimal	15520.95	38473	15403.78	-	-	-	872	219	445.96 s
		4	optimal	16775.12	16887	16660.19	-	-	-	228	1158	366.25 s
		5	feasible	16052.16	112789	15896.46	0.43%	16052.16	15983.13	1846	614	1800 s
	G	1	optimal	19318.53	1941	19164.02	-	-	-	169	1569	193.90 s
		2	feasible	17285.56	101693	16981.43	0.49%	17285.56	17201.32	1176	1259	1800 s
		3	optimal	18330.56	11710	18124.07	-	-	-	1411	1643	507.90 s
		4	optimal	16918.89	26130	16705.42	-	-	-	765	1349	679.78 s
		5	optimal	17831.11	882	17740.29	-	-	-	993	450	86.49 s
3	B	1	optimal	19375.71	1719	19252.32	-	-	-	493	2076	109.82 s
		2	feasible	17417.22	96600	17084.61	0.46%	17417.22	17336.46	1181	1587	1800 s
		3	optimal	18418.22	1278	18236.62	-	-	-	3033	2295	276.35 s
		4	feasible	17543.05	13405	16800.16	3.66%	17543.05	16901.82	9232	1389	1800 s
		5	feasible	18134.03	15435	17828.3	1.04%	18134.03	17945.94	5538	905	1800 s
	G	1	optimal	19375.71	4356	19237.37	-	-	-	102	1652	271.19 s
		2	feasible	17426.94	91600	17048.24	1.10%	17426.94	17235.94	1139	1071	1800 s
		3	optimal	18418.22	3346	18314.41	-	-	-	1093	1479	194.32 s
		4	feasible	17145.65	36521	16817.74	0.98%	17145.65	16977.25	3462	1286	1800 s
		5	optimal	18064.63	114098	17934.08	-	-	-	610	766	1533.82 s

Table B.6: *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	0	14571.12	-	-	-	0	220	6.54 s
		2	optimal	15001.64	303	14946.79	-	-	-	17	317	35.13 s
		3	optimal	15279.49	56	15233.85	-	-	-	0	419	26.97 s
		4	optimal	16517	59	16478.28	-	-	-	84	740	33.39 s
		5	optimal	15678.67	387	15649.38	-	-	-	217	508	51.07 s
	G	1	optimal	19317.62	1919	19170.06	-	-	-	945	1344	170.34 s
		2	feasible	17259.36	13050	17014.02	0.42%	17259.36	17187.35	17547	960	1800 s
		3	optimal	18313.12	8071	18169.48	-	-	-	1949	670	307.94 s
		4	optimal	16822.21	5492	16661.8	-	-	-	1515	1175	235.41 s
		5	optimal	17708.67	154	17667.14	-	-	-	307	198	21.18 s
2	B	1	feasible	15387.5	58700	14939.94	2.30%	15387.5	15034.06	2968	582	1800 s
		2	feasible	15908.11	29226	15199.21	4.05%	15908.11	15264.19	6764	541	1800 s
		3	optimal	15520.95	43065	15403.02	-	-	-	844	372	537.47 s
		4	optimal	16775.12	9593	16665.8	-	-	-	293	1302	218.34 s
		5	feasible	16271.27	91900	15893.59	1.77%	16271.27	15982.56	1982	541	1800 s
	G	1	optimal	19318.53	1941	19164.02	-	-	-	169	1569	189.39 s
		2	feasible	17285.56	103600	16981.43	0.49%	17285.56	17201.71	1176	1259	1800 s
		3	optimal	18330.56	11710	18124.07	-	-	-	1411	1643	494.93 s
		4	optimal	16918.89	26130	16705.42	-	-	-	765	1349	703.68 s
		5	optimal	17831.11	882	17740.29	-	-	-	993	450	88.09 s
3	B	1	feasible	20005.86	74791	15462.18	21.94%	20005.86	15616.46	2159	864	1800 s
		2	feasible	17091.85	89772	15535.51	7.76%	17091.85	15765.4	1794	1012	1800 s
		3	feasible	16109.19	119399	15678.32	2.05%	16109.19	15778.8	1569	968	1800 s
		4	unknown	-	76679	17023.85	-	-	17196.35	1857	1300	1800 s
		5	unknown	-	92320	16307.27	-	-	16418.83	1346	1265	1800 s
	G	1	optimal	19375.71	4356	19237.37	-	-	-	102	1652	270.5 s
		2	feasible	17426.92	95040	17048.24	1.09%	17426.92	17236.92	1139	1071	1800 s
		3	optimal	18418.22	3346	18314.41	-	-	-	1093	1479	197.94 s
		4	feasible	17145.65	39734	16817.74	0.95%	17145.65	16982.5	3462	1286	1800 s
		5	optimal	18064.63	114098	17934.08	-	-	-	610	766	1531.59 s

Table B.7: *high*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	14577.3	0	14571.12	-	-	-	0	224	7.11 s
		2	optimal	15001.64	303	14946.79	-	-	-	17	337	32.07 s
		3	optimal	15279.49	56	15233.85	-	-	-	0	464	28.66 s
		4	optimal	16517	59	16478.28	-	-	-	84	814	33.95 s
		5	optimal	15678.67	387	15649.38	-	-	-	217	622	51.4 s
	G	1	optimal	19317.62	1919	19170.06	-	-	-	945	1501	167.04 s
		2	feasible	17253.16	12341	17014.02	0.39%	17253.16	17185.14	15814	1161	1800 s
		3	optimal	18313.12	8071	18169.48	-	-	-	1949	674	319.34 s
		4	optimal	16822.21	5492	16661.8	-	-	-	1515	1228	229.96 s
		5	optimal	17708.67	154	17667.14	-	-	-	307	287	20.21 s
2	B	1	feasible	15387.5	58600	14939.94	2.30%	15387.5	15034.08	2926	593	1800 s
		2	feasible	15860.84	29307	15199.21	3.76%	15860.84	15264.73	6756	573	1800 s
		3	optimal	15520.95	43065	15403.02	-	-	-	844	422	536.59 s
		4	optimal	16775.12	9593	16665.8	-	-	-	293	1395	217.38 s
		5	feasible	16222.27	94076	15893.59	1.48%	16222.27	15982.77	1998	661	1800 s
	G	1	optimal	19318.53	1941	19164.02	-	-	-	169	1743	196.42 s
		2	feasible	17285.56	101900	16981.43	0.49%	17285.56	17201.37	1176	1477	1800 s
		3	optimal	18330.56	11925	18124.07	-	-	-	1411	1664	514.52 s
		4	optimal	16918.89	26130	16705.42	-	-	-	765	1419	676.26 s
		5	optimal	17831.11	882	17740.29	-	-	-	993	546	87.56 s
3	B	1	feasible	20005.86	75738	15462.17	21.94%	20005.86	15616.86	2133	881	1800 s
		2	feasible	17091.85	90300	15535.51	7.76%	17091.85	15765.52	1803	1053	1800 s
		3	feasible	16109.19	120943	15678.32	2.05%	16109.19	15779.06	1569	1033	1800 s
		4	feasible	18119.9	76681	17024.08	5.11%	18119.9	17193.5	1807	1409	1800 s
		5	unknown	-	92333	16307.27	-	-	16418.83	1346	1400	1800 s
	G	1	optimal	19375.71	4356	19237.37	-	-	-	102	1840	271.86 s
		2	feasible	17426.92	92983	17048.24	1.09%	17426.92	17236.47	1147	1079	1800 s
		3	optimal	18418.22	3346	18314.41	-	-	-	1093	1517	196.42 s
		4	feasible	17128.71	83100	16823.04	0.53%	17128.71	17038.06	628	1474	1800 s
		5	optimal	18064.63	114098	17934.08	-	-	-	610	867	1516.86 s

Table B.8: *high*30-6 Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	22837.94	9430	22048.86	-	-	-	5637	0	419.71 s
		2	optimal	19876.76	7814	19440.36	-	-	-	2255	0	188.95 s
		3	optimal	23096.93	5243	22527.68	-	-	-	2036	0	146.46 s
		4	optimal	17509.82	3984	17230.32				1418	0	90.06 s
		5	feasible	18736.31	24215	18131.85	0.20%	18736.31	18699.36	9272	0	1800 s
	G	1	optimal	25497.66	2673	25286.57	-	-	-	1674	0	72.20 s
		2	optimal	26211.7	1808	25893.17	-	-	-	1847	0	53.92 s
		3	optimal	24646.9	13076	24096.58	-	-	-	5305	0	598.61 s
		4	optimal	26648.67	12349	25994.66	-	-	-	9092	0	690.42 s
		5	optimal	25633.72	3546	25115.23	-	-	-	2043	0	92.64 s
2	B	1	unknown	-	46220	22046.25	-	-	22741.87	7776	0	1800 s
		2	unknown	-	53402	19168.24	-	-	19924.19	8051	0	1800 s
		3	feasible	23582.17	61400	22508.92	1.58%	23582.17	23208.54	4902	0	1800 s
		4	feasible	19258.08	72494	17156.75	8.39%	19258.08	17642.82	4551	0	1800 s
		5	unknown	-	74219	18111.95	-	-	18639.58	6348	0	1800 s
	G	1	feasible	27937.84	20100	25180.89	9.01%	27937.84	25419.16	21046	0	1800 s
		2	optimal	26341.65	13183	25837.21	-	-	-	4740	0	526.00 s
		3	feasible	25666.45	39536	24021.37	4.11%	25666.45	24609.81	10686	0	1800 s
		4	feasible	28316.15	22202	26016.94	6.76%	28316.15	26403.33	16298	0	1800 s
		5	feasible	26031	22755	25128.08	1.87%	26031	25544.58	13009	0	1800 s
3	B	1	unknown	-	55087	22125.65	-	-	22806.83	5712	0	1800 s
		2	unknown	-	65160	19421.16	-	-	19941.65	4797	0	1800 s
		3	unknown	-	75391	22718.05	-	-	23285.06	3904	0	1800 s
		4	unknown	-	49891	17160.08	-	-	17808.61	2466	0	1800 s
		5	unknown	-	75330	18111.95	-	-	18731.9	4732	0	1800 s
	G	1	unknown	-	34000	25184.01	-	-	25501.66	10371	0	1800 s
		2	feasible	26669.65	45715	25805.79	0.43%	26669.65	26555.08	4574	0	1800 s
		3	unknown	-	39061	24021.37	-	-	24660.91	9242	0	1800 s
		4	unknown	-	34006	25949.7	-	-	26647.12	11174	0	1800 s
		5	feasible	28297.37	40956	25099.74	9.13%	28297.37	25713.69	8562	0	1800 s

Table B.9: *high*30-6 with Inequalities (3.6), (3.7) & (3.8) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	22837.94	16381	22048.86	-	-	-	7437	131	946.58 s
		2	optimal	19876.76	7103	19874.9	-	-	-	1397	109	143.08 s
		3	optimal	23096.93	4040	22522.31	-	-	-	1347	106	96.37 s
		4	optimal	17509.82	3695	17233.37				1912	132	112.15 s
		5	feasible	18731.81	31274	18131.23	0.23%	18731.81	18688.35	7189	108	1800 s
	G	1	optimal	25497.66	2682	25170.94	-	-	-	2735	93	109.34 s
		2	optimal	26211.7	747	25852.58	-	-	-	2047	85	55.09 s
		3	optimal	24646.9	6435	24180.3	-	-	-	3349	158	265.78 s
		4	optimal	26648.67	8899	26151.53	-	-	-	8138	144	591.82 s
		5	optimal	25633.72	2249	25099.72	-	-	-	2056	109	87.80 s
2	B	1	unknown	-	67612	22138.44	-	-	22704.51	6773	133	1800 s
		2	feasible	21349.18	45554	19199.17	6.31%	21349.18	20001.4	6578	125	1800 s
		3	feasible	23618.23	57966	22518.55	1.71%	23618.23	23214.7	4600	123	1800 s
		4	feasible	20091.14	84432	17155.15	12.05%	20091.14	17670.8	4504	131	1800 s
		5	unknown	-	47011	18197.65	-	-	18527.47	5129	125	1800 s
	G	1	feasible	26126.92	24815	25221.75	2.30%	26126.92	25526.52	11290	125	1800 s
		2	feasible	26341.65	15328	25860.98	0.09%	26341.65	26316.29	9458	99	1800 s
		3	feasible	25915.12	20064	24123.13	5.06%	25915.12	24604.51	13783	156	1800 s
		4	unknown	-	21509	25920.85	-	-	26376.22	19134	127	1800 s
		5	feasible	26350.57	21308	25110.32	3.28%	26350.57	25486.97	14722	137	1800 s
3	B	1	unknown	-	49081	22102.62	-	-	22788.18	5135	131	1800 s
		2	unknown	-	65402	19180.45	-	-	20111	4354	119	1800 s
		3	unknown	-	88954	22509.55	-	-	23300.45	3188	115	1800 s
		4	unknown	-	97981	17189.21	-	-	17852.54	2758	140	1800 s
		5	unknown	-	64379	18197.65	-	-	18839.81	3959	125	1800 s
	G	1	unknown	-	30858	25236.66	-	-	25555.83	9284	122	1800 s
		2	feasible	26900.75	42519	25807.39	1.80%	26900.75	26416.17	5813	87	1800 s
		3	unknown	-	33402	24175.24	-	-	24886.09	9279	164	1800 s
		4	feasible	28986.82	31087	25995.24	7.44%	28986.82	26829.07	11364	144	1800 s
		5	feasible	27356.29	34976	25040.13	6.44%	27356.29	25595.7	9487	118	1800 s

Table B.10: *high*30-6 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	feasible	22844.01	84833	22418.96	0.17%	22844.01	22805.4	947	246	1800 s
		2	optimal	19876.76	10773	19471.69	-	-	-	793	159	191.39 s
		3	optimal	23096.93	116707	22815.72	-	-	-	490	207	1735.30 s
		4	optimal	17509.82	5865	17279.87				442	187	127.01 s
		5	feasible	18767.32	81000	18310.61	0.69%	18767.32	18638.51	478	318	1800 s
	G	1	optimal	25497.66	4624	25303.99	-	-	-	304	237	115.26 s
		2	optimal	26211.7	12321	25909.13	-	-	-	905	163	258.67 s
		3	optimal	24646.9	55503	24240.97	-	-	-	838	437	1099.98 s
		4	optimal	26648.67	41958	26273.72	-	-	-	1425	364	973.04 s
		5	feasible	25643.44	101600	25304.02	0.12%	25643.44	25613.52	570	367	1800 s
2	B	1	feasible	23996.1	122901	23110.84	2.66%	23996.1	23357.5	490	447	1800 s
		2	feasible	21418.99	134654	20004.62	5.42%	21418.99	20257	1726	292	1800 s
		3	feasible	23419.82	62634	23161.38	0.25%	23419.82	23362.28	583	338	1800 s
		4	unknown	-	149298	17604.49	-	-	17817.62	1294	378	1800 s
		5	unknown	-	137406	18819.21	-	-	18986.52	945	548	1800 s
	G	1	feasible	25903.54	80275	25524.67	0.47%	25903.54	25780.86	542	610	1800 s
		2	optimal	26341.65	6835	26075.06	-	-	-	571	341	162.91 s
		3	feasible	25203.93	101294	25203.93	1.08%	25203.93	24931.82	509	748	1800 s
		4	feasible	27109.74	46967	26609.45	0.64%	27109.74	26936.88	752	527	1800 s
		5	feasible	25909.83	92295	25587.73	0.27%	25909.83	25839.84	448	596	1800 s
3	B	1	unknown	-	139978	24068.35	-	-	24321.8	698	741	1800 s
		2	unknown	-	146211	20841.36	-	-	21009.54	731	464	1800 s
		3	unknown	-	131066	23732.72	-	-	23885.11	1137	479	1800 s
		4	unknown	-	123644	18368.71	-	-	18538.49	300	592	1800 s
		5	unknown	-	117600	19584.29	-	-	19713.4	644	691	1800 s
	G	1	feasible	27002.3	80448	26003.81	3.07%	27002.3	26172.39	1977	666	1800 s
		2	optimal	26642.2	64790	26368.06	-	-	-	382	543	1083.18 s
		3	feasible	25950.53	115101	25012.01	2.33%	25950.53	25347.26	782	872	1800 s
		4	feasible	28140.14	131400	27302.01	2.24%	28140.14	27510.44	346	1084	1800 s
		5	feasible	26422.67	76400	26022.57	0.77%	26422.67	26220.68	413	723	1800 s

Table B.11: *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	22837.94	2148	22609.8	-	-	-	687	979	85.37 s
		2	optimal	19876.76	3302	19541.01	-	-	-	1916	265	145.47 s
		3	optimal	23096.93	1498	22931.85	-	-	-	511	662	45.42 s
		4	optimal	17509.82	679	17335.34				670	692	29.36 s
		5	optimal	18731.81	12793	18475.25	-	-	-	464	789	245.82 s
	G	1	optimal	25497.66	2681	25335.28	-	-	-	989	237	60.55 s
		2	optimal	26211.7	2221	26023.47	-	-	-	882	425	72.79 s
		3	optimal	24646.9	1635	24525.63	-	-	-	178	963	56.61 s
		4	optimal	26648.67	8519	26433.57	-	-	-	1680	702	335.94 s
		5	optimal	25633.72	4295	25447.42	-	-	-	599	630	83.31 s
2	B	1	feasible	23983.9	133600	23230.03	2.21%	23983.93	23454.09	1139	1047	1800 s
		2	feasible	20903.07	143721	20039.41	3.07%	20903.07	20261.44	1813	931	1800 s
		3	optimal	23416.82	120811	23219	-	-	-	300	1092	1676.82 s
		4	feasible	19537.61	211000	17649.81	8.87%	19537.61	17805.03	832	1087	1800 s
		5	feasible	20717.24	159150	18892.45	7.93%	20717.24	19073.67	808	1526	1800 s
	G	1	feasible	25913.34	87411	25571.84	0.42%	25913.34	25804.67	320	1087	1800 s
		2	optimal	26341.65	2421	26149.26	-	-	-	492	679	107.87 s
		3	feasible	25141.88	91186	24830.48	0.32%	25141.88	25062.25	460	1566	1800 s
		4	optimal	27096.74	13779	26801.7	-	-	-	902	1749	439.63 s
		5	optimal	25909.83	28703	25704.49	-	-	-	348	1414	448.11 s
3	B	1	unknown	-	130829	24179.89	-	-	24321.98	917	1329	1800 s
		2	unknown	-	133225	20852.03	-	-	21071.75	868	1218	1800 s
		3	feasible	24925.48	190000	23793.36	3.99%	24925.48	23930.72	433	1094	1800 s
		4	unknown	-	111551	18387.36	-	-	18517.24	546	1195	1800 s
		5	unknown	-	120750	19628.32	-	-	19783.56	602	1528	1800 s
	G	1	feasible	26841.48	97380	26006.5	2.27%	26841.48	26233.02	1099	1331	1800 s
		2	optimal	26642.2	19693	26386.96	-	-	-	460	1073	600.51 s
		3	feasible	25778.66	120200	25306.35	1.17%	25778.66	25477.27	651	1670	1800 s
		4	feasible	28010.31	122100	27398.25	1.36%	28010.31	27628.56	456	1870	1800 s
		5	feasible	26408.29	77957	26072.59	0.41%	26408.29	26300.12	649	1444	1800 s

Table B.12: *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	22837.94	1742	22672.21	-	-	-	678	711	108.26 s
		2	optimal	19876.76	7571	19681.33	-	-	-	651	402	157.41 s
		3	optimal	23096.93	2397	22945.58	-	-	-	264	527	103.48 s
		4	optimal	17509.82	747	17365.13	-	-	-	705	393	40.38 s
		5	optimal	18731.81	6869	18589.85	-	-	-	380	945	201.89 s
	G	1	optimal	25497.66	177	25436.74	-	-	-	284	353	51.46 s
		2	optimal	26211.7	299	26143.24	-	-	-	344	478	74.17 s
		3	optimal	24646.9	39	24604.13	-	-	-	50	861	54.24 s
		4	optimal	26648.67	1490	26595.98	-	-	-	227	775	97.04 s
		5	optimal	25633.72	568	25551.74	-	-	-	635	569	88.92 s
2	B	1	feasible	24024.68	114200	23439.86	1.79%	24024.68	23593.46	1373	790	1800 s
		2	feasible	22130.86	106000	20254.18	7.73%	22130.86	20419.23	1479	890	1800 s
		3	optimal	23416.82	4382	23332.67	-	-	-	136	877	127.08 s
		4	unknown	-	102065	17729.86	-	-	17854.18	1087	1272	1800 s
		5	feasible	21157.92	75400	19044.67	9.42%	21157.92	19165.76	1452	1602	1800 s
	G	1	feasible	25897.38	64761	25609.29	0.21%	25897.38	25842.19	354	926	1800 s
		2	optimal	26341.65	941	26237.49	-	-	-	246	1029	90.95 s
		3	feasible	25173.5	93663	24906.32	0.45%	25173.5	25060.11	337	1637	1800 s
		4	optimal	27096.74	20585	26916.38	-	-	-	255	1526	506.64 s
		5	optimal	25909.83	26055	25717.83	-	-	-	290	839	532.45 s
3	B	1	unknown	-	64246	24395.82	-	-	24573.21	972	1693	1800 s
		2	feasible	23251.98	88400	21089.18	8.55%	23251.98	21263.94	819	1634	1800 s
		3	feasible	25660.69	100900	23950.35	6.37%	25660.69	24025.76	1004	1132	1800 s
		4	unknown	-	63248	18550.42	-	-	18725.39	513	2435	1800 s
		5	unknown	-	59300	19784.19	-	-	19992.87	432	2342	1800 s
	G	1	feasible	26764.37	85508	26105.29	1.86%	26764.37	26267.01	672	1397	1800 s
		2	optimal	26642.2	5651	26498.46	-	-	-	273	1064	208.45 s
		3	feasible	25747.72	102304	25396.11	0.79%	25747.72	25545.18	109	2040	1800 s
		4	feasible	28154.64	68000	27535.91	1.64%	28154.64	27692.26	682	1761	1800 s
		5	feasible	26376.16	74207	26117.78	0.32%	26376.16	26291.95	239	1616	1800 s

Table B.13: *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	22837.94	1714	22672.21	-	-	-	678	709	104.56 s
		2	optimal	19876.76	4627	19678.42	-	-	-	1154	384	154.58 s
		3	optimal	23096.93	2397	22945.58	-	-	-	264	528	102.76 s
		4	optimal	17509.82	747	17365.13	-	-	-	705	393	41.16 s
		5	optimal	18731.81	2937	18586.99	-	-	-	1048	994	178.99 s
	G	1	optimal	25497.66	177	25436.74	-	-	-	284	353	52.08 s
		2	optimal	26211.7	299	26143.24	-	-	-	344	478	76.57 s
		3	optimal	24646.9	39	24604.13	-	-	-	50	856	54.12 s
		4	optimal	26648.67	1424	26584.92	-	-	-	911	762	106.18 s
		5	optimal	25633.72	568	25551.74	-	-	-	635	567	87.65 s
2	B	1	feasible	24097.36	104061	23439.86	2.11%	24097.36	23588.76	1733	789	1800 s
		2	feasible	22116	106700	20254.18	7.67%	22116	20419.54	1457	890	1800 s
		3	optimal	23416.82	4382	23332.67	-	-	-	136	876	130.82 s
		4	unknown	-	110614	17729.86	-	-	17855.53	999	1269	1800 s
		5	unknown	-	73862	19044.67	-	-	19044.67	1614	1601	1800 s
	G	1	feasible	25903.34	61900	25609.29	0.30%	25903.34	25824.89	332	926	1800 s
		2	optimal	26341.65	941	26237.49	-	-	-	246	1029	91.22 s
		3	feasible	25151.28	89300	24904.49	0.37%	25151.28	25059.17	446	1568	1800 s
		4	optimal	27096.74	20585	26916.38	-	-	-	255	1526	510.96 s
		5	optimal	25909.83	26055	25717.83	-	-	-	290	839	553.50 s
3	B	1	unknown	-	68033	24395.82	-	-	24574.79	1011	1693	1800 s
		2	feasible	23251.98	94646	21089.18	8.54%	23251.98	21265.6	808	1634	1800 s
		3	unknown	-	104828	23951.97	-	-	24031.84	1093	1158	1800 s
		4	unknown	-	66877	18550.42	-	-	18727.36	539	2435	1800 s
		5	unknown	-	62761	19851.92	-	-	20013.13	468	2159	1800 s
	G	1	feasible	26764.37	86878	26105.29	1.86%	26764.37	26267.5	672	1397	1800 s
		2	optimal	26642.2	5651	26498.46	-	-	-	273	1064	215.17 s
		3	feasible	25747.72	101502	25396.11	0.79%	25747.72	25545.09	109	2040	1800 s
		4	feasible	28153.64	65651	27535.91	1.64%	28153.64	27690.66	681	1761	1800 s
		5	feasible	26376.16	71111	26117.78	0.32%	26376.64	26290.88	239	1616	1800 s

Table B.14: *high*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	22837.94	1714	22672.21	-	-	-	678	715	103.51 s
		2	optimal	19876.76	4627	19678.42	-	-	-	1154	401	156.63 s
		3	optimal	23096.93	2397	22945.58	-	-	-	264	565	107.17 s
		4	optimal	17509.82	747	17365.13	-	-	-	705	445	45.71 s
		5	optimal	18731.81	6058	18590.07	-	-	-	666	1034	243.96 s
	G	1	optimal	25497.66	177	25436.74	-	-	-	284	455	85.03 s
		2	optimal	26211.7	299	26143.24	-	-	-	344	596	89.95 s
		3	optimal	24646.9	39	24604.13	-	-	-	50	1009	121.24 s
		4	optimal	26648.67	1424	26584.92	-	-	-	911	959	114.48 s
		5	optimal	25633.72	568	25551.74	-	-	-	635	808	109.21 s
2	B	1	feasible	23941.55	98182	23442.6	1.45%	23941.55	23594.11	1372	814	1800 s
		2	feasible	22172.84	101801	20254.18	7.91%	22172.84	20419.89	1731	918	1800 s
		3	optimal	23416.82	4382	23332.67	-	-	-	136	918	131.22 s
		4	unknown	-	105063	17732.33	-	-	17862.04	1167	1268	1800 s
		5	unknown	-	83303	19053.98	-	-	19165.02	1149	1737	1800 s
	G	1	feasible	25897.34	67591	25614.67	0.18%	25897.34	25850.38	243	1246	1800 s
		2	optimal	26341.65	941	26237.49	-	-	-	246	1157	108.39 s
		3	feasible	25190.5	93497	24914.73	0.56%	25190.5	25048.82	394	1784	1800 s
		4	optimal	27096.74	32069	26908.11	-	-	-	145	1725	801.45 s
		5	optimal	25909.83	26055	25717.83	-	-	-	290	1087	566.26 s
3	B	1	unknown	-	87530	24396.85	-	-	24580.09	681	1421	1800 s
		2	feasible	21959.87	108340	21078.93	3.40%	21959.87	21212.44	672	1438	1800 s
		3	unknown	-	101889	23951.97	-	-	24031.32	1093	1205	1800 s
		4	unknown	-	68710	18532.21	-	-	18710.97	514	1721	1800 s
		5	unknown	-	62650	19843.4	-	-	20013.39	521	2187	1800 s
	G	1	feasible	26868.88	49720	26103.63	2.29%	26868.88	26252.69	905	1308	1800 s
		2	optimal	26642.2	5663	26498.46	-	-	-	273	1204	233.2 s
		3	feasible	25742.84	87094	25397.66	0.77%	25742.84	25544.38	278	2182	1800 s
		4	feasible	28069.02	62305	27544.5	1.27%	28069.02	27712.66	256	2199	1800 s
		5	feasible	26376.16	70587	26117.78	0.33%	26376.16	26290.33	241	1879	1800 s

Table B.15: *low*50-3 Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	381	3718.05	-	-	-	238	0	21.02 s
		2	optimal	4512.96	1340	4099.48	-	-	-	450	0	42.45 s
		3	optimal	4451.44	671	4307.61	-	-	-	4479	0	234.03 s
		4	optimal	4405.84	109	4319.14				223	0	9.01 s
		5	optimal	4218.37	7908	4042.48	-	-	-	1409	0	154.54 s
	G	1	feasible	4878.45	23662	4502.66	1.22%	4878.45	4819.12	9226	0	1800 s
		2	feasible	4473.9	15212	3765.39	4.45%	4473.9	4274.99	15492	0	1800 s
		3	feasible	5558.48	3688	3894.61	28.50%	5558.48	3974.29	29824	0	1800 s
		4	optimal	4541.02	12728	4325.99	-	-	-	4104	0	388.70 s
		5	feasible	4887.45	1889	4183.55	8.77%	4887.45	4458.5	26421	0	1800 s
2	B	1	feasible	6227.57	27110	3677.96	34.19%	6227.57	4098.54	11358	0	1800 s
		2	unknown	-	20613	4002.98	-	-	4451.79	16795	0	1800 s
		3	feasible	5015.14	55516	4207.12	8.98%	5015.14	4564.62	5477	0	1800 s
		4	optimal	4662.84	48823	4251.46	-	-	-	1110	0	595.38 s
		5	feasible	5188.62	41215	4047.81	16.51%	5188.62	4331.99	6265	0	1800 s
	G	1	feasible	4923.79	6326	4366.05	4.35%	4923.79	4709.53	17356	0	1800 s
		2	feasible	4533.46	11943	3796.6	8.24%	4533.46	4159.82	16246	0	1800 s
		3	feasible	6322.08	4693	3727.41	38.40%	6322.08	3894.49	26326	0	1800 s
		4	feasible	4563.24	23200	4288.75	2.17%	4563.24	4464.1	8263	0	1800 s
		5	unknown	-	2883	4161.4	-	-	4353.69	29557	0	1800 s
3	B	1	feasible	8993.51	61055	3677.96	51.59%	8993.51	4355.61	5359	0	1800 s
		2	unknown	-	44402	4002.98	-	-	4708.9	4805	0	1800 s
		3	unknown	-	75941	4207.12	-	-	4623.23	4869	0	1800 s
		4	unknown	-	25650	4251.46	-	-	4763.39	10625	0	1800 s
		5	unknown	-	50696	4083.82	-	-	4369.88	5393	0	1800 s
	G	1	feasible	5868.24	8072	4366.05	20.66%	5868.24	4655.61	21000	0	1800 s
		2	feasible	4520.65	18929	3796.6	8.46%	4520.65	4138.29	13383	0	1800 s
		3	unknown	-	19796	3727.41	-	-	3865.7	19796	0	1800 s
		4	feasible	4904.13	13600	4288.75	9.75%	4904.13	4426	10763	0	1800 s
		5	feasible	5065.69	20666	4161.4	11.50%	5065.69	4483.4	11204	0	1800 s

Table B.16: *low*50-3 with Inequalities (3.6), (3.7) & (3.8) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	381	3718.05	-	-	-	238	0	20.86 s
		2	optimal	4512.96	1340	4099.48	-	-	-	450	0	42.98 s
		3	optimal	4451.44	492	4307.61	-	-	-	2655	6	114.06 s
		4	optimal	4405.84	109	4319.14				223	0	9.41 s
		5	optimal	4218.37	7908	4042.48	-	-	-	1409	0	178.81 s
	G	1	feasible	4898.34	10737	4427.07	2.61%	4898.34	4770.65	13346	13	1800 s
		2	feasible	4505.92	14705	3765.39	4.72%	4505.92	4293.24	14674	3	1800 s
		3	feasible	5558.48	3651	3894.61	28.50%	5558.48	3974.29	28412	0	1800 s
		4	optimal	4541.02	12728	4325.99	-	-	-	4104	0	396.09 s
		5	feasible	4813.82	1744	4183.55	7.21%	4813.82	4466.68	24106	7	1800 s
2	B	1	feasible	6227.57	27003	3677.96	34.19%	6227.57	4098.36	11225	0	1800 s
		2	unknown	-	20612	4002.98	-	-	4451.79	16730	0	1800 s
		3	feasible	6185.54	37400	4207.12	26.80%	6185.54	4527.72	9666	1	1800 s
		4	optimal	4662.84	48823	4251.46	-	-	-	1110	0	604.02 s
		5	feasible	7687.28	43572	4047.81	43.79%	7687.28	4321.32	6129	2	1800 s
	G	1	feasible	5556.09	5322	4366.05	16.63%	5556.09	4632.2	20715	14	1800 s
		2	feasible	4527.68	12112	3767.64	7.75%	4527.68	4176.72	17651	4	1800 s
		3	feasible	6322.08	4675	3727.41	38.40%	6322.08	3894.49	25493	0	1800 s
		4	feasible	4779.66	13736	4288.75	7.33%	4779.66	4429.32	10808	1	1800 s
		5	feasible	4879.26	1961	4161.4	9.76%	4879.26	4403.16	23394	6	1800 s
3	B	1	feasible	8964.53	60202	3677.96	51.54%	8964.53	4344.57	5417	0	1800 s
		2	unknown	-	43862	4002.98	-	-	4708.01	4805	0	1800 s
		3	unknown	-	60849	4207.12	-	-	4738.53	5082	1	1800 s
		4	unknown	-	25609	4251.46	-	-	4763.39	10625	0	1800 s
		5	feasible	7858	80932	4083.9	42.66%	7858	4505.62	3501	8	1800 s
	G	1	feasible	5907.88	5151	4366.05	21.72%	5907.88	4624.67	22731	14	1800 s
		2	feasible	4557.77	8865	3767.64	6.15%	4557.77	4277.61	14526	4	1800 s
		3	unknown	-	12368	3727.41	-	-	3865.6	19796	0	1800 s
		4	feasible	4779.66	13736	4288.75	7.33%	4779.66	4429.32	10808	1	1800 s
		5	feasible	5161.47	24147	4161.4	13.69%	5161.47	4455.13	11897	6	1800 s

Table B.17: *low*50-3 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	73	3924.17	-	-	-	116	56	18.54 s
		2	optimal	4512.96	206	4248.45	-	-	-	121	49	19.88 s
		3	optimal	4451.44	8655	4322.34	-	-	-	652	38	111.65 s
		4	optimal	4405.84	136	4365.34				122	8	10.02 s
		5	optimal	4218.37	2494	4109.88	-	-	-	113	98	53.46 s
	G	1	feasible	4874.57	18577	4518.59	2.78%	4874.57	4738.57	7419	132	1800 s
		2	feasible	4487.76	123874	3875.57	5.38%	4487.76	4246.32	2705	115	1800 s
		3	feasible	4882.57	18077	3950.72	16.82%	4882.57	4061.58	13597	173	1800 s
		4	optimal	4541.02	49967	4391.13	-	-	-	3089	108	1230.92 s
		5	feasible	4856.02	38649	4236.36	9.25%	4856.02	4407	12186	123	1800 s
2	B	1	feasible	4658.86	140000	4268.01	4.73%	4658.86	4438.59	937	149	1800 s
		2	feasible	4977.17	151748	4478.19	4.25%	4977.17	4765.8	920	150	1800 s
		3	feasible	4709.44	139800	4463.51	0.73%	4709.44	4675.18	1301	115	1800 s
		4	optimal	4662.84	29861	4539.13	-	-	-	1109	134	1219.56 s
		5	feasible	7351.76	37100	4325.74	40.37%	7351.76	4384.22	8624	144	1800 s
	G	1	feasible	4876.36	82003	4478.09	1.99%	4876.36	4779.26	2670	220	1800 s
		2	unknown	-	27278	3835.82	-	-	4045.46	14970	164	1800 s
		3	unknown	-	6217	3927.65	-	-	4032.28	22489	313	1800 s
		4	unknown	-	11020	4364.56	-	-	4429.56	20813	145	1800 s
		5	unknown	-	5752	4250.91	-	-	4322.05	25905	189	1800 s
3	B	1	unknown	-	131100	4713.36	-	-	4890.15	1317	214	1800 s
		2	unknown	-	159420	4801.35	-	-	4952.77	1926	181	1800 s
		3	unknown	-	130002	4698.83	-	-	4873.06	1900	159	1800 s
		4	unknown	-	54238	4823.77	-	-	4936.89	1792	169	1800 s
		5	unknown	-	111609	4688.04	-	-	4767.14	1906	228	1800 s
	G	1	feasible	5018.33	20228	4539.14	6.96%	5018.11	4669.11	11887	333	1800 s
		2	feasible	5177.76	9035	3870.14	20.32%	5177.76	4125.87	14938	213	1800 s
		3	feasible	5683.95	8282	4043.22	26.71%	5683.95	4165.83	13682	485	1800 s
		4	feasible	4576.25	39590	4393.12	1.17%	4576.25	4522.75	4436	202	1800 s
		5	feasible	5038.42	24959	4359.62	7.51%	5038.42	4660.03	4153	332	1800 s

Table B.18: *low*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	58	3929.68	-	-	-	186	524	26.30 s
		2	optimal	4512.96	114	4250.1	-	-	-	143	265	20.87 s
		3	optimal	4451.44	1509	4326.95	-	-	-	696	533	71.14 s
		4	optimal	4405.84	113	4368.34				209	95	14.17 s
		5	optimal	4218.37	1388	4117.98	-	-	-	117	1027	46.23 s
	G	1	optimal	4874.57	21638	4648.06	-	-	-	6483	484	1055.17 s
		2	feasible	4473.59	26372	4166.96	0.78%	4473.59	4438.67	13514	836	1800 s
		3	feasible	4706.86	11979	4137.26	8.40%	4706.86	4311.42	14271	891	1800 s
		4	optimal	4541.02	8328	4453.66	-	-	-	892	473	109.78 s
		5	feasible	4812.82	3536	4585.54	2.47%	4812.82	4694.1	25971	1247	1800 s
2	B	1	feasible	5044.38	98909	4272.09	12.28%	5044.38	4424.71	1471	1876	1800 s
		2	feasible	4844.37	38530	4488.16	0.79%	4844.37	4805.89	4387	1361	1800 s
		3	optimal	4705.74	88138	4472.09	-	-	-	1000	1442	1530.53 s
		4	optimal	4662.84	19619	4547.18	-	-	-	1059	1748	391.16 s
		5	feasible	5234	39065	4333.95	15.82%	5234	4406.15	5462	1935	1800 s
	G	1	feasible	4876.36	34179	4618.63	1.60%	4876.36	4798.16	7428	1089	1800 s
		2	feasible	4475.4	39576	4129.08	0.51%	4475.4	4452.4	7843	937	1800 s
		3	feasible	4653.61	10871	4302.65	1.48%	4653.61	4584.62	13265	2687	1800 s
		4	feasible	4552.09	16524	4413.38	0.72%	4552.09	4519.16	11126	585	1800 s
		5	feasible	4848.38	12745	4494.94	4.60%	4848.38	4625.26	14502	1092	1800 s
3	B	1	unknown	-	109197	4725.55	-	-	4900.67	1484	2474	1800 s
		2	unknown	-	68639	4823.5	-	-	5091.52	2172	1982	1800 s
		3	unknown	-	107603	4705.58	-	-	4871.31	2436	1468	1800 s
		4	unknown	-	86804	4823.16	-	-	4970.69	2611	1742	1800 s
		5	unknown	-	137002	4693.92	-	-	4805.24	1981	2139	1800 s
	G	1	feasible	4890.86	103389	4655.64	0.52%	4890.86	4865.64	344	1149	1800 s
		2	feasible	4496.58	22886	4140.91	2.80%	4496.58	4370.79	9251	997	1800 s
		3	optimal	4664.7	43677	4351.75	-	-	-	718	1733	868.49 s
		4	optimal	4571.46	13288	4450.51	-	-	-	2458	725	385.87 s
		5	feasible	4982.94	28213	4613.33	2.44%	4092.94	4861.26	4505	1088	1800 s

Table B.19: *low*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	0	4047.18	-	-	-	1	162	5.27 s
		2	optimal	4512.96	2605	4448.81	-	-	-	1010	134	98.87 s
		3	optimal	4451.44	14	4408.99	-	-	-	17	319	18.67 s
		4	optimal	4405.84	45	4368				129	107	12.56 s
		5	optimal	4218.37	4	4214.62	-	-	-	16	205	10.93 s
	G	1	optimal	4874.57	6124	4710.78	-	-	-	2148	781	257.51 s
		2	optimal	4473.59	5681	4210.59	-	-	-	7604	675	577.83 s
		3	feasible	4650.12	16774	4304.78	3.96%	4650.12	4465.93	16399	1120	1800 s
		4	optimal	4541.02	2020	4461.09	-	-	-	2265	501	145.25 s
		5	feasible	4887.45	1338	4468.84	5.82%	4887.45	4603	26920	759	1800 s
2	B	1	feasible	5234.42	37209	4423.09	14.71%	5234.42	4464.65	4217	782	1800 s
		2	feasible	4939.18	121451	4698.04	3.54%	4939.18	4764.3	3170	319	1800 s
		3	optimal	4705.74	49233	4588.72	-	-	-	795	530	714.46 s
		4	optimal	4662.84	21039	4574.12	-	-	-	1650	660	488.81 s
		5	feasible	4901.49	72600	4453.41	8.04%	4901.49	4507.43	4640	221	1800 s
	G	1	feasible	4883.23	7832	4661.86	1.63%	4883.23	4803.64	13078	1092	1800 s
		2	optimal	4475.4	2475	4243.18	-	-	-	2240	1380	263.71 s
		3	feasible	4664.07	9800	4285.39	4.81%	4664.07	4439.62	15833	1477	1800 s
		4	optimal	4547.24	13471	4426.46	-	-	-	989	762	327.85 s
		5	feasible	4823.7	14340	4626.53	1.68%	4823.7	4742.87	10733	944	1800 s
3	B	1	feasible	5979.49	125000	4923.69	15.83%	5979.49	5032.89	1210	818	1800 s
		2	unknown	-	100082	4871.44	-	-	5024.62	3944	85	1800 s
		3	unknown	-	104400	4877.93	-	-	4971.39	2060	663	1800 s
		4	feasible	5648.86	102981	4938.43	9.66%	5648.86	5103.01	1738	1337	1800 s
		5	unknown	-	109089	4855.36	-	-	4936.99	2092	561	1800 s
	G	1	feasible	4890.86	18673	4712.29	0.66%	4890.86	4858.61	9009	1278	1800 s
		2	feasible	4496.65	13259	4229.6	1.33%	4496.65	4436.86	9847	857	1800 s
		3	optimal	4664.7	4903	4584	-	-	-	864	1374	174.95 s
		4	optimal	4571.46	3684	4480.42	-	-	-	410	757	170.15 s
		5	feasible	5962.07	9055	4662.44	19.92%	5962.07	4774.71	12393	1060	1800 s

Table B.20: *low*50-3 with Inequalities (3.6), (3.7) and (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	0	4047.18	-	-	-	0	170	3.40 s
		2	optimal	4512.96	8044	4449.85	-	-	-	868	197	132.60 s
		3	optimal	4451.44	34	4411.92	-	-	-	17	381	21.61 s
		4	optimal	4405.84	91	4370.84	-	-	-	111	97	10.76 s
		5	optimal	4218.37	7	4209.14	-	-	-	32	206	9.81 s
	G	1	optimal	4874.57	6124	4710.78	-	-	-	2148	781	268.69 s
		2	optimal	4473.59	5681	4210.59	-	-	-	7604	675	600.32 s
		3	feasible	4650.12	16752	4304.78	3.97%	4650.12	4465.69	16400	1120	1800 s
		4	optimal	4541.02	2020	4461.09	-	-	-	2265	501	144.74 s
		5	feasible	4887.44	1303	4468.84	5.83%	4887.44	4602.65	26292	759	1800 s
2	B	1	feasible	4888.82	95207	4422.52	8.17%	4888.82	4489.45	1605	521	1800 s
		2	feasible	4939.18	67888	4697.9	3.56%	4939.18	4763.33	4524	304	1800 s
		3	feasible	4726.78	92523	4588.59	1.15%	4726.78	4672.62	2378	324	1800 s
		4	optimal	4662.84	16898	4577.07	-	-	-	652	894	314.67 s
		5	feasible	4590.57	147337	4453.41	1.32%	4590.57	4529.82	1352	207	1800 s
	G	1	feasible	4883.23	7615	4661.86	1.66%	4883.23	4802.4	11876	1092	1800 s
		2	optimal	4475.4	2475	4243.18	-	-	-	2240	1380	269.13 s
		3	feasible	4664.07	9791	4285.39	4.81%	4664.07	4439.62	15748	1477	1800 s
		4	optimal	4547.24	13471	4426.46	-	-	-	989	762	327.33 s
		5	feasible	4823.7	14381	4626.53	1.67%	4823.7	4742.96	10733	944	1800 s
3	B	1	unknown	-	92810	4922.19	-	-	5038.86	1347	804	1800 s
		2	unknown	-	99200	4871.44	-	-	5024.47	3915	85	1800 s
		3	unknown	-	73117	4877.78	-	-	4974.77	2455	389	1800 s
		4	unknown	-	85916	4939.68	-	-	5132.59	1553	933	1800 s
		5	unknown	-	80528	4855.99	-	-	4962.61	1160	1209	1800 s
	G	1	feasible	4890.86	18989	4712.29	0.63%	4890.86	4859.85	9201	1278	1800 s
		2	feasible	4496.65	16047	4229.6	1.07%	4496.65	4448.46	10004	857	1800 s
		3	optimal	4664.7	4903	4584	-	-	-	864	1374	172.41 s
		4	optimal	4571.46	3684	4480.42	-	-	-	410	757	163.12 s
		5	feasible	5965.07	10252	4662.44	19.90%	5965.07	4777.95	12634	1060	1800 s

Table B.21: *low*50-3 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) and (3.19) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	4047.18	0	4047.18	-	-	-	0	175	4.09 s
		2	optimal	4512.96	8044	4449.85	-	-	-	868	229	132.17 s
		3	optimal	4451.44	34	4411.92	-	-	-	17	427	22.64 s
		4	optimal	4405.84	91	4370.84	-	-	-	111	153	11.44 s
		5	optimal	4218.37	7	4209.14	-	-	-	32	289	10.76 s
	G	1	optimal	4874.57	6124	4710.78	-	-	-	2148	901	234.4 s
		2	optimal	4473.59	5681	4210.59	-	-	-	7604	831	585.15 s
		3	feasible	4650.12	16755	4304.78	3.97%	4650.12	4465.72	16531	1315	1800 s
		4	optimal	4541.02	2020	4461.09	-	-	-	2265	744	150.69 s
		5	feasible	4887.44	1180	4468.84	5.88%	4887.44	4600.14	24527	1031	1800 s
2	B	1	feasible	4861.88	43792	4422.52	7.87%	4861.88	4479.29	2197	536	1800 s
		2	feasible	4939.18	67683	4697.9	3.56%	4939.18	4763.2	4533	341	1800 s
		3	feasible	4726.78	92917	4588.59	1.15%	4726.78	4672.78	2394	375	1800 s
		4	feasible	4662.84	43873	4575.49	0.19%	4662.84	4654.11	4660	863	1800 s
		5	feasible	4590.57	147180	4453.41	1.32%	4590.57	4529.8	1352	293	1800 s
	G	1	feasible	4883.23	7762	4661.86	1.64%	4883.23	4803.22	12423	1228	1800 s
		2	optimal	4475.4	2475	4243.18	-	-	-	2240	1549	265.52 s
		3	feasible	4664.07	9800	4285.39	4.81%	4664.07	4439.62	15845	1695	1800 s
		4	optimal	4547.24	13402	4426.46	-	-	-	989	1018	333.55 s
		5	feasible	4823.7	13470	4626.54	1.71%	4823.7	4741.1	10631	1228	1800 s
3	B	1	unknown	-	93202	4922.19	-	-	5038.87	1348	834	1800 s
		2	unknown	-	99299	4871.44	-	-	5024.5	3921	124	1800 s
		3	unknown	-	73403	4877.78	-	-	4974.89	2455	444	1800 s
		4	unknown	-	85288	4939.68	-	-	5132.51	1551	1013	1800 s
		5	unknown	-	92302	4877.06	-	-	4969.72	1557	1422	1800 s
	G	1	feasible	4890.86	18964	4712.29	0.64%	4890.86	4859.7	9201	1430	1800 s
		2	feasible	4496.65	16500	4229.6	1.04%	4496.65	4449.96	10004	1032	1800 s
		3	optimal	4664.7	4903	4584	-	-	-	864	1612	175.62 s
		4	optimal	4571.46	3684	4480.42	-	-	-	410	1022	164.67 s
		5	feasible	5962.07	9002	4662.44	19.92%	5962.07	4774.71	12393	1358	1800 s

Table B.22: *low*30-6 Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	8052.73	10949	7357.64	-	-	-	5200	0	563.31 s
		2	optimal	7629.99	12492	7212.05	-	-	-	1800	0	188.26 s
		3	optimal	8136.21	13446	7639.78	-	-	-	2153	0	221.94 s
		4	optimal	7501.8	7788	7234.6	-	-	-	1146	0	129.23 s
		5	feasible	7228.63	44794	6678.47	0.04%	7228.63	7225.52	5830	0	1800 s
	G	1	optimal	7179.95	3347	6453.65	-	-	-	1328	0	73.31 s
		2	feasible	8241.28	6002	7345.99	6.80%	8241.28	7680.67	32459	0	1800 s
		3	optimal	6871.53	2326	6105.13	-	-	-	4388	0	151.70 s
		4	optimal	7429.14	3348	6953.64	-	-	-	3286	0	132.91 s
		5	optimal	7106.38	33307	6561.24	-	-	-	6832	0	1568.84 s
2	B	1	unknown	-	79920	7403.39	-	-	7854.72	6334	0	1800 s
		2	unknown	-	49011	6923.97	-	-	7789.66	9036	0	1800 s
		3	feasible	8792.26	82979	7851.7	6.99%	8792.26	8178.09	3415	0	1800 s
		4	unknown	-	109420	7113.13	-	-	7475.93	4729	0	1800 s
		5	unknown	-	102520	6731.12	-	-	7196.66	4205	0	1800 s
	G	1	feasible	7962.92	51532	6198.67	9.58%	7962.92	7199.77	9079	0	1800 s
		2	unknown	-	23206	6927.52	-	-	7421.25	17506	0	1800 s
		3	feasible	7480.27	13487	6176.79	8.54%	7480.27	6841.62	17687	0	1800 s
		4	feasible	8806.6	29603	6925.92	17.54%	8806.6	7261.57	13862	0	1800 s
		5	unknown	-	14697	6600.77	-	-	6807.3	24378	0	1800 s
3	B	1	unknown	-	82605	7375.06	-	-	7884.89	4615	0	1800 s
		2	unknown	-	71019	6943.35	-	-	8017.21	4538	0	1800 s
		3	unknown	-	95069	7644.95	-	-	8236.29	3255	0	1800 s
		4	unknown	-	168253	7113.21	-	-	7928.37	2091	0	1800 s
		5	unknown	-	96113	6735.12	-	-	7240.79	3196	0	1800 s
	G	1	unknown	-	65002	6398.31	-	-	7316.4	6163	0	1800 s
		2	unknown	-	38146	6940.91	-	-	7458.05	7991	0	1800 s
		3	unknown	-	37638	6295.29	-	-	6917.18	7702	0	1800 s
		4	unknown	-	65565	6924.23	-	-	7617.98	4204	0	1800 s
		5	unknown	-	62297	6600.77	-	-	6962.97	8005	0	1800 s

Table B.23: *low*30-6 with Inequalities (3.6), (3.7) & (3.8) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	feasible	8096.19	32813	7394.23	1.40%	8096.19	7983.13	10076	115	1800 s
		2	optimal	7629.99	7131	7202.56	-	-	-	2382	124	272.52 s
		3	optimal	8136.21	9989	7642.39	-	-	-	2104	77	180.12 s
		4	optimal	7501.8	6177	7186.37	-	-	-	1609	142	155.47 s
		5	feasible	7228.63	37621	6685.58	0.24%	7228.63	7211.62	7129	105	1800 s
	G	1	optimal	7179.95	2348	6455.17	-	-	-	1573	122	67.08 s
		2	feasible	8259.33	7477	7456.06	6.04%	8259.33	7760.7	34903	165	1800 s
		3	optimal	6871.53	1492	6213	-	-	-	6240	146	170.70 s
		4	optimal	7429.14	2068	7002.26	-	-	-	7809	83	217.31 s
		5	feasible	7107.08	18283	6555.01	1.79%	7107.08	6979.96	11563	122	1800 s
2	B	1	unknown	-	63103	7434.45	-	-	7825.79	5825	144	1800 s
		2	unknown	-	54083	6927.18	-	-	7514.39	8795	121	1800 s
		3	feasible	9063.23	80682	7651.51	10.08%	9063.23	8149.17	4782	139	1800 s
		4	unknown	-	96880	7166.27	-	-	7558.96	4751	137	1800 s
		5	unknown	-	97807	6750.3	-	-	7159.81	4410	131	1800 s
	G	1	feasible	8709.99	42490	6485.06	18.33%	8709.99	7112.99	10743	183	1800 s
		2	unknown	-	14398	6905.95	-	-	7616.07	29930	85	1800 s
		3	feasible	7346.04	12400	6129.23	6.33%	7346.04	6881.23	16464	90	1800 s
		4	feasible	8938.74	29204	7006.68	18.06%	8938.74	7324.53	11207	109	1800 s
		5	feasible	8540.25	44835	6621.92	19.38%	8540.25	6885.42	12482	171	1800 s
3	B	1	unknown	-	68606	7226.34	-	-	7780.5	4506	125	1800 s
		2	unknown	-	91637	6951.68	-	-	7364.74	3839	135	1800 s
		3	unknown	-	103817	7862.57	-	-	8362.04	2972	174	1800 s
		4	unknown	-	154943	7114.13	-	-	7796.75	2399	116	1800 s
		5	unknown	-	96402	6762.71	-	-	7235.56	3423	138	1800 s
	G	1	unknown	-	62602	6201.52	-	-	7149.47	5410	109	1800 s
		2	unknown	-	50656	6911.93	-	-	7527.66	7338	91	1800 s
		3	unknown	-	50880	6152.2	-	-	6902.84	8233	127	1800 s
		4	feasible	22507.51	87265	6963.18	66.50%	22507.51	7539.36	4556	90	1800 s
		5	unknown	-	56425	6630.17	-	-	7006.58	8503	158	1800 s

Table B.24: *low*30-6 with Inequalities (3.6), (3.7) & (3.8) and (3.18) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	feasible	8057.33	112200	7679.82	1.65%	8057.33	7924.22	636	420	1800 s
		2	optimal	7629.99	11621	7232.24	-	-	-	2830	151	259.05 s
		3	optimal	8136.21	55033	7874.32	-	-	-	705	246	935.65 s
		4	optimal	7501.8	37061	7250.65	-	-	-	609	218	571.35 s
		5	feasible	7228.63	102213	6787.83	0.04%	7228.63	7200.24	760	308	1800 s
	G	1	feasible	7270.91	170300	6583.49	3.71%	7270.91	7000.92	480	639	1800 s
		2	feasible	8277.02	106800	7557.1	5.53%	8277.02	7819.09	514	830	1800 s
		3	feasible	7173.05	164400	6169.93	8.34%	7173.05	6574.59	766	491	1800 s
		4	optimal	7429.14	80011	7061.75	-	-	-	1622	523	1525.14 s
		5	feasible	7115.16	106900	6717.91	2.72%	7115.16	6921.33	1728	487	1800 s
2	B	1	feasible	9645.04	147119	8389.79	11.27%	9645.04	8557.69	1069	744	1800 s
		2	feasible	9108.91	213100	7776.64	12.95%	9108.91	7929.79	439	485	1800 s
		3	feasible	8442.43	71506	8196.24	0.64%	8442.43	8388.05	287	482	1800 s
		4	feasible	9744.97	157100	7624.12	20.30%	9744.97	7766.07	1332	602	1800 s
		5	unknown	-	140818	7312.45	-	-	7431.38	490	722	1800 s
	G	1	feasible	7683.9	91350	6937.22	4.36%	7683.9	7349.29	930	1003	1800 s
		2	feasible	8858.77	84100	8403.06	1.47%	8858.77	8728.69	730	1328	1800 s
		3	feasible	7324.29	127200	6447.73	7.92%	7324.29	6744.34	785	791	1800 s
		4	feasible	8036.38	132931	7264.44	6.30%	8036.38	7529.87	562	858	1800 s
		5	feasible	7608.3	118864	7151.49	3.73%	7608.3	7324.26	446	980	1800 s
3	B	1	unknown	-	102000	9317.55	-	-	9436.55	946	711	1800 s
		2	feasible	12693.86	151000	8600.52	31.20%	12693.86	8732.85	629	902	1800 s
		3	unknown	-	114972	8771	-	-	8881.79	922	626	1800 s
		4	unknown	-	118330	8349.72	-	-	8499.01	389	917	1800 s
		5	unknown	-	111603	8103.77	-	-	8211.93	345	875	1800 s
	G	1	unknown	-	104782	7686.98	-	-	7860.41	563	1273	1800 s
		2	unknown	-	133318	9558.85	-	-	9702.95	842	1584	1800 s
		3	feasible	8977.23	101200	6884.75	20.80%	8977.23	7109.97	684	1289	1800 s
		4	feasible	9102.86	111215	7811.97	11.94%	9102.97	8015.99	1058	1093	1800 s
		5	feasible	9521.1	140400	7827.2	16.21%	9521.1	7978.2	567	1203	1800 s

Table B.25: *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18) and (3.20) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	8052.73	6685	7911.42	-	-	-	540	1259	152.45 s
		2	optimal	7629.99	8605	7293.08	-	-	-	2091	312	167.17 s
		3	optimal	8136.21	9377	7972.8	-	-	-	675	591	153.44 s
		4	optimal	7501.8	1775	7319.44	-	-	-	595	653	40.69 s
		5	optimal	7228.63	34664	6993	-	-	-	1316	1125	963.57 s
	G	1	optimal	7179.95	3281	6947.94	-	-	-	542	940	76.59 s
		2	optimal	8164.5	13136	8014.68	-	-	-	93	1811	170.06 s
		3	optimal	6871.53	522	6735.4	-	-	-	589	770	31.52 s
		4	optimal	7429.14	2062	7251.44	-	-	-	1987	907	92.03 s
		5	optimal	7107.08	6456	6918.69	-	-	-	334	1414	105.64 s
2	B	1	feasible	9494.97	181100	8508.62	8.77%	9494.97	8661.9	871	1099	1800 s
		2	unknown	-	156207	7817.59	-	-	8031.59	1713	1136	1800 s
		3	feasible	8442.43	129747	8304.77	0.08%	8442.43	8435.61	177	1230	1800 s
		4	unknown	-	129432	7633.42	-	-	7745.25	1214	1179	1800 s
		5	unknown	-	143016	7398.84	-	-	7554.69	1161	1473	1800 s
	G	1	feasible	7689.42	109206	7206.36	2.25%	7689.42	7516.11	492	1499	1800 s
		2	feasible	8848.61	80790	8511.09	0.05%	8848.61	8808.02	2116	2856	1800 s
		3	feasible	7184.42	139301	6883.43	2.36%	7184.42	7014.68	732	1705	1800 s
		4	feasible	7969.67	83894	7408.26	4.58%	7969.67	7604.74	1296	1708	1800 s
		5	feasible	7581.86	153443	7235.25	2.82%	7581.86	7367.88	267	2021	1800 s
3	B	1	unknown	-	115413	9434.87	-	-	9547.99	760	1573	1800 s
		2	unknown	-	139950	8618.66	-	-	8737.67	613	1200	1800 s
		3	unknown	-	127817	8851.17	-	-	8922.95	815	1223	1800 s
		4	unknown	-	95158	8373.38	-	-	8477.98	605	1473	1800 s
		5	unknown	-	118447	8167.51	-	-	8277.3	400	1522	1800 s
	G	1	feasible	9858.57	128805	7835.06	18.64%	9858.57	8021.24	836	1955	1800 s
		2	unknown	-	125465	9603.74	-	-	9758.02	1438	3002	1800 s
		3	feasible	8086.98	150200	7164.23	9.56%	8086.98	7314	253	2013	1800 s
		4	feasible	10130.91	114900	7929.08	20.60%	10130.91	8044.29	949	2200	1800 s
		5	feasible	8646.18	170200	7873.95	6.92%	8648.18	8048	431	2053	1800 s

Table B.26: *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20) and (3.17) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	8052.73	4814	7966.99	-	-	-	515	894	93.28 s
		2	optimal	7629.99	7126	7438.45	-	-	-	532	562	201.36 s
		3	optimal	8136.21	4169	8007.58	-	-	-	871	719	140.34 s
		4	optimal	7501.8	3350	7399.29	-	-	-	577	483	69.90 s
		5	optimal	7228.63	8406	7118	-	-	-	693	1196	212.76 s
	G	1	optimal	7179.95	18	7166.89	-	-	-	62	788	58.81 s
		2	optimal	8164.5	0	8164.5	-	-	-	45	1632	68.97 s
		3	optimal	6871.53	0	6871.53	-	-	-	56	744	52.27 s
		4	optimal	7429.14	0	7429.14	-	-	-	79	539	25.01 s
		5	optimal	7107.08	26	7094.65	-	-	-	7	834	46.73 s
2	B	1	feasible	9371.42	106870	8753.61	5.60%	9371.42	8846.54	1314	1032	1800 s
		2	feasible	9163.67	133802	8024.04	10.82%	9163.67	8172.34	628	981	1800 s
		3	optimal	8442.43	7401	8417.11	-	-	-	247	809	157.37 s
		4	unknown	-	107200	7731.58	-	-	7854.88	917	1200	1800 s
		5	unknown	-	74811	7613.74	-	-	7689.29	988	1962	1800 s
	G	1	feasible	7683.9	93100	7430	1.72%	7683.9	7551.61	367	1221	1800 s
		2	optimal	8848.61	4164	8771.1	-	-	-	286	1654	157.61 s
		3	feasible	7152.52	67943	7030.59	0.63%	7152.52	7107.17	1180	1656	1800 s
		4	feasible	8002.85	110770	7612.47	3.87%	8002.85	7692.88	678	2468	1800 s
		5	feasible	7563.91	100153	7393.81	0.18%	7563.91	7550.26	633	1105	1800 s
3	B	1	unknown	-	63609	9639.07	-	-	9809.85	492	2200	1800 s
		2	unknown	-	102527	8826.44	-	-	9005.04	915	1831	1800 s
		3	unknown	-	79035	9065.4	-	-	9088.59	742	4804	1800 s
		4	unknown	-	74510	8519.95	-	-	8716.9	548	2457	1800 s
		5	unknown	-	61080	8411.44	-	-	8536.68	365	2837	1800 s
	G	1	feasible	9829.77	103836	8049.7	16.58%	9829.77	8199.84	887	1720	1800 s
		2	unknown	-	73322	9887.78	-	-	10004.31	849	2346	1800 s
		3	unknown	-	69821	7351.36	-	-	7504.38	1422	1769	1800 s
		4	feasible	11979.55	59878	8123.02	31.11%	11979.55	8252.86	1387	3160	1800 s
		5	feasible	8528.68	136700	8161.65	2.28%	8528.68	8334.24	359	1832	1800 s

Table B.27: *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15) and (3.24) & (3.25) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	8052.73	4814	7966.99	-	-	-	515	895	93.77 s
		2	optimal	7629.99	7126	7438.45	-	-	-	532	562	201.42 s
		3	optimal	8136.21	4169	8007.58	-	-	-	871	722	143.65 s
		4	optimal	7501.8	3350	7399.29	-	-	-	577	483	71.39 s
		5	optimal	7228.63	6598	7119	-	-	-	829	1166	187.71 s
	G	1	optimal	7180.49	18	7166.89	-	-	-	62	789	55.53 s
		2	optimal	8164.5	0	8164.5	-	-	-	45	1632	65.98 s
		3	optimal	6871.53	0	6871.53	-	-	-	56	744	49.21 s
		4	optimal	7429.14	0	7429.14	-	-	-	79	539	23.99 s
		5	optimal	7107.08	26	7094.65	-	-	-	7	834	60.92 s
2	B	1	feasible	9479.77	106100	8753.61	6.68%	9479.77	8846.22	1200	1033	1800 s
		2	feasible	9338.65	134400	8024.04	12.49%	9338.65	8172.43	597	981	1800 s
		3	optimal	8442.43	19724	8416.53	-	-	-	201	1019	287.07 s
		4	unknown	-	97181	7730.44	-	-	7835.28	905	1472	1800 s
		5	unknown	-	74218	7613.74	-	-	7689.2	988	1962	1800 s
	G	1	feasible	7683.9	90769	7430	1.73%	7683.9	7550.92	367	1221	1800 s
		2	optimal	8848.61	4164	8771.1	-	-	-	286	1654	155.17 s
		3	feasible	7152.52	67235	7030.59	0.64%	7152.52	7106.9	1159	1656	1800 s
		4	feasible	8002.85	109000	7612.47	3.88%	8002.85	7692.54	693	2468	1800 s
		5	feasible	7563.91	97900	7393.81	0.20%	7563.91	7549	633	1105	1800 s
3	B	1	unknown	-	64402	9639.07	-	-	9809.97	501	2200	1800 s
		2	unknown	-	107578	8831.45	-	-	9001.69	753	1590	1800 s
		3	unknown	-	79924	9061.07	-	-	9081.96	649	4025	1800 s
		4	unknown	-	74097	8519.95	-	-	8716.63	548	2457	1800 s
		5	unknown	-	61420	8411.44	-	-	8536.98	365	2837	1800 s
	G	1	unknown	-	86998	8044.89	-	-	8200.07	1111	1570	1800 s
		2	unknown	-	71212	9887.78	-	-	10013.89	847	2346	1800 s
		3	unknown	-	69467	7351.36	-	-	7504.33	1416	1769	1800 s
		4	feasible	29081.54	59503	8123.02	71.61%	29081.54	8252.83	1102	3160	1800 s
		5	feasible	8528.68	134312	8161.65	2.28%	8528.68	8334.04	359	1832	1800 s

Table B.28: *low*30-6 with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17), (3.15), (3.24) & (3.25) and (3.19) Results

m	Ins.	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
1	B	1	optimal	8052.73	4814	7966.99	-	-	-	515	905	92.94 s
		2	optimal	7629.99	7126	7438.45	-	-	-	532	601	214.69 s
		3	optimal	8136.21	4169	8007.58	-	-	-	871	789	158.54 s
		4	optimal	7501.8	5442	7399.21	-	-	-	561	544	90.38 s
		5	optimal	7228.63	6598	7119	-	-	-	829	1276	225.56 s
	G	1	optimal	7179.95	18	7166.89	-	-	-	62	936	115.82 s
		2	optimal	8164.5	6	8163.4	-	-	-	32	1888	88.16 s
		3	optimal	6871.53	0	6871.53	-	-	-	56	992	80.99 s
		4	optimal	7429.14	0	7429.14	-	-	-	79	824	35.02 s
		5	optimal	7107.08	13	7094.65	-	-	-	8	1158	71.84 s
2	B	1	unknown	-	86780	8753.61	-	-	8842.82	2092	1054	1800 s
		2	feasible	9338.65	127819	8024.04	12.50%	9338.65	8171.48	593	1031	1800 s
		3	optimal	8442.43	2623	8415.71	-	-	-	167	1073	138.18 s
		4	unknown	-	95111	7730.44	-	-	7834.94	841	1564	1800 s
		5	unknown	-	60858	7613.92	-	-	7683.36	1073	2370	1800 s
	G	1	feasible	7657.9	88550	7435.74	1.13%	7657.9	7571.06	540	1233	1800 s
		2	optimal	8848.61	8381	8772.26	-	-	-	187	1645	323.29 s
		3	feasible	7247.87	56913	7030.36	2.34%	7247.87	7078.49	2412	1641	1800 s
		4	feasible	8002.85	106700	7612.47	3.88%	8002.85	7692.3	718	2772	1800 s
		5	optimal	7563.91	26844	7394.93	-	-	-	839	1484	680.88 s
3	B	1	unknown	-	70268	9632.69	-	-	9812.25	653	2117	1800 s
		2	unknown	-	107644	8831.45	-	-	9001.72	619	1649	1800 s
		3	unknown	-	63294	9058.48	-	-	9081.18	443	6539	1800 s
		4	unknown	-	65822	8519.95	-	-	8671.69	426	2104	1800 s
		5	unknown	-	48669	8404.11	-	-	8540.06	206	3538	1800 s
	G	1	unknown	-	76478	8069.15	-	-	8192.07	1323	1730	1800 s
		2	unknown	-	70675	9892.06	-	-	10030.24	865	2397	1800 s
		3	unknown	-	81200	7347.29	-	-	7444.05	983	1752	1800 s
		4	feasible	26551	65400	8044.21	69.15%	26551	8191.07	832	3072	1800 s
		5	feasible	8554.26	136976	8103.26	2.79%	8554.26	8315.81	276	2162	1800 s

Table B.29: Large Instances with Inequalities (3.6), (3.7) & (3.8), (3.18), (3.20), (3.17) and (3.15) Results

Ineq (3.24) & (3.25)	Instance	No	Status	Obj.	Nodes	Root LB	Opt Gap	BUB	BLB	Lazy	User	CPU
with	$high$75-3	1	optimal	17312.05	6463	17200.83	-	-	-	2929	772	1100.8 s
		2	optimal	19485.18	2500	19424.54	-	-	-	1147	1156	459.67 s
		3	optimal	19216.74	682	19124.57	-	-	-	20	894	117.5 s
		4	optimal	19842.19	535	19805.73	-	-	-	137	1045	144.63 s
		5	feasible	18247.55	10216	18102.94	0.14%	18247	18222	5672	733	1800 s
	$high$100-3	1	feasible	22904.1	7809	22418.53	1.87%	22904.1	22475.4	1528	1074	1800 s
		2	feasible	22424.06	13191	22237.35	0.44%	22424.06	22326.07	1849	607	1800 s
		3	feasible	27857.9	655	24146.86	13.29%	27857.06	24155.37	4264	812	1800 s
		4	unknown	-	3382	25842.48	-	-	25888.86	1899	1802	1800 s
		5	feasible	22650.91	15829	22443.61	0.46%	22650.91	22547.16	1423	813	1800 s
	$high$125-3	1	unknown	-	5553	28769.68	-	-	28806.05	799	932	1800 s
		2	unknown	-	1769	27814.81	-	-	27828.81	2168	1245	1800 s
		3	unknown	-	2054	28156.57	-	-	28177.97	2509	1086	1800 s
		4	feasible	29381.41	2915	29266.79	-	-	29340.07	80	3939	1800 s
		5	unknown	-	5610	28671.67	-	-	28721.32	300	2848	1800 s
without	$high$75-3	1	optimal	17312.05	7929	17207.12	-	-	-	3042	999	1111.92 s
		2	optimal	19485.18	1125	19433.37	-	-	-	377	1035	225.37 s
		3	optimal	19216.74	434	19125.14	-	-	-	12	876	118.54 s
		4	optimal	19842.19	414	19805.8	-	-	-	219	934	152.42 s
		5	optimal	18247.55	12816	18096.09	-	-	-	851	783	911.19 s
	$high$100-3	1	optimal	17312.05	7929	17207.12	-	-	-	3042	999	1111.92 s
		2	optimal	19485.18	1125	19433.37	-	-	-	377	1035	225.37 s
		3	optimal	19216.74	434	19125.14	-	-	-	12	876	118.54 s
		4	optimal	19842.19	414	19805.8	-	-	-	219	934	152.42 s
		5	optimal	18247.55	12816	18096.09	-	-	-	851	783	911.19 s
	$high$125-3	1	feasible	28877.11	5969	28768.03	0.14%	28877.11	28837.86	534	1062	1800 s
		2	unknown	-	370	27821.03	-	-	27829.29	1601	1208	1800 s
		3	unknown	-	7195	28155.55	-	-	28190.86	567	1258	1800 s
		4	feasible	29433.61	5442	29258.3	4.66%	29433.61	29296.54	184	2664	1800 s
		5	feasible	30915.1	4252	28659.22	7.19%	30915.1	28692.96	523	1827	1800 s

Appendix C

Changes in the Results with the Inequalities

In Tables C.1-C.21, we report the percentages of the changes in the lower bound at the root node (Root LB), total time spent (CPU), optimality gap (Opt Gap) and best lower bound (BLB) found in the time limit while valid inequalities are used cumulatively. In the tables, the row corresponding to "none" demonstrates the values of the Root LB, CPU, Opt Gap (%) or BLB when none of the valid inequalities is used. Following rows demonstrate the percentage improvements in the Root LB, CPU, Opt Gap or BLB if the corresponding inequalities are used as well as the inequalities in the previous rows.

Table C.1: Changes in Root LB for Three Periods and One Vehicle

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	14224.62	14501.3	14917.16	16359.94	15516.08
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	-0.01%
(3.18)	1.62%	0.76%	1.55%	0.65%	0.27%
(3.20)	0.11%	0.00%	0.08%	0.07%	0.12%
(3.17)	0.73%	2.30%	0.49%	-0.02%	0.48%
(3.15), (3.24) & (3.25)	-0.04%	0.00%	-0.01%	0.02%	-0.01%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%
	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	18546.74	16361.88	17810.14	16404.56	17504.89
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.37%	1.91%	0.37%	0.34%	0.85%
(3.20)	3.08%	2.07%	1.56%	1.03%	0.02%
(3.17)	-0.09%	-0.03%	0.08%	0.19%	0.05%
(3.15), (3.24) & (3.25)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%
	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	3718.05	4099.48	4307.61	4319.14	4042.48
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	5.54%	3.63%	0.34%	1.07%	1.67%
(3.20)	0.14%	0.04%	0.11%	0.07%	0.20%
(3.17)	2.99%	4.68%	1.90%	-0.01%	2.35%
(3.15), (3.24) & (3.25)	0.00%	0.02%	0.07%	0.07%	-0.13%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%
	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	4502.66	3765.39	3894.61	4325.99	4183.55
(3.6), (3.7) & (3.8)	-1.68%	0.00%	0.00%	0.00%	0.00%
(3.18)	2.07%	2.93%	1.44%	1.51%	1.26%
(3.20)	2.87%	7.52%	4.72%	1.42%	8.24%
(3.17)	1.35%	1.05%	4.05%	0.17%	-2.54%
(3.15), (3.24) & (3.25)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

Table C.2: Changes in Root LB for Six Periods and One Vehicle

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	22048.86	19440.36	22527.68	17230.32	18131.85
(3.6), (3.7) & (3.8)	0.00%	2.24%	-0.02%	0.02%	0.00%
(3.18)	1.68%	-2.03%	1.30%	0.27%	0.99%
(3.20)	0.85%	0.36%	0.51%	0.32%	0.90%
(3.17)	0.28%	0.72%	0.06%	0.17%	0.62%
(3.15), (3.24) & (3.25)	0.00%	-0.01%	0.00%	0.00%	-0.02%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.02%

	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	25286.57	25893.17	24096.58	25994.66	25115.23
(3.6), (3.7) & (3.8)	-0.46%	-0.16%	0.35%	0.60%	-0.06%
(3.18)	0.53%	0.22%	0.25%	0.47%	0.81%
(3.20)	0.12%	0.44%	1.17%	0.61%	0.57%
(3.17)	0.40%	0.46%	0.32%	0.61%	0.41%
(3.15), (3.24) & (3.25)	0.00%	0.00%	0.00%	-0.04%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	7357.64	7212.05	7639.78	7234.6	6678.47
(3.6), (3.7) & (3.8)	0.50%	-0.13%	0.03%	-0.67%	0.11%
(3.18)	3.86%	0.41%	3.03%	0.89%	1.53%
(3.20)	3.02%	0.84%	1.25%	0.95%	3.02%
(3.17)	0.70%	1.99%	0.44%	1.09%	1.79%
(3.15), (3.24) & (3.25)	0.00%	0.00%	0.00%	0.00%	0.01%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	6453.65	7345.99	6105.13	6953.64	6561.24
(3.6), (3.7) & (3.8)	0.02%	1.50%	1.77%	0.70%	-0.09%
(3.18)	1.99%	1.36%	-0.69%	0.85%	2.49%
(3.20)	5.54%	6.05%	9.16%	2.69%	2.99%
(3.17)	3.15%	1.87%	2.02%	2.45%	2.54%
(3.15), (3.24) & (3.25)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	-0.01%	0.00%	0.00%	0.00%

Table C.3: Changes in CPU for Three Periods and One Vehicle

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	135.7	43.5	277.31	19.76	1618.86
(3.6), (3.7) & (3.8)	-0.69%	0.16%	2.97%	-5.21%	30.72%
(3.18)	85.02%	-9.09%	83.00%	19.72%	91.54%
(3.20)	7.43%	5.77%	51.56%	-74.74%	26.19%
(3.17)	56.90%	31.74%	-35.44%	-31.11%	67.17%
(3.15), (3.24) & (3.25)	19.89%	-15.29%	10.16%	12.67%	-122.07%
(3.19)	-8.72%	8.71%	-6.27%	-1.68%	-0.65%

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	1800	26.27
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	10.92%
(3.18)	0.00%	0.00%	0.00%	0.00%	25.00%
(3.20)	96.30%	0.00%	30.60%	0.00%	16.22%
(3.17)	-149.39%	0.00%	75.37%	87.31%	-35.27%
(3.15), (3.24) & (3.25)	-2.63%	0.00%	-0.07%	-3.03%	-6.47%
(3.19)	1.94%	0.00%	-3.70%	-2.32%	4.58%

	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	21.02	42.45	234.03	9.01	154.54
(3.6), (3.7) & (3.8)	0.76%	-1.25%	51.26%	-4.42%	-15.70%
(3.18)	11.12%	53.75%	2.11%	-6.51%	70.10%
(3.20)	-41.86%	-4.98%	36.28%	-41.42%	13.52%
(3.17)	79.96%	-373.74%	73.76%	11.36%	76.36%
(3.15), (3.24) & (3.25)	35.48%	-34.12%	-15.75%	14.33%	10.25%
(3.19)	-20.29%	0.32%	-4.77%	-6.32%	-9.68%

	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	388.7	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	-1.90%	0.00%
(3.18)	0.00%	0.00%	0.00%	-210.77%	0.00%
(3.20)	41.38%	0.00%	0.00%	91.08%	0.00%
(3.17)	75.60%	67.90%	0.00%	-32.31%	0.00%
(3.15), (3.24) & (3.25)	-4.34%	-3.89%	0.00%	0.35%	0.00%
(3.19)	12.76%	2.53%	0.00%	-4.11%	0.00%

Table C.4: Changes in CPU for Six Periods and One Vehicle

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	419.712	188.953	146.455	90.064	1800
(3.6), (3.7) & (3.8)	-125.53%	24.28%	34.20%	-24.52%	0.00%
(3.18)	-90.16%	-33.76%	-1700.71%	-13.24%	0.00%
(3.20)	95.26%	23.99%	97.38%	76.88%	86.34%
(3.17)	-26.81%	-8.21%	-127.84%	-37.54%	17.87%
(3.15), (3.24) & (3.25)	3.42%	1.80%	0.70%	-1.93%	11.34%
(3.19)	1.00%	-1.33%	-4.29%	-11.05%	-36.30%
	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	72.202	53.918	598.608	690.42	92.644
(3.6), (3.7) & (3.8)	-51.43%	-2.17%	55.60%	14.28%	5.23%
(3.18)	-5.42%	-369.55%	-313.87%	-64.42%	-1950.07%
(3.20)	47.47%	71.86%	94.85%	65.48%	95.37%
(3.17)	15.02%	-1.89%	4.19%	71.12%	-6.74%
(3.15), (3.24) & (3.25)	-1.21%	-3.24%	0.22%	-9.42%	1.43%
(3.19)	-63.27%	-17.47%	-124.02%	-7.82%	-24.60%
	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	563.309	188.263	221.942	129.227	1800
(3.6), (3.7) & (3.8)	-219.54%	-44.76%	18.85%	-20.31%	0.00%
(3.18)	0.00%	4.94%	-419.47%	-267.50%	0.00%
(3.20)	91.53%	35.47%	83.60%	92.88%	46.47%
(3.17)	38.81%	-20.45%	8.54%	-71.80%	77.92%
(3.15), (3.24) & (3.25)	-0.53%	-0.03%	-2.36%	-2.13%	11.77%
(3.19)	0.89%	-6.59	-10.37%	-26.60%	-20.16%
	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	73.31	1800	151.697	132.91	1568.835
(3.6), (3.7) & (3.8)	8.50%	0.00%	-12.53%	-63.50%	-14.73%
(3.18)	-2583.52%	0.00%	-954.49%	-601.84%	0.00%
(3.20)	95.74%	90.55%	98.25%	93.97%	94.13%
(3.17)	23.22%	59.44%	-65.84%	72.82%	55.77%
(3.15), (3.24) & (3.25)	5.58%	4.33%	5.86%	4.09%	-30.38%
(3.19)	-108.57%	-33.62%	-64.58%	-45.98%	-17.93%

Table C.5: Changes in Opt Gap One Vehicle

Inequalities	*high*50-3-*gen* 1	2	3	4	5
none	1.69	2.25	0.52	3.25	-
(3.6), (3.7) & (3.8)	-20.71%	11.11%	0.00%	15.69%	-
(3.18)	-23.53%	-18.00%	-153.85%	31.75%	-
(3.20)	100.00%	91.95%	100.00%	59.89%	-
(3.17)	-	-121.05%	-	100.00%	-
(3.15), (3.24) & (3.25)	-	0.00%	-	-	-
(3.19)	-	7.14%	-	-	-

Inequalities	*low*50-3-*gen* 1	2	3	4	5
none	1.22	4.45	28.50	-	8.77
(3.6), (3.7) & (3.8)	-113.93%	-6.07%	0.00%	-	17.79%
(3.18)	-6.51%	-13.98%	40.98%	-	-28.29%
(3.20)	100.00%	85.50%	50.06%	-	73.30%
(3.17)	-	100.00%	52.86%	-	-135.63%
(3.15), (3.24) & (3.25)	-	-	-0.25%	-	-0.17%
(3.19)	-	-	0.00%	-	-0,86%

Table C.6: Changes in BLB One Vehicle

Inequalities	*high*50-3-*gen* 1	2	3	4	5
none	19056.48	16947.19	18217.4	16541.94	-
(3.6), (3.7) & (3.8)	-0.38%	0.03%	0.02%	0.01%	-
(3.18)	-0.59%	-0.22%	-0.71%	0.35%	-
(3.20)	-	1.76%	-	0.81%	-
(3.17)	-	-0.14%	-	-	-
(3.15), (3.24) & (3.25)	-	0.00%	-	-	-
(3.19)	-	-0.01%	-	-	-

Inequalities	*low*50-3-*gen* 1	2	3	4	5
none	4819.12	4274.99	3974.29	-	4458.5
(3.6), (3.7) & (3.8)	-1.01%	0.43%	0.00%	-	0.18%
(3.18)	-0.67%	-1.09%	2.20%	-	-1.34%
(3.20)	-	4.53%	6.15%	-	6.51%
(3.17)	-	-	3.58%	-	-1.94%
(3.15), (3.24) & (3.25)	-	-	-0.01%	-	-0.01%
(3.19)	-	-	0.00%	-	-0.05%

Table C.7: Changes in Root LB for Three Periods and Two Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	14382.36	14412.8	14957.57	16287.79	15581.82
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	2.77%	4.11%	2.22%	2.14%	1.28%
(3.20)	0.15%	0.14%	0.10%	0.12%	0.12%
(3.17)	0.93%	-0.27%	0.65%	0.02%	0.61%
(3.15)	0.00%	1.43%	0.00%	0.03%	-0.02%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	18439.34	16651.14	17695.23	16262.27	17396.18
(3.6), (3.7) & (3.8)	-0.01%	0.12%	0.00%	0.00%	0.00%
(3.18)	1.44%	-0.14%	0.59%	1.50%	1.55%
(3.20)	2.55%	2.03%	1.97%	1.05%	0.24%
(3.17)	-0.08%	-0.03%	-0.15%	0.16%	0.18%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	3677.96	4002.98	4207.12	4251.46	4047.81
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	16.04%	11.87%	6.09%	6.77%	6.87%
(3.20)	0.10%	0.22%	0.19%	0.18%	0.19%
(3.17)	3.53%	4.68%	2.61%	0.59%	2.76%
(3.15)	-0.01%	0.00%	0.00%	0.06%	0.08%
(3.19)	0.00%	0.00%	0.00%	-0.03%	0.00%

	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	4366.05	3796.6	3727.41	4288.75	4161.4
(3.6), (3.7) & (3.8)	0.00%	-0.76%	0.00%	0.00%	0.00%
(3.18)	2.57%	1.81%	5.37%	1.77%	2.15%
(3.20)	3.14%	7.65%	9.55%	1.12%	5.74%
(3.17)	0.94%	2.76%	-0.40%	0.30%	2.93%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

Table C.8: Changes in Root LB for Six Periods and Two Vehicles

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	22046.25	19168.24	22508.92	17156.75	18111.95
(3.6), (3.7) & (3.8)	0.42%	0.16%	0.04%	-0.01%	0.47%
(3.18)	4.39%	4.20%	2.85%	2.62%	3.42%
(3.20)	0.52%	0.17%	0.25%	0.26%	0.39%
(3.17)	0.90%	1.07%	0.49%	0.45%	0.81%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.01%	0.00%	0.00%	0.01%	0.05%

	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	25180.89	25837.21	24021.37	26016.94	25128.08
(3.6), (3.7) & (3.8)	0.16%	0.09%	0.42%	-0.37%	-0.07%
(3.18)	1.20%	0.83%	4.48%	2.66%	1.90%
(3.20)	0.18%	0.28%	-1.48%	0.72%	0.46%
(3.17)	0.15%	0.34%	0.31%	0.43%	0.05%
(3.15)	0.00%	0.00%	-0.01%	0.00%	0.00%
(3.19)	0.02%	0.00%	0.04%	-0.03%	0.00%

	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	7403.39	6923.97	7851.7	7113.13	6731.12
(3.6) (3.7) & (3.8)	0.42%	0.05%	-2.55%	0.75%	0.28%
(3.18)	12.85%	12.26%	7.12%	6.39%	8.33%
(3.20)	1.42%	0.53%	1.32%	0.12%	1.18%
(3.17)	2.88%	2.64%	1.35%	1.29%	2.90%
(3.15)	0.00%	0.00%	-0.01%	-0.01%	0.00%
(3.19)	0.00%	0.00%	-0.01%	0.00%	0.00%

	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	6198.67	6927.52	6176.79	6925.92	6600.77
(3.6), (3.7) & (3.8)	4.62%	-0.31%	-0.77%	1.17%	0.32%
(3.18)	6.97%	21.68%	5.20%	3.68%	8.00%
(3.20)	3.88%	1.29%	6.76%	1.98%	1.17%
(3.17)	3.10%	3.05%	2.14%	2.76%	2.19%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.08%	0.01%	0.00%	0.00%	0.02%

Table C.9: Changes in CPU for Three Periods and Two Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	285.545	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	-1.84%	0.00%
(3.18)	0.00%	18.03%	0.00%	-119.68%	0.00%
(3.20)	0.00%	-21.99%	32.07%	28.60%	0.00%
(3.17)	0.00%	0.00%	63.53%	19.70%	0.00%
(3.15)	0.00%	0.00%	-20.52%	40.38%	0.00%
(3.19)	0.00%	0.0%	0.16%	0.44%	0.00%

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	20.27%
(3.18)	0.00%	0.00%	0.00%	0.00%	95.04%
(3.20)	92.17%	0.00%	56.55%	0.00%	-26.74%
(3.17)	-37.57%	0.00%	35.05%	62.23%	4.11%
(3.15)	2.33%	0.00%	2.55%	-3.52%	-1.85%
(3.19)	-3.71%	0.00%	-3.96%	3.90%	0.60%

	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	595.38	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	-1.45%	0.00%
(3.18)	0.00%	0.00%	0.00%	-101.91%	0.00%
(3.20)	0.00%	0.00%	14.97%	67.93%	0.00%
(3.17)	0.00%	0.00%	53.32%	-24.96%	0.00%
(3.15)	0.00%	0.00%	-151.94%	35.63%	0.00%
(3.19)	0.00%	0.00%	0.00%	-472.03%	0.00%

	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.20)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.17)	0.00%	85.35%	0.00%	81.79%	0.00%
(3.15)	0.00%	-2.06%	0.00%	0.16%	0.00%
(3.19)	0.00%	1.34%	0.00%	-1.90%	0.00%

Table C.10: Changes in CPU for Six Periods and Two Vehicles

Inequalities	*high*30-6-*bench* 1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.20)	0.00%	0.00%	6.84%	0.00%	0.00%
(3.17)	0.00%	0.00%	92.42%	0.00%	0.00%
(3.15)	0.00%	0.00%	-2.95%	0.00%	0.00%
(3.19)	0.00%	0.00%	-0.31%	0.00%	0.00%

Inequalities	*high*30-6-*gen* 1	2	3	4	5
none	1800	526.00	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	-242.21%	0.00%	0.00%	0.00%
(3.18)	0.00%	90.95%	0.00%	0.00%	0.00%
(3.20)	0.00%	33.78%	0.00%	75.58%	75.11%
(3.17)	0.00%	15.69%	0.00%	-15.24%	-18.82%
(3.15)	0.00%	-0.30%	0.00%	-0.85%	-3.95%
(3.19)	0.00%	-18.82%	0.00%	-56.85%	-2.31%

Inequalities	*low*30-6-*bench* 1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.20)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.17)	0.00%	0.00%	91.26%	0.00%	0.00%
(3.15)	0.00%	0.00%	-8.42%	0.00%	0.00%
(3.19)	0.00%	0.00%	51.87%	0.00%	0.00%

Inequalities	*low*30-6-*gen* 1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.20)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.17)	0.00%	91.24%	0.00%	0.00%	0.00%
(3.15)	0.00%	1.55%	0.00%	0.00%	0.00%
(3.19)	0.00%	-108.35%	0.00%	0.00%	62.17%

Table C.11: Changes in Opt Gap for Three Periods and Two Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	15.51	-	4.81	-	7.35
(3.6), (3.7) & (3.8)	0.00%	-	-149.90%	-	-
(3.18)	45.00%	-	84.86%	-	-
(3.20)	83.59%	-	-	-	7.14%
(3.17)	0.71%	-1.52%	-	-	58.65%
(3.15)	65.47%	-504.48%	-	-	-311.63%
(3.19)	0.00%	7.16%	-	-	16.38%
	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	9.18	3.58	6.45	7.09	2.56
(3.6), (3.7) & (3.8)	46.30%	-7.82%	31.01%	48.24%	-
(3.18)	53.75%	16.06%	71.01%	-	-
(3.20)	-	87.96%	-	-	-
(3.17)	-	-25.64%	-	-	-
(3.15)	-	0.00%	-	-	-
(3.19)	-	0.00%	-	-	-
	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	34.19	-	8.98	-	16.51
(3.6), (3.7) & (3.8)	0.00%	-	-198.44%	-	-165.23%
(3.18)	86.17%	-	97.28%	-	7.81%
(3.20)	-159.62%	81.41%	-	-	60.81%
(3.17)	-19.79%	-348.10%	-	-	49.18%
(3.15)	44.46%	-0.56%	-	-	83.58%
(3.19)	3.67%	0.00%	0.00%	-	0.00%
	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	4.35	8.24	38.40	2.17	-
(3.6), (3.7) & (3.8)	-282.30%	5.95%	0.00%	-237.79%	-
(3.18)	88.03%	-	-	-	-
(3.20)	19.60%	-	-	-	-
(3.17)	-1.87%	-	-225.00%	-	63.48%
(3.15)	-1,84%	-	0.00%	-	0.60%
(3.19)	1.20%	-	0.00%	-	-2.40%

Table C.12: Changes in Opt Gap for Six Periods and Two Vehicles

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	-	-	1.58	8.39	-
(3.6), (3.7) & (3.8)	-	-	-8.23%	-43.62%	-
(3.18)	-	14.10%	85.38%	-	-
(3.20)	16.92%	43.36%	-	-	-
(3.17)	19.00%	-151.79%	-	-	-18.79%
(3.15)	-17,88%	0.78%	-	-	-
(3.19)	31.28%	-3.13%	-	-	-

	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	9.01	-	4.11	6.76	1.87
(3.6), (3.7) & (3.8)	74.47%	-	-23.11%	-	-75.40%
(3.18)	79.57%	-	78.66%	-	91.77%
(3.20)	10.64%	-	70.37%	-	-
(3.17)	50.00%	-	-40.63%	-	-
(3.15)	-42.86%	-	17.78%	-	-
(3.19)	40.00%	-	-51.35%	-	-

	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	-	-	6.99	-	-
(3.6), (3.7) & (3.8)	-	-	-44.21%	-	-
(3.18)	-	-	93.65%	-	-
(3.20)	22.18%	-	87.50%	-	-
(3.17)	36.15%	-	-	-	-
(3.15)	-19.29%	-15.43%	-	-	-
(3.19)	-	-0.08%	-	-	-

	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	9.58	-	8.54	17.54	-
(3.6), (3.7) & (3.8)	-91.34%	-	25.88%	-2.96%	-
(3.18)	76.21%	-	-25.12%	65.12%	80.75%
(3.20)	48.39%	96.60%	70.20%	27.30%	24.40%
(3.17)	23.56%	-	73.31%	15.50%	93.62%
(3.15)	-0.58%	-	-1.59%	-0.26%	-11.11%
(3.19)	34.68%	-	-265.63%	0.00%	-

Table C.13: Changes in BLB for Three Periods and Two Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	14640.28	14972.55	15336.41	-	15796.88
(3.6), (3.7) & (3.8)	0.00%	0.00%	-0.26%	-	0.06%
(3.18)	1.90%	2.46%	0.80%	-	0.63%
(3.20)	0.47%	-0.33%	-	-	0.16%
(3.17)	0.33%	-0.09%	-	-	0.33%
(3.15)	-0.02%	-0.08%	-	-	0.00%
(3.19)	0.00%	0.00%	-	-	0.00%

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	18809.4	16925.22	17837.28	16501.54	17596.78
(3.6), (3.7) & (3.8)	0.19%	-0.05%	0.84%	0.55%	-
(3.18)	0.75%	-0.14%	0.67%	0.28%	-
(3.20)	-	1.93%	-	1.11%	-
(3.17)	-	-0.10%	-	-	-
(3.15)	-	0.00%	-	-	-
(3.19)	-	0.00%	-	-	-

	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	4098.54	4451.79	4564.62	-	4331.99
(3.6), (3.7) & (3.8)	0.00%	0.00%	-0.81%	-	-0.25%
(3.18)	8.30%	7.05%	3.26%	-	1.46%
(3.20)	-0.31%	0.84%	-	-	0.50%
(3.17)	0.90%	-0.87%	-	-	2.30%
(3.15)	0.56%	-0.02%	-	-	0.50%
(3.19)	-0.23%	0.00%	0.00%	-	0.00%

	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	4709.53	4159.82	3894.49	4464.1	4353.69
(3.6), (3.7) & (3.8)	-1.64%	0.41%	0.00%	-0.78%	1.14%
(3.18)	3.17%	-3.14%	3.54%	0.01%	-1.84%
(3.20)	0.40%	10.06%	13.70%	2.02%	7.02%
(3.17)	0.11%	-	-3.16%	-	2.54%
(3.15)	-0.03%	-	0.00%	-	0.00%
(3.19)	0.00%	-	0.00%	-	-0.04%

Table C.14: Changes in BLB for Six Periods and Two Vehicles

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	22741.87	19924.19	23208.54	17642.82	18639.58
(3.6), (3.7) & (3.8)	-0.16%	0.39%	0.03%	0.16%	-0.60%
(3.18)	2.88%	1.28%	0.64%	0.83%	2.48%
(3.20)	0.41%	0.02%	-	-0.07%	0.46%
(3.17)	0.59%	0.78%	-	0.28%	0.48%
(3.15)	-0.02%	0.00%	-	0.01%	-0.63%
(3.19)	0.02%	0.00%	-	0.04%	0.63%
	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	25419.16	-	24609.81	26403.33	25544.58
(3.6), (3.7) & (3.8)	0.42%	-	-0.02%	-0.10%	-0.23%
(3.18)	1.00%	-	1.33%	2.13%	1.38%
(3.20)	0.09%	-	0.52%	-	-
(3.17)	0.15%	-	-0.01%	-	-
(3.15)	-0.07%	-	0.00%	-	-
(3.19)	0.10%	-	-0.04%	-	-
	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	7854.72	7789.66	8178.09	7475.93	7196.66
(3.6), (3.7) & (3.8)	-0.37%	-3.53%	-0.35%	1.11%	-0.51%
(3.18)	9.35%	5.53%	2.93%	2.74%	3.79%
(3.20)	1.22%	1.28%	0.57%	-0.27%	1.66%
(3.17)	2.13%	1.75%	-	1.42%	1.78%
(3.15)	0.00%	0.00%	-	-0.25%	0.00%
(3.19)	-0.04%	-0.01%	-	0.00%	-0.08%
	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	7199.77	7421.25	6841.62	7261.57	6807.3
(3.6), (3.7) & (3.8)	-1.21%	2.63%	0.58%	0.87%	1.15%
(3.18)	3.32%	14.61%	-1.99%	2.80%	6.37%
(3.20)	2.27%	0.91%	4.01%	0.99%	0.60%
(3.17)	0.47%	-	1.32%	1.16%	2.48%
(3.15)	-0.01%	-	0.00%	0.00%	-0.02%
(3.19)	0.27%	-	-0.40%	0.00%	-

Table C.15: Changes in Root LB for Three Periods and Three Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	14434.55	14412.8	14957.57	16287.79	15526.63
(3.6), (3.7) & (3.8)	-0.09%	0.00%	0.00%	0.00%	0.00%
(3.18)	5.62%	6.46%	3.79%	4.16%	4.01%
(3.20)	0.07%	0.12%	0.09%	0.00%	0.05%
(3.17)	1.43%	1.38%	0.86%	0.33%	0.92%
(3.15)	0.02%	-0.24%	0.04%	0.01%	0.01%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	18439.34	16651.14	17678.72	16262.27	17396.18
(3.6), (3.7) & (3.8)	-0.01%	0.12%	0.00%	0.00%	0.00%
(3.18)	1.96%	0.50%	1.47%	2.57%	2.42%
(3.20)	2.41%	1.97%	1.66%	0.72%	0.06%
(3.17)	-0.08%	-0.21%	0.43%	0.10%	0.59%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.03%	0.00%

	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	3677.96	4002.98	4207.12	4251.46	4083.82
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	28.15%	19.94%	11.69%	13.46%	14.79%
(3.20)	0.26%	0.46%	0.14%	-0.01%	0.13%
(3.17)	4.19%	0.99%	3.66%	2.39%	3.44%
(3.15)	-0.03%	0.00%	0.00%	0.03%	0.01%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.43%

	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	4366.05	3796.6	3727.41	4288.75	4161.4
(3.6), (3.7) & (3.8)	0.00%	-0.76%	0.00%	0.00%	0.00%
(3.18)	3.96%	2.72%	8.47%	2.43%	4.76%
(3.20)	2.57%	7.00%	7.63%	1.31%	5.82%
(3.17)	1.22%	2.14%	5.34%	0.67%	1.06%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%	0.00%

Table C.16: Changes in Root LB for Six Periods and Three Vehicles

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	22125.65	19421.16	22718.05	17160.08	18111.95
(3.6), (3.7) & (3.8)	-0.10%	-1.24%	-0.92%	0.17%	0.47%
(3.18)	8.89%	8.66%	5.43%	6.86%	7.62%
(3.20)	0.46%	0.05%	0.26%	0.10%	0.22%
(3.17)	0.89%	1.14%	0.66%	0.89%	0.79%
(3.15)	0.00%	0.00%	0.01%	0.00%	0.34%
(3.19)	0.00%	-0.05%	0.00%	-0.10%	-0.04%

	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	25184.01	25805.79	24021.37	25949.7	25099.74
(3.6), (3.7) & (3.8)	0.21%	0.01%	0.64%	0.18%	-0.24%
(3.18)	3.04%	2.17%	3.46%	5.03%	3.92%
(3.20)	0.01%	0.07%	1.18%	0.35%	0.19%
(3.17)	0.38%	0.42%	0.35%	0.50%	0.17%
(3.15)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.19)	-0.01%	0.00%	0.01%	0.03%	0.00%

	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	7375.06	6943.35	7644.95	7113.21	6735.12
(3.6), (3.7) & (3.8)	-2.02%	0.12%	2.85%	0.01%	0.41%
(3.18)	28.94%	23.72%	11.55%	17.37%	19.83%
(3.20)	1.26%	0.21%	0.91%	0.28%	0.79%
(3.17)	2.16%	2.41%	2.42%	1.75%	2.99%
(3.15)	0.00%	0.06%	-0.05%	0.00%	0.00%
(3.19)	-0.07%	0.00%	-0.03%	0.00%	-0.09%

	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	6398.31	6940.91	6295.29	6924.23	6600.77
(3.6), (3.7) & (3.8)	-3.08%	-0.42%	-2.27%	0.56%	0.45%
(3.18)	23.95%	38.29%	11.91%	12.19%	18.05%
(3.20)	1.93%	0.47%	4.06%	1.50%	0.60%
(3.17)	2.74%	2.96%	2.61%	2.45%	3.65%
(3.15)	-0.06%	0.00%	0.00%	0.00%	0.00%
(3.19)	0.30%	0.04%	-0.06%	-0.97%	-0.72%

Table C.17: Changes in CPU for Three Vehicles

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.20)	93.90%	0.00%	84.65%	0.00%	0.00%
(3.17)	-146.94%	0.00%	29.68%	0.00%	14.79%
(3.15)	0.25%	0.00%	-1.86%	0.00%	0.15%
(3.19)	-0.50%	0.00%	0.77%	0.00%	0.96%
	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.20)	0.00%	0.00%	51.75%	78.56%	0.00%
(3.17)	0.00%	0.00%	79.86%	55.90%	0.00%
(3.15)	0.00%	0.00%	1.45%	4.13%	0.00%
(3.19)	0.00%	0.00%	-1.86%	-0.95%	0.00%
	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	1800	1800	1800	1800	1800
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%	0.00%
(3.18)	0.00%	39.82%	0.00%	0.00%	0.00%
(3.20)	0.00%	44.56%	0.00%	0.00%	0.00%
(3.17)	0.00%	65.29%	0.00%	0.00%	0.00%
(3.15)	0.00%	-3.22%	0.00%	0.00%	0.00%
(3.19)	0.00%	-8.38%	0.00%	0.00%	0.00%

Table C.18: Changes in Opt Gap for Three Periods and Three Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	-	-	11.38	12.66	-
(3.6), (3.7) & (3.8)	-	-	-	0.00%	-
(3.18)	54.78%	-	-	39.81%	-
(3.20)	-	-	-40.56%	-	-
(3.17)	-	-	-	-	-
(3.15)	-	-	-	-	-
(3.19)	0.00%	0.00%	0.00%	-	-

	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	5.30	4.83	-	5.82	0.73
(3.6), (3.7) & (3.8)	-	-17.18%	-	-201.55%	-32.88%
(3.18)	-	45.76%	95.68%	61.71%	43.30%
(3.20)	-	85.02%	-	45.54%	-89.09%
(3.17)	-	-139.13%	-	73.22%	-
(3.15)	-	0.91%	-	3.06%	-
(3.19)	-	0.00%	-	44.21%	-

	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	51.59	-	-	-	-
(3.6), (3.7) & (3.8)	0.10%	-	-	-	-
(3.18)	-	-	-	-	-
(3.20)	-	-	-	-	-
(3.17)	-	-	-	-	-
(3.15)	-	-	-	-	-
(3.19)	-	-	-	-	-

	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	20.66	8.46	-	9.75	11.50
(3.6). (3.7) & (3.8)	-5.13%	27.30%	-	24.82%	-19.04%
(3.18)	67.96%	-230.41%	-	84.04%	45.14%
(3.20)	92.53%	86.22%	-	-	67.51%
(3.17)	-26.92%	52.50%	-	-	-716.39%
(3.15)	4.55%	19.55%	-	-	0.10%
(3.19)	-1.59%	2.80%	-	-	-0.10%

Table C.19: Changes in Opt Gap for Six Periods and Three Vehicles

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	-	-	-	-	-
(3.6), (3.7) & (3.8)	-	-	-	-	-
(3.18)	-	-	-	-	-
(3.20)	-	-	-	-	-
(3.17)	-	-	-59.65%	-	-
(3.15)	-	0.12%	-	-	-
(3.19)	-	60.19%	-	-	-

	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	-	0.43	-	-	9.13
(3.6), (3.7) & (3.8)	-	-318.60%	-	-	29.46%
(3.18)	-	-	-	69.89%	88.04%
(3.20)	26.06%	-	49.79%	39.29%	46.75%
(3.17)	18.06%	-	32.48%	-20.59%	21.95%
(3.15)	0.00%	-	0.00%	0.00%	0.00%
(3.19)	-23.12%	-	2.53%	22.56%	-3.12%

	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	-	-	-	-	-
(3.6), (3.7) & (3.8)	-	-	-	-	-
(3.18)	-	-	-	82.05%	-
(3.20)	-	-	54.04%	-72.53%	57.31%
(3.17)	11.05%	-	-	-51.02%	67.05%
(3.15)	-	-	-	-130.18%	0.00%
(3.19)	-	-	-	3.44%	-22.37%

Table C.20: Changes in BLB for Three Periods and Three Vehicles

	*high*50-3-*bench*				
Inequalities	1	2	3	4	5
none	14808.2	15192.56	15538.64	16775.5	15970.78
(3.6), (3.7) & (3.8)	2.51%	-0.01%	0.20%	0.00%	-0.50%
(3.18)	1.28%	2.79%	0.70%	1.57%	2.21%
(3.20)	0.33%	0.12%	0.14%	0.10%	-0.02%
(3.17)	1.26%	0.53%	0.71%	3.86%	1.14%
(3.15)	-0.01%	0.31%	-0.21%	-2.92%	-0.03%
(3.19)	0.00%	0.00%	0.00%	-0.02%	0.00%
	*high*50-3-*gen*				
Inequalities	1	2	3	4	5
none	18881.34	16939.23	18001.34	16534.2	17944.28
(3.6), (3.7) & (3.8)	0.33%	-0.04%	-0.38%	-0.03%	-0.24%
(3.18)	0.30%	0.52%	1.54%	1.53%	0.36%
(3.20)	-	1.86%	-	0.72%	-0.11%
(3.17)	-	-0.58%	-	0.45%	-
(3.15)	-	0.01%	-	0.03%	-
(3.19)	-	0.00%	-	0.33%	-
	*low*50-3-*bench*				
Inequalities	1	2	3	4	5
none	4355.61	4708.9	4623.23	4763.39	4369.88
(3.6), (3.7) % (3.8)	-0.25%	-0.02%	2.49%	0.00%	3.11%
(3.18)	12.56%	5.20%	2.84%	3.64%	5.80%
(3.20)	0.22%	2.80%	-0.04%	0.68%	0.80%
(3.17)	2.70%	-1.31%	2.05%	2.66%	2.74%
(3.15)	0.12%	0,00%	0,07%	0.58%	0.52%
(3.19)	0,00%	0.00%	0.00%	0.00%	0.14%
	*low*50-3-*gen*				
Inequalities	1	2	3	4	5
none	4655.61	4138.29	3865.7	4426	4483.4
(3.6), (3.7) & (3.8)	-0.66%	3.37%	0.00%	0.08%	-0.63%
(3.18)	0.96%	-3.55%	7.77%	2.11%	4.60%
(3.20)	4.21%	5.94%	-	-	4.32%
(3.17)	-0.14%	1.51%	-	-	-1.78%
(3.15)	0.03%	0.26%	-	-	0.07%
(3.19)	0.00%	0.03%	-	-	-0.07%

Table C.21: Changes in BLB for Six Periods and Three Vehicles

	*high*30-6-*bench*				
Inequalities	1	2	3	4	5
none	22806.83	19941.65	23285.06	17808.61	18731.9
(3.6), (3.7) & (3.8)	-0.08%	0.85%	0.07%	0.25%	0.58%
(3.18)	6.73%	4.47%	2.51%	3.84%	4.64%
(3.20)	0.00%	0.30%	0.19%	-0.11%	0.36%
(3.17)	1.03%	0.91%	0.40%	1.12%	1.06%
(3.15)	0.01%	0.01%	0.03%	0.01%	0.10%
(3.19)	0.02%	-0.25%	0.00%	-0.09%	0.00%

	*high*30-6-*gen*				
Inequalities	1	2	3	4	5
none	25501.66	26555.08	24660.91	26647.12	25713.69
(3.6), (3.7) & (3.8)	0.21%	-0.52%	0.91%	0.68%	-0.46%
(3.18)	2.41%	-	1.85%	2.54%	2.44%
(3.20)	0.23%	-	0.51%	0.43%	0.30%
(3.17)	0.13%	-	0.27%	0.23%	-0.03%
(3.15)	0.00%	-	0.00%	-0.01%	0.00%
(3.19)	-0.06%	-	0.00%	0.08%	0.00%

	*low*30-6-*bench*				
Inequalities	1	2	3	4	5
none	7884.89	8017.21	8236.29	7928.37	7240.79
(3.6), (3.7) & (3.8)	-1.32%	-8.14%	1.53%	-1.66%	-0.07%
(3.18)	21.28%	18.58%	6.22%	9.01%	13.49%
(3.20)	1.18%	0.06%	0.46%	-0.25%	0.80%
(3.17)	2.74%	3.06%	1.86%	2.82%	3.13%
(3.15)	0.00%	-0.04%	-0.07%	0.00%	0.00%
(3.19)	0.02%	0.00%	-0.01%	-0.52%	0.04%

	*low*30-6-*gen*				
Inequalities	1	2	3	4	5
none	7316.4	7458.05	6917.18	7617.98	6962.97
(3.6), (3.7) & (3.8)	-2.28%	0.93%	-0.21%	-1.03%	0.63%
(3.18)	9.94%	28.90%	3.00%	6.32%	13.87%
(3.20)	2.05%	0.57%	2.87%	0.35%	0.87%
(3.17)	2.23%	2.52%	2.60%	2.59%	3.56%
(3.15)	0.00%	0.10%	0.00%	0.00%	0.00%
(3.19)	-0.10%	0.16%	-0.80%	-0.75%	-0.22%

Appendix D

Average Improvements in the Results with the Inequalities

In Tables D.1-D.3, we report the averages of percentage improvements in the lower bound at the root node (Root LB), total time spent (CPU), optimality gap (Opt Gap) and best lower bound (BLB) found in the time limit while valid inequalities are cumulatively used for different number of vehicles. The numbers in the parenthesis represent the number of instances for which using the corresponding inequality is advantageous and the number instances for which using the corresponding inequality has positive or negative impact. The detailed results for the percentages of the changes in Root LB, CPU, Opt Gap and BLB can be found in Appendix C.

Table D.1: Percentage Improvements in the Results for One Vehicle

	Root LB			
Inequalities	*high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%	0.00%
(3.18)	0.97% (5/5)	0.77% (5/5)	2.45% (5/5)	1.84% (5/5)
(3.20)	0.10% (4/5)	1.55% (5/5)	0.11% (5/5)	4.95% (5/5)
(3.17)	1.00% (4/5)	0.11% (3/5)	2.98% (4/5)	1.66% (4/5)
(3.15), (3.24) & (3.25)	0.02% (1/5)	0.00%	0.05% (3/4)	0.00%
(3.19)	0.00%	0.00%	0.00%	0.00%

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	1.13% (2/3)	0.48% (2/5)	0.21% (3/5)	1.00% (4/5)
(3.18)	1.06% (4/5)	0.46% (5/5)	1.94% (5/5)	1.67% (4/5)
(3.20)	0.59% (5/5)	0.58% (5/5)	1.82% (5/5)	5.29% (5/5)
(3.17)	0.37% (5/5)	0.44% (5/5)	1.20% (5/5)	2.41% (5/5)
(3.15), (3.24) & (3.25)	0.00% (0/2)	0.00% (0/1)	0.01% (1/1)	0.00%
(3.19)	0.02% (1/1)	0.00% (0/1)	0.00%	0.00% (0/1)

	CPU			
Inequalities	*high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	11.28% (3/5)	10.92% (1/1)	26.01% (2/5)	0.00% (0/1)
(3.18)	69.82% (4/5)	25.00% (1/1)	34.27% (4/5)	0.00% (0/1)
(3.20)	22.74% (4/5)	47.71% (3/3)	24.90% (2/5)	66.23% (2/2)
(3.17)	51.94% (3/5)	81.34% (2/4)	60.36% (4/5)	71.75% (2/3)
(3.15), (3.24) & (3.25)	14.24% (3/5)	0.00% (0/4)	20.02% (3/5)	0.35% (1/3)
(3.19)	8.71% (1/5)	2.95% (3/4)	0.32% (1/5)	7.65% (2/3)

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	29.24% (2/4)	25.04% (3/5)	18.85% (1/4)	8.50% (1/4)
(3.18)	0.00% (0/4)	0.00% (0/5)	4.94% (1/3)	0.00% (0/3)
(3.20)	75.97% (5/5)	75.01% (5/5)	69.99% (5/5)	94.53% (5/5)
(3.17)	17.87% (1/5)	30.11% (3/5)	41.76% (3/5)	52.81% (4/5)
(3.15), (3.24) & (3.25)	4.32% (4/5)	0.83% (2/5)	11.77% (1/5)	4.97% (4/5)
(3.19)	1.00% (1/5)	0.00% (0/5)	0.89% (1/5)	0.00% (0/5)

Inequalities	Opt Gap	
	*high*50-3-*gen*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	13.40% (2/3)	17.79% (1/3)
(3.18)	31.75% (1/4)	40.78% (1/4)
(3.20)	87.96% (4/4)	77.22% (4/4)
(3.17)	100.00% (1/2)	50.95% (2/3)
(3.15), (3.24) & (3.25)	0.00%	0.00% (0/1)
(3.19)	7.14% (1/1)	0.00% (0/1)

Inequalities	BLB	
	*high*50-3-*gen*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.02% (3/4)	0.31% (2/3)
(3.18)	0.35% (1/4)	2.20% (1/4)
(3.20)	1.29% (2/2)	5.73% (3/3)
(3.17)	0.00% (0/1)	3.58% (1/2)
(3.15), (3.24) & (3.25)	0.00%	0.00% (0/1)
(3.19)	0.00% (0/2)	0.00% (0/1)

Table D.2: Percentage Improvements in the Results for Two Vehicles

Root LB

Inequalities	*high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.00%	0.12% (1/2)	0.00%	0.00% (0/1)
(3.18)	2.73% (5/5)	1.27% (4/5)	9.53% (5/5)	1.84% (5/5)
(3.20)	0.13% (5/5)	1.57% (5/5)	0.18% (5/5)	5.44% (5/5)
(3.17)	0.55% (4/5)	0.17% (2/5)	2.83% (5/5)	1.73% (4/5)
(3.15)	0.73% (2/3)	0.00%	0.07% (2/3)	0.00%
(3.19)	0.00%	0.00%	0.00% (0/1)	0.00%

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.27% (4/5)	0.22% (3/5)	0.38% (4/5)	2.04% (3/5)
(3.18)	3.50% (5/5)	2.21% (5/5)	9.39% (5/5)	9.11% (5/5)
(3.20)	0.32% (5/5)	0.41% (4/5)	0.91% (5/5)	3.02% (5/5)
(3.17)	0.74% (5/5)	0.26% (5/5)	2.21% (5/5)	2.65% (5/5)
(3.15)	0.00%	0.00% (0/1)	0.00% (0/2)	0.00%
(3.19)	0.02% (3/3)	0.03% (2/3)	0.00% (0/1)	0.04% (3/3)

CPU

Inequalities	*high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.00% (0/1)	20.27% (1/1)	0.00% (0/1)	0.00%
(3.18)	18.03% (1/2)	95.34% (1/1)	0.00% (0/1)	0.00%
(3.20)	30.34% (2/3)	74.36% (2/3)	41.45% (2/2)	0.00%
(3.17)	41.62% (2/2)	33.80% (3/4)	53.32% (1/2)	83.57% (2/2)
(3.15)	40.38% (1/2)	2.44% (2/4)	35.63% (1/2)	0.00% (0/2)
(3.19)	0.30% (2/2)	2.25% (2/4)	0.00% (0/1)	1.34% (1/2)

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.00%	0.00% (0/1)	0.00%	0.00%
(3.18)	0.00%	90.95% (1/1)	0.00%	0.00%
(3.20)	6.84% (1/1)	61.49% (3/3)	0.00%	0.00%
(3.17)	92.42% (1/1)	15.69% (1/3)	91.26% (1/1)	91.24% (1/1)
(3.15)	0.00% (0/1)	0.00% (0/3)	0.00% (0/1)	0.00% (0/1)
(3.19)	0.00% (0/1)	0.00% (0/3)	51.87% (1/1)	62.17% (1/2)

Inequalities	Opt Gap *high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.00% (0/1)	41.85% (3/4)	0.00% (0/2)	0.00% (0/3)
(3.18)	64.93% (2/2)	46.94% (3/3)	63.75% (3/3)	88.03% (1/1)
(3.20)	45.37% (2/2)	87.96% (1/1)	71.11% (2/3)	19.60% (1/1)
(3.17)	29.68% (2/3)	0.00% (0/1)	49.18% (1/3)	63.48% (1/3)
(3.15)	0.00% (0/3)	0.00%	64.02% (2/3)	0.60% (1/2)
(3.19)	11.77% (2/2)	0.00%	3.67% (1/1)	1.20% (1/2)

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.00% (0/2)	74.47% (1/3)	0.00% (0/1)	25.88% (1/3)
(3.18)	49.74% (2/2)	83.33% (3/3)	93.65% (1/1)	74.03% (3/4)
(3.20)	30.14% (2/2)	40.51% (2/2)	54.84% (2/2)	53.38% (5/5)
(3.17)	19.00% (1/3)	50.00% (1/2)	36.15% (1/1)	51.50% (4/4)
(3.15)	0.78% (1/2)	17.78% (1/2)	0.00% (0/2)	0.00% (0/4)
(3.19)	31.28% (1/2)	40.00% (1/2)	0.00% (0/1)	34.68% (1/2)

Inequalities	BLB *high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.06% (1/2)	0.53% (3/4)	0.00% (0/2)	0.78% (2/4)
(3.18)	1.45% (4/4)	0.57% (3/4)	5.02% (4/4)	2.24% (3/5)
(3.20)	0.32% (2/3)	1.52% (2/2)	0.67% (2/3)	6.64% (5/5)
(3.17)	0.33% (2/3)	0.00% (0/1)	1.60% (2/3)	1.33% (2/3)
(3.15)	0.00% (0/2)	0.00%	0.53% (2/3)	0.00% (0/1)
(3.19)	0.00%	0.00%	0.00% (0/1)	0.00% (0/1)

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.19% (3/5)	0.42% (1/4)	1.11% (1/5)	1.31% (4/5)
(3.18)	1.62% (5/5)	1.46% (4/4)	4.87% (5/5)	6.78% (4/5)
(3.20)	0.30% (3/4)	0.31% (2/2)	1.18% (4/5)	1.76% (5/5)
(3.17)	0.53% (4/4)	0.15% (1/2)	1.77% (4/4)	1.36% (4/4)
(3.15)	0.01% (1/3)	0.00% (0/1)	0.00% (0/1)	0.00% (0/2)
(3.19)	0.23% (3/3)	0.10% (1/2)	0.00% (0/3)	0.27% (1/2)

Table D.3: Percentage Improvements in the Results for Three Vehicles

Inequalities	Root LB			
	*high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.00% (0/1)	0.12% (1/2)	0.00%	0.00% (0/1)
(3.18)	4.81% (5/5)	1.78% (5/5)	17.61% (5/5)	4.47% (5/5)
(3.20)	0.08% (4/4)	1.36% (5/5)	0.25% (4/5)	4.87% (5/5)
(3.17)	0.98% (5/5)	0.37% (3/5)	2.93% (5/5)	2.07% (5/5)
(3.15)	0.02% (4/5)	0.00%	0.02% (2/3)	0.00%
(3.19)	0.00%	0.03% (1/1)	0.43% (1/1)	0.00%

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.32% (2/5)	0.42% (4/5)	0.85% (4/5)	0.51% (2/5)
(3.18)	7.49% (5/5)	3.52% (5/5)	20.28% (5/5)	20.88% (5/5)
(3.20)	0.22% (5/5)	0.36% (5/5)	0.69% (5/5)	1.71% (5/5)
(3.17)	0.87% (5/5)	0.36% (5/5)	2.35% (5/5)	2.88% (5/5)
(3.15)	0.18% (2/2)	0.00%	0.06% (1/2)	0.00% (0/1)
(3.19)	0.00% (0/3)	0.02% (2/3)	0.00% (0/3)	0.17% (2/5)

Inequalities	CPU		
	*high*50-3-*gen*	*low*50-3-*gen*	*high*30-6-*gen*
(3.6), (3.7) & (3.8)	0.00%	0.00%	0.00%
(3.18)	0.00%	0.00%	39.82% (1/1)
(3.20)	89.28% (2/2)	65.16% (2/2)	44.56% (1/1)
(3.17)	22.24% (2/3)	67.88% (2/2)	65.29% (1/1)
(3.15)	0.20% (2/3)	2.79% (2/2)	0.00% (0/1)
(3.19)	0.87% (2/3)	0.00% (0/2)	0.00% (0/1)

Inequalities	Opt Gap *high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	0.00%	0.00% (0/3)	0.10% (1/1)	26.06% (2/4)
(3.18)	47.30% (2/2)	61.61% (4/4)	0.00%	65.71% (3/4)
(3.20)	0.00% (0/1)	65.28% (2/3)	0.00%	82.09% (3/3)
(3.17)	0.00%	73.22% (1/2)	0.00%	52.50% (1/3)
(3.15)	0.00%	1.99% (2/2)	0.00%	8.07% (3/3)
(3.19)	0.00%	44.21% (1/1)	0.00%	2.80% (1/3)

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.00%	29.46% (1/2)	0.00%	0.00%
(3.18)	0.00%	78.97% (2/2)	0.00%	82.05% (1/1)
(3.20)	0.00%	40.47% (4/4)	0.00%	55.68% (2/3)
(3.17)	0.00% (0/1)	24.16% (3/4)	0.00%	39.05% (2/3)
(3.15)	0.12% (1/1)	0.00%	0.00%	0.00% (0/1)
(3.19)	60.19% (1/1)	12.55% (2/4)	0.00%	3.44% (1/2)

Inequalities	BLB *high*50-3-*bench*	*high*50-3-*gen*	*low*50-3-*bench*	*low*50-3-*gen*
(3.6), (3.7) & (3.8)	1.36% (2/4)	0.33% (1/5)	2.80% (2/4)	1.73% (2/4)
(3.18)	1.71% (5/5)	0.85% (5/5)	6.01% (5/5)	3.86% (4/5)
(3.20)	0.17% (4/5)	1.29% (2/3)	1.13% (4/5)	4.82% (3/3)
(3.17)	1.50% (5/5)	0.45% (1/2)	2.54% (4/5)	1.51% (1/3)
(3.15)	0.319% (1/5)	0.02% (2/2)	0.32% (4/4)	0.12% (3/3)
(3.19)	0.00% (0/1)	0.33% (1/1)	0.14% (1/1)	0.03% (1/2)

Inequalities	*high*30-6-*bench*	*high*30-6-*gen*	*low*30-6-*bench*	*low*30-6-*gen*
(3.6), (3.7) & (3.8)	0.44% (4/5)	0.60% (3/5)	1.53% (1/5)	0.78% (2/5)
(3.18)	4.44% (5/5)	2.31% (4/4)	13.72% (5/5)	12.41% (5/5)
(3.20)	0.28% (3/4)	0.37% (4/4)	0.63% (4/5)	1.34% (5/5)
(3.17)	0.90% (5/5)	0.21% (3/4)	2.72% (5/5)	2.70% (5/5)
(3.15)	0.03% (5/5)	0.00% (0/1)	0.00% (0/2)	0.10% (1/1)
(3.19)	0.02% (1/3)	0.08% (1/2)	0.03% (2/4)	0.16% (1/5)