Composite-Variable Modeling for Large-Scale Problems in Transportation and Logistics

by

Amy Ellen Mainville Cohn

A.B., Harvard University (1991)

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2002

© Massachusetts Institute of Technology 2002

Signature of Author .....
Sloan School of Management
9 May 2002

Certified by .....
Cynthia Barnhart
Professor, Civil and Environmental Engineering
Co-Director, Center for Transportation and Logistics
Thesis Supervisor

Accepted by .....
James B. Orlin
Edward Pennell Brooks Professor of Operations Research
Co-director, Operations Research Center

Composite-Variable Modeling for Large-Scale Problems in Transportation and Logistics

by

Amy Ellen Mainville Cohn

Submitted to the Sloan School of Management
on 9 May 2002, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Operations Research

Abstract

Numerous important real-world problems are found in the areas of transportation and logistics. Many of these problems pose tremendous challenges due to characteristics such as complex networks, tightly constrained resources, and very large numbers of heavily inter-connected decisions. As a result, mathematical models can be critical in solving these problems. These models, however, can be computationally challenging or even intractable. In this thesis we discuss how greater tractability can sometimes be achieved with *composite-variable models* – models in which individual binary variables encompass multiple decisions.

In Part I, we discuss common challenges found in solving large-scale transportation and logistics problems. We introduce the idea of composite variables and discuss the potential benefits of composite-variable models. We also note some of the drawbacks of these models and discuss approaches to addressing these drawbacks. In Parts II and III, we demonstrate these ideas using two real-world examples, one from airline planning and the other from service parts logistics. We build on our experience from these two applications in Part IV, providing some broader insights for composite-variable modeling. We focus in particular on the dominance property seen in the service parts logistics example and on the fact that we can relax the integrality of the composite variables in the airline planning example. In both cases, we introduce broader classes of problems in which these properties can also be found. We offer conclusions in Part V.

The contributions of the thesis are three-fold. First, we provide a new model and solution approach for an important real-world problem from the airline industry. Second, we provide a framework for addressing challenging problems in service parts logistics. Third, we provide insights into how to construct composite-variable models for greater tractability. These insights can be useful not only in solving large-scale problems, but also in integrating multiple stages within a planning environment, developing better heuristics for solving large problems in real time, and providing users with greater control in trading off solution time and quality.

Thesis Supervisor: Cynthia Barnhart
Title: Professor, Civil and Environmental Engineering
Co-Director, Center for Transportation and Logistics

Contents

I An Introduction To Composite-Variable Modeling 13

1 Introduction 15

1.1 Thesis Overview 15

1.2 Challenges In Transportation And Logistics 16

1.2.1 Complex Networks 17

1.2.2 Tightly Constrained Resources 18

1.2.3 Large Numbers Of Heavily Inter-Connected Decisions 18

1.2.4 Complex Feasibility Rules And Objective Functions 20

1.3 Mathematical Modeling For Transportation And Logistics: Benefits And Difficulties 20

1.3.1 Non-Linearities 21

1.3.2 Large Numbers Of Constraints 21

1.3.3 Large Numbers Of Integer Variables 21

1.3.4 Weak Linear Programming Relaxations 22

1.4 An Introduction To Composite-Variable Modeling 23

1.4.1 Deriving Composite-Variable Models From Basic Formulations 24

1.4.2 Benefits And Drawbacks Of Composite Re-Formulations 26

II Improving Crew Scheduling By Incorporating Key Maintenance Routing Decisions: The Extended Crew Pairing Model 29

2 Introduction 31

3 Problem Definition **34**

3.1 The Airline Planning Process 34

3.2 The Crew Pairing Problem 37

3.2.1 The Set Partitioning Formulation 37

3.3 The Maintenance Routing Problem 38

3.4 The Link Between Crew Pairing And Maintenance Routing 41

3.5 An Example 42

3.6 Integrated Approaches 43

4 The Extended Crew Pairing Model **45**

4.1 The ECP Model 45

4.2 Reducing Problem Size Through Variable Elimination 48

4.2.1 Uniqueness 48

4.2.2 Maximal Sets 50

4.2.3 Theoretical Bounds And Computational Results 51

4.2.4 Identifying UM Columns 54

4.3 Relaxing The Integrality Of The Maintenance Variables 55

4.3.1 Quality Of The LP Relaxation 60

5 Implementing ECP **70**

5.1 Generating Columns 71

5.1.1 Generating Crew Pairings 71

5.1.2 Generating Maintenance Routing Solution Columns 72

5.1.3 When To Generate Columns 74

5.2 Generating Lower Bounds 76

5.2.1 Generating Minimally Infeasible Short Connect Sets 78

5.3 Permitting Crews To Fly Multiple Sub-Fleets 81

5.4 Proof-of-Concept 83

6 Conclusions **85**

III Composite-Variable Modeling For Service Parts Logistics **87**

7 Introduction **89**

8 The High-Cost, Low-Demand Stocking Problem **91**

9 Extending The Problem Scope **94**

9.1 The Capacitated High-Cost, Low-Demand Stocking Problem 94

9.2 A Basic Modeling Approach 94

9.3 Computational Experience 95

10 A Composite-Variable Modeling Approach **98**

10.1 Benefits Of The Composite Model 99

10.2 Controlling The Number Of Variables In *CHCLD-C* 100

10.2.1 Implementing Branch-and-Price 102

11 Solving Additional Problems In *SPL* **105**

IV Insights For Composite-Variable Modeling **109**

12 Composite-Variable Definition **111**

12.1 Integrated Airline Planning Example 111

12.2 Service Parts Logistics Example 112

12.3 An Alternative SPL Formulation 113

12.4 Trading Off Size And Complexity In Defining Composites 116

13 Strengthening The LP Relaxation **118**

14 Dominance **123**

15 Relaxing The Integrality Of The Composite Variables **127**

16 Conclusions For Part IV **131**

V Conclusions And Suggestions For Further Research 133

List of Figures

3-1 Airline planning process 35
3-2 Sequential approach 42
3-3 Example data 42

4-1 Extended model with both maintenance solutions 47
4-2 Example 49

5-1 Model synergy 81

8-1 Potential stocking locations 92
8-2 Feasible covering 92

13-1 Instance in which warehouse formulation has tighter LP relaxation 121
13-2 Instance in which SKU formulation has tighter LP relaxation 122

List of Tables

4.1 Number of UMI forced turn sets 54

5.1 Proof-of-concept 83

9.1 Results for CHCLD-B 97

10.1 Results for CHCLD-C 101

Acknowledgments

During my time at MIT, I have been fortunate to have the support of a great many individuals and institutions. These acknowledgments can only begin to demonstrate my appreciation.

First, I would like to acknowledge the financial support of Draper Laboratories, the National Science Foundation, United Airlines, and the Global Airline Industry Program. I would also like to acknowledge the assistance of Keith Howells and Juwono Bong of UPS.

In addition, I would like to thank those who have guided me through the administrative maze of MIT, including Danielle Colarusso, Veronica Mignott, and especially Paulette Mosley, who has finally agreed to release me from my orientation responsibilities. I am particularly grateful to Maria Marangiello for not only taking care of me but my son as well on our many trips across campus, and to Laura Rose who has been a part of so many milestones during my time at MIT.

My research experience has been a joy, largely due to the opportunity to learn, laugh, and most importantly eat with the Large-Scale Optimization Group, including Manoj Lohatepanont, Tim Kniker, Michael Clarke, and Chris Nielsen. I have also benefitted from the ongoing advice and guidance of Steve Kolitz, and the research diversions of George Polak.

The OR community at MIT is tremendous, with supportive classmates and faculty too numerous to mention. Special thanks go to my officemates and comrades in the job search – Fernando Ordonez, Alp Muharremoglu, John Bossert, and Dessi Pachamanova. I would also like to acknowledge Andy Armacost for providing not only research advice but parenting tips and my very own MIT "Mommy Group." Jeremie Gallien has been an ongoing source of support and good cheer since our first days as graduate students. I cannot imagine life at MIT without the continual friendship of Ozlem Ergun. Amr Farahat merits high praise not only for providing me with countless new research ideas, but for often doing so while pushing a baby stroller.

I have also been very fortunate to work over the past two years with a fantastic research partner, Wilson Tandiono. Our collaboration has not only been very productive, with much of the burden of modeling, coding, and writing shared between us, but also lots of fun! I hope that we will have the chance to work together again some day.

I also owe a great debt to my committee members, Professors Jim Orlin and Yossi Sheffi. Their dissertation advice is just the final stage in many, many years of guidance and insight.

It truly takes a village to raise a child, and my work would not have been possible without all those who pitched in to help care for my son Tommy. These include the many families of Eastgate who have formed such a nurturing community, and especially our wonderful sitters, Maria Larcade, Dianne Wheeler, and Isabel Esquitin. I would also like to thank Peter Beinart and the staff of The New Republic for making it possible for my husband to be in Boston over these past years. Another invaluable resource has been gradmoms@mit.edu, and especially Glaire Anderson. Further special thanks go to Monica Lafond and Ken Truesdale for still being around whenever I had the chance to reappear after long disappearances.

Without question, my biggest source of support has been my extended family. In particular, my parents have helped in countless ways, from pep talks to baby-sitting to home cooked meals. I am eternally grateful for their love, support, and encouragement.

I cannot possibly express my gratitude for the support of my advisor, Professor Cynthia Barnhart. She has not only advised my research, but provided me with a role model on how to be a good teacher, a good parent, and a good person, and it has been a great honor not only to have worked with her, but to have known her. I leave her with great sadness but also with anticipation of exciting future collaborations.

Finally, I dedicate this work, and all that I do, to my husband and son. Jonathan and Tommy have made every day a joy, every success all the sweeter, and every setback less daunting. Our life together as a family will always be my greatest accomplishment.

Part I

An Introduction To Composite-Variable Modeling

Chapter 1

Introduction

Numerous important real-world problems are found in the areas of transportation and logistics. These problems are often characterized by large networks spanning both space and time, and by tightly constrained resources such as raw materials, production capacity, time, and manpower. As a result, decision makers typically face a very large number of decisions that are heavily inter-connected and, therefore, must be considered simultaneously. Complex feasibility rules and objective functions can make these problems even more challenging.

Mathematical models can therefore be critical for making optimal decisions and sometimes even for finding feasible solutions. Conventional models are often computationally challenging, however, due to complicating factors such as non-linearities, large numbers of constraints, and weak linear programming relaxations.

In this thesis, we discuss how the use of alternative models based on *composite variables* – that is, binary variables that encompass multiple decisions – can sometimes achieve greater tractability for such transportation and logistics problems.

1.1 Thesis Overview

In Part I of the thesis, we note some of the challenges commonly encountered in addressing transportation and logistics problems. We then consider the importance of mathematical models in solving such problems, and recognize factors that can lead to intractability within these models. We next introduce the idea of a composite variable. We discuss potential benefits of

composite-variable models, then discuss their drawbacks and how we might address them.

In Part II, we demonstrate these ideas using an example from airline planning. Specifically, we look at integrating two key planning problems, crew scheduling and aircraft routing. A direct integration of the existing models for these two problems is inflexible and in many instances intractable. We instead pose a composite-variable model that enables us to improve model tractability and the potential for further extensions to the problem definition by leveraging the fact that only some of the aircraft routing information is relevant to crew schedulers.

We provide a second example in Part III, in which we use composite-variable modeling to build a framework for solving service parts logistics problems. By representing SKU stocking policies with composite variables, we can capture interactions between SKUs while taking advantage of the breadth of literature on how to stock individual SKUs.

In these two composite models, we observe a number of common characteristics that enhance tractability. These include tighter linear programming relaxations, the presence of *dominance* to limit the number of variables that must be included in the models, and structural properties that can be exploited for greater tractability. We also observe that these properties are a function of the variable definition – that is, which decisions we have grouped together to form the composites – and that variable definition is perhaps the most important component of composite-variable modeling. We build on these observations in Part IV, presenting insights for other composite-variable modelers. We focus in particular on the dominance property seen in the service parts logistics example and on the fact that we can relax the integrality of the composite variables in the airline planning example. In both cases, we introduce broader classes of problems in which these properties can be found.

We conclude the thesis in Part V with a review of our contributions and a discussion of areas for further research.

1.2 Challenges In Transportation And Logistics

Transportation and logistics play important roles in today's global economy. For example, the Mid-Atlantic Transportation and Logistics Institute estimates that transportation and logistics costs make up approximately twenty percent of the United States GNP ([51]). Efficient

transportation and logistics planning is critical for profitability and competitive advantage in industries ranging from passenger airlines to breakfast cereal manufacturers to PC producers. The proliferation of technology for real-time tracking as well as the volume of consumer information captured on the internet has resulted in a wealth of data that in theory could lead to significantly better decision making. In reality, this explosion in data, along with the growth of network complexity, makes it nearly impossible for planners to make quality decisions without the assistance of sophisticated tools.

In this section, we highlight some of the complexities associated with transportation and logistics decisions that motivate the need for mathematical modeling.

1.2.1 Complex Networks

Underlying most transportation and logistics problems are complex networks that encompass elements of both space and time. The following are just a few examples.

- Manufacturing networks contain nodes corresponding to suppliers, manufacturing facilities (and, in fact, multiple stages within the manufacturing process), and one or more echelons within the distribution network. The arcs of the network correspond to the flow of raw materials from suppliers, partially finished products through the manufacturing process, completed products to warehouses, retailers, and consumers, and replacement parts to customers needing repairs.

- Networks for passenger airlines are based on nodes that represent airports at specific points in time (typically corresponding to take-offs and landings) and arcs that represent flights. The flow of aircraft, flight crews, and passengers must all be controlled simultaneously over such a network, which spans a time period ranging from 24 hours to three months or more, and encompasses thousands of flights per day.

- In less-than-truckload freight carrier networks, nodes represent many different types of facilities at many different points in time. These facilities include *end-of-line terminals* where freight originates and terminates, *breakbulk facilities* where freight is consolidated, and *relay stations*, used in the movement of loaded trailers. The flow across such networks

includes not only the freight itself, but the flow of individual drivers, trailers, and tractors, each of which can take separate paths through the system.

It is not uncommon for transportation and logistics networks to have hundreds of nodes and thousands of arcs, span multiple time zones, and cross international borders. Furthermore, the number of commodities flowing across these networks can be enormous – in many cases, in the tens of thousands or more. Thus, the sheer size and complexity of the networks alone can make decision making quite challenging.

1.2.2 Tightly Constrained Resources

Many transportation and logistics problems are also characterized by constrained resources. Examples include limitations on:

- the number of aircraft in a passenger or freight network,
- the amount of capacity on a transportation arc,
- the amount of time allowed between a customer's order and its delivery,
- the production levels during a particular manufacturing shift,

and

- the quantity of goods that can be stored in a warehouse.

1.2.3 Large Numbers Of Heavily Inter-Connected Decisions

As a result of these characteristics, transportation and logistics problems often contain a very large number of decisions, many of them closely inter-connected.

In the airline fleet assignment problem, for example, it is common to have several thousand flights and ten or more fleet types, resulting in tens of thousands of possible pairings of fleet types to flights. In an express package delivery network, we may have tens of thousands of origin-destination pairs, each of which may have many different possible routes. In a service parts logistics network, we may need to make stocking and transportation decisions for hundreds of thousands of SKUs.

What makes these large problems particularly difficult is the fact that the decisions are all inter-connected, because of both the network structure and the tightly constrained resources. As a result, all of these decisions need to be made simultaneously – solving a large problem by breaking it down into smaller sequential sub-problems will typically result in a sub-optimal solution and, in many cases, even infeasibility.

Consider the following examples.

- In many networks, flow through the system must form a *circulation*. This is true of aircraft, truck drivers, railway locomotives, and others. When a pilot lands in a particular airport at a particular time, for example, his or her next flight assignment must depart from the same airport at some subsequent time, prior to any other assignments. Furthermore, the pilot must eventually return to the location from which he or she originated. Thus the simple decision of assigning a pilot to a flight impacts many other decisions throughout the network.

- Warehouse capacity is limited in many problem instances. Deciding to use warehouse capacity for one set of goods limits our ability to assign other goods to that warehouse. As a result, we may make decisions to assign other goods to different warehouses, again creating limitations for subsequent decisions, and so forth. Thus, a single warehouse stocking decision can have impact on all other warehouses in the system. Furthermore, these warehousing decisions might then impact transportation decisions as well.

- Making a decision about which supplier to use for a particular raw material can impact the time at which that material becomes available. Such a decision impacts the scheduling of the production process, the timing and demand for transportation services, the need for capacity within warehouses, and the availability of finished products.

These examples serve to highlight the fact that in many transportation and logistics problems, a single decision can potentially impact virtually every other aspect of the problem. Thus, we often must consider the complete set of decisions simultaneously.

1.2.4 Complex Feasibility Rules And Objective Functions

A further complicating factor is that many transportation and logistics problems have complex feasibility rules and objective functions.

Feasibility rules stem from the physical process itself, from government regulations (such as those designed for worker, consumer, and environmental safety), and from labor negotiations that govern working conditions. One example of these rules is the requirement that passenger aircraft must undergo routine maintenance after a certain number of flying hours. This maintenance often can only occur at certain airports. This rule, then, affects not only the assignment of aircraft to flights, but also the sequence in which these flights occur.

Objective functions are often complex as well. For example:

- Pilots at major U.S. carriers are paid not by yearly salary or even solely as a function of their flying time. Instead, for a given sequence of flights, pilots are paid based on the maximum of three quantities: the flying time itself, some fraction times the total time on duty, and a minimum pay quantity.
- In transportation decisions, costs are often based on some fixed cost plus some per-unit cost.
- Volume discounts often apply when purchasing raw materials.

These complex feasibility rules and objective functions further add to the challenge faced by the decision maker.

1.3 Mathematical Modeling For Transportation And Logistics: Benefits And Difficulties

Mathematical models can be essential in allowing decision makers to take a global view of such complex systems. However, more than one formulation can be used to model most problems; the chosen formulation will have great impact on tractability.

Problems can usually be broken down into a collection of *basic* or *elemental* decisions. These are questions like, "How must freight should I flow over arc A?" or "Should I open a warehouse

at location L?" If we define one variable for each of these decisions, there will often be an intuitive way to then state the constraints and objective function. Such a formulation may be easy to pose and explain, but is not necessarily the most tractable model. The following are a number of factors commonly associated with basic formulations that can greatly hamper their tractability.

1.3.1 Non-Linearities

As we discussed in Section 1.2.4, many transportation and logistics problems contain complex rules and objective functions. As a result, basic formulations often contain non-linear objective functions and/or constraints. For example, the cost associated with flowing freight over an arc in a less-than-truckload freight network is not linear with respect to the quantity of freight, but rather a step function that reflects the integer number of trailers required to carry the freight. Another example is found in airline crew scheduling. If we define variables to represent decisions such as "Should I assign crew c to flight f?" then we cannot capture with linear constraints the rules governing maximum crew flying time between rest periods.

1.3.2 Large Numbers Of Constraints

In order to capture all of the information associated with a complex system, a basic formulation often requires a very large number of constraints. An example of this appears in Part III, where the basic formulation for problem instances of a realistic size can contain more than half a million constraints. Even in a formulation that has linear constraints, a linear objective function, and strictly continuous variables, such large numbers of constraints can be problematic, due to the computational cost of inverting the corresponding matrices.

1.3.3 Large Numbers Of Integer Variables

Transportation and logistics problems often contain questions of the form "Should I...?" (for example, "Should I open warehouse w?") or "How many...?" (for example, "How many rail cars should I assign?"). These questions naturally correspond to integer variables. Thus, in addition to very large numbers of constraints, we also often see very large numbers of integer variables in formulations for transportation and logistics problems. For example, in the airline

fleet assignment problem, we have one binary decision variable for each possible matching of a flight to a fleet type. It is not uncommon for a transportation or logistics formulation to contain tens or even hundreds of thousands of integer variables.

1.3.4 Weak Linear Programming Relaxations

Many basic transportation and logistics formulations are further complicated by very weak linear programming (LP) relaxations. Consider, for example, any of a number of problems in which we determine both how much capacity to provide in our system and how to use this capacity. Such problems include:

- express package delivery problems, where we want to determine what type of aircraft to assign to each flight and how to use these flights to transport packages from their origins to their destinations,
- less-than-truckload problems in which we want to determine where to offer *direct services*, that is, point-to-point trailer movements, and how to flow freight over the resulting network,

  and
- distribution network design problems where we want to determine which warehouses to open and what products to stock in each of these warehouses.

In such cases, the capacity decisions are often either integer (for example, how many trucks to flow over an arc, how many workers to assign to a shift, and so on) or binary (whether or not to open a warehouse, for example, or whether or not to assign a particular aircraft type to a given flight). Independent of whether the decision variables associated with using this capacity are integer or binary, we typically find that the required capacity is rarely an exact match to the assigned capacity. For example, we might route 150 passengers over a particular airline flight. If our choice is to assign to that flight an aircraft that has either 100 (x_{100}) or 200 (x_{200}) seats, the optimal solution to the LP relaxation might be to set $x_{200} = .75$, given that there is no added value in paying for excess capacity.

Such weak linear programming relaxations are common amongst many basic transportation and logistics models. This can lead to very large branch-and-bound trees, resulting in difficulty not only in finding optimal, but in some cases even feasible, solutions.

1.4 An Introduction To Composite-Variable Modeling

The difficulties described in the previous section can sometimes be overcome through *composite-variable modeling*. We use the term *composite variable* to refer to a binary variable that encompasses multiple elemental decisions. For example, we might use a composite variable to represent the collective decisions, "Should we open warehouse w, stock products p_1, p_2, and p_3 there, and use this warehouse to supply these products to customers c_1 and c_2?"

Depending on how we define a composite – that is, which decisions we group together within a single variable – benefits of composite-variable modeling can include:

- removing non-linear objectives and/or constraints,
- strengthening the LP relaxation,

  and
- eliminating blocks of complicating constraints.

In many cases, composite-variable models also have additional structural properties that can be exploited.

The idea of composite-variable modeling is of course not a new one. Successful uses appear in the literature for a wide range of applications. For example, Vanderbeck ([57]) uses a composite-variable formulation to solve the classical cutting stock and cutting strip problems. Armacost et al. ([6]) use a composite-variable formulation to solve an express shipment service network design problem. Almost all of the literature on airline crew scheduling uses composite-variable formulations (see Barnhart et al. [14] for a survey of this literature). These are just a few examples from the literature on composite-variable modeling.

We contribute to this literature in two ways. First, we use composite-variable modeling to develop new formulations for two important real-world problems, one in integrated airline planning and the other in service parts logistics. These problems are discussed in Parts II and

III. Second, we consider in Part IV the broader question of how composite-variable modeling can be used to solve challenging problems in transportation and logistics. We are interested not only in how composite-variable models can used for solving large-scale problems, but also for integrating multiple stages within a planning environment, developing better heuristics for solving large problems in real time, and providing users with greater control in trading off solution time and quality.

1.4.1 Deriving Composite-Variable Models From Basic Formulations

There are many ways in which to construct a composite-variable model. We focus on the broad class of models that are derived from mixed integer programs (*MIPs*). Moreover, we restrict our attention to those models where the decisions grouped together to form a composite are all represented by binary variables in the original *MIP*.

Given an *MIP* and a set x of binary variables to be used to form a composite, we can re-order rows and columns and re-state the constraints, writing the problem in the following form. We refer to this as the *basic model*.

B:

$$\min c_x x + c_y y$$

$$st$$

$$A_x x = b_x$$

$$A_y y = b_y$$

$$D_x x + D_y y = g$$

$$x_i \in \{0,1\} \quad \forall i = 1,2,...,n_x$$

$$y_i \in Y_i \quad \forall i = 1,2,...,n_y.$$

Here, c_x and x are vectors of dimension n_x, c_y and y are vectors of dimension n_y, A_x is a matrix of dimension m_x by n_x, b_x is a vector of dimension m_x, A_y is a matrix of dimension m_y by n_y, b_y is a vector of dimension m_y, D_x and D_y are matrices of dimension m by n_x and m by

n_y respectively, and g is a vector of dimension m. Each element x_i of the vector x is a binary variable. For any variable y_i in y, values are restricted to be in the set Y_i.

Such a model can be re-formulated by forming a *composite* from the variables in x. Note that there are 2^{n_x} possible combinations of values for the n_x binary variables in x. We associate with these combinations the set of composite variables $z = \{z_1, z_2, ...z_{2^{n_x}}\}$. We use s_i to represent the set of elements from x which take value 1 in composite variable z_i. Given this notation, we can re-write the basic formulation B in terms of these new composite variables. If we remove from the set z those elements that do not satisfy the constraints originally written as $A_x x = b_x$, and denote the remaining (feasible) composite variables by $\widehat{z}$, then the composite-variable formulation is written as

C:

$$\min c_{\widehat{z}}\widehat{z} + c_y y$$

$$st$$

$$A_y y = b_y$$

$$D_{\widehat{z}}\widehat{z} + D_y y = g$$

$$e'\widehat{z} = 1$$

$$\widehat{z}_i \in \{0,1\} \quad \forall \widehat{z}_i \in \widehat{z}$$

$$y_i \in Y_i \quad \forall i = 1, 2, ..., n_y,$$

where

$$(c_{\widehat{z}})_i \equiv \sum_{j \in s_i} (c_x)_j \,,$$

$$\left(D_{\widehat{z}}\right)_{ij} \equiv \sum_{k \in s_j} (D_x)_{ik} \,,$$

and e is a vector of dimension $|\widehat{z}|$ comprised of all 1's

1.4.2 Benefits And Drawbacks Of Composite Re-Formulations

Such a re-formulation has the obvious benefit of reducing the number of constraints, completely eliminating those constraints originally stated as $A_x x = b_x$. As we discuss in Part IV, this re-formulation also has an LP relaxation that is always as tight as the LP relaxation of the basic formulation, and in many instances is strictly tighter. Depending on the particular model, there are often other structural benefits as well. Examples of such benefits appear in both of the applications discussed in the following parts. These benefits can have significant impact on tractability.

These benefits come at the expense of an exponential increase in the number of binary variables in the model. We have replaced the set of binary variables x, which is $O(n_x)$, with the set $\widehat{z}$, which is $O(2^{n_x})$. In practice, however, we are often able to control the number of composite variables required to find a provably optimal solution in three ways.

First, observe that we restrict ourselves to the set $\widehat{z}$ rather than the set z. That is, we ignore all composites corresponding to values for the vector x which are infeasible for the constraints

$$A_x x = b_x.$$

In many transportation and logistics problems, by nature of the complex network structure and the tightly constrained resources, we can select the set x such that $\widehat{z}$ is significantly smaller than z, thus greatly reducing the size of the model.

Second, we frequently observe in composite-variable models for transportation and logistics the presence of *dominance*. Variable $\widehat{z}_i$ is *dominated* if for any solution in which $\widehat{z}_i = 1$, there exists another solution of equal or better value in which $\widehat{z}_i = 0$. Dominated variables can thus be disregarded without sacrificing solution quality. In Parts II and III, we see two examples of how dominance can greatly enhance the tractability of a composite-variable model. We discuss broader issues concerning dominance in Part IV.

Third, it is not necessary to enumerate all feasible, non-dominated composite variables explicitly in order to find an optimal solution. We can instead solve a composite-variable model using the *branch-and-price* algorithm (Barnhart et al. [13]), in which we solve large integer programs by using column generation to solve the individual LP relaxations. In order

to use branch-and-price successfully, however, we must construct our composites so as to have an efficient *pricing problem* to identify negative reduced cost columns, as well as a *branching strategy* that is consistent with this pricing problem.

All three of these methods for controlling problem size rely heavily on the way in which we define our composite variables – that is, which binary variables we include within the set x. This is perhaps the most important decision in formulating a composite-variable model. In the next two parts, we use examples to demonstrate these ideas. Part IV discusses these issues for broader classes of problems.

Part II

Improving Crew Scheduling By Incorporating Key Maintenance Routing Decisions: The Extended Crew Pairing Model

Chapter 2

Introduction

The airline planning process is made up of many large and complex problems. Of these, crew scheduling is of particular importance. Crew costs are the second largest operating expense faced by the airlines (after fuel); thus, improving the quality of the crew schedule by even a small amount can have significant financial benefits (Anbil et al. [2], Anbil et al. [3], Butchers et al. [20], Gershkoff [37], and Graves et al. [39]).

A key step in solving the crew scheduling problem is to select a minimum cost set of *crew pairings* – sequences of flights which can be flown by a single crew. This step is known as the *crew pairing problem*. These chosen pairings are then combined to form complete schedules for individual crews. The quality of the crew schedule therefore depends on the set of feasible crew pairings. This feasible set can be impacted significantly by decisions made earlier in the planning process.

For example, one restriction on a valid pairing is that two sequential flights can only be assigned to the same crew if the time between the flights (known as *connection* or *sit time*) is sufficient for the crew members to travel through the terminal, from the arrival gate of one flight to the departure gate of the next. This minimum connection time can be relaxed if both flights have been assigned to the same aircraft, because the crew remains with the aircraft. We use the term *short connect* to refer to a connection which is feasible for a crew only if the two sequential flights comprising that connection have been assigned to a common aircraft.

The assignment of aircraft to flights occurs during the *maintenance routing problem*, which typically precedes crew scheduling in the airline planning process. In the maintenance routing

problem, aircraft are assigned to strings of flights so as to ensure that every aircraft will have adequate opportunity to receive maintenance. This assignment of aircraft to flights determines the set of short connects in the network, thereby impacting the set of feasible pairings permitted in the subsequent crew scheduling problem. Given that the maintenance routing problem does not consider the impact of short connects on the crew scheduling problem, solving the maintenance routing and crew scheduling problems sequentially can result in a sub-optimal solution. The goal of our research is to address this limitation.

Prior works by Klabjan et al. ([44]) and Cordeau et al. ([31]) demonstrate the impact of short connect selection on crew scheduling, and present models for addressing this limitation. Klabjan et al. reverse the order in which they address the maintenance routing and crew pairing problems. They solve the crew pairing problem first, assuming all short connects to be valid. They next solve the maintenance routing problem, in which any short connect used in the crew pairing solution is required to be included in the maintenance routing solution. This approach yields significant improvements for many real-world problem instances. It does not guarantee maintenance feasibility, however, and it is possible to generate non-pathological instances in which the short connects in a crew pairing solution lead to maintenance infeasibility.

Cordeau et al. ([31]) present an integrated approach with approximate crew costs, in which they link maintenance routing and crew pairing models by a set of additional constraints. There is one constraint for each short connect which enforces the rule that a chosen crew pairing can only contain that short connect if a chosen maintenance routing string also contains it. They present a Benders decomposition approach, consisting of a maintenance routing master problem and a crew pairing sub-problem, along with a heuristic branching strategy. They present computational results for a number of problem instances and compare the quality of their heuristic to a more basic approach.

In our research, we have developed the *Extended Crew Pairing Model (ECP)* which further contributes to this literature. We have focused on three key objectives:

- First, we want to guarantee a maintenance-feasible crew pairing solution.
- Second, we want to provide the user with the flexibility to trade off between solution time and quality. When solved completely, our model yields an exact solution to the integrated

maintenance routing and crew pairing problem. Alternatively, our approach can be used heuristically to find quality solutions more quickly, while still guaranteeing maintenance feasibility.

- Third, we want to leverage the fact that only a fraction of the decisions made in the maintenance routing phase impact the crew pairing problem. By including in the Extended Crew Pairing Model only those maintenance decisions which are relevant, we can reduce the size of the model significantly.

The structure of Part II is as follows. In Chapter 3 we review the airline planning process, paying special attention to the crew pairing and maintenance routing problems. We then demonstrate the link between these two problems, presenting a small example to show how a sequential approach can lead to sub-optimality. In Chapter 4 we present the *Extended Crew Pairing Model (ECP)*. We prove theoretical results that allow us to improve the tractability of this model by eliminating a large number of variables and by relaxing the integrality requirement of many others. In Chapter 5 we discuss details associated with using ECP in practice, including column generation techniques and an algorithm for generating lower bounds. We provide our conclusions and future directions in Chapter 6.

Chapter 3

Problem Definition

3.1 The Airline Planning Process

The airline planning process is a large and complex one, incorporating decisions about flights, aircraft, crews, and passengers. Several authors have presented structural overviews of this planning process. They include Barnhart and Talluri [18], Clarke and Smith [26], and Lohatepanont [48]. Implicit within all of these overviews are the following facts:

- Airline planning involves an enormous number of decisions.
- Many of these decisions are tightly inter-related.
- Global optimization is intractable.
- Sequential solution approaches lead to sub-optimality.

For example, we present one such hierarchy in Figure 3-1 (Barnhart and Talluri [18]). This depicts four key sequential phases of the airline planning process.

The *schedule generation problem* determines the set of flights to be offered by the airline during a given time period. This includes selecting markets, frequencies, and specific flight times. Schedule generation is often considered to be the most difficult of the four problems in that there are an infinite number of feasible schedules. Furthermore, it is difficult even to determine the cost of a given schedule, because it depends on the results of all three remaining problems. In addition, the cost of a schedule depends on information, such as passenger demand,

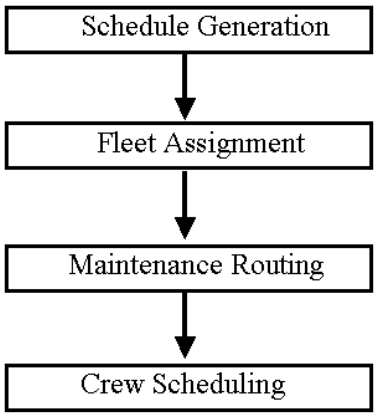


Figure 3-1: Airline planning process

which is not only unknown at the time of the planning process but also varies from day to day over the time period for which the schedule will be in effect (Etschmaier and Mathaisel [36]). Research efforts addressing this problem therefore primarily consist of models that seek to improve an existing schedule. Rexing et al. [53] and Klabjan et al. [44] present models that allow the flight times to vary within a given time window. Barnhart et al. [15] present a model that considers an extended schedule of flights and determines which of those flights to keep and which to discard.

Given a solution to the schedule generation problem, the *fleet assignment problem* then assigns a specific aircraft type to each flight in the schedule. The goal of this problem is to match aircraft capacity to flight demand so as to minimize operating expenses and lost revenue incurred due to insufficient capacity, while maintaining network balance and satisfying constraints on the number of available aircraft of each type. This problem has received much attention in the literature (see, for example, surveys in Barnhart and Talluri [18] and Clarke and Smith [26]), and academic advances have lead to significant real-world savings.

The *maintenance routing problem* constructs a set of *aircraft strings* that enable each aircraft to undergo required routine maintenance on a timely basis while ensuring that every flight is assigned to exactly one aircraft. One maintenance routing problem is solved for each fleet type. We discuss this problem in detail in Section 3.3.

The *crew scheduling problem* is also solved once per fleet type (or, in some cases, once per *fleet family*, a collection of fleet types with the same crew requirements). It assigns cockpit and cabin crews to flights. Crew scheduling is often done in two phases. First, sequences of flights known as *crew pairings* are constructed so that each flight is included in exactly one pairing. Then, schedules for individual crew members are constructed from these pairings to satisfy requirements such as time-off, training, etc. Section 3.2 provides a detailed discussion of the crew pairing problem for cockpit crews. We refer the reader to Barnhart et al. [14], Barnhart and Talluri [18], Clarke and Smith [26], and Desaulniers et al. [35] for detailed discussions of the crew scheduling problem as a whole.

Clearly, sub-optimal results might occur when these problems are solved in a sequential fashion, given their interdependence. For example, the flights chosen during schedule design are the foundation for the three remaining problems. Similarly, the fleet assignment solution partitions the flights, determining the flight networks to be solved in each of the maintenance routing and crew scheduling problems. Nonetheless, a fully integrated approach has yet to be solved, due to the size and complexity of these problems. A number of researchers have instead improved solution quality by integrating portions of the planning model. For example, Rexing et al. [53] incorporate elements of schedule generation into the fleet assignment problem. Barnhart et al. [9] integrate fleet assignment and maintenance routing. Clarke et al. [24] consider maintenance and crew issues in the fleet assignment problem. Barnhart et al. [16] combine elements of fleet assignment and crew scheduling. We contribute to this body of work with an extended approach to crew pairing optimization that incorporates key decisions made in the maintenance routing phase.

In the sections that follow, we describe in greater detail the crew pairing and maintenance routing problems. We then demonstrate the link between them and present a small example to show how solving them sequentially can lead to sub-optimality.

3.2 The Crew Pairing Problem

A fundamental building block of a crew member's schedule is a *duty period*. A duty can be thought of as a day's schedule of flights, to be followed by a period of rest. Duties are subject to a number of restrictions governing feasibility. These are imposed both by government agencies, such as the FAA in the U.S., and by labor agreements within the airline. Some of the most essential include: maximum flying hours per duty, maximum duty lengths, and minimum and maximum "sit" times between consecutive flights. The "cost" of a duty is often expressed by U.S. airlines as the maximum of three components, each measured in minutes. These are the *total flying time* in the duty; some fraction times the *total elapsed time* in the duty; and a *guaranteed minimum quantity*.

Given the concept of a duty, we can then think of a pairing as a sequence of consecutive duties, interspersed with rest periods, that begins and ends at the crew's home base. Like duties, pairings are subject to complicated rules and a non-linear cost function. Pairing rules may include restrictions on the number of days away from base and the amount of rest between duties. Similar to a duty, U.S. airlines often express the cost function of a pairing as the maximum of three components – the sum of the *duty costs*; some fraction times the *total time away from base (TAFB)*; and some *guaranteed minimum pairing time*.

3.2.1 The Set Partitioning Formulation

There is extensive literature addressing the crew pairing problem. Solution approaches include the works of Anbil et al. [4], Ball and Roberts [8], Beasley and Cao [19], Chu et al. [23], Crainic and Rousseau [32], Desaulniers et al. [34], Hoffman and Padberg [42], Klabjan et al. [43], Klabjan and Schwan [45], Lavoie et al. [46], Levine [47], and Vance et al. [56].

Anbil et al. [2], Anbil et al. [3], Andersson et al. [5], Butchers et al. [20], Gershkoff [37], and Graves et al. [39] discuss techniques in use in industry.

Barnhart et al. [12] and Barnhart and Shenoi [17] discuss long-haul crew scheduling problems.

Surveys appear in Barnhart and Talluri [18], Clarke and Smith [26], Desaulniers et al. [35] and Barnhart et al. [14].

Because of the complexity of the pairing structure, almost all of this literature formulates the crew pairing model as a *set partitioning problem*. In such a formulation, there is one binary decision variable y_p for each pairing $p \in P$ (where P is the set of all feasible pairings). We define δ_{fp} to be an *indicator variable*, with value one if pairing p contains flight $f \in F$ (where F is the set of all flight legs) and value zero otherwise. Given this, we write the crew pairing problem as

CP:

$$\min \sum_{p \in P} c_p y_p \tag{3.1}$$

$$st$$

$$\sum_{p \in P} \delta_{fp} y_p = 1 \quad \forall\ f \in F \tag{3.2}$$

$$y_p \in \{0, 1\} \qquad \forall p \in P. \tag{3.3}$$

The objective (3.1) minimizes the cost of the chosen set of pairings. The cover constraints (3.2) and integrality constraints (3.3) state that, for each flight f, the number of chosen pairings containing that flight must be exactly one. This formulation eliminates the need to incorporate complicated network rules. It also allows us to linearize the cost function, because the cost associated with each pairing is computed "off-line." Note, however, that the number of variables (that is, the number of feasible pairings in the flight network) is exponentially large – often exceeding hundreds of millions. Thus, the majority of the crew pairing literature focuses on solution techniques for solving this large-scale integer program.

3.3 The Maintenance Routing Problem

The purpose of the maintenance routing problem is to ensure that individual aircraft can be assigned to flights in such a way that they have adequate opportunity for routine maintenance checks. In some discussions of this problem, the objective function maximizes *through revenues* – the potential revenue associated with offering passengers direct rather than connecting flight itineraries. In practice, however, this revenue is hard to capture accurately, and the financial

impact is far less significant than crew costs (Cordeau et al.[31], Klabjan et al. [44]). Therefore, as in much of the literature, we focus on the feasibility aspect of maintenance routing.

In the U.S., the FAA requires aircraft to undergo an assortment of regular maintenance checks (Gopalan and Talluri [38], Barnhart and Talluri [18], Talluri [55], Clarke et al. [25]). These requirements are strictly enforced, with the FAA grounding aircraft that exceed the maximum time between maintenance checks. The standard maintenance routing problem focuses on *A checks*, which must occur at most every 65 flight hours. These checks occur at specific maintenance stations within the network, and typically are conducted overnight. In order to ensure that the A check requirement is satisfied, most airlines use the more conservative requirement that an aircraft must spend at least every third (for some fleet types, at least every fourth) night at a maintenance station.

As is the case with crew pairing, it is difficult within a network flow framework to formulate this model – keeping track of the time since the last maintenance opportunity is quite cumbersome. Therefore, a string-based approach is often used (for example, see Barnhart et al. [9]). In such a formulation, we have one binary decision variable d_r for each feasible *route string* r, where a route string is defined to be a sequence of non-repeating consecutive flights, which begins and ends at maintenance stations (not necessarily the same one) and does not exceed the maximum time between maintenance checks. We denote the set of feasible route strings by R.

In addition to cover constraints similar to those found in the crew pairing model, we also need to ensure that the number of aircraft needed does not exceed the number available. In order to do so, we introduce a set of nodes N and *ground arc variables* g_n^+ and g_n^-. Nodes represent the points in space and time at which route strings begin or end (and thus, aircraft are needed or become available). Given a node n that represents time t at station s, the corresponding ground arc variables g_n^- and g_n^+ represent the number of aircraft on the ground at station s immediately prior to and immediately following time t, respectively. By maintaining balance of flow at each node, we can ensure that the aircraft flow forms a circulation.

Given the number of aircraft both on route strings and on ground arcs, we can then compute the total number of aircraft in use at some arbitrary time T, which we refer to as the *countline*. Since aircraft flow is a circulation, if the number of aircraft in use at time T does not exceed

the available number of aircraft (which we denote by K), then *at no time* is the number of available aircraft exceeded. We define τ_r to be the number of times that route string r crosses the countline. Similarly, N^T is the set of nodes with corresponding ground arcs g_n^+ spanning the countline. We define α_{fr} to be an indicator variable, with value one if route string r contains flight f and value zero otherwise. Given these definitions, we now state the maintenance routing problem.

MR:

$$\min \sum_{r \in R} c_r d_r \tag{3.4}$$

$$st$$

$$\sum_{r \in R} \alpha_{fr} d_r = 1 \qquad \forall\ f \in F \tag{3.5}$$

$$\sum_{\substack{r \text{ ends at} \\ \text{node } n}} d_r + g_n^- - \sum_{\substack{r \text{ starts at} \\ \text{node } n}} d_r - g_n^+ = 0 \qquad \forall\ n \in N \tag{3.6}$$

$$\sum_{r \in R} \tau_r d_r + \sum_{n \in N^T} g_n^+ \leq K \tag{3.7}$$

$$d_r \in \{0, 1\} \qquad \forall r \in R \tag{3.8}$$

$$g_n^+,\ g_n^- \geq 0 \qquad \forall n \in N. \tag{3.9}$$

The objective function (3.4) minimizes the cost of the chosen route strings – we set the coefficients c_r to zero, given that we are concerned only with finding a feasible solution. The first set of constraints (3.5) are cover constraints stating that each flight must be included in exactly one chosen route string. The second set of constraints (3.6) are *balance constraints*. They ensure that the flow on route strings and ground arcs forms a circulation. The balance constraint for a specific node n states that the number of aircraft on route strings terminating

at n plus the flow on the ground arc into n must equal the number of aircraft on route strings originating at n plus the flow on the ground arc out of n. Constraint (3.7) ensures that the total number of aircraft in use at time T (and thus at any point in time) does not exceed the number of aircraft in the fleet, namely K. Finally, note that although the route string variables are required to be binary (3.8), the integrality of the ground arc variables can be relaxed (3.9), as discussed in Hane et al. ([41]).

As in the crew pairing problem, maintenance routing is complicated by an exponentially large number of valid route strings. Clarke and Smith [26] and Barnhart and Talluri [18] provide surveys of the solution literature.

3.4 The Link Between Crew Pairing And Maintenance Routing

One restriction on a valid crew pairing is that the time between two sequential flights within a duty must satisfy some minimum connection time. In other words, the time must be sufficient, among other things, for the crew to move through the terminal from the first flight's arrival gate to the second flight's departure gate. This travel time is not necessary if the two flights have been assigned to a common aircraft in the maintenance routing solution. Thus, certain short connections are permitted within the crew pairing network only if they appear in the maintenance routing solution. We refer to such flight connections as *short connects*.

Based on this link between crew pairing and maintenance routing, we can view the sequential approach to solving these two problems in the following way (see Figure 3-2). First, we find a feasible solution to the maintenance routing problem. This solution contains some (possibly empty) set of short connects. We then use this short connect set to augment the set of feasible connections in the crew pairing network, which in turn is used to generate feasible crew pairings. Finally, we use these pairings in solving the crew pairing problem.

Clearly, then, these two problems are interdependent. Because different solutions to the maintenance routing problem lead to different sets of valid pairings, the maintenance routing solution might can significantly impact the quality of the crew pairing solution.

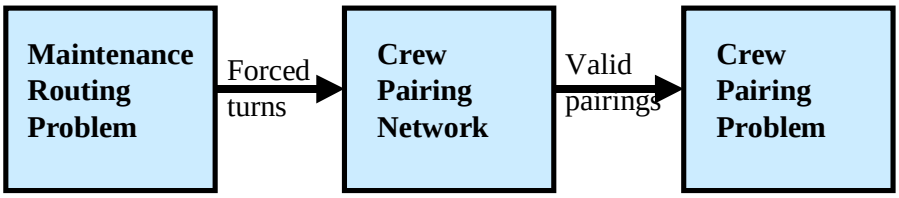


Figure 3-2: Sequential approach

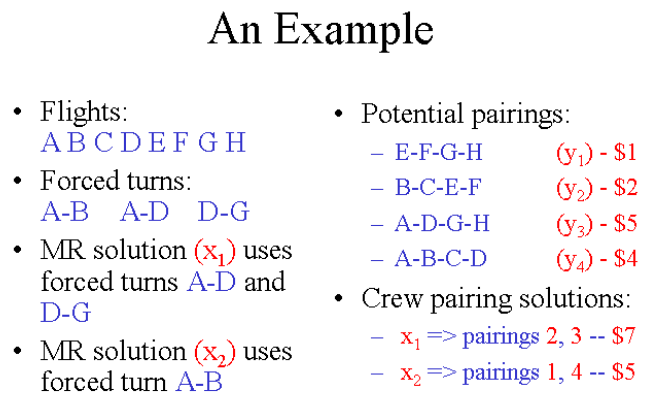


Figure 3-3: Example data

3.5 An Example

We present a small example to demonstrate how a sequential approach can lead to sub-optimality. The data for this example is provided in Figure 3-3. We consider eight flights, denoted A, B, C, D, E, F, G, and H. There are three potential short connects in the network – connections A-B, A-D, and D-G. Two feasible maintenance routing solutions are considered. The first includes short connect A-B. The second includes short connects A-D and D-G. There are four potential crew pairings, as shown in the figure.

In this problem instance, the two maintenance routing solutions define two different sets of feasible crew pairings and thus two different crew pairing problems. For maintenance routing

solution 1, pairing 4 is excluded because it contains short connects A-D and D-G. Thus, the optimal solution in this case uses pairings 1 and 3, at a cost of 3. For maintenance routing solution 2, pairing 3 is excluded because it contains short connect A-B. In this case, the optimal solution uses pairings 2 and 4, at a cost of 2.

This simple example demonstrates how different maintenance solutions might result in different crew pairing solutions. Computational experiments presented in Klabjan et al. [44] and Cordeau et al. [31] demonstrate that these differences can have significant financial impact in real-world instances, providing motivation for an integrated approach.

3.6 Integrated Approaches

There are a number of ways one might go about integrating maintenance routing and crew pairing decisions in order to yield improved solutions. Before describing our approach, we conclude this section with a review of some alternatives.

In Klabjan, et al. [44], the crew pairing problem is solved first, with all pairings permitted regardless of the short connects they contain. The maintenance routing problem is subsequently solved, with the short connects that were used by the crew pairing solution required to be used in the maintenance routing solution as well. Although successful in many instances, this approach does not guarantee maintenance feasibility and non-pathological counter-examples can be found.

In order to ensure maintenance feasibility, another approach is to integrate the original basic models directly. This is considered by Cordeau et al. [31]. They maintain the crew pairing and maintenance string variables and the original sets of constraints. In addition, they add one constraint for each short connect, permitting the model to choose a pairing containing that short connect only if the short connect is also included in a chosen maintenance string. They then solve the resulting combined model using Benders decomposition. Although their model uses an approximation of the true crew costs and is not designed for hub-and-spoke networks, their computational results support the value of integrating these two problems.

A third approach would be to solve the crew pairing model many times, each time using the short connects of a different maintenance routing solution to determine the set of feasible

pairings. The crew pairing solution with the best objective value would be selected, along with the corresponding maintenance routing solution. This approach has many benefits – it can use directly the most current maintenance routing and crew pairing models and solution algorithms, it guarantees a feasible solution at every iteration, and it allows solution time and quality to be traded off through the selection of the number of maintenance routing solutions considered. A key difficulty in this approach, however, is that it requires multiple crew pairing problems to be solved. In the section that follows, we present an *Extended Crew Pairing Model (ECP)* that takes the same philosophical approach as this third option but considers multiple maintenance routing solutions while solving a single crew pairing problem.

Chapter 4

The Extended Crew Pairing Model

4.1 The ECP Model

In the *Extended Crew Pairing Model (ECP)*, we make decisions not only about which crew pairings to include, but which maintenance routing solution to select as well. In modeling this, we start with the basic crew pairing model and add a collection of variables which represent *complete solutions* to the maintenance routing problem. Thus, we eliminate the need for maintenance routing cover, balance, and count constraints, instead using a single *convexity constraint*, which forces the model to select exactly one maintenance routing solution. We further add one additional constraint for each short connect, specifying that a crew pairing containing that short connect cannot be chosen unless the chosen maintenance routing solution also contains that short connect.

In stating the model, we use the following notation:

- P is the set of *all* feasible pairings p (that is, all short connects are included in the crew pairing network).
- F is the set of flights to be covered in the network.
- S is the set of all feasible maintenance routing solutions.
- T is the set of potential short connects.
- c_p is the cost of pairing p.

- δ_{fp} is an indicator variable which has value 1 if flight f is covered by pairing p and 0 otherwise.
- η_{tp} is an indicator variable that has value 1 if short connect t is included in pairing p and 0 otherwise.
- β_{ts} is an indicator variable that has value 1 if short connect t is included in maintenance solution s and 0 otherwise.
- y_p is the binary decision variable associated with pairing p. If $y_p = 1$ then pairing p is included in the solution; if $y_p = 0$ then it is not.
- x_s is the binary decision variable associated with maintenance solution s. If $x_s = 1$ then maintenance solution s is chosen; if $x_s = 0$ then it is not.

Given this notation, we state the model as

ECP:

$$\min \sum_{p \in P} c_p y_p \tag{4.1}$$

$$st$$

$$\sum_{p \in P} \delta_{fp} y_p = 1 \qquad \forall f \in F \tag{4.2}$$

$$\sum_{s \in S} \beta_{ts} x_s - \sum_{p \in P} \eta_{tp} y_p \geq 0 \qquad \forall t \in T \tag{4.3}$$

$$\sum_{s \in S} x_s = 1 \tag{4.4}$$

$$x_s \in \{0, 1\} \qquad \forall s \in S \tag{4.5}$$

$$y_p \in \{0, 1\} \qquad \forall p \in P. \tag{4.6}$$

The objective (4.1) and the cover constraints (4.2) are the same as in the basic crew pairing model (see Section 3.2). Constraint (4.4), in conjunction with the integrality requirements (4.5), ensures the selection of exactly one solution to the maintenance routing problem. Constraint set (4.3) eliminates pairings which contain short connects not included in this selected maintenance solution.

Clearly, this model yields an optimal integrated solution if all maintenance routing solutions are included. Furthermore, even if only a subset of the maintenance solutions are included, we are still guaranteed a feasible solution to the integrated model. As an example, in Figure 4-1 we present the *ECP* constraint matrix for the problem instance from Section 3.4.

	x_1	x_2	y_1	y_2	y_3	y_4		rhs	
	0	0	0	0	1	1	=	1	A
	0	0	0	1	1	0	=	1	B
Flights:	0	0	0	1	1	0	=	1	C
	0	0	0	0	1	1	=	1	D
	0	0	1	1	0	0	=	1	E
	0	0	1	1	0	0	=	1	F
	0	0	1	0	0	1	=	1	G
	0	0	1	0	0	1	=	1	H
	1	0	0	0	−1	0	≥	0	A-B
Forced turns:	0	1	0	0	0	−1	≥	0	A-D
	0	1	0	0	0	−1	≥	0	D-G
Convexity:	1	1	0	0	0	0	=	1	Conv.

Figure 4-1: Extended model with both maintenance solutions

The benefit of this approach is that it allows the crew pairing objective function to influence the selection of the maintenance routing solution and thus the set of short connects. Associated with this model, however, are a number of concerns regarding tractability. How many maintenance solutions will be needed in this new model? How can they be identified? Will the new model have too many binary variables? We address these concerns in the remainder of this section.

4.2 Reducing Problem Size Through Variable Elimination

In theory, we could find an optimal solution to *ECP* by including one column for every feasible maintenance routing solution. In practice, however, this is unlikely to be a tractable option, given the exponentially large number of feasible maintenance solutions. In this section, we leverage the fact that *only certain key maintenance routing decisions impact the crew pairing problem*, thereby allowing us to significantly decrease the number of maintenance variables required to ensure a provably optimal solution.

Throughout our discussion of *ECP,* we have been referring to *maintenance routing solution variables.* An examination of the constraint matrix, however, highlights the fact that most maintenance routing decisions are implicit in the variable definition, with only short connect information stated explicitly. In other words, we can think of a maintenance column not as a full-blown specification of a feasible solution to the maintenance routing problem, but as a short connect *set for which a feasible maintenance routing solution exists*. We refer to this as a *maintenance-feasible short connect set.* This has important ramifications for the size and structure of *ECP*, because there may be many different feasible maintenance routing solutions all associated with the same set of short connects. In particular, when solving the crew pairing model for a given short connect set, we do not care *how* the aircraft are routed, but only that a feasible solution exists.

We claim that in order to guarantee an optimal solution to *ECP*, it is not necessary to include one column for each feasible maintenance routing solution. Instead, it is sufficient to include one column for each *unique and maximal (UM)* maintenance-feasible short connect set. We define this terminology in the following sections.

4.2.1 Uniqueness

Consider a basic example with eight flights (A, B, C, D, E, F, G, and H) and one short connect (A-B), as shown in Figure 4-2-a. Suppose that there are two feasible maintenance solutions of two route strings each (A-B-C-G and D-E-F-H or A-B-F-H and D-E-C-G), shown in Figures 4-2-b and 4-2-c. Note that these two solutions will result in identical columns in *ECP*, with 1's in the row for short connect A-B and in the convexity constraint, and 0's throughout the rest

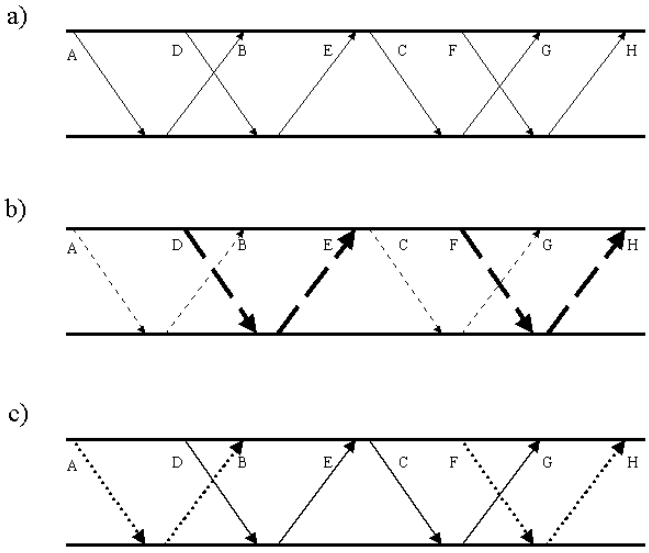


Figure 4-2: Example

of the column. Therefore, we need to include only one of these two redundant columns in *ECP* – both solutions define the same set of feasible crew pairings.

We refer to this potential for reducing the required number of columns in *ECP* as *uniqueness*. In other words, we only need one column for each unique maintenance-feasible short connect set, rather than one per distinct maintenance routing solution. To gain some sense of the impact of this reduction, consider the following example. For an actual airline problem instance, containing 41 flights, we selected a random set of short connects and began generating distinct feasible maintenance routing solutions, each containing exactly these short connects. We aborted this process after generating over 8,700 distinct solutions. All 8,700 solutions can be reduced to a single column in *ECP*.

Because the number of potential short connects is typically only a small fraction of the number of possible aircraft connections in the network, we believe that similar reductions in the number of required maintenance columns will be found in most real-world instances.

4.2.2 Maximal Sets

In the previous section, we discussed how the set of distinct maintenance routing solutions could be reduced to the set of distinct maintenance-feasible short connect sets. We take this idea one step further by defining the concept of a *maximal set*. Consider the case where one maintenance solution contains three short connects, denoted by U-V, W-X, and Y-Z. Another solution contains short connects U-V and Y-Z only. Clearly, any crew pairing that is feasible for the second maintenance routing solution is also feasible for the first solution. Therefore, given that we are only concerned with finding a *feasible* maintenance routing solution, we can disregard the second column. In doing so we do not eliminate from *ECP* any feasible crew pairing solutions, because the chosen maintenance solution tells us which short connects are *permissible* for the crews, not *required*.

More generally, we note that it is necessary only to include columns that are *maximal* – that is, columns representing a maintenance-feasible short connect set for which adding any additional short connects would result in maintenance infeasibility. In our small example, the feasibility of set {U-V, W-X, Y-Z} means that columns representing short connect sets {U-V}, {W-X}, {Y-Z}, {U-V, W-X}, {U-V, Y-Z} and {W-X, Y-Z} can all be discarded.

By noting that we only need to consider those maintenance solution columns whose short connect sets are *unique and maximal (UM)*, we dramatically reduce the number of columns required in our model. Consider another example, in which we looked at a problem instance containing 61 flights. This network yielded well over 25,000 distinct solutions to the maintenance routing problem. However, only **four** of these solutions were unique and maximal.

Based on the ideas of uniqueness and maximal sets, we state the following theorem.

Theorem 1 *In order to guarantee an optimal solution to* ECP*, it is sufficient to include only those columns that represent unique and maximal maintenance-feasible short connect sets.*

Proof. Given the set of all maintenance feasible routing solutions, we can remove any column that does not correspond to a *UM* short connect set without eliminating any feasible crew pairing solutions. ■

In the following section we provide theoretical upper bounds as well as practical limits on the number of columns required to guarantee optimality. We also present supporting computational

results.

4.2.3 Theoretical Bounds And Computational Results

Consider a network with k potential short connects. A weak upper bound on the number of required maintenance variables is therefore 2^k – the number of unique combinations of short connects. This bound, however, only uses uniqueness and not maximal sets. For example, the 2^k bound assumes that the set containing all k short connects is maintenance feasible. If this is the case, then the number of required columns is not 2^k, but simply one, because all other columns are dominated by this exhaustive column. Similarly, if the k columns each representing a single short connect are all maximal, then all larger sets must be infeasible by definition; therefore the number of required columns is k.

In Theorem 2, we present a tighter bound that uses not only uniqueness but maximal sets as well.

Theorem 2 *For a problem instance with k potential short connects, an upper bound on the number of* UM *maintenance-feasible short connect sets is* $\binom{k}{\lceil \frac{k}{2} \rceil}$.

Proof. In order to prove this, we prove the stronger claim that for k given objects, the largest collection of unique subsets that we can choose, such that no one of these subset is fully contained in another chosen subset, has cardinality of at most $\binom{k}{\lceil \frac{k}{2} \rceil}$. We refer to such a collection as a *collection without dominance*. For example, given objects $\{a, b, c, d, e\}$, one collection without dominance is $\{\{a\}, \{b, c\}, \{b, d, e\}\}$, with cardinality three. The collection $\{\{a, b\}, \{a, b, d\}\}$ is *not* without dominance, because $\{a, b\}$ is a subset of $\{a, b, d\}$.

We claim that a collection without dominance of maximum cardinality can be found by generating all subsets of size $\lceil \frac{k}{2} \rceil$. We prove this by showing that, given *any* collection without dominance, we can form a new collection without dominance of equal or greater cardinality by replacing all subsets of size $s < \lceil \frac{k}{2} \rceil$ with at least as many subsets of size $s + 1$, and all subsets of size $s > \lceil \frac{k}{2} \rceil$ with at least as many subsets of size $s - 1$.

$s < \lceil \frac{k}{2} \rceil$:

Suppose that in X, a given collection without dominance, the subsets of smallest cardinality contain $s < \lceil \frac{k}{2} \rceil$ elements. Let G denote the group of m subsets of size s in this collection, and

let i denote the number of distinct objects in G. For example, if we have a problem instance with $k = 7$ objects, denoted a through g, and the collection X is $\{\{a,b\},\{a,c\},\{d,e\},\{b,c,e,f,g\}\}$ then $s = 2$, $G = \{\{a,b\},\{a,c\},\{d,e\}\}$, $m = 3$, and $i = 5$ (a, b, c, d, and e).

We consider two cases:

- If $i < k$, then for each subset in G we can form $(k - i)$ new subsets of size $s + 1$, each of which is created by adding one element not found in G. In the preceding example, we would replace $\{a,b\}$ with $\{a,b,f\}$ and $\{a,b,g\}$; $\{a,c\}$ with $\{a,c,f\}$ and $\{a,c,g\}$; etc. Given that $k > i$, each subset in G is replaced by at least one new subset, and thus the new collection (which is clearly without dominance) does not decrease in cardinality..

- If $i = k$, we replace each subset with the $(k - s)$ subsets of cardinality $s + 1$ that can be formed from it. For example, if $k = 5$ and $G = \{\{a,b\},\{a,c\},\{d,e\}\}$ then from $\{a,b\}$ we could form $\{a,b,c\}$, $\{a,b,d\}$, and $\{a,b,e\}$. However, this may result in duplicate subsets. At most, for each new subset of size $s + 1$, we may form $s + 1$ versions. In our example, $\{a,b,c\}$ can be formed by adding a to $\{b,c\}$, b to $\{a,c\}$, and c to $\{a,b\}$. Thus, we replace the m subsets with at least $\frac{m*(k-s)}{s+1}$ subsets. Given that $s < \lceil \frac{k}{2} \rceil$, basic algebraic manipulation shows that $\frac{k-s}{s+1} \geq 1$ and thus we have not decreased the cardinality of the collection. Again, this new collection is clearly without dominance.

$s > \lceil \frac{k}{2} \rceil$:

We next consider the case when the largest subsets are of size $s > \lceil \frac{k}{2} \rceil$. Here, we replace each of the m subsets of size s with the s new subsets of size $s - 1$ that can be formed by deleting one element. For example, we would replace $\{b,c,e,f,g\}$ with $\{c,e,f,g\}$, $\{b,e,f,g\}$, $\{b,c,f,g\}$, $\{b,c,e,g\}$, and $\{b,c,e,f\}$. Again, this may result in duplicate subsets being formed. For example, if $k = 5$ and $s = 4$ then $\{a,b,c\}$ might be formed from both $\{a,b,c,d\}$ and $\{a,b,c,e\}$. In total, each subset may be formed in no more than $k - (s - 1)$ ways. Thus, we replace the m subsets with at least $\frac{m*s}{k-s+1}$ new subsets. Given that $s > \lceil \frac{k}{2} \rceil$, algebraic manipulation shows that $\frac{s}{k-s+1} \geq 1$ and thus we have not decreased the cardinality of the collection, which again is clearly without dominance.

Thus, given any collection without dominance, we can construct another collection without dominance of the same or greater cardinality, in which all subsets are of size $\lceil \frac{k}{2} \rceil$. Clearly, the

cardinality of this collection is bounded from above by $\binom{k}{\lceil \frac{k}{2} \rceil}$. ■

This bound is still weak in that it doesn't incorporate all relevant information. In particular, the bound assumes that no two short connects are mutually exclusive. This is often not the case, given that flights must be covered by only one route string. For example, consider three flights, A, B, and C. Given short connects A-B and A-C, at most one of these can be included in any given maintenance solution. This fact is particularly significant because a large number of short connect opportunities occur at *banks*. A bank is a set of closely timed incoming flights followed by a set of closely timed outgoing flights at a hub airport. Banks create many short connect opportunities. For example, five incoming flights followed by five outgoing flights might result in 25 potential short connects. However, clearly only at most five of these short connects can appear in any one given maintenance solution, because each incoming flight has only one outgoing connection.

Thus, this bound can be strengthened even further in many cases, depending on the structure of the problem instance. For example, suppose that the k short connects could be grouped into m sets of size $\frac{k}{m}$ each, where each short connect in a given set shares the same first flight (e.g. a set might represent short connects $A-B$, $A-C$, $A-D$, and $A-E$). Then we can replace the bound $\binom{k}{\lceil \frac{k}{2} \rceil}$ with $\binom{m}{\lceil \frac{m}{2} \rceil} * \left(\frac{k}{m}\right)^{\lceil \frac{m}{2} \rceil}$ – that is, we select clusters of size $\lceil \frac{m}{2} \rceil$ of short connect sets and then for each set in the short connect cluster, there are $\frac{k}{m}$ choices for the actual short connect. This can result in a significant improvement in the bound. For example, if k is 30 and m is 5, then $\binom{k}{\lceil \frac{k}{2} \rceil} = \binom{30}{15} = 1.551 \times 10^8$ while $\binom{m}{\lceil \frac{m}{2} \rceil} * \left(\frac{k}{m}\right)^{\lceil \frac{m}{2} \rceil} = \binom{5}{3} * \left(\frac{30}{5}\right)^3 = 2160$.

To get a sense for how many columns are needed in real-world instances, we considered five fleet types from the daily problem of a major U.S. airline for a given time period. For each of these instances, we computed both of the bounds stated earlier, the actual number of *UM* short connect sets, and the ratio of the actual number to the $\binom{k}{\lceil \frac{k}{2} \rceil}$ bound. These are provided in Table 4.1.

The first column of this table indicates the number of flights in each network. The second column indicates k, the number of potential short connects. The two bounds, 2^k and $\binom{k}{\lceil \frac{k}{2} \rceil}$ are given next. The next column contains the actual number of *UM* columns for this network. The final column contains the ratio of the actual number to the bound $\binom{k}{\lceil \frac{k}{2} \rceil}$. Note that not only is the number of *UM* columns consistently smaller than our current best bound, but furthermore

# Flights	k	2^k	$\binom{k}{\lceil \frac{k}{2} \rceil}$	UM Col.'s	Ratio
81	5	32	10	3	.3
77	7	128	35	4	.114
41	9	512	126	2	1.587×10^{-2}
224	12	4096	924	6	6.493×10^{-3}
43	30	1.073×10^9	1.551×10^8	38	2.449×10^{-7}

Table 4.1: Number of UMI forced turn sets

that as the number of short connects grows, the ratio decreases significantly. This further supports the idea that the number of *UM* short connect sets is dramatically smaller than the number of feasible maintenance routing solutions.

4.2.4 Identifying UM Columns

Of course, it is not sufficient that the number of required columns be small. We must also be able to identify these columns in a reasonable fashion – for example, we don't want to have to generate *all* 25,000 feasible maintenance routing solutions from our earlier example in order to isolate those four which are unique and maximal.

We can identify *UM* maintenance routing solutions by solving a series of maintenance routing problems with side constraints and a modified objective function. We begin by solving the maintenance routing problem with the objective of maximizing the total number of short connects included in the solution. We associate with each route string r a coefficient c_r, where c_r is the number of short connects included in route string r. Our objective function is then

$$\min \sum -c_r d_r.$$

Clearly, this problem yields a *UM* set of short connects. Let T represent the set of potential short connects in the network and let T^1 represent the set of short connects used in the solution to this first iteration of the modified maintenance routing problem. We then update the model by adding a constraint stating that the solution must contain at least one short connect which is *not* in T^1. Such a constraint can be modeled by stating that the sum of all chosen route strings which contain at least one short connect in $T \backslash T^1$ must be greater than or equal to one. We then re-solve the model and generate a new solution with a new corresponding set of

short connects T^2. This set will be *UM*, because it is the largest cardinality set which is not a subset of another solution. We iterate, adding cuts and re-solving the model, until all *UM* short connect sets have been identified.

Thus, it is possible to generate n *UM* columns in roughly the amount of time it takes to solve n maintenance routing problems. In fact, it may take significantly less time, because we are actually *re-solving* the maintenance routing problem, having simply added a cut at each iteration.

4.3 Relaxing The Integrality Of The Maintenance Variables

Even a relatively small number of additional binary variables can greatly impact the tractability of an integer program. In this section, we prove that in solving *ECP*, *it is not necessary to enforce the integrality of the maintenance routing solution variables.* Consider what happens to *ECP* when we relax the integrality of the maintenance solution variables x. We refer to this as the *Relaxed Extended Crew Pairing Model (RECP).* Its formulation is the same as that for *ECP*, except that we replace equation (4.5) with

$$x_s \in [0,1] \ \forall s \in S. \tag{4.7}$$

We claim that, given an optimal solution to *RECP*, it is trivial to construct a corresponding feasible solution to *ECP* with the same objective value. Because *RECP* is a relaxation of *ECP*, this solution will be optimal for *ECP*. This implies that *ECP* can be solved by solving *RECP*, a model with the same number of binary variables as the crew pairing model alone.

Before proving this claim, we first introduce notation and define the relationship between optimal solutions to *RECP* and *ECP*. Consider (x^*, y^*), an optimal solution to *RECP* with objective value z^*_{RECP}. Let B^* represent the set of maintenance routing solutions that appear in this optimal solution – that is, $B^* \equiv \{s : x^*_s > 0\}$. Let N^* represent the set of remaining maintenance routing solutions – that is, $N^* \equiv \{s : x^*_s = 0\}$. Set $\overline{y}_p = y^*_p$ for all pairings $p \in P$. Choose any one maintenance solution $s' \in B^*$ (i.e. $x^*_{s'} > 0$) and set $\overline{x}_{s'} = 1$. Let all other $\overline{x}_s = 0$. Define $\overline{X}_{B^*} = \{\overline{x}_s : s \in B^*\}$ and $\overline{X}_{N^*} = \{\overline{x}_s : s \in N^*\}$. Finally, let T' represent the set of short connects used by crew pairings in the solution y^* – that is, $T' \equiv \{t : \sum_{p \in P} \eta_{tp} y^*_p = 1\}$.

We now prove that $(\overline{x}, \overline{y})$ is an optimal solution to *ECP*. We begin by defining *LECP*, a *limited* version of *ECP*, in which the values of some of the variables ($\overline{y}$ and $\overline{X}_{N^*}$) are specified and we solve for the remaining variables ($\overline{X}_{B^*}$).

LECP:

$$\min 0 \tag{4.8}$$

$$st$$

$$\sum_{s \in B^*} \beta_{ts} x_s \geq 1 \ \forall \ t \in T' \tag{4.9}$$

$$\sum_{s \in B^*} x_s = 1 \tag{4.10}$$

$$x_s \in \{0, 1\} \ \forall s \in B^*. \tag{4.11}$$

Lemma 3 *$\overline{X}_{B^*}$ is feasible for* LECP*.*

Proof. Define

$$S_t \equiv \{s : \ \beta_{ts} = 1, s \in B^*\}$$

for any $t \in T'$. That is, S_t is the set of all maintenance routing solutions that use short connect t and appear in the solution to *RECP*.

Suppose $\beta_{t\widehat{s}} = 0$ for some $\widehat{s} \in B^*$ and some $t \in T'$.

By definition and by (4.3),

$$\sum_{s \in S_t} x_s^* \geq 1.$$

But

$$x_{\widehat{s}}^* > 0$$

and

$$\widehat{s} \notin S_t,$$

implying that

$$\sum_{s \in S_t} x_s^* + x_{\widehat{s}}^* > 1.$$

However, by (4.4),

$$\sum_{s \in S_t} x_s^* + x_{\widehat{s}}^* \leq \sum_{s \in S} x_s^* = 1 -$$

a contradiction.

Hence,

$$\beta_{ts} = 1 \ \forall t \in T', \forall s \in B^*$$

and thus (4.9) is redundant to (4.10).

Therefore, *LECP* can be written equivalently as

$$\min 0$$

subject to

$$\sum_{s \in B^*} x_s = 1$$

$$x_s \in \{0, 1\} \ \forall s \in B^*.$$

By construction, $\overline{X}_{B^*}$ is feasible for *LECP*. ■

Theorem 4 *Given that* $\overline{X}_{B^*}$ *is feasible for* LECP, $(\overline{x}, \overline{y})$ *is an optimal solution to* ECP.

Proof. An equivalent formulation for *ECP* ((4.1) - (4.6)) is

$$\min \sum_{s \in B^*} 0x_s + \sum_{s \in N^*} 0x_s + \sum_{p \in P} c_p y_p \tag{4.12}$$

$$st$$

$$\sum_{p \in P} \delta_{fp} y_p = 1 \ \forall \ f \in F \tag{4.13}$$

$$\sum_{s \in B^*} \beta_{ts} x_s + \sum_{s \in N^*} \beta_{ts} x_s - \sum_{p \in P} \eta_{tp} y_p \geq 0 \ \forall t \in T \tag{4.14}$$

$$\sum_{s \in B^*} x_s + \sum_{s \in N^*} x_s = 1 \tag{4.15}$$

$$x_s \in \{0, 1\} \ \forall s \in S \tag{4.16}$$

$$y_p \in \{0, 1\} \ \forall p \in P. \tag{4.17}$$

Set

$$y = \overline{y} \equiv y^*$$

and

$$x_s = \overline{x}_s \equiv x_s^* \equiv 0 \ \forall s \in N^*.$$

Given that $y = y^*$ and $x_s = 0 \ \forall s \in N^*$, (4.12) is equivalent to (4.8); (4.15) is equivalent to (4.10); and (4.16) is equivalent to (4.11). In addition, (4.13) and (4.17) are unnecessary because y^* is feasible for (4.2) and (4.6) in *ECP*.

By definition, for any $t \in T'$,

$$\sum_{p \in P} \eta_{tp} \overline{y}_p = 1;$$

and for all other t,

$$\sum_{p \in P} \eta_{tp} \overline{y}_p = 0.$$

Hence, given that $x_s = 0 \ \forall s \in N^*$ and $x_s \geq 0 \ \forall s \in B^*$, (4.14) is equivalent to (4.9). Thus, if $\overline{X}_{B^*}$ is feasible for *LECP* then $(\overline{x}, \overline{y})$ is feasible for *ECP*.

For fixed y, minimizing (4.12) is equivalent to minimizing (4.8), and

$$\sum_{p \in P} c_p \overline{y}_p \equiv \sum_{p \in P} c_p y_p^* \equiv z_{RECP}^* \leq z_{ECP}^*,$$

so $(\overline{x}, \overline{y})$ is an optimal solution to *ECP*. ■

We conclude by noting that this relaxation property holds in the more general case where c_s is not restricted to be 0. In order to take into account non-zero maintenance routing coefficients, we construct an integer solution $(\widetilde{y}, \widetilde{x})$ to *ECP* in which we again let $\widetilde{y} = y^*$ and $\widetilde{x}_s = 0 \; \forall s \in N^*$. However, instead of choosing any maintenance routing solution s' such that $x_{s'}^* > 0$, we instead choose s' such that $x_{s'}^* > 0$ *and*

$$c_{s'} \leq c_s \quad \forall s \in B^*. \tag{4.18}$$

We let $\widetilde{x}_{s'} = 1$ and $\widetilde{x}_s = 0$ for all other $s \in B^*$.

Corollary 5 *$(\widetilde{x}, \widetilde{y})$ is an optimal solution to* ECP *with zero or non-zero maintenance costs.*

Proof. Let z_{RECP-G}^* be the optimal solution to *RECP* with non-zero maintenance costs. Feasibility of $(\widetilde{x}, \widetilde{y})$ follows from the same logic as used in lemma 3 and theorem 4.

By (4.15), (4.18), and construction,

$$\begin{aligned}
&\sum_{s \in S} c_s \widetilde{x}_s + \sum_{p \in P} c_p \widetilde{y}_p \\
&= c_{s'} + \sum_{p \in P} c_p y_p^* \\
&= c_{s'} \sum_{s \in B^*} x_s^* + \sum_{p \in P} c_p y_p^* \\
&\leq \sum_{s \in B^*} c_s x_s^* + \sum_{p \in P} c_p y_p^* \\
&= z_{RECP-G}^* \; .
\end{aligned}$$

■

In fact, $c_{s'}$ must be strictly equal to $c_{s''}$ for all other s'' in B^*. Otherwise, we would have

$$\sum_{s \in S} c_s \widetilde{x}_s + \sum_{p \in P} c_p \widetilde{y}_p < z^*_{RECP-G},$$

which contradicts the optimality of z^*_{RECP-G}. Thus, even with non-zero maintenance cost coefficients, we can still choose any $s \in B^*$ as our maintenance routing solution.

4.3.1 Quality Of The LP Relaxation

When solving a large-scale integer program, the quality of the LP relaxation can greatly impact computational performance. In this section, we present a brief discussion concerning the strength of the LP relaxation of *ECP*, relative to a more direct approach to integrating the problems.

Consider a direct integration of the string-based maintenance routing and crew pairing problems (these models are presented in Sections 3.3 and 3.2). In addition to the individual models, we include one linking constraint for each short connect t of the form

$$\sum_{r \in R} \phi_{tr} d_r - \sum_{p \in P} \eta_{tp} y_p \geq 0$$

where ϕ_{tr} is an indicator variable that has value one if route string r contains short connect t and zero otherwise. These constraints restrict the crew pairing solution from using a short connect unless the maintenance routing solution contains that short connect. We refer to this as the *basic integrated model (BIM)*.

BIM:

$$\min \sum_{p \in P} c_p y_p \tag{4.19}$$

$$st$$

$$\sum_{r \in R} \alpha_{fr} d_r = 1 \qquad \forall\ f \in F \tag{4.20}$$

$$\sum_{\substack{r \text{ ends at} \\ \text{node } n}} d_r + g_n^- - \sum_{\substack{r \text{ starts at} \\ \text{node } n}} d_r - g_n^+ = 0 \qquad \forall\ n \in N \tag{4.21}$$

$$\sum_{r \in R} \tau_r d_r + \sum_{n \in N^T} g_n^+ \leq K \tag{4.22}$$

$$\sum_{p \in P} \delta_{fp} y_p = 1 \quad \forall\ f \in F \tag{4.23}$$

$$\sum_{r \in R} \phi_{tr} d_r - \sum_{p \in P} \eta_{tp} y_p \geq 0 \qquad \forall t \in T \tag{4.24}$$

$$d_r \in \{0, 1\} \qquad \forall r \in R \tag{4.25}$$

$$g_n^+,\ g_n^- \geq 0 \qquad \forall n \in N. \tag{4.26}$$

$$y_p \in \{0, 1\} \qquad \forall p \in P. \tag{4.27}$$

We claim that the LP relaxation of *ECP*, denoted by ECP^L, provides a lower bound that is equal to or better than that obtained from BIM^L, the LP relaxation of *BIM*.

Theorem 6 *Given any instance of the integrated crew scheduling and maintenance routing problem, the optimal objective value of* ECP^L *will be equal to or greater than the optimal objective value of* BIM^L.

Proof. We first show that for any feasible solution $\{\widehat{x}_s, \widehat{y}_p\}$ to ECP^L, there exists a solution $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ to BIM^L with the same cost. To construct this solution, we begin by observing that for every short connect set s, there exists at least one maintenance routing solution containing exactly those short connects. For every s we select one such maintenance solution arbitrarily and define $\psi_{sr} \equiv 1$ if the solution corresponding to s includes route string r and 0 otherwise. We also define $\vartheta_{ns}^{+/-}$ to be the flow on the ground arc out of/into node n in the solution corresponding to s. We can then construct $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ as follows:

$$\widetilde{d}_r = \sum_{s \in S} \psi_{sr} \widehat{x}_s \qquad \forall r \in R$$

$$\widetilde{g}_n^{+/-} = \sum_{s \in S} \vartheta_{ns}^{+/-} \widehat{x}_s \qquad \forall n \in N$$

$$\widetilde{y}_p = \widehat{y}_p \qquad \forall p \in P.$$

We claim that $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ is feasible for BIM^L.

First, observe that for any $f \in F$

$$\sum_{r \in R} \alpha_{fr} \widetilde{d}_r \tag{4.28}$$

$$= \sum_{r \in R} \alpha_{fr} \sum_{s \in S} \psi_{sr} \widehat{x}_s \tag{4.29}$$

$$= \sum_{s \in S} \widehat{x}_s \sum_{r \in R} \psi_{sr} \alpha_{fr} \tag{4.30}$$

$$= \sum_{s \in S} \widehat{x}_s \tag{4.31}$$

$$= 1, \tag{4.32}$$

where (4.31) follows from (4.30) because every feasible maintenance routing solution contains exactly one route covering each flight. Thus, $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ satisfies (4.20).

Next, observe that for every $n \in N$

$$\sum_{\substack{r \text{ ends at} \\ \text{node } n}} \widetilde{d}_r + \widetilde{g}_n^- - \sum_{\substack{r \text{ starts at} \\ \text{node } n}} \widetilde{d}_r - \widetilde{g}_n^+ \tag{4.33}$$

$$= \sum_{r \text{ ends at node } n} \sum_{s \in S} \psi_{sr} \widehat{x}_s + \sum_{s \in S} \vartheta^-_{ns} \widehat{x}_s - \sum_{r \text{ starts at node } n} \sum_{s \in S} \psi_{sr} \widehat{x}_s - \sum_{s \in S} \vartheta^+_{ns} \widehat{x}_s \tag{4.34}$$

$$= \sum_{s \in S} (\sum_{r \text{ ends at node } n} \psi_{sr} + \vartheta^-_{ns} - \sum_{r \text{ starts at node } n} \psi_{sr} - \vartheta^+_{ns}) \widehat{x}_s \tag{4.35}$$

$$= 0, \tag{4.36}$$

where (4.36) follows from (4.35) because any feasible solution to the maintenance routing problem satisfies balance at all nodes. Thus, $\{\widetilde{d}_r, \widetilde{g}^{+/-}_n, \widetilde{y}_p\}$ satisfies (4.21).

Next, observe that

$$\sum_{r \in R} \tau_r \widetilde{d}_r + \sum_{n \in N^T} \widetilde{g}^+_n \tag{4.37}$$

$$= \sum_{r \in R} \tau_r \sum_{s \in S} \psi_{sr} \widehat{x}_s + \sum_{n \in N^T} \sum_{s \in S} \vartheta^+_{ns} \widehat{x}_s \tag{4.38}$$

$$= \sum_{s \in S} (\sum_{r \in R} \tau_r \psi_{sr} + \sum_{n \in N^T} \vartheta^+_{ns}) \widehat{x}_s \tag{4.39}$$

$$\leq \sum_{s \in S} K \widehat{x}_s \tag{4.40}$$

$$= K, \tag{4.41}$$

where (4.40) follows from (4.39) because any feasible solution to the maintenance routing problem satisfies the count constraint. Thus, $\{\widetilde{d}_r, \widetilde{g}^{+/-}_n, \widetilde{y}_p\}$ satisfies (4.22).

Next, observe that for any $f \in F$,

$$\sum_{p \in P} \delta_{fp} \widetilde{y}_p \tag{4.42}$$

$$= \sum_{p \in P} \delta_{fp} \widehat{y}_p \tag{4.43}$$

$$= 1 \tag{4.44}$$

and thus $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ satisfies (4.23).

Next, observe that for any $t \in T$,

$$\sum_{r \in R} \phi_{tr} \widetilde{d}_r - \sum_{p \in P} \eta_{tp} \widetilde{y}_p \tag{4.45}$$

$$= \sum_{r \in R} \phi_{tr} \sum_{s \in S} \psi_{sr} \widehat{x}_s - \sum_{p \in P} \eta_{tp} \widehat{y}_p \tag{4.46}$$

$$= \sum_{s \in S} \left(\sum_{r \in R} \phi_{tr} \psi_{sr} \right) \widehat{x}_s - \sum_{p \in P} \eta_{tp} \widehat{y}_p \tag{4.47}$$

$$= \sum_{s \in S} \beta_{ts} \widehat{x}_s - \sum_{p \in P} \eta_{tp} \widehat{y}_p \tag{4.48}$$

$$\geq 0, \tag{4.49}$$

where (4.48) follows from (4.47) by definition. Thus, $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ satisfies (4.24).

Additionally, by construction, $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ clearly satisfies (4.25), (4.26), and (4.27).

Thus, $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ is a feasible solution to BIM^L.

Furthermore, we observe that by construction, $\{\widetilde{d}_r, \widetilde{g}_n^{+/-}, \widetilde{y}_p\}$ has the same cost – $\sum_{p \in P} c_p \widehat{y}_p$ – as $\{\widehat{x}_s, \widehat{y}_p\}$.

Thus, given that this is true for *any* solution to ECP^L, clearly the optimal objective value of ECP^L will always be equal to or greater than the optimal objective value of BIM^L.

Finally, to show that ECP^L can in some cases produce *strictly* tighter bounds than BIM^L, we provide the following example. Consider a network with four flights, denoted $\{A, B, C, D\}$, and one potential short connect, AC. There are four possible route strings – $\{ACB$, AD, CB, $CBD\}$. Each of these flight sequences is also a potential crew pairing, with costs {1, 2, 2, 2}.

Finally, we assume that there is an infinite supply of aircraft available, allowing us to disregard the maintenance routing balance and count constraints.

The LP relaxation associated with *BIM* is therefore:

$$\min y_1 + 2y_2 + 2y_3 + 2y_4$$

$$st$$

$$d_1 + d_2 = 1 \tag{4.50}$$

$$d_1 + d_3 + d_4 = 1 \tag{4.51}$$

$$d_1 + d_3 + d_4 = 1 \tag{4.52}$$

$$d_2 + d_4 = 1 \tag{4.53}$$

$$y_1 + y_2 = 1 \tag{4.54}$$

$$y_1 + y_3 + y_4 = 1 \tag{4.55}$$

$$y_1 + y_3 + y_4 = 1 \tag{4.56}$$

$$y_2 + y_4 = 1 \tag{4.57}$$

$$d_1 - y_1 \geq 0 \tag{4.58}$$

$$d_1, d_2, d_3, d_4, y_1, y_2, y_3, y_4 \in [0, 1]$$

where constraints (4.50) - (4.53) are the maintenance cover constraints for flights A - D, constraints (4.54) - (4.57) are the crew cover constraints for flights A - D, and constraint (4.58) is the linking constraint for short connect AC.

The optimal solution is to assign value $\frac{1}{2}$ to the first, second, and fourth route strings and similarly to the crew pairings. This combination allows us to partially utilize the short connect AC and, in doing so, benefit from the less expensive crew pairing ACB. The cost of this solution is 2.5.

Now consider the LP relaxation of *ECP*. The only feasible maintenance solution that can be

constructed from the four possible routes strings is comprised of strings 2 and 3. This solution does not include short connect AC and thus we cannot take advantage of the less expensive crew pairing. The LP relaxation associated with ECP is:

$$\min y_1 + 2y_2 + 2y_3 + 2y_4$$

$$st$$

$$y_1 + y_2 = 1 \tag{4.59}$$

$$y_1 + y_3 + y_4 = 1 \tag{4.60}$$

$$y_1 + y_3 + y_4 = 1 \tag{4.61}$$

$$y_2 + y_4 = 1 \tag{4.62}$$

$$-y_1 \geq 0 \tag{4.63}$$

$$x_1 = 1 \tag{4.64}$$

$$x_1, y_1, y_2, y_3, y_4 \in [0, 1]$$

where constraints (4.59) - (4.62) are the crew cover constraints for flights A - D, constraint (4.63) is the linking constraint for short connect AC, and constraint (4.64) is the maintenance routing convexity constraint. The optimal solution is to assign value 1 to the second and third crew pairings and to select the sole available maintenance routing solution. The optimal cost is 4, and this solution is in fact optimal for the IP as well. ■

In this section, we have proven that ECP has a tighter LP relaxation than BIM by a constructive argument. It is worth noting that this can also be shown using an argument based on Dantzig-Wolfe Decomposition. Because a decomposition-based argument can be used in many composite variable models, we provide this alternate proof as well.

Proof. Consider the LP relaxation BIM^L. Using Dantzig-Wolfe Decomposition (Dantzig and Wolfe [33]), we can write an equivalent model and then show that the set of feasible solutions for this equivalent formulation is a superset of the feasible solutions for ECP^L.

Define E to be the set of extreme points of the polyhedron formed by

$$\sum_{r \in R} \alpha_{fr} d_r = 1 \qquad \forall\ f \in F \tag{4.65}$$

$$\sum_{\substack{r \text{ ends at} \\ \text{node } n}} d_r + g_n^- - \sum_{\substack{r \text{ starts at} \\ \text{node } n}} d_r - g_n^+ = 0 \qquad \forall\ n \in N \tag{4.66}$$

$$\sum_{r \in R} \tau_r d_r + \sum_{n \in N^T} g_n^+ \leq K \tag{4.67}$$

$$0 \leq d_r \leq 1 \qquad \forall r \in R \tag{4.68}$$

$$g_n^+,\ g_n^- \geq 0 \qquad \forall n \in N \tag{4.69}$$

and let ξ_{er} be the (possibly fractional) number of aircraft assigned to route r in extreme point e. We can then re-write BIM^L as

$$\min \sum_{p \in P} c_p y_p$$

$$st$$

$$\sum_{p \in P} \delta_{fp} y_p = 1 \quad \forall\ f \in F \tag{4.70}$$

$$\sum_{r \in R} \phi_{tr} \sum_{e \in E} \xi_{er} \lambda_e - \sum_{p \in P} \eta_{tp} y_p \geq 0 \qquad \forall t \in T \tag{4.71}$$

$$\sum_{e \in E} \lambda_e = 1 \tag{4.72}$$

$$0 \leq \lambda_e \leq 1 \qquad \forall e \in E \tag{4.73}$$

$$0 \leq y_p \leq 1 \qquad \forall p \in P, \tag{4.74}$$

because any feasible solution $\{d_r\}$ can be written as a convex combination of the extreme points in E. We can then re-write (4.71) as

$$\sum_{e \in E} \left(\sum_{r \in R} \phi_{tr} \xi_{er} \right) \lambda_e - \sum_{p \in P} \eta_{tp} y_p \geq 0 \qquad \forall t \in T.$$

Now consider a second model, formed analogously by considering the set E' of feasible solutions to

$$\sum_{r \in R} \alpha_{fr} d_r = 1 \qquad \forall\ f \in F \tag{4.75}$$

$$\sum_{\substack{r \text{ ends at} \\ \text{node } n}} d_r + g_n^- - \sum_{\substack{r \text{ starts at} \\ \text{node } n}} d_r - g_n^+ = 0 \qquad \forall\ n \in N \tag{4.76}$$

$$\sum_{r \in R} \tau_r d_r + \sum_{n \in N^T} g_n^+ \leq K \tag{4.77}$$

$$d_r \in \{0, 1\} \qquad \forall r \in R \tag{4.78}$$

$$g_n^+,\ g_n^- \geq 0 \qquad \forall n \in N. \tag{4.79}$$

The convex hull of E' is clearly contained in the polyhedron formed by convex combinations of E. Therefore, the following model clearly has a smaller feasible space, because it does not include the fractional extreme points of E:

$$\min \sum_{p \in P} c_p y_p$$

st

$$\sum_{p \in P} \delta_{fp} y_p = 1 \quad \forall \ f \in F \tag{4.80}$$

$$\sum_{e \in E'} \left(\sum_{r \in R} \phi_{tr} \xi_{er} \right) \lambda_e - \sum_{p \in P} \eta_{tp} y_p \geq 0 \qquad \forall t \in T. \tag{4.81}$$

$$\sum_{e \in E'} \lambda_e = 1 \tag{4.82}$$

$$0 \leq \lambda_e \leq 1 \qquad \forall e \in E' \tag{4.83}$$

$$0 \leq y_p \leq 1 \qquad \forall p \in P. \tag{4.84}$$

By observing that $E' = S$ and that $\sum_{r \in R} \phi_{tr} \xi_{er} = \beta_{ts}$, we see that this model is in fact ECP^L, and thus ECP^L is tighter than BIM^L. ■

Chapter 5

Implementing ECP

In theory, the Extended Crew Pairing Model is simply a set partitioning problem with side constraints. In practice, given that it contains the already difficult basic crew pairing model itself, *ECP* is a mixed-integer program with an exponential number of binary crew pairing variables. In this section, we discuss some of the implementation issues associated with solving this computationally challenging problem.

Mixed-integer programs are typically solved using a *branch-and-bound* algorithm, in which a series of increasingly constrained linear programming relaxations are solved to find a solution to the original problem. Due to the exponential size of the LP relaxations in *ECP*, it may be necessary to solve them using column generation. This technique of imbedding column generation within a branch-and-bound framework has been referred to in the literature as *branch-and-price* (Barnhart et al. [13]).

When solving a problem using branch-and-price, whether heuristically or to optimality, the user has a wide array of choices to make. These include decisions about how to solve the LP relaxations, branching strategies, and bounding and pruning techniques. In this section, we focus on two particular implementation issues. First, we discuss how column generation can be used to solve the LP relaxations of *ECP*. Second, we present one method for generating the optimal solution value; such bounds are important for establishing optimality gaps. We also present a brief discussion of how to adapt the Extended Crew Pairing Model in those cases where crews are allowed to fly more than one fleet type. Finally, we conclude the section with a computational proof-of-concept.

5.1 Generating Columns

We begin by reviewing the concept of column generation. Recall that in solving a linear program with the simplex method, we first find a basic feasible solution and an associated set of dual values. Based on these duals, we identify a non-basic variable with negative reduced cost *(NRC)* and pivot it into the basis. We repeat this process until no *NRC* variable can be found. This satisfies the optimality conditions and the simplex algorithm terminates. In *column generation*, we do not consider all of the variables explicitly. We instead begin with only a subset of the variables; this is called the *restricted master problem (RM)*. Once we've solved the restricted master problem to optimality, we use the resulting dual values to seek *NRC* variables not included in the restricted master but contained in the original problem. These variables are typically identified by solving a *pricing problem*. A pricing problem is an optimization problem in which there is a one-to-one correspondence between variables in the original problem and solutions in the pricing problem. The cost associated with a solution in the pricing problem corresponds to the reduced cost of the associated variable in the original problem. Thus, if the optimal solution to the pricing problem has a strictly negative value, its corresponding variable in the original problem has negative reduced cost and is therefore a valid pivot variable; we add this variable to the restricted master and re-optimize. If the optimal solution to the pricing problem has value zero, this tells us that there are no *NRC* columns, and therefore the optimal solution to the restricted master is optimal for the original problem as well.

5.1.1 Generating Crew Pairings

In order to see how crew pairings can be generated in *ECP*, we first discuss column generation in the context of the basic crew pairing model. As previously discussed, the basic crew pairing model is a set partitioning problem, with one cover constraint for each flight and one variable for each pairing. Because of the large number of variables, column generation is often used to find negative reduced cost pairings to pivot into the basis. The reduced cost of a pairing is the cost of the pairing minus the sum of the dual values associated with the flights contained in it. If we denote the dual variable associated with the cover constraint for flight f by π_f then we

can write the pricing problem as

$$\min c_p - \sum_{f \in F} \delta_{fp} \pi_f.$$

$$st$$

$$p \in P.$$

Barnhart et al. [14] provide a survey of techniques used in solving such problems. Typically, a crew pairing network is defined – for example, with nodes representing flights and arcs representing feasible flight connections. A specialized algorithm is then used to find the shortest path in this network, subject to side constraints ensuring that a path corresponds to a feasible pairing. The length of a path is defined to be the reduced cost of the associated pairing and thus the shortest path corresponds to the most negative reduced cost pairing.

In the Extended Crew Pairing Model, column generation is still needed; however, we must also take into account the dual variables associated with the short connect linking constraints. If we denote by γ_t the dual variable associated with short connect t, then the reduced cost of a pairing becomes

$$c_p - \sum_{f \in F} \delta_{fp} \pi_f + \sum_{t \in T} \eta_{tp} \gamma_t.$$

By assigning the dual values associated with short connects to the corresponding connection arcs in the crew pairing network, we can solve this pricing problem using the same technique as is used when solving the basic crew pairing problem. In other words, by simply modifying some of the input parameters, we can use the same pairing generator for both the basic and extended crew pairing models.

5.1.2 Generating Maintenance Routing Solution Columns

When solving *ECP*, column generation may also be used to identify additional maintenance solution columns. If we denote by σ the dual variable associated with the convexity constraint, then the reduced cost of a maintenance solution column is

$$-\sum_{t \in T} \beta_{ts} \gamma_t - \sigma.$$

[Recall that the cost of a maintenance column in ECP is 0.] The pricing problem can thus be written as

$$\min -\sum_{t \in T} \beta_{ts} \gamma_t$$

$$st$$

$$s \in S.$$

If the optimal solution to this pricing problem has objective value less than σ, then we have identified a new negative reduced cost column. Otherwise, we have established that no new NRC maintenance columns exist for the current dual values.

Let $R(s)$ denote the set of route strings found in maintenance routing solution s. Clearly

$$-\sum_{t \in T} \beta_{ts} \gamma_t = \sum_{r \in R(s)} \left(-\sum_{t \in T} \phi_{tr} \gamma_t \right).$$

We can therefore formulate the maintenance pricing problem as a basic maintenance routing problem in which the cost coefficient c_r associated with route string r is

$$-\sum_{t \in T} \phi_{tr} \gamma_t.$$

We can enhance this pricing problem by adding an appropriately small constant Δ to each of the duals γ_t. This ensures that if the short connects associated with one feasible solution are a subset of the short connects in another feasible solution, then the second solution will have an objective value that is lower than that of the first. [Note that the Δs must be removed from the duals before comparing the optimal solution to σ.] Thus, any new column being added to the restricted master will be UM.

5.1.3 When To Generate Columns

In theory, the column generation pricing problem is used to identify the *most* negative reduced cost variable not already included in the restricted master. This variable is then pivoted into the basis, new duals are computed, and a new pricing problem is solved. In practice, multiple *NRC* variables can often be generated in a single iteration. For example, in the basic crew pairing model, the algorithm used in solving the pricing problem can be modified to identify the n shortest paths. The corresponding n pairings can then be added to the restricted master collectively, and several pivots may occur before the next pricing problem is solved.

When solving *ECP*, the user may choose to generate two sets of columns at every iteration, by solving both the crew pairing and the maintenance routing pricing problems. There might be frequent iterations, however, in which *NRC* crew pairings are found, but no new *NRC* maintenance columns exist with respect to the current duals. It might therefore be computationally more efficient to solve the crew pricing problem at every iteration but the maintenance pricing problem only periodically. The frequency with which the maintenance pricing problem should be solved will also depend on how many maintenance columns were included in the original restricted master.

Alternatively, we would like to have a method for easily determining whether a *NRC* maintenance column exists for the duals associated with the current iteration. If no new column exists, then we can bypass the maintenance pricing problem for this iteration. The following section provides one such condition.

Sufficient Condition For Bypassing The Maintenance Pricing Problem

Let γ_t^i denote the dual value associated with the constraint for short connect t, given an optimal solution to the i^{th} iteration of the restricted master problem, and let σ^i denote the dual value associated with the convexity constraint for iteration i. Let X^i denote the solution to the i^{th} maintenance pricing problem, which is based on these duals. Let T be the set of all possible short connects and let T^i denote the set of short connects used in X^i. We claim the following:

Theorem 7 *If*

$$\gamma_t^i \leq \gamma_t^{i+1} \ \forall \ t \in T^i$$

and

$$\gamma_t^i \geq \gamma_t^{i+1} \; \forall \; t \in \{T \backslash T^i\}$$

then X^i is an optimal solution to the $i+1^{th}$ maintenance pricing problem. Thus, because this column is already included in the restricted master, it is not necessary to solve the $i+1^{th}$ maintenance pricing problem.

Proof. Let $\widehat{X}$ be any feasible solution to the pricing problem and let $\widehat{T}$ be the set of short connects used in $\widehat{X}$. By the optimality of X^i for the i^{th} pricing problem,

$$-\sum_{t \in \widehat{T}} \gamma_t^i - \sigma^i \geq -\sum_{t \in T^i} \gamma_t^i - \sigma^i.$$

We can re-write this as

$$-\sum_{t \in \{\widehat{T} \cap T^i\}} \gamma_t^i - \sum_{t \in \{\widehat{T} \backslash T^i\}} \gamma_t^i - \sigma^i \geq -\sum_{t \in \{T^i \cap \widehat{T}\}} \gamma_t^i - \sum_{t \in \{T^i \backslash \widehat{T}\}} \gamma_t^i - \sigma^i$$

and thus

$$\sum_{t \in \{\widehat{T} \backslash T^i\}} \gamma_t^i \leq \sum_{t \in \{T^i \backslash \widehat{T}\}} \gamma_t^i. \tag{5.1}$$

The reduced cost of $\widehat{X}$ with respect to the duals γ^{i+1} is

$$-\sum_{t \in \widehat{T}} \gamma_t^{i+1} - \sigma^{i+1}.$$

This can be re-written as

$$-\sum_{t \in \{\widehat{T} \cap T^i\}} \gamma_t^{i+1} - \sum_{t \in \{\widehat{T} \backslash T^i\}} \gamma_t^{i+1} - \sigma^{i+1}$$

$$\geq -\sum_{t \in \{\widehat{T} \cap T^i\}} \gamma_t^{i+1} - \sum_{t \in \{\widehat{T} \backslash T^i\}} \gamma_t^{i} - \sigma^{i+1} \tag{5.2}$$

$$\geq -\sum_{t \in \{\widehat{T} \cap T^i\}} \gamma_t^{i+1} - \sum_{t \in \{T^i \backslash \widehat{T}\}} \gamma_t^{i} - \sigma^{i+1} \tag{5.3}$$

$$\geq - \sum_{t \in \{\widehat{T} \cap T^i\}} \gamma_t^{i+1} - \sum_{t \in \{T^i \setminus \widehat{T}\}} \gamma_t^{i+1} - \sigma^{i+1} \tag{5.4}$$

$$= - \sum_{t \in T^i} \gamma_t^{i+1} - \sigma^{i+1},$$

where (5.2) and (5.4) are by supposition and (5.3) follows from (5.1). Thus, X^i is an optimal solution to the $i + 1^{th}$ pricing problem. ■

Corollary 8 *If, for any iteration $j \leq i$,*

$$\gamma_t^j \leq \gamma_t^{i+1} \ \forall \ t \in T^j$$

and

$$\gamma_t^j \geq \gamma_t^{i+1} \ \forall \ t \in \{T \setminus T^j\}$$

then X^j is an optimal solution to the $i + 1^{th}$ pricing problem. Thus, because this column is already included in the restricted master, it is not necessary to solve the $i + 1^{th}$ maintenance pricing problem.

5.2 Generating Lower Bounds

When using branch-and-bound, it is helpful to have strong lower bounds in order to establish optimality gaps. One way to find a lower bound for *ECP* is to solve a basic crew pairing problem in which all short connects are permitted. Given the resulting set of short connects used, which we denote by $F^{\mathbf{0}}$, we can then solve a maintenance routing problem to determine whether this crew solution is maintenance feasible. If so, we have an optimal solution to *ECP*. Otherwise, we have a lower bound. In this case, we can tighten this lower bound by adding a cut to the crew pairing problem that prohibits the current, maintenance-infeasible, solution. For example, if the current optimal solution is $Y^{\mathbf{0}}$ and this solution uses $N^{\mathbf{0}}$ pairings, then such a cut might take the form

$$\sum_{p \in P : y_p^0 = 1} y_p \leq N^{\mathbf{0}} - 1. \tag{5.5}$$

We can add this cut and re-solve the crew pairing problem to yield a tighter lower bound. By repeatedly solving this *Constrained Crew Pairing Problem (CCP)*, adding new cuts at each

iteration, we can tighten the lower bound as much as desired.

These cuts are not very efficient, however. For example, when we solve the $i+1^{th}$ iteration of *CCP*, the new solution might contain a different set of pairings from the i^{th} iteration but the same maintenance-infeasible set of short connects. Thus, a better cut would be one that prohibits the corresponding short connect set F^i rather than the set of pairings Y^i. Such a cut can be written as

$$\sum_{p \in P} \sum_{t \in F^i} \eta_{tp} y_p \leq |F^i| - 1. \tag{5.6}$$

We can tighten this new constraint even further. One inefficiency associated with constraint (5.6) is that we may generate several "nested" solutions in successive iterations that are all infeasible. For example, consider the case where $F^i = \{A, B, C, D\}$. Perhaps this solution is maintenance infeasible because short connects A and B are incompatible. These two short connects might be desirable to the crew pairing problem, however. Therefore, our next three iterations might yield solutions containing short connect sets $\{A, B, C\}$, $\{A, B, D\}$, and then $\{A, B\}$, all of which are maintenance infeasible.

We could have bypassed these intermediate iterations by prohibiting short connect set $\{A, B\}$ rather than the original set $\{A, B, C, D\}$. More generally, we want the new cut to represent a *minimally infeasible* subset $F^{i'}$of F^i, meaning that $F^{i'}$ is also maintenance infeasible but any proper subset of $F^{i'}$ is maintenance feasible. The new cut is then written as:

$$\sum_{p \in P} \sum_{t \in F^{i'}} \eta_{tp} y_p \leq |F^{i'}| - 1.$$

Just as we want the maintenance-feasible short connect sets that correspond to columns in *ECP* to be as *large* as possible, we want maintenance-infeasible short connect sets corresponding to constraints in *CCP* to be as *small* as possible. That is, in the same way that using maximal short connect sets allows us to minimize the number of maintenance columns needed in *ECP*, using minimally infeasible short connect sets in *CCP* allows us to minimize the number of maintenance cuts needed.

5.2.1 Generating Minimally Infeasible Short Connect Sets

By adding cuts based on minimally infeasible short connect sets we can improve the efficiency of our bounding approach. But how do we identify such minimally infeasible sets? Given a short connect set F^i from the i^{th} iteration of *CCP*, we first solve a variation of the maintenance routing problem in which our objective is to maximize the number of short connects in F^i included in the solution. If the optimal solution to this problem contains all of the short connects in F^i, then the crew pairing solution is maintenance feasible and therefore optimal for *ECP*. Otherwise, given that the short connect set is infeasible, we then seek the smallest maintenance-infeasible subset of F^i to use as a new cut in *CCP* – we refer to this problem as the *minimally infeasible set problem (MIS)*.

Consider a minimally infeasible subset $F^{i'}$. For every feasible solution to the maintenance routing problem, there must be at least one element of $F^{i'}$ *not* included in this solution. Otherwise, there would be a maintenance solution containing all short connects in $F^{i'}$, contradicting its infeasibility. To solve *MIS*, we let f_t be a binary decision variable indicating whether or not $t \in F^i$ is part of the minimally infeasible set, and use $T(s)$ to denote the set of short connects occurring in maintenance solution s. We can then formulate *MIS* as

MIS:

$$\min \sum_{t \in F^i} f_t$$

$$st$$

$$\sum_{t \in F^i \setminus T(s)} f_t \geq 1 \quad \forall\, s \in S$$

$$f_t \in \{0,1\} \qquad \forall t \in F^i.$$

For example, if the initial infeasible short connect set is $F^i = \{A, B, C, D\}$ and the maintenance routing problem has three feasible solutions, containing short connects $\{A, C\}$, $\{B, C, D\}$, and $\{A, C, D\}$ respectively, then the minimally infeasible set problem would be:

$$\min f_A + f_B + f_C + f_D$$

$$st$$

$$f_B + f_D \geq 1 \tag{5.7}$$

$$f_A \geq 1 \tag{5.8}$$

$$f_B \geq 1 \tag{5.9}$$

$$f_A,\ f_B,\ f_C,\ f_D\ \in \{0,1\}. \tag{5.10}$$

The optimal solution to this is $\{A, B\}$ and thus we add the cut

$$\sum_{p \in P} \sum_{t \in \{A,B\}} \eta_{tp} y_p \leq 1$$

to eliminate any crew pairing solutions containing both short connects A and B.

Note that constraint (5.7) is redundant – it is dominated by constraint (5.9). Note also that constraints (5.8) and (5.9) correspond to short connect sets $\{B, C, D\}$ and $\{A, C, D\}$, which are both maximal. This helps to highlight the fact that in the *MIS* model, we do not need to include one constraint for each feasible maintenance routing solution, but simply one for each maximal short connect set. This fact is important from an implementation standpoint, as it can significantly decrease the number of constraints required in the model.

Even given this decrease in constraints, it may be appropriate to use *cut generation* in solving *MIS*. In this approach, we begin by solving a restricted version of *MIS* which contains just the set of constraints corresponding to the *UM* columns currently included in the *ECP* model. Given the solution to *MIS*, we solve a maintenance routing problem attempting to identify a *UM* maintenance solution containing this short connect set. If we find such a *UM* maintenance solution, then the short connects in the restricted *MIS* solution are maintenance feasible and we have therefore identified a violated inequality. We add the cut corresponding to this *UM* short connect set to our restricted *MIS* and repeat. Note that we can also add a new column to *ECP*, as we have found a *UM* maintenance solution not currently contained in the restricted master.

If we *do not* find a *UM* short connect set containing the short connects in the current solution to the restricted *MIS*, this implies that the solution to the restricted *MIS* is feasible for the original *MIS* and therefore optimal. Thus, we have found a minimally infeasible short

connect set. We add the corresponding cut to *CCP* and solve to find a new lower bound on *ECP*.

We summarize the steps to *CCP* as follows:

1. Initialize *CCP* to be the basic crew pairing model in which all short connects are assumed to be feasible crew connections.

2. Solve the current version of *CCP*; the optimal objective value provides an updated lower bound on *ECP*. Let F^i denote the short connects used in this solution.

3. Solve a maintenance routing problem to find the *UM* short connect set using the maximum number of elements from F^i. If all elements from F^i are included, the current crew pairing solution is optimal and the algorithm terminates. Otherwise, we have established that F^i is a maintenance infeasible short connect set, and we can construct a new cut for *CCP*

4. Initialize an instance of *MIS*, corresponding to F^i, to contain one constraint for every *UM* short connect set in *ECP*.

5. Solve the current version of *MIS*; denote the solution by $F^{i'}$.

6. Solve a maintenance routing problem to find the *UM* short connect set using the maximum number of elements from $F^{i'}$. If one or more elements from $F^{i'}$ *are not* included, then $F^{i'}$ is maintenance infeasible and thus a minimally infeasible short connect set has been identified. Add the cut corresponding to $F^{i'}$ to *CCP* and return to step 2.

7. If all elements from $F^{i'}$ *are* included in the maintenance routing solution, then we have identified a violated cut for *MIS*. We add this cut to the restricted *MIS* and return to step 5. We also add the maintenance routing solution to *ECP*, given that by supposition it is a *UM* short connect set not currently included in *ECP*.

It is left to the user's discretion to determine how frequently to update the lower bound by generating a new cut and re-solving *CCP*.

It is interesting to note the synergies between *ECP* and *CCP*. We demonstrate this in Figure 5-1. This figure presents the output of each of the sub-problems that we have described and how this output may be used. *ECP* and *CCP* generate increasingly tighter upper and lower

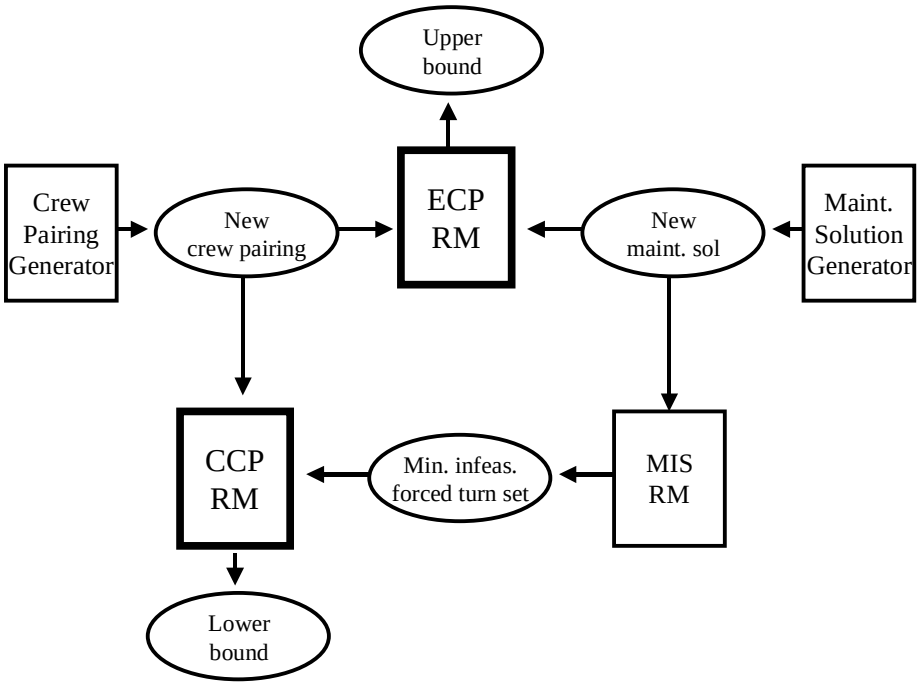


Figure 5-1: Model synergy

bounds, respectively, allowing us to determine when the optimality gap is sufficiently small to terminate the algorithm. Crew pairings generated when solving *ECP* can also be added to the restricted master for *CCP* and vice versa. Maintenance solutions generated for *ECP* lead to new cuts for *MIS* and new cuts for *MIS* lead to new columns for *ECP*. Furthermore, the fact that there is a one-to-one correspondence between maintenance columns in *ECP* and cuts in *MIS* means that the bounds on the number of maintenance columns required by *ECP* can be applied to the number of cuts in *MIS* as well. In addition, solutions to *MIS* lead to new cuts for *CCP*. Finally, note that the user has complete control over how to best utilize this collection of models.

5.3 Permitting Crews To Fly Multiple Sub-Fleets

Throughout this paper, we have assumed that there is a one-to-one correspondence between maintenance routing problems and crew pairing problems. In practice, it is often the case that a crew is qualified to fly several different aircraft types (we'll refer to these as sub-fleets) within a

given fleet family. Therefore, we must solve a number of different maintenance routing problems in association with this single crew pairing problem. It is not difficult to incorporate this into *ECP*, and the benefits described in the previous sections still exist.

Denoting the set of sub-fleets by M and the set of solutions to the maintenance routing problem for sub-fleet m by S^m, we can write the Extended Crew Pairing Model with Sub-Fleets (*ECPSF*) as

ECPSF:

$$\min \sum_{p \in P} c_p y_p \tag{5.11}$$

$$st$$

$$\sum_{p \in P} \delta_{fp} y_p = 1 \qquad \forall\ f \in F \tag{5.12}$$

$$\sum_{m \in M} \sum_{s \in S^m} \beta_{ts} x_s^m - \sum_{p \in P} \eta_{tp} y_p \geq 0 \qquad \forall\ t \in T \tag{5.13}$$

$$\sum_{s \in S^m} x_s^m = 1 \qquad \forall\ m \in M \tag{5.14}$$

$$x_s^m \in \{0,1\} \qquad \forall\ s \in S, m \in M \tag{5.15}$$

$$y_p \in \{0,1\} \qquad \forall\ p \in P. \tag{5.16}$$

Note that we can still relax the integrality of the x variables because short connects are specific to a single sub-fleet. Furthermore, the properties of uniqueness and maximal sets can still be used to minimize the number of maintenance columns required. Finally, we note that the fact that we can partition the flights by sub-fleet with respect to the maintenance routing solution columns can have a significant impact on tractability. This is because we only need to worry about maintenance solutions to the smaller problems rather than solving the maintenance routing problem over the full complement of flights included in the crew pairing network.

Problem Instance	ECP Solution	Lower Bound	Optimality Gap
A	31,396.10	31,396.10	0.0%
B	25,498.60	25,076.60	1.7%

Table 5.1: Proof-of-concept

5.4 Proof-of-Concept

Given that a significant optimality gap has been shown to exist between the sequential and integrated approaches (Cordeau et al. [31]), we wanted to get some sense of how difficult it is to capture a significant portion of this potential for improvement by using *ECP.* as a heuristic To test the efficacy of our approach, we performed a limited experiment in which a small number of maintenance columns were generated in advance, all feasible crew pairings were generated, and we solved the resulting problem (without using column generation to identify additional maintenance solutions). The purpose of the experiment was to provide a proof-of-concept of our methodology, rather than exhaustive computational results, in order to motivate airlines and other researchers to evaluate these ideas using their own state-of-the-art solvers.

We considered two actual airline problem instances, both containing approximately 125 flights. We used the following strategy for generating maintenance columns in both instances. First, we generated the ten *UM* maintenance-feasible short connect sets of largest cardinality. Then, we identified those short connects which did not appear in any of these columns. For each of these short connects, we generated the largest cardinality *UM* maintenance-feasible short connect set which contains it, thus ensuring that each short connect was included in at least one maintenance solution column and therefore had the potential to be included in a crew pairing. As a result of this approach, problem instance A had 16 maintenance columns and problem instance B had 20 maintenance columns. We then solved *ECP* using a complete enumeration of all feasible pairings.

In order to analyze the quality of our solution, we also generated a lower bound on the optimal objective value by computing the optimal crew pairing solution when *all* short connects are permitted, independent of maintenance feasibility. Table 5.1 presents our results.

This table demonstrates that with no more than twenty maintenance solution columns, we are within two percent of the optimality gap in both of these problem instances. We believe that this is largely due to the fact that the number of short connects used in an optimal integrated

solution is often much smaller than the total number of short connects. For example, problem instance A had 58 possible short connects. The 16 short connect columns that we generated contained on average just over 38 short connects each. The optimal solution only used 9 short connects. In problem instance B there were 68 possible short connects. The 20 short connect columns that we generated contained on average 36.5 short connects each. The optimal solution only used 10. Thus, by using maximal sets, we have a good chance of capturing the optimal set of short connects even with only a small number of short connect columns.

Chapter 6

Conclusions

In this part of the thesis we have presented the Extended Crew Pairing Model. This is a new approach to solving an important real-world problem – the integration of maintenance routing and crew scheduling. We have advanced the existing research in this area by focusing on three main goals: guaranteeing maintenance feasibility; providing user flexibility for trading off between solution time and quality; and leveraging the fact that only a fraction of the maintenance routing decisions are relevant to the crew pairing problem. We have satisfied these goals in a model that has no more binary decision variables than the basic crew pairing model alone. Furthermore, this model is flexible in that it can directly incorporate new advances in maintenance solvers and pairing generators; it can easily be extended to address additional maintenance routing constraints; and it can be used in a column generation framework. We have also demonstrated that the LP relaxation of *ECP* is tighter than that of the basic integrated approach, and we have proposed one method for generating lower bounds on the optimal objective value. Finally, we believe that our model has greater potential for extendibility to include additional airline planning decisions.

Part III

Composite-Variable Modeling For Service Parts Logistics

Chapter 7

Introduction

When companies purchase machines such as computers, medical equipment, or aircraft, they often purchase a *service contract* as well. Such a contract specifies that, should the machine break down, a technician and any necessary repair parts will arrive on-site within a fixed period of time. Depending on how critical the machine is to a company's operations, this time period may be as little as two hours. *Service parts logistics (SPL)* is the study of how to manage the repair parts – how many of each part to stock, where to stock them, and how to transport them – in order to meet these tight time commitments. The repair parts may be managed by an internal organization, a third-party provider, or some combination of the two.

According to one study (Cohen et al. [30]), after-sales service revenues can be as high as 30% of product sales, with the majority of this revenue coming from service contracts. Furthermore, good SPL can boost customer loyalty and thus increase future sales.

In developing a good service parts plan, SPL providers have two conflicting objectives. In order to minimize the likelihood of a service failure, as well as to minimize the cost of transporting parts from the warehouse to the customer in a timely fashion, parts should be kept in a wide variety of locations and in significant quantities. Conversely, inventory and warehousing costs are minimized by consolidating small numbers of parts in a limited number of locations. The stochasticity of part demands and the large number of distinct parts (also known as *SKUs*) add to the complexity of the planning process. Furthermore, the short life cycle of many products results in a proliferation of parts with infrequent demands.

There are many challenging problems in SPL – in network design (how many echelons,

how many warehouses and where to locate them), in parts management within this network (how many of each part to stock and where, what re-order policy to use), and operationally (how to meet a specific part demand). Further complicating these problems are issues such as *multiple part failures* (that is, a given machine failure may correspond to multiple part failures, or it may be unclear in advance which repair part is needed), *repairable parts* (damaged parts can sometimes be returned for repair and put back into the system), and *recycling and disposal requirements* (damaged parts might be required to be handled in a special way after replacement). As a result, service parts logistics provides a rich problem area for research.

SPL research dates back at least to the late 1960's and the work of Sherbrooke ([54]). The extensive subsequent research includes work in characterizing the performance of given repair parts models (Graves [40]), determining inventory levels for a given parts network (Cohen et al. [29]), addressing the issue of multiple dependent failures (Cheung and Hausman [22]), considering the post-repair phase (Muckstadt and Isaac [52]), and permitting emergency lateral transshipments (Axsäter [7]). There are also a number of papers discussing real-world industry applications, including Saturn (Cohen et al. [27]) and IBM (Cohen et al. [28]). This is just a small sample of the literature – we refer the reader to Wang [59] and Alfredsson and Verrijdt [1] for extensive surveys.

We contribute to the SPL literature by considering how composite-variable models can improve tractability for some of these difficult problems. We begin by considering the *high-cost, low-demand stocking problem* in Chapter 8, presenting a basic modeling approach that allows us to easily solve problem instances of realistic size. In Chapter 9 we expand the scope of this problem to incorporate warehouse capacity constraints, and show that a basic modeling approach becomes intractable in many instances. We present a composite-variable modeling approach for this problem in Chapter 10, with computational results to demonstrate its improved tractability. We conclude in Chapter 11 by discussing how the composite-variable modeling framework might be extended further to address more complex problems within service parts logistics.

Chapter 8

The High-Cost, Low-Demand Stocking Problem

Much of the literature in service parts logistics looks at how to stock a particular SKU, so as to ensure some specified level of service for that repair part while minimizing costs. We begin by examining this problem for the special case of a part that is both high cost (in some cases, as high as tens or hundreds of thousands of dollars per part) and very low demand. By low demand, we mean that the probability of more than one failure of that particular part *system wide* within the replenishment lead time is sufficiently small, according to the user's risk tolerance. In other words, should one installation of that part fail, it is highly unlikely that a second failure of the same part will occur anywhere within the system before the first repair part is replenished.

Given that a repair part satisfies this definition of low demand, we no longer need to consider *how many* of this part to stock, but simply *where* to stock it. That is, we need to choose a minimum-cost set of warehouses in which to stock the part such that for every location where the part is installed at least one warehouse *in range* of that location must be included in this set. (A warehouse is in range of a location if the transportation time between the two points does not exceed the time window specified by the service contract)

We refer to this as the *high-cost, low-demand stocking problem*, depicted with an example in Figures 8-1 and 8-2. Figure 8-1 shows five warehouses (represented by squares) and nine

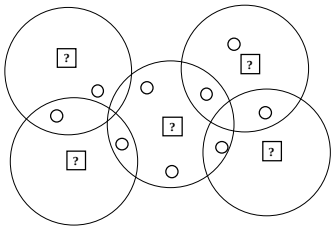


Figure 8-1: Potential stocking locations

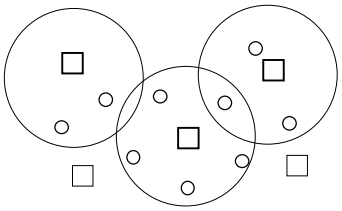


Figure 8-2: Feasible covering

installation points for the part (represented by small circles). A ring around each warehouse indicates the set of installations in range of the warehouse. As seen in Figure 8-2, by stocking the part in only three of the five warehouses, we can still ensure that each installation has access to a repair part.

This problem can be modeled as a *set covering problem*, with one binary decision variable x_w for each warehouse w. If $x_w = 1$ then warehouse w stocks the part in question, else not. We have one cover constraint for each installation of that part, stating that the installation must be in range of at least one warehouse in which the part is stocked.

We define the following notation:

- W is the set of warehouses.
- I is the set of locations where the part is installed.
- $r(i)$ is the set of warehouses that can serve installation i.
- c_w is the cost of stocking the part in warehouse w.

The high-cost, low-demand stocking model ($HCLD$) can then be formulated as:

$$\min \sum_{w \in W} c_w x_w$$

$$st$$

$$\sum_{w \in r(i)} x_w \geq 1 \qquad \forall i \in I$$

$$x_w \in \{0, 1\} \qquad \forall w \in W.$$

This model minimizes the cost of installing the part in the chosen warehouse set, subject to the constraint that every installation of that part must have access to at least one warehouse where the part is stocked within the time range specified by the service contract.

There is an extensive body of literature on the set covering problem, providing both exact and heuristic solution methods (for a survey, see Ceria et al. [21]). In our experience, for problem instances of a realistic size (on the order of one hundred warehouses and one thousand demand points) we found that even the most basic implementation of the model yielded solutions in a matter of seconds or less.

Although this problem has real-world relevance in that it can help limit the number of high-cost, low-demand parts which must be stocked in a network, it is nonetheless quite limited in scope. We do not consider parts with higher demand patterns (for which stocking decisions are typically more challenging), nor do we consider the interactions between parts, such as those caused by shared resources within the network. What is the impact on the tractability of the *HCLD* model as we expand our problem scope?

Chapter 9

Extending The Problem Scope

9.1 The Capacitated High-Cost, Low-Demand Stocking Problem

A key reason why we are able to solve the high-cost, low-demand stocking problem so easily is that we treat each SKU independently. Ideally, however, we would like to consider the interactions *between* parts within our decision making. To gain some sense of how this would impact the tractability of the basic modeling approach, we next consider a variation of the original problem, in which we continue to focus only on high-cost, low-demand parts, but also take into account warehouse capacity constraints. We refer to this as the *capacitated high-cost, low-demand stocking problem.*

9.2 A Basic Modeling Approach

We can extend the *HCLD* model to incorporate warehouse capacity by defining one binary decision variable x_{sw} for each SKU s and warehouse w. If $x_{sw} = 1$ then warehouse w stocks part s, else not. We still have one cover constraint for each installation of each part – that is, the model contains one *HCLD* problem for each SKU. We then link these individual problems together with a set of capacity constraints, one per warehouse.

In addition to the notation used previously, we define the following:

- S is the set of SKUs.

- $i(s)$ is the set of installations for SKU s.
- c_{sw} is the cost of stocking part s in warehouse w.
- b_s is the weight of part s.
- CAP_w is the capacity for warehouse w.

The basic capacitated model ($CHCLD$-B) can then be formulated as:

$$\min \sum_{s \in S} \sum_{w \in W} c_{sw} x_{sw}$$

$$st$$

$$\sum_{w \in r(i)} x_{sw} \geq 1 \qquad \forall s \in S,\ i \in i(s) \tag{9.1}$$

$$\sum_{s \in S} b_s x_{sw} \leq CAP_w \qquad \forall w \in W \tag{9.2}$$

$$x_{sw} \in \{0, 1\} \qquad \forall s \in S, w \in W.$$

The objective function minimizes the total cost associated with assigning parts to warehouses. Constraint set (9.1) ensures that every installation of every part has access to a warehouse stocking that part. Constraint set (9.2) ensures that the combined weight of the parts stocked in a given warehouse does not exceed the capacity of that warehouse.

9.3 Computational Experience

To assess the tractability of this model, we tested it on several problem instances. These problem instances are loosely based on the network of UPSLG SPL, the service parts logistics division

of the UPS Logistics Group. However, we have varied the parameters (warehouse capacities and part costs, weights, and distributions) in order to assess performance under a wide array of circumstances. We have also varied problem size to assess how the model scales. The small problem instances (S) have 10 SKUs, 33 warehouses, and approximately 5,000 demand points. The medium instances (M) have 100 SKUs, 99 warehouses, and 1,000 demand points. The large instances (B) have 500 SKUs, 99 warehouses, and 1,000 demand points.

We present results in Table 1. We indicate by "Solved" or "Not Solved"whether or not we were able to solve the instance. The next two columns of the table provide the number of variables and constraints in the model after being pre-processed in CPLEX. The final column states the number of nodes explored in the branch-and-bound tree in finding an optimal solution. All computational work was done on a HP9000 Model D370 running HPUX 10.20 and CPLEX 6.5 with 256 MB of memory.

We were able to solve almost all of the small problem instances to optimality using *CHCLD-B* (in *S2*, CPLEX aborted after solving one million nodes; in *S3*, CPLEX ran out of memory at approximately 800,000 nodes). Of the medium problem instances, however, only *M16* could be solved (this is an infeasible instance, which *CHCLD-B* was able to identify easily). An optimal solution could not be found for the remaining medium instances – in most cases, due to insufficient memory. In all three of the large instances, not only did we fail to find an optimal solution, but we could not even solve the initial LP relaxation – CPLEX terminated during the pre-processing stage due to insufficient memory.

Clearly, two major obstacles to tractability with this model are the very large number of constraints (an instance with 500 parts and an average of 1000 demands per part will have half a million constraints!) and the weakness of the LP relaxation, resulting in long times in branch-and-bound. In the next section, we use a composite-variable modeling approach to address these deficiencies.

Instance	**Solve?**	#	#	#
		Col.'s	**Rows**	**Nodes**
S1	Solved	320	2250	10
S2	Not solved	320	2282	NA
S3	Not solved	320	2282	NA
S4	Solved	320	2282	1
S5	Solved	320	2250	14
S6	Solved	320	2282	1,200
S7	Solved	320	2282	412,939
S8	Solved	320	2282	1
S9	Solved	320	2282	1
S10	Solved	320	2250	10
S11	Solved	320	2250	10
S12	Solved	320	2263	1
S13	Solved	320	2270	2,800
S14	Solved	320	2267	174,400
S15	Solved	320	2265	100
M16	Solved	9600	16396	1
M18	Not solved	9312	15839	NA
M19	Not solved	8832	15032	NA
M20	Not solved	8148	4446	NA
M23	Not solved	8501	4827	NA
B24	Not solved	49,500	500,099	NA
B25	Not solved	49,500	500,099	NA
B28	Not solved	49,500	500,099	NA

Table 9.1: Results for CHCLD-B

Chapter 10

A Composite-Variable Modeling Approach

Composite-variable modeling has been useful in solving many challenging transportation and logistics problems for which a basic modeling approach has proven intractable. Examples of successful use include Armacost and Barnhart ([6]), Lohatepanont and Barnhart ([49]), and Barnhart, Farahat, and Lohatepanont ([10]), in which the benefits of composite modeling have included: eliminating non-linearities in the objective function or constraints, decreasing the number of constraints, and strengthening the quality of the LP relaxation.

There are a number of ways in which we might define a composite variable within the capacitated high-cost, low-demand stocking problem. We have chosen to group together all decisions associated with a particular SKU. Specifically, we define a composite variable x_{sg} to take value one if part s is stocked in exactly those warehouses in group g and zero otherwise. Note that we only define a variable x_{sg} if group g satisfies all the cover constraints for part s. Before stating this composite model, we introduce the following additional notation:

- $g(s)$ is the set of warehouse groups that provide full coverage for SKU s.
- c_{sg} is the cost of stocking part s in warehouse group g.

The composite model ($CHCLD$-C) can then be formulated as:

$$\min \sum_{s \in S} \sum_{g \in g(s)} c_{sg} x_{sg}$$

$$st$$

$$\sum_{g \in g(s)} x_{sg} = 1 \qquad \forall s \in S \tag{10.1}$$

$$\sum_{s \in S} \sum_{\substack{g \in g(s): \\ w \in g}} b_s x_{sg} \leq CAP_w \qquad \forall w \in W \tag{10.2}$$

$$x_{sg} \in \{0, 1\} \qquad \forall s \in S, g \in g(s).$$

The objective minimizes the total cost of assigning parts to warehouses. Constraint set (10.1) ensures that for each SKU, we choose exactly one group of warehouses in which to stock it. Constraint set (10.2) ensures that for each warehouse, the collective weight of the parts assigned to it does not exceed its capacity.

10.1 Benefits Of The Composite Model

As demonstrated in Table 1, a key deficiency in the basic modeling approach (*CHCLD-B*) is its large number of constraints. Our composite variable definition allows us to eliminate all of the cover constraints associated with a particular SKU, replacing them with a single constraint that selects exactly one warehouse group for that SKU. Hence, we have removed the need for cover constraints within the model by capturing them within the variable definition. This has a marked impact on the size of the problems. In particular, comparing Tables 1 and 2, we see that the number of constraints in the large problem instances drops from more than half a million to fewer than one thousand. This has significant impact on our ability to solve the LP

relaxations quickly within branch-and-bound.

A second benefit of the composite model is that it provides a tighter LP relaxation, as we discuss in Part IV. This improvement is common to many composite modeling approaches and can be seen in the literature from several other applications, including Armacost and Barnhart ([6]) and Barnhart, Farahat, and Lohatepanont ([10]). This fact is also discussed in greater theoretical detail in Vanderbeck ([57]). The impact of this on computational performance is substantial. As we see in Table 2, for many problem instances, the LP relaxation of the composite model has an integer solution and therefore branching is not required. In the remaining instances, an optimal integer solution is found after solving no more than 500 LP relaxations in the branch-and-bound tree. In contrast, it is often necessary to solve hundreds of thousands of LP's in the basic modeling approach, causing memory issues and intractability.

10.2 Controlling The Number Of Variables In *CHCLD-C*

The *CHCLD-C* model benefits from a tighter LP relaxation and significantly fewer constraints than the *CHCLD-B* model. These benefits come at a cost – as we group together decisions, we have the potential for an exponential increase in the number of binary variables in the model. In practice, we have observed (see Table 2) that solving the composite model actually requires *fewer* binary variables than the basic model. We control the number of variables in three ways.

The first is through *feasibility*. For a given SKU, we do not define a variable for *every* combination of warehouses, but only for those groups of warehouses that cover all demands for that SKU.

The second is through *dominance*. Consider, for example, an SKU s for which warehouses A, B, and C provide full coverage. Thus, we would define a composite variable $x_{s\{A,B,C\}}$. Any superset of these warehouse ($\{A,B,C,D\}$, $\{A,B,C,E\}$, $\{A,B,C,D,E\}$, and so on) also provides full coverage and therefore satisfies our definition of a valid variable. However, such variables will always have value zero in an optimal solution – they have higher cost and use more capacity without providing any additional benefits. Therefore, they do not need to be included in the model. More generally, we only need to define a variable for every *minimally feasible* warehouse set – that is, a set of warehouses that satisfies the cover constraints but

Instance	**Solve?**	#	#	#
		Col.'s	**Rows**	**Nodes**
				1
S1	Solved	20	43	1
S2	Solved	90	43	3
S3	Solved	260	43	1
S4	Solved	260	43	1
S5	Solved	20	43	1
S6	Solved	84	43	1
S7	Solved	163	43	1
S8	Solved	258	43	1
S9	Solved	267	43	1
S10	Solved	20	43	1
S11	Solved	20	43	1
S12	Solved	67	43	16
S13	Not solved	NA	43	NA
S14	Solved	132	43	210
S15	Solved	79	43	34
M16	Solved	17,200	199	1
M18	Solved	297	199	1
M19	Solved	594	199	129
M20	Solved	351	199	7
M23	Solved	933	199	297
B24	Solved	3,432	599	1
B25	Solved	2,988	599	498
B28	Solved	1,000	599	1

Table 10.1: Results for CHCLD-C

for which any subset of those warehouses is infeasible. Limiting the model to consider only minimally feasible warehouse sets further reduces the number of variables in the model.

The third is through *branch-and-price*. The branch-and-price algorithm, surveyed by Barnhart et al. [13], is a variation of branch-and-bound, in which we use column generation to solve the LP relaxations. This allows us to leverage the dual information in identifying promising warehouse groups, rather than enumerating them all explicitly.

10.2.1 Implementing Branch-and-Price

Two critical issues in successfully implementing branch-and-price are in identifying the *pricing problem* for the column generation and the *branching strategy* within branch-and-bound.

Pricing Problem

Within column generation, we use a pricing problem to identify negative reduced cost variables to pivot into the basis, rather than evaluating all variables explicitly. In *CHCLD-C*, a variable represents the decision to stock part s in warehouse group g, where g by definition must satisfy the cover constraints for part s. The reduced cost associated with this variable is

$$\sum_{w \in g} c_{sw} - \sigma_s - \sum_{w \in g} \pi_w,$$

where σ_s is the dual variable associated with the convexity constraint for part s and π_w is the dual variable associated with the capacity constraint for warehouse w. Re-arranging this to

$$\sum_{w \in g} (c_{sw} - \pi_w) - \sigma_s,$$

we can write the pricing problem for part s as

$$\min \sum_{w \in W} (c_{sw} - \pi_w)\, x_w$$

$$st$$

$$\sum_{w \in r(d)} x_w \geq 1 \qquad \forall i \in i(s) \tag{10.3}$$

$$x_w \in \{0, 1\}.$$

The objective function incorporates penalties for stocking the part in warehouses with binding capacity constraints. Constraint set (10.3) ensures that the warehouse set satisfies the coverage requirements for the part.

This pricing problem is a simple variation of *HCLD*, with dual information incorporated into the objective function to discourage the use of warehouses with binding capacity constraints. As noted in Chapter 8, in our experience this problem has been easy to solve in most instances.

If the objective value for the solution to the pricing problem for part s is strictly less than σ_s, then we have identified a negative reduced cost variable that can potentially improve the current solution when added to the restricted master. We re-solve the augmented restricted master and then re-solve the pricing problem with the updated dual information. If in a given iteration we solve a separate pricing problem for each SKU and do not identify *any* negative reduced cost columns, then the algorithm terminates with an optimal solution to the current LP relaxation.

Note that the issue of dominance is incorporated in this pricing problem, so that we never need to consider non-minimal sets. Given that

$$c_{sw} > 0$$

and

$$\pi_w \leq 0,$$

the reduced cost of any superset of g will never be less than the reduced cost of g. Therefore, the pricing problem will always identify minimally feasible warehouse groups.

Branching Strategy

In using branch-and-bound to solve a problem with binary variables, conventional branching is based on *variable dichotomy*. Given an LP solution in which a binary variable has fractional value, we create one problem in which we set that variable to zero and another in which we set it to one. Such a branching strategy is not likely to work well in the composite approach *CHCLD-C* for two reasons.

First, this strategy produces a very imbalanced tree. Given a part s and a warehouse group g, when we set $x_{sg} = 1$, we are fully determining the stocking of part s, where as in setting $x_{sg} = 0$, we are ruling out only one of potentially many stocking choices for part s.

The second, and perhaps more important, reason not to use this conventional branching strategy is its impact on the pricing problem. For each added constraint of the form $x_{sg} = 0$, we must take into account the dual variable associated with forbidding this warehouse group/part pairing. We cannot implement such a group-based dual in our current, part-based pricing problem.

In order to continue using our original pricing problem throughout the branch-and-bound tree, we instead branch on part-warehouse pairs. In this approach, similar to that used in many applications of branch-and-price, we branch on the more elemental decision of: *part s {is / is not} stocked in warehouse w*. This branching strategy has a number of benefits.

First, it produces a more balanced tree. Second, it results in a tree that is less deep – there are fewer potential branching decisions. Third, it is easy to incorporate within the pricing problem. For a node in which we add the constraint "part s *must* be stocked in warehouse w," when we solve the pricing problem for part s, we set $x_w = 1$. For a node in which we add the constraint "part s must *not* be stocked in warehouse w," we set $x_w = 0$ in the pricing problem for part s. Finally, note that in addition to allowing us to maintain the same pricing problem structure at every node of the branch-and-bound tree, this branching strategy has an added benefit: the pricing problems become smaller as you move deeper into the tree, when memory issues are more pressing.

Chapter 11

Solving Additional Problems In *SPL*

The capacitated high-cost, low-demand stocking problem is still limited in its scope. Furthermore, it can be difficult to estimate the parameter CAP_w, given that warehouse capacity is used to hold other types of repair parts as well as those that are high-cost and low-demand. We conclude Part III by discussing ways in which the structure of the *CHCLD-C* model might be extended in order to address problems of greater scope.

Additional Part Characteristics

So far, we have only considered the inventory cost of a part and how much warehouse capacity it uses. In more extensive models, we might want to take into account other characteristics of a part, such as the cost of transporting it to demand points or special handling requirements. Doing so in a basic modeling approach might require extensive modifications and new solution methodologies. In a composite approach, we might be able to address these enhancements by instead modifying the variable definition.

For example, suppose we want to take into account transportation costs in our capacitated high-cost, low-demand stocking problem. This enables us to recognize those instances where the cost of stocking a part in additional locations might be outweighed by a decrease in transportation costs to some customer locations. To incorporate these costs in the basic model, we need a new set of binary variables to capture the assignment of customers to warehouse locations. We also need constraints for each part ensuring that each customer is assigned to a warehouse stocking that part.

Within the composite modeling framework, we instead capture transportation costs simply by modifying the costs associated with the composite variables. For example, for a given part s and warehouse group g, we identify the transportation cost component by matching each customer requiring part s to the warehouse in group g that has the lowest transportation cost while still ensuring delivery within the specified time window. Observe that we can capture this new cost structure in the branch-and-price algorithm by modifying the pricing problem. Instead of solving a set covering problem to identify a new composite variable, we solve a facility location problem. The master problem (i.e. the convexity and capacity constraints) remain unchanged, as does the branching strategy.

Other Classes Of Parts

So far, we have only considered parts which are high-cost and low-demand. Clearly, different (and typically more difficult) stocking policies will exist for parts with different demand characteristics. Much of the SPL literature focuses on these more challenging stocking questions, often for a single part. These models can be quite complex, for example, because of non-linearities in both the objective function and the constraints.

Within a composite modeling framework, we can consider part interactions by again imbedding these complexities within the variable definition, leveraging what we know from the literature on how to solve the stocking problem for a particular part, and capturing the interaction between parts within the master problem. For example, we might expand the capacitated model to include parts that are low-demand. Different stocking policies might exist for overnight, four-hour, and two-hour service constraints. The literature on stocking individual parts could provide state-of–the-art characterization of the pricing problem for these part classes. We could continue to use our existing master problem ($CHCLD$-C) and solution methodology, simply adjusting the weighting coefficients in the capacity constraints in order to account for the quantity of a part stocked in a particular warehouse.

Part Interactions

In some cases, the stocking decisions for one part are tied to the stocking decisions for another part. For example, it is not always clear in advance which part a technician will need; therefore,

certain parts tend to be requested together. Or it might make sense when replacing one part to also replace other parts with similar failure patterns, or for ease of access. It might therefore be beneficial to stock such parts in the same location. To capture part interactions such as these, we might extend the definition of a composite to encompass the collective stocking decisions for more than one part.

When defining the model, it is essential to find the appropriate level of grouping. As more parts are grouped together within a composite variable, the accuracy of the model can improve, but this typically comes at the cost of an increase in the number of variables and decreased tractability.

Simulation

Much of the difficulty in determining part stocking policies stems from the variability of demand. In fact, it is not always possible to provide exact expressions for the cost of a particular stocking decision or to determine whether a stocking decision satisfies the service requirements. Simulation may be useful in such cases.

As an example of how we might incorporate simulation without a composite-variable modeling approach, we again consider the case of including high-demand parts in the warehouse-constrained stocking problem. Given a stocking option for a particular part (that is, the number of copies of that part to stock in each warehouse, a replenishment policy, and so forth), it is not always possible to determine through closed form expression the cost of this option, nor whether it satisfies the service requirements. To do so, we might instead use simulation. If service requirements are met, we would create a corresponding composite variable. By addressing the stochasticity at the variable level rather than the master level, we can continue to use the existing master problem, which simply contains convexity and capacity constraints, and the sophisticated tools that we have at our disposal for solving such integer programs. As new advances are made in the literature in terms of individual part stocking decisions, we can incorporate these advances in our composite model by updating the pricing problems, without having to modify the master problem.

Heuristics

Due to their complexity, many SPL problems cannot be solved to optimality; instead, much of the literature focuses on developing quality heuristics. In practice, even exact models are often solved heuristically rather than to optimality, in order to find solutions more quickly. Composite-variable modeling can be helpful in providing the user with control over this trade off between solution time and quality. By controlling the number of composites included in a model, the user can limit the solution time. Furthermore, user expertise, in conjunction with dual information, can be leveraged in identifying composites which are likely to be part of a good solution.

Part IV

Insights For Composite-Variable Modeling

Chapter 12

Composite-Variable Definition

As we have seen in the examples of Parts II and III, composite-variable modeling can sometimes improve tractability for challenging problems in transportation and logistics. Composite-variable models can also be useful for integrating multiple stages within a planning environment, developing better heuristics for solving large problems in real time, and providing users with greater control in trading off solution time and quality.

The success of a composite-variable model largely depends on the manner in which we define the composites. In this chapter, we review the composite-variable definitions used in the examples of Parts II and III. We then introduce an alternative composite-variable model for the SPL problem which does *not* improve tractability, and highlight some of the lessons learned from this experience. We conclude the chapter by identifying some of the key trade-offs to be made when developing composite-variable models.

12.1 Integrated Airline Planning Example

In the Extended Crew Pairing Model of Part II, we define a composite based on the *complete set* of maintenance routing decision variables $\{x_r\}$, where x_r takes value one if maintenance route string r is included in the solution and zero otherwise. We replace the set $\{x_r\}$ with the set $\{z_s\}$, where z_s is a composite variable that takes value one if we select maintenance solution s (recall that a maintenance solution is simply a feasible assignment of binary values to the routes represented by the set $\{x_r\}$)and zero otherwise.

As a result of this re-formulation, we are able to:

- decrease the number of constraints (by eliminating the original maintenance routing constraints, which are now implicit in the variable definition),
- tighten the LP relaxation,
- control the number of composite variables through dominance (we need not consider *all* maintenance routing solutions, but only those associated with unique and maximal short connect sets),
- reduce the number of binary variables (by recognizing that we can relax the integrality of the composite variables),

and

- exploit the fact that only a fraction of the maintenance routing information is relevant to the crew scheduler.

We can solve this re-formulated model using branch-and-price, taking advantage of the fact that we can use a modified version of the existing maintenance routing solver to generate negative reduced cost composite variables. Furthermore, this pricing problem (with appropriate cost coefficients) will always yield a dominant column (that is, a maximal set of short connects). We do not need to develop any special branching strategies beyond those used in crew pairing optimization, because the composite maintenance variables will always have integer values once we achieve integer values for the crew variables.

12.2 Service Parts Logistics Example

In the composite model for the capacitated high-cost, low-demand stocking problem of Part III, for each SKU $\widehat{s}$ we define a composite based on all of the associated warehousing decisions $\{x_{\widehat{s}w}\}$, where $x_{\widehat{s}w}$ takes value one if SKU $\widehat{s}$ is stocked in warehouse w and zero otherwise. For a given SKU $\widehat{s}$ we replace the set of decisions $\{x_{\widehat{s}w}\}$ with the set $\{x_{\widehat{s}g}\}$, where g is a group of warehouses that collectively satisfy the demand for coverage of SKU $\widehat{s}$. Thus, $x_{\widehat{s}g}$ takes value one if SKU $\widehat{s}$ is assigned to exactly those warehouses in group g and zero otherwise.

As a result of this re-formulation, we are able to:

- decrease the number of constraints (by eliminating the original cover constraints, which are now implicit in the variable definition),
- tighten the LP relaxation,

and

- control the number of composite variables through dominance (we only need to consider minimal warehouse groups).

In solving this re-formulated model with branch-and-price, we leverage the fact that we can solve the uncapacitated version of this problem easily for a single SKU, and that this uncapacitated problem can be used to identify negative reduced cost columns. Furthermore, columns identified in this way will always be dominant (that is, they will always correspond to minimal warehouse sets). For our branching strategy, we branch on the original elemental decisions x_{sw}, requiring in one branch of the tree that SKU s *must* be stocked in warehouse w and in the other branch that it must *not*. As a result, we are able to use the same pricing problem throughout the tree. This branching strategy also results in a less deep and more balanced tree, and has the further advantage of yielding progressively smaller pricing problems as we move deeper into the tree, where memory management is of greater concern.

12.3 An Alternative SPL Formulation

Of course, composite-variable modeling does not always improve tractability and in many cases will make a problem significantly *less* tractable. Valuable insights can often be gained by examining such failures.

Consider, for example, an alternative SPL model in which we form composites by grouping together SKU decisions for a given warehouse, rather than warehouse decisions for a given SKU. For each warehouse $\overline{w}$, we construct a composite from the variable group $\{x_{s\overline{w}}\}$. For a given warehouse $\overline{w}$ we replace the set of decisions $\{x_{s\overline{w}}\}$ with the set $\{x_{\overline{w}c}\}$, where c is defined to be a collection of SKUs that do not collectively exceed the capacity of warehouse $\overline{w}$. Thus, $x_{\overline{w}c}$ takes value one if warehouse $\overline{w}$ stocks exactly those SKUs in collection c and zero otherwise.

If we denote by $C(w)$ the set of all feasible SKU collections for warehouse w, then we can formulate the warehouse-based capacitated model ($CHCLD$-W) as:

$$\min \sum_{w \in W} \sum_{c \in C(w)} c_{wc} x_{wc}$$

$$st$$

$$\sum_{w \in r(i)} \sum_{\substack{c \in C(w): \\ s \in c}} x_{wc} \geq 1 \qquad \forall s \in S,\ i \in i(s) \tag{12.1}$$

$$\sum_{c \in C(w)} x_{wc} = 1 \qquad \forall w \in W \tag{12.2}$$

$$x_{wc} \in \{0,1\} \qquad \forall w \in W, c \in C(w),$$

where

$$c_{wc} \equiv \sum_{s \in c} c_{sw.}$$

Constraints (12.1) ensure that every installation of every SKU is in range of at least one warehouse that is assigned a collection containing that SKU. Constraints (12.2) ensure that every warehouse is assigned exactly one collection.

Although this model improves upon the basic model from the perspective of having a provably tighter LP relaxation, it does not provide any computational benefits in our experience. In fact, not only did it fail to outperform the basic model in a single instance, but it was actually *intractable* for many problem instances that could be solved using either the basic or the SKU-based formulation. There are three key reasons that we have identified for this failure.

First, this formulation does not address the key cause of intractability in the basic model – the very large number of cover constraints. These constraints still remain in the warehouse-based formulation, leaving us with very large LPs. Furthermore, because we are using column generation, we are required to solve these large LPs many times. Thus, the LPs themselves prove to be a bottleneck operation in many instances.

A second difficulty appears to stem from the lack of dominance. In the SKU-based formulation, we can disregard all feasible composite variables except those associated with warehouse groups that are minimal. In contrast, for any collection of SKUs that satisfies the capacity

constraint for a particular warehouse, there exists an instance in which this collection is part of the optimal solution. Thus, we must include all feasible composite variables in the warehouse-based formulation. As a result, we have a significantly larger pool of variables from which to draw.

Finally, we believe that the warehouse-based formulation suffered from a lack of good initial columns. In the SKU-based formulation, one obvious choice for initial columns is to solve the uncapacitated stocking problem for each part. In other words, we find minimum cost columns that satisfy the cover constraints without considering the capacity constraints. Such an approach has the benefit of providing an optimal solution with just these initial columns for any instance in which capacity is not binding. Furthermore, the objective of this unconstrained problem encourages columns that will satisfy the disregarded capacity constraints – in order to keep the cost of a composite variable low, we want to put an SKU in as few warehouses as possible, thereby also trying to minimize the overall use of warehouse capacity. For those instances where capacity *is* binding at some subset of the warehouses, the dual information then encourages the pricing problem to identify additional minimal warehouse sets that re-distribute inventory to other warehouses with available capacity. We have found this set of initial columns, in conjunction with a collection of artificial variables to ensure feasibility, to lead to satisfactory convergence of the dual variables, resulting in a fairly small number of non-optimal columns being generated.

In contrast, we have no such "natural" set of initial columns in the warehouse-based formulation. If we take an analogous approach to generating columns by seeking minimum cost columns that satisfy the capacity constraints while disregarding the cover constraints, we are left with individual knapsack problems. Given that we are minimizing, the initial solution for each warehouse will be to stock the empty set. Clearly, this works "against" the cover constraints, which are sure to be violated by this solution. Alternatively, if we generate initial columns by focusing on the cover constraints, then our objective will be to stock large numbers of SKUs in each warehouse in order to cover as many demands as possible – this directly conflicts with our objective of minimizing the amount of inventory kept in the system. In practice, experimenting with a number of different approaches to generating initial columns, we consistently found slow convergence and substantial degeneracy, resulting in a very large number of non-optimal

columns being generated.

12.4 Trading Off Size And Complexity In Defining Composites

The most important decision in formulating a composite-variable model is which variables to group together to form the composites, as is highlighted in the preceding examples. In most problems, we have a wide range of choices of how we do this, trading off smaller numbers of variables and greater complexity in the master problem against larger numbers of variables and less complexity in the master problem but greater complexity in the sub-problems.

At one end of the spectrum, we can keep each basic decision variable distinct. Such a formulation prevents exponential growth in the number of variables but possibly includes non-linearities, weak LP relaxations, and other complicating factors. At the other end of the spectrum, we can group *all* variables together to form one enormous composite. Each composite variable represents a feasible solution to the original problem. The master problem becomes trivial. It is simply a single convexity constraint, which requires us to choose one solution from the feasible set. The difficulty here is of course the fact that the pricing problem to generate these composite variables is itself the original problem.

In between these two extremes, we have an exponential number of possible alternatives. How do we go about choosing from amongst these alternatives, and what should our objective be in making this decision? One clear issue is that we want to choose a variable definition that addresses the causes of intractability in the basic formulation. For example, in the SPL problem, one of the reasons for the success of the SKU-based composite model is the fact that it eliminates the need for the cover constraints, which cause the LP relaxations to be a bottleneck in the basic formulation. In other problems, we might want to focus on a composite definition that removes non-linearities, or that tightens a weak LP relaxation.

Another issue is that we want to ensure that we can solve the composite model, most likely through branch-and-price. Therefore we need to define composite variables with corresponding tractable LP relaxations, easy-to-solve pricing problems, an effective branching strategy, and a branch-and-bound tree of manageable size.

All of these concerns make determining the composite variable definition a challenging

task. In the chapters that follow, we discuss three properties that have been beneficial in our experience, and discuss broader classes of composite variable definitions for which these properties apply.

Chapter 13

Strengthening The LP Relaxation

As we discussed in Part I, a key cause of intractability in many transportation and logistics models is the inherent weakness of the linear programming relaxation. Composite-variable models provide an important benefit by producing tighter linear programming relaxations, in many cases greatly improving computational performance.

Recall, for example, the computational results in Part III, demonstrating that the composite formulation had significantly smaller branch-and-bound trees than did the basic formulation. In general, if we define a composite model as described in Part I, we will always have an LP relaxation that is at least as tight as that for the basic model. We present a proof of this fact in the following theorem; Vanderbeck ([57]) provides an extensive discussion of this topic.

Theorem 9 *Given a basic formulation*

B*:*

$$\min c_x x + c_y y$$

$$st$$

$$A_x x = b_x$$

$$A_y y = b_y$$

$$D_x x + D_y y = g$$

$$x_i \in \{0, 1\} \quad \forall i = 1, 2, ..., n_x$$

$$y_i \in Y_i \quad \forall i = 1, 2, ..., n_y,$$

and the composite-variable model corresponding to vector x

C*:*

$$\min c_{\widehat{z}}\widehat{z} + c_y y$$

$$st$$

$$A_y y = b_y$$

$$D_{\widehat{z}}\widehat{z} + D_y y = g$$

$$e'\widehat{z} = 1$$

$$\widehat{z}_i \in \{0, 1\} \quad \forall \widehat{z}_i \in \widehat{z}$$

$$y_i \in Y_i \quad \forall i = 1, 2, ..., n_y,$$

let Z^B_{LP} *denote the optimal solution to the LP relaxation of* B *and let* Z^C_{LP} *denote the optimal solution to the LP relaxation of* C*. Then* $Z^B_{LP} \leq Z^C_{LP}$*.*

Proof. Consider any vector $\widetilde{y}$. Recall that each composite variable $\widehat{z}_i$ corresponds to an assignment of binary values to the variables in x. Thus, we can re-write any solution $(\widetilde{\widehat{z}}, \widetilde{y})$ to the LP relaxation of C in terms of the original variables x. By definition of $\widehat{z}$, this vector x must lie within the convex hull of

$$A_x x = b_x$$

$$D_x x = g - D_y \widetilde{y}$$

$$x_i \in \{0, 1\} \quad \forall i = 1, 2, ..., n_x.$$

This convex hull is contained within the polyhedron

$$A_x x = b_x$$

$$D_x x = g - D_y \widetilde{y}$$

$$x_i \in [0,1] \quad \forall i = 1, 2, ..., n_x,$$

which corresponds to the LP relaxation of B. ■

This tells us that any composite formulation will have an LP relaxation that is as least as tight as the basic formulation. Furthermore, the composite formulation will be *strictly* tighter unless the polyhedron

$$A_x x = b_x$$

$$0 \leq x_i \leq 1 \quad \forall i$$

has integer extreme points. Thus, using a composite-variable formulation can help us to address intractabilities caused by weak LP relaxations.

This also suggests that when choosing between two different composite-variable definitions, we should pay careful attention to the relative quality of their LP relaxations. In some cases, one formulation will be provably tighter than another. For example, if we considered an alternative SPL formulation where we formed a composite for SKU $\widetilde{s}$ but used the basic variables x_{sw} for all other SKUs, this would have a provably weaker LP relaxation than the formulation in which composites were formed for *all* SKUs.

On the other hand, there will not always be a clear relationship between the LP relaxations of two different composite-variable formulations. In Figures 13-1 and 13-2, for example, we present two instances of the SPL problem. In the first instance, the SKU-based composite formulation has a tighter LP relaxation than the warehouse-based composite formulation. In the second instance, the converse is true. Thus, we cannot always choose between two composite formulations solely as a function of the relative strengths of their LP relaxations. Furthermore, as we have seen in comparing the basic formulation for the SPL problem against the warehouse-based composite formulation, having a strictly tighter LP relaxation does not guarantee improved tractability. In the next two chapters, we next present two additional characteristics that can be useful in evaluating the quality of a composite-variable definition.

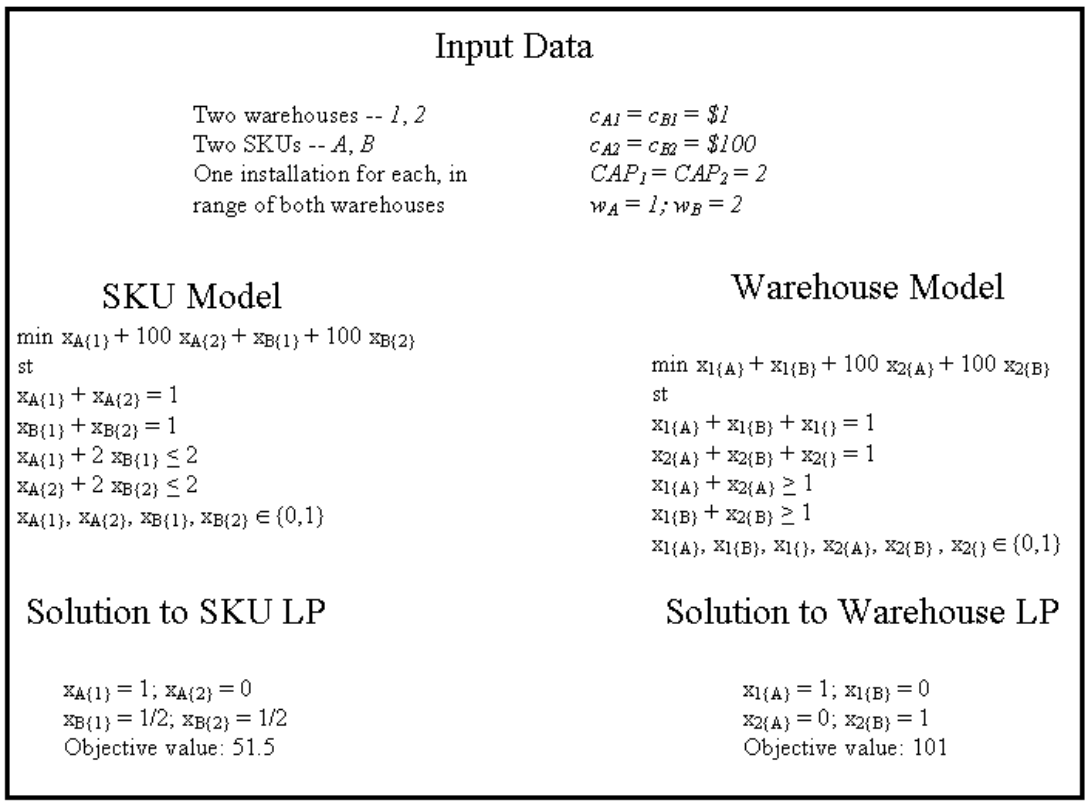


Figure 13-1: Instance in which warehouse formulation has tighter LP relaxation

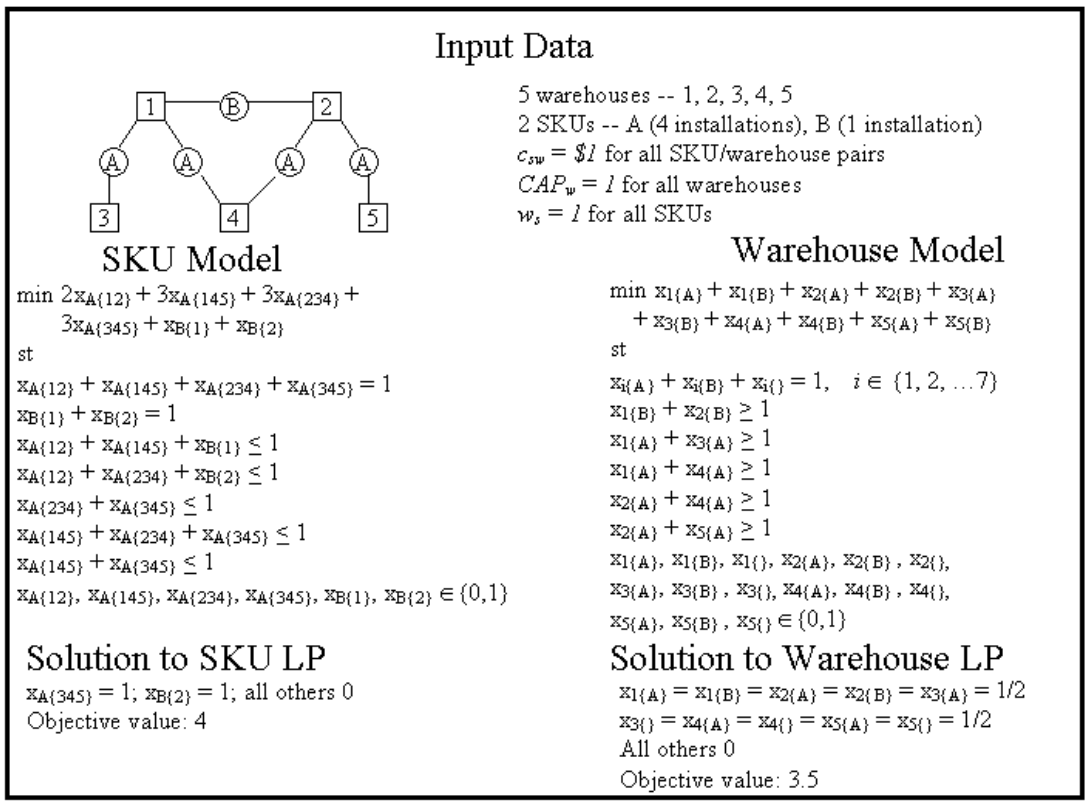


Figure 13-2: Instance in which SKU formulation has tighter LP relaxation

Chapter 14

Dominance

Although improving the strength of the linear programming relaxation can be a significant benefit of composite modeling, it is not always sufficient to establish tractability. In the warehouse-based model for the SPL problem, for example, we saw a composite formulation that performed worse than the basic formulation, in spite of a tighter LP relaxation. For the warehouse-based model, the increased difficulty in solving the individual nodes of the branch-and-bound tree outweighed the benefit of having to consider fewer nodes.

When comparing two composite formulations we might want to also consider their relative size, not only with respect to the number of constraints, but also with respect to the number of composite variables that the formulations contain. In general, defining composites based on larger sets of variables x will result in larger pools of composite variables from which to draw, which may adversely impact tractability. It is not sufficient to solely compare the size of the variable sets x, however. We must also consider whether the resulting set of composite variables can be reduced through *dominance*. We define dominance as follows.

Definition 10 *Composite variable $\widehat{z}_i$ is* dominated *if, for any feasible solution in which $\widehat{z}_i = 1$, there exists another feasible solution with equal or better value in which $\widehat{z}_i = 0$. A composite variable that is not dominated is said to be* dominant.

Clearly, we can exclude all dominated composite variables from our model without sacrificing optimality. If we can identify a pricing problem that only generates dominant variables, then a composite model with dominance may be computationally preferable to one without because

of the reduction in number of variables that must be considered.

Consider as an example the composite for a particular SKU in the SKU-based formulation for the SPL problem. If a warehouse group does not satisfy the cover constraints for that SKU, then the group is infeasible and we do not include the corresponding composite variable in our model. Conversely, if a warehouse group is feasible with respect to the cover constraints but contains a proper subset that is also feasible with respect to the cover constraints, then we do not need to include in our model the composite variable corresponding to the larger warehouse set. This set has higher cost and uses more of a constrained resource (warehouse capacity) without providing any additional benefits. Therefore, it will never be part of an optimal solution. Thus, it is sufficient to include in our model only the dominant warehouse groups – those groups that are minimal with respect to satisfying the cover constraints.

In general, as we try to determine the best collection of variables x over which to define our composites, it may be beneficial to seek out dominance. As is seen both in our experience and in the literature of others using composite-variable modeling, there are may different forms of dominance that have been observed. We present here one form, that seen in the SKU-based SPL model, and describe a broader class of problems for which this form of dominance applies.

Consider a vector x of binary variables in a mixed-integer program of the form

$$\min \quad c_x x + c_y y$$

$$st$$

$$A_x x = b_x$$

$$A_y y = b_y$$

$$G_x x + G_y y \leq h$$

$$x_i \in \{0, 1\} \quad \forall i,$$

and recall that s_i is the set of variables in x that have value one in composite variable $\widehat{z}_i$.

Theorem 11 *If all elements of the vector c_x and the matrix G_x are non-negative, then all composite variables formed from the set x are either infeasible or dominated except those which*

are minimal. *Composite variable j is minimal if it is feasible and any other composite variable k, for which $s_k \subset s_j$, is infeasible.*

Proof. Consider a composite variable $\widehat{z}_i$ which is feasible but not minimal. Then there must exist some $\widehat{z}_j$, another feasible composite variable, for which $s_j \subset s_i$. Any solution $\overline{y}$ that is feasible in conjunction with $\widehat{z}_i$ must also be feasible in conjunction with $\widehat{z}_j$. This is because all of the coefficients of G_x are non-negative and $s_j \subset s_i$. Furthermore, the solution $\{\widehat{z}_j, \overline{y}\}$ has lower objective value than the solution $\{\widehat{z}_i, \overline{y}\}$ because

$$\sum_{k \in S_j} c_k^X \leq \sum_{k \in Si} c_k^X .$$

Again, this is due to the fact that $s_j \subset s_i$ and that $(c_x)_k \geq 0$ for all k. Thus, $\widehat{z}_i$ is dominated and can be excluded from the formulation without loss of optimality. ■

Thus, for any variable set x satisfying these conditions, it is sufficient to only include minimal composite variables. In the following theorem, we prove that the pricing problem for such a model will always generate minimal composite variables.

Theorem 12 *Given a composite re-formulation of a model with the above structure, the pricing problem will always identify minimal (i.e., dominant) composite variables.*

Proof. By contradiction. Let $\widehat{z}_i$ be the optimal solution to an iteration of the pricing problem with dual values α_k associated with the k^{th} linking constraint and σ associated with the convexity constraint. Suppose that $\widehat{z}_i$ is *not* dominant. Then there exists some feasible composite variable $\widehat{z}_d$ such that $s_d \subset s_i$. The reduced cost of $\widehat{z}_i$ is

$$\sum_{j \in s_i} (c_x)_j - \sigma - \sum_{k=1}^{n} \left(\sum_{m \in s_i} (G_x)_{km} \right) \alpha_k$$

which is greater than

$$\sum_{j \in s_d} (c_x)_j - \sigma - \sum_{k=1}^{n} \left(\sum_{m \in s_d} (G_x)_{km} \right) \alpha_k,$$

given that $s_d \subset s_i$, $(c_x)_j \geq 0$ for all j, and $\alpha_k \leq 0$ for all k. This contradicts the supposition that $\widehat{z}_i$ is an optimal solution to the pricing problem. ■

We also observe that any composite model based on a variable set x that satisfies these conditions has an obvious candidate set of initial columns. Specifically, we can start with one or more composite variables corresponding to optimal solutions to the problem

$$Min \quad c_x x$$

$$st$$

$$A_x x = b_x$$

$$x_i \in \{0, 1\} \quad \forall i.$$

As with the example described earlier for the SKU-based model of the SPL problem, such initial columns have the benefit of being optimal whenever the linking constraints are not binding. Furthermore, this approach "works with" the linking constraints in the sense that the smaller the set (which minimizes the objective function) the less of the linking resource is used.

To summarize, when determining how to group variables so as to form composites, it can be useful to look for dominance. We have presented here conditions for a broad class in which one form of dominance exists. These conditions occur whenever the decisions in set x have non-negative cost and decisions in x and y are linked by the common use of limited resources. Other problem structures may result in alternative forms of dominance, and these conditions should be sought out as well when determining variable definition.

Chapter 15

Relaxing The Integrality Of The Composite Variables

We next consider the property found in the Extended Crew Pairing Model that enabled us to relax the integrality of the composite variables, and considering other classes of problems that might also possess this property.

In constructing the *ECP* model, we begin with a model that contains two sets of binary variables. Variables in $\{x_r\}$ are associated with the decisions of whether to assign aircraft to strings of flights known as maintenance route strings. Variables in $\{y_p\}$ are associated with the decisions of whether to assign crews to strings of flights known as crew pairings. In the composite model *ECP*, we replace the set $\{x_r\}$ with a collection of composite variables $\{z_s\}$, where z_s takes value one if maintenance solution s is chosen and zero otherwise. We showed that the integrality of these composite variables could be relaxed. That is, given integer values for the variables in $\{y_p\}$, the remaining polyhedron has integer extreme points. Thus, we will never need to branch on the composite variables.

In the following theorem, we present a broader class of problems for which this property applies.

Theorem 13 *Given a mixed integer program, if there exists a subset of binary variables x such*

that the rows and columns can be re-arranged in the form

$$\min c_x x + c_y y$$

$$st$$

$$A_x x = b_x$$

$$Cx \leq e$$

$$A_y y = b_y$$

$$Cx - Dy \geq 0$$

$$x_i \in \{0,1\} \quad \forall i$$

$$y_j \text{ integer } \forall j$$

where e is a vector of appropriate dimension containing all ones, $C_{ij} \in \{0,1\}$ for any element of C, and any element of D is integer, then we can relax the integrality of composites formed from the set X.

Proof. Consider the re-formulation

$$\min c_z z + c_y y$$

$$st$$

$$e' z = 1$$

$$A_y y = b_y$$

$$\widetilde{C} z - Dy \geq 0$$

$$z_i \in [0,1] \quad \forall i$$

$$y_j \text{ integer } \forall j$$

where

$$c_i^Z = \sum_{j \in S_i} c_j^X,$$

$$\widetilde{C}_{ij} = \sum_{k \in S_j} C_{ik},$$

and we have relaxed the integrality of the composite variables.

Consider any assignment of integer values $\widehat{y}$. We can re-write the remaining problem with respect to z as

$$\min c_z z$$

$$st$$

$$e' z = 1$$

$$\widetilde{C} z \geq D\widehat{y}$$

$$z_i \in [0, 1] \quad \forall i.$$

We prove by contradiction that this polyhedron is either infeasible or has integer extreme points. First, observe that any element of the vector $\widetilde{C}z$ is at most one due to the fact that all elements of $\widetilde{C}$ are either zero or one that

$$e' z = 1.$$

Thus whenever any element of $D\widehat{y}$ is strictly greater than one, the problem will be infeasible.

What about when all elements of $D\widehat{y}$ have value of one or less? Consider a fractional extreme point. Let z_p and z_q be two basic variables in this solution, and examine their corresponding columns. The first element of both columns is one (this is the convexity constraint).

Next consider any constraint i from the set

$$\widetilde{C} z \geq D\widehat{y}$$

for which $[D\widehat{y}]_i = 1$. In order for $\left[\widetilde{C}z\right]_i = 1$, both $\widetilde{C}_{ip}$ and $\widetilde{C}_{iq}$ (that is, the i^{th} elements of both columns) must have value one. This is due to the fact that $\left[\widetilde{C}z\right]_i \leq e' z = 1$.

Now consider any constraint j from the set

$$\widetilde{C}z \geq D\widehat{y}$$

for which $[D\widehat{y}]_j < 1$. In this case $\widetilde{C}_{jp}$ and $\widetilde{C}_{jq}$ can differ (that is, it is possible for the j^{th} element of one column to be one and the other to be zero). However, in this case the slack variable corresponding to the j^{th} constraint must be basic.

Thus, the columns associated with z_p and z_q are identical except in possibly some elements k, in which case the k^{th} slack variable is basic. This contradicts the linear independence of a basic solution. ■

This suggests that in determining which variables to group together to form a composite, it may be useful to determine whether there exists a set x that satisfies the above criteria. Such a condition occurs whenever we can divide the variables of a problem into two sets, x and y, where the relationship between them is such that some decisions in y rely on choosing one decision from a subset within x. For example, decisions in y might relate to constructing passenger itineraries, and decisions in x might relate to assigning flights to time slots. Alternatively, decisions in y might relate to assigning customers to warehouses and decisions in x might relate to selecting building sites. Many other instances of such relationships exist throughout transportation and logistics.

Chapter 16

Conclusions For Part IV

We conclude this part of the thesis by recognizing that one of the greatest challenge in formulating a tractable composite-variable model is to group the decision variables appropriately. We have described a number of different characteristics that are of use in some cases – stronger LP relaxations, the presence of dominance, the opportunity to relax the integrality of the composite variables – and presented broad problem classes in which these properties can be found. We note that many other properties exist as well, and that in developing future composite-variable models, attention should be paid not only to establishing tractable models, but also to developing further insights that can be applied in other applications.

Part V

Conclusions And Suggestions For Further Research

In this thesis, we have considered the role composite-variable models can play in improving tractability for challenging problems in transportation and logistics. The contributions of the thesis are three-fold. First, we provide a new model and solution approach to solve an important real-world problem from the airline industry. Second, we provide a framework for addressing challenging problems in service parts logistics. Third, we provide insights into how to define composite variables for greater tractability. These insights can be useful not only in solving large-scale problems, but also in integrating multiple stages within a planning environment and in developing better heuristics for solving large problems in real time to allow users greater control in trading off solution time and quality.

We believe there to be many interesting avenues yet to be pursued in this research area. These include topics related to specific applications, as well as more general topics concerning composite modeling techniques. We suggest just a few of these topics.

- **Integrated airline planning:** Significant cost savings may be achieved at most major airlines by further integrating the planning process to include not only crew scheduling and aircraft routing, but the fleet assignment problem as well. To date, this combined problem has been too large to be solved in a reasonable amount of time. Composite-variable modeling might allow greater tractability.

- **Service parts logistics:** We are greatly interested in expanding the scope of the problem considered in Part III. It is important to incorporate policies for repair parts which are high demand, to consider additional problem constraints beyond warehouse capacity, and to incorporate transportation costs as well as inventory costs.

- **Dominance:** We have observed in our study of different composite formulations for the SPL problem that dominance seems to play a role in convergence. Is their a link between dominance and degeneracy? Does the presence of dominance play a role beyond simply reducing the number of composite variables to be considered? Why is such a reduction useful in a branch-and-price context, in which we do not fully enumerate the variables?

- **Simulation:** Another factor that significantly complicates many transportation and lo-

gistics problems is the presence of stochasticity. Finding closed-form expressions can often be difficult in such environments. We are interested in considering how simulation might be incorporated into composite-variable modeling to help address this difficulty. For example, it is not always possible to find a closed form expression for the cost of a SPL stocking policy for a high-demand part. Could we instead determine this cost through simulation, if our composite variable represents the complete stocking policy? What does this mean for the pricing problem? Can we establish any optimality conditions or bounds on solution quality?

- **Other application areas:** We believe that there are many other application areas to which composite-variable modeling can contribute. Some promising areas are: problems that integrate multiple stages of a planning process; real-time environments in which solution quality must be traded off against solution time; complex environments where a tractable basic model captures insufficient detail; and stochastic environments in which simulation can be imbedded in the variable definition.

Bibliography

[1] P. Alfredsson and J. Verrijdt (1999). "Modeling Emergency Supply Flexibility in a Two-Echelon Inventory System." *Management Science* 45: 1416 - 1431.

[2] R. Anbil, C. Barnhart, L. Hatay, E. Johnson, and V. Ramakrishnan (1993). "Crew-Pairing Optimization at American Airlines Decision Technologies." T. Ciriano and R. Leachman (eds.). *Optimization in Industry: Mathematical Programming and Modeling Techniques in Practice*, John Wiley and Son, England, 31 - 36.

[3] R. Anbil, E. Gelman, B. Patty, and R. Tanga (1991). "Recent Advances in Crew-Pairing Optimization at American Airlines." *Interfaces* 21: 62 - 74.

[4] R. Anbil, R. Tanga, and E. Johnson (1992). "A Global Approach to Crew-Pairing Optimization." *IBM Systems Journal* 31: 71 - 78.

[5] E. Andersson, E. Housos, N. Kohl, and D. Wedelin (1998). "Crew Pairing Optimization." *Operations Research in the Airline Industry.* G. Yu (editor), International Series in Operations Research and Management Science, 228 - 258.

[6] A. Armacost, C. Barnhart, and K. Ware. "Composite Variable Formulations for Express Shipment Service Network Design." *Transportation Science* 36.

[7] S. Axsäter (1990). "Modelling Emergency Lateral Transshipments in Inventory Systems." *Management Science* 36: 1329 - 1338.

[8] M. Ball and A. Roberts (1985). "A Graph Partitioning Approach to Airline Crew Scheduling." *Transportation Science* 19: 107 - 126.

[9] C. Barnhart, N. Boland, L. Clarke, E. Johnson, G. Nemhauser, R. Shenoi (1998). "Flight String Models for Aircraft Fleeting and Routing." *Transportation Science* 32: 208 - 220.

[10] C. Barnhart, A. Farahat, and M. Lohatepanont. "Airline Fleet Assignment: An Alternative Model and Solution Approach Based on Composite Variables." Submitted to *Operations Research*.

[11] C. Barnhart, C. Hane, and P. Vance (2000). "Using Branch-and-Price-and-Cut to Solve Origin-Destination Integer Multicommodity Flow Problems." *Operations Research* 48: 318 - 326.

[12] C. Barnhart, E. Johnson, R. Anbil, and L. Hatay (1994). "A Column Generation Technique, for the Long-Haul Crew Assignment Problem." *Optimization in Industry: Volume II*, T. Ciriano and R. Leachman (eds.). John Wiley and Son, England, 7 - 22.

[13] C. Barnhart, E. Johnson, G. Nemhauser, M. Savelsbergh, and P. Vance. (1998). "Branch-and-Price: Column Generation for Solving Huge Integer Programs." *Operations Research* 46: 316 - 329.

[14] C. Barnhart, E. Johnson, G. Nemhauser, and P. Vance (1999). "Crew Scheduling." *Handbook of Transportation Science*, Randolph W. Hall (editor), Kluwer Academic Publisher, 493 - 521.

[15] C. Barnhart, M. Lohatepanont, and N. Krishnamurthy (1998). "Airline Schedule Evaluation and Optimization." INFORMS Seattle 1998.

[16] C. Barnhart, F. Lu, and R. Shenoi (1998). "Integrated Airline Scheduling." *Operations Research in the Air Industry*, G. Yu (editor), International Series in Operations Research and Management Science, 9: 384 - 403.

[17] C. Barnhart and R. Shenoi (1998). "An Approximate Model and Solution Approach for the Long-Haul Crew Pairing Problem." *Transportation Science* 32: 221 - 231.

[18] C. Barnhart and K. Talluri (1997). "Airline Operations Research." *Design and Operation of Civil and Environmental Engineering Systems*, pp. 435 - 469, edited by Profs. Arthur McGarity and Charles ReVelle, 1997.

[19] J. Beasley and B. Cao (1996). "A Tree Search Algorithm for the Crew Scheduling Problem." *European Journal Of Operational Research* 94: 517 - 526.

[20] E. Butchers, P. Day, A. Goldie, S. Miller, J. Meyer, D. Ryan, A. Scott, and C. Wallece (2000). "Optimised Crew Scheduling at Air New Zealand." Working paper.

[21] S. Ceria, P. Nobili, and A. Sassano (1997) "Set Covering Problem." M. Dell'Amico, F. Maffioli, and S. Martello (eds.). *Annotated Bibliographies in Combinatorial Optimization*, Wiley-Interscience Series in Discrete Mathematics and Optimization, 415 - 428.

[22] K. Cheung and W. Hausman (1995). "Multiple Failures in a Multi-Item Spares Inventory Model." *IIE Transactions* 27: 171 - 180.

[23] H. Chu, E. Gelman, and E. Johnson (1997). "Solving Large Scale Crew Scheduling Problems." *European Journal of Operational Research* 97: 260 - 268.

[24] L. Clarke, C. Hane, E. Johnson, and G. Nemhauser (1996). "Maintenance and Crew Considerations in Fleet Assignment." *Transportation Science* 30: 249 - 260.

[25] L. Clarke, E. Johnson, G. Nemhauser, and Z. Zhu (1996). "The Aircraft Rotation Problem." *Annals of Operations Research* 69: 33 - 46.

[26] M. Clarke and B. Smith (2000). "The Impact of Operations Research on the Evolution of the Airline Industry: A Review of the Airline Planning Process." Research paper, Sabre Inc.

[27] M. Cohen, C. Cull, H. Lee, and D. Willen (2000). "Saturn's Supply-Chain Innovation: High Value in After-Sales Service." *Sloan Management Review* Summer: 93 - 101.

[28] M. Cohen, P. Kamesam, P. Kleindorfer, H. Lee and A. Tekerian (1990). "Optimizer: IBM's Multi-Echelon Inventory System for Managing Service Logistics." *INTERFACES* 20: 65 - 82.

[29] M. Cohen, P. Kleindorfer, and Lee (1986). "Optimal Stocking Policies for Low Usage Items in Multi-Echelon Inventory Systems." *Naval Research Logistics Quarterly* 33: 17 - 38.

[30] M. Cohen, Y. Zheng, and V. Agrawal (1997). "Service Parts Logistics: A Benchmark Analysis." *IIE Transactions* 29: 627 - 639.

[31] J. Cordeau, G. Stojković, F. Soumis, and J, Desrosiers (2000). "Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling." *Technical Report G-2000-37,* GERAD, École Polytechnique de Montréal.

[32] T. Crainic and K. Rousseau (1987). "The Column Generation Principle and the Airline Crew Scheduling Problem." *INFOR* 25: 136 - 151.

[33] G. Dantzig and P. Wolfe (1960). "The Decomposition Principle for Linear Programs." *Operations Research* 8: 101 - 111.

[34] G. Desaulniers, J. Desrosiers, Y. Dumas, S. Marc, B. Rioux, M. Solomon, and F. Soumis (1997). "Crew Pairing at Air France." *European Journal of Operational Research* 97: 245 - 259.

[35] G. Desaulniers, J. Desrosiers, M. Gamache, and F. Soumis (1998). "Crew Scheduling in Air Transportation." T. Crainic and G. Laporte (eds.). *Fleet Management and Logistics*, Kluwer Academic Publishers, 169 - 185.

[36] M. Etschmaier and D. Mathaisel (1985). "Airline Scheduling: An Overview." *Transportation Science* 19: 127 - 138.

[37] I. Gershkoff (1989). "Optimizing Flight Crew Schedules." *INTERFACES* 19: 29 - 43.

[38] R. Gopalan and K. Talluri (1998). "The Aircraft Maintenance Routing Problem." *Operations Research* 46: 260 - 271.

[39] G. Graves, R. McBride, I. Gershkoff, D. Anderson, and D. Mahidhara (1993). "Flight Crew Scheduling." *Management Science* 39: 736 - 745.

[40] S. Graves (1985). "A Multi-Echelon Inventory Model for a Repairable Item with One-for-One Replenishment." *Management Science* 31: 1247 - 1256.

[41] C. Hane, C. Barnhart, E. Johnson, R. Marsten, G. Nemhauser, and G. Sigismondi (1995). "The Fleet Assignment Problem: Solving a Large-Scale Integer Program." *Mathematical Programming* 70: 211 - 232.

[42] K. Hoffman and M. Padberg (1993). "Solving Airline Crew Scheduling Problems by Branch-and-Cut." *Management Science* 39: 657 - 682.

[43] D. Klabjan, E. Johnson, and G. Nemhauser (1999). "Solving Large Airline Crew Scheduling Problems: Random Pairing Generation and Strong Branching." *Technical Report TLI/LEC-99-11*, Georgia Institute of Technology.

[44] D. Klabjan, E. Johnson, G. Nemhauser, E. Gelman, and S. Ramaswamy (1999). "Airline Crew Scheduling with Time Windows and Plane Count Constraints." *Technical Report TLI/LEC-99-12*, Georgia Institute of Technology.

[45] D. Klabjan and K. Schwan (1999). "Airline Crew Pairing Generation in Parallel." *Technical Report TLI/LEC-99*, Georgia Institute of Technology.

[46] S. Lavoie, M. Minoux, and E. Odier (1988). "A New Approach for Crew Pairing Problems by Column Generation with an Application to Air Transportation." *European Journal of Operational Research* 35: 45 - 58.

[47] D. Levine (1996). "Application of a Hybrid Genetic Algorithm to Airline Crew Scheduling." *Computers and Operations Research* 23: 547 - 558.

[48] M. Lohatepanont (2000). Personal correspondence.

[49] M. Lohatepanont and C. Barnhart. "Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment." Accepted with minor changes, *Transportation Science*, June 2001 (revised submission November 2001).

[50] R. Martin (1987). "Generating Alternative Mixed-Integer Programming Models Using Variable Redefinition." *Operations Research* 35: 820 - 831.

[51] www.mdot.state.md.us/mmcp/IndustryForum/Transportation_and_logistics.ppt.

[52] J. Muckstadt and M. Isaac (1981). "An Analysis of Single Item Inventory Systems With Returns." *Naval Research Logistics Quarterly* 28: 237 - 254.

[53] B. Rexing, C. Barnhart, T. Kniker, A. Jarrah, and N. Krishnamurthy (2000). "Airline Fleet Assignment with Time Windows." *Transportation Science* 34: 1 - 20.

[54] C. Sherbrooke (1968). "METRIC: A Multi-Echelon Technique for Recoverable Item Control." *Operations Research* 16: 122 -141.

[55] K. Talluri (1998). "The Four-Day Aircraft Maintenance Routing Problem." *Transportation Science* 32: 43 - 53.

[56] P. Vance, C. Barnhart, E. Johnson, and G. Nemhauser (1997). "Airline Crew Scheduling: A New Formulation and Decomposition Algorithm." *Operations Research* 45: 188 - 200.

[57] F. Vanderbeck (2000). "On Dantzig-Wolfe Decomposition in Integer Programming and Ways to Perform Branching in a Branch-and-Price Algorithm." *Operations Research* 48: 111 - 128.

[58] F. Vanderbeck and L. Wolsey (1996). "An Exact Algorithm for IP Column Generation." *Operations Research Letters* 19: 151 - 159.

[59] Y. Wang (2001). "Service Parts Logistics: Modeling, Analysis, and Application." Disseration, The Wharton School, University of Pennsylvania.