A LOT SIZING PROBLEM IN DELIBERATED AND CONTROLLED CO-PRODUCTION SYSTEMS

by

Bahadır Pamuk

B.S., Industrial Engineering, Boğaziçi University, 2016

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Industrial Engineering
Boğaziçi University
2018

ACKNOWLEDGEMENTS

I sincerely thank my supervisors Prof. Caner Taşkın and Prof. Semra Ağralı for their endless support and insightful guidance. I cordially thank to Prof. Özgür Özlük, Assoc. Prof. Ali Tamer Ünal, and Assoc. Prof. Mehmet Önal for their participation in my thesis committee and their precious remarks. I am very grateful to Prof. Ümit Bilge for welcoming me as a member of BUFAIM team.

I thank Banu Kabakulak for her invaluable comments and help, especially in developing the pattern fitting heuristic. Finally, I would like to thank to my colleagues Gökalp Erbeyoğlu and Mert Şahinkoç for their valuable remarks.

Last but not the least, I would like to thank to my family for encouragement and support. I am indebted to you, for making me the person who I am.

This thesis is supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under grant no 116M555. I also want to thank TÜBİTAK for financially supporting me during my graduate studies under the programme BİDEB-2210A.

ABSTRACT

A LOT SIZING PROBLEM IN DELIBERATED AND CONTROLLED CO-PRODUCTION SYSTEMS

Deliberated and controlled co-production can be defined as the production of different products simultaneously where production parameters are known and co-production is deliberate. We study an extension of the lot sizing problem in a deliberated and controlled co-production system, and show that it is NP-Hard. We investigate special cases of the problem for which it is polynomially solvable, and propose solution techniques for those special cases. We propose four mixed integer programming model formulations based on single item uncapacitated lot sizing and simple plant location formulations. We show that solution spaces of the linear relaxations of the proposed formulations are equal. We propose valid inequalities for the problem and show that our proposed valid inequalities added to the model with a separation algorithm improve the linear relaxation lower bound by more than %20 for all test instances. We propose a pattern fitting heuristic that aims to find initial feasible solutions for a commercial solver. We propose another heuristic based on Wagner-Whitin's algorithm to create integer feasible solutions from fractional solutions. We show that the average optimality gap is reduced by at least %10 with proposed improvements to MIP formulations. We also show that the quality of integer feasible solutions is increased within a given time limit.

ÖZET

İSTEMLİ VE KONTROLLÜ BİRLİKTE ÜRETİM SİSTEMLERİNDE ÖBEK BÜYÜKLÜĞÜ BELİRLEME PROBLEMİ

İstemli ve kontrollü birlikte üretim sistemleri, farklı ürünlerin eş zamanlı olarak birlikte üretildiği, üretim parametrelerinin bilindiği ve birlikte üretimin istemli olarak yapıldığı üretim sistemleri olarak tanımlanabilir. İstemli ve kontrollü bir birlikte üretim sisteminde öbek büyüklüğü belirleme probleminin NP-Zor bir problem olduğu gösterildi. Bu problemin polinom zamanlı çözülebilen versiyonları belirtilip polinom zamanlı çözüm yolları önerildi. Tek ürünlü kapasite kısıtsız öbek büyüklüğü belirleme ile basit tesis yerleşimi modellemelerinden yola çıkılarak 4 adet karma tam sayılı programlama modeli geliştirildi. Bu farklı modellerin doğrusal gevşetmelerinin olurlu bölgelerinin aynı olduğu gösterildi. Bu problem için geçerli eşitsizlikler önerilip, bu eşitsizliklerin doğrusal gevşetme alt sınırını bütün testlerde yüzde 20'den daha fazla arttırdığı gösterildi. Olurlu tamsayı çözümler bulmak ve dal ve sınır algoritmasına bir ilk çözüm olarak verebilmek için bir şekil benzetme sezgisel yöntemi geliştirildi. Başka bir sezgisel yöntem de tamsayı olmayan çözümleriden tamsayı çözümler elde etmek için Wagner-Whitin' in algoritmasından yola çıkılarak geliştirildi. Önerilen geliştirmeler yapıldıktan sonra, karma tam sayılı programlama modellerine göre, eniyileme farkının yüzde 10 oranında azaldığı belirlenmiştir ve belirlenen zaman limiti içerisinde bulunan olurlu tamsayı çözümlerin kalitesinde de artış olmuştur.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

c_t^i	Composite variable cost of CPU i in period t, defined as $c_t^i = p_t^i + \sum_{j \in J(i)} \sum_{k=t}^{T} h_k^j \alpha_i^j$
d_t^j	Demand of product j in period t
d_{tk}^j	Cumulative demand of product j between periods t and k, defined as $d_{tk}^j = \sum_{t'=t}^{k} d_{t'}^j$
f_t^i	Fixed cost of producing CPU i in period t
h_t^j	Unit holding cost of product j in period t
I	Set of co-production units, indexed by i
$I(j)$	Set of co-production units, that produce product j, $I(j) \subseteq I$
J	Set of products, indexed by j
$J(i)$	Set of products produced by CPU i, $J(i) \subseteq J$
p_t^i	Variable cost of producing a unit of CPU i in period t
s_t^j	Decision variable, ending inventory of product j in period t
T	Set of time periods in the planning horizon, indexed by t
x_t^i	Decision variable, amount CPU of i produced in period t
X^{DCCP}	Set of feasible solutions for DCCP
X^{LS-U}	Set of feasible solutions for LS-U
y_t^i	Decision variable, equals 1 if CPU i is produced in period t; 0 otherwise
α_i^j	Amount of product j obtained when 1 unit of CPU i is produced
$\Theta_{jtt'}$	Decision variable, amount of product j produced in period t to be consumed in period t'
$\Theta_{tt'}^{ij}$	Decision variable, amount of product j produced using CPU i in period t to be consumed in period t'

LIST OF ACRONYMS/ABBREVIATIONS

B&B	Branch and Bound
CPU	Co-production Unit
CPWW	Co-production Wagner-Whitin Heuristic
DCCP	Deliberated and Controlled Co-production
DLSP	Dynamic Lot Sizing Problem
DP	Dynamic Programming
ELS	Economic Lot Sizing
ELSP	Economic Lot Scheduling Problem
IDP	Interval Division Property
FIFO	First In First Out
LHS	Left Hand Side
LP	Linear Program
LS-U	Uncapacitated Lot Sizing
MIP	Mixed Integer Programming
NP	Nondeterministic Polynomial Time
RHS	Right Hand Side
SWC	Substitution with Conversion
SWO	Substitution without Conversion
ZIP	Zero Inventory Policy

1. INTRODUCTION

Co-production is defined as producing several different products simultaneously in the same production run. This occurs either due to physical or chemical nature of the production system or in order to effectively use scarce resources. Co-produced units may differ only in quality as in semi-conductor production [1, 2], or they may be completely different products, as in float glass production [3, 4]. The key point here is to have sufficient difference in products in the sense that there is a need for the company to differentiate these products.

Co-production can be categorized using two criteria: control and deliberation. Co-production can be controlled or uncontrolled based on the production parameters. If the company has control over the production parameters, such as the number of units co-produced (production rates) for each type of product, then this type of co-production is said to be controlled. On the other hand, co-production can be uncontrolled or too expensive to control due to the structure of the process. For example, in semi-conductor production [1,2] the products will always differ in speed with current production methods due to the randomness of the process which makes the process uncontrolled. On the other hand, producing different products from the same metal sheet in a forming press machine can be classified as controlled since product types and ratios will depend on the die used. If the decision maker has the option to manufacture each product separately or by means of co-production, then co-production is deliberated. For example, co-generating different types of energy (electricity, steam, etc.) in energy industry [5] is deliberated and controlled, whereas semi-conductor production [1, 2] is neither deliberated nor controlled. In this thesis we consider deliberated and controlled co-production systems.

Co-production is inherently present in float glass manufacturing. Continuous flow of glass is cut into different sizes, and different size glasses are considered as different products. In Figure 1.1, we see a robot arm that picks a glass product from production line. In the float glass production system described in [4], glasses are also classified as different quality products based on the number of defects on the glass surface. Figure 1.2 illustrates co-production in float glass manufacturing where existence of random errors on the glass surface necessitates simultaneous production of products having various dimensions and quality levels. There are also production systems in which co-production is optional. For example, with the mould shown in Figure 1.3, two different products can be simultaneously produced using a single mould.

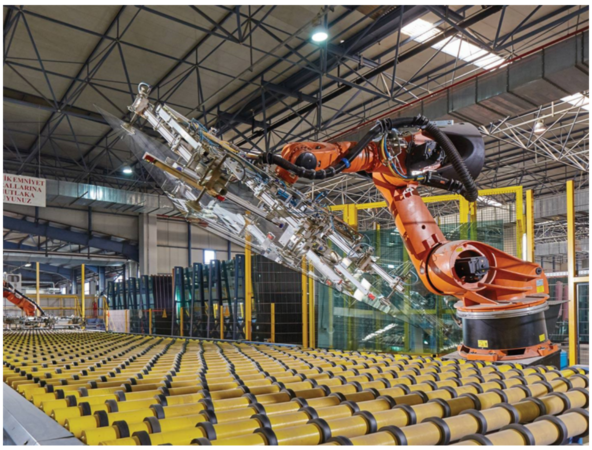

Figure 1.1. Robot Arm Picking Cut Glass in Float Glass Manufacturing.

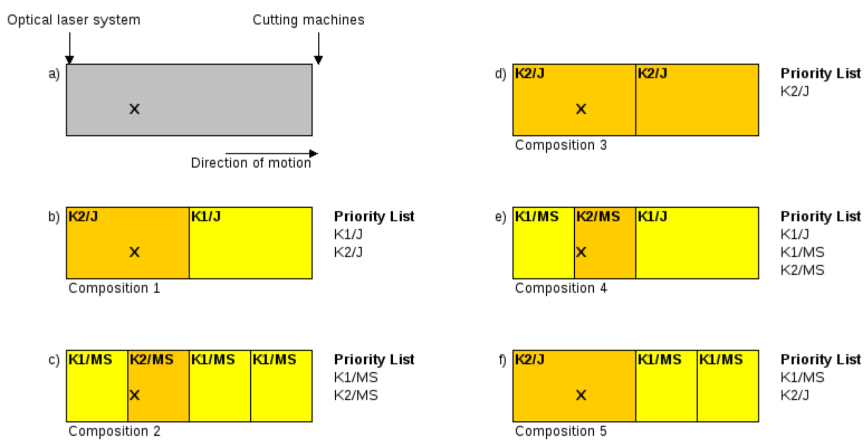

Figure 1.2. Possible Cut Locations for Float Glass Resulting in Different Compositions for a Given Defect Location Marked with x.

The decision of what and when to produce has an enormous importance since having excessive inventories or backlogging products can add up to costs, and it may determine the difference between a profitable company and bankruptcy. This phenomenon is studied under "Lot Sizing Problems" in the literature. While lot sizing problems are studied extensively under many scenarios or extensions, the co-production setting is not well studied. This thesis aims to fill in this gap in the literature.

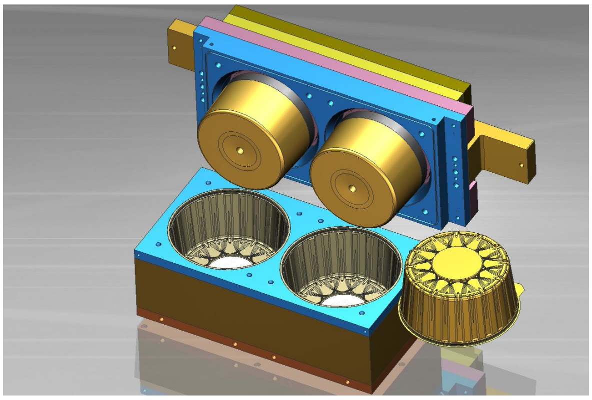

Figure 1.3. Example of a Two Product Mould.

When certain characteristics over some by-products are present, co-production structure can be omitted; and hence, the problem can be solved by traditional lot sizing methods. One such example is given in Section 4.4 where one co-product's demand is significantly lower than other products, and it is automatically satisfied considering the other products. In this case, there is no need to consider co-production explicitly. Another special case is omitting some of the co-products that have relatively low holding cost or low demand. In this case, it might be possible to find optimal or near optimal solutions by traditional lot sizing methods. However, when co-production is internally present in the production system, as in glass or semi conductor production, aforementioned simplifications would not be possible. Therefore, there is a need to specifically study lot sizing in co-production environments.

In this thesis we study a lot sizing problem in a deliberated and controlled co-production environment, where products are non-substitutable and have dynamic deterministic demand over a finite planning horizon. We refer to this problem as Deliberated and Controlled Co-production Problem (DCCP). DCCP can be modeled similarly to the well known Dynamic Lot Sizing Problem (DLSP) [6]. Dynamic programming (DP) techniques are used extensively to solve lot sizing problems [6]. However DP used for DLSP cannot be simply adapted to our problem, as we show in Chapter 3 DCCP is NP-Hard, and does not possess characteristics of regular lot sizing problems such as zero inventory policy (ZIP). We propose different mixed integer programming (MIP) formulations for DCCP, and develop valid inequalities to be added with a separation algorithm for solving the models in a reasonable amount of time.

The remainder of this thesis is as follows: Literature review is given in Chapter 2. We define our problem in Chapter 3. We provide polynomially solvable cases of DCCP in Chapter 4, and propose alternative model formulations in Chapter 5. We propose improvements to our models in Chapter 6. This thesis concludes with experiments in Chapter 7, followed by conclusions and future research directions in Chapter 8.

2. LITERATURE REVIEW

Lot sizing problems are well studied in the literature. Since Wagner and Whitin have published their seminal paper [6], substantial research has been done in the area. Various versions of lot sizing problems have been studied. For an extensive review, we refer the reader to [7] and to its updated version [8].

Discrete models with big time buckets where multiple items of a single product can be produced within a single period, constitute a big portion of lot sizing literature. In these models time is modeled as a finite sequence of discrete time points, and a period is defined as the time interval between consecutive time points. In the simplest case, there is a single product that has time varying demand. The only constraint in that case is having demand satisfied either by production or from the inventory, and backlogging is not allowed. Fixed and variable costs of production and inventory holding costs are also time varying. In this simplest case, so called "dynamic lot sizing problem (DLSP)", there is no capacity constraint. DLSP can be solved in $O(T^2)$ time with Wagner and Whitin's original Dynamic Program (DP) [6], which is improved to $O(Tlog(T))$ by [9–11] independently. In the case of no speculative motives for holding inventory, which requires the variable cost of producing and holding an inventory of a product to be greater than or equal to the variable production cost of that product in the following periods, DLSP is solvable in $O(T)$ [8].

The lot sizing problem has several extensions in the literature. These are primarily based on the length of the planning horizon, number of layers, number and type of products, presence of capacity or resource limitations, demand type, allowance of backlogging, etc. In the case of elastic demands, where demand is a function of the product, pricing decisions are also included in the problem [7].

DP is widely used in solving lot sizing problems. In classical lot sizing problems, "zero inventory policy (ZIP)", which can be defined as the amount of production in a production period must cover exactly the sum of demands until the next production period or last period in the planning horizon, holds. This property allows developing DPs that solve the problem efficiently. There is another property called Interval Division Property (IDP) that holds for some lot sizing problems. IDP is defined in [12] as if there exist n many production periods, then it is possible to divide the planning horizon into n many sets, where each set has consecutive indices, and assign each production period to its corresponding set (first production period to first period set, etc.) exactly satisfying that set's total demand. Inventory deteriorates in time (perishable inventory) and deterioration rates depend on age and the period of production, whereas inventory costs depend on the age of the stock and the period. Backorder is not allowed, and inventory and production costs are non-decreasing concave functions. In this case IDP holds and DP recursion is based on IDP [12].

The convex hull of the classical lot sizing problem DLSP is given by Pochet and Wolsey [13]. They provide facet-defining inequalities and convex hulls of the feasible solutions of DLSP, DLSP with backlogging, DLSP with constant capacity, and DLSP with start-up costs [13].

DLSP with multiple products and one way product substitution, in which substitution of a product is possible with a higher quality product, is studied in [14]. Multiple products can be seen as different quality levels of the same products in this context. Two versions of this problem are studied. In the first one, which is called substitution with conversion (SWC), the higher level product is converted to a lower level before the substitution. In the second one, called substitution without conversion (SWO), the substitution is done without a conversion operation. Both versions are shown to be NP-Hard in the strong sense. Then, the equivalence to a minimum concave-cost network flow problem is shown, and a DP is proposed for both SWC and SWO. Chen and Thizy [15], study multiple item capacitated lot sizing problem with no backorders, and prove that the problem is strongly NP-Hard when capacities are not constant.

Another extension of lot sizing problem is the coordinated lot sizing problems that include product families. In coordinated lot sizing problems a family of products has a shared fixed setup cost, which is to be paid whenever one or more products of a family is produced [16]. A minor setup cost also exists for individual products inside a product family. Variable costs of production is similar to that of DLSP. [17] proposes a Lagrangian heuristic for capacitated version of the same problem. Having product families does not capture the notion of co-production. Despite having a shared fixed cost for production families, products inside the same family is not necessarily produced simultaneously.

Bitran and Gilbert [1] and Bitran and Dasu [2] focus on co-production with random yields in semiconductor production. In their context, it is possible to substitute a lower tier product with a higher tier one. This is called serially nested co-production. The problem is divided into two sub-problems as “morning problem”, in which production decisions are given, and “afternoon problem”, in which products are allocated to customers after yields are known. In Bitran and Dasu [2], the objective is to maximize the expected profit whereas in Bitran and Gilbert [1] it is the minimization of expected cost comprised of production, inventory holding, and shortage costs. Latter also studies impacts of alternative downgrading policies.

Öner and Bilgiç [3] study an uncontrolled co-production system in float glass manufacturing with constant holding cost rate and fixed sequence independent setup costs where substitution of products is not allowed. They develop a continuous economic lot scheduling model (ELSP) to find a common cycle schedule. Co-products are not only different in the quality aspect but also in the size. Taşkın and Ünal [4] also study a co-production system in float glass manufacturing focusing on tactical level planning. They develop two mathematical models to be solved consecutively for colored and clear glass for a glass manufacturer.

Tomlin and Wang [18] consider pricing decisions together with co-production using stochastic demand model of utility maximizing customers in a single period.

There are two products and two customer classes. Cost functions are linear in demand although findings are applicable when cost is convex in demand. The substitution of lower tier products with higher ones are allowed.

Vidal Carreras *et. al.* [19] study a deliberated and controlled co-production system with non substitutable demand. Similar to [3], their model is a continuous ELSP with an aim to find a common cycle time. Costs are constant, fixed and sequence independent. Only two products are considered.

Rafiei *et. al.* [20] consider a co-production system with sequence dependent setup times and demand uncertainty. There are production families, and recipes in the same production family require no changeover cost. It is a case study on demand driven wood re-manufacturing mills. They propose a three step methodology to solve wood re-manufacturing industrial problem.

Co-production is also studied within the context of chemical sciences and sustainability. In [21], waste is treated as a co-product, which is also an input to the system, to achieve zero waste. A case study on co-production of decarbonized synfuels and electricity is studied in [22]. Another example of co-production in chemical sciences is [5], which studies co-production of dimethyl ether and electricity.

Ağralı's study [23] is a starting point of this thesis. In that work there is only a single set of products to be produced simultaneously, which makes the co-production controlled but not deliberated. The decision to be made is to when and how much to produce that specific set of products. It is shown that ZIP holds for at least one of the products of the set, and a DP recursion is given that can be solved in polynomial time. In contrast, in this thesis where deliberated co-production exist, the decision maker has the option of to co-produce or not to co-produce.

To the best of our knowledge, there is no research that considers deliberated and controlled co-production in the dynamic deterministic lot sizing literature. Our contribution to the lot sizing literature can be summarized as follows: (i) our problem includes the decision of co-produce or not if the production system in context allows it; and (ii) we consider a co-production setting in which the decision maker can choose from multiple ways of co-producing a single product. From practicality point of view, our problem is expected to be found at production systems in which discrete products are produced, preferably small parts, where multiple products can be produced simultaneously by fitting multiple products into a single manufacturing resource.

3. PROBLEM DEFINITION

We consider a production system in which several products have dynamic deterministic demand over a planning horizon. We call a possible combination of products that can be co-produced together with individual production ratios as a "co-production unit (CPU)". Our system has more than one CPU, and production decisions over the planning horizon are given per CPU rather than per product. The costs incurred are fixed and variable costs of production of CPUs, and the inventory holding costs of products. All cost data, demand information, and production ratios for each CPU are deterministic and known; hence, the system is controlled. It is also possible to produce a product individually by having a single product in a CPU; therefore, our production system is deliberated. Like demand, all costs are also dynamic and time dependent. Initial inventory levels are assumed to be zero. If initial and final inventory levels are given and there are lower bounds for inventory levels, demand data can be modified accordingly in order to get initial and final inventory levels and restrictions on inventory lower bound to zero. The objective is to find a production plan with minimum possible cost, consisting of fixed and variable costs of co-production and inventory holding cost of products, that satisfies all demand in time without backlogging.

Like in DLSP, time is modeled as a finite sequence of discrete time points, which are indexed as $t \in T$. Products are indexed by $j \in J$, which have dynamic deterministic demand, d_t^j, and unit holding cost, h_t^j. There are finitely many CPUs indexed by $i \in I$, and each produce a finite set of predefined products $J(i)$. When one CPU of type i is to be produced, all the products inside set $J(i)$ are co-produced with certain production ratios, α_i^j. Each CPU has a dynamic deterministic fixed cost, f_t^i, and variable cost, c_t^i.

There may be several ways to produce one product, i.e., multiple CPUs may produce the same product with non-identical production ratios and production costs. It may also be possible for a CPU to include only one product, which allows flexibility in the deliberated co-production. As an example, consider a production system given in Figure 3.1 where three products are produced from a metal sheet with six CPUs.

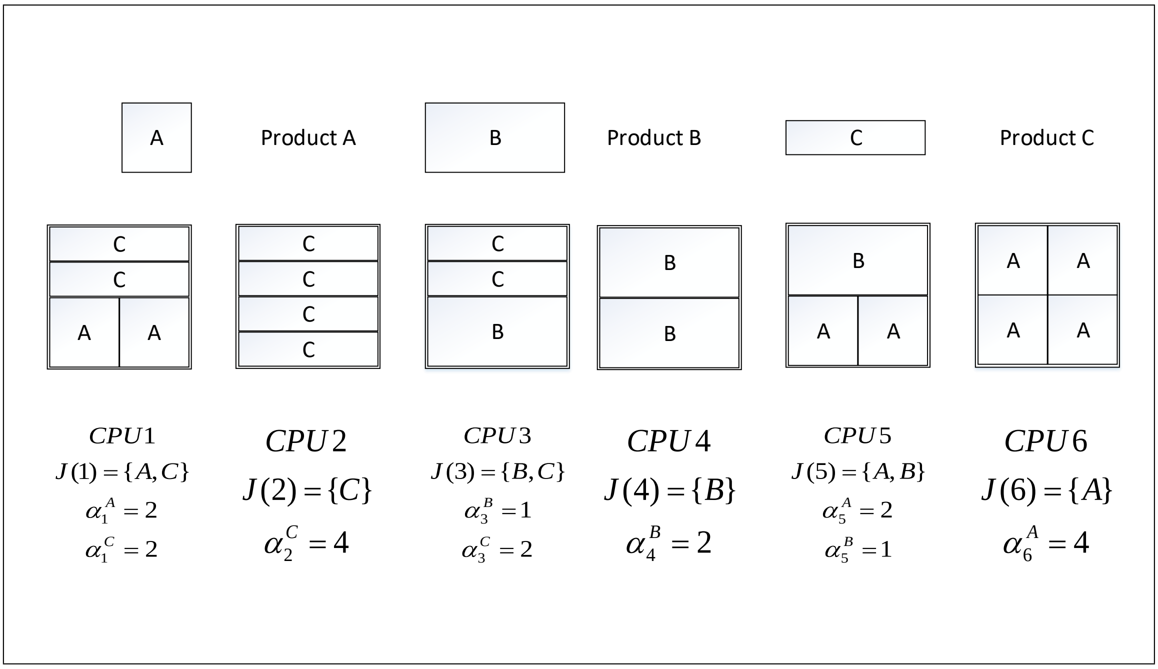

Figure 3.1. Example of Co-production Units.

There are three products; A, B, and C in the example problem given in Figure 3.1 with certain demands. The demand of the products should be satisfied by making production decisions on six possible CPUs over the planning horizon. CPU1, CPU3 and CPU5 are composed of more than one product in contrast to CPU2, CPU4 and CPU6. As an example, consider CPU1: it is composed of two units of product A and two units of product C, therefore $\alpha_1^A = \alpha_1^C = 2$. CPU1 does not produce product B implying $\alpha_1^B = 0$. When a decision of x amount of production is made for CPU1 in a production period t, $2x$ amount of product A and $2x$ amount of product C is produced in that period.

3.1. Computational Complexity

We analyze the computational complexity of the problem that we consider in this section. In its simplest form, without capacities or backlogging, we show that decision version of Deliberated and Controlled Co-production (DCCP) is NP-Complete. This is due to its close relationship with the set covering problem. Informally it can be explained as follows: We have finite amount of CPUs $i \in I$, and each produce a subset of products $J_i \subseteq J$ with respect to production ratios α_i^j. J_i can be defined for each i as $\{j' \in J_i\ ,\ j' : \alpha_i^{j'} > 0\}$. When we restrict α_i^j to be either 0 or 1, each CPU i represent a fixed subset of products J_i, and all $j \in J_i$ needs to be co-produced by one unit each whenever a unit of CPU i is produced, accruing costs f_t^i and c_t^i. We further restrict our planning horizon to only one period ($|T| = 1$), fixed costs of production to 1 ($f_1^i = 1$), and variable costs to 0 ($c_1^i = 0$), for all $i \in I$. Since $|T| = 1$, holding costs (h_t^j) are irrelevant. Finally, we further reduce all product demands to 1 ($d_1^j = 1 \ \ \forall j \in J$). With aforementioned parameter settings, our problem reduces to simply selecting fewest number of subsets $J_i \subseteq J$ that collectively produce each product at least once. This problem reduces to the well known minimum (set) cover problem, which is known to be NP-Complete [24].

Proposition 3.1. *Deliberated and Controlled Co-production (DCCP) is NP-Complete.*

Proof. DCCP decision problem can be formulated as follows:

Deliberated and Controlled Co-production

Instance: Finite sets, J of "products", I of "co-production units", and T of "time periods", a positive number K. Production ratios: $\alpha_i^j \in N, \quad \forall j \in J$ and $\forall i \in I$. Fixed costs: $f_t^i \in N, \quad \forall t \in T$ and $\forall i \in I$. Variable costs: $p_t^i \in N, \quad \forall t \in T$ and $\forall i \in I$. Demands: $d_t^j \in N, \quad \forall t \in T$ and $\forall j \in J$. Holding Costs: $h_t^j \in N, \quad \forall t \in T$ and $\forall j \in J$.

Question: Is there a feasible production plan (i.e., all demands are satisfied), having total cost no more than K? Mathematically, this question is if

$$\sum_{t \in T} \Big(\sum_{i \in I} (f_t^i y_t^i + p_t^i x_t^i) + \sum_{j \in J} h_t^j s_t^j \Big) \leq K,$$

where x_t^i is the amount of production of CPU i in period t, $y_t^i = 1$ if $x_t^i > 0$, and $y_t^i = 0$, otherwise; and s_t^j is the amount of inventory of product j in period t.

As the first part of the proof, we need to show that our problem lies in NP. Given an instance of sets I, J, T, data α_i^j, f_t^i, p_t^i, d_t^j, h_t^j, a positive number K, and a "guess" x_t^i; we need to show if we can validate a "yes guess" in polynomial time. There are two things we should be sure of; "the guess" should be feasible, and it does not exceed the cost limit K. In order to show that the first part is doable, we propose the algorithm given in Figure 3.2.

Set $y_t^i = 1$ if $x_t^i > 0$ and $y_t^i = 0$ otherwise. Doable in $\mathrm{O}(|T||I|)$.
Set $s_t^j = \sum_{k \leq t} \sum_{i \in I} (x_k^i \alpha_i^j - d_k^j)$ and if all $s_t^j \geq 0$ then the "guess" is feasible. Doable in $\mathrm{O}(|T||J|)$.
if $\sum_{t \in T} \left(\sum_{i \in I} (f_t^i y_t^i + c_t^i x_t^i) + \sum_{j \in J} h_t^j s_t^j \right) \leq K$ **then**
  The "guess" is valid. Doable in $\mathrm{O}(|T|(|J| + |I|))$.
end if

Figure 3.2. Guess Validation Algorithm.

A reasonable encoding scheme, having size bounded polynomially by set sizes of I, J, and T, can be found for the problem easily. Algorithm in Figure 3.2 can prove if a yes guess is in fact a yes instance in polynomial time. Therefore DCCP is in NP. As the second part of the proof, we will show that DCCP contains a known NP-Complete problem as a special case. Consider *Minimum Cover* problem as given in [25] for that purpose:

Minimum (Set) Cover
Instance: Collection C of subsets of a finite set S, positive integer $K \leq |C|$.
Question: Does C contain a cover for S of size K or less, i.e, a subset $C' \subseteq C$ with $|C'| \leq K$ such that every element of S belongs to at least one member of C'?

In fact we can restrict DCCP to *Minimum Cover* by allowing instances having $|T| = 1$, $\alpha_i^j \in \{0, 1\}$, $f_1^i = 1$, $c_1^i = 0$, $d_1^j = 1$ $\forall j \in J$ and $\forall i \in I$. The transformation of any *Minimum Cover* instance to DCCP is as follows: Consider a *Minimum Cover* instance. Set S corresponds to set J in DCCP instance. Let every subset in set C be indexed by i such that $C = \{C_i\}$. The creation of set $i \in I$ in DCCP is complete. Let

$|T| = 1,\ f_1^i = 1,\ c_1^i = 0,\ d_1^j = 1 \quad \forall j \in J, \forall i \in I$. Now for each $i \in I$ and $j \in J$ define α_i^j as in equation (3.1), and define subsets $J_i \in J$ for each $i \in I$ as $J_i = \{j | j : \alpha_i^j = 1\}$.

$$\alpha_i^j = \begin{cases} 1 & , \quad j \in C_i; \\ 0 & , \quad j \notin C_i. \end{cases} \tag{3.1}$$

Selecting a subset $C' \subseteq C$ in *Minimum Cover* will correspond to selecting $I' \subseteq I$ CPUs to produce in one period DCCP. We now write the total cost of restricted DCCP. Note that $|T| = 1$:

$$\sum_{t \in T} (\sum_{i \in I} (1y_t^i + 0x_t^i) + \sum_{j \in J} 0s_t^j) = \sum_{i \in I} y_1^i \tag{3.2}$$

Assume that an arbitrary *Minimum Cover* instance is a yes instance. There exists a subset $C' \subseteq C$ such that $|C'| \leq K$ and $\bigcup C' = S$. We can write Equation (3.3) since selecting $C' \subseteq C$ in *Minimum Cover* corresponds to selecting $I' \subseteq I$ in DCCP:

$$\sum_{i \in I} y_t^i = |I'| = |C'| \leq K. \tag{3.3}$$

Selecting $\alpha_i^j \in \{0, 1\}$ and corresponding subsets J_i as described previously will ensure the following since $\bigcup C' = S$;

$$\bigcup_{i \in I'} J_i = J \tag{3.4}$$

which means every product $j \in J$ will be produced at least once by selecting $I' \subseteq I$ for production. Since $d_1^j = 1$ for all $j \in J$, the production plan feasibility is guaranteed by (3.4). By (3.3), corresponding DCCP instance is valid if and only if *Minimum Cover* instance is valid, and by (3.4), DCCP instance is feasible if and only if *Minimum Cover* instance is feasible. With aforementioned transformation, any instance of *Minimum*

Cover, can be transformed to a specific instance of DCCP. Transformation having polynomial time complexity easily follows. □

3.2. Mathematical Model (IP1)

A solution of DCCP will provide the amount of CPUs produced in each period. Let the decision variable x_t^i denote the amount of production of CPU i in period t, and let binary variable y_t^i take value of 1 if a production of CPU i takes place in period t. The last set of decision variables s_t^j denotes the ending inventory level of product j in period t. Then, we can give the mixed-integer programming (MIP) formulation for our problem as follows:

$$\textbf{IP1:}\ \text{minimize} \quad \sum_{t\in T}\left(\sum_{i\in I}\left(f_t^i y_t^i + p_t^i x_t^i\right) + \sum_{j\in J} h_t^j s_t^j\right) \tag{3.5}$$

$$\text{subject to} \quad s_{t-1}^j + \sum_{i\in I}\alpha_i^j x_t^i - s_t^j = d_t^j, \qquad \forall t\in T, j\in J \tag{3.6}$$

$$x_t^i \le \max_{j\in J(i)}\left\{\frac{d_{tT}}{\alpha_i^j}\right\} y_t^i, \qquad \forall t\in T, i\in I \tag{3.7}$$

$$x_t^i \ge 0, \qquad \forall t\in T, i\in I \tag{3.8}$$

$$s_t^j \ge 0, \qquad \forall t\in T, j\in J \tag{3.9}$$

$$y_t^i \in \{0,1\}, \qquad \forall t\in T, i\in I. \tag{3.10}$$

The objective function (3.5) consists of fixed and variable costs arising from production of CPUs, and the holding cost of products summed over the planning horizon. Constraint set (3.6) includes inventory flow balance constraints. Constraint set (3.7) forces binary production variables y_t^j to take the value of 1 whenever x_t^is take positive values. The term $\max_{j\in J(i)}\{\frac{d_{tT}}{\alpha_i^j}\}$ is an upper bound for the x_t^i variables, and acts like a big-M value. Constraints (3.8) – (3.10) are non-negativity and binary constraints.

4. POLYNOMIALLY SOLVABLE CASES

The decision version of DCCP is proven to be NP-Complete (see section 3.1). This means that the time required to solve problems increase exponentially with the problem size, and it is not possible to find optimum solutions for practical size instances in a reasonable amount of time unless $P = NP$. In order to understand characteristics of the problem, we analyze some special cases of the problem, and propose polynomial time solution techniques for those special cases in this section.

4.1. No Fixed Cost Requirement

When the fixed cost of a production is negligible or no setup is needed for production, fixed costs of CPUs can be neglected. Without the fixed cost, the MIP of the problem reduces to a linear program since binary variables y_t^i are no longer needed. LP model when $f_t^i = 0$ is given in (4.1) – (4.4).

$$\text{minimize} \quad \sum_{t \in T} \left(\sum_{i \in I} c_t^i x_t^i + \sum_{j \in J} h_t^j s_t^j \right) \tag{4.1}$$

$$\text{subject to} \quad s_{t-1}^j + \sum_{i \in I} \alpha_i^j x_t^i - s_t^j = d_t^j, \qquad \forall t \in T, j \in J \tag{4.2}$$

$$x_t^i \geq 0, \qquad \forall t \in T, i \in I \tag{4.3}$$

$$s_t^j \geq 0, \qquad \forall t \in T, j \in J. \tag{4.4}$$

Since LPs can be solved in polynomial time using interior point algorithms, this special case is polynomially solvable.

4.2. Mutually Exclusive Co-production Units

In some production environments, products may form family structures, and may be possible to have mutually exclusive product families. This results in having mutually exclusive CPUs (i.e, $J_k \cap J_l = \emptyset \quad \forall k, l \in I, k \neq l$). In this structure, DCCP is separable by CPUs. For this case the polynomial time DP algorithm suggested by [23], in which this problem is given under by-production case, can be applied for each CPU. Let $G(t)$ be the cost of an optimal solution to the instance of dynamic uncapacitated lot sizing problem of co-products within a single CPU with a planning horizon consisting of periods from t to T; $t = 1, ..., T$. Recurrence relation of the DP to solve for each CPU i separately can be adapted from [23] as:

$$G_i(t) = \begin{cases} \min\limits_{t<l\leq T+1} \left\{ f_t^i + \left(c_t^i + \sum_{j\in J(i)} \sum_{s=t}^T \alpha^k h_s^j \right) \bar{D}_{t,l-1} + G_i(l) \right\} & \text{, if} \max\limits_{k=1,\ldots,K} d_t^k > 0; \\ \min \left[G_i(t+1), \min\limits_{t<l\leq T+1} \left\{ f_t^i + \left(c_t^i + \sum_{j\in J(i)} \sum_{s=t}^T \alpha^k h_s^j \right) \bar{D}_{t,l-1} + G_i(l) \right\} \right] & \text{, if} \max\limits_{k=1,\ldots,K} d_t^k = 0; \end{cases} \tag{4.5}$$

where

$$\bar{D}_{t,l-1} = \max \left[0, \max_{j\in J(i)} \left\{ \frac{d_{t,l-1}^j}{\alpha_i^j} - s_{t-1}^j \right\} \right], \tag{4.6}$$

$$s_{t-1}^j = D_{1,t-1}\alpha_i^j - d_{1,t-1}^j, \tag{4.7}$$

$$D_{1,t-1} = \max_{j\in J(i)} \left\{ \frac{d_{1,t-1}^j}{\alpha_i^j} \right\}. \tag{4.8}$$

As stated in [9], a straightforward application of this recursion leads to an $O(T^2)$ algorithm; however, it can be further improved to $O(TlogT)$ using techniques in [9] as stated by [23]. By solving a polynomial time DP for each CPU $i \in I$, this special case of the problem can be solved in polynomial time using the algorithm given in Figure 4.1.

for CPU $i \in I$ **do**

Solve a DP for CPU i using the recursion relation in Equation 4.5. The solution is the optimal production plan for CPU i.

end for

Figure 4.1. Algorithm for Mutually Exclusive Co-production Units.

4.3. Two Products per Co-production Unit

When a DCCP has no more than two products per CPU, $|J_i| \leq 2$, a single planning period, $|T| = 1$, no variable costs for CPUs, $c_1^i = 0$, identical fixed costs for CPUs, $f_1^i = f_1^{i'}$, unit demand for all its products, $d_1^j = 1$, and maximum production ratio of 1 amongst its products, $\alpha_i^j \leq 1$, then the DCPP is polynomially solvable. Note that DCCP can be restricted to a *Minimum Cover* problem by allowing instances having $|T| = 1$, $\alpha_i^j \in \{0, 1\}$, $f_1^i = 1$, $c_1^i = 0$, $d_1^j = 1 \quad \forall j \in J$ and $\forall i \in I$. The transformation of any *Minimum Cover* problem instance to DCCP instance is explained in Section 3.1.

Minimum Cover with $c \in C$ and $|c| \leq 2$ can be solved in polynomial time by matching techniques [25]. Therefore any DCCP having $|T| = 1$, $\alpha_i^j \in \{0, 1\}$, $d_1^j = 1$, $f_1^i = 1$, $c_1^i = 0$, and $|J_i| \leq 2$, should also be solved polynomially. Consider an instance with $|J| = 4, |I| = 4$, $J_1 = \{1, 2\}$, $J_2 = \{2, 3\}$, $J_3 = \{3\}$, and $J_4 = \{4\}$. Network representation of the example is given in Figure 4.3, where products are represented with nodes and CPUs are represented with arcs. Algorithm given in Figure 4.2 can be used to solve this special case.

(i) Eliminate all $i' \in I$, $|J_{i'}| = 1$ from set I if $J_{i'}$ is included in at least one $J_{i''}$, $|J_{i''}| = 2$.
(ii) Select all $i' \in I$ that cannot be deleted by step (i). Note that with first two steps we can reduce any problem with $|J_i| \leq 2$ to a problem with $|J_i| = 2$.
(iii) Solve maximum matching problem having set J as vertices and set I as edges.
(iv) If matching from (iii) is a perfect matching, it is the optimal solution to set covering problem. Else, add one edge for covering each uncovered vertex. Since matching algorithm and post operations are all polynomial, the whole operation is polynomial.

Figure 4.2. Algorithm for Special Case: Two Products per Co-production Unit.

The application of the algorithm given in Figure 4.2 for the example instance given in Figure 4.3 is as follows: At step (i), I_3 is eliminated since $J_2 = \{1, 3\}$. At step (ii) I_4 is selected since product 4 does not exist in any other CPU. At step (iii), maximum cardinality matching algorithm is solved. The solution would be one of the two arcs selected. At step (iv), for any uncovered vertex, one edge is selected. This preserves optimality since if selecting an arc would cover two vertices instead of one, maximum matching algorithm would have selected it in the first place.

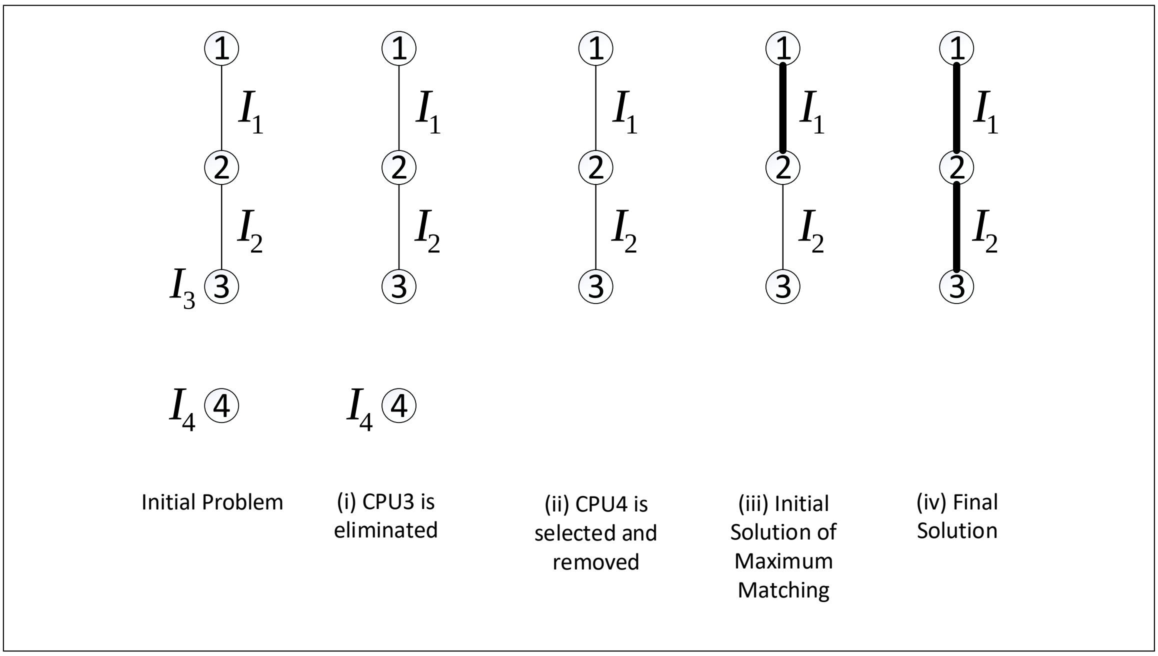

Figure 4.3. Application of Proposed Algorithm for Two Products per Co-production Unit on an Example.

4.4. Main Product and By-Products

Consider a DCCP setting in which a common product is being produced in all CPUs. When demand data satisfies a specific condition (4.9), then it is possible to automatically satisfy the main product's demand by considering only by-products.

$$\sum_{m=1}^{t} d_m^k \leq \sum_{b \in J-\{k\}} \sum_{m=1}^{t} d_m^b \qquad \forall t \in T. \tag{4.9}$$

In this special case, one DLSP is solved for each by-product, and the corresponding production amounts for by-products are optimal production quantities of corresponding CPUs in DCCP setting.

Let the common product $k \in J$ be called a "main product" satisfying $k \in J(i), \forall i \in I$. In other words, a main product is a product that is produced by all CPUs. Let us further simplify this special case by allowing only $|J_i| = 2$, and $\alpha_i^j \in \{0, 1\}$. In this case, let $b \in J - \{k\}$ be by-products. Furthermore, assume that demand data satisfy the inequality (4.9).

Then, the demand for main product k should be automatically satisfied when all by-products' demands are satisfied. Note that $|I| = |J| - 1$. Let the last product in set J be the main product k. The fixed and variable costs defined for each CPU can also be defined with respect to by-products $b \in J - \{k\}$:

$$p_t^j = p_t^i, \qquad \forall j \in J - \{k\}; \tag{4.10}$$

$$f_t^j = f_t^i, \qquad \forall j \in J - \{k\}. \tag{4.11}$$

Note that ZIP holds for all $j \in J - \{k\}$. We give the objective function of this special case as:

$$\text{minimize} \qquad \sum_{t \in T} \left(\sum_{j \in J - \{k\}} \left(f_t^j y_t^j + p_t^j x_t^j + h_t^j s_t^j \right) + h_t^k s_t^k \right). \tag{4.12}$$

Inventory variables s_t^j can be written in terms of production variables x_t^j as:

$$s_t^j = \sum_{l=1}^{t} (x_l^j - d_l^j), \qquad \forall j \in J - \{k\}, \forall t \in T; \tag{4.13}$$

$$s_t^k = \sum_{j \in J - \{k\}} \left(\sum_{l=1}^{t} x_l^j - \sum_{l=1}^{t} d_l^k \right), \qquad \forall t \in T. \tag{4.14}$$

The mathematical model without inventory variables is:

$$\text{minimize} \quad \sum_{t\in T}\left(\sum_{j\in J-\{k\}}\left(f_t^j y_t^j + c_t^j x_t^j\right) - \sum_{j\in J}\sum_{l=1}^{t} h_l^j d_l^j\right) \tag{4.15}$$

$$\text{subject to} \quad x_t^j \le d_{tT}^j y_t^i, \qquad \forall t\in T, j\in J-\{k\}; \tag{4.16}$$

$$\sum_{t=1}^{T} x_t^j = d_{1T}^j, \qquad \forall j\in J-\{k\}; \tag{4.17}$$

$$\sum_{l=1}^{t} x_l^j \ge d_{1t}^j, \qquad \forall t\in T, j\in J-\{k\}; \tag{4.18}$$

$$x_t^j \ge 0, \qquad \forall t\in T, j\in J-\{k\}; \tag{4.19}$$

$$y_t^j \in \{0,1\}, \qquad \forall t\in T, j\in J-\{k\}; \tag{4.20}$$

where

$$c_t^j = p_t^j + \sum_{l=t}^{T}(h_l^j + h_l^k). \tag{4.21}$$

Note that the constant term $\sum_{t\in T}\sum_{j\in J}\sum_{l=1}^{t} h_l^j d_l^j$ can be omitted in the objective function. As it can be seen from the model, this special case is separable by by-products $j \in J-\{k\}$. Then, it is possible to solve this problem by solving a DLSP for each by-product b. This problem can be solved by the recursion in Equation (4.5) using c_t^j in Equation (4.21).

5. ALTERNATIVE MIP FORMULATIONS

In this chapter we develop alternative mixed integer programming formulations for DCCP. First, we reduce the number of variables of IP1 by representing inventory variables, s_t^j, in terms of production variables, x_i^t, and demand, d_j^t. We call the resulting formulation as IP2. Then, we propose more formulations, ELS1 and ELS2, based on simple plant location formulation of single item uncapacitated lot sizing (LS-U). We give the details of all formulations and show that they are equivalent in the sense that the feasible regions of their linear relaxations are equal.

5.1. Inventory Variable Free Formulation (IP2)

Inventory variables, s_t^j, can be written in terms of production variables and demand in Equation (5.1), where $I(j)$ is the set of co-production units that produce product j.

$$s_t^j = \sum_{k=1}^{t} \big(\sum_{i \in I(j)} x_k^i \alpha_i^j - d_k^j \big) \tag{5.1}$$

Then, the objective function can be given as:

$$\sum_{t \in T} \left(\sum_{i \in I} (f_t^i y_t^i + p_t^i x_t^i) + \sum_{j \in J} h_t^j \sum_{k=1}^{t} \big(\sum_{i \in I(j)} x_k^i \alpha_i^j - d_k^j \big) \right). \tag{5.2}$$

Sets I and $I(j)$ in Equation (5.2) can be merged since $\alpha_i^j = 0$ for $i \notin I(j)$, and the constant term can be separated as shown in Equation (5.3):

$$\sum_{t \in T} \sum_{i \in I} \left(f_t^i y_t^i + p_t^i x_t^i + \sum_{j \in J} \sum_{k=1}^{t} h_t^j x_k^i \alpha_i^j \right) - \sum_{t \in T} \sum_{j \in J} \sum_{k=1}^{t} h_t^j d_k^j. \tag{5.3}$$

Let us replace J with $J(i)$ since $\alpha_i^j = 0$ for $j \notin J(i)$:

$$\sum_{t\in T}\sum_{i\in I}\left(f_t^i y_t^i + p_t^i x_t^i + \sum_{j\in J(i)}\sum_{k=1}^{t} h_t^j x_k^i \alpha_i^j\right) - \sum_{t\in T}\sum_{j\in J}\sum_{k=1}^{t} h_t^j d_k^j \tag{5.4}$$

which then reduces to Equation (5.5), when x_t^i and x_k^i terms are merged together:

$$\sum_{t\in T}\sum_{i\in I}\left(f_t^i y_t^i + c_t^i x_t^i\right) - \sum_{t\in T}\sum_{j\in J}\sum_{k=1}^{t} h_t^j d_k^j \tag{5.5}$$

where,

$$c_t^i = p_t^i + \sum_{j\in J(i)}\sum_{k=t}^{T} h_k^j \alpha_i^j. \tag{5.6}$$

Finally, resulting formulation IP2 is given in Equations (5.7) – (5.11) where d_{tk}^j is the cumulative demand of product j between periods t and k.

$$\textbf{IP2:}\ \text{minimize} \quad \sum_{t\in T}\sum_{i\in I}\left(f_t^i y_t^i + c_t^i x_t^i\right) - \sum_{t\in T}\sum_{j\in J}\sum_{k=1}^{t} h_t^j d_k^j \tag{5.7}$$

$$\text{subject to} \quad \sum_{k=1}^{t}\sum_{i\in I(j)} \alpha_i^j x_k^i \geq d_{1t}^j, \qquad \forall t \in T, j \in J \tag{5.8}$$

$$x_t^i \leq \max_{j\in J(i)}\left\{\frac{d_{tT}}{\alpha_i^j}\right\} y_t^i, \qquad \forall t \in T, i \in I \tag{5.9}$$

$$x_t^i \geq 0, \qquad \forall t \in T, i \in I \tag{5.10}$$

$$y_t^i \in \{0,1\}, \qquad \forall t \in T, i \in I. \tag{5.11}$$

We note that IP2 formulation may improve solution times since it has fewer number of variables compared to IP1 formulation. However, the constraint matrix is denser than IP1, which may create computational difficulties.

5.2. Plant Location Formulation 1 (ELS1)

The simple plant location formulation of LS-U is given in [9], and it is shown by [26] that this formulation gives the convex hull of LS-U. We develop ELS1 formulation based on the simple plant location formulation of LS-U, in which production variables are disaggregated in terms of periods where produced items are consumed by demand. Let $\Theta_{jtt'}$ continuous variables represent the production amount of product $j \in J$, that is produced in period $t \in T$ to be consumed in period $t' \in T, t' \geq t$. However, getting rid of x_t^i variables does not appear to be possible, due to having production costs depend on the amount of CPUs produced, not products. Therefore, constraints (5.14) are needed to relate $\Theta_{jtt'}$ variables to x_t^i variables, and the demand satisfaction constraint is revised as in Equation (5.13). Other constraints and the objective function remain the same as that of IP2:

$$\textbf{ELS1:} \text{ minimize } \quad \sum_{t\in T}\sum_{i\in I}\left(f_t^i y_t^i + c_t^i x_t^i\right) - \sum_{t\in T}\sum_{j\in J}\sum_{k=1}^{t} h_t^j d_k^j \tag{5.12}$$

$$\text{subject to} \quad \sum_{t\leq t'} \Theta_{jtt'} = d_{t'}^j, \quad \forall t' \in T, j \in J \tag{5.13}$$

$$\sum_{t'\geq t} \Theta_{jtt'} \leq \sum_{i\in I(j)} \alpha_i^j x_t^i, \quad \forall t \in T, j \in J \tag{5.14}$$

$$x_t^i \leq \max_{j\in J(i)} \left\{\frac{d_{tT}}{\alpha_i^j}\right\} y_t^i, \quad \forall t \in T, i \in I \tag{5.15}$$

$$x_t^i \geq 0, \quad \forall t \in T, i \in I \tag{5.16}$$

$$\Theta_{jtt'} \geq 0, \quad \forall t, t' \in T, j \in J \tag{5.17}$$

$$y_t^i \in \{0,1\}, \quad \forall t \in T, i \in I. \tag{5.18}$$

5.3. Plant Location Formulation 2 (ELS2)

We propose another formulation that is based on the simple plant location formulation of LS-U, called ELS2. In ELS2 formulation, production variables are not only disaggregated in terms of periods in which produced items are consumed by demand, but also in terms of the CPU they are produced by. $\Theta_{jtt'}$ variables of ELS1 are replaced by $\Theta_{tt'}^{ij}$, which gives the amount of product j produced using CPU i in period t to be consumed in period t', and necessary changes are applied to constraints (5.20)–(5.22). Note that ELS2 formulation has higher number of constraints and variables than ELS1 formulation due to Equation (5.21), and the disaggregation of $\Theta_{jtt'}$ into $\Theta_{tt'}^{ij}$, respectively. Then, the formulation can be given as:

$$\textbf{ELS2:} \text{ minimize } \quad \sum_{t\in T}\sum_{i\in I}\left(f_t^i y_t^i + c_t^i x_t^i\right) - \sum_{t\in T}\sum_{j\in J}\sum_{k=1}^{t} h_t^j d_k^j \tag{5.19}$$

$$\text{subject to} \quad \sum_{i\in I(j)}\sum_{t\leq t'}\Theta_{tt'}^{ij} = d_{t'}^j, \qquad \forall t' \in T, j \in J \tag{5.20}$$

$$\sum_{t'\geq t}\Theta_{tt'}^{ij} \leq \alpha_i^j x_t^i, \qquad \forall i \in I, t \in T, j \in J(i) \tag{5.21}$$

$$x_t^i \leq \max_{j\in J(i)}\left\{\frac{d_{tT}}{\alpha_i^j}\right\} y_t^i, \qquad \forall t \in T, i \in I \tag{5.22}$$

$$x_t^i \geq 0, \qquad \forall t \in T, i \in I \tag{5.23}$$

$$\Theta_{tt'}^{ij} \geq 0, \qquad \forall t, t' \in T, j \in J, i \in I \tag{5.24}$$

$$y_t^i \in \{0,1\}, \qquad \forall t \in T, i \in I. \tag{5.25}$$

5.4. Equivalence of Alternative Model Formulations

In order to show the equivalence of two linear mathematical models one can show any feasible solution of one model corresponds to some, also feasible, solution of the other model having the same objective value. This way one can be sure that feasible region of the first model is included in the feasible region of the second model. If the reverse also holds, then the models are said to be equivalent [27]. In this section, the equivalence will be shown explicitly between the linear relaxations of IP1 and IP2, IP2 and ELS1, ELS1 and ELS2.

The difference between IP1 and IP2 in constraints is the form of demand satisfaction constraints (3.6) and (5.8) respectively. Consider a feasible solution $(\hat{x}, \hat{y}, \hat{s})$

of linear relaxation of IP1. Since initial inventories are zero, $s_0^j = 0 \quad \forall j$, $\hat{x}$ satisfies Constraint (5.8) for $t = 1$. For $t > 1$, Constraints (5.8) can be obtained by summing up Constraints (3.6) from 1 to t. Therefore $\hat{x}$ satisfies (5.8), and $(\hat{x}, \hat{y})$ is a feasible solution to linear relaxation of IP2. Consider a feasible solution $(\bar{x}, \bar{y})$ of linear relaxation of IP2. Since (5.8) are summed up version of (3.6) from 1 to t, $(\bar{x}, \bar{y}, \bar{s})$ is feasible with respect to linear relaxation of IP1 where $\bar{s}$ is calculated with $(\bar{x}, \bar{y})$ using Equation (5.1). In Section 5.1, we show that both formulations have the same objective function. Therefore linear relaxations of the formulations IP1 and IP2 are equal.

Let us show the equivalence between linear relaxations of IP2 and ELS1. Consider constraints of the form (5.9) and (5.15). For a given feasible solution (x, y) of any of the models, the other model is also feasible with respect to (5.9) and (5.15). Let $(\hat{x}_t^i, \hat{y}_t^i, \hat{\Theta}_{jtt'})$ be a feasible solution of relaxed ELS1 formulation. Then, constraints (5.13) should hold for $\hat{\Theta}_{jtt'}$. Constraints (5.26) are summed up versions of constraints (5.13) from 1 to t. We get Equation (5.27) when indices of two summations are switched:

$$\sum_{z=1}^{t} \sum_{k=1}^{z} \hat{\Theta}_{jkz} = \sum_{z=1}^{t} d_z^j, \qquad \forall t \in T, j \in J; \tag{5.26}$$

$$\sum_{k=1}^{t} \sum_{z=k}^{t} \hat{\Theta}_{jkz} = d_{1t}^j, \qquad \forall t \in T, j \in J. \tag{5.27}$$

Constraints (5.14) should also hold for any feasible solution. Equation (5.28) is found when Constraints (5.14) are summed up from 1 to t. Equation (5.29) is the combination of Equations (5.27) and (5.28). Using Equation (5.29) we can conclude that $\hat{x}$ satisfy constraints (5.8); and hence, $(\hat{x}_t^i, \hat{y}_t^i)$ is feasible with respect to relaxed IP2 formulation. Their objective values are the same since both formulations share the same objective function.

$$\sum_{k=1}^{t}\sum_{z=k}^{T}\hat{\Theta}_{jkz} \leq \sum_{k=1}^{t}\sum_{i\in I(j)} \alpha_i^j \hat{x}_k^i, \qquad \forall t \in T, j \in J; \tag{5.28}$$

$$d_{1t}^j = \sum_{k=1}^{t}\sum_{z=k}^{t}\hat{\Theta}_{jkz} \leq \sum_{k=1}^{t}\sum_{z=k}^{T}\hat{\Theta}_{jkz} \leq \sum_{k=1}^{t}\sum_{i\in I(j)} \alpha_i^j \hat{x}_k^i, \qquad \forall t \in T, j \in J. \tag{5.29}$$

Now, let $(\hat{x}_t^i, \hat{y}_t^i)$ be a solution of relaxed IP2 formulation. Unfortunately, reverse mapping of x_t^i variables of IP2 formulation into $\Theta_{jtt'}$ variables of ELS1 formulation is not unique. This is due to the fact that some production is done not to satisfy demand but they are produced mandatorily due to co-production nature of the production environment. Since $\Theta_{jtt'}$ variables only represent consumed production and there may be excess production, it is possible to shift production-consumption assignment in terms of $\Theta_{jtt'}$ variables around. Therefore, using a simple first-in-first-out (FIFO) rule, it is possible to map any $(\hat{x}_t^i, \hat{y}_t^i)$ solution of relaxed IP2 formulation to a $(\hat{x}_t^i, \hat{y}_t^i, \hat{\Theta}_{jtt'})$ solution of relaxed ELS1 formulation. The proposed algorithm is shown in Figure 5.1.

Equivalence between linear relaxations of ELS1 and ELS2 follows from the relation between $\Theta_{jtt'}$ and $\Theta_{tt'}^{ij}$ variables. $\Theta_{tt'}^{ij}$ variables are CPU disaggregated version of $\Theta_{jtt'}$ variables. Let $(\hat{x}_t^i, \hat{y}_t^i, \hat{\Theta}_{tt'}^{ij})$ be a solution to ELS2 formulation. Then by setting $\hat{\Theta}_{jtt'} = \sum_{i\in I}\hat{\Theta}_{tt'}^{ij}$, $(\hat{x}_t^i, \hat{y}_t^i, \hat{\Theta}_{jtt'})$ will be a solution to ELS1 formulation. Let $(\bar{x}_t^i, \bar{y}_t^i, \bar{\Theta}_{jtt'})$ be a solution to ELS1 formulation. We need to map $\bar{\Theta}_{tt'}^{ij}$ arbitrarily from $\bar{\Theta}_{jtt'}$ variables, and this mapping is not unique. This mapping can be done with an algorithm similar to the one given in Figure 5.1.

We have shown that the feasible regions of relaxed IP1 and IP2 formulations, IP2 and ELS1 formulations, and ELS1 and ELS2 formulations are equal. Therefore, the feasible regions of all proposed models' linear relaxations are equal.

for Each product $j \in J$ **do**
  Make copy of demand vector d_t^j into D_t for all $t \in T$
  Make copy of production vector $\sum_{i \in I(j)} x_t^i \alpha_i^j$ into P_t for all $t \in T$
  for Each period $t \in T$ **do**
    for $p \in \{1, ..., T\}$ **do**
      if $P_p > D_t$ **then**
        $\Theta_{jpt} = D_t$
        $P_p = P_p - D_t$
        break
      else
        $\Theta_{jpt} = P_p$
        $D_t = D_t - P_p$
      end if
    end for
  end for
end for

Figure 5.1. Algorithm for mapping $\hat{\Theta}_{jtt'}$ from $\hat{x}_t^i$ using FIFO.

6. MODEL IMPROVEMENTS

In this chapter we first give valid inequalities that improve the lower bound obtained from linear relaxation of the model. Then, we provide two heuristics in order to improve the solution times.

6.1. Valid Inequalities

Valid inequalities, in general, improve the solution time required to solve integer programing formulations by narrowing the solution space. Although valid inequalities are not necessary to define the problem, they are satisfied for any feasible solution. Therefore, they could be violated by some fractional solutions of a branch and bound tree but they never eliminate any integer feasible solution. However, in some cases there exists exponential number of valid inequalities with respect to the problem size. This makes it inefficient to include all valid inequalities in the formulation. Hence, it is computationally more efficient to add valid inequalities that are violated by the fractional solution of the node relaxation during the branch and bound search in order to improve the lower bound.

Pochet and Wolsey [28] give Proposition 6.1 and Theorem 6.2 for the classical uncapacitated lot sizing problem LS-U.

Proposition 6.1. *Let $l \in T$, $L = \{1, ..., l\}$ and $S \subseteq L$, then the (l, S) inequality*

$$\sum_{q \in S} x_q \leq \sum_{q \in S} d_{ql} y_q + s_l \tag{6.1}$$

is valid for X^{LS-U}.

Theorem 6.2. *Inequalities of the form* (6.1)*, which are exponentially many, give complete description of conv(X^{LS-U}). Proof is in [28].*

By using inequalities (6.1), we develop valid inequalities given by Equation (6.2) for our problem. We prove that these inequalities are valid for DCCP in Proposition 6.3.

Proposition 6.3. *Let* $l \in T$*,* $L = \{1, ..., l\}$*,* $S \subseteq L$*,and* $j \in J$ *then the* (l, S, j) *inequality*

$$\sum_{i \in I(j)} \sum_{q \in S} x_q^i \alpha_i^j \leq \sum_{q \in S} d_{ql}^j \left(\sum_{i \in I(j)} y_q^i \right) + s_l^j \tag{6.2}$$

is valid for X^{DCCP}*.*

Proof. Consider a point $(s, y) \in X^{DCCP}$. If $\sum_{q \in S} \sum_{i \in I(j)} y_q^i = 0$, then as $\sum_{i \in I(j)} \sum_{q \in S} x_q^i = 0$, $s_l^j \geq 0$, the equality is satisfied. Otherwise let $t = min\{q \in S : \sum_{i \in I(j)} y_q^i > 0\}$. Then consider the following:

$$\sum_{i \in I(j)} \sum_{q \in S} x_q^i \alpha_i^j \leq \sum_{i \in I(j)} \sum_{q=t}^{l} x_q^i \alpha_i^j \leq d_{tl}^j + s_l^j \leq \sum_{q \in S} d_{ql}^j \left(\sum_{i \in I(j)} y_q^i \right) + s_l^j \tag{6.3}$$

First part of the inequality (6.3) follows from non-negativity of $x_q^i \alpha_i^j$ terms and the definition of subset S and time index t. The second part follows from flow conservation equations. Finally, the last part holds using $\sum_{i \in I(j)} y_t^i \geq 1$ and the non-negativity of y_t^i. □

Remark. *Inequalities of the form* (6.2)*, does not give complete description of* $conv(X^{DCCP})$*.*

Equation (6.4) is valid for DCCP because of inventory flow constraints, where $l \in T$, $L = \{1, ..., l\}$, $S \subseteq L$.

$$\sum_{i \in I(j)} \sum_{q \in L} x_q^i \alpha_i^j = d_{1l}^j + s_l^j \tag{6.4}$$

By using Equation (6.4) and Inequality (6.2), we obtain Inequality (6.5).

$$\sum_{i \in I(j)} \sum_{q \in L \setminus S} x_q^i \alpha_i^j + \sum_{q \in S} d_{ql}^j \left(\sum_{i \in I(j)} y_q^i \right) \geq d_{1l}^j \tag{6.5}$$

Note that valid inequalities of the form (6.5) are exponentially many. However, they can be separated by inspection using the algorithm given in Figure 6.1. A straightforward application of the algorithm leads to O(n^2) complexity whereas O($nlog(n)$) is doable by adapting improvement proposed in [28]. Assume a fractional solution (x_q^{i*}, y_q^{i*}) to apply separation algorithm given in Figure 6.1. Note that this separation is perfect, i.e. the algorithm finds all violated valid inequalities for a given solution.

```
for Each product j ∈ J do
  for l = 1, ..., T do
    Calculate D_l^j = Σ_{q=1}^{l} min{ Σ_{i∈I(j)} x_q^{i*} α_i^j , d_{ql}^j ( Σ_{i∈I(j)} y_q^{i*} ) }
    if D_l^j < d_{1l}^j then
      return j, L = {1, ..., l}, S = { q ∈ L : Σ_{i∈I(j)} x_q^{i*} α_i^j > d_{ql}^j ( Σ_{i∈I(j)} y_q^{i*} ) }
    end if
  end for
end for
```

Figure 6.1. Algorithm for (l, S, j) Separation.

Inequalities of the form (6.2) and (6.5) are not very tight when more than one CPU produces an item in a period due to term $\sum_{i \in I(j)} y_q^i$. Let us define a new binary variable called z_j^t to facilitate production of product j in period t. Additional constraints to be added to the model are (6.6) and (6.7):

$$y_t^i \leq z_j^t \qquad \forall j \in J, i \in I(j), t \in T; \tag{6.6}$$

$$\sum_{i \in I(j)} y_t^i \geq z_j^t \qquad \forall j \in J, t \in T. \tag{6.7}$$

Then, replacing $\sum_{i \in I(j)} y_q^i$ in (6.2) with z_j^q results in inequalities (6.8), which can be transformed into (6.9) using (6.4):

$$\sum_{i \in I(j)} \sum_{q \in S} x_q^i \alpha_i^j \leq \sum_{q \in S} d_{ql}^j z_j^q + s_l^j; \tag{6.8}$$

$$\sum_{i \in I(j)} \sum_{q \in L \setminus S} x_q^i \alpha_i^j + \sum_{q \in S} d_{ql}^j z_j^q \geq d_{1l}^j. \tag{6.9}$$

Then, z_j^q transformed (l, S, j) separation algorithm is shown in Figure 6.2 for a solution (x_q^{i*}, z_j^{q*}).

```
for Each product j ∈ J do
  for l = 1, ..., T do
    Calculate D_l^j = Σ_{q=1}^{l} min{ Σ_{i∈I(j)} x_q^{i*} α_i^j , d_{ql}^j z_j^{q*} }
    if D_l^j < d_{1l}^j then
      return j, L = {1, ..., l}, S = { q ∈ L : Σ_{i∈I(j)} x_q^{i*} α_i^j > d_{ql}^j z_j^{q*} }
    end if
  end for
end for
```

Figure 6.2. Algorithm for (l, S, j) Separation using z_j^q variables.

6.1.1. Valid Inequalities for Alternative Model Formulations

Valid inequalities described in this chapter can be applied to ELS1 and ELS2 formulations using the same logic as follows:

$$\sum_{q \in L \setminus S} \sum_{t' \geq q} \Theta_{jqt'} + \sum_{q \in S} d_{ql}^j \left(\sum_{i \in I(j)} y_q^i \right) \geq d_{1l}^j, \tag{6.10}$$

$$\sum_{q \in L \setminus S} \sum_{t' \geq q} \sum_{i \in I(j)} \Theta_{qt'}^{ij} + \sum_{q \in S} d_{ql}^j \left(\sum_{i \in I(j)} y_q^i \right) \geq d_{1l}^j. \tag{6.11}$$

The idea used in writing Inequalities (6.6) and (6.7) can also be applied to Equation (6.10) and (6.11), converting them into (6.12) and (6.13) for ELS1 and ELS2, respectively:

$$\sum_{q \in L \setminus S} \sum_{t' \geq q} \Theta_{jqt'} + \sum_{q \in S} d^j_{ql} z^q_j \geq d^j_{1l}, \tag{6.12}$$

$$\sum_{q \in L \setminus S} \sum_{t' \geq q} \sum_{i \in I(j)} \Theta^{ij}_{qt'} + \sum_{q \in S} d^j_{ql} z^q_j \geq d^j_{1l}. \tag{6.13}$$

Inequalities (6.12) and (6.13) are tighter than the original valid inequalities due to constraints (5.14) and (5.21) for ELS1 and ELS2, respectively.

6.2. Heuristics

Branch and bound algorithm (B&B) is a structural way of searching integer feasible solution space by constructing a tree of possible solutions. B&B is faster than explicit enumeration since it is possible to prune some branches of B&B tree using the global lower and upper bound information available. So far, we have investigated how to improve lower bounds using valid inequalities in Section 6.1. In this section we discuss two heuristics that we develop for DCCP.

Heuristics can be used to give initial solutions to B&B or to create feasible solutions from fractional node relaxation solutions of branch and bound tree. First heuristic we discuss is named as Pattern Fitting heuristic. Its sole purpose is to provide an initial solution to B&B. The second heuristic is named as CPWW (Co-Production Wagner Whitin), which can create many feasible solutions from a fractional node relaxation solution. Both of these heuristics will be discussed in detail in their respective sections.

6.2.1. Pattern Fitting Heuristic

A pattern fitting heuristic is developed in order to give an initial solution to the commercial solver. Product coverage of a CPU is defined for this heuristic as

```
Initialize R_j, C_j, X_it, Y_it, S_jt = 0, minratio = ∞
for Period t ∈ T do
  for Product j ∈ J do
    R_j = R_j + d_t^j
  end for
  if R_j ≤ 0 then
    C_j = 1
  else
    C_j = 0
  end if
  while C_j = 0  ∃j ∈ J do
    for CPU i ∈ I do
      for Product j ∈ J(i) do
        ratio = R_j/α_i^j
        if ratio > X_it then
          X_it = ratio
        end if
      end for
      for Product j ∈ J(i) do
        S_jt = X_it * α_i^j − R_j
      end for
      Define J'(i) = {j | R_j > 0, j ∈ J(i)}
      if J'(i) ≠ ∅ then
        CTCR = {f_i^t + X_it * p_i^t + Σ_{j∈J(i)} h_j^t * S_jt}/|J'(i)|
      end if
      if CTCR < minratio then
        minratio = CTCR
        CPUindex = i
      end if
    end for
    if CPUindex ≥ 0 then
      Y_it = 1
      for Product j ∈ J(i) do
        C_j = 1
        R_j = R_j − X_{CPUindex,t} * α_{CPUindex}^j
      end for
    end if
  end while
end for
return X_it, Y_it, S_jt
```

Figure 6.3. Pattern Fitting Heuristic.

the number of products that CPU can produce, which have uncovered demand in considered period t. Our heuristic works as follows: starting from the first period, the algorithm tries to cover all demand. The CPU, which has the lowest cost to product coverage ratio is selected and that CPU is produced at an amount that covers all demand of products that CPU is producing. Those products are marked as covered, and the algorithm selects the next CPU with minimum cost to product coverage ratio until all products are covered for first period. Then, the excess production is reduced from demand for the next period and the algorithm continues for period 2, and so on to the last period. This algorithm is given in Figure 6.3.

6.2.2. Co-Production Wagner Whitin Heuristic (CPWW)

One can continuously improve the upper bound of B&B search by creating good integer feasible solutions from fractional node relaxation solutions. Co-Production Wagner Whitin Heuristic (CPWW) is developed in order to create feasible solutions from fractional node relaxation solutions of branch and bound tree. CPWW can create many feasible solutions from a fractional node relaxation solution due to randomness of the ordering of the products considered.

This heuristic works as follows: given a fractional solution vector of y_i^t, each product is considered in a random order. For each period, the product considered is assigned to a CPU from its available CPUs. In this assignment, the CPU that has the highest y_i^t value is selected. The assumption is if any CPU has higher y_i^t value in a fractional solution, then it has higher likelihood to appear in an optimal solution. Next, Wagner - Whitin algorithm is called for the product considered, considering selected CPU's cost parameters as if it was a single product having demand d_j^t/α_i^j. Note that different CPUs can be selected for different periods for the same product. Production amounts of other co-produced products are then reflected in the demand matrix, and the algorithm continues until every product is considered. The algorithm is given in Figure 6.4.

Given fractional solution y_i^t

for Product $j \in J(i)$ in random order **do**

  for CPU $i \in I(j)$ **do**

    temp = 0

    for Period $t \in T$ **do**

      if $y_i^t > temp$ **then**

        temp = y_i^t

        $SelectedCPUs(t) = i$

      end if

    end for

    Apply Wagner - Whitin Algorithm using costs of SelectedCPUs for each period and demand matrix.

    Reflect production of other co-produced products in demand matrix

  end for

end for

Figure 6.4. Co-Production Wagner Whitin Heuristic.

7. COMPUTATIONAL EXPERIMENTS AND RESULTS

Computational experiments are done on 4 problem sets, each having 10 instances. Time limit is set to 20 minutes per instance for all tests. We implement the models and algorithms in C++ programming language and we use IBM CPLEX 12.8 in 64 bit mode for MIP solutions. Benchmark tests are performed on an Intel Core i7-3820 3.6 GHz machine with 32GB RAM and 10MB cache, running Windows 10 operating system.

7.1. Data Generation Process

There is no available data library that we can directly use in the deliberated co-production in dynamic deterministic lot sizing literature. Therefore, we generate random data sets for experimentation. We use the data generation process used by [29] to generate a production planning problem where applicable. The procedure described in [29] has 20 products across three product families as shown in Table 7.1. Each product has a lower and an upper bound on demand for each period to be determined using uniform distribution. When more than 20 products are present in an instance, mod operation is used to determine the family of a product. For example, product 21 belongs to the first family whereas, product 26 belongs to the second product family. Holding costs and bounds on fixed costs used by [29] are given in Table 7.2. Variable costs given in Table 7.2, however, do not exist in the context of [29]. Values of 10, 15, and 20 are taken as variable costs for product families 1, 2, and 3 respectively.

Table 7.1. Product Families and Demand Data Used in Experimentation.

Family	1	1	1	1	1	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Demand LB	20	40	15	25	80	80	15	40	40	55	20	20	30	30	20	30	30	50	60	40
Demand UB	40	60	25	65	120	120	25	60	60	85	40	40	50	50	40	70	50	100	90	80

We create co-production units randomly while ensuring that each product is a part of at least one CPU to ensure feasibility. A parameter named *density* is used to determine the number of products inside each CPU. Each CPU can produce at least two, and at most *density* many products.

Table 7.2. Holding Cost, Fixed Cost, and Variable Cost Used in Experimentation.

Product Family	**Holding Cost**	**Fixed Cost LB**	**Fixed Cost UB**	**Variable Cost**
1	1	50	150	10
2	1.75	100	200	15
3	1.5	100	200	20

The fixed cost of a CPU is determined by taking %75 of the sum of the fixed costs of products that CPU includes. Similarly, the variable cost of a CPU is taken as %90 of the sum of the variable costs of its products. Problem sets used are summarized in Table 7.3.

Table 7.3. Problem Set Descriptions.

	Period	**Product**	**CPU**	**Density**
Problem Set 1	24	40	200	3
Problem Set 2	24	40	200	6
Problem Set 3	36	40	200	3
Problem Set 4	36	40	200	6

7.2. Comparison of Alternative Model Formulations

CPLEX parameter settings used for experimentation are in Table 7.4. At this stage of tests, all four model formulations are tested without any model improvements in all problem sets. Summarized results can be seen in Table 7.5. Columns of the Table 7.5 given from the left to right are as follows: Average time spent in the B&B algorithm in seconds excluding the time spent in node zero, average optimality gap, average number of processed nodes of B&B tree, number of best solutions found per test instance. The reader may refer to Tables A.1-A.4 for extensive results.

Table 7.4. CPLEX Parameter Settings.

Timelimit	1200
Threads	8
MIPGap	0

IP1 formulation bests all other formulations in terms of average gap, and provides the best feasible solution at the end of the time limit in 31 of 40 test instances. IP2 performs worse than IP1 in the number of best solutions found and the average gap in all three problem sets. Computational difficulties arising from the denser constraint matrix of IP2 outweighs the possible benefits arising from fewer number of variables compared to IP1.

Table 7.5. Summary of Results for Alternative Model Formulations.

	Time Spent in B&B				% Gap				Processed Nodes				# Best Sol.			
Problem Set	**IP1**	**IP2**	**ELS1**	**ELS2**	**IP1**	**IP2**	**ELS1**	**ELS2**	**IP1**	**IP2**	**ELS1**	**ELS2**	**IP1**	**IP2**	**ELS1**	**ELS2**
Problem Set 1	749.07	63.83	71.14	369.39	9.65	13.39	11.11	19.13	604	9	0	2	6	2	2	0
Problem Set 2	0.04	0.06	0.05	0.06	18.21	25.55	26.93	38.11	0	0	0	0	8	2	0	0
Problem Set 3	6.58	0.12	0.06	0.09	11.07	20.77	14.80	24.92	0	0	0	0	7	2	1	0
Problem Set 4	0.04	0.14	0.05	0.09	22.72	37.71	32.13	47.64	0	0	0	0	10	0	0	0

In 20 minutes time limit, except IP1 formulation in problem set 1, most of the instances resulted in zero or very small number of branch and bound nodes processed. CPLEX was unable to finish processing node zero in order to start processing other nodes of the branch and bound tree.

It is interesting to note that, although IP2 formulation has a higher number of best solution found than ELS1, the latter has smaller average gap than IP2 except in problem set 2. Note that all formulations have the same linear relaxation objective value (RlxObj column of Tables A.1-A.4) as shown theoretically in Section 5.4.

ELS2 formulation performs the worst, and gives the highest average gap values for all four problem sets. Notice that ELS2 formulation has the highest number of constraints for any given problem instance due to the constraints of the form (5.21). As density parameter increases from 3 to 6, only the number of constraints for ELS2 increases due to constraint (5.21), which in turn greatly increases the problem size for ELS2 formulation resulting in the worst performance.

As a conclusion of this step, IP1 formulation outperforms IP2 and ELS2 has the worst performance. Therefore, results of formulations IP2 and ELS2 will not be shown in upcoming stages of experimentations.

7.3. Comparison of Valid Inequalities in IP1 and ELS1 Formulations

In this section effects of adding valid inequalities by using the separation algorithm given in Section 6.1 to IP1 and ELS1 formulations with and without z_j^t variables, as given in Section 6.1, will be discussed. This experimentation will be based on LP relaxations of these formulations. We repeatedly solve LP relaxation of a model and add violated valid inequalities as needed. Algorithms for separating valid inequalities from a given fractional solution is given in Figure 6.1 and Figure 6.2 for separating without z_j^t and with z_j^t variables, respectively. Implementing these algorithms directly caused some problems. For example, if a valid inequality that is violated by only a small amount is added to the formulation, CPLEX may not register it as a violated constraint due to numerical tolerances of CPLEX. This results in an infinite loop in some problem instances. Therefore, we restrict ourself to add valid inequalities that are violated by a specified value, called *epsilon*, to register as violated valid inequalities. Upon preliminary experimentation it is observed that the value of epsilon does not matter as long as it is not close to zero. Hence the value of *epsilon* is set to 20 for all runs with valid inequalities.

Another problem that we experienced while adding valid inequalities is there is no mechanism to stop generating valid inequalities when they no longer improve current lower bound. In order to remedy this problem, we stop searching for valid inequalities when the percentage increase in the lower bound as a result of adding valid inequalities is less than 0.5. We set the root algorithm parameter of CPLEX to dual simplex for this part of experimentation. We provide a summary of results in Table 7.6. Columns of Table 7.6 include: Average time spent in seconds generating valid inequalities, average increase in the lower bound compared to LP relaxation, average % gap between new lower bound and the best integer feasible solution obtained in Section 7.2, average number of original constraints, and average number of added valid inequalities. Detailed results are given in Appendix section A.2.

In Section 6.1, where valid inequalities for DCCP are proposed, we note that with an additional variable referred as z_j^t, proposed valid inequalities in fact generate tighter relaxations. We see higher number of valid inequalities added to the models for z_j^t variations due to higher number of valid inequalities cutting any given fractional solution. However, as we see in the second column of Table 7.6, the average increase in the lower bound of linear relaxation of the problem does not always increase with the addition of z_j^t variables to the models IP1 and ELS1. On the contrary, adding z_j^t variables reduces the increase of lower bound of linear relaxation for IP1 despite an increase in the average number of valid inequalities. Additionally, the time required to create valid inequalities also increases with the addition of z_j^t variables to IP1 formulation. For ELS1 model, z_j^t variables increase the average percent increase in LP lower bound for problem sets 1 and 2 by 0.03 and 0.01, respectively, and reduced by 0.05 and 0.09 for problem sets 3 and 4, respectively.

The average number of constraints in the model increases approximately 4 times with additional z_j^t variables for both IP1 and ELS1 formulations for all problem sets. This increase in the constraint number is higher in problem sets 2 and 4, where density parameter is higher than that of problem sets 1 and 3. This will be a major drawback for using additional z_j^t variables when solving corresponding IP's to optimality due to increased model size.

Table 7.6. Summary of Results for Valid Inequalities.

	Avg. Time Spent				**Avg. Increase % in LP LB**				**Avg. % Gap with Best Soln.**				**Avg. Number of Constraints**				**Avg. Number of Valid Ineq.**			
Problem Set	**IP1**	**IP1Z**	**ELS1**	**ELS1Z**	**IP1**	**IP1Z**	**ELS1**	**ELS1Z**	**IP1**	**IP1Z**	**ELS1**	**ELS1Z**	**IP1**	**IP1Z**	**ELS1**	**ELS1Z**	**IP1**	**IP1Z**	**ELS1**	**ELS1Z**
Problem Set 1	49.74	75.83	50.53	41.08	24.21	23.87	24.21	24.24	21.48	21.82	21.48	21.45	5760	18708	6720	19668	11925	13298	26329	11952
Problem Set 2	118.22	209.27	99.45	96.15	22.96	21.46	22.96	22.97	37.10	38.79	37.10	37.09	5760	26028	6720	26988	12426	14620	27177	11955
Problem Set 3	240.58	423.60	272.91	461.05	23.69	22.93	23.75	23.70	24.34	25.12	24.29	24.33	8640	28062	10080	29502	11924	28543	26322	26325
Problem Set 4	563.27	568.24	485.82	693.06	20.95	15.22	20.89	20.80	46.09	53.74	46.17	46.29	8640	39042	10080	40482	11917	23481	26250	25508

Valid inequalities generated using LP relaxations of formulations IP1 and ELS1 perform similarly in average increase in LP lower bound and average time spent. However, IP1 achieves the same level of lower bound increase with fewer number of valid inequalities compared to ELS1 formulation. Therefore, valid inequalities generated using LP relaxation of IP1 formulation are more efficient compared to that of ELS1 formulation.

In summary, additional z_j^t variables are not justifiable with a small percentage of lower bound increase in LP relaxation in some of the problem sets, while introducing a very large number of constraints to both formulations. Additionally, IP1 formulation has the same increase in LP lower bound with ELS1 formulation and this increase is achieved with lesser number of valid inequalities. Therefore, only IP1 formulation and valid inequalities without z_j^t variables will be considered in upcoming stages of experimentations.

7.4. Results of Heuristics

In this section, IP1 formulation is tested using four different setting and compared to the original MIP implementation. In the first setting, valid inequalities are implemented in IP1 using callback structure of CPLEX. In the second setting, the pattern fitting heuristic to generate initial solution is added to the IP1. Next, valid inequalities are tested together with pattern fitting heuristic. Lastly, CPWW is also added to IP1 together with valid inequalities and pattern fitting heuristic. Results are provided in Table 7.7. Columns of Table 7.7 from left to right show: Average amount of time spent in seconds in pattern fitting heuristic, average objective function value of the solution generated using pattern fitting heuristic, average amount of time spent in seconds in node 0 of B&B tree, average amount of time spent in the B&B algorithm in seconds excluding the time spent in node zero, average % optimality gap, average number of B&B nodes considered, number of best solutions found per test instance, average number of valid inequalities generated, average number of times CPLEX callback is called.

Table 7.7. Summary of Results for IP1 in Different Settings.

Problem Set	Valid Inequalities	PatternFitting	CPWW	InitHeurTime	InitHeurObj	Node0Time	B&BTime	% Gap	Nnodes	# Best Sol.	Ncuts	Ncallback
Problem Set 1	-	-	-	-	-	450.9	749.1	9.65	604	0	-	-
	+	-	-	-	-	500.2	699.9	9.30	449	0	2649	851
	-	+	-	0.0	1270611	497.2	702.8	8.91	549	**4**	-	-
	+	+	-	0.0	1270611	533.9	666.1	**8.64**	422	**4**	2669	814
	+	+	+	0.0	1270611	603.6	597.3	**8.64**	243	2	3062	435
Problem Set 2	-	-	-	-	-	1205.1	0.0	18.21	0	1	-	-
	+	-	-	-	-	1204.6	0.0	18.27	0	1	1834	17
	-	+	-	0.0	1674308	1176.0	27.6	**15.15**	0	0	-	-
	+	+	-	0.0	1674308	1203.6	0.0	15.44	0	**5**	1845	17
	+	+	+	0.0	1674308	1204.0	0.0	15.44	0	3	2164	15
Problem Set 3	-	-	-	-	-	1193.9	6.6	11.07	0	1	-	-
	+	-	-	-	-	1200.4	0.0	10.82	0	1	4001	13
	-	+	-	0.0	1931058	1199.6	0.8	**9.76**	0	**4**	-	-
	+	+	-	0.0	1931058	1200.5	0.0	9.96	0	3	4023	14
	+	+	+	0.0	1931058	1200.7	0.0	10.07	0	1	4565	12
Problem Set 4	-	-	-	-	-	1202.9	0.0	22.72	0	0	-	-
	+	-	-	-	-	1203.0	0.0	24.12	0	0	2967	7
	-	+	-	0.0	2635466	1203.0	0.0	**18.62**	0	3	-	-
	+	+	-	0.0	2635466	1204.1	0.0	19.14	0	**6**	3058	7
	+	+	+	0.0	2635466	1202.9	0.1	19.50	0	1	3304	7

We observe that callback implemented valid inequalities alone reduces the average percent gap only in small density problem sets, sets 1 and 3, compared to base IP1 formulation. In high density problem sets, sets 2 and 4, we observe an increase in the average percent gap. Pattern fitting heuristic improves the solution quality compared to base IP1 formulation with or without valid inequalities. CPWW however, has no effect on average percent gap in problem sets 1 and 2, and results in an increase in problem sets 3 and 4.

Pattern fitting heuristic without valid inequalities results in best average percent gap in problem sets 2 and 4. However, using valid inequalities in addition to the pattern fitting heuristic results in increased number of best integer solutions found. For problem set 2, IP1 with pattern fitting heuristic was unable to provide any best solution in 10 test instances, while resulting in best average percent gap. Using Valid inequalities together with pattern fitting heuristic however, provided 5 best integer solutions out of 10. Similar pattern is also seen in results for problem set 4.

We observe that using pattern fitting heuristic together with valid inequalities improves upon IP1 formulation. Adding CPWW heuristic to the mix however, does not increase the number of best solutions found or decrease average percent gap. As a result of computational experimentation, we conclude that solving IP1 formulation with pattern fitting heuristic and valid inequalities without z_j^t variables yields best results.

8. CONCLUSION

In this thesis, we study lot a sizing problem in deliberated and controlled co-production setting. This problem, to the best of our knowledge, was not addressed in the literature previously.

In the first part of this thesis, we define and structure DCCP. We prove that DCCP is an NP-Hard problem. Hence, we cannot hope to find a polynomial time algorithm to solve it. Therefore, we provide several special cases for which DCPP is polynomially solvable and propose solution techniques for those cases.

We propose four alternative MIP formulations for DCCP. Two of these formulations (IP1 and IP2) are similar to original LS-U formulation. The other two (ELS1 and ELS2) are based on the simple plant location formulation of LS-U. Then, we show that all proposed formulations are equivalent in terms of their linear programming relaxations.

In order to reduce the solution times and increase solution qualities of proposed MIP formulations, we focus on finding valid inequalities. We show that inequalities converted from *l-s* inequalities of LS-U are valid for DCCP. We show that proposed valid inequalities can increase the LP bound by at least %20 for our test cases.

We propose two different heuristics to help with the upper bounds. According to our computational experiments, proposed pattern fitting heuristic results in the lowest average percent gap for 3 out of 4 problem sets. Pattern fitting heuristic together with valid inequalities provides the highest number of best solution found in 3 out of 4 problem sets.

We achieve at least %10 improvement in average percent gap over the IP1 formulation with proposed model improvements, pattern fitting heuristic and valid inequalities. We also show that at the end of the time limit models with our modifications provide better feasible solutions.

Introducing backlogging option and capacity restrictions to DCCP could be a possible future research direction. However, we do believe that, before tackling harder variations of DCCP, we should be able to find efficient solution techniques to the plain version of the problem. Implementing more clever heuristics could be a possible improvement on solution times and quality. CPWW heuristic proposed in this thesis generates solutions in which multiple items are produced well over their demand levels in some cases, and hence it can be improved. A standalone heuristic option, rather than to help B&B search, is also a possibility to generate feasible solutions to problem instances that are too difficult to solve exactly.

REFERENCES

1. Bitran, G. R. and S. M. Gilbert, "Co-production processes with random yields in the semiconductor industry", *Operations Research*, Vol. 42, No. 3, pp. 476–491, 1994.

2. Bitran, G. R. and S. Dasu, "Ordering policies in an environment of stochastic yields and substitutable demands", *Operations Research*, Vol. 40, No. 5, pp. 999–1017, 1992.

3. Öner, S. and T. Bilgiç, "Economic lot scheduling with uncontrolled co-production", *European Journal of Operational Research*, Vol. 188, No. 3, pp. 793–810, 2008.

4. Taşkın, Z. C. and A. T. Ünal, "Tactical level planning in float glass manufacturing with co-production, random yields and substitutable products", *European Journal of Operational Research*, Vol. 199, No. 1, pp. 252–261, 2009.

5. Zhou, L., S. Hu, Y. Li and Q. Zhou, "Study on co-feed and co-production system based on coal and natural gas for producing DME and electricity", *Chemical Engineering Journal*, Vol. 136, No. 1, pp. 31–40, 2008.

6. Wagner, H. M. and T. M. Whitin, "Dynamic version of the economic lot size model", *Management Science*, Vol. 5, No. 1, pp. 89–96, 1958.

7. Brahimi, N., S. Dauzere-Peres, N. M. Najid and A. Nordli, "Single item lot sizing problems", *European Journal of Operational Research*, Vol. 168, No. 1, pp. 1–16, 2006.

8. Brahimi, N., N. Absi, S. Dauzere-Peres and A. Nordli, *Single-Item Dynamic Lot-Sizing Problems: An Updated Survey*, Tech. rep., Working paper, 2016.

9. Wagelmans, A., S. Van Hoesel and A. Kolen, "Economic lot sizing: an O (n log

n) algorithm that runs in linear time in the Wagner-Whitin case", *Operations Research*, Vol. 40, No. 1-supplement-1, pp. S145–S156, 1992.

10. Aggarwal, A. and J. K. Park, "Improved algorithms for economic lot size problems", *Operations Research*, Vol. 41, No. 3, pp. 549–571, 1993.

11. Federgruen, A. and M. Tzur, "A simple forward algorithm to solve general dynamic lot sizing models with n periods in 0 (n log n) or 0 (n) time", *Management Science*, Vol. 37, No. 8, pp. 909–925, 1991.

12. Hsu, V. N., "Dynamic economic lot size model with perishable inventory", *Management Science*, Vol. 46, No. 8, pp. 1159–1169, 2000.

13. Pochet, Y. and L. A. Wolsey, "Polyhedra for lot-sizing with Wagner-Whitin costs", *Mathematical Programming*, Vol. 67, No. 1-3, pp. 297–323, 1994.

14. Hsu, V. N., C.-L. Li and W.-Q. Xiao, "Dynamic lot size problems with one-way product substitution", *IIE Transactions*, Vol. 37, No. 3, pp. 201–215, 2005.

15. Chen, W.-H. and J.-M. Thizy, "Analysis of relaxations for the multi-item capacitated lot-sizing problem", *Annals of Operations Research*, Vol. 26, No. 1, pp. 29–72, 1990.

16. Robinson, P., A. Narayanan and F. Sahin, "Coordinated deterministic dynamic demand lot-sizing problem: A review of models and algorithms", *Omega*, Vol. 37, No. 1, pp. 3–15, 2009.

17. Robinson, E. P. and F. B. Lawrence, "Coordinated Capacitated Lot-Sizing Problem with Dynamic Demand: A Lagrangian Heuristic", *Decision Sciences*, Vol. 35, No. 1, pp. 25–53, 2004.

18. Tomlin, B. and Y. Wang, "Pricing and operational recourse in coproduction systems", *Management Science*, Vol. 54, No. 3, pp. 522–537, 2008.

19. Vidal-Carreras, P. I., J. P. Garcia-Sabater and J. R. Coronado-Hernandez, "Economic lot scheduling with deliberated and controlled coproduction", *European Journal of Operational Research*, Vol. 219, No. 2, pp. 396–404, 2012.

20. Rafiei, R., M. Nourelfath, J. Gaudreault, L. A. De Santa-Eulalia and M. Bouchard, "Dynamic safety stock in co-production demand-driven wood remanufacturing mills: A case study", *International Journal of Production Economics*, Vol. 165, pp. 90–99, 2015.

21. Clark, J. H., T. J. Farmer, L. Herrero-Davila and J. Sherwood, "Circular economy design considerations for research and process development in the chemical sciences", *Green Chemistry*, Vol. 18, No. 14, pp. 3914–3934, 2016.

22. Larson, E. D., G. Fiorese, G. Liu, R. H. Williams, T. G. Kreutz and S. Consonni, "Co-production of decarbonized synfuels and electricity from coal+ biomass with CO 2 capture and storage: an Illinois case study", *Energy & Environmental Science*, Vol. 3, No. 1, pp. 28–42, 2010.

23. Ağralı, S., "A dynamic uncapacitated lot-sizing problem with co-production", *Optimization Letters*, Vol. 6, No. 6, pp. 1051–1061, 2012.

24. Volker Strassen (auth.), J. W. T. J. D. B. e., Raymond E. Miller, *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972*, The IBM Research Symposia Series, Springer US, 1 edn., 1972.

25. Garey, M. R. and D. S. Johnson, "Computers and intractability: a guide to the theory of NP-completeness. 1979", *San Francisco, LA: Freeman*, Vol. 58, 1979.

26. Krarup, J. and O. Bilde, "Plant location, set covering and economic lot size: An 0 (mn)-algorithm for structured problems", *Numerische Methoden bei Optimierungsaufgaben Band 3*, pp. 155–180, Springer, 1977.

27. Taşkın, Z. C. and T. Ekim, "Integer programming formulations for the minimum weighted maximal matching problem", *Optimization Letters*, Vol. 6, No. 6, pp. 1161–1171, 2012.

28. Pochet, Y. and L. A. Wolsey, *Production planning by mixed integer programming*, Springer Science & Business Media, 2006.

29. Graves, S. C., "Using Lagrangean techniques to solve hierarchical production planning problems", *Management Science*, Vol. 28, No. 3, pp. 260–275, 1982.

APPENDIX A: RESULTS

A.1. Experiments on Alternative Model Formulations

Table A.1. Experiments with Alternative Model Formulations on Problem Set 1.

Model	LPTime	RlxObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.
IP1	0.10	681132	486	929060	1056890	713.58	1055160	11.85	314	5760
	0.10	720805	651	982898	1077270	549.24	1075630	8.59	253	5760
	0.08	720478	337	980186	1109420	863.37	1105120	11.22	582	5760
	0.10	684124	370	921840	1018560	829.73	1014380	9.10	807	5760
	0.08	703107	404	997078	1120870	796.39	1117610	10.76	719	5760
	0.09	714616	572	935113	1030780	628.33	1028580	9.03	143	5760
	0.10	745050	367	1019280	1114630	833.42	1113090	8.41	1086	5760
	0.08	693586	675	965924	1085140	525.22	1082400	10.70	170	5760
	0.10	718966	312	973868	1066660	888.30	1066190	8.61	1296	5760
	0.11	727199	337	1002600	1094200	863.09	1093660	8.27	670	5760
Average:	**0.09**	**710906**	**451**	**970785**	**1077442**	**749.07**	**1075182**	**9.65**	**604**	**5760**
IP2	0.44	681132	1200	897747	1046300	0.02	1046300	14.20	0	5760
	0.45	720805	1103	965069	1108000	96.51	1108000	12.78	0	5760
	0.29	720478	1200	942971	1109220	0.09	1109220	14.99	0	5760
	0.30	684124	1127	894052	1022840	72.97	1022840	12.45	0	5760
	0.29	703107	1200	961307	1107260	0.09	1107260	13.18	0	5760
	0.33	714616	1200	897175	1057240	0.09	1057240	15.14	0	5760
	0.35	745050	1090	999590	1125830	109.79	1125830	10.99	0	5760
	0.30	693586	1200	910994	1096820	0.09	1096820	16.94	0	5760
	0.32	718966	841	952347	1074000	358.66	1072690	11.00	91	5760
	0.46	727199	1200	966365	1101070	0.02	1101070	12.23	0	5760
Average:	**0.35**	**710906**	**1136**	**938762**	**1084858**	**63.83**	**1084727**	**13.39**	**9**	**5760**
ELS1	0.39	681132	1200	924676	1076990	0.02	1076990	14.14	0	6720
	0.35	720805	1200	976797	1118010	0.02	1118010	12.63	0	6720
	0.25	720478	1157	979582	1098030	43.22	1098030	10.79	0	6720
	0.25	684124	1201	920015	1030560	0.02	1030560	10.73	0	6720
	0.25	703107	1044	997394	1115930	155.65	1114680	10.50	0	6720
	0.25	714616	1201	931363	1060200	0.02	1060200	12.15	0	6720
	0.26	745050	927	1016330	1117530	272.50	1112740	8.55	0	6720
	0.25	693586	1201	961964	1100200	0.05	1100200	12.56	0	6720
	0.24	718966	1166	972889	1075540	34.41	1075540	9.54	0	6720
	0.33	727199	994	1001790	1110650	205.51	1107670	9.53	0	6720
Average:	**0.28**	**710906**	**1129**	**968280**	**1090364**	**71.14**	**1089462**	**11.11**	**0**	**6720**
ELS2	3.84	681132	881	932439	1214570	315.55	1214570	23.17	0	18048
	4.55	720805	739	980466	1189260	456.88	1169650	16.00	0	17928
	2.97	720478	984	980599	1214490	213.05	1214490	19.18	0	17736
	3.71	684124	716	923049	1143560	480.70	1121270	17.62	0	17712
	3.27	703107	848	993829	1434470	348.74	1232620	19.34	0	17832
	3.43	714616	917	934928	1201400	279.19	1201400	22.15	0	17568
	3.22	745050	609	1017830	1218960	587.96	1218960	16.45	18	17376
	4.15	693586	681	966547	1325480	514.89	1251160	22.71	0	17856
	3.02	718966	931	974856	1117820	266.08	1117820	12.74	0	17640
	4.12	727199	965	999051	1280220	230.91	1280220	21.90	0	17784
Average:	**3.63**	**710906**	**827**	**970359**	**1234023**	**369.39**	**1202216**	**19.13**	**2**	**17748**

Table A.2. Experiments with Alternative Model Formulations on Problem Set 2.

Model	LPTime	RlxObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.
IP1	0.18	678396	1200	927031	1134910	0.04	1134910	18.32	0	5760
	0.21	717620	1201	1000470	1265390	0.05	1265390	20.94	0	5760
	0.19	715936	1200	970448	1215730	0.04	1215730	20.18	0	5760
	0.20	684632	1208	967404	1154000	0.04	1154000	16.17	0	5760
	0.20	698281	1210	924268	1106480	0.04	1106480	16.47	0	5760
	0.20	713263	1207	975484	1195520	0.04	1195520	18.40	0	5760
	0.20	743252	1213	1047380	1264460	0.01	1264460	17.17	0	5760
	0.21	694447	1201	1053140	1282460	0.03	1282460	17.88	0	5760
	0.21	717974	1210	987691	1200660	0.04	1200660	17.74	0	5760
	0.19	722978	1200	957788	1179610	0.04	1179610	18.80	0	5760
Average:	**0.20**	**708678**	**1205**	**981110**	**1199922**	**0.04**	**1199922**	**18.21**	**0**	**5760**
IP2	1.03	678396	1199	841574	1139830	0.10	1139830	26.17	0	5760
	1.08	717620	1199	922692	1300580	0.01	1300580	29.06	0	5760
	1.13	715936	1199	901864	1185950	0.01	1185950	23.95	0	5760
	1.01	684632	1199	889979	1233300	0.10	1233300	27.84	0	5760
	1.11	698281	1199	861111	1119840	0.01	1119840	23.10	0	5760
	1.03	713263	1199	896286	1205380	0.01	1205380	25.64	0	5760
	1.08	743252	1199	949876	1276160	0.01	1276160	25.57	0	5760
	1.22	694447	1199	970287	1324830	0.18	1324830	26.76	0	5760
	1.21	717974	1199	908060	1204160	0.10	1204160	24.59	0	5760
	1.00	722978	1200	894061	1158460	0.10	1158460	22.82	0	5760
Average:	**1.09**	**708678**	**1199**	**903579**	**1214849**	**0.06**	**1214849**	**25.55**	**0**	**5760**
ELS1	0.60	678396	1208	912464	1189740	0.05	1189740	23.31	0	6720
	0.64	717620	1204	982812	1379840	0.05	1379840	28.77	0	6720
	0.66	715936	1200	945164	1280730	0.05	1280730	26.20	0	6720
	0.60	684632	1204	951406	1403800	0.05	1403800	32.23	0	6720
	0.68	698281	1203	908355	1160360	0.05	1160360	21.72	0	6720
	0.64	713263	1208	952866	1342280	0.05	1342280	29.01	0	6720
	0.67	743252	1206	1020940	1506850	0.05	1506850	32.25	0	6720
	0.68	694447	1203	1011280	1399280	0.05	1399280	27.73	0	6720
	0.63	717974	1204	962156	1293740	0.05	1293740	25.63	0	6720
	0.60	722978	1208	937928	1209450	0.05	1209450	22.45	0	6720
Average:	**0.64**	**708678**	**1205**	**958537**	**1316607**	**0.05**	**1316607**	**26.93**	**0**	**6720**
ELS2	13.83	678396	1186	907996	1494200	0.06	1494200	39.23	0	25416
	12.11	717620	1188	975855	1810650	0.05	1810650	46.10	0	26040
	15.12	715936	1185	962741	1461490	0.06	1461490	34.13	0	25728
	11.98	684632	1188	958218	1681740	0.06	1681740	43.02	0	24744
	12.70	698281	1187	919254	1397340	0.06	1397340	34.21	0	25032
	11.19	713263	1189	944941	1544470	0.06	1544470	38.82	0	24912
	12.69	743252	1187	1055690	1652950	0.05	1652950	36.13	0	24936
	12.65	694447	1187	1043870	1786240	0.06	1786240	41.56	0	24840
	13.14	717974	1187	980488	1571640	0.06	1571640	37.61	0	25248
	10.19	722978	1190	948824	1361760	0.05	1361760	30.32	0	23784
Average:	**12.56**	**708678**	**1188**	**969788**	**1576248**	**0.06**	**1576248**	**38.11**	**0**	**25068**

Table A.3. Experiments with Alternative Model Formulations on Problem Set 3.

Model	LPTime	RlxObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.
IP1	0.16	1015710	1201	1384010	1572380	0.05	1572380	11.98	0	8640
	0.15	1074590	1201	1466470	1637540	0.01	1637540	10.45	0	8640
	0.11	1074700	1201	1468070	1666610	0.05	1666610	11.91	0	8640
	0.15	1020520	1200	1382100	1532400	0.02	1532400	9.81	0	8640
	0.12	1048560	1201	1494060	1696470	0.05	1696470	11.93	0	8640
	0.12	1066830	1200	1392990	1587880	0.01	1587880	12.27	0	8640
	0.15	1111240	1201	1527300	1696350	0.05	1696350	9.97	0	8640
	0.12	1034230	1200	1435900	1673680	0.05	1673680	14.21	0	8640
	0.16	1072020	1135	1457580	1595190	65.41	1595190	8.63	0	8640
	0.16	1084200	1200	1501260	1660170	0.05	1660170	9.57	0	8640
Average:	**0.14**	**1060260**	**1194**	**1450974**	**1631867**	**6.58**	**1631867**	**11.07**	**0**	**8640**
IP2	1.32	1015710	1199	1261120	1588250	0.13	1588250	20.60	0	8640
	1.21	1074590	1199	1356660	1684260	0.12	1684260	19.45	0	8640
	0.96	1074700	1199	1321010	1663610	0.13	1663610	20.59	0	8640
	1.11	1020520	1199	1272840	1555360	0.14	1555360	18.16	0	8640
	0.99	1048560	1199	1333630	1755460	0.13	1755460	24.03	0	8640
	1.30	1066830	1199	1265130	1585480	0.02	1585480	20.21	0	8640
	1.04	1111240	1199	1381300	1708670	0.13	1708670	19.16	0	8640
	1.09	1034230	1199	1241740	1671980	0.13	1671980	25.73	0	8640
	0.95	1072020	1199	1349430	1645490	0.14	1645490	17.99	0	8640
	1.35	1084200	1199	1354760	1732400	0.13	1732400	21.80	0	8640
Average:	**1.13**	**1060260**	**1199**	**1313762**	**1659096**	**0.12**	**1659096**	**20.77**	**0**	**8640**
ELS1	0.73	1015710	1200	1356870	1611360	0.07	1611360	15.79	0	10080
	0.70	1074590	1200	1447440	1699500	0.07	1699500	14.83	0	10080
	0.40	1074700	1200	1444880	1656390	0.07	1656390	12.77	0	10080
	0.56	1020520	1200	1363080	1630240	0.06	1630240	16.39	0	10080
	0.52	1048560	1200	1477660	1754900	0.07	1754900	15.80	0	10080
	0.41	1066830	1200	1378510	1667550	0.06	1667550	17.33	0	10080
	0.55	1111240	1200	1503460	1722300	0.07	1722300	12.71	0	10080
	0.44	1034230	1200	1416520	1690250	0.02	1690250	16.19	0	10080
	0.44	1072020	1200	1438520	1647800	0.07	1647800	12.70	0	10080
	0.54	1084200	1200	1482010	1713930	0.02	1713930	13.53	0	10080
Average:	**0.53**	**1060260**	**1200**	**1430895**	**1679422**	**0.06**	**1679422**	**14.80**	**0**	**10080**
ELS2	11.58	1015710	1189	1370400	1868360	0.10	1868360	26.65	0	27072
	14.65	1074590	1186	1435170	1922250	0.08	1922250	25.34	0	26892
	7.15	1074700	1193	1429090	1937790	0.10	1937790	26.25	0	26604
	10.24	1020520	1190	1365190	1852130	0.08	1852130	26.29	0	26568
	9.41	1048560	1191	1462710	1990820	0.09	1990820	26.53	0	26748
	8.62	1066830	1192	1381040	1785990	0.10	1785990	22.67	0	26352
	9.83	1111240	1190	1513160	1928180	0.08	1928180	21.52	0	26064
	12.12	1034230	1188	1436720	1949300	0.08	1949300	26.30	0	26784
	8.38	1072020	1192	1438780	1867520	0.09	1867520	22.96	0	26460
	13.64	1084200	1187	1451090	1926970	0.10	1926970	24.70	0	26676
Average:	**10.56**	**1060260**	**1190**	**1428335**	**1902931**	**0.09**	**1902931**	**24.92**	**0**	**26622**

Table A.4. Experiments with Alternative Model Formulations on Problem Set 4.

Model	LPTime	RlxObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.
IP1	0.23	1013030	1200	1362200	1755740	0.05	1755740	22.41	0	8640
	0.24	1071350	1200	1456570	2036840	0.04	2036840	28.49	0	8640
	0.24	1070080	1200	1429260	1846890	0.05	1846890	22.61	0	8640
	0.22	1021380	1202	1425320	1866690	0.05	1866690	23.64	0	8640
	0.24	1043030	1205	1357700	1678810	0.04	1678810	19.13	0	8640
	0.24	1065530	1202	1433450	1937950	0.04	1937950	26.03	0	8640
	0.25	1109840	1207	1548340	1956390	0.04	1956390	20.86	0	8640
	0.25	1035490	1200	1552980	2025670	0.03	2025670	23.33	0	8640
	0.24	1071520	1207	1451690	1872780	0.05	1872780	22.48	0	8640
	0.22	1079610	1205	1408240	1721350	0.01	1721350	18.19	0	8640
Average:	**0.24**	**1058086**	**1203**	**1442575**	**1869911**	**0.04**	**1869911**	**22.72**	**0**	**8640**
IP2	2.85	1013030	1197	1177380	1878480	0.15	1878480	37.32	0	8640
	2.42	1071350	1198	1251070	2052380	0.15	2052380	39.04	0	8640
	2.37	1070080	1198	1258550	1955890	0.16	1955890	35.65	0	8640
	2.98	1021380	1197	1209130	1965870	0.01	1965870	38.49	0	8640
	3.24	1043030	1197	1196380	1774580	0.15	1774580	32.58	0	8640
	2.11	1065530	1198	1232250	2049440	0.15	2049440	39.87	0	8640
	3.06	1109840	1197	1298210	2093190	0.14	2093190	37.98	0	8640
	3.34	1035490	1197	1305250	2166510	0.15	2166510	39.75	0	8640
	3.32	1071520	1197	1271550	2159570	0.15	2159570	41.12	0	8640
	2.69	1079610	1198	1258470	1943160	0.16	1943160	35.24	0	8640
Average:	**2.84**	**1058086**	**1197**	**1245824**	**2003907**	**0.14**	**2003907**	**37.71**	**0**	**8640**
ELS1	1.75	1013030	1202	1338600	1926900	0.02	1926900	30.53	0	10080
	1.75	1071350	1201	1440220	2111300	0.07	2111300	31.79	0	10080
	1.75	1070080	1199	1386030	2052460	0.01	2052460	32.47	0	10080
	1.66	1021380	1201	1389700	2154000	0.02	2154000	35.48	0	10080
	1.80	1043030	1201	1330000	1863970	0.06	1863970	28.65	0	10080
	1.79	1065530	1202	1389920	2164150	0.07	2164150	35.78	0	10080
	1.91	1109840	1200	1482350	2286250	0.06	2286250	35.16	0	10080
	1.79	1035490	1200	1464860	2224290	0.07	2224290	34.14	0	10080
	1.76	1071520	1201	1402220	2046480	0.07	2046480	31.48	0	10080
	1.72	1079610	1201	1385200	1866300	0.07	1866300	25.78	0	10080
Average:	**1.77**	**1058086**	**1201**	**1400910**	**2069610**	**0.05**	**2069610**	**32.13**	**0**	**10080**
ELS2	40.67	1013030	1160	1185090	2329430	0.09	2329430	49.13	0	38124
	39.21	1071350	1161	1303270	2766600	0.09	2766600	52.89	0	39060
	42.88	1070080	1157	1261190	2390750	0.09	2390750	47.25	0	38592
	35.92	1021380	1164	1243290	2624370	0.09	2624370	52.63	0	37116
	36.09	1043030	1164	1247690	2197900	0.09	2197900	43.23	0	37548
	39.52	1065530	1161	1288230	2394400	0.10	2394400	46.20	0	37368
	37.78	1109840	1163	1434530	2582040	0.09	2582040	44.44	0	37404
	38.94	1035490	1161	1360130	2936010	0.09	2936010	53.67	0	37260
	50.35	1071520	1150	1292270	2533000	0.09	2533000	48.98	0	37872
	30.60	1079610	1170	1318520	2127170	0.11	2127170	38.02	0	35676
Average:	**39.20**	**1058086**	**1161**	**1293421**	**2488167**	**0.09**	**2488167**	**47.64**	**0**	**37602**

A.2. Experiments on Valid Inequalities

Table A.5. Experiments with Valid Inequalities on Problem Set 1.

Model	LPTime	RlxObj	InitialRlxObj	Increase %	BestFeas	Gap%	NConstr.	Ncuts	Ncallback	CutTime
Basic Model	45.01	838098	681132	23.04	1046300	24.84	5760	11914.00	24	0.18
	17.30	888942	720805	23.33	1075630	21.00	5760	11898.00	24	0.19
	122.40	903520	720478	25.41	1098030	21.53	5760	11882.00	24	0.18
	21.04	834357	684124	21.96	1014380	21.58	5760	11890.00	24	0.18
	57.86	908306	703107	29.18	1107260	21.90	5760	12012.00	25	0.19
	96.99	869050	714616	21.61	1028580	18.36	5760	11913.00	24	0.20
	27.22	920539	745050	23.55	1112740	20.88	5760	11909.00	24	0.18
	55.72	876403	693586	26.36	1082400	23.50	5760	11971.00	24	0.19
	35.61	891207	718966	23.96	1066190	19.63	5760	11904.00	24	0.18
	18.26	899563	727199	23.70	1093660	21.58	5760	11952.00	24	0.18
Average:	**49.74**	**882999**	**710906**	**24**	**1072517**	**21.48**	**5760**	**11924.50**	**24**	**0.19**
Basic Model with z Variables	107.68	836290	681132	22.78	1046300	25.11	19008	12129.00	24	0.16
	30.90	888916	720805	23.32	1075630	21.00	18888	12018.00	24	0.17
	149.94	903106	720478	25.35	1098030	21.58	18696	12700.00	25	0.17
	41.46	834357	684124	21.96	1014380	21.58	18672	11977.00	24	0.16
	80.59	907456	703107	29.06	1107260	22.02	18792	12821.00	25	0.17
	189.43	868635	714616	21.55	1028580	18.41	18528	12846.00	24	0.16
	55.05	919970	745050	23.48	1112740	20.95	18336	12512.00	24	0.16
	84.27	876148	693586	26.32	1082400	23.54	18816	12461.00	24	0.17
	66.78	890403	718966	23.84	1066190	19.74	18600	12306.00	24	0.16
	41.23	899280	727199	23.66	1093660	21.62	18744	12494.00	24	0.17
Average:	**84.73**	**882456**	**710906**	**24**	**1072517**	**21.56**	**18708**	**12426.40**	**24**	**0.16**
ELS1	63.22	837863	681132	23.01	1046300	24.88	6720	11942.00	24	0.19
	14.55	888970	720805	23.33	1075630	21.00	6720	11886.00	24	0.21
	107.32	903599	720478	25.42	1098030	21.52	6720	11878.00	24	0.20
	24.26	834357	684124	21.96	1014380	21.58	6720	11890.00	24	0.20
	49.27	908344	703107	29.19	1107260	21.90	6720	12017.00	25	0.22
	109.02	869004	714616	21.60	1028580	18.36	6720	11914.00	24	0.20
	29.55	920539	745050	23.55	1112740	20.88	6720	11909.00	24	0.21
	57.23	876537	693586	26.38	1082400	23.49	6720	11945.00	24	0.21
	34.36	891204	718966	23.96	1066190	19.63	6720	11910.00	24	0.20
	16.53	899574	727199	23.70	1093660	21.58	6720	11953.00	24	0.20
Average:	**50.53**	**882999**	**710906**	**24**	**1072517**	**21.48**	**6720**	**11924.40**	**24**	**0.20**
ELS1 with z Variables	60.24	838291	681132	23.07	1046300	24.81	19968	11926.00	24	0.12
	15.61	888970	720805	23.33	1075630	21.00	19848	11885.00	24	0.12
	168.57	903835	720478	25.45	1098030	21.49	19656	11921.00	24	0.11
	20.54	834484	684124	21.98	1014380	21.56	19632	11915.00	24	0.12
	59.07	908363	703107	29.19	1107260	21.90	19752	12009.00	25	0.12
	147.75	868880	714616	21.59	1028580	18.38	19488	11772.00	24	0.11
	40.20	920477	745050	23.55	1112740	20.89	19296	11925.00	24	0.11
	45.59	876529	693586	26.38	1082400	23.49	19776	11946.00	24	0.13
	40.51	891174	718966	23.95	1066190	19.64	19560	11922.00	24	0.12
	21.07	899622	727199	23.71	1093660	21.57	19704	11947.00	24	0.12
Average:	**61.91**	**883063**	**710906**	**24**	**1072517**	**21.47**	**19668**	**11916.80**	**24**	**0.12**

Table A.6. Experiments with Valid Inequalities on Problem Set 2.

Model	LPTime	RlxObj	InitialRlxObj	Increase %	BestFeas	Gap%	NConstr.	Ncuts	Ncallback	CutTime
IP1	66.33	838199	678396	23.56	1134910	35.40	5760	11972	25	0.29
	102.07	888130	717620	23.76	1265390	42.48	5760	11960	25	0.31
	152.59	862404	715936	20.46	1185950	37.52	5760	11921	24	0.30
	176.80	872606	684632	27.46	1154000	32.25	5760	11985	25	0.28
	69.51	836861	698281	19.85	1106480	32.22	5760	11966	24	0.28
	116.35	877321	713263	23.00	1195520	36.27	5760	11935	24	0.27
	111.87	915738	743252	23.21	1264460	38.08	5760	11967	24	0.28
	103.72	895111	694447	28.90	1282460	43.27	5760	11913	25	0.28
	228.65	862356	717974	20.11	1200660	39.23	5760	11961	24	0.28
	54.37	862690	722978	19.32	1158460	34.28	5760	11960	24	0.25
Average:	**118.22**	**871142**	**708678**	**22.96**	**1194829**	**37.10**	**5760**	**11954**	**24.4**	**0.28**
IP1 with z Variables	136.56	814688	678396	20.09	1134910	39.31	26376	14984	22	0.24
	235.36	879219	717620	22.52	1265390	43.92	27000	13964	24	0.24
	240.86	850418	715936	18.78	1185950	39.45	26688	14790	23	0.24
	243.10	862261	684632	25.95	1154000	33.83	25704	14768	24	0.24
	176.01	826372	698281	18.34	1106480	33.90	25992	14869	24	0.24
	175.77	868267	713263	21.73	1195520	37.69	25872	15288	24	0.24
	173.04	905524	743252	21.83	1264460	39.64	25896	15069	24	0.24
	159.94	884774	694447	27.41	1282460	44.95	25800	14329	25	0.24
	428.75	856282	717974	19.26	1200660	40.22	26208	14869	24	0.24
	123.26	858181	722978	18.70	1158460	34.99	24744	13274	24	0.22
Average:	**209.27**	**860599**	**708678**	**21.46**	**1194829**	**38.79**	**26028**	**14620**	**23.8**	**0.24**
ELS1	49.23	838289	678396	23.57	1134910	35.38	6720	11971	25	0.26
	88.70	888130	717620	23.76	1265390	42.48	6720	11960	25	0.25
	114.64	862412	715936	20.46	1185950	37.52	6720	11923	24	0.27
	165.26	872606	684632	27.46	1154000	32.25	6720	11985	25	0.27
	58.77	836868	698281	19.85	1106480	32.22	6720	11966	24	0.27
	95.77	877321	713263	23.00	1195520	36.27	6720	11935	24	0.25
	101.50	915793	743252	23.21	1264460	38.07	6720	11965	24	0.26
	92.22	895111	694447	28.90	1282460	43.27	6720	11913	25	0.25
	176.29	862223	717974	20.09	1200660	39.25	6720	11982	24	0.25
	52.13	862690	722978	19.32	1158460	34.28	6720	11960	24	0.24
Average:	**99.45**	**871144**	**708678**	**22.96**	**1194829**	**37.10**	**6720**	**11956**	**24.4**	**0.26**
ELS1 with z Variables	50.81	838481	678396	23.60	1134910	35.35	27336	11977	25	0.10
	76.84	888129	717620	23.76	1265390	42.48	27960	11963	25	0.10
	114.13	862436	715936	20.46	1185950	37.51	27648	11922	24	0.10
	205.70	872631	684632	27.46	1154000	32.24	26664	11986	25	0.10
	47.29	837162	698281	19.89	1106480	32.17	26952	11970	24	0.10
	92.76	877327	713263	23.00	1195520	36.27	26832	11936	24	0.10
	128.84	915816	743252	23.22	1264460	38.07	26856	11957	24	0.10
	72.56	895111	694447	28.90	1282460	43.27	26760	11913	25	0.10
	129.52	862258	717974	20.10	1200660	39.25	27168	11960	24	0.10
	43.06	862946	722978	19.36	1158460	34.24	25704	11963	24	0.10
Average:	**96.15**	**871230**	**708678**	**22.97**	**1194829**	**37.09**	**26988**	**11955**	**24.4**	**0.10**

Table A.7. Experiments with Valid Inequalities on Problem Set 3.

Model	LPTime	RlxObj	InitialRlxObj	Increase %	BestFeas	Gap%	NConstr.	Ncuts	Ncallback	CutTime
IP1	305.11	1242880	1015710	22.37	1572380	26.51	8640	26292	33	0.532597
	67.76	1322590	1074590	23.08	1637540	23.81	8640	26403	34	0.507972
	556.89	1344790	1074700	25.13	1656390	23.17	8640	26416	34	0.497241
	85.39	1231080	1020520	20.63	1532400	24.48	8640	26128	32	0.492788
	256.92	1361260	1048560	29.82	1696470	24.62	8640	26665	36	0.514914
	470.99	1276990	1066830	19.70	1585480	24.16	8640	25975	31	0.472169
	153.34	1363790	1111240	22.73	1696350	24.38	8640	26186	33	0.485766
	280.60	1311070	1034230	26.77	1671980	27.53	8640	26359	35	0.510488
	142.96	1320700	1072020	23.20	1595190	20.78	8640	26393	33	0.495737
	85.80	1339020	1084200	23.50	1660170	23.98	8640	26470	34	0.508449
Average:	**240.58**	**1311417**	**1060260**	**23.69**	**1630435**	**24.34**	**8640**	**26329**	**33.5**	**0.50**
IP1 with z Variables	462.37	1230730	1015710	21.17	1572380	27.76	28512	28015	32	0.451916
	150.75	1315690	1074590	22.44	1637540	24.46	28332	26874	33	0.449759
	828.81	1334790	1074700	24.20	1656390	24.09	28044	29962	33	0.473944
	175.68	1231090	1020520	20.63	1532400	24.48	28008	27137	32	0.438652
	520.62	1358030	1048560	29.51	1696470	24.92	28188	30213	36	0.50139
	859.81	1265410	1066830	18.61	1585480	25.29	27792	28103	29	0.419298
	278.99	1335700	1111240	20.20	1696350	27.00	27504	28013	30	0.433132
	525.80	1308130	1034230	26.48	1671980	27.81	28224	29627	35	0.490183
	280.79	1326740	1072020	23.76	1595190	20.23	27900	28886	34	0.467341
	152.43	1326170	1084200	22.32	1660170	25.19	28116	28602	33	0.449289
Average:	**423.60**	**1303248**	**1060260**	**22.93**	**1630435**	**25.12**	**28062**	**28543**	**32.7**	**0.46**
ELS1	359.26	1242220	1015710	22.30	1572380	26.58	10080	26291	33	0.770678
	79.62	1322680	1074590	23.09	1637540	23.80	10080	26396	34	0.79557
	637.97	1345120	1074700	25.16	1656390	23.14	10080	26293	34	0.744016
	105.14	1231080	1020520	20.63	1532400	24.48	10080	26128	32	0.651806
	269.15	1361400	1048560	29.84	1696470	24.61	10080	26691	36	0.676656
	557.07	1276950	1066830	19.70	1585480	24.16	10080	25983	31	0.622346
	163.54	1364130	1111240	22.76	1696350	24.35	10080	26126	33	0.65476
	266.89	1310470	1034230	26.71	1671980	27.59	10080	26352	35	0.663467
	195.69	1327070	1072020	23.79	1595190	20.20	10080	26492	34	0.658291
	94.81	1339020	1084200	23.50	1660170	23.98	10080	26470	34	0.666427
Average:	**272.91**	**1312014**	**1060260**	**23.75**	**1630435**	**24.29**	**10080**	**26322**	**33.6**	**0.69**
ELS1 with z Variables	494.63	1243120	1015710	22.39	1572380	26.49	29952	26164	33	0.437462
	56.94	1322620	1074590	23.08	1637540	23.81	29772	26469	34	0.433637
	1226.06	1345200	1074700	25.17	1656390	23.13	29484	26196	34	0.430822
	93.90	1231080	1020520	20.63	1532400	24.48	29448	26203	32	0.416461
	453.52	1361440	1048560	29.84	1696470	24.61	29628	26740	36	0.454057
	1497.34	1277950	1066830	19.79	1585480	24.06	29232	25892	31	0.407294
	137.71	1363750	1111240	22.72	1696350	24.39	28944	26224	33	0.427128
	380.01	1310540	1034230	26.72	1671980	27.58	29664	26466	35	0.444504
	197.26	1320650	1072020	23.19	1595190	20.79	29340	26389	33	0.423744
	73.18	1339020	1084200	23.50	1660170	23.98	29556	26503	34	0.430019
Average:	**461.05**	**1311537**	**1060260**	**23.70**	**1630435**	**24.33**	**29502**	**26325**	**33.5**	**0.43**

Table A.8. Experiments with Valid Inequalities on Problem Set 4.

Model	LPTime	RlxObj	InitialRlxObj	Increase %	BestFeas	Gap%	NConstr.	Ncuts	Ncallback	CutTime
IP1	287.06	1246830	1013030	23.08	1755740	40.82	8640	26486	34	0.82
	645.44	1320700	1071350	23.27	2036840	54.22	8640	26491	34	0.82
	691.60	1247990	1070080	16.63	1846890	47.99	8640	25129	28	0.73
	1193.72	1307870	1021380	28.05	1866690	42.73	8640	26622	36	0.78
	166.19	1201380	1043030	15.18	1678810	39.74	8640	24427	26	0.68
	632.84	1305220	1065530	22.49	1937950	48.48	8640	26429	34	0.77
	520.90	1362070	1109840	22.73	1956390	43.63	8640	26484	34	0.77
	426.68	1341250	1035490	29.53	2025670	51.03	8640	26515	36	0.78
	972.43	1243660	1071520	16.07	1872780	50.59	8640	24854	27	0.69
	95.81	1214620	1079610	12.51	1721350	41.72	8640	22418	22	0.57
Average:	**563.27**	**1279159**	**1058086**	**20.95**	**1869911**	**46.09**	**8640**	**25586**	**31.1**	**0.74**
IP1 with z Variables	351.77	1159410	1013030	14.45	1755740	51.43	39564	26813	23	0.63
	821.78	1288390	1071350	20.26	2036840	58.09	40500	29442	31	0.74
	15.13	1094930	1070080	2.32	1846890	68.68	40032	5615	4	0.11
	1137.27	1290650	1021380	26.36	1866690	44.63	38556	32765	35	0.76
	7.94	1060630	1043030	1.69	1678810	58.28	38988	4246	3	0.08
	773.13	1276200	1065530	19.77	1937950	51.85	38808	31941	31	0.75
	800.81	1336390	1109840	20.41	1956390	46.39	38844	32627	32	0.74
	1105.86	1327370	1035490	28.19	2025670	52.61	38700	31990	36	0.76
	605.20	1201160	1071520	12.10	1872780	55.91	39312	24364	21	0.55
	63.50	1151510	1079610	6.66	1721350	49.49	37116	15009	12	0.30
Average:	**568.24**	**1218664**	**1058086**	**15.22**	**1869911**	**53.74**	**39042**	**23481**	**22.8**	**0.54**
ELS1	363.76	1246830	1013030	23.08	1755740	40.82	10080	26486	34	0.81
	420.14	1320720	1071350	23.28	2036840	54.22	10080	26489	34	0.83
	498.64	1247870	1070080	16.61	1846890	48.00	10080	25135	28	0.74
	1344.07	1307870	1021380	28.05	1866690	42.73	10080	26622	36	0.82
	129.18	1195430	1043030	14.61	1678810	40.44	10080	23989	25	0.68
	496.20	1305220	1065530	22.49	1937950	48.48	10080	26429	34	0.81
	578.06	1362070	1109840	22.73	1956390	43.63	10080	26483	34	0.80
	357.25	1341250	1035490	29.53	2025670	51.03	10080	26515	36	0.80
	586.04	1243490	1071520	16.05	1872780	50.61	10080	24849	27	0.73
	84.90	1214620	1079610	12.51	1721350	41.72	10080	22418	22	0.60
Average:	**485.82**	**1278537**	**1058086**	**20.89**	**1869911**	**46.17**	**10080**	**25542**	**31.0**	**0.76**
ELS1 with z Variables	405.16	1246920	1013030	23.09	1755740	40.81	41004	26488	34	0.43
	519.96	1320720	1071350	23.28	2036840	54.22	41940	26489	34	0.43
	737.44	1241830	1070080	16.05	1846890	48.72	41472	24769	27	0.37
	2019.43	1303460	1021380	27.62	1866690	43.21	39996	26591	35	0.43
	165.25	1201400	1043030	15.18	1678810	39.74	40428	24435	26	0.36
	717.45	1305220	1065530	22.49	1937950	48.48	40248	26429	34	0.44
	810.68	1362010	1109840	22.72	1956390	43.64	40284	26502	34	0.43
	774.92	1341360	1035490	29.54	2025670	51.02	40140	26516	36	0.44
	702.08	1237500	1071520	15.49	1872780	51.34	40752	24447	26	0.36
	78.24	1214620	1079610	12.51	1721350	41.72	38556	22418	22	0.30
Average:	**693.06**	**1277504**	**1058086**	**20.80**	**1869911**	**46.29**	**40482**	**25508**	**30.8**	**0.40**

A.3. Experiments on Heuristics

Table A.9. Experiments with Heuristics on Problem Set 1.

Model	LPTime	RlxObj	InitHeurTime	InitHeurObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.	Ncuts	Ncallback	NHeurSoln	BestHeur
IP1 + Cuts	0.16	678396	-	-	1215	922721.00	1105040	0.01	1105040	16.50	0	5760	1786	13	-	-
	0.19	717620	-	-	1200	1002670.00	1274290	0.03	1274290	21.32	0	5760	1663	17	-	-
	0.18	715936	-	-	1200	971112.00	1211550	0.01	1211550	19.85	0	5760	1794	19	-	-
	0.17	684632	-	-	1206	965750.00	1187890	0.04	1187890	18.70	0	5760	2043	12	-	-
	0.19	698281	-	-	1200	923827.00	1116440	0.01	1116440	17.25	0	5760	1865	13	-	-
	0.19	713263	-	-	1200	976702.00	1223010	0.04	1223010	20.14	0	5760	1788	15	-	-
	0.19	743252	-	-	1213	1052560.00	1259590	0.04	1259590	16.44	0	5760	1940	15	-	-
	0.19	694447	-	-	1202	1054060.00	1283320	0.04	1283320	17.87	0	5760	1737	25	-	-
	0.19	717974	-	-	1210	988402.00	1183780	0.01	1183780	16.50	0	5760	2077	27	-	-
	0.18	722978	-	-	1200	954396.00	1165430	0.03	1165430	18.11	0	5760	1648	18	-	-
Average:	**0.18**	**708678**	**-**	**-**	**1204.58**	**981220**	**1201034**	**0.03**	**1201034**	**18.27**	**0.00**	**5760**	**1834.10**	**17**	**-**	**-**
IP1 + Init Heur	0.16	678396	0.005	1366170	1200	920913.00	1063640	0.03	1063640	13.42	0	5760	-	-	-	-
	0.19	717620	0.004	1626040	1204	997423.00	1208980	0.04	1208980	17.50	0	5760	-	-	-	-
	0.19	715936	0.005	1651370	1200	972643.00	1181090	0.03	1181090	17.65	0	5760	-	-	-	-
	0.18	684632	0.006	1493070	1206	970876.00	1151330	0.04	1151330	15.67	0	5760	-	-	-	-
	0.20	698281	0.005	1512950	1211	924952.00	1082780	0.04	1082780	14.58	0	5760	-	-	-	-
	0.19	713263	0.005	1880150	1200	973848.00	1160530	0.03	1160530	16.09	0	5760	-	-	-	-
	0.19	743252	0.005	1648120	1214	1053450.00	1230730	0.03	1230730	14.40	0	5760	-	-	-	-
	0.21	694447	0.004	2312640	1113	1054080.00	1273430	86.75	1273120	17.19	0	5760	-	-	-	-
	0.19	717974	0.004	1925160	1011	988485.00	1161910	188.84	1159100	14.68	0	5760	-	-	-	-
	0.18	722978	0.005	1327410	1200	955826.00	1065690	0.01	1065690	10.31	0	5760	-	-	-	-
Average:	**0.19**	**708678**	**0.005**	**1674308**	**1176.05**	**981250**	**1158011**	**27.58**	**1157699**	**15.15**	**0.00**	**5760**	**-**	**-**	**-**	**-**
IP1 +Cuts + Init Heur	0.16	678396	0.005	1366170	1200	925664.00	1060020	0.01	1060020	12.68	0	5760	1606	15	-	-
	0.19	717620	0.004	1626040	1204	997008.00	1205370	0.04	1205370	17.29	0	5760	1706	15	-	-
	0.18	715936	0.004	1651370	1200	969558.00	1171120	0.04	1171120	17.21	0	5760	1820	17	-	-
	0.18	684632	0.006	1493070	1207	963281.00	1145520	0.03	1145520	15.91	0	5760	1963	12	-	-
	0.19	698281	0.005	1512950	1200	925502.00	1085660	0.04	1085660	14.75	0	5760	2014	16	-	-
	0.19	713263	0.005	1880150	1200	975211.00	1164680	0.03	1164680	16.27	0	5760	1927	14	-	-
	0.19	743252	0.005	1648120	1214	1047340.00	1267350	0.01	1267350	17.36	0	5760	2094	14	-	-
	0.20	694447	0.004	2312640	1200	1052900.00	1262970	0.04	1262970	16.63	0	5760	1809	21	-	-
	0.19	717974	0.004	1925160	1210	987511.00	1172870	0.04	1172870	15.80	0	5760	1951	21	-	-
	0.18	722978	0.005	1327410	1200	956806.00	1069360	0.03	1069360	10.53	0	5760	1558	21	-	-
Average:	**0.18**	**708678**	**0.005**	**1674308**	**1203.64**	**980078**	**1160492**	**0.03**	**1160492**	**15.44**	**0.00**	**5760**	**1844.80**	**17**	**-**	**-**
IP1 + Cuts + Init Heur + WWCP	0.17	678396	0.006	1366170	1200	921240.00	1062990	0.04	1062990	13.34	0	5760	1903	13	520	17800979
	0.20	717620	0.004	1626040	1206	994618.00	1216820	0.04	1216820	18.26	0	5760	1947	14	560	10772245
	0.19	715936	0.005	1651370	1209	967616.00	1153180	0.04	1153180	16.09	0	5760	2179	15	600	15190746
	0.19	684632	0.005	1493070	1201	964499.00	1149660	0.04	1149660	16.11	0	5760	2279	12	480	14159661
	0.19	698281	0.004	1512950	1200	924822.00	1089150	0.04	1089150	15.09	0	5760	2247	13	520	15878926
	0.19	713263	0.005	1880150	1200	973771.00	1156840	0.04	1156840	15.82	0	5760	2105	12	480	14234884
	0.20	743252	0.005	1648120	1215	1045600.00	1257890	0.04	1257890	16.88	0	5760	2558	13	520	18789992
	0.21	694447	0.004	2312640	1200	1051540.00	1264360	0.04	1264360	16.83	0	5760	2129	18	720	12987575
	0.20	717974	0.004	1925160	1210	988240.00	1159490	0.05	1159490	14.77	0	5760	2442	22	880	17988366
	0.18	722978	0.005	1327410	1200	955476.00	1075590	0.01	1075590	11.17	0	5760	1852	20	800	6729107
Average:	**0.19**	**708678**	**0.005**	**1674308**	**1204.02**	**978742**	**1158597**	**0.04**	**1158597**	**15.44**	**0.00**	**5760**	**2164.10**	**15**	**608**	**14453248**

Table A.10. Experiments with Heuristics on Problem Set 2.

Model	LPTime	RlxObj	InitHeurTime	InitHeurObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.	Ncuts	Ncallback	NHeurSoln	BestHeur
IP1 + Cuts	0.10	681132	-	-	700	929285.00	1038730	499.53	1036300	10.25	113	5760	2434	235	-	-
	0.10	720805	-	-	661	983260.00	1076580	539.05	1076010	8.58	165	5760	1960	332	-	-
	0.09	720478	-	-	371	980492.00	1107830	829.38	1105450	11.23	544	5760	3026	1018	-	-
	0.10	684124	-	-	394	921918.00	1017510	806.41	1017510	9.37	634	5760	2493	1175	-	-
	0.08	703107	-	-	357	997636.00	1109260	843.35	1107670	9.91	640	5760	2826	1179	-	-
	0.08	714616	-	-	601	935414.00	1032610	599.37	1030640	9.20	146	5760	2439	301	-	-
	0.09	745050	-	-	341	1019090.00	1112920	859.44	1112220	8.32	965	5760	3007	1841	-	-
	0.08	693586	-	-	750	966742.00	1071690	450.41	1069040	9.56	90	5760	2705	198	-	-
	0.10	718966	-	-	352	974594.00	1077720	848.08	1068440	8.71	866	5760	2816	1607	-	-
	0.11	727199	-	-	477	1003320.00	1090220	723.52	1089640	7.89	329	5760	2781	628	-	-
Average:	**0.09**	**710906**	**-**	**-**	**500.16**	**971175**	**1073507**	**699.85**	**1071292**	**9.30**	**449.20**	**5760**	**2648.70**	**851**	**-**	**-**
IP1 + Init Heur	0.10	681132	0.004	1188530	583	929563.00	1021970	616.93	1020850	8.88	274	5760	-	-	-	-
	0.10	720805	0.004	1273700	546	983047.00	1086960	653.55	1082400	9.14	279	5760	-	-	-	-
	0.08	720478	0.004	1217610	399	980394.00	1105140	801.08	1104320	11.17	536	5760	-	-	-	-
	0.10	684124	0.004	1277560	397	921912.00	996398	802.96	995024	7.34	809	5760	-	-	-	-
	0.07	703107	0.004	1471510	325	996613.00	1102790	875.40	1101700	9.48	770	5760	-	-	-	-
	0.09	714616	0.004	1157960	815	934973.00	1031440	384.99	1029890	9.15	43	5760	-	-	-	-
	0.09	745050	0.004	1312870	393	1019160.00	1102040	806.82	1102040	7.47	1042	5760	-	-	-	-
	0.07	693586	0.003	1214330	736	966150.00	1078320	464.38	1076440	10.20	98	5760	-	-	-	-
	0.10	718966	0.004	1281680	377	974046.00	1060520	823.18	1059340	8.02	1063	5760	-	-	-	-
	0.11	727199	0.004	1310360	402	1002660.00	1094510	798.35	1093990	8.28	574	5760	-	-	-	-
Average:	**0.09**	**710906**	**0.004**	**1270611**	**497.24**	**970852**	**1068009**	**702.76**	**1066599**	**8.91**	**548.80**	**5760**	**-**	**-**	**-**	**-**
IP1 +Cuts + Init Heur	0.10	681132	0.003	1188530	564	929448.00	1018080	636.18	1017590	8.60	307	5760	2720	617	-	-
	0.10	720805	0.004	1273700	677	982728.00	1076980	523.43	1074870	8.47	127	5760	1886	266	-	-
	0.09	720478	0.003	1217610	447	980795.00	1092220	753.44	1092220	10.17	438	5760	3073	836	-	-
	0.10	684124	0.003	1277560	362	921856.00	1003090	838.44	1000030	7.79	756	5760	2720	1404	-	-
	0.07	703107	0.003	1471510	418	997852.00	1098380	781.70	1096110	8.95	567	5760	2616	1094	-	-
	0.08	714616	0.004	1157960	689	935401.00	1040320	510.99	1035530	9.63	120	5760	2474	254	-	-
	0.10	745050	0.004	1312870	388	1019310.00	1104880	812.05	1102250	7.49	740	5760	2808	1434	-	-
	0.07	693586	0.003	1214330	832	966761.00	1066610	368.04	1064650	9.18	0	5760	3061	38	-	-
	0.10	718966	0.004	1281680	416	974064.00	1061320	784.34	1058080	7.87	783	5760	2968	1455	-	-
	0.10	727199	0.004	1310360	547	1003510.00	1094740	652.85	1094440	8.26	385	5760	2359	742	-	-
Average:	**0.09**	**710906**	**0.004**	**1270611**	**533.85**	**971173**	**1065662**	**666.15**	**1063577**	**8.64**	**422.30**	**5760**	**2668.50**	**814**	**-**	**-**
IP1 + Cuts + Init Heur + WWCP	0.10	681132	0.004	1188530	745	929851.00	1010630	456.95	1010560	7.93	108	5760	3144	201	8040	9928806
	0.10	720805	0.004	1273700	624	983236.00	1076720	576.12	1075740	8.56	147	5760	2231	279	11160	6237196
	0.08	720478	0.004	1217610	506	980743.00	1096450	695.12	1095340	10.41	269	5760	3601	451	18040	4338908
	0.11	684124	0.004	1277560	461	921989.00	996070	741.00	994297	7.25	298	5760	2818	507	20280	5535321
	0.08	703107	0.003	1471510	427	997978.00	1097510	774.58	1097510	9.05	398	5760	3327	698	27920	3996360
	0.09	714616	0.004	1157960	960	935562.00	1032750	239.89	1032750	9.38	0	5760	2749	37	1480	7279785
	0.10	745050	0.003	1312870	443	1019610.00	1100210	759.17	1100010	7.27	533	5760	3255	978	39120	8478951
	0.08	693586	0.004	1214330	914	966620.00	1073590	286.11	1071910	9.81	0	5760	3265	37	1480	6744058
	0.10	718966	0.003	1281680	417	973794.00	1065170	784.57	1065170	8.45	418	5760	3301	704	28160	8752852
	0.11	727199	0.003	1310360	540	1003420.00	1094990	659.86	1094630	8.29	254	5760	2926	454	18160	7061627
Average:	**0.09**	**710906**	**0.004**	**1270611**	**603.63**	**971280**	**1064409**	**597.34**	**1063792**	**8.64**	**242.50**	**5760**	**3061.70**	**435**	**17384**	**6835386**

Table A.11. Experiments with Heuristics on Problem Set 3.

Model	LPTime	RlxObj	InitHeurTime	InitHeurObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.	Ncuts	Ncallback	NHeurSoln	BestHeur
IP1 + Cuts	0.15	1015710	-	-	1200	1378400.00	1571180	0.05	1571180	12.27	0	8640	3950	11	-	-
	0.14	1074590	-	-	1200	1465370.00	1636250	0.04	1636250	10.44	0	8640	2804	12	-	-
	0.11	1074700	-	-	1201	1465730.00	1646190	0.05	1646190	10.96	0	8640	4347	11	-	-
	0.14	1020520	-	-	1200	1380390.00	1527530	0.05	1527530	9.63	0	8640	3940	14	-	-
	0.11	1048560	-	-	1200	1492760.00	1702790	0.05	1702790	12.33	0	8640	4153	14	-	-
	0.11	1066830	-	-	1200	1390260.00	1581970	0.05	1581970	12.12	0	8640	3875	11	-	-
	0.13	1111240	-	-	1200	1525530.00	1681410	0.05	1681410	9.27	0	8640	4234	14	-	-
	0.12	1034230	-	-	1200	1434090.00	1638300	0.05	1638300	12.46	0	8640	4558	11	-	-
	0.15	1072020	-	-	1200	1458310.00	1599940	0.05	1599940	8.85	0	8640	3840	18	-	-
	0.15	1084200	-	-	1200	1497510.00	1660330	0.01	1660330	9.81	0	8640	4311	13	-	-
Average:	**0.13**	**1060260**	**-**	**-**	**1200.43**	**1448835**	**1624589**	**0.04**	**1624589**	**10.82**	**0.00**	**8640**	**4001.20**	**13**	**-**	**-**
IP1 + Init Heur	0.15	1015710	0.006	1813350	1200	1382910.00	1545170	0.04	1545170	10.50	0	8640	-	-	-	-
	0.14	1074590	0.005	1905270	1201	1467020.00	1621860	0.04	1621860	9.55	0	8640	-	-	-	-
	0.11	1074700	0.005	1823570	1201	1466920.00	1653380	0.04	1653380	11.28	0	8640	-	-	-	-
	0.15	1020520	0.005	1988720	1200	1381730.00	1485580	0.05	1485580	6.99	0	8640	-	-	-	-
	0.11	1048560	0.004	2292140	1201	1493820.00	1667200	0.04	1667200	10.40	0	8640	-	-	-	-
	0.12	1066830	0.006	1730110	1200	1393450.00	1569420	0.04	1569420	11.21	0	8640	-	-	-	-
	0.14	1111240	0.006	1984280	1200	1525600.00	1666150	0.01	1666150	8.44	0	8640	-	-	-	-
	0.12	1034230	0.005	1832140	1201	1437450.00	1604310	0.04	1604310	10.40	0	8640	-	-	-	-
	0.15	1072020	0.005	1948320	1192	1457540.00	1608180	8.10	1608180	9.37	0	8640	-	-	-	-
	0.15	1084200	0.005	1992680	1200	1500870.00	1657710	0.05	1657710	9.46	0	8640	-	-	-	-
Average:	**0.13**	**1060260**	**0.005**	**1931058**	**1199.61**	**1450731**	**1607896**	**0.85**	**1607896**	**9.76**	**0.00**	**8640**	**-**	**-**	**-**	**-**
IP1 +Cuts + Init Heur	0.15	1015710	0.005	1813350	1200	1376550.00	1538180	0.05	1538180	10.51	0	8640	4365	11	-	-
	0.14	1074590	0.006	1905270	1201	1464510.00	1631470	0.05	1631470	10.23	0	8640	3150	13	-	-
	0.12	1074700	0.006	1823570	1200	1466030.00	1646410	0.05	1646410	10.96	0	8640	4605	13	-	-
	0.14	1020520	0.005	1988720	1200	1379990.00	1501880	0.05	1501880	8.12	0	8640	3399	15	-	-
	0.12	1048560	0.004	2292140	1201	1493760.00	1652450	0.05	1652450	9.60	0	8640	4171	15	-	-
	0.12	1066830	0.006	1730110	1201	1387690.00	1563790	0.05	1563790	11.26	0	8640	3966	11	-	-
	0.14	1111240	0.005	1984280	1200	1525320.00	1672760	0.02	1672760	8.81	0	8640	4366	16	-	-
	0.11	1034230	0.005	1832140	1201	1429350.00	1606690	0.05	1606690	11.04	0	8640	4160	10	-	-
	0.15	1072020	0.005	1948320	1201	1458790.00	1612750	0.04	1612750	9.55	0	8640	3975	20	-	-
	0.15	1084200	0.005	1992680	1200	1498720.00	1657170	0.05	1657170	9.56	0	8640	4073	13	-	-
Average:	**0.13**	**1060260**	**0.005**	**1931058**	**1200.47**	**1448071**	**1608355**	**0.05**	**1608355**	**9.96**	**0.00**	**8640**	**4023.00**	**14**	**-**	**-**
IP1 + Cuts + Init Heur + WWCP	0.15	1015710	0.005	1813350	1200	1377310.00	1539450	0.06	1539450	10.53	0	8640	4636	11	440	35120231
	0.15	1074590	0.005	1905270	1200	1462840.00	1636810	0.01	1636810	10.63	0	8640	3388	12	480	26000665
	0.12	1074700	0.006	1823570	1201	1465110.00	1662190	0.05	1662190	11.86	0	8640	5059	12	480	16379134
	0.15	1020520	0.005	1988720	1203	1379800.00	1490030	0.06	1490030	7.40	0	8640	3852	14	560	14271942
	0.12	1048560	0.005	2292140	1200	1492090.00	1659350	0.05	1659350	10.08	0	8640	4884	14	560	14743228
	0.12	1066830	0.005	1730110	1200	1390430.00	1561540	0.01	1561540	10.96	0	8640	5004	11	440	17336680
	0.14	1111240	0.006	1984280	1200	1523850.00	1666200	0.05	1666200	8.54	0	8640	4901	13	520	18492435
	0.12	1034230	0.005	1832140	1200	1429970.00	1619380	0.06	1619380	11.70	0	8640	4749	11	440	17282380
	0.16	1072020	0.006	1948320	1200	1456810.00	1604620	0.05	1604620	9.21	0	8640	4599	14	560	18756523
	0.15	1084200	0.005	1992680	1200	1498210.00	1660700	0.05	1660700	9.78	0	8640	4576	12	480	20196762
Average:	**0.14**	**1060260**	**0.005**	**1931058**	**1200.66**	**1447642**	**1610027**	**0.05**	**1610027**	**10.07**	**0.00**	**8640**	**4564.80**	**12**	**496**	**19857998**

Table A.12. Experiments with Heuristics on Problem Set 4.

Model	LPTime	RlxObj	InitHeurTime	InitHeurObj	Node0Time	Node0LB	Node0UB	B&BTime	ObjVal	% Gap	Nnodes	Nconstr.	Ncuts	Ncallback	NHeurSoln	BestHeur
IP1 + Cuts	0.23	1013030	-	-	1200	1343210.00	1836080	0.01	1836080	26.84	0	8640	3243	7	-	-
	0.24	1071350	-	-	1200	1455450.00	2014390	0.05	2014390	27.75	0	8640	2748	8	-	-
	0.24	1070080	-	-	1209	1408410.00	1830790	0.05	1830790	23.07	0	8640	2987	7	-	-
	0.22	1021380	-	-	1204	1408030.00	1848370	0.05	1848370	23.82	0	8640	3252	7	-	-
	0.24	1043030	-	-	1200	1350480.00	1745350	0.04	1745350	22.62	0	8640	2946	6	-	-
	0.24	1065530	-	-	1204	1423460.00	1821200	0.04	1821200	21.84	0	8640	3013	6	-	-
	0.24	1109840	-	-	1200	1519460.00	2040650	0.05	2040650	25.54	0	8640	2941	6	-	-
	0.24	1035490	-	-	1200	1536960.00	1976630	0.05	1976630	22.24	0	8640	2709	7	-	-
	0.24	1071520	-	-	1206	1436430.00	1938010	0.01	1938010	25.88	0	8640	3213	7	-	-
	0.22	1079610	-	-	1207	1396030.00	1780400	0.05	1780400	21.59	0	8640	2617	8	-	-
Average:	**0.23**	**1058086**	**-**	**-**	**1202.98**	**1427792**	**1883187**	**0.04**	**1883187**	**24.12**	**0.00**	**8640**	**2966.90**	**7**	**-**	**-**
IP1 + Init Heur	0.23	1013030	0.007	2054180	1205	1360220.00	1595060	0.04	1595060	14.72	0	8640	-	-	-	-
	0.24	1071350	0.006	2526780	1203	1464020.00	1880400	0.04	1880400	22.14	0	8640	-	-	-	-
	0.24	1070080	0.007	2569540	1200	1421220.00	1797690	0.04	1797690	20.94	0	8640	-	-	-	-
	0.23	1021380	0.008	2271620	1205	1419230.00	1747330	0.04	1747330	18.78	0	8640	-	-	-	-
	0.24	1043030	0.007	2311850	1200	1363890.00	1626290	0.04	1626290	16.13	0	8640	-	-	-	-
	0.24	1065530	0.007	3058240	1204	1430900.00	1781280	0.05	1781280	19.67	0	8640	-	-	-	-
	0.23	1109840	0.007	2658840	1200	1537150.00	1931360	0.04	1931360	20.41	0	8640	-	-	-	-
	0.24	1035490	0.007	3777990	1207	1543920.00	1936540	0.04	1936540	20.27	0	8640	-	-	-	-
	0.24	1071520	0.006	3101580	1206	1450060.00	1797890	0.04	1797890	19.35	0	8640	-	-	-	-
	0.22	1079610	0.008	2024040	1200	1410370.00	1635670	0.01	1635670	13.77	0	8640	-	-	-	-
Average:	**0.24**	**1058086**	**0.007**	**2635466**	**1203.04**	**1440098**	**1772951**	**0.04**	**1772951**	**18.62**	**0.00**	**8640**	**-**	**-**	**-**	**-**
IP1 +Cuts + Init Heur	0.23	1013030	0.008	2054180	1200	1345120.00	1598860	0.05	1598860	15.87	0	8640	3025	7	-	-
	0.24	1071350	0.007	2526780	1203	1456710.00	1869000	0.04	1869000	22.06	0	8640	2633	8	-	-
	0.23	1070080	0.006	2569540	1207	1403720.00	1781110	0.05	1781110	21.19	0	8640	3066	7	-	-
	0.22	1021380	0.008	2271620	1204	1410260.00	1731390	0.05	1731390	18.55	0	8640	3140	7	-	-
	0.23	1043030	0.006	2311850	1202	1348780.00	1673770	0.05	1673770	19.42	0	8640	3220	7	-	-
	0.25	1065530	0.007	3058240	1204	1422320.00	1756730	0.05	1756730	19.04	0	8640	2864	7	-	-
	0.24	1109840	0.006	2658840	1200	1529540.00	1925350	0.05	1925350	20.56	0	8640	3439	7	-	-
	0.24	1035490	0.007	3777990	1207	1535390.00	1922120	0.04	1922120	20.12	0	8640	3169	7	-	-
	0.24	1071520	0.006	3101580	1206	1445550.00	1839060	0.05	1839060	21.40	0	8640	3278	8	-	-
	0.21	1079610	0.007	2024040	1208	1406860.00	1620150	0.01	1620150	13.16	0	8640	2747	9	-	-
Average:	**0.23**	**1058086**	**0.007**	**2635466**	**1204.05**	**1430425**	**1771754**	**0.04**	**1771754**	**19.14**	**0.00**	**8640**	**3058.10**	**7**	**-**	**-**
IP1 + Cuts + Init Heur + WWCP	0.23	1013030	0.008	2054180	1204	1335060.00	1595440	0.06	1595440	16.32	0	8640	2875	6	240	48032001
	0.25	1071350	0.007	2526780	1202	1447800.00	1878460	0.05	1878460	22.93	0	8640	3025	7	280	34775320
	0.24	1070080	0.007	2569540	1200	1405460.00	1800940	0.05	1800940	21.96	0	8640	3456	7	280	38996072
	0.24	1021380	0.007	2271620	1207	1410400.00	1735380	0.09	1735380	18.73	0	8640	3445	7	280	35803933
	0.25	1043030	0.007	2311850	1205	1363980.00	1645160	0.01	1645160	17.09	0	8640	3435	6	240	34837369
	0.25	1065530	0.006	3058240	1204	1412310.00	1766480	0.05	1766480	20.05	0	8640	2974	6	240	36488728
	0.25	1109840	0.007	2658840	1200	1525680.00	1906740	0.05	1906740	19.98	0	8640	3612	6	240	37231539
	0.26	1035490	0.006	3777990	1207	1512840.00	1935940	0.06	1935940	21.86	0	8640	3411	6	240	33192703
	0.25	1071520	0.006	3101580	1200	1437510.00	1838280	0.05	1838280	21.80	0	8640	3626	7	280	40824243
	0.22	1079610	0.008	2024040	1200	1401550.00	1634180	0.05	1634180	14.23	0	8640	3176	9	360	26033450
Average:	**0.24**	**1058086**	**0.007**	**2635466**	**1202.87**	**1425259**	**1773700**	**0.05**	**1773700**	**19.50**	**0.00**	**8640**	**3303.50**	**7**	**268**	**36621536**