A RESERVATIONS-BASED RAILWAY NETWORK OPERATIONS MANAGEMENT SYSTEM

Edwin Reese Kraft

A DISSERTATION
in
Systems Engineering

Presented to the Faculties of the University of Pennsylvania in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

1998

_____
Supervisor of Dissertation

_____
Graduate Group Chairperson

DEDICATION

To my wife Rose, who gave up a comfortable life in Jacksonville, Fl orida, sold half of our belongings and moved with our two children, Melissa and Paul, into a one-bedroom apartment in downtown Philadelphia for two years. Rose continued to support me in this endeavor as a two-year strategy turned into a five-and-a-half year odyssey. She orchestrated seven relocations within four years, and endured a financial crisis every six to twelve months when my research funding or consulting contracts expired. Rose continued to support my efforts to complete this research even after our son Paul was diagnosed with autism three years ago.

ACKNOWLEDGEMENTS

When I was wrapping up my Master's degree back in 1983, Professor Vukan R. Vuchic was the first to suggest I consider pursuing a doctorate. By that time, I'd already been offered a position in the Chessie System Railroads' management training program, so I decided to start my railroading career instead. However, over the years, I never forgot that conversation, so Dr. Vuchic is really the one who started it all.

My wife Rose suggested in December 1991 that I might consider going back to school, so I took her up on her most generous offer.

My parents Ed and Gretchen Kraft convinced me that a doctorate wouldn't cost any more than a new car, and would be much more valuable. Of course this turned out to be only a fraction of the true financial cost, but this coaching at a critical moment still helped us make the decision to go ahead and do it.

My grandmother, Mrs. Margaret Reese and father-in-law Dr. Brendan Liddell provided ongoing financial support, without which we couldn't have made it. Their support has also been invaluable in meeting the extra costs of our son Paul's intensive autism therapy program over the past three years. My mother-in-law, Dr. Elaine Liddell moved in with us to share some of our living expenses, which allowed us to purchase a decent home.

Professor Edward K. Morlok played a critical role in gaining my readmission to Penn in 1992, and in lining up financial support for my first two years. Serving as my academic advisor and as a member of my dissertation committee, he directed my course of studies to help me pass the Qualifying exams, and to provide raw material for my dissertation research.

Professor Marshall Fisher's dual ascent procedure for solving the Generalized Assignment Problem, first introduced to me in his logistics course, inspired the Dynamic Car Scheduling solution approach taken here.

Professor Monique Guignard-Spielberg first introduced me to the mathematics of subgradient optimization as well as to other useful integer programming techniques, and served on my dissertation committee. In her final exam, I really learned the importance of solving Lagrangian Relaxation sub-problems to optimality, when a GAMS run-time parameter wasn't set properly.

Professor Patrick T. Harker facilitated my pursuit of a research topic of my own choosing, in spite of the absence of any direct railroad industry funding support. As my dissertation advisor, he strongly influenced the direction of this research, particularly the decision to implement a rolling horizon simulation model to evaluate the results. His tireless review of, and prompt feedback provided on, numerous drafts and redrafts of earlier versions of the dissertation are very much appreciated.

Dr. Marc Meketon's service on my committee provided invaluable assistance, particularly by strengthening the literature review in the revenue management area, and through his attention to detail in numerous mathematical aspects of this research.

The responsibility for any errors, of course, continues to lie solely with the author.

University financial support is gratefully acknowledged. My first year expenses were covered by a University Fellowship; and for the last three years, part-time tuition support was provided from United Parcel Service funds.

ABSTRACT

A RESERVATIONS-BASED RAILWAY NETWORK OPERATIONS MANAGEMENT SYSTEM

Edwin R. Kraft (Author)
Patrick T. Harker (Supervisor)

A shipment scheduling and operational control method is developed and tested, to help railroads become more competitive for high revenue, service sensitive freight. A bid-price based, profit maximizing revenue management approach is proposed to allow a rail carrier to develop achievable and market sensitive quotations of delivery time for new shipments calling in. Bid prices are derived using a subgradient step size algorithm; a modified shortest path procedure is used to solve the decomposed subproblems.

Once the service quotation has been developed, a deterministic, cost minimizing, multicommodity network flow model, including train capacity and integral flow constraints is solved to manage shipments moving on the railroad in real time. This model dynamically reroutes shipments to take advantage of all available train capacity in the network, while still meeting the committed delivery times on priority shipments. A customized dual ascent procedure, using a tabu search approach as an anti-cycling mechanism, adapts the previous solution any time new information is received.

Both procedures have been integrated into a rolling horizon simulation model. Simulation results indicate up to a 10 point improvement in railway operating ratio may be achievable through implementation of this shipment management strategy.

Keywords: Railroad, Railway, Revenue Management, Yield Management, Car Scheduling, Shipment Scheduling, Bid Price, Optimization, Simulation, Service Monitoring, Subgradient Algorithm, Tabu Search, Service Management

TABLE OF CONTENTS

1 Introduction

1.1 The Need to Improve Rail Service Reliability 1

1.2 A Service Differentiation Strategy 3

1.3 The Need to Maintain Flexibility. 4

1.4 Car Scheduling as a Service Management Tool 6

1.5 A Comparison with Airline Yield Management 9

1.6 Multi-Pathing as a Means to Improve Reliability 18

1.7 Research Statement 20

1.8 Structure of the Dissertation 20

2 Rail-Related Literature Review

2.1 General 22

2.2 Mechanical Component Reliability 24

2.3 Tactical Operating Plan Development 27

2.4 Empty Equipment Distribution 31

2.5 Train Dispatching and Control Systems 34

2.6 Rail Terminal Research. 39

2.7 A Taxonomy of Approaches 42

2.8 Positioning of this Research in the Taxonomy 45

3 A Model for Dynamic Car Scheduling: Problem Formulation

3.1 Problem Definition 48

3.2 Formulation Issues 52

3.3 Formulation: Dynamic Car Routing Model 57

3.4 Formulation: Train Segment Pricing Model 62

3.5 Dual Formulation 68

3.5.1 General Formulation of the Dual 73

3.6 Representing Forecast Demand Uncertainty 74

3.7 Limitations on the Scope of this Research 82

3.8 Requirements for Solution Algorithms 82

4 Solution Algorithms

4.1 Chapter Outline 85

4.2 The Lagrangian Relaxation 87

4.2.1 Shortest Path Subproblems with Gains 90

4.2.2 Shortest Path Literature Review 92

4.2.3 Determining the Stage or “Depth” of each Node 94

4.2.4 Determination of Shortest Paths 96

4.3 Mathematical Programming Literature Review 97

4.4 A Dual Adjustment Procedure for Dynamic Car Scheduling 105

4.4.1 Mathematical Properties of the Dual Adjustment Procedure 107

4.4.2 Simple Examples of the Dual Adjustment Procedure 114

4.4.3 Avoiding Overcorrection of the Dual Variables 122

4.4.4 Formal Definition of the Dynamic Car Scheduling Algorithm 129

4.4.5 Two Possible Future Efficiency Improvements 137

4.4.6 Convergence of the Dynamic Car Scheduling Algorithm 138
4.4.7 Rolling Horizon Testing of Dynamic Car Scheduling 143
4.5 Train Segment Pricing Model 150
4.5.1 Subgradient Solution Procedure 152
4.5.2 Obtaining a Feasible Solution and Integration with Dynamic Car Scheduling 156
4.5.3 Algorithmic Performance 161
4.6 Regulating the Application of Penalty Costs 166

5 Rolling Horizon Simulation Testing

5.1 Chapter Organization 173
5.2 Definition of Test Scenarios. 175
5.3 Simulation Model Description 181
5.4 Economic Performance Evaluation 188
5.5 Simulation Test Results: Operating Performance 193
5.5.1 Transit Time Comparisons versus “Base ETA” 199
5.5.2 Transit Time Comparisons versus “Commitment ETA” 204
5.6 Simulation Test Results: Economic Performance 205
5.7 Two Special Rolling Horizon Scenarios 213

6 Summary and Conclusions

6.1 Evaluating the Research Contribution 216
6.2 Usability of the Research and its Value to Railroads 217
6.3 Future Research Possibilities 219

Bibliography 222

A Example Problems of Dynamic Car Scheduling and Train Segment Pricing, from Chapter 4 245

B Adapting Kwon's [1994] Test Problem for the Rolling Horizon Simulation 280

C Rolling Horizon Simulation Test Results 287

LIST OF FIGURES

1.1 Current Car Scheduling System Status 8

1.2 Proposed Real Time Service Commitment Process 10

1.3 Three Links in Series 18

1.4 Three Links in Parallel 19

2.1 Hierarchical Decisions in Rail Operations. 23

2.2 Research Taxonomy by Function and Time 43

3.1 Proposed Reservation/Booking and Dynamic Car Scheduling Process 51

3.2 Train Route Segment Definition 59

3.3 The Revenue Value of Capacity 63

3.4 "Acceptance Function" gives the Probability a certain Service Offer will be Accepted 64

3.5 Example Network for Dual Formulation. 70

3.6 Simplex Tableau for Example Network 71

4.1 Modified Shortest Path Subproblem for Commodity "k". 90

4.2 Simple Example of Dual Adjustment Heuristic 116

4.3 More Difficult Example, with "Pooling" 117

4.4 Difficult Example, with “Tabus” 118
4.5 Avoiding an Overcorrection 124
4.6 Diverting with Thru Blocks 126
4.7 Original vs Adjusted Path Costs 127
4.8 Car Scheduling Algorithm: Performance Comparison 128
4.9 Effect of Leg Locking. 145
4.10 DCS Duality Gap by Time 149
4.11 Required Number of Iterations 149
4.12 Required CPU Time 149
4.13 Remaining Infeasibility 161
4.14 Cumulative Correction 162
4.15 Upper Bound and Primal Solution 163
4.16 CPU Time Distribution 164
4.17 TSP Gap by Day 165
4.18 Splitting a Flow with a Penalty Cost in a No-Slack Condition 167
4.19 Splitting a Flow with a Penalty Cost One Days’ Schedule Slack 168
4.20 Split Shipments in Scenario 3 Test Run 169
4.21 Splitting a Flow with a Penalty Cost Extending the
Service Commitment 171

5.1 Levels of Aggregation in Rail Operations 175
5.2 “Flat” Blocking Network 176
5.3 Dynamic Blocking and Train Schedule Network 178
5.4 Key Comparisons in Simulation Analysis 181
5.5. Proposed Real Time Service Commitment Process 183

5.6 Distribution of Penalty Cost 191
5.7 Enroute Inventory by Scenario 194
5.8 DCS Inventory in 15% Reduced Capacity Case 196
5.9 Stranded Shipments 197
5.10 Cars Originated vs Terminated 198
5.11 DCS Call % by Day 199
5.12 Transit Time vs Base Delivery 201
5.13 Days Later than Base vs Average Hourly Cost 202
5.14 Scenario 4 Deliveries 203
5.15 Days Later than Base vs Penalty Cost 203
5.16 Average Daily Contribution 207
5.17 Average Daily Contribution Difference 208
5.18 Contribution vs Capacity 211
5.19 Enroute Inventory in Reduced Capacity Scenarios 212
5.20 Overbooking Coefficient versus Contribution 213

A.1 First Look Ahead Example 247
A.2 Second Look Ahead Example 248
A.3 Third Look Ahead Example 249
A.4 Fourth Look Ahead Example 250
A.5 Train Service Network used in Test Examples 251
A.6 Train Service Blocking Network 251
A.7 Local Trains for Test Example 252
A.8 Yard Cost File 253
A.9 Raw Demand File 253

A.10 Sweep Up with Look Ahead: 10 Cars Capacity 254

A.11 Sweep Down with Look Ahead: 10 Cars Capacity 261

A.12 Sweep Up with Look Ahead: 6 Cars Capacity 269

A.13 Sweep Down with Look Ahead: 6 Cars Capacity 272

A.14 Standard Step Size Procedure Fails to Solve Linear Network Flow Problems But Can Still Produce a Tight Lower Bound 276

B.1 Train Segment Utilization from Kwon [1994] 283

B.2 Train Capacity Utilization from Dynamic Car Scheduling Scenario #3: Base Case 283

B.3 Train Capacity Utilization from Dynamic Car Scheduling Scenario #3: Base Capacity Reduced by 15% 284

LIST OF TABLES

3.1 Example acceptance function for ρ=2 66

3.2 Sample Demand Distribution 77

4.1 Performance of the Dynamic Car Scheduling Algorithm 146

4.2 Comparison of First, Last and Best Iteration 147

4.3 Performance of the Dynamic Car Scheduling Algorithm with
Outside Iterations turned “off” 148

5.1 “C” Values from Student-t Distribution 193

5.2 Performance vs Commitment by Penalty Cost for Scenario 4 205

5.3 Contribution by Scenario 206

5.4 Revenue Ratios by Scenario 209

5.5 Capacity versus CPU Seconds 211

B.1 Dynamic Car Scheduling - Base Case

Individual Train Load Factors 285

B.2 Dynamic Car Scheduling - Reduced Capacity Scenario

Individual Train Load Factors 286

C.1 Transit Time Distribution by Hourly Cost for Scenario 1 288

C.2 Transit Time Distribution by Hourly Cost for Scenario 2 289

C.3 Transit Time Distribution by Hourly Cost for Scenario 3 290

C.4 Transit Time Distribution by Hourly Cost for Scenario 4 290

C.5 Transit Time Distribution by Penalty Cost for Scenario 4 290

C.6 Transit Time Distribution by Hourly Cost for Scenario 3 291

C.7 Slack Time Added by Penalty Cost for Scenario 4 291

C.8 Transit Time Distribution by Hourly Cost with Penalty Costs

on Shipments costing more than $10/hour 292

C.9 Transit Time Distribution by Logit Priority Coefficient 293

C.10 Transit Time Distribution by Logit Priority Coefficient with

Penalty Costs on Shipments having Priority Coefficient < 2 293

CHAPTER 1

Introduction

1.1 The Need to Improve Rail Service Reliability

A major challenge in railroading today is to improve transit time reliability to the point where railcar shipments can compete with trucks on a service basis. While it may never be economically possible for railroads to match trucking speed (Keaton [1991]), it should at least be possible to improve service reliability. This must be done in a manner which does not sacrifice the efficiency and cost advantage of rail movement. With such improvements, railroads will be able to offer a highly reliable service with reasonable transit times, at a much lower cost than truck movement.

Customers moving towards Just-In-Time (JIT) inventory systems require ever increasing quality and reliability in transportation service. Trucks continue to gain market share at the expense of rail (OECD [1992], pg. 55). The question is whether railroads can reverse this trend of market share loss in general manifest freight. To accomplish this

reversal, they will have to do a better job of meeting customer's needs. From OECD [1992] (pg. 100):

> Although it may seem to be otherwise, trends in logistics are not primarily technology- driven. Customers do not ask for high speed in transport, nor for continuous information on the goods, nor for containers and quick load systems. They ask for reliable delivery of goods at the right time and place, without damage and at a fair price. Customers need to have the feeling that their goods transported are in capable hands.

Early work in the logistics field included such developments as the Economic Order Quantity (EOQ) model which raised customer awareness of such considerations as stock-out cost, inventory holding cost, and order size. Roberts [1975] compiled an inventory of logistics costs including Origin Inventory, In Transit Inventory, Destination Inventory, Safety Stock, Stockout Costs, Loss and Damage, Perishability and Order Cost. The market has since moved beyond these simple notions of cost minimization. Following OECD [1992] (pg. 14):

> Recently, the demand of consumers has diversified and the concept of mass production has necessitated a shift from a "Production to Stock" concept to a "Production to Order" concept. Various combinations of goods options or varieties of goods are provided by the manufacturer, and the consumer can select the one which he or she thinks best. In other words, consumers can enjoy "custom-made mass- produced" goods. As a result, the inventory of each variety and its lot size of production have become small compared to that of the simply mass-produced variety.

The trend towards smaller lot sizes and more frequent shipments is not simply cost-minimizing behavior. It represents a fundamental strategic shift as manufacturers strive to differentiate their product offerings to tailor them to precisely meet the needs of particular consumer market segments. These changes are part of a well documented global pattern and include (OECD [1992] pg. 28 and 55):

- Concentrating on core business (and contracting out support functions such as logistics services)
- Geographical widening of product sourcing (while often simultaneously reducing the number of suppliers) resulting in an increasing average length of haul.

- Concentrating production and inventory (larger installations at fewer locations)
- The switch to road transport. In many countries road has gained market share at the expense of rail and waterway.

The switch to road transport is commonly associated with the adoption of modern logistics practice. The availability of improved highway infrastructure has accelerated this trend. However, if railroads can learn to deliver reliable service, then the longer hauls and concentration of production and inventory documented by OECD ought to favor more, not less rail movement in the future.

1.2 A Service Differentiation Strategy

Railroads can increase their market share by tailoring service to individual customer needs. Kwon [1994] established that different customers have different service requirements; therefore, there is no single definition of what is "reliable service". Each customer evaluates rail service in the context of their own shipping needs. Reliability is just one of a number of service attributes including cost, transit time, and safety (Mercer [1991]). Since most customers ship *cars* or trailers and not whole trains, this is the level at which traffic priorities need to be effective.

In the past, the concept of car-level priorities has come to be viewed by railroads as a high cost operational nuisance. Priority moves in yards have been handled on an un-planned exception basis, outside the normal flow of work, using very expensive methods such as "cherry picking" which require special switch engine moves for one or two cars. This has created resistance within operating departments to any expansion of car-level service differentiation.

However, an effectively managed priority system can help control costs as well as improve customer service. A priority system "levels the peaks" and "fills the valleys" of

demand, improving train capacity utilization, stabilizing flow throughout the network and dramatically reducing operating cost. It should yield an operational improvement rather than become an operational burden. The key is to identify priority cars *before* they are classified in yards, so that all cars can be sorted into the correct tracks the first time. This is technically feasible but requires high data integrity and a highly disciplined, schedule driven train operating philosophy.

A two-tier priority traffic system has been implemented by the French National Railways (Dejax and Bookbinder [1991]), who report they will add a third tier.

Today, priority service is generally provided on a dedicated train service network rather than on an individual carload basis, further described below. Dedicated automotive service networks have proven both the necessity and validity of the service differentiation concept. However, few industries generate enough traffic to support their own dedicated train networks. An operating strategy is proposed here so that reliable service, comparable to that currently provided automotive customers can be offered to general manifest freight customers but *without requiring a dedicated train network.*

1.3 The Need to Maintain Flexibility

Railroad management does not control customer demands, but must respond to them as best they can. Railroads' production processes are affected by factors outside management's control such as the weather. It is absolutely essential to be able to respond to gross fluctuations in demand and to operating incidents.

Nevertheless, operating flexibility should be implemented in such a manner as to support rather than frustrate achievement of service reliability goals. In particular, the current practice of basing customer service commitments on a given schedule, but then not operating the service after the commitment has been made, should be abolished. It has been

demonstrated many times (see, for example Kraft [1995]) that keeping to a regular schedule is necessary for reliable service. What is required to provide reliable manifest service is no mystery: the railroad industry already does it every day for intermodal and automotive
traffic. These highly reliable service networks have the following general characteristics:

- Scheduled network, precision adherance to schedule. Trains are often scheduled to accomodate some primary customer's needs, but other customers' traffic is usually handled on the same trains.

- No train cancellations, but extra sections can be operated as needed.

- A traffic priority system (as implemented by Santa Fe Railway, described in the next section) improves train capacity utilization and reduces the need to run unplanned extras.

The cost advantages of operating a regularly scheduled network include tangible benefits of locomotive, crew, and physical plant utilization; but also intangibles such as employee quality of life, morale and the reliability inherent in executing the same routine every day. If these benefits can be obtained without having to sacrifice train capacity utilization or terminal operating efficiency, this would achieve the "best of both worlds," and that is the objective of this research.

The most important difference between the current and proposed management processes would be the utilization of demand forecast information. This would facilitate a shift from a reactive to a proactive management style. Rather than waiting for actual demand to materialize before making a decision whether or not to operate (annul or consolidate) a train, the emphasis shifts to the question of when and where extra trains should be operated. Decisions to provide additional capacity beyond the base plan would

likely be made 24-72 hours in advance. Individual cars would be dynamically routed by an automated system, utilizing available train capacity on a priority basis.

This eliminates the trade-off between the cost of providing service and the level of reliability offered. It provides consistent and predictable service to those customers who require it, but still operating full trains, making efficient use of terminal capacity and controlling costs. In this context, "Quality is Free," but it requires a sophisticated information system as well as management commitment to operate scheduled train service.

Flexibility is retained by downsizing the base service plan, providing a "safety stock" of strategically positioned reserve resources, then selectively adding capacity when justified by the demand forecast.

1.4 Car Scheduling as a Service Management Tool

A car scheduling system computes a unique "trip plan" for each car using the railway operating plan. Based on car classification rules and train schedules, a trip plan shows the planned sequence of trains and yards, including scheduled arrival and departure times for each car. Trip plans are maintained in real time on a database which is available for operational planning purposes. Car scheduling can predict the number of cars moving on each train over the next 24 hours, and the number of cars passing through each yard.

Current systems build a trip plan in two steps. First the car is assigned a "yard block" or classification code. This determines to what yard the car will be sent next, usually based on the car's ultimate destination along with other criteria such as the commodity, car type and size. Most railroads use a lookup ("tag") table to apply these codes. However, Norfolk Southern (see Baugher [1993]) is using a shortest path algorithmic approach on a link-node blocking network to assign classification codes.

Once the block assignment has been made, all trains which can carry that block are identified. The car is assigned to the earliest possible outbound train without regard to train capacity. Most current systems will assign more cars than the train can carry.

However, Santa Fe Railway (Gorman [1994]) has cleverly approached this problem from another direction. Their system schedules intermodal shipments (but not railcars) using a "last train" or critical path approach. Shipments are initially scheduled to the latest possible train that still meets the delivery commitment, so operating management's problem is to figure out which shipments to advance ahead of schedule, rather than which shipments to "bump" to later trains. The choice of which trailers to advance is manually made by the intermodal terminal manager based on shipment "due date" and operating convenience. Critical path scheduling spreads current demand across several trains, so it may understate the amount by which today's train has actually overflowed.

Since the first car scheduling system was implemented in the mid-1970's on the Missouri Pacific Railroad [1977], these systems have spread across the entire North American railroad industry (Bailey [1995]). But Martland [1993] still found, in spite of these improvements, ". . . the overall level of performance for the boxcar does not appear to be dramatically different in 1990 from what it was in earlier years."

Bailey [1995] surveyed the implementation status of Car Scheduling systems at Class I railroads. Current car scheduling systems were judged to perform well at post audit or measurement functions — their initial design objectives — but were cited as lacking in their ability to assist in real time decision making. Figure 1.1, from Bailey's report, depicts a passive implementation of car scheduling, which measures service failure after the fact, but is not capable of intervening in actual operations to prevent failures from happening in the first place.

Fig. 1.1: Current Car Scheduling System Status
(from Bailey [1995])

Works Well	Does Not Work Well
Trip Plans	Real Time Workload Forecasting
Service Performance Measurement	Real Time Tactical Decision Support
Process Performance Measurement	Real Time Locomotive Requirements
	Real Time Crew Requirements

Bailey [1995] found that concerns with current car scheduling systems were prevalent, but did not seem to present a consensus view of what was most important to "fix." While some concerns identified by Bailey [1995] might be interpreted as a desire to tranform car scheduling into a more pro-active role, Bailey [1995] found that ". . . with the exception of total re-writes of the system, current new development still is primarily focused on measurement tools and customer information. Functionality that was initially designed and enhanced is still important, but real time and longer term forecasting of workload and resource requirements have become, and will continue to be important for future development."

Yet Bailey [1995] also found that ". . . repondents did not show confidence that railroads would successfully partner and work together for better solutions." Harker [1995] compares the implications of master versus real time train scheduling strategies, reporting that railroads are contemplating both of these plus all combinations. How can

industrywide consensus exist on what *car* scheduling enhancements should be progressed, if there is no agreement on what *train* operating strategy the system should support?

Car scheduling systems are routinely used today to develop schedule quotations, which form the basis of service commitments to customers. By adding train capacity constraints, and developing schedule quotations on a real-time basis, in essence a reservation would be taken. Real-time schedule quotation would close the gap between expectations and reality by verifying both the availability of equipment and line haul capacity to meet service commitments. Real-time quotation can suggest what promises can be made without setting up a service failure.

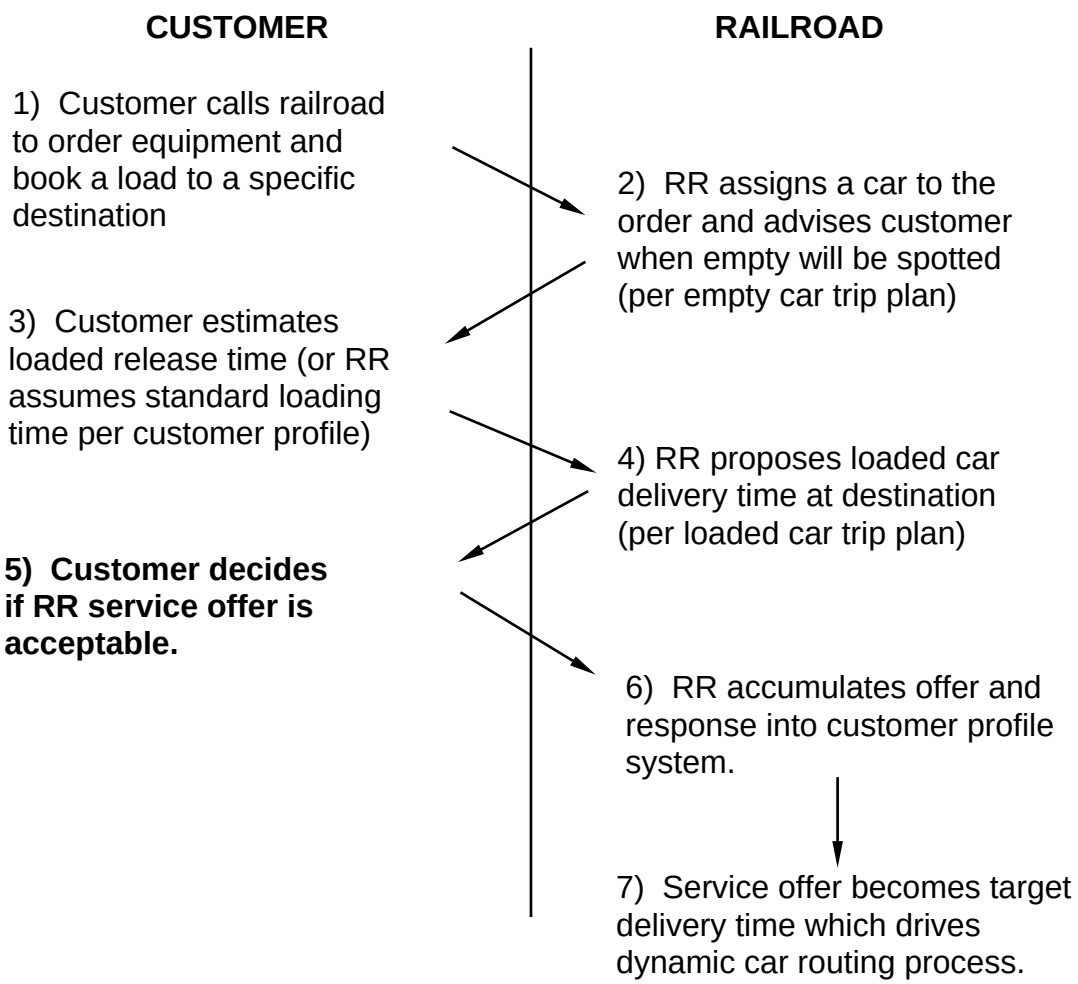
Since this capability would build on current business processes, it should be relatively straightforward to implement. The primary barrier is a technical one, since this process would greatly depend on the performance and reliability of the computerized shipment routing algorithms; which will be the focus of this dissertation. As shown in Figure 1.2, a schedule quotation step could be added into the empty equipment ordering process. Once a quotation has been developed, the customer has the option to accept or reject the service offer.

1.5 A Comparison with Airline Yield Management

Within this service quotation framework, a rail carrier might not necessarily want to allocate capacity on a first come, first served basis. Instead, it might make sense to hold some capacity in reserve if there is a reasonable expectation that high revenue, service sensitive business will call in later.

Prototypes for possible rail yield management systems already exist in other transportation modes. Campbell and Morlok [1994] have concluded that airline yield management concepts are applicable to rail but ". . . railroad yield management systems

Figure 1.2 : Proposed Real Time Service Commitment Process



should not be simple adaptations of airline systems. Instead railroad yield management should exploit variable capacity and differentiated service to improve cost and service performance." The rail freight yield management problem possesses several unique

characteristics, as compared with airlines:

- Airlines have a “mass market” orientation, whereas railroads provide freight service to a relatively small number of industrial customers. The railroad should know its customers individually; marketing may negotiate transportation contracts, specifying unique price and service characteristics required by each customer. These contracts establish a framework for a long term business relationship, whereas most airline yield management models view the customer relationship only in terms of the current transaction.
- While air travelers may be able to choose from several airlines, and general freight shippers might choose from a long list of trucking companies, most likely a rail customer is directly served by only one or two railroads. For certain commodities, it might not be economical to produce a product in a certain place or ship to certain markets except by rail. Still, when a contract is signed, the customer expects that transportation will be provided at the price agreed to, and that service levels will generally fall within a contractually agreed upon range of transit time and reliability.

This makes outright rejection of loads more difficult for railroads than if the customer had a large number of competitive rail options. The irony is that high value loads having many competitive alternatives (where trucking is an option) are ones the carrier is least likely to want to reject. Still, there could be room for a railroad to negotiate delivery times for individual shipments, within the overall parameters of the governing transportation contract.

Since prices are contracted in advance, when the customer calls the reservations center, the discussion should focus on the question of *when* service can be provided, not at what price. The railroad does not reject any offered loads. However, the *customer* can always choose to reject service offers and ship by another mode.

Shippers of low-rated commodities will find they can get the best service offers if they are willing to make reservations early, or commit to purchase capacity on a “take or pay” basis. By comparison, the leisure airline traveler generally gets their best deal by booking early, and a nonrefundable fare is the equivalent of the “take or pay” shipping contract.

- In the intermodal distribution chain, the presence of third parties prevents railroads from working directly with the ultimate customer. Suppose a shipper agent knows a particular load is not immediately needed by the consignee. The agent also knows this information is valuable to the railroad, that the railroad can derive an operational benefit from knowing it. The railroad must establish a differentiated pricing structure to induce the shipper agent to share the data. The shipper agent can force the rail carrier to share the benefits of yield management by controlling information.

In contrast, rail carload freight has a long history of strong price and service differentiation based on commodity. It is probably unnecessary for railroads to establish a differentiated pricing structure beyond what already exists: for the most part, railroads already possess the information they need to differentiate service without having to offer additional price incentives. A railcar-oriented implementation of yield management is likely to take on a distinctly different form, as compared to intermodal or airline applications of the same concepts.

- The requirement to account for empty repositioning makes freight yield management more complicated, in some respects, than its airline counterpart. On an airline between the same city pairs, full fare business travelers will *always* be preferred over leisure travelers paying a discounted fare. The airline would like to fill the airplane exclusively with business travelers, if it could. This leads to the airline concept of “nested” fare

classes (Smith, Leimkuhler, and Darrow [1992]), where a full fare ticket can always be sold to a business-class traveler even if the predetermined business class allocation happens to be sold out.

This strict hierarchical relationship may not always hold true in the case of freight, because of the cost of empty equipment repositioning. The first few vehicles into a zone have a high probability of finding backhauls, representing highly valuable business. Additional loads become less profitable, such that some other business may take priority at a certain point. Most generally, train capacity allocation and empty equipment distribution should be determined as a simultaneous solution to a single mathematical program.

Unfortunately, much rail freight moves in special or dedicated equipment, or rail traffic lanes are so imbalanced that the marginal empty return rate is 100%. This simplifies railroad yield management by decoupling train capacity allocation from empty equipment distribution. Every load will have the same profitability independent of the number of loads sent, as is assumed here. Simultaneous optimization of train capacity allocation and empty equipment allocation will be left for future research.

- Rail carriers have the ability to displace a lower priority shipment , even at an intermediate terminal, in favor of a newly arrived load, provided sufficient slack time exists in the delivery schedule to allow this. Airlines do not normally “bump” passengers based on the fare class of the ticket they hold.

At least three distinct approaches to implementation of yield management are reported in the literature. These include: Capacity Allocation by Fare Class, Bid Price, and Direct Price Adjustment.

The yield management literature has been strongly influenced by the capabilities and limitations of airline reservations systems. As reported by Williamson [1992], most airline reservations systems were designed 20-30 years ago when the market environment

was much simpler. These reservations systems allow for physical control of seat inventories at the fare class and flight leg level, rather than by origin-destination. A major emphasis in the literature has been on the development of good heuristics which would approximate a "network optimal" solution but actually manipulate seat allocations at the fare class/flight leg level.

Williamson's [1992] dissertation evaluates a variety of different approaches using a rolling horizon simulation model. It includes an broad survey of current airline yield management approaches, providing an excellent introduction to the field. Belobaba's [1987] "EMSR heuristic" allocates capacity among different fare classes by setting equal the expected marginal revenue of each fare class on each flight leg. The reasoning underlying this strategy is well explained by Elkins [1991] (pg 7-8):

> If we reserve a unit of capacity (an airline seat or a hotel room or 30 seconds of television advertising time) for the exclusive use of a potential customer who has a 70 percent probability of wanting it and is in a market segment with a price of $100 per unit, then the expected revenue for that unit is $70. Faced with this situation 10 times, we would expect that 7 times the customer would appear and pay us $100 and 3 times he would fail to materialize and we would get nothing. We would collect a total of $700 for the 10 units of capacity or an average of $70 per unit.
>
> Suppose another customer appeared and offered us $60 for the unit, in cash, on the spot. Should we accept his offer? No; because as long as we are able to keep a long term perspective, we know that a 100 percent probability of getting $60 gives us an expected revenue of only $60. Over 10 occurances we would only get $600 following the "bird in the hand" strategy.
>
> We should never sell a unit of capacity for less than we expect to receive for it from another customer, but if we can get more for it, the extra revenue goes right to the bottom line.

The second approach, the "bid price" method, works by calculating "opportunity cost" for units sold. Utilizing this method, one should never sell a unit of capacity for less than its opportunity cost, even though the direct revenue impact may be positive. In Elkins' example, selling the unit for $60 would produce a *net loss* of $10 because the opportunity cost for that unit is $70. Following Williamson [1992] (pg 90-92):

The idea behind the bid price approach is to establish a "cutoff" value for each flight leg which can be used to make decisions whether to accept or reject different ODF (origin-destination-fare class) requests. The difference in the methodology of the bid price approach, when compared to other conventional seat inventory approaches, is that ODF inventories are either open to bookings or closed; there are no explicit booking limits for different ODF's. For a single leg itinerary, a fare class is open for bookings if the corresponding fare is greater than the bid price, or shadow price, for the leg. For a multi-leg itinerary, the total fare must be greater than the sum of the bid prices from the respective flight legs it traverses.

One advantage of the bid price approach is that it is a very simple method of managing seat inventories. Hence, it would be very easy to implement in a reservations system when compared to OD and fare class approaches. The disadvantage of the bid price approach, however, is its open/closed control philosophy. If a given ODF passes the bid price criteria, that ODF remains open to bookings until the bid prices are revised. Thus, in order for the network bid price approach to be an effective seat inventory control approach, frequent revisions would be necessary, requiring both reoptimization and reforecasting. For a truly optimal system, revisions would be necessary on a real-time basis.

Powell et al [1988b] (pg. 34-35) has implemented opportunity costing, called "Total System Contribution (TSC)", in a real-time truck load planning system. Powell defines:

TSC = r(i,j) - c(i,j) + VP(j) - VM(i)

where

r(i,j) = the revenue earned on a load going from i to j
c(i,j) = direct operating cost of hauling a load from i to j
VP(j) = the marginal contribution of an additional truck at j
VM(i) = the marginal contribution of one less truck at i

The last two terms, VP(j) and VM(i) represent the opportunity costs for alternative uses of the truck. Thus (pg. 35):

Apparently profitable loads may have a negative TSC if there are even more profitable opportunities being passed up. At the same time, a seemingly poor load can appear attractive if the best alternative is to hold the truck in a poor region or move it empty.

The TSC statistic is not only an intuitively reasonable measure of the value of a load to the system, it also rests on a solid mathematical foundation. The expression is drawn from optimization theory where it is known as a shadow price or reduced cost. The practical application of this approach is in its ability to give the planner the value (to the entire system) of each load over the entire planning horizon. This statistic, in conjunction with longer-term considerations of customer relations, is used to make clear accept-or-reject recommendations.

The TSC statistic also applies to empty movements. Typically, deficit regions will have a high marginal contribution for each truck, so empty moves into such areas will have positive TSC's. If an empty move produces a negative TSC, that move should not be undertaken. Powell [1988b] (pg. 23) concludes:

> How much a given load contributes to the carrier's profit should be determined by taking into account not only the revenue minus the direct cost of that load, but also the expected earnings of the trailer upon its arrival at the destination.

Powell [1987] proposes a "Regional Impact Model" to determine, under uncertainty, the expected contribution associated with each unit of incremental capacity in a zone. (This is known as the expected recourse function.) The general principle is that additional vehicles in a zone have diminishing marginal returns. This assumes the most lucrative opportunities will be taken by the first vehicles to arrive in a zone. Later-arriving vehicles have a lower probability of finding a good outbound load at their destination, and a higher probability of having to either reposition empty or settle for a marginal load.

Although Powell [1987] never labeled it as such, his model clearly implements a bid-price yield management approach, applied to a freight transportation problem in the trucking industry. This dissertation will propose a bid-price approach for real-time management of rail freight traffic.

A third approach, direct price adjustment, is proposed by Gallego and Van Ryzin [1994a], [1994b]. In this approach, rather than choosing from a menu of pre-established prices, each instance (flight leg or train departure) would be priced individually. In an optimal solution, the price should jump after every sale because there is one less unit to sell in the remaining time until departure. Gallego and Van Ryzin [1994a] show that a fixed price policy (where a price, once established for a specific departure, doesn't change) is optimal for the deterministic case, and is asymptotically optimal for the stochastic case as

either time or volume approaches infinity. Gallego and Van Ryzin [1994b] extend this result to the multi-leg, multi-fare class case. They suggest that current yield management practices might tend to approximate this model's "effective price" which they define as the weighted average price of all tickets sold.

Gallego and Van Ryzin's formulation is extremely data intensive. It requires that the entire demand curve be known for every O-D-fare class market as a function of price; whereas, as a practical matter, it is probably difficult enough to simply derive a reliable "point estimate" demand forecast at a single given price. The network pricing formulation in [1994b] is even more data-intensive, allowing a price vector demand function which should include all cross-elasticities of demand.

In order to computationally solve this model, Gallego and Van Ryzin [1994b] made the simplifying assumption that demand functions are separable: that is, demand for any given flight depends only on the price for that flight, and not on any other flights' price. But an airline traveler's decision is certainly not independent of price of different flight options between the same origin-destination pair. For example, consider the case where one could fly from Baltimore to Pittsburgh at 8 AM for $300 or fly at 11 AM for $150. Some customers are likely to shift from the 8 AM flight to the 11 AM flight to take advantage of this price differential.

In the railroad context of this research, suppose the 8 AM train is filling up and a customer calls offering low-priority freight. The rail carrier might want to offer that customer a booking on the 11 AM train instead of the 8 AM, if there is a high probability the customer will accept the 11 AM service offer.

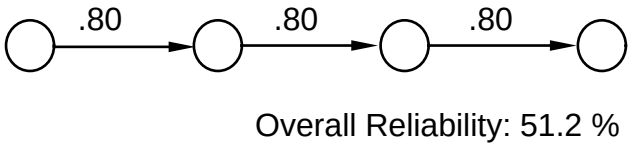
Capturing the interdependence among different "traveling options" is absolutely central to this research. As well, this research assumes that all prices have been set in advance, based on governing transportation contracts and are not subject to renegotiation

on an individual load basis. This research, therefore, fits into a more traditional "yield management" framework, choosing from a selection of preestablished prices and fare classes, rather than attempting to optimize railroad pricing decisions in real time.

1.6 Multi-Pathing as a Means to Improve Reliability

Railroads have a complex production process as compared to other transportation modes, particularly trucking. Single car shipments, in particular, have many opportunities to miss connections or otherwise fall behind schedule. Suppose a car must pass through three intermediate yards before reaching its destination, as shown in Figure 1.3. Each connection has a reliability of 80%. The combined reliability is the product of individual connection reliabilities, .8 x .8 x .8 or 51.2%. Whenever links are connected in series, overall reliability is *lower* than that of any individual link.

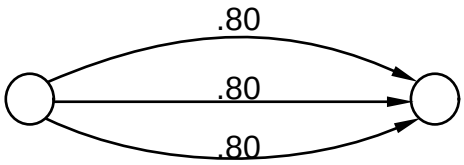
Figure 1.3: Three Links in Series



Overall Reliability: 51.2 %

In contrast, Figure 1.4 shows three links in parallel, each link again having a reliability of 80%. Mathematically, the reliability of this system is 1 - (.2 x .2 x .2) or 99.2 %. Whenever unreliable links are connected in parallel, overall reliability is *higher* than that of any individual link.

Figure 1.4: Three Links in Parallel



Overall Reliability:

In railroading, the most successful strategy for reliability improvement so far, has been to reduce the number of links or yard handlings required by any given car. Yet where traffic volume has not been sufficient to support a “bypass block” or “run through” train, it has proven very difficult to improve individual link or connection reliabilities to extremely high levels required to compete with trucking, using a “single pathing” approach.

One way of improving connection reliabilities is through introduction of a traffic priority system. This system is only appropriate for high value, extremely time sensitive traffic where either schedule slack cannot be introduced, or all the slack has been used up. It becomes prohibitively expensive to try to improve reliability for all traffic using a single pathing approach.

Mathematically, the “multiple pathing” philosophy can succeed even if individual link probabilities are not very good, but requires additional slack time in the origin-destination schedule. Fixed table-driven shipment routing (or “blocking”) strategies are too inflexible and cannot utilize all alternative paths which might exist in the network. To make best use of available slack time and gain the most reliability benefit, dynamic car routing is needed in order to identify and utilize all available paths.

1.7 Research Statement

The focus of this research will be on the development of practical solution algorithms for very large scale, real world railroad shipment routing problems. The primary research contribution will be the development of a new algorithmic approach to implement a railway reservations-based scheduling process. The goal is to establish the technical feasibility of capacity-constrained car scheduling approaches for real world rail networks; to develop practical, yet mathematically based solution approaches which can produce near optimal results in an acceptable time frame. In addition to measuring duality gaps and solution times directly, the algorithms will be tested by integrating them into a stochastic rolling horizon simulation. A published MIT test data set (Kwon [1994]) will be used to define a twelve yard train service network and traffic flows. This will establish the computational effectiveness of the proposed solution algorithms, and allow measurement of operational impacts on train capacity utilization, transit time and service reliability.

1.8 Structure of the Dissertation

Chapter 2 reviews related railroad and transportation literature. A summary at the end of the chapter focuses on methods for reducing transit time variability.

Chapter 3 defines a mathematical formulation based on a multicommodity network flow (MCNF) model, and proposes a specific strategy for implementing a freight railroad reservation-based business process. The formulation is decomposed into two separate subproblems, one deterministic, the other stochastic (linked through dual prices and joint capacity constraints).

Chapter 4 proposes Lagrangian Relaxation based solution algorithms for the two subproblems. Computational performance will be evaluated operating both stand-alone and

embedded in a rolling horizon simulation model. This chapter also includes a review of related mathematical programming literature.

Chapter 5 presents results from a rolling horizon simulation test of the two algorithms, measuring the attainable level of improvement in service reliability, train capacity utilization and railway profitability resulting from a reservations-based business process. The car scheduling procedures will be fine tuned to optimize their performance in the simulated real-time application.

Chapter 6 summarizes and concludes the research results. This chapter evaluates the research contribution, usability and value to the railroad industry, and possible future research directions are proposed.

CHAPTER 2

Rail-Related Literature Review

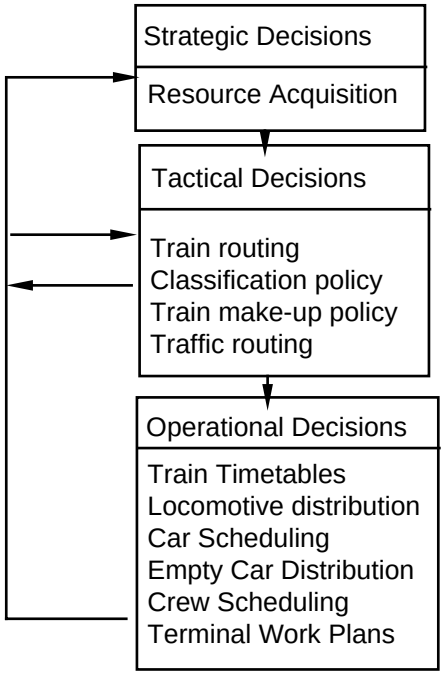
2.1 General

Assad [1980a] classified rail planning decisions into strategic, tactical and operational categories based on planning horizon, investment requirements and the level of decision making. Typical decisions in each category are shown in Figure 2.1. This review will concentrate on the tactical and operational areas. Topics to be covered include: mechanical reliability, operating plan development or network design, empty equipment distribution, train dispatching, advanced train control systems, and rail yard operations.

Several studies will be cited to dispel the myth that mechanical reliability is the primary cause of railroad service failures. Instead, the literature strongly suggests that rail technology is mechanically capable of operating at a much higher level of reliability than currently observed. Factors which cause service failures are largely under management control. Indeed, to the degree that mechanical problems do contribute to service failures, many of these might be preventable through better preventive maintenance policies (Little et. al. [1991] pg. 249.)

Rail network design has traditionally been a very active area of rail research. It is possible that past emphasis on finding an “optimal plan” has been misplaced. An optimal plan is not necessary to provide reliable service, however, any plan must be at least *feasible* to have any hope of successful implementation. To produce reliable service, a simple feasibility check is needed to identify and resolve resource conflicts within the service design process, rather than waiting to resolve these conflicts at execution time. Very little research into plan feasibility at the network level was found, for example, Gohring [1971] developed a locomotive and caboose cycling application at the Southern Railway to ensure feasibility of power assignments to a set of train schedules.

Figure 2 .1 : Hierarchical Decisions in Rail Operations
(from Kwon [1994])



Empty equipment distribution is crucial to railroad service reliability. If the shipper cannot get the empty placed on time or at all, it is unlikely that the load will arrive at its destination on time. State-of-the-art optimization programs designed for empty equipment management explicitly incorporate both demand and transit time uncertainty. These models are closely related to the yield management approaches proposed here.

Line and terminal operations literature is reviewed because these are the places where the plan is translated into reality. An understanding of the capabilities and limitations of line and terminal operations is essential to develop an efficient yet reliable plan. Jovanovic and Harker [1990] first proposed schedule adherance, rather than delay minimization as the objective function for train dispatching and identified infeasibilities in real-world operating plans, for example two trains scheduled to meet in a place with no passing siding. The feasibility of terminal operating plans was researched by the M.I.T. rail group (Duffy [1994], Schlenker [1994] and Dong [1994]).

This “plan feasibility” question should be generalized to complex network-level issues such as whether there are sufficient locomotives and crews to cover all assignments. Assuming that infeasibilities in the network plan will be identified, the service design department will need a means of resolving these conflicts. New methodologies being developed for track time pricing, for example the auction mechanism proposed by Harker and Hong [1993] should be extended to apply to the allocation of other resources such as locomotives and crews as well.

2.2 Mechanical Component Reliability

Basic research into mechanical components has yielded many advances such as continuous welded rails, concrete ties, double stack cars, aluminum coal cars, better braking systems and powerful, fuel efficient locomotives. There is no doubt that these advances have contributed substantially to the economics and productivity of railroad

operations, and to the industry's ability to effectively meet customer needs. Yet mechanical component reliability continues to be cited by some as the primary cause of railroad service failures.

However, the evidence indicates that freight car, locomotive or track reliability problems are not the root cause of most origin-destination service failures today. Martland [1993] (pg. 13) found:

> It is noteworthy that even if the railroad had perfect technology, only 30 percent of the delays would disappear; 65% of the delays require better management of resources (terminal management, train management, and power distribution).

Sheaffer and Stern [1986] (pg. 113-114) studied locomotive reliability on the Grand Trunk Western Railway and found:

> Of the 1,097 locomotive failures which occurred during the study period, only 356 of these caused a train to be delayed. When a train was delayed due to a locomotive failure, the train was delayed approximately 65 minutes.

> The impact of locomotive reliability on service reliability was shown to be very small. This is a result of relatively infrequent failures resulting in train delay and delay times which are generally short. This result is consistent with Belovaric's findings. His study on the Boston & Albany mainline of Penn Central found very little line haul transit time effect resulting from equipment reliability.

> Service reliability is influenced to a much greater extent by the availability of locomotives (as compared to on-line mechanical failures.)

Martland [1993] (pg. 13) estimated which root-causes contributed the most delay in a sample of general merchandise traffic:

- *Power availability* delays, which include delays to trains due to locomotives not being in position to move the requisite tonnage (24.4% of all train delays)
- *Terminal* delays, including yard congestion, cars not switched in time, cars moved on other than scheduled trains, etc (20.2%)
- *Train* delays, which reflect management decisions regarding which trains to run, and with what resources, including maximum tonnage, annulled due to lack of traffic, train consolidations, etc. (20%)
- *Mechanical* delays, including shoppings of defective cars or locomotives (16%)

- *Line* delays, reflecting delays enroute such as track work, curfew, train meets, etc. (13.3%)

- *Other*, including derailments, unknown causes, no bills, etc (6.1%)

Martland [1993] analyzed the causes of delays to intermodal shipments moving on TTX Company flat cars. He concluded that mechanical failures are not the root cause of most service unreliability, thus the solution of the service reliability problem requires development of better "network management" strategies rather than a continued focus on railroad hardware development (pg. 17):

> 80% of the delays to high priority, high quality shipments were left unexplained after accounting for mechanical delays and holiday disruptions. This suggests that mechanical reliability is not the root cause of unreliability, even for cars which are rarely in terminals.
>
> All the recent work, then, suggests that freight reliability appears to be more a matter of management than of railroad technology. The delays to cars and the delays to trains which can be attributed to failed equipment, track, or technology are modest, while those which are due to the management of resources constitute a clear majority. This suggests that the focus of research to improve service reliability must be shifted in a manner which will provide tools to manage the
> ` railroad rather than to technologies and hardware.

Evidence suggests that many in-service component failures are, in fact, preventable through planned maintenance policies, not inevitable. Following Little et. al. [1991] (pg. 249):

> Railroad car owners could achieve considerable performance improvements at lower costs by replacing on-condition policies with a planned maintenance strategy. Better results can be obtained by using statistical methods to estimate the characteristic lives of components and then aggressively replacing those components which are expected to fail during the next maintenance
> ` interval.

In an interview with Railway Age [1993], TTX Company's Bob Hulick complained that:

> AAR specs are for minimum acceptable. But we have the right and in some cases the duty to exceed those specs, for reliability and service considerations. We willingly pay more for a premium component. The problem Gus Welty wrote about is that if the component is replaced in service, it will be replaced with the AAR minimum, and the reason is that the billing that's allowed is only for the minimum AAR-spec product.

Ray Burton, president of TTX, reinforced Bob Hulick's point:

> This situation is inhibiting people from making improvements on important components. The rules and regulations in the industry inhibit progress. Inventors know that the rules inhibit the sale of a better product. And this is something that I think we, as an industry, really have to address.

2.3 Tactical Operating Plan Development

The literature in this field is dominated by a variety of mixed-integer programming models by Morlok and Peterson [1970], Bodin [1980], Assad [1980b], Crainic [1984] [1986], Haghani [1989], Keaton [1989] , and most recently, Gorman [1995]. Powell [1986c] addresses a closely related network design problem in LTL trucking.

Morlok and Peterson's original formulation [1970a] attempts to optimize the blocking plan, train make-up policy and train schedules simultaneously. This research was published in the TRF Proceedings under an incorrect title [1970b] and unfortunately most later researchers seemed to be unaware of this work. The paper is worth rediscovering since it represents the only known published formulation which explicitly solves for train schedules as well as for average frequency. Only in very recent times has Gorman [1995] reopened this line of research.

Later models by Assad [1980b], Crainic [1984] and Keaton [1989] do not solve for schedules but for train frequency only. These models do not provide a complete solution to the service design problem as Morlok's formulation would. By relaxing the maximum train length constraint, Keaton was able to obtain a tight lower bound on the optimal solution. However, this lower bound was obtained at the price of relaxing constraints which are critical to the real-world decision making process. Crainic [1987] converted his model into a strategic planning tool, STAN, and applied it on a Brazilian railroad study.

Haghani [1989] proposed a simultaneous solution to the train routing, makeup and empty car distribution problems. Again, this model solves for train frequency only. Since the most distinguishing feature of this model is the inclusion of empty flows, it would have been useful to see some analysis of how empties would have been routed with separate models versus how they were routed in the simultaneous solution. If this comparison was ever performed, however, the results were not reported.

Keaton [1992] developed a model for scheduling train arrival and departure times at one terminal, apparently to be used as a post-processor for his network model results. He recognized difficulties of extending the approach to a network of terminals. Hong and Harker [1990] reopened the train schedule line of research by proposing a two-stage network scheduling model, taking line and terminal capacity constraints into account, but development work on this model was never completed.

A common weakness of all these optimization approaches is their failure to address service reliability. The models in general minimize cost, and one cost component is average car time spent in yards. None of the models attempt to compute variability in delays or in the origin-to-destination trip time distributions. Keaton [1991] published a rather discouraging paper concluding that "significant reductions in transit time will require a large increase in the number of train connections and operating cost." But Keaton's analysis did not look at methods of improving service reliability which might have greater value to shippers and a lower cost to railroads.

The LTL trucking network design problem is mathematically related to the railroad train scheduling problem. Powell implemented interactive optimization systems for Ryder/PIE (Powell and Sheffi [1989]) and later for Yellow Freight (Braklow, Graham, Hassler, Peck, Powell[1992]) to find opportunities to bypass breakbulks, and to perform interactive "what if" analysis in real time on various bypass strategies.

Keaton [1994] examines the structure of the LTL trucking industry and concludes that two distinct types of service networks have evolved: direct service or at most one transload for regional trucking carriers, and indirect service via breakbulks for the long distance LTL carriers. The implications for rail network design should be obvious. Simply put, if the distance is short, the savings by running longer trains may not fully recover the cost to consolidate the traffic in terminals, in this case it is more advantageous to operate short trains on frequent intervals directly between nearby terminals, without any intermediate yards. For long distance trains, line haul costs dominate terminal handling costs, therefore it becomes worthwhile to consolidate traffic in intermediate terminals to handle the long distance traffic in fully utilized trains.

Recent efforts have utilized "artificial intelligence" rather than traditional optimization approaches. They also derive specific schedules as opposed to frequency only. Huntley, Brown, Sappington and Markowicz [1993] developed a simulated annealing approach to grain train scheduling for CSX Transportation. This model suggests consolidation/gathering points to aggregate partial trainload shipments into full unit trains.

Gorman [1995] has developed what might be the first commercially successful "start from scratch" operating plan developer using a modified genetic algorithm approach. He reported that this approach yielded a 4-5% improvement in cost, and better on-time performance compared with the Santa Fe Railway's current plan. However, the model-generated plan was so different from the current operating pattern that it was unclear how to implement it. Gorman's recent efforts have focused on how to limit the scope of changes to make the recommendations more implementable.

The heuristic "automated blocking model" (ABM) design proposed by Kornhauser and Mayewski [1983] and refined by Van Dyke [1986] represents the current state-of-the-practice in the rail industry. ABM model usage more closely resembles a simulation rather

than an optimization process. The Norfolk Southern Railway [1992] is incorporating this design into their real time car scheduling system, saying that the shortest-path algorithms built into the model "are very fast . . . but the shortest paths have proven to be quite accurate."

The Service Planning Model approach by McCarren and Martland [1980] incorporates "PMake" functions which estimate the probability of making connections in freight yards. The SPM is one of the few railway network models to address trip time reliability directly. SPM input is a table of *average* daily origin-destination car volumes; uncertainty in demand is handled implicitly in the "PMake" function.

Simulation-based approaches seem to be reemerging for both tactical planning and operational control in the railroad industry. Simulation approaches were unsuccessfully tried in the early 1970's: the AAR and L&N Railroad each developed highly detailed network simulation models. In those early years, simulation proved impractical due to the models' high computational intensity, data requirements, and poor user interface.

Today's computers can execute the models quickly; carriers have extensive databases from which input data can be gathered; and user interface technology has greatly improved. Theoretical developments by Ho [1992] and others have even endowed simulation modeling with optimization capabilities.

A stochastic network simulation model can handle any input distribution of demand and translate that into an output distribution of transit time — something that neither the PMake nor the queueing approaches can do. Simulation approaches have been used in ConRail's planning models developed by American Airlines Decision Technology; the MIT Rail Group developed a detailed stochastic network simulation model (see Kwon [1994]) to support future research needs.

Strasser [1993] used a simulation model to evaluate whether it was better to have a policy of holding for late arriving inbounds or to always depart the outbounds on time. She coupled a simulation-based "Railroad Performance" model with a "Shipper Modal Selection" model to determine the effect of different policies on railroad costs and market share. She found that "the current combination of 24-hour train frequency, 3-hour yard time and no-hold dispatching promotes the most negative effect on shipper modal selection in terms of reliability and transit time. A change in scheduling can have only a positive effect on shipper modal selection." Strasser advocates allowing more connection time and implementing a "hold for late arrival" policy as a means of improving service.

Kwon [1992] reached a similar conclusion, saying, "Always running trains on schedule is not the best policy; delaying some trains allows more connections to be made and thereby improves network reliability. However, if line-haul performance is improved, then schedule-oriented train dispatching policies perform better; it may even be possible to reduce the slack in scheduled yard time and have more schedule-oriented train departures without hurting network reliability if trains run on time. Finally, if railroads over-emphasize capacity utilization in train dispatching (by using extreme long-hold policies and allowing frequent train cancellations), they will exacerbate network reliability."

2.4 Empty Equipment Distribution

Much of the operating plan design literature ignores demand variability (by using "average day" volumes) and with the exception of the SPM, ignores transit time variability as well. In contrast, variability is explictly handled in most literature dealing with empty equipment distribution and management.

Philip and Sussman [1977] applied a standard inventory stockout model, based on the tradeoff between holding and stockout costs, to estimate the optimal inventory of empty

cars to be held at one terminal. Clearly, however, there are only a certain number of freight cars to go around, so really the question is how this fixed supply ought to be optimally distributed among all the terminals on the network. Mendiratta and Turnquist [1981] address this problem by suggesting a two-level dual pricing approach, coupling a network flow model with a terminal inventory model.

Jordan and Turnquist [1983] extend Mendiratta's work to capture both the network flows and inventory considerations in a single formulation based on the work of Cooper and LeBlanc [1977]. Powell [1986a] explored the application of this framework to the dynamic vehicle allocation problem in the trucking industry. Powell [1988] later commented "What is most disturbing about this line of research is that it fundamentally represents a very simple set of assumptions and yet produces an alarmingly complex model. There does not appear to be much room left for further relaxing the assumptions and still having a workable mathematical model." He began working on an alternative formulation of the problem as a classical Markov decision process. This literature through the mid-1980's is reviewed by Dejax and Crainic [1987]. Powell [1991] gives a survey of TSP and cycling based vehicle routing applications in the trucking industry.

Powell's newer formulations are based on discrete demand distributions, usually Poisson, rather than the continuous "normal" distribution. This research eventually led to a system installed at North American Van Lines (Powell, Sheffi, Nickerson, Butterbaugh, and Atherton [1988b]). Frantzeskakis and Powell [1990] proposed the "SLAP" algorithm, which works by decomposing the problem by region and making a linear approximation to the expected recourse function. However, for large multi-stage problems, the successive linear approximations can accumulate errors, therefore, a more precise method, SCAM, has been developed by Powell and Cheung [1992]. This recent research is summarized by Powell, Jaillet, and Odoni [1993]. Powell [1995] is now applying computer simulation

using stochastic gradient techniques to both the Dynamic Vehicle Allocation and Network Design problems.

Meanwhile, a number of rail carriers have implemented equipment distribution systems based on deterministic linear network models, see Turnquist and Markowicz [1989] and Markowicz and Turnquist [1990]. The "Multilevel Reload System" at the Association of American Railroads (Glickman and Sherali [1985]) manages a nationally-pooled fleet of automobile-carrying railcars; this model also includes "equity" constraints to ensure fair distribution of savings to each participating carrier.

Nozick [1992] studied service differentiation strategies for intermodal carriers, including simultaneous optimization (for deterministic demand) of empty equipment distribution. This model has been used to address the efficiency and organization of intermodal drayage, and the effects of a traffic priority system on fleet sizing and intermodal car and trailer fleet management.

Campbell [1996] researched the application of "yield management" techniques to intermodal applications, including the allocation of trailer, railcar and train slot capacity. His research extends yield management techniques originally developed for fixed capacity networks, adapting Belobaba's [1987] "EMSR heuristic" to apply to flexible capacity networks as well.

Holmberg, Joborn and Lundgren [1996] propose a deterministic empty equipment distribution model based on a time-space network of train schedules. All trains go directly between terminals; if a train connects more than two terminals, it is treated as several trains between consecutive terminal pairs. Thus, the model does not incorporate a blocking network, but it does enforce segment-specific train capacity constraints. It is assumed that loads will have first priority on available train capacity, therefore the empty problem is solved sequentially, utilizing remaining train capacity after all loads have been routed.

Inclusion of train capacities require that the problem be solved simultaneously for all empty equipment types; distribution decisions can no longer be decoupled by car type.

The problem is formulated as an integer multicommodity network flow problem which is solved using three different algorithms: the subgradient method, a branch and bound approach using network CPLEX, and a linear relaxation using PPRN. Duality gaps less than 1% were customarily attained using the subgradient method; but the branch-and-bound CPLEX code solved the test problems exactly within an acceptable time frame. The problems were sized based on the Swedish Rail network; it is not clear how well these computational results would scale to a U.S.-sized problem, but increasing the problem size might tend to favor the subgradient approach (see Ali, Kennington and Shetty [1988]).

Holmberg, Joborn and Lundgren's [1996] formulation of the empty car distribution problem is highly compatible with the approach used in this dissertation for the loaded car routing problem. Both approaches use a similar problem structure and solution algorithm, which suggests the future research possibility of attempting a simultaneous solution. The most significant weakness of this approach is its reliance on a deterministic demand forecast, which renders the results highly sensitive to errors in the demand forecast.

2.5 Train Dispatching and Control Systems

Three approaches to rail line modeling: analytical, simulation, and optimization, are documented in the literature. Optimization approaches have been specialized to track time pricing. We will not attempt a complete bibliography here but just a general overview.

- **Analytical models**

Petersen [1974] proposed the first probability-theory based model of train delay on single track lines, assuming random arrivals and departures throughout the day. Kraft

[1988] extended this approach to take multiple train interactions into account and compared the results with myopic and optimized train dispatching. Chen and Harker [1990] extended Petersen's model to apply to the case of a scheduled operation. Hallowell [1993] further refined this formulation to dynamically determine train priorities based on schedule adherance rather than a fixed priority dispatching scheme. Hallowell's model has been proposed as a means of generating target times for a real time optimizing train dispatching system.

Peat Marwick Mitchell Co. [1975] performed a parametric analysis on the output of their simulation model, summarizing the results in equation form. This approach is regression rather than probability theory based.

- **Event oriented simulation models**

This category includes the Canadian National Route Capacity Model (Welch and Gussow [1986]), ALK's Line Capacity Analysis System (Van Dyke and Davis [1991]), Petersen and Taylor's [1982] model, and Kraft's [1982] "Jam Capacity" analysis.

A general problem with this type of model has been the tendency to "lock up" or deadlock with no forward train moves possible. Published research on lockup prevention is quite sparse, although the problem itself is well known. The only paper dealing explicitly with this subject is the one by Petersen and Taylor [1983]. Those authors propose a simple sufficient, but not necessary condition, "simple meetability" for identifying potential lock up situations. However, Welch and Gussow [1986], found that Petersen's test was overly restrictive in some circumstances, and "prevents many moves which are both feasible and desirable . . . This resulted in situations which had only one local resolution, did not represent reasonable dispatching and involved unnecessarily long train delays."

- **Optimization models**

This includes the early work at the Southern Railway (Sauder and Westerman [1983]) whose objective function was to minimize the weighted sum of train delays. Kraft [1987] embedded branch-and-bound into a simulation model, suggesting that train delay should be minimized subject to all scheduled trains meeting their schedules. Jovanovic and Harker [1990] proposed schedule adherance, rather than delay minimization as the objective function, requiring that all trains have schedules. Higgins, Kozan and Ferreira [1995] refined the branch and bound approach by suggesting branching priorities and improving the lower bound, as well as studying methodologies for passing siding location.

Influenced by the work of Harker, many recent optimization models have used as their objective function "minimize tardiness relative to schedule" rather than delay minimization. One criticism that has been leveled against schedule-adherance based train dispatching is that any reliability benefit would be dissipated in yards. However, Martland and Smith [1990] (pp. 286-287) found that:

> More reliable train operations and better ETA's would improve train connection reliability and allow more efficient allocation of yard crews and other terminal resources. While dramatic improvements in overall service should not be expected, reductions of perhaps 6 to 12 hours in average trip times and substantial improvements in reliability appear to be realistic.

Citing studies on the Southern and Boston & Maine, they conclude:

> There is clear evidence that reductions in line time would not be dissipated in terminals, but instead would quite possibly lead to additional savings in terminal time.

Jovanovic and Harker [1990] noted that ". . . on a fully-signalized railroad territory, there is not a great need to install new technology (such as ATCS) other than software in order to achieve reductions in train tardiness and delay . . . adequate information and control capabilities are often provided by the existing control system."

Schedule-adherance based dispatching systems require the specification of target times of arrival for every train at key points. Much recent research has focused on how to generate and maintain an appropriate set of target times in real time, a prerequisite to the implementation of tardiness-minimizing dispatching algorithms.

Hallowell [1993] starts with the current positions of trains and projects their positions forward in time, taking into account both free running and expected interference delay, using an analytical model to project the latter. Kraay and Harker [1994] propose a large scale nonlinear program to minimize the weighted deviation of target times relative to schedule, subject to feasibility and minimum running time constraints.

Target times produced by these two formulations behave quite differently. Consider, for example, a loosely constrained case where all trains have ample schedule slack. If all the feasibility constraints are non-binding in Kraay's formulation, that formulation should simply return the original strategic schedules for use as tactical targets. In contrast, Hallowell's method would start with the current positions of all trains, add free running and an allowance for reasonable interference delays, and return a more aggressive set of target times, all tighter than the actual schedule. This aggressive set of targets should make the operational dispatching algorithm work harder, find solutions which produce less overall delay, but also require more CPU time to solve.

- **Track Time Pricing**

Track time pricing is the development of opportunity costs for the use of railway track infrastructure, for use of specific "schedule slots" at a certain place and time. Interest in track time pricing has been motivated by the creation of separate rail infrastructure authorities in Sweden and other European countries; and in the United States, by the move to distinct rail freight "service networks" and the 1996 renegotiation of the Amtrak basic agreement.

Optimization represents a natural approach for this work, by utilizing the dual variable as a pricing mechanism. A primary motivation behind the development of pricing mechanisms is to allow for decentralization of decision making, so game-theoretic approaches to study the behavior of decentralized organizational structures are also pertinent. Harker and Hong [1993] embed their analytic line delay model [1990] into a non-linear program and solve it to minimize the deviation of actual schedules from those requested. In general approach, at least, Harker and Hong's formulation resembles Kraay's, but Harker and Hong model the rail line interactions at a higher level of detail, they return the dual variable values, and do not attempt a network-wide simulation.

Brannlund, Lindberg, Nilsson and Nou [1994] offer a different approach to costing track time based on Lagrangian relaxation. This approach treats train schedules as deterministic: it does not handle the natural variability in daily operations which is implicit in Harker and Hong's model. Also, it assumes an absolute priority of passenger trains over freight, which might be sufficient for the Swedish corridor studies, but is inadequate to handle complexities of multi-priority networks such as coal, intermodal, automotive and general manifest, as well as passenger, all operating on the same tracks with schedule-adherance dispatching in place.

While not a panacea for improving service reliability, it is clear that improvements in train dispatching could produce a marked improvement in transit times and reliability. In many cases improvements could be readily attained for minimal cost, not requiring investment in satellite or transponder-based ATCS communications technology. Accelerated implementation of improved meet/pass planning software is clearly an economic investment for the rail industry to make.

2.6 Rail Terminal Research

Research into rail terminal operations has generally followed one of three approaches:

- **Queueing and Simulation models**

Probably the "granddaddy" of all yard simulation models is the Canadian National's TRIM model (Engelberg and Yager [1982]). This model is extremely labor intensive, essentially a computer-assisted manual simulation. In spite of the effort required to use TRIM, the analysis has proven very useful so that the model is not only still in use today, but has also been adopted by other carriers, particularly CSXT [1994].

Queueing model approaches include the paper by Turnquist and Daskin [1982] based on earlier work by Petersen [1977a][1977b] on bulk processing queues. A primary conclusion of this work was that irregular departures of outbound trains increase both the mean and the variance of connection delay. The writers cautioned that because these queuing models are based on a number of simplifying assumptions, they should be viewed primarily as screening tools. Crainic and Gendreau [1985] present a simple formula for estimating yard delay if the batch size is not too small:

$$w_q^h = \frac{1}{2\mu}\left(1 + \frac{\rho}{K(1-\rho)}\right)$$

where:

w_q^h = Expected Yard Delay

λ = arrival rate

$1 / \mu$ = average interservice time interval

K = batch size

ρ = utilization coefficent; $\rho = \lambda / K\mu$

This formula links yard delay with service frequency and average train capacity utilization. It has been embedded inside Crainic's [1986] network model for service network design.

Bulk-queueing models can handle demand variability to predict *mean* processing times; but they have difficulty predicting the *variance* of yard processing times, which is vital to determining the frequency of missed connections and overall origin-destination trip time reliability. From Powell [1986b], "unfortunately, it is not possible to calculate any of the higher moments of the waiting time distribution except for certain special cases with simple Poisson arrivals and bulk service with no control."

- **Process P-Make or "Achievability"**

This is an offshoot of earlier service reliability efforts at M.I.T. The approach (see Duffy [1994], Schlenker [1994] and Dong [1994]) calls for an application of industrial engineering techniques to estimate processing times for each task to be accomplished within a yard. Then, based on these processing time estimates along with standard deviation, modeling tools are applied to evaluate the "performance" of the plan, the degree to which

the plan, if executed, accomplishes goals such as making connections, and the probability that the plan can be accomplished, its "achievability."

- **Optimization Approaches and Real-Time Control**

Allen and Rennicke [1977] propose a system which assumes fixed track to block assignments and uses a combinatorial search algorithm to optimize the hump sequence.

The objective function is to maximize the total number of cars making their first available outbound connection.

Yagar and Saccomanno [1983] propose a two-step approach to optimizing the hump sequence, also assuming fixed track to block assignments. Their objective is to minimize the average length of time cars spend in the yard as well as to minimize rehump cars. All available trains are prescreened, then the sequence of the surviving candidate trains is optimized using dynamic programming. Their assumptions might be overly restrictive since they assume that the yardmaster cannot "swing" a block to a new track; instead, all cars humped between block "lockout time" and the time the track is reopened are assumed to be rehumped.

SRI International [1983] (see also Wong, Elliott and Hathorne [1979]) proposed a system which was field tested at SP's West Colton yard in 1984. This design uses a rule-based heuristic to suggest dynamic track to block assignments. However, it takes the hump sequence as given. As a sequential heuristic it has no formal objective function, however, the stated design goal is to "maximize the use of classification tracks and to minimize trim engine effort." It also employs a fairly sophisticated process for scheduling pullout leads. However, SP chose not to proceed with the implementation of this system, citing a lack of sufficient benefits (Dingle [1984]).

Quanshou and Yuanjin [1992] developed a system whose primary function is to determine, in the case of a very large terminal having two yards, one for each direction "up" and "down", which yard should process the train. This is based on an analysis of the cars on each train and their outbound connections, whether there is still time to make those connections, and whether the scheduled outbound trains have already reached capacity.

Kraft and Spielberg [1993] proposed to simultaneously optimize both the hump sequence and dynamic block to track assignments using a network-based, mixed integer

formulation. However, this formulation was only tested using a “toy” problem of 3 trains, 4 time periods, 3 blocks and 2 tracks, not practical for any real applications. Kraft has since developed a prototype system for dynamic block to track assignment for the Union Pacific Railroad [1995].

2.7 A Taxonomy of Approaches

Figure 2.2 categorizes research as strategic, tactical or operational on the vertical axis, and by application area on the horizontal axis. Cutoff times among the categories are somewhat arbitrary, but applications near the bottom of the chart are more “real time” than the ones higher up.

The column headings of Figure 2.2 summarize the research described in each section of this chapter. The headings focus on the management processes which support day-to-day operations of the railroad, as opposed to very long term strategic or financial planning. The first heading, “Network Operations” is based on operating plan development literature described in Section 2.3. Then equipment distribution, dispatching and yard operations columns correspond to Sections 2.4 through 2.6 of this chapter.

Real time processes generally depend on processes shown at a higher level in the same column. In train dispatching, for example, track time pricing (Harker and Hong [1993]) should resolve conflicts in the master schedule and provide guidelines for resolution of operational conflicts at lower stages. Then within 8-12 hours of train operation, target arrival times are selected based on the schedules actually planned to operate that day. Finally the real time train dispatching algorithm determines meet/pass locations to maximize schedule adherance to the requested target times. The concept of minimizing tardiness relative to schedule, applied in this dissertation to determine routings of individual freight cars, was directly adapted from the train dispatching literature (Jovanovic and Harker [1990]).

Figure 2.2: Research Taxonomy by Function and Time

	Network Operations	Equipment Distribution	Road Operations	Yard Operations
Strategic > 1 Month	AADT [1994], Strasser [1993], Kwon [1994]			
Tactical				
2 Weeks	Op Plan Dev: See (1) Below		Harker and Hong [1993],Brannlund, Lindberg, Nilsson and Nou [1994]	Crainic and Gendreau [1985], Duffy [1994], Schlenker [1994], Dong [1994]
1Week				
3 Days	Huntley, Brown, Sappington and Markowicz [1993]			
Operational		Equip Dist See (2) Below		
24 Hrs				
8 Hrs	Powell [1986c]		Hallowell [1993], Kraay and Harker [1994]	
3 Hrs				Allen and Rennicke [1977], Yagar and Saccomanno [1983] , SRI Int'l [1983], Quanshou and Yuanjin [1992], Kraft and Spielberg [1993]
1 Hour	Car Scheduling Systems: Kwon [1994], Baugher [1993], Gorman [1994], MoPac RR [1977]	North American Van Lines (Powell, Sheffi, Nickerson, Butterbaugh, and Atherton [1988]	Sauder and Westerman [1983], Kraft [1987], Jovanovic and Harker [1990]	
15 Min				

(1) Operating Plan Development Literature by: Morlok and Peterson [1970], Bodin [1980], Assad [1980b], Crainic [1984] [1986], Haghani [1989], Keaton [1989] , Gorman [1995], ABM Kornhauser and Mayewski [1983], Van Dyke [1986] , SPM McCarren and Martland [1980]

(2) Equipment Distribution Literature by: Philip and Sussman [1977],Mendiratta and Turnquist [1981] , Jordan and Turnquist [1983], Powell [1986a] ,Powell [1988],Turnquist and Markowicz [1989], Nozick [1992]

A well-formulated master schedule helps the real time control processes perform better. Conversely, a poorly formulated master schedule passes many conflicts and infeasibilities through to be resolved at the lower levels. This harms service reliability.

Why not, then, focus research first at the top levels and work down? The reason is that this approach tends to entrench existing business processes, that is, to "pave the cowpaths" (Harker [1995]). By focusing first on the real time control area, tactical and strategic planning tools can be developed later to fully exploit the capabilities of the real time control process. For example, Jovanovic and Harker's [1990] early work in train dispatching was in real time control; in later years, the research focus moved up towards the tactical and strategic levels.

In the "Network Operations" area, development of dynamic car routing in real time will obsolete almost the entire body of existing literature on railway operating plan development: because nearly all that literature assumes a fixed, tag-table based shipment routing strategy. As well, most operating plan development models are deterministic, based on "average day" volumes, often not even recognizing day-of-week differences. If the results of this dissertation are ever implemented in real world rail systems, almost all the current research on railway operating plan development will have to be revisited.

The only known published terminal operations model compatible with dynamic car routing is the work by Kraft and Spielberg [1993]. This model was further developed on a proprietary basis by Kraft for the Union Pacific Railroad [1995]. More research will also be required in terminal operations so that rail yards will be better able to get the right cars on the right trains.

2.8 Positioning of This Research in the Taxonomy

This research fits into the lower left-hand corner of Figure 2.2, in the "operational, network operations" category. It is closely related to previous work by Kwon[1994], Baugher [1993], Gorman [1994], Campbell [1996] and the Missouri Pacific Railroad [1977]. Earlier research by White and Wrathall [1970] fits here as well, along with closely-related LTL trucking problems solved by Powell [1986c], Powell and Sheffi [1989], Braklow, Graham, Hassler, Peck and Powell [1992], Farvolden, Powell and Lustig [1993] and Barnhart and Sheffi [1993].

The Missouri Pacific [1977] citation describes functional requirements of the rail industry's current table-driven car scheduling process, but it does not incorporate optimization techniques. White and Wrathall's [1970] early paper proposed an optimization approach, but their problem formulation was not practical for large scale problems using the computer hardware and software available at the time. Baugher [1993] describes implementation of a shortest path algorithm within the Norfolk Southern's ITMS car scheduling system, but the solution is based on a "flat" blocking network; it does not incorporate train schedules or train capacity constraints. Gorman's [1994] presentation describes the Santa Fe Railway's current car scheduling process, which again is table-driven, not an optimization-based system.

Kwon's [1994] dissertation, along with several LTL shipment routing models listed above, are most closely related to this research. The key improvements developed here are, first, an integral flow guarantee for the deterministic part of the problem; second, modeling customer behavior in the reservations and booking process using the "acceptance function" as defined in Section 3.4; and the adoption of a profit-maximizing, yield management based model formulation, also described in Section 3.4. Another distinction of this research is the use of specialized subgradient and dual-ascent algorithms, specifically intended to cope

with extremely large sized problems; whereas the previous research has generally tended to rely on linear-programming based pivoting algorithms. Finally, although Kwon [1994] developed both a simulation model and an optimization-based traffic assignment model, these were not integrated together in his dissertation. In this research, the optimization code is fully embedded inside the simulation model, which is used to measure process performance on a rolling horizon basis.

As compared with the LTL trucking research, it is interesting to note a primary focus of that research seems to be on the identification of opportunities for adding direct trucks to bypass breakbulk terminals. In the LTL systems, this is done in a real time mode. In railroads, it is difficult to make last-minute changes in shipment routing, particularly after cars have already been switched into classification tracks, so this service design function has traditionally been performed as part of the tactical planning process. Although extra trains are sometimes operated, railroad shipment routing tables tend to remain fairly static in the short term. Although some rail carriers might be considering the possibility of adopting this type of dynamic operation, that is not the primary focus of this research. Instead, the focus here is simply on determining the optimal routing for a set of shipments assuming a fixed capacity network of blocks and trains, and on determining what service commitments should be offered to customers based on this same set of operating and capacity constraints.

Two new models and solution algorithms for managing a freight railroad reservation-based business process in real time will be developed here. The first model determines what service offers ought to be made on loads calling in, supporting the railroad's reservation center. It is understood that an offer for delayed service incurs some probability that the customer will reject the offer — this is factored into the formulation

Once a shipment has been accepted, it must be managed consistently with its delivery commitment; the second model, a dynamic car routing program, supports the needs of classification yards by suggesting blocks and trains in which each car should ride.

CHAPTER 3

A Model for Dynamic Car Scheduling: Problem Formulation

3.1 Problem Definition

Two related problems are considered here: first, what delivery appointment time, if any, should be offered to maximize profit when a customer calls in a load; second, how should a carrier manage cars already on line so they will arrive by their appointment times? The problem is to determine which routings cars should take through a space/time network to *maximize profit or minimize cost*. Essentially, it is a traffic assignment problem (see Fukushima [1984], Tobin and Friesz [1988], Aashtiani and Magnanti [1981]).

Clearly the two problems are related since if too much traffic is accepted or service offers are too aggressive, the dynamic car routing algorithm will not be able to achieve the target delivery times. To prevent this, the two problems must be linked through joint capacity constraints and/or equilibrium dual prices. These two problems are sufficiently distinct in required inputs, outputs and solution time frame to justify different solution approaches, even though their mathematical structure is very similar.

Consider a centralized railway reservations center operating with a sophisticated, real time shipment management system. The function of the center is to proactively manage and monitor all shipments currently on line in a manner consistent with their delivery commitments, as well as to offer price and service quotations on new shipments called in. The objective of the reservation center is to maximize the railroad's total profit on cars currently moving, plus expected profit from future business:

$$\text{Max Profit} = \left((\text{Revenues}_{\text{Det}} - \text{Costs}_{\text{Det}}) + E\ (\text{Revenues}_{\text{Fut}} - \text{Costs}_{\text{Fut}}) \right)$$

$\text{Revenues}_{\text{Det}}$ represents the revenues which will be earned from the cars currently on-line if they are all delivered on time. Since these cars are physically in the custody of the rail carrier, and the price quotation has already been accepted, revenue is a fixed number and can be treated as a constant. $\text{Costs}_{\text{Det}}$ includes all operating cost associated with moving cars currently on line. If revenue would be reduced as a result of cars being delivered late, this can be treated as a penalty and included in $\text{Costs}_{\text{Det}}$. Thus the first part of the objective function reduces to minimization of operating plus penalty costs of all cars currently on line. There is no guarantee that any given shipment will remain profitable, and shipments can no longer be rejected once they start moving. This portion of the objective function will be handled by a "Dynamic Car Scheduling" model.

The second part of the objective function maximizes expected profit from future, as yet uncertain demands. Whether the demand materializes is not within the carrier's control. However, once a prospective load is called in, then the probability of that load being "booked" depends on the service offer made. As described in Section 3.4, this service offer decision leads to both a direct revenue and cost impact, therefore both the revenue and cost terms must remain within the objective function. This will be the focus of the "Train Segment Pricing" model.

Because the formulation has been decomposed based on time, these two models must be coordinated through joint dual variables and/or capacity constraints. The best way to accomplish this will be further examined in the rolling horizon simulation testing phase.

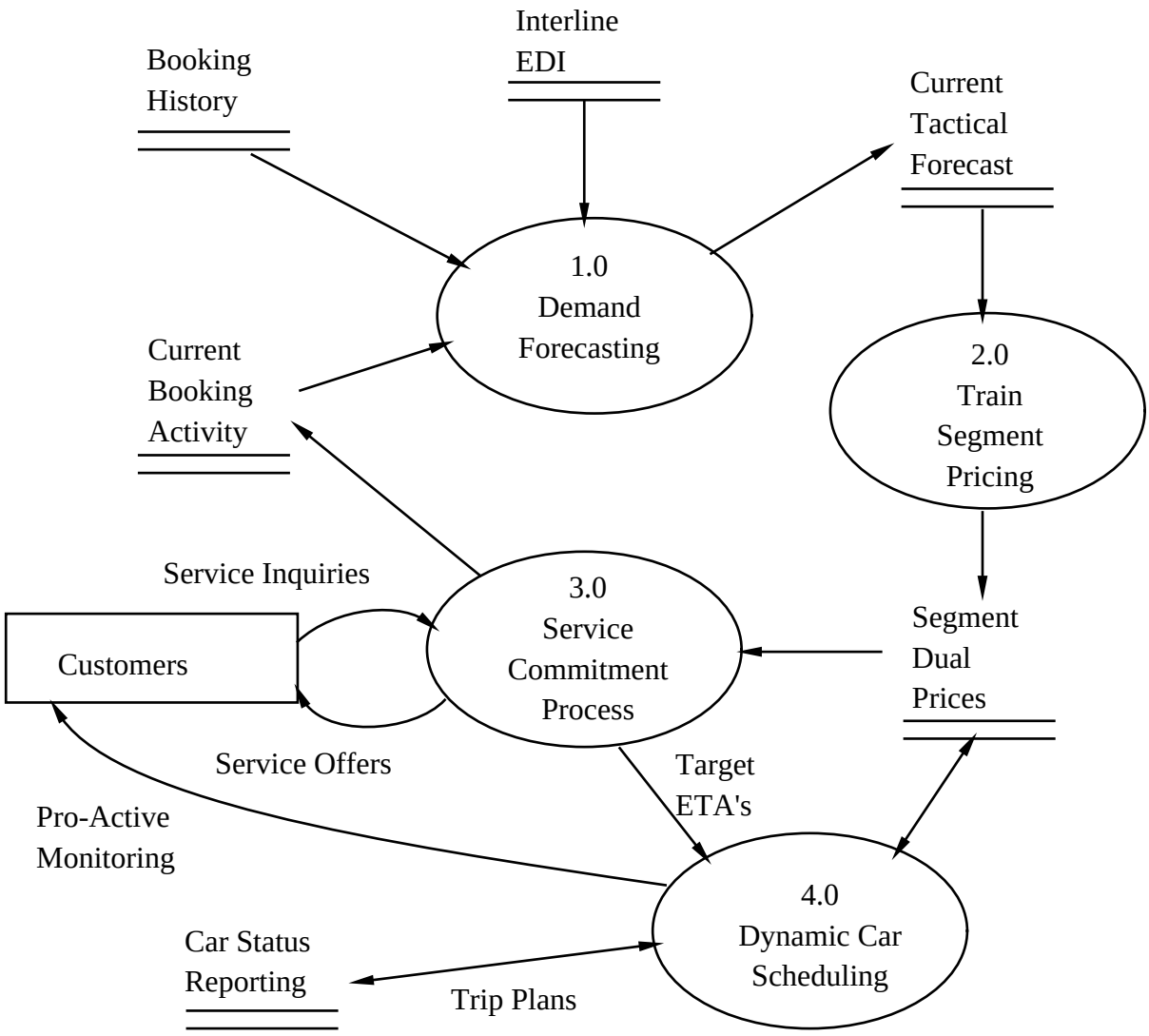
- The "Train Segment Pricing" model accepts a forecast of future traffic flows which have not materialized yet. Its focus is to *estimate dual prices for each train segment* . These prices are used to determine appropriate service commitments when the actual loads call in. A primal feasible solution is calculated only to validate the quality of the dual prices by establishing that these prices lead to a tight upper bound. This feasible solution is not really needed for any other purpose.
- In contrast, the "Dynamic Car Routing" model must route *real* shipments in real time, producing trip plans and instructions for classification of cars in yards. A target delivery time for each shipment is determined when each order is taken, so this model's purpose is to develop an operating plan to *deliver all shipments by the agreed-upon time at minimum cost* . A feasible solution, satisfying an integral flow requirement is needed because the application is routing physical shipments which cannot be split in half. Mathematically, the "Dynamic Car Routing" formulation is a special case of the "Train Segment Pricing" model.

Figure 3.1 shows how the modules resulting from this decomposed formulation fit back together. The process starts with a demand forecast (1.0), which must be periodically adjusted as demand materializes throughout the day. The demand forecast methodology is not the focus of this dissertation and is left to future researchers.

Given an updated demand forecast, the train segment pricing model (2.0) updates dual prices for each train segment. These dual prices directly influence service offers developed by the real time service quotation module. This quotation process should produce an overall aggressive but achievable set of service offers. Service quotation should be

a good predictor of the actual capabilities of the dynamic car scheduling process. Solution of the train segment pricing model can be time consuming, but fortunately it need not be solved in real time.

Figure 3.1: Proposed Reservations/Booking and Dynamic Car Scheduling Process



Based on train segment dual prices, the service offer is determined in (3.0) using the modified shortest path procedure described in Section 4.2. As loads are called in, the carrier evaluates each one in terms of revenue, profitability, impact on overcapacity train segments and service sensitivity of the customer. This can be done very quickly so that a service quotation can be developed in just a few seconds. Each accepted load is assigned a target delivery time and a tardiness penalty cost.

Finally, the dynamic car scheduling module (4.0) produces a trip plan for each car and, more importantly, shipment routing instructions to directly drive car classification in each yard. This process continuously monitors all movement reportings to ensure that all shipments remain on schedule.

3.2 Formulation Issues

A comparison can be made with the previous work of Jovanovic and Harker [1990] and Hallowell [1993] in the train dispatching area. Jovanovic and Harker [1990] developed a real time dispatching algorithm to minimize tardiness of trains relative to schedule. In theory, a master schedule could supply the target times for Jovanovic's algorithm, but problems with this approach soon became apparent. For example, the master schedule did not take into account the operation of extra trains. If a train fell behind schedule, master schedule-based target times could become unachievable.

These problems suggested the need for a method of developing target times which takes real-time traffic conditions into account, which was developed by Hallowell [1993]. Hallowell showed that this real time method of developing aggressive but achievable target times makes the dispatching algorithm "work harder," leading to a tighter compliance with plan and improving performance.

A master-schedule driven approach in the car scheduling application would likely suffer similar difficulties. Under light traffic conditions, master schedule-based quotations

should be “tightened up” to take advantage of available capacity in the system. Otherwise, some customers whose needs could actually be met would be quoted unnecessarily long transit times, resulting in a higher customer “balking” rate than necessary. In heavy traffic conditions, master schedule-based commitments might not be achievable, leading to service failures. For these reasons, a real time service quotation capability will be developed, and a master-scheduling based service quotation approach will not be further pursued here.

Past researchers have approached this as a deterministic multicommodity network flow problem (MCNF) with side constraints on train capacities. Each railcar or individual shipment is modeled as a separate commodity. This approach was first suggested by White and Wrathall [1970]. A practical problem is the large problem size with literally thousands of commodities. Specialized algorithms are required to make such a formulation practical, such as suggested by Kwon [1994], Barnhart and Sheffi [1993] or Farvolden, Powell and Lustig [1993] and Jones, Lustig, Farvolden and Powell [1993].

It is interesting to note that White and Wrathall’s [1970] mathematical programming formulation even predates the Missouri Pacific’s [1977] car scheduling implementation, which uses a traditional data processing approach. Thus, the basic mathematical structure of this problem has been known for a long time, although the computer hardware and software available in 1970 prevented any practical application of this concept. Another two decades would pass before researchers would again attempt to apply mathematical programming approaches to this shipment routing problem.

A simplistic way of addressing the overcapacity train issue would be to rank cars in order of priority and load the train network sequentially. When a train fills up, that link is “blocked” and the next best path is used for subsequent cars. This approach was taken in a system built by American Airlines for the Santa Fe [1994], and its performance will be further examined in Chapter 4.

The telecommunications industry has been dynamically routing phone calls for many years. An analogy could be drawn to packet-switching systems where the packets could be compared to freight cars and the telephone lines to trains. There is a rich literature on this subject (see, for example, Hajek and Ogier [1984], Moss and Segall [1977,1978]) but there are also many fundamental differences between telecommunications systems and rail transportation systems. Hajek and Ogier [1984] model traffic as a continuous flow, which assumption is valid only when the number of packets in the network is large compared to the number of nodes. Their objective is to minimize the delay of messages waiting for transmission capacity; however, transmission itself seems to have zero cost and require zero time. They describe the problem as an "optimal evacuation" problem and the solution is essentially a network max-flow problem rather than cost minimization or profit maximization.

The railroad car scheduling problem must take into account:

- The discrete nature of railcar shipments, a continuous flow model cannot apply;
- The fact that each shipment can have a different service commitment, revenue and cost;
- Line haul movement occurs in batch and at a specific time, rather than on a continuous basis, which takes time and has a cost;
- Yard handling incurs a fixed cost plus a variable cost which depends on how long a shipment remains in the yard.

These factors all point back towards the traditional MCNF formulation. While considerable information about each shipment currently moving on line is known, unfortunately there exists a great deal of uncertainty in future demand forecasts. In a similar application, Powell [1987] modeled freight transportation demand as a Poisson process.

The nature of the problem changes significantly when the customer calls in their loads for the day. Prior to this event, the carrier can only project future demand based on some probability distribution. After the customer calls in, however, this distribution function can be replaced by the realization of actual demand. These “before” and “after” problems are distinctly different. This suggests that the railroad shipment routing problem might be decomposed into two subproblems, as follows:

• A dynamic car scheduling, cost-minimizing model applied only to “initial inventory” of shipments currently on line. The problem is to deliver all shipments within their customer committed delivery times at minimum cost. The current position and destination of each car are known. Each car has an agreed-upon delivery time window, with a penalty cost for early or late arrival. Penalty costs represent actual payments or rate reductions, an estimate of loss of goodwill, loss of future revenues or profits, or some combination of these. Flows must be routed subject to an integral flow constraint since boxcars cannot be split in half.

Required inputs of the dynamic car scheduling model include, for every shipment currently on-line:

- Number of cars
- Current location
- Destination
- Time and mileage cost
- Scheduled delivery time
- Tardiness penalty cost

The model produces:

- Trip plans for every car
- Train segment and yard workload projections
- Car classification instructions for every yard
- Dual variable prices for every train segment

• A train segment pricing, profit-maximizing model which assigns forecast demands for 7-10 days into the future. The problem is to determine what service offers to make to maximize expected profit, taking cost, revenue, the probability the demand will materialize, and the probability the customer will accept the offer into account. If the customer accepts the service offer, this appointment time becomes the base from which penalties for early or late deliveries are calculated once the shipment starts moving.

Required inputs include, for every forecast demand:

- Number of cars
- Origin
- Destination
- Time and mileage cost
- Revenue
- Probability the shipment will materialize
- Customer service sensitivity parameter

The model produces dual price estimates for every train segment. As will be further described in Chapter 4, based on these dual prices, the optimal service quotation can easily be determined using a modified shortest path algorithm. Train loading and yard workload for the next 7-10 days is also projected.

Profit maximization is not the only possible objective. Another possibility would be to directly minimize weighted tardiness, as proposed by Jovanovic and Harker [1990], or to minimize the maximum tardiness as proposed by Luss and Smith [1986]. The literature on some of these alternative approaches is well developed: Luss [1987] extended their minimax methodology to the case of non-linear cost functions, and Klein, Luss and Smith [1992] extended it to multiperiod problems.

Each of these possible objective functions offers certain advantages in terms of solution methodology and execution speed. For the case of minimum weighted tardiness, for example, one could route shipments using some myopic heuristic, and if it turned out that all shipments were delivered on time, that would be considered an optimal solution. A profit-maximizing requirement would further be to identify the *least cost* way to deliver all the shipments on time: clearly a more difficult challenge.

Since this model might recommend rejecting shipments, it must prove the unprofitability of offered traffic to justify turning it away. This requires an economics-based, profit maximizing formulation.

3.3 Formulation: Dynamic Car Routing Model

The Dynamic Car Routing model formulation is just a standard Multi Commodity Network Flow (MCNF) problem with coupling constraints on train capacities. It is very similar to the model proposed by White and Wrathall [1970] except formulated in link rather than path variables. Each discrete shipment, which can consist of one or more cars, is represented as a separate commodity. Since the application is for routing of real freight cars, a special integral flow constraint must be included.

Define the following sets:

N = The set of nodes representing points in space and time when trains arrive or depart yards. In the mathematical formulation, a single fictitious "Super Sink" node "Ω" serves as the common destination of all shipments. This allows the model to choose from several possible delivery times in the space/time network, while still satisfying network conservation of flow constraints for all nodes. The subset of "termination" nodes, each node corresponding to a different delivery time, from which shipment "k" can be delivered to the consignee is defined by $T_k \subset N$.

A = The set of links representing train schedules in space and time, and "inventory" holding links in yards, including those links connecting the termination nodes $T_k \subset N$ for each commodity $k \in K$ to the fictitious Super Sink "Ω". Each link $(i,j) \in A$ for commodity $k \in K$ has a flow volume x_{ij}^k and cost c_{ij}^k.

K = The set of demands: each forecast shipment $k \in K$ has an integral number of cars actually shipped, Θ_k, and a target delivery time. Deliveries at other times may incur a penalty, which is built into the link cost coefficients c_{ij}^k on the links connecting into the Super Sink node Ω.

Then G={N,A} defines an acyclic graph representing a network of train and yard processing links in space and time. This network is replicated for each commodity $k \in K$.

S = The set of train route segments, based on intermediate pickup or setoff locations as shown in Figure 3.2. Each link $(i,j) \in A$ has an associated set of train segments $S_{ij} \subset S$. Each segment $s \in S$ has an associated set of links $A_s \subset A$ which defines the inverse relationship.

Define the following input data:

C_s = Capacity of train segment $s \in S$, in cars.

Θ_k = An integral number of cars actually shipped for commodity $k \in K$.

c_{ij}^k = Cost for commodity $k \in K$ moving over link $(i,j) \in A$, including fictitious links connecting to the supersink Ω which may contain penalty cost terms for late shipment deliveries.

Define the following variables:

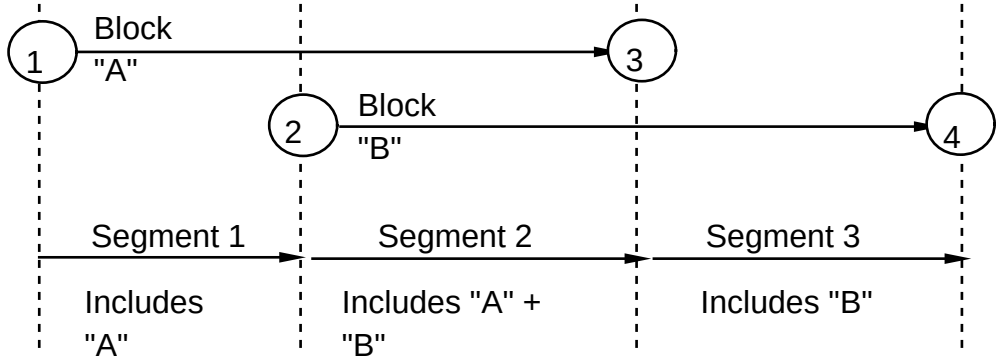
x_{ij}^k = Number of cars of commodity $k \in K$ moving over link $(i,j) \in A$

u_s = Dual variable associated with segment $s \in S$.

φ_{ij}^k = Adjusted cost of car $k \in K$ moving over link $(i,j) \in A$.

$$\varphi_{ij}^k = c_{ij}^k - \sum_{s \in S_{ij}} u_s \quad \text{for all links } (i,j) \in A.$$

Fig. 3.2: Train Route Segment Definition



Since the c_{ij}^k cost coefficients and u_s dual variables always have opposite signs, subtracting the two in the above definition of φ_{ij}^k makes them additive, in fact, by absolute value. For example, Figure 3.2 shows a train which handles two blocks of cars: Block

"A", picked up at node 1 and set off at node 3; and Block "B", picked up at node 2 and set off at node 4. This train's route would be divided into three segments; segment break points occur where pick up or set off activity occurs. The cost of the link representing block "A" would be adjusted by subtracting the dual price of train route segments 1 and 2. Thus the adjusted cost for link (1,3) is $\varphi_{13}^k = c_{13}^k - u_1 - u_2$.

For all commodities $k \in K$, the following identity holds by definition for the subsets S_{ij} and A_s. This identity will be shown using a simple example derived from Figure 3.2:

$$\sum_{(i,j)\in A} \sum_{s\in S_{ij}} u_s \, x_{ij}^k = \sum_{s\in S} \sum_{(i,j)\in A_s} u_s \, x_{ij}^k \qquad (3.3.1)$$

- "Left Hand Side" first indexing by link, then by associated segment:

  $A = \{ (1,3), (2,4) \}$ (All links in the network)

  $S_{13} = \{ 1, 2 \}$; $S_{24} = \{ 2, 3 \}$ (Segments associated with each link by index #)

- "Right Hand Side" first indexing by segment, then by associated link:

  $S = \{ 1, 2, 3 \}$ (All segments by index #)

  $A_1 = \{ (1,3) \}$; $A_2 = \{ (1,3), (2,4) \}$; (Links associated with each segment)

  $A_3 = \{ (2,4) \}$

  Both yield the same result: $\sum_{s\in S} \sum_{(i,j)\in A_s} u_s \, x_{ij}^k = u_1x_{13}^k + u_2x_{13}^k + u_2x_{24}^k + u_3x_{24}^k$

Then the dynamic car scheduling problem can be formulated as a standard multicommodity network flow problem (MCNF) with an added integral flow constraint.

Only cars already received are assigned trip plans using this model. The dynamic car scheduling problem can be written as:

$$\text{Min} \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \quad \text{for all } k \in K, (i,j) \in A. \qquad (3.3.2)$$

subject to:

$$\sum_{n \in T_k} x_{n\Omega}^k = \Theta_k \quad \text{for all } k \in K, (n,\Omega) \in A, \qquad (3.3.3)$$

(Ω = the "Supersink" node)

$$\sum_i x_{iQ}^k - \sum_j x_{Qj}^k = 0 \qquad (3.3.4)$$

for all $k \in K$, $(i,Q) \in A$, $(Q,j) \in A$,

($Q \neq$ the shipment origin or supersink "Ω")

Equations 3.3.3 and 3.3.4 are standard network flow conservation constraints for the supersink and intermediate nodes, respectively. The origin node flow conservation constraint is redundant and has been omitted.

$$\sum_{k \in K} \sum_{(i,j) \in A_S} x_{ij}^k \leq C_S \quad \text{for all train segments } s \in S \qquad (3.3.5)$$

(Capacity Coupling constraint)

$$x_{ij}^k \in \{\text{Nonnegative Integers}\} \qquad (3.3.6)$$

for all $k \in K$, $(i,j) \in A$ (Integrality Constraint)

In this cost minimizing formulation, the c_{ij}^k cost coefficients will be positive and u_S dual variables associated with (3.3.5) will be negative.

3.4 Formulation: Train Segment Pricing Model

The proposed train segment pricing model is a generalization of the dynamic car scheduling formulation. At a centralized reservations or booking office, forecast demand is not certain to materialize; even if a load *does* call in, the customer is not certain to accept the carrier's price and service offer. The lower the price or the faster the service, the more likely the customer will accept the carrier's offer. The reservations center needs a tool to help them determine what service offer to make, taking this tradeoff into account.

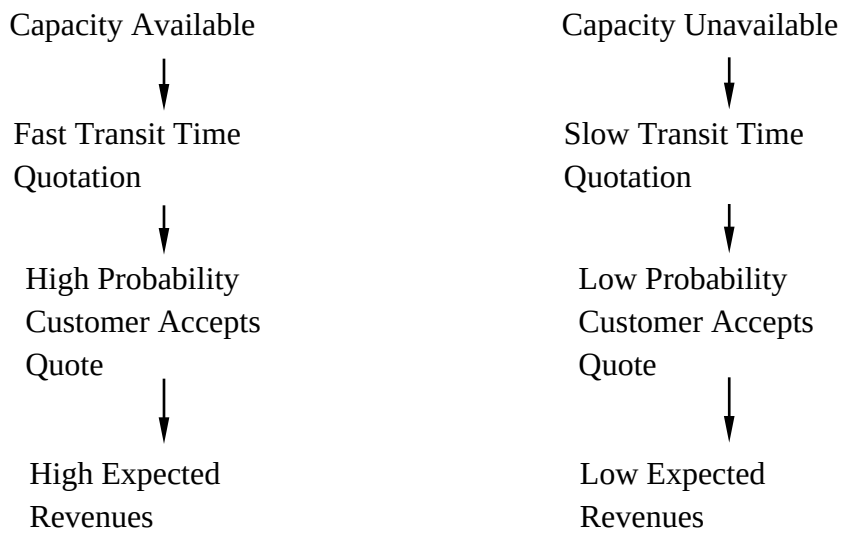
In theory, rail carriers should provide capacity so they can sell space on trains to generate revenue and make a profit. In the past, however, decisions whether or not to provide train capacity have often been based on cost considerations. This framework provides, for the first time, an ability to understand revenue as well as cost implications of a decision to provide capacity, and the ability to incorporate that information into real time decision making.

Figure 3.3 shows how the carrier's decision to provide capacity leads to both a direct revenue and cost impact. If train capacity is available, the shipment should move on the first available train via the lowest cost routing, taking into account the time value of the freight and hourly cost of the equipment it moves in. This policy minimizes the railroad's cost, and maximizes expected revenues by maximizing the probability the customer will accept the offer, so it maximizes expected profit. If capacity is not available, a later delivery must be offered. Then the price might need to be reduced and costs will likely be higher, reflecting the longer transit time. Or, the customer might reject the offer entirely.

In this application, cost coefficients apply to each train block, and are input data based on both the physical characteristics of the territory traversed and on transit time. Typically each block carries an "allowable traffic" attribute. Thus, an intermodal block

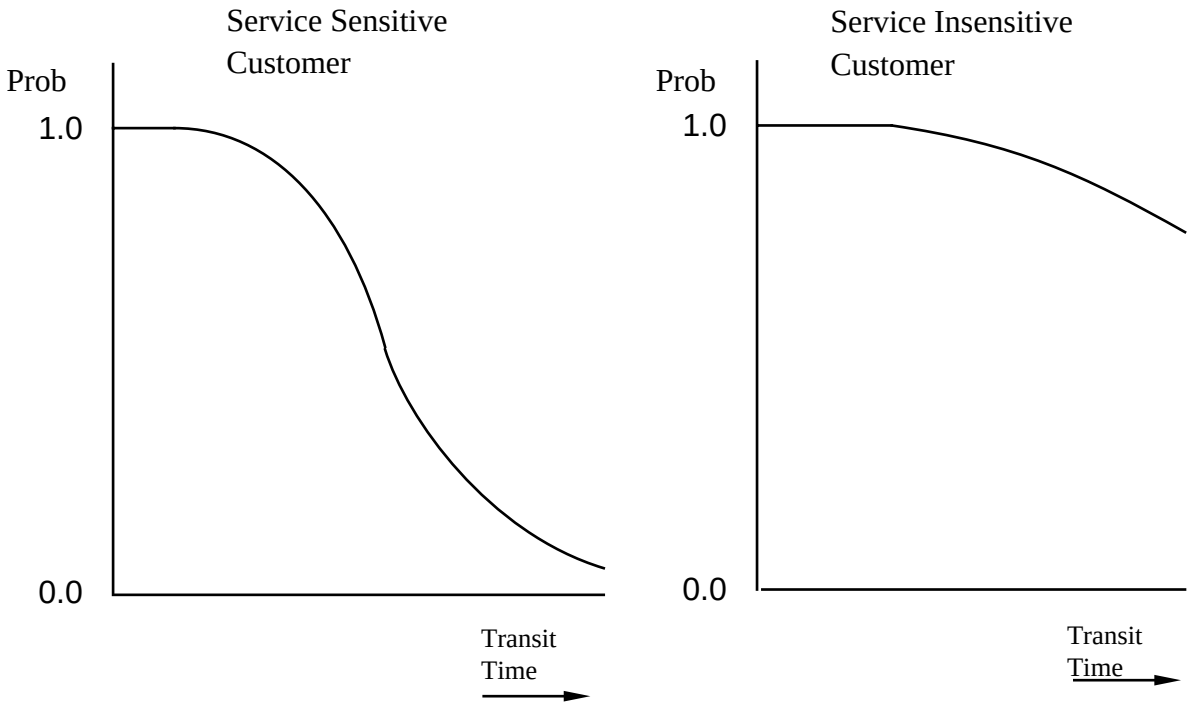
would be restricted to carry only intermodal traffic and would not even be available for consideration for routing general merchandise traffic. Choosing only from allowable paths, usually the least expensive path will also turn out to be the fastest.

Figure 3.3: The Revenue Value of Capacity



The proposed "acceptance function," shown in Figure 3.4, quantifies the probability the customer will accept a service offer as a function of offered transit time. This function must be calibrated for each customer and OD flow, based on mutually agreed contract terms and historical customer behavior. Underlying the acceptance function is an assumption that the longer the transit time quotation, the less likely the customer will accept the carrier's service offer. Especially for highly profitable traffic, this provides a clear incentive for the carrier to offer the most rapid service possible, to maximize the likelihood the customer will accept the service offer.

Figure 3.4: "Acceptance Function" gives the probability a certain Service Offer will be accepted.



To implement the acceptance function on a computer, a specific functional form must be assumed. The logit function has extensive theoretical basis for application in modeling economic utility, and is often used in consumer choice modeling. See, for example, Manrai [1995], Koning and Ridder [1994], Borsch-Supan [1990], Green and Krieger [1988], and Anderson, DePalma and Thisse [1988]. The logit function has a nice "S" shape and is easy to calibrate:

$$P(\text{ Accepts Quote}) = \frac{e^{-}}{1 + e^{-}}$$

where:

α, β = Calibration parameters, described below

Δ = The difference between the transit time quoted and the "base" transit time per the railway operating plan using the *most efficient* routing. "Base" transit time is calculated using a minimum cost shortest path calculation with all $u_s = 0$.

The definition of Δ proposed above assumes the customer knows the "base line" or best case service level and would not have called had that service offer been unacceptable. The probability of customer acceptance for this "base line" service offer equals 1.00 by definition. Recapture probabilities for later delivery offers would be calculated using the logit function.

To better understand the significance of the logit coefficients, note that if $\alpha = \beta$ and $\Delta = 1$ day, then P(Accepts Quote) = .50. The larger α and β, the steeper the slope of the acceptance function. To further simplify the model, define:

ρ = The number of days beyond the base transit time which corresponds to the 50% acceptance level.

Then set $\alpha = 4.394$ and $\beta = \alpha / \rho$. With these substitutions:

$$\text{P(Accepts Quote)} = \frac{e^{4.394 - 4.394 \ /}}{1 + e^{4.394 - 4.394 \ /}}$$

The ρ substitution makes the function easier to calibrate, reducing the data requirements of this formulation. It requires only an empirical estimate of what service

offer would result in a 50% customer acceptance probability. This question could be asked of the sales representative for new business or calibrated based on historical booking behavior for old business. Sample values are given in Table 3.1 for the case $\rho = 2$.

Table 3.1: Example Acceptance Function for $\rho = 2$

Lateness Quoted $= \Delta$	0	1	2	3	4
Probability Accept	1.00	.90	.50	.10	.01

Define:

K = The set of demands, as before; however now each *forecast* shipment $k \in K$ has an expected number of offered cars ξ_k, an acceptance coefficient ρ_k which measures the customer's service sensitivity (used as a parameter of the logit function) and revenue which can depend on the offered delivery time. This revenue is built into the link cost coefficients c_{ij}^k on the links connecting into the Super Sink node "Ω".

ξ_k = Expected offered demand for commodity "k", a forecast origin-destination demand offered at a particular point in time. ξ_k can be viewed simply as an average demand, a parameter of a random distribution such as Poisson. In rolling horizon testing in Chapter 5, the true value of ξ_k is assumed to be known and simulated demands are randomly generated based on an integerized Normal(ξ_k,ξ_k) distribution, approximating the Poisson (used in Kraft [1995]). In a real world application, ξ_k might be estimated based on historical data. If ξ_k were estimated as a sample mean, ξ_k would also be a random variable. However, this formulation treats ξ_k as a known quantity.

B_s = A user-supplied booking limit, which can be greater or less than C_s, the actual capacity of train segment $s \in S$.

P_{nk} = The probability customer of commodity $k \in K$ accepts the service offer for shipment delivery at node $n \in T_k$.

The Train Segment Pricing model can then be written as:

$$\text{Max} \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \qquad (3.4.1)$$

subject to:

$$\sum_{n \in T_k} (1/P_{nk}) x_{n\Omega}^k \leq \xi_k \quad \text{for all } k \in K, (n,\Omega) \in A, \qquad (3.4.2)$$
(Ω = the "Supersink" node)

$$\sum_i x_{iQ}^k - \sum_j x_{Qj}^k = 0 \qquad (3.4.3)$$
for all $k \in K$, $(i,Q) \in A$, $(Q,j) \in A$,
($Q \neq$ the shipment origin or supersink "Ω")

$$\sum_{k \in K} \sum_{(i,j) \in A_S} x_{ij}^k \leq B_S \qquad (3.4.4)$$
for all train segments $s \in S$
(Booking Limit Coupling constraint)

$$x_{ij}^k \geq 0 \qquad (3.4.5)$$
for all i,j,k
(Nonnegativity)

Changes made to the original Dynamic Car Routing formulation include:

- The cost minimizing objective function 3.3.2 has been converted in 3.4.1 to maximize profit instead. This implies a reversal of sign in the c_{ij}^k coefficients. All costs c_{ij}^k are represented as negative and revenues as positive quantities. The c_{ij}^k coefficients on links incident to the super sink are positive and *now contain revenues earned, rather than penalty cost* for late delivery. Dual variables u_S associated with (3.4.4) are nonnegative.

- Equation 3.3.3 becomes 3.4.2. The "gain" coefficients in 3.4.2 represent the customer acceptance probability for an offered delivery at termination node "n". Since acceptance probabilities are present, the revenue is not certain to occur, but if the customer rejects the service offer, the cost is avoided too. The gain coefficient reduces the expected flow volumes x_{ij}^k to proportionally reduce both cost and revenue on every link, so that if all flow is routed over a single path, then $x_{ij}^k = \xi_k * P_{nk}$ on utilized links (which is simply equation 3.4.2 restated). If all $P_{nk} = 1.00$, then the problem reduces to a standard MCNF formulation.
- The equality in 3.3.3 is converted to an inequality in 3.4.2, allowing shipments to be rejected.
- The integral flow constraint 3.3.6 is replaced with a simple nonnegative flow constraint 3.4.5, because the fractional "gain" coefficient representing probabilities in 3.4.2 makes integral flows impossible to achieve. Also, "expected" origin-destination demand ξ_k in the Train Segment Pricing formulation is not required to be integral to begin with.
- A user-adjustable "booking limit" B_s is substituted as the right hand side of constraint 3.4.4, in place of actual train segment capacity C_s, used in 3.3.5.

3.5 Dual Formulation

The dual linear program of the Train Segment Pricing model formulation provides a number of useful insights. The dual optimality conditions lead directly to the formula which is used to select the optimal delivery time in the proposed subgradient solution algorithm. Also, analysis of the dual establishes that the shadow prices on the train segment capacity constraints, u_s, are directly additive to the original link costs c_{ij}. The w_i^k dual variables are the reduced costs associated with the original network flow conservation constraints; their value gives the cumulative distance of each node "i" from the root node of the shortest path tree.

In a profit maximizing formulation, all link costs c_{ij} are input as negative quantities, except for the fictitious arcs connecting the termination nodes T_k with the supersink node "Ω". These fictitious arcs are the only links to have positive c_{ij} , which represents the revenue earned if the shipment is delivered at node $i \in T_k$. All $u_s \geq 0$ because they are associated with a "less than or equal to" inequality constraint. The w_i^k duals are associated with equality constraints; therefore, they are theoretically unrestricted, however, due to the structure of the input data in this problem, all $w_i^k \leq 0$ except at the super sink node Ω. At the super sink node "Ω", w_i^k should not be negative if shipments can be rejected, so in the Train Segment Pricing model formulation, all $w_\Omega^k \geq 0$. In the Dynamic Car Scheduling formulation, shipments which are already moving cannot be rejected, so the value of w_Ω^k is unrestricted. The remainder of this section will work out the dual based on the Train Segment Pricing formulation given in Section 3.4, which is the more general of the two models.

The dual formulation will be derived by working out a simple example first, then the general formulation will be stated. Figure 3.5 depicts a simple two commodity, ten node, twelve link MCNF problem. The TSP simplex tableau for this problem is given in Figure 3.6. From this tableau, the dual constraints (for commodity 1) can be written as:

$$w_2^1 + u_1 \geq c_{12}^1 \qquad (3.5.1)$$

$$w_3^1 \geq c_{13}^1 \qquad (3.5.2)$$

$$-w_3^1 + w_4^1 + u_2 \geq c_{34}^1 \qquad (3.5.3)$$

$$-w_2^1 + w_4^1 \geq c_{24}^1 \qquad (3.5.4)$$

$$-w_2^1 + (1/p_{25}^1)\, w_5^1 \geq c_{25}^1 \qquad (3.5.5)$$

$$-w_4^1 + (1/p_{45}^1)\, w_5^1 \geq c_{45}^1 \qquad (3.5.6)$$

The constraints for commodity 2 would be structured identically to the above.

Fig 3.5: Example Network for Dual Formulation

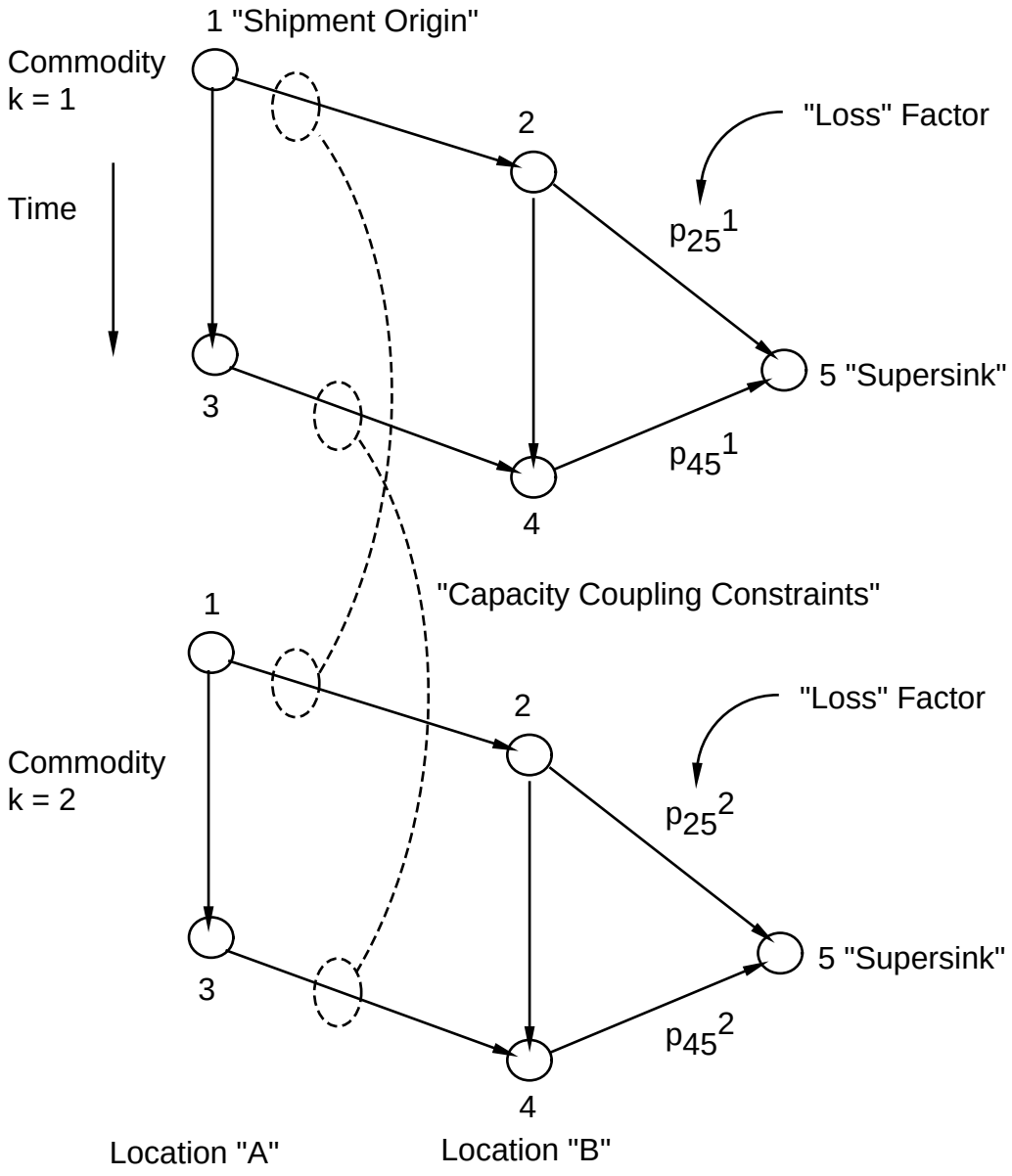


Fig 3.6: Simplex Tableau for Example Network

	Decision Variables												Right Hand Side	Dual Var's
	x_{12}^1	x_{13}^1	x_{34}^1	x_{24}^1	x_{25}^1	x_{45}^1	x_{12}^2	x_{13}^2	x_{34}^2	x_{24}^2	x_{25}^2	x_{45}^2		
Node 2_1	1			-1	-1								0	w_2^1
Node 3_1		1	-1										0	w_3^1
Node 4_1			1	1		-1							0	w_4^1
Node 5_1					$1/p_{25}^1$	$1/p_{45}^1$							1	w_5^1
Node 2_2							1			-1	-1		0	w_2^2 2
Node 3_2								1	-1				0	w_3^2 2
Node 4_2									1	1		-1	0	w_4^2 2
Node 5_2											$1/p_{25}^2$	$1/p_{45}^2$	2	w_5^2 2
Capy 1-2	1						1						C_1	u_1
Capy 3-4			1						1				C_2	u_2
	c_{12}^1	c_{13}^1	c_{34}^1	c_{24}^1	c_{25}^1	c_{45}^1	c_{12}^2	c_{13}^2	c_{34}^2	c_{24}^2	c_{25}^2	c_{45}^2	Z	

Notes:

- Node i_k is node i for Commodity k.

- The dual formulation has one constraint for each link, whereas the primal formulation has one constraint for every node (except for the shipment origin node, which is redundant and has been eliminated.)

- Dual variables w_i^k on flow conservation constraints give the cumulative distance from the origin node 1 for node i, commodity k.

- In this simplified example, there are no "overlapping" segment capacity constraints, dual variable u_1 corresponds to the coupling constraints for links1-2, variable u_2 couples links 3-4.

First, note that equations (3.5.1) and (3.5.2) lack terms for $-w_1^k$. The w_1^k terms are missing here because the flow conservation constraint for the origin node (1) was omitted from the primal formulation, as a redundant row. In the dual formulation, that is essentially the same as fixing all w_1^k equal to zero.

Second, note that the u_1 and u_2 dual variables in equations (3.5.1) and (3.5.3) can be rearranged as follows:

$$-w_3^1 + w_4^1 \geq c_{34}^1 - u_2 \qquad (3.5.3)$$

Since link cost coefficients in the TSP formulation are all negative by definition (except for the supersink links, where revenues are represented as positive quantities) and u_s's are positive, this shows that the dual price adjustments are additive to the original link cost coefficients.

Equations (3.5.2) and (3.5.4) lack u_s dual variables because they are yard inventory holding links, and train capacity constraints do not apply. The current formulation does not model any fixed car capacity constraint in the terminals. Similarly, equations (3.5.5) and (3.5.6) do not include u_s dual variables because they are fictitious super sink links, having no capacity constraint. Since the TSP formulation is for profit maximization, the dual objective function is:

$$\min \ \xi_1 w_5^1 + \xi_2 w_5^2 + C_1 u_1 + C_2 u_2 \qquad (3.5.7)$$

Equation 3.5.7 says to minimize the weighted sum of shadow prices of the coupling capacity constraints u_s, plus the weighted sum of "w_5^k" terms. The "w_5^k" terms require further interpretation. Reorganize (3.5.5) to solve for w_5^1:

$$-w_2^1 + (1/p_{25}^1)\, w_5^1 \geq c_{25}^1 \qquad (3.5.5)$$

$$(1/p_{25}^1)\, w_5^1 \geq c_{25}^1 + w_2^1$$

$$w_5^1 \geq p_{25}^1 (w_2^1 + c_{25}^1)$$

Remember, in the TSP formulation, the dual variable w_2^1 will be a negative quantity. In an optimal solution, its absolute value will represent the minimum cumulative cost to move from the shipment origin node "1" to possible termination node "2".

The revenues to be earned by delivering shipment 1 at node or time "2" are represented by $c_{25}1$, which is a positive quantity. Substituting these "English" definitions for the variables, the implications of (3.5.5) become clear:

$$w_5^1 \geq \text{Probability of Customer Acceptance (Revenue - Cost)}$$

This condition can only be satisfied for all possible delivery nodes, if w_5^1 is set equal to the *greatest possible* expected profit. Otherwise, at least one dual constraint will be violated and the solution would be infeasible. Select:

$$w_\Omega^k = \max p_{i\Omega}^k (w_i^k + c_{i\Omega}^k) \qquad \text{for all links (i,j) inbound to the Super Sink "}\Omega\text{".}$$

This procedure, in fact, will actually be proposed in Section 4.2.1 to select delivery times in the TSP subproblems.

3.5.1 General Formulation of the Dual

The general formulation of the dual has three types of constraints:

1) Links which originate at the shipment origin node i, are missing their $-w_i^k$ terms because the origin node flow conservation constraint has been dropped from the primal formulation.

2) "Intermediate" links which do not touch either the origin or supersink nodes appear as classic "network dual" constraints. This form of constraint is the underlying motivation for the design of label-correcting shortest path algorithms, for example.

3) Links which terminate at the super sink node Ω incorporate the special probability terms which represent the likelihood of customer acceptance of the service offer.

The general dual formulation is:

$$\text{Min} \sum_{k \in K} \xi_k w_\Omega^k + \sum_{s \in S} C_s u_s \qquad (3.5.8)$$

where Ω is the "Supersink" node

subject to:

$$w_j^k + \sum_{s \in S_{ij}} u_s \geq c_{ij}^k \qquad (3.5.9)$$

for each link $(i,j) \in A$ where "i" is the origin of shipment "k", for all $k \in K$

$$w_j^k - w_i^k + \sum_{s \in S_{ij}} u_s \geq c_{ij}^k \qquad (3.5.10)$$

for each link $(i,j) \in A$ where "i" is NOT the origin of shipment "k", and $j \neq \Omega$, for all $k \in K$

$$(1/p_{i\Omega}^k)\, w_\Omega^k - w_i^k \geq c_{i\Omega}^k \qquad (3.5.11)$$

for each link $(i,j) \in A$ where $j = \Omega$, for all $k \in K$

3.6 Representing Forecast Demand Uncertainty

Charnes and Cooper [1963] define a "Chance Constrained" formulation as a problem of the following kind:

optimize f(c,x), subject to:

$$P(Ax \leq b) \geq \alpha$$

where:

- "A" is a matrix of linear program constraint coefficients
- "x" is a vector of linear program decision variables
- "b" is a vector of linear program right hand side values
- "c" is a vector of objective function coefficients
- "α" is the acceptable probability of constraint violation

wherein “P” means “probability” and A, b, c are not necessarily constant but have, in general, some or all of their elements as random variables.

An alternative formulation of the train segment pricing model would treat uncertainty in future demands by replacing capacity coupling constraint 3.4.4 with a “chance constraint” — given an uncertain demand forecast, the goal would be to keep the probability of overflowing the train within acceptable limits, specified by the parameter α. An alternative chance constrained form of equation 3.4.4 follows:

$$P\left\{ \sum_{k} \sum_{(i,j)\in a_S} x_{ij}^{k} > C_S \right\} \leq \alpha \quad \text{for all } s \in S$$

(Chance Constrained Version)

A second change in the formulation would be that “N_k” (the number of cars in each shipment) becomes a random variable instead of a constant. Since only a discrete number of cars can be shipped, an appropriate assumed distribution function for “N_k” might be the Poisson. The network flow conservation constraints propagate randomness from “N_k” into the x_{ij}^{k} decision variables, thence into the chance constraint above. The relationship would be even more direct if the problem were reformulated using path variables rather than link-node representation.

Charnes and Cooper show that, for the case of a continuous, symmetric distribution which is completely defined by its first two moments, the chance constraints can be transformed into a deterministic form which corresponds to the interior of an elliptic hyperboloid, a convex set. Thus a linear stochastic problem is transformed into a deterministic problem having a *linear objective function with nonlinear constraints*. Charnes and Cooper give an example of applying this methodology to a machine loading problem. This particular transformation is not applicable here since the assumed distribution of “N_k” has a discrete, not a continuous density function.

An alternative approach was pioneered by Cooper and LeBlanc [1977] who define a “Stochastic Transportation Problem” for the case where transportation costs and supplies are known, but demand at destination nodes is uncertain. An inventory cost term, based on expected shortage or surplus cost, is incorporated into the objective function, leading to a problem with a *non linear objective function, but linear constraints*.

The choice whether to model a certain problem as a stochastic program or to utilize chance constraints is really up to the analyst. Either choice leads to nonlinearity in the deterministic equivalent. If objective function coefficients vary, then a stochastic program formulation naturally suggests itself. If instead constraint coefficients or right hand sides take on random values (the case here), it is impossible to guarantee feasibility 100% of the time. The direct approach is simply for the user to specify some target feasibility level “α” — the chance constraint approach.

Some early models (Kraft, Oum, and Tretheway [1986]) used a chance constrained approach for both airline overbooking and seat allocation. A more sophisticated way of dealing with infeasibility is to assess a penalty cost for constraint violations and let the optimization program choose the best value of α. This converts a chance constrained formulation into a stochastic program. The article by Smith, Leimkuhler, and Darrow [1992] described the modern yield management approach of equilibrating the expected value of revenue between fare classes. This newer approach eliminates the need for management to specify an a priori α cutoff value — this is now determined implicitly by the program. This approach is attractive if the cost of constraint violations can be precisely and accurately quantified.

If, however, the cost of violation is hard to quantify (dominated by such "soft" considerations as lost customer goodwill) then a chance constrained approach is both a more direct and honest treatment. As a practical matter, an appropriate booking limit B_S for each segment can be determined through experience. For example, consider the case of an airplane having 100 seats. The airline might determine it is safe to sell up to 110 seats because of no-shows. If this flight is the last plane out in the evening, the airline might want to authorize the sale of only 90 seats to save room for up to 10 missed connections.

Implementation of a chance constrained or probabilistic formulation for Train Segment Pricing would increase the input data requirements. A probability distribution of demand, as shown in Table 3.2, would have to be calibrated for every origin destination pair that has any significant probability of producing traffic. Any requirement to calibrate not only the first, but also the second and possibly higher moments of demand would present a daunting implementation challenge, at a time when rail carriers are not known to have developed any kind of short term, origin destination demand forecasting at all.

Table 3.2: Sample Demand Distribution

Number of Cars	0	1	2	3	4+
Probability	.4066	.3659	.1647	.0494	.0134

Expected # of Cars = 0.9 per day

In contrast, a linear-program based, deterministic formulation requires only a forecast of expected, average demand by origin destination. When benchmarked against the much more complex non linear models, such LP-based formulations have performed well.

Williamson [1992] (pp. 68-69) gives a linear programming formulation for an airline network yield management problem:

$$\text{Max} \sum_{odf} f_{odf}\, x_{odf} \quad \text{(Maximize fare x volume = Revenue)} \qquad (3.6.1)$$

subject to:

$$\sum_{odf} x_{odf} \leq C_j \quad \text{(Seats sold cannot exceed flight capacity } C_j\text{)} \qquad (3.6.2)$$

$$x_{odf} \leq D_{odf} \quad \text{(Seats sold cannot exceed ODF demand } D_{odf}\text{)} \qquad (3.6.3)$$

$$x_{odf} \geq 0 \quad \text{for all odf (Nonnegativity)} \qquad (3.6.4)$$

Using a rolling horizon simulation model, Williamson [1992] compared the performance of this deterministic LP versus a nonlinear, probabilistic variant of the same network model, and versus the leg-based EMSR heuristic (Belobaba [1987]). She concluded (pp. 173-174, 177, 184):

> Although the probabilistic solution performs better in the partitioned, non-nested case, the *nested deterministic methods consistently outperform the nested probabilistic methods.* The reason for this is that the probabilistic network solution tends to "overprotect" seats for the more desirable and higher fare class ODF's. While more desirable ODF's can have access to additional seats through nesting without explicitly allocating such seats to the ODF's, excess seats allocated to the more desirable ODF's are not made available to less desirable ODF's for those cases where the seats are not used.
>
> It is not the case that the deterministic network seat allocations are the "optimal" nested ODF allocations, it is just that the deterministic seat allocations tend to be closer to the optimal nested protection levels than the probabilistic allocations from a static, non nested formulation of the network seat inventory control problem.
>
> As was the case for the nested deterministic and nested probabilistic approaches, the bid price approach based on the deterministic formulation of the network seat inventory control problem consistently outperforms the probabilistic bid price approach. This overprotection of seats leads to probabilistic bid prices which are often *higher* than the deterministic prices and sometimes "too high", particularly on critical flight legs.

It is hard to generalize Williamson's findings, however, beyond the scenarios tested in her disseration. Two factors appear to have possibly influenced her results: the assumed frequency of capacity allocation updates, and the specific non linear formulation tested.

In Williamson's disseration, the precise frequency of bid price updates is not stated; but since "reading days" are discussed several times in the description of the rolling horizon simulation, apparently bid price updates did not occur more frequently than once each simulated day. Belobaba [1989] found, as the frequency of updates to bid prices or allocations approaches real time, the impact of nesting diminishes and eventually disappears entirely. Thus, as long as the model is solved frequently enough, it is not necessary to include nesting in the mathematical formulation of the optimization model. A non-nested, non-linear, stochastic formulation will produce the optimal solution even for the nested problem, provided the result is updated often enough.

Recently, Talluri and Van Ryzin [1996] found that the particular Probabilistic Non Linear Programming (PNLP) formulation which Williamson tested has poor theoretical properties, since it will develop inconsistent bid prices depending on how finely demand is disaggregated in the model input data. It's not clear whether the poor performance of nested PLNP in Williamson's tests was due to this aggregation problem, or if it might simply be due to the bid prices not having been updated frequently enough.

Alternative nonlinear, probabilistic revenue management formulations do exist, however, which have good theoretical convergence properties. Recent research has focused on the application of stochastic, dynamic programming techniques to the calculation of provably optimal bid prices (see Lee and Hersh [1993]; Papastavrou, Rajagopalan and Kleywegt [1996]; Stone and Diamond [1992]). Such dynamic programming approaches model the details of the booking process much more carefully than conventional math-programming based approaches. Of course, the input data requirements are correspondingly

increased, and processing all of this highly detailed information increases the required CPU execution time well beyond what is required by conventional math-programming based methods.

Talluri and Van Ryzin's [1996] paper also discusses the strengths and weaknesses of the bid price control method in general, and presents several conditions under which bid price controls might not lead to optimal decision making. However, they still conclude that a bid price control scheme is "close to being globally" optimal, and are continuing to research improved methods for developing more accurate bid prices.

In spite of Williamson's [1992] computational results, it is generally accepted that a non-nested, non-linear, stochastic formulation will dominate the performance of the linear deterministic model, provided that an *appropriate* non linear formulation is used, and if the bid prices are updated frequently enough.

Nevertheless, the deterministic model is a robust, simple and commonly accepted approach used in a large number of real world revenue management systems, including Amtrak's. Implementation of non linear, or stochastic, dynamic programming approaches for the Train Segment Pricing problem would require concurrent improvements to railroad demand forecasting capability, so both topics will be left to future researchers.

The linear formulation proposed by Williamson [1992] is not dissimilar to the model proposed here for the railroad shipment routing problem. The primary difference is that Williamson's model is formulated in path variables, whereas the model proposed here uses a link-node network structure. However, path-based formulations for similar freight shipping problems have been proposed by other researchers, including White and Wrathall [1970], Farvolden, Powell and Lustig [1993] and Kwon [1994].

Given Williamson's computational results, this deterministic formulation does not seem an unreasonable choice for a "starter" railroad yield management system. As well,

the simpler linear formulation using deterministic booking limits offers the following practical advantages:

1) Input data requirements for the demand forecast are less demanding since only the expected value of flow, not the entire density function, must be forecast.

2) It is easier to understand and use, since users can specify train segment booking limits in easily understandable units of feet and tons, rather than having to guess to calibrate α, with unpredictable results. The user interface is much friendlier.

3) The expected value formulation is numerically much easier to solve, yet provides a comparable level of end user functionality.

By assuming a Poisson distribution of demand, it becomes a simple matter to relate booking limits B_s to the physical segment capacity C_s. Consider the case where the maximum physical segment capacity C_s is 2 units. If $\lambda = 0.9$ cars/day, using the Poisson probabilities shown in Table 3.2, the probability of exceeding capacity would be .0494 + .0134 or .0628. If $\alpha = .05$ were the level of tolerance, this would be considered a constraint violation. The equivalent booking limit B_s can be found by solving the equation of the Poisson distribution for the maximum λ that will still satisfy the chance constraint:

$$\sum_{k=0}^{2} \frac{e \quad k}{k!} < 0.95 \qquad \text{(Solve for)}$$

The derived booking limit is $B_s = .819$, based on physical segment capacity $C_s = 2$ and maximum acceptable overflow probability $\alpha = .05$. In a practical application a much higher α limit would probably be used, depending on the type of traffic offered and how much “postponable” or lower priority traffic is present in the traffic mix. For typical large C_s greater than about 30 cars, C_s would approximately equal B_s when $\alpha = .50$. The impact of

adjusting the overflow probabilities α and booking limits B_s will be experimentally assessed in Section 5.7.

3.7 Limitations on the Scope of this Research

The current formulation models the availability of "slots" on trains, predicting when a car might be delivered at its destination. Destinations would be assigned to empty cars using the current empty car distribution system. The extension to simultaneously determine loaded shipment routing and empty car destinations will be left to a future researcher.

Given a planned train network, the model routes the cars and produces trip plans. The model can be operated in "what if" mode to see what would happen if the operating plan or train capacities were changed. Dual variable prices along with projected car flows might be used as triggering mechanisms to guide heuristic search routines to look for the most likely opportunities to improve the tactical operating plan. Any tactical plan modification would be profitable if, when cars are reflowed over the revised network, the objective function improvement exceeds the fixed cost of operating the additional trains.

However, just because a plan modification would be "profitable" does not guarantee that there are sufficient crews or power to operate all such trains. Any proposed modifications should be "sanity tested" or constrained by the crew and locomotive management systems to ensure they are feasible before presenting them as options to the car scheduling process.

3.8 Requirements for Solution Algorithms

This Chapter outlined formulations for two related shipment routing problems. These two problems are sufficiently distinct in inputs, required outputs and solution time frame to justify different solution approaches, which will be presented in Chapter 4.

• The “train segment pricing” model accepts a projection of future traffic which has not, as yet, materialized. Since this algorithm works with expected values and not with realizations of actual traffic, non integral flows predominate. Its primary focus is to *estimate dual prices for each train segment* , which are used to determine appropriate service commitments when the actual loads call in. This concept for scheduling delivery appointment times is modeled after current motor carrier industry practice and implements a “bid price” revenue management approach. Under a standard bid price approach, a shipment “k” should be accepted if there exists for at least one possible routing:

$$\mathbf{Rev}_k > \sum_{\substack{\text{All links (i,j)} \\ \text{traversed}}} c_{ij}$$

Beyond this, however, a modified shortest path algorithm is proposed in Section 4.2.1 to suggest the *optimal* delivery appointment time for each shipment from the rail carriers’ point of view. A feasible solution can be obtained using a Lagrangian heuristic, but is only of secondary importance, calculated primarily to validate the quality of the dual prices by establishing that these prices lead to a tight upper bound.

• The “dynamic car scheduling” algorithm routes *real* shipments in *real time,* producing trip plans and instructions for classification of cars in yards. Since each shipment already has a target delivery time, this program’s job is simply to develop a plan to *deliver all shipments by the agreed-upon time at minimum cost* . An integral flow requirement must be satisfied because the application is for routing of real railroad cars which cannot be split in half.

The two main requirements for dynamic car scheduling, then, are to consistently produce a high quality solution and to maintain this solution current in real time. Unfor-

tunately, these two goals are in conflict. Provable, true optimality may not be a reasonable design objective; but if the solutions are not optimal, at least they need to be defensible and as good as a reasonable human decision maker might have made under the same circumstances.

CHAPTER 4

Solution Algorithms

4.1 Chapter Outline

Algorithms for solving both the train segment pricing and dynamic car scheduling problems will be presented here. Both are based on a Lagrangian relaxation (LR) of train capacity constraints which decouples the multicommodity network flow (MCNF) formulation into a series of shortest path problems. Subgradient and dual adjustment approaches have been successful in other network modeling applications, particularly if the goal is to quickly obtain a near optimal, near feasible solution. The most important reason for using LR is the size of practical problems. An LR-based algorithm avoids matrix inversion, yet still develops an optimality bound so the duality gap can be measured. In addition, integral flow requirements in the dynamic car scheduling model cannot easily be satisfied by linear programming-based methods; a true integer programming approach such as LR must be used.

A standard subgradient step-size algorithm is used to solve the train segment pricing problem. The problem is extremely large since all flows forecast for a week or more must be assigned to a space-time network. Substantial computational effort may be required because every "modified" (see Section 4.2.1) shortest path subproblem is resolved at each iteration. Fortunately, this problem need not be solved in real time. No

attempt is made to split flows over more than one route or to equilibrate alternative path costs. This all-or-nothing assignment reduces degeneracy in the final solution (as discussed by Powell [1989]) and improves the speed of the algorithm, but may leave a small duality gap. The algorithm is readily adaptable to large scale parallel processing, so real-world sized problems can almost certainly be solved in an acceptable time frame.

In contrast, the dynamic car scheduling problem must be solved in real time, adapting the previous solution to keep it current any time new information is received. The key to being able to accomplish this appears to be restricting the traffic to only the cars currently moving on the railroad. As well as limiting the problem size, it reduces degeneracy in the optimal solution to a manageable level, making a dual adjustment solution approach attainable. *This functionality is very similar to what today's car scheduling systems provide,* with the added feature of taking train segment capacities into account.

By limiting adjustments only to price increases, only flows directly impacted by a price increase must be reassigned. By avoiding the need for matrix inversion, and by sharply limiting the need to recalculate shortest path subproblems, the goal is to develop a robust, adaptive, real time control algorithm for dynamic railroad shipment routing.

Section 4.2 presents the LR used here and discusses some of its mathematical properties. It also presents a solution algorithm for solving shortest path subproblems resulting from this LR. Then Section 4.3 reviews mathematically-oriented literature in the area of multicommodity network flows, Lagrangian relaxation, subgradient optimization and dual ascent techniques.

Section 4.4 presents a deterministic dual adjustment algorithm for routing the cars which are already on the railroad. The formal statement is given in Section 4.4.1, followed by a discussion of some of its properties and practical performance in Section 4.4.2.

Section 4.5 presents a train segment pricing algorithm, based on a subgradient step size procedure, for loading forecast demands on the train service network for 7-10 days into the future. The algorithm is formally stated in Section 4.5.1. At each iteration, a primal heuristic presented in Section 4.5.2 can be applied to produce a feasible solution and allow measurement of the duality gap. A discussion of the convergence characteristics of this algorithm and computational test results follows in Section 4.5.3.

4.2 The Lagrangian Relaxation

Introducing dual variables $u_s \leq 0$ for each train segment, train segment capacity constraints in the Dynamic Car Scheduling problem (3.3.5) can be dualized, bringing them into the objective function.

$$\text{Max} \{ \text{Min} \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k + \sum_{s \in S} u_s \left(C_s - \sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k \right) \} \quad (4.2.1)$$

subject to: "Conservation of Flow Constraints (3.3.3 and 3.3.4)"

This decouples the problem into independent pure shortest path problems for each commodity "k". Since network problems always have integral solutions, the integral flow constraint (3.3.6) is then satisfied automatically. Applying the definition of φ_{ij}^k:

$$\varphi_{ij}^k = c_{ij}^k - \sum_{s \in S_{ij}} u_s$$

equation 4.2.1 can be reorganized as:

$$\text{Max} \left\{ \text{Min} \sum_{k \in K} \sum_{(i,j) \in A} \varphi_{ij}^k x_{ij}^k + \sum_{s \in S} u_s \, C_s \right\} \quad (4.2.2)$$

The interpretation of the "Max { Min" notation of (4.2.2) is as follows. First, fix a set of values for the dual variables u_S. Solve for "k" independent shortest paths as cost minimization problems. After each iteration, adjust the values of u_S in such a manner as to maximize the expression given inside the braces. This result will give a lower bound on the value of the optimal objective function to the original cost minimizing problem.

For the Train Segment Pricing (TSP) problem, the relaxation is very similar, except that B_S is used in the Pricing model in place of C_S, and because the TSP is for profit maximization, the dual variables $u_s \geq 0$ and the order of the "Max { Min" is reversed. The result will give an upper bound on the value of the optimal objective function to the original profit maximizing problem. The lagrangian relaxation for the TSP problem is:

$$\text{Min} \left\{ \text{Max} \sum_{k \in K} \sum_{(i,j) \in A} \varphi_{ij}^k x_{ij}^k + \sum_{s \in S} u_s \; B_S \right\} \qquad (4.2.3)$$

A Lagrangian relaxation has Geoffrion 's [1974] "integrality" property if, after the complicating constraints have been relaxed, the subproblems naturally have all integer solutions. This is true here. Geoffrion showed that if a Lagrangian relaxation has this property, the LR bound can be no better than the linear program (LP) bound.

A number of successful relaxations having the "integrality" property are reported in the literature. These include the classical solution to the Traveling Saleman problem by Held and Karp [1970] [1971], work on the set covering problem by Etcheberry [1977], and the database location problem by Fisher and Hochbaum [1980]. The solution to the Traveling Salesman problem by Held and Karp [1970] is successful because the LP solution gives a tight bound on the IP objective function, and the LR is much faster to solve than the LP. In many cases the LP has an exact integer solution, and in these cases the Lagrangian relaxation will converge to the optimum; otherwise, Held and Karp show that

any duality gap corresponds to the possible existence of nonintegral extreme points. Held and Karp's second paper [1971] does not change the relaxation but proposes a faster subgradient search procedure for the optimal dual values. Held and Karp show how this relaxation can drastically reduce the size of branch and bound trees.

Fisher [1981] reports that Lagrangian Relaxations with the integrality property can be successful when:

- The LP closely approximates the original integer problem or
- The method used to optimize the dual problem (usually the subgradient method) is more powerful than methods for solving the (generally large) LP relaxation of the original problem.

Fisher and Hochbaum [1980] show that both conditions do not necessarily have to be met if the plan is to embed the Lagrangian Relaxation inside a branch and bound algorithm. While tighter bounds are always desirable in order to fathom branches sooner, a Lagrangian Relaxation may still produce considerable performance improvement *if only it runs faster than the original LP*. A branch and bound algorithm would still build the same tree as it would using a standard LP approach, only it would do so much faster using Lagrangian relaxation to solve the subproblems. This appears to be the strategy in Fisher and Hochbaum [1980]. A Lagrangian relaxation having the integrality property may not initially produce a very tight bound, however that bound will tighten and approach the optimal value as branching proceeds.

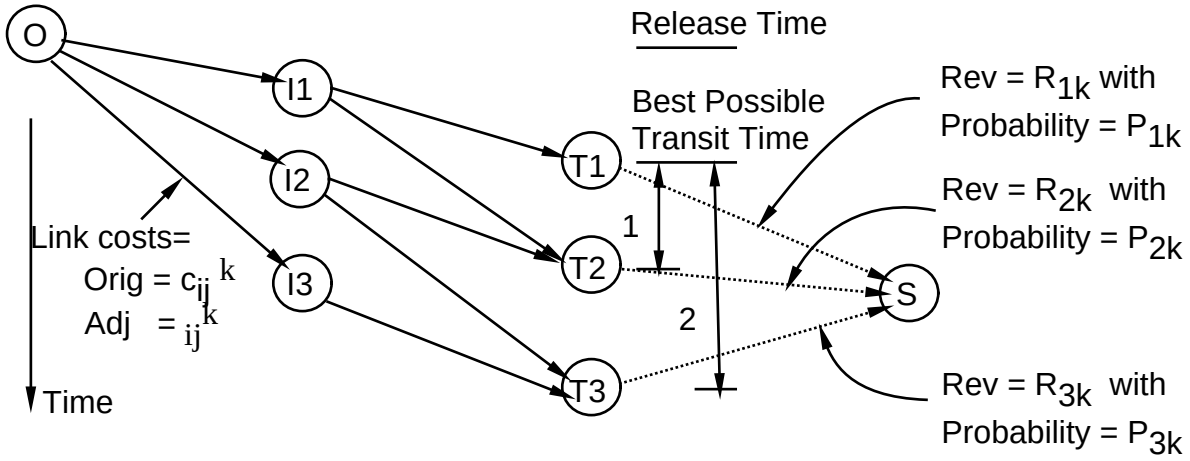
Since branch and bound will not be utilized here, a tight bound is necessary, however, Nozick [1992] solved a closely-related network problem as a linear program, and simply by applying a rounding heuristic to generate feasible integer solutions, was able to obtain tight bounds compared to the original LP solution (within approximately 1%) . This result suggests that the LP bound will in fact give a reasonably tight bound on the optimal

integer solution. Further, duality gaps of less than 1% have been quickly attained in computational tests of the subgradient algorithm on this problem. Due to complicating side constraints, the LP is very large and difficult to solve, but shortest path subproblems can be solved very quickly. Both of Fisher's conditions for a successful relaxation are met here.

4.2.1 Shortest Path Subproblems with Gains

Shortest path subproblems in the dynamic car scheduling application are solved by a dynamic programming approach. A customized "bucket-sorting" algorithm specialized for acyclic networks will be proposed in Sections 4.2.3 and 4.2.4. A slight modification is required in the last stage of the normal shortest path recursion to handle gain coefficients present in the Train Segment Pricing model. The reservations center would utilize this modified shortest path algorithm on a stand-alone basis to determine service offers for new shipments calling in. The remainder of this section will describe the required modifications to implement the Train Segment Pricing algorithm.

Figure 4.1: Modified Shortest Path Subproblem for Commodity "k"



Before considering algorithms for the overall Pricing problem, it might be useful to first consider how the subproblems can be solved, given a set of dual prices u_s for each train route segment, as shown in Figure 4.1. The shipment starts at origin node "O" and needs to move to any of the termination nodes T_{1k}, T_{2k} or T_{3k}. These termination nodes all correspond to the same physical destination, but having deliveries at different times.

Revenues R_{nk} and costs C_{nk} will be treated separately here, even though they are combined into a single cost coefficient c_{ij}^k in the formal definition of the problem. Depending on which delivery time is offered, a different price R_{nk} might apply, and the customer may have a different probability of accepting the service offer, P_{nk}. The cost to move to delivery node "n" is C_{nk}. The P_{nk} probabilities are input data; they can be calculated ahead of time because they depend only on the delivery time and not on the path used. The carrier wishes to determine which delivery offer will maximize expected profit $\Pi_{nk} = P_{nk} (R_{nk} - C_{nk})$.

This problem can be solved as follows:

1) The first step is to adjust the original link costs c_{ij}^k by adding dual prices u_s for train segments spanned by each link (i,j). The shortest path calculation can alway
s be based on adjusted link costs φ_{ij}^k. The validity of this equation is established by equation 3.5.9 of the dual formulation.

2) Starting at the origin node, find the shortest paths to each termination node $n \in T_{nk}$. Determine the costs of reaching each node as C_{nk}. This is done using a normal shortest path calculation.

3) Calculate expected profit Π_{nk} of moving via shortest path to each node $n \in T_{nk}$ as:

$$\Pi_{nk} = P_{nk} (R_{nk} - C_{nk})$$

where P_{nk} is the probability the customer accepts the transit time quotation for shipment "k" if the service offer is for delivery at node "n", and R_{nk} is the associated revenue. This is verified by equation 3.5.11 of the dual formulation.

4) Select the optimal node n* as argmax (Π_{nk}). The optimal routing of this flow is the set of links k $\in$ { Links on the shortest path from origin to node n* }. The optimal objective function value for this flow is Π_{n*k}. If all Π_{nk}'s are negative, the load should be rejected outright. This is verified by equation 3.5.5 of the dual formulation, and the related discussion.

This calculation is performed by examining all possible delivery nodes for each car. This replaces the last stage of the regular dynamic programming recursion, requiring practically no additional computational effort.

4.2.2 Shortest Path Literature Review

The acyclic structure of the space-time network creates an opportunity to use a highly specialized and efficient version of the shortest path algorithm. Also, since the shortest path routine will be called repetitively, further efficiencies can be gained through preprocessing the network and storing certain information — the "stage number" of each node — in advance. The following presents a brief review of shortest path literature, followed by a description of the customized shortest path algorithm used in this dissertation.

Dijkstra [1959] proposed his famous algorithm for finding shortest path trees in cyclic networks. It is a "label setting" procedure, based on the idea of adding one link onto the shortest path tree at a time, always starting at the node having least distance from the root node. Pape [1974] compares algorithms by Moore and D'Esopo, claiming these more

efficient than Dijkstra's, although no complexity analysis is given. Pape's paper discusses implementation details using circular and linked lists.

Frederickson [1987] gives a very complex algorithm that subdivides networks into smaller parts and has a worst case complexity of $O(n (\log n)^{1/2})$ time, compared to n^2 for Dijkstra's algorithm. However, that algorithm does not exploit the advantages inherent in an acyclic network.

Florian, Nguyen and Pallottino [1981] describe an algorithm for updating a previously computed shortest path tree, if it is desired to switch to a new root node: these concepts have potential application here. They give the worst-case complexity of this algorithm as $O(n^2)$, the same as Dijkstra's. This worst case analysis might be misleading however, since if the new root node is "nearby," only a few iterations on average should be required to update the tree.

Dial, Glover, Karney and Klingman [1979] delve into many implementation details, which proved especially helpful in designing the shortest path algorithm used here. They describe the "Dijkstra Address Calculation Sort" (also known as a "bucket sort") where an array of size $(l_{max} + 1)$ stores pointers to linked lists of nodes filed according to their temporary node potentials. All nodes in a "bucket" have the same potential, so information can be both filed and retrieved without searching.

Divoky and Hung [1990] study the performance of shortest path algorithms when they are embedded as subproblems in large scale network optimizations. In the context of the minimum cost flow problem, Divoky and Hung's concern seems to be that some decomposition techniques will create links with zero cost, leading to degeneracy in the shortest path subproblems, adversely affecting the performance of the shortest path algorithms. The formulation proposed here will not create zero cost links anyway, since all costs start out positive and negative dual variables are not allowed. They found that bucket-

sorting algorithms, such as the one used in this dissertation, tend to perform best in embedded applications.

The above are all general-purpose shortest path algorithms. An even more efficient approach customized for *acyclic* shortest paths has been implemented along the lines suggested by Bradley, Hax and Magnanti [1977]. The algorithm proceeds in two steps:

- First, each node is assigned a "stage," the maximum number of links depth from some "root" node, which depends only on network topology and not on link costs. The stage numbers can be computed once, ahead of time, and stored.
- Then the shortest path can be computed by visiting each node in the fixed sequence defined in the first step without ever having to repeat the stage number assignment.

Since the stage number is always integral, its use is directly compatible with the bucket sorting approach. For a cyclic network, Dijkstra's address calculation sort requires that all link lengths be integral. However, for an acyclic network, the stage number can be used in lieu of cumulative distance to determine the sequence in which nodes are visited. Link lengths can be any positive real number, since the visitation order doesn't depend on link lengths.

4.2.3 Determining "Stage" or Depth of Each Node

Define:

sn = The current stage number being processed

SN = The maximum stage number

n = The current node number

stg_n = The stage number assigned to node n

fs_n = The set of nodes contained in the forward star of node n

f = A specific node in the forward star, $f \in fs_n$

P_{sn} = The set of pending node numbers for stage "sn"

The "forward star of node n" is the set of nodes which are directly reachable by traversing any single link originating at node "n."

Step 1:

$P_{sn} = \varnothing$ for all $sn \leq SN$.

$sn = 0$.

Step 2:

For each node n having no incoming links, do:

$stg_n = 0$.

$P_0 = P_0 \cup \{ n \}$.

Step 3:

For each node $n \in P_{sn}$, do:

For each node $f \in fs_n$, do:

If node number f appears in P_s but hasn't been processed yet, remove it to avoid redundant work.

$stg_f = sn + 1$.

$P_{sn+1} = P_{sn+1} \cup \{ f \}$.

Remove node n from P_{sn}.

Step 4:

If $P_{sn} = \varnothing$ then $sn = sn + 1$.

If still $P_{sn} = \varnothing$ then end else go to Step 3.

4.2.4 Determination of Shortest Paths

Define:

sn = The current stage number being processed

SN = The maximum stage number

n = The current node number

stg_n = The stage number assigned to node n

$cost_n$ = The cost assigned to node n

$pred_n$ = The predecessor of node n

fs_n = The set of nodes contained in the forward star of node n

f = A specific node in the forward star, $f \in fs_n$

P_{sn} = The set of pending node numbers for stage "sn"

c_{ij}^k = Cost of car $k \in K$ moving over link $(i,j) \in A$.

Step 1:

$P_{sn} = \varnothing$ for all $sn \leq SN$.

$cost_n = \infty$ for all $n \in N$.

Set the current node n to the origin of the shipment to be routed.

For each node $f \in f_{sn}$, do:

$P_{stgf} = P_{stgf} \cup \{ f \}$.

$cost_f = cost_n + c_{nf}^k$.

$pred_f = n$.

$sn = stg_n + 1$.

Step 2:

For each node $n \in P_{sn}$, do:

For each node $f \in fs_n$, do:

$P_{stgf} = P_{stgf} \cup \{ f \}$. (Unless $f \in P_{stgf}$ already)

If $cost_f > cost_n + c_{nf}^k$ then:

$cost_f = cost_n + c_{nf}^k$.

$pred_f = n$.

Remove node n from P_{sn}.

Step 3:

If $P_{sn} = \varnothing$ then $sn = sn + 1$.

If sn > SN then end else go to Step 2.

4.3 Mathematical Programming Literature Review

At least five distinct approaches to the general multicommodity network flow problem are reported in the literature: specialized simplex methods, heuristic approaches, interior point methods, nonlinear penalty/barrier function methods, and subgradient optimization. Kennington [1978] and Assad [1978] survey some of the earlier literature on this subject.

It is well known that the "network simplex" algorithm maintains a triangular basis which requires only back substitution, not matrix inversion to solve. The specialized simplex based multicommodity algorithms cannot entirely avoid the need to perform matrix inversion, but they try to minimize it by partitioning the matrix into network-structured and non network-structured components.

Hu [1963] gives an adaptation of the Out-of-Kilter algorithm (see Barr, Glover, and Klingman [1974]) to a multicommodity network flow problem. Some examples of

specialized simplex algorithms include the approaches of Grigoriadis and White [1972], Elam, Glover and Klingman [1979], Glover and Klingman [1981], Chen and Saigal [1977], Chen and Engquist [1986], Hartman and Lasdon [1972], Brown and McBride [1984], McBride [1985], McBride and Mamer [1993], and Farvolden, Powell and Lustig [1993].

McBride's approach [1985] is generalized to handle any kind of a network subproblem with complicating side constraints. Network data is represented in standard link-node form. His implementation incorporates a network simplex approach for solving subproblems without requiring inversion, carefully linked into a standard simplex algorithm which handles the side constraints. Thus the size of the matrix which must be inverted is limited to the side constraints.

McBride and Mamer [1993] built a primal allocation heuristic to "jump start" the network simplex at a near optimal solution, further reducing execution time. This heuristic creates an allocation of capacity to each link proportional to the resources used by that link in a trial solution with all capacity constraints relaxed. He reports that the "heuristic is not only fast, but obtains very good solutions" and is apparently often used on a stand-alone basis. This heuristic works well for the manufacturing logistics problems McBride has been solving, but it is not clear if it would transfer to the car scheduling problem. Proportional capacity allocation would create many non-integral flows.

Barnhart and Sheffi [1993] propose a heuristic solution to a shipment routing problem in the LTL trucking industry which is very similar to the railroad freight car scheduling problem posed here. Each shipment is a separate commodity, moving over a time space network where the side constraints represent truck capacities. Barnhart's heuristic uses a link-node formulation, like McBride's, but it may terminate without fully

solving the problem; also it apparently works best when network capacity far exceeds the amount required.

Farvolden's approach (see Farvolden, Powell and Lustig [1993]) solves the same trucking shipment routing problem as Barnhart's, but using a radically different method. First, Farvolden's approach is a systematic modification of the simplex method designed to produce an optimal linear programming solution, it is not a heuristic. Second, Farvolden reformulates the problem into a path-based formulation. Although Farvolden's algorithm does not incorporate network simplex, partitioning still allows specialization of pivot operations to gain efficiency. In most cases pivots can be completed without matrix inversion.

Farvolden (see Jones, Lustig, Farvolden and Powell [1993]) argues that her path based approach is so efficient that it can be used to solve general multicommodity flow problems, not just shipment routing problems. She proposes that general MCNF problems be decomposed into individual O-D pair flows, which are then solved quicker by her algorithm than using Dantzig-Wolfe decomposition with a network simplex algorithm.

Kwon [1994] outlines a railroad freight car scheduling problem similar to the one proposed here, formulated in path variables rather than link-node. He applies a standard column generation approach, similar to but less sophisticated than Farvolden's. Kwon has reported solution times of approximately 20 minutes for a small test problem, which still leaves considerable room for improvement.

Both McBride and Farvolden compare their algorithms' performance with that of the interior point code OB-1 (see Marsten, Subramanian, Lustig and Shanno [1990]). Although OB-1 clearly outperforms the standard simplex algorithm, both McBride and Farvolden argue that their customized codes compare favorably with OB-1 performance. McBride finds that "the purely interior point algorithms are less able to exploit sparsity and

therefore take considerably more space." Farvolden's PPLP code "solved problems of the LTL data set up to 50 times faster than the OB-1 solution and problems of the second data set up to 64 times faster."

Nagamochi and Ibaraki [1989] establish sufficient, but not necessary conditions under which MCNF problems will have integral flows. In [1990] they define a class CB (capacity balanced networks) of planar directed networks, where K is the number of commodities and |V| is the number of nodes. A network in CB satisfies the following conditions: (1) The graph is directed, planar and acyclic. (2) All nodes without entering arcs and all nodes without outgoing arcs are located on the boundary of the outer face of the graph. (3) Each commodity has exactly one source and one sink, where the sink is located on the boundary. (4) Each node is capacity balanced. It is shown that this class of networks has the integral flow property. They give an example of a multi-item, multi-stage production scheduling problem which can be easily transformed into a class CB network through addition of some fictitious flows.

Nagamochi and Ibaraki's research does not focus on solution algorithms but rather states some general mathematical conditions under which integral solutions to MCNF problems can be expected. While this research cannot be immediately applied to the dynamic car scheduling problem, future theoretical developments along these lines should be watched closely to see if they might be applicable.

Linear programming (LP) approaches can be used to solve integer programs, particularly where the LP solution is expected to be integral or near integral. If some fractional variable is encountered, branch and bound can be used until an integral solution is obtained. Miliotis [1976] proposes a LP solution to the traveling salesman problem, adding anticycling constraints on an exception basis only when needed and using branch-and-bound to eliminate fractional variables. Schrage [1975] proposes an LP solution to

variable upper bound problems such as the P-median, using a specialized simplex approach along with branch-and-bound. Using this type of approach, perhaps the integer MCNF problem (with a linear objective function) could be solved by embedding Farvolden's algorithm inside branch and bound. Many newer branch and bound applications are reported in the literature, but state-of-the-art research is more often based on Lagrangian rather than LP relaxations because these generally give tighter bounds and more rapid convergence.

Pinar and Zenios [1993] use a penalty function to eliminate non-network constraints from the MCNF problem. This approach converts a linear network problem with side constraints into a nonseparable nonlinear network problem without side constraints. This nonlinear problem is solved using a simplical decomposition algorithm which induces separability in the objective function. The penalty function approach has much in common with Lagrangian Relaxation, since both attempt to "price off" excess flows. Penalty functions are clearly the method of choice for MCNF problems with nonlinear costs, although linear costs can be handled as well. In closely related research, Schultz and Meyer [1991] propose a barrier-type algorithm.

Shapiro [1977] compares three methods for solving standard network optimization problems: out-of-kilter, primal-dual and subgradient optimization, but does not discuss the multicommodity case.

There are at least four references in the literature to subgradient methods applied to generalized network problems. Kennington and Shalaby [1977] utilize subgradient search to develop a resource-directive decomposition heuristic technique for obtaining good solutions to large multicommodity network cost minimization problems. The lower bound on optimality is taken from the solution to the unconstrained version of the problem, but

apparently there is no attempt to tighten or update this lower bound as the algorithm proceeds. It is not a Lagrangian relaxation based method and does not enforce an integral flow requirement.

Ali, Kennington and Shetty [1988] worked on a single commodity network problem with complicating side constraints requiring equal flows on certain links. Furthermore they required integral flows on all links. By relaxing the side constraints they obtain single commodity network subproblems. The Lagrangian problem is solved using a standard subgradient procedure. They report obtaining tight bounds very quickly in spite of the fact their relaxation has the integrality property. They remark that the subgradient technique "is best suited for a real-world situation in which one must quickly produce near-feasible, near-optimal solutions."

Bertsekas and Tseng [1988] apply Lagrangian relaxation to standard single commodity network flow problems, with and without gains. They dualize the conservation of flow constraints and then use what amounts to a dual adjustment method to find optimal multipliers. Their method includes "Flow Augmentation" and "Price Adjustment" phases which are similar to methods proposed here for the car scheduling problem. The algorithm terminates when complementary slackness conditions are satisfied. Their relaxation obviously possesses the integrality property, but this poses no difficulty, since the solution to network problems with integer demands and capacities is also guaranteed to be integral.

The key to their success is that "the ascent directions used . . . lead to comparable improvement per iteration as the direction of maximal rate of ascent but can be computed with considerably less overhead." They report speedup factors of an order of magnitude over the fastest primal-dual codes available in 1985, this speedup factor increasing with problem size.

Bertsekas and Tseng [1988] (pg. 111) report a practical advantage of using relaxation methods:

> For example, suppose we solve a problem and then modify it (by changing a few arc capacities and/or node supplies). To solve the modified problem using the relaxation method, we use as starting node prices the prices obtained from the earlier solution, and change the arc flows that violate the new capacity constraints to their new capacity bounds. Typically, this starting solution is close to optimal, and solution of the modified problem is extremely fast. By contrast, to solve the modified problem using primal simplex, one must provide a starting basis.
>
> The basis obtained from the earlier solution will typically not be a basis for the modified problem. As a result, a new starting basis must be constructed, and there are no simple ways to choose this basis to be nearly optimal.

Thus once an initial solution has been obtained, a relaxation-based algorithm can quickly "bootstrap" itself from one solution to the next as status updates are reported.

Bryson [1991] analyzes the case of a single commodity network flow problem with a complicating "knapsack" side constraint, and the requirement that all flows must be integral. The problem is solved using a subgradient algorithm to obtain a "good" initial solution, then a dual simplex pivoting method is used to reach the final solution. Bryson's algorithm must be embedded into a branch and bound program to ensure optimality since the relaxation used has the integrality property, and the additional side constraint would almost certainly introduce fractional flows into the optimal LP solution.

Held, Wolfe and Crowder [1974] establish a convergence guarantee on the subgradient procedure, so that the subgradient procedure can often find a stronger lower bound than dual adjustment techniques which may not always converge to the optimum. However, the convergence of the subgradient approach can also slow down as the optimum is approached, at this time it may become more beneficial to switch over to another method such as a dual adjustment heuristic.

This is consistent with the findings of Etcheberry [1977] who reports that, in a set covering problem, the subgradient method was faster than linear programming unless a

very high accuracy is required. This is compatible with the ultimate requirements for a real time process control system, where it is very important to get a good solution quickly, rather than to solve the problem optimally but outside the required system response time. Etcheberry continues that "the nature of the subgradient optimization algorithm eliminates the necessity of periodic reinversions or similar techniques (no numerical errors, no eta vectors, etc.) Anticycling procedures are not needed either. . . the previous results and considerations lead us to the conclusion that a subgradient iteration takes much less computational effort than a pivot iteration. (A subgradient optimization algorithm should also use less computer memory than an LP algorithm.)"

In some instances, such as Rosenwein [1986], the subgradient method is used first to get an approximate solution quickly, then a customized dual adjustment is used to "fine tune" the subgradient solution. In other cases, such as Kedia [1985], dual adjustment is used first, followed by the subgradient procedure to converge on the optimum. Dual adjustment heuristics tend to modify only one or a few variables at a time so that additional branch-and-bound nodes can be fathomed very quickly; a subgradient iteration may require more time than a dual adjustment procedure, but produce a better bound, and fathom nodes more quickly. The best choice of algorithm is highly problem specific.

Ryu [1993] applied Lagrangian relaxation to produce tight gaps in the Capacitated Plant Location Problem, Multi-Item Capacitated Lot Sizing Problem, Dynamic Production Scheduling Problem and Multilayer Plant Location Problem. Chajakis [1993] worked on a set of related problems in job shop scheduling, vehicle routing and forest management. His research is primarily concerned with the application of Lagrangian Substitution in the case where the integer subproblems have no specific structure but, nonetheless, yield very strong bounds. These subproblems are solved in reasonable time using a variety of standard approaches including "lifting" and double-contracting branching priorities. Ryu

and Chajakis establish that Lagrangian relaxation can be successful even if the subproblems do not possess any special structure, provided the subproblem bounds are strong enough. It is then worthwhile to solve the subproblems even if a general purpose branch and bound solver must be used.

In contrast, this research will show that Lagrangian relaxation can be successful even for relatively weak relaxations possessing the integrality property, provided highly specialized algorithms can solve the subproblems quickly enough, and if the ultimate solution is expected to be integral or near-integral.

4.4 A Dual Adjustment Procedure for Dynamic Car Scheduling

The Dynamic Car Scheduling algorithm is intended for real time routing of all shipments currently moving on the railroad. The process maintains a feasible and near-optimal set of freight car schedules, both satisfying integral flow constraints and respecting train segment capacity limits.

Since this is a real time process control application, rapid system response time is a key design consideration. Yet, it is still important to obtain a good solution — one with a tight duality gap — without having to take the time to utilize branch and bound to prove optimality in every case. Instead of branch and bound, a dual adjustment process will be proposed, as an adaptation of the successful approach to the Generalized Assignment Problem (GAP) by Fisher, Jaikumar and Van Wassenhove [1986]. The GAP asks what is the best way to assign “n” jobs to “m” servers, subject to knapsack constraints on each server. If one considers freight cars as “jobs” and trains as “servers”, then this analogy between the GAP and railroad car scheduling problem makes sense.

This research extends Fisher’s algorithm to Multi Commodity Network Flow problems. Previously, Kedia [1985] also extended Fisher’s GAP dual adjustment approach to set partitioning, set packing and set covering problems, and to a vehicle

routing/scheduling problem. Guignard and Rosenwein (see Rosenwein [1986], Guignard and Rosenwein [1989a],[1989b]) further extended Fisher's GAP approach by refining branching priorities, utilizing a subgradient method to strengthen the root node lower bound, adding a surrogate constraint to the subproblems to strengthen the bound, and devising a primal heuristic to aid in fathoming branches and reducing memory requirements. Later, Rosenwein [1991] applied Lagrangean decomposition in a similar manner to strengthen bounds in the constrained assignment problem.

The goal here is to develop a set of dual adjustment rules which consistently produce a tight gap and an intuitively satisfactory solution when the algorithm is operated on a "one shot" basis: that is, rapidly drive excess flows off overcapacity links and stop. Etcheberry [1977] suggests such rules should be simple and fast, going for fast execution per iteration even if it means more iterations overall. Keeping the rules simple has obvious advantages for the supportability and maintainability of computer code, as well.

A complicating factor for the design of this algorithm is coping with the degeneracy which tends to be present in the Multi Commodity Network Flow formulation of this problem. Powell [1989] discusses the phenomenon of degeneracy in linear network problems and the fact that such problems might have many alternative primal optimal solutions. Wardrop's [1952] principles of traffic equilibrium state that in any optimal solution, the cost of all utilized routes between the same origin and destination must be equal, and less than or equal to the cost of any unutilized route. Fukushima [1984] showed, using the Kuhn-Tucker conditions, that this amounts to the same thing as LP complementary slackness. For linear problems, Wardrop's requirement that costs be equilibrated on any utilized route implies a degenerate solution for any flows which must be split over more than one path

. This degeneracy, however, also extends to the dual problem. Powell showed that the dual variables are only subgradients, not gradients of the objective function. This means from time to time any dual algorithm will face a "cost allocation" decision — which will have to be decided using seemingly arbitrary "tie breaking" rules such as allocating all the cost increase to the upstream segment. Degeneracy is a concern in LP-type algorithms because it can lead to nonimproving pivots and large increases in execution time. Degeneracy's effect on the dual heuristic presented here is even more profound — if not effectively dealt with, degeneracy can cause the heuristic to enter an infinite loop, or to continue to make progress, but at an arbitrarily slow rate.

To counteract this looping tendency, the concept of a "tabu search" was adapted from the artificial intelligence literature. For each flow, a list of previously-tried saturated train segments is maintained. In *diversion cost* calculations, the flow is prevented from returning to any segments from which it had been previously been diverted. However, after link costs are adjusted, actual flow assignment is still performed on a true shortest-path basis. This tabu search approach provides a positive anti-cycling mechanism, but the resulting flow assignments still generally produce a tight duality gap.

4.4.1 Mathematical Properties of the Dual Adjustment Procedure

Ignoring for a moment complications introduced by degeneracy, the following gives a simplified overview of the Dynamic Car Scheduling algorithm:

- Initially, all dual variables are set to zero, and shipments are assigned onto the train network on a shortest path basis. This assignment will likely produce some over capacity train segments.
- Overcapacity segments are processed one at a time starting with the segments latest in time to "sweep up," or earliest in time to "sweep down." For each overcapacity segment "s" and each shipment "k" utilizing "s", the diversion cost δ_s^k is calculated by

blocking segment "s", recalculating the shortest path for each shipment, and taking the difference between revised and original path costs. Shipments assigned to each overcapacity segment are then rank ordered from lowest to highest diversion cost δ_s^k.

- Usually, the algorithm tries to reroute all excess cars, restoring feasibility to the segment's traffic assignment. In some cases, due to interaction with other overcapacity segments, the program may choose to divert fewer cars. Once the algorithm has determined how many cars to divert, the required dual variable increase, Δu_s, is easily found. To divert "k*" cars, the required dual variable increase is the diversion cost of the k*'th car, $\Delta u_s = \delta_s^{k*}$ determined by counting down the sorted list of shipments.

This positively drives off the segment any shipments having a diversion cost $\delta_s^k < \Delta u_s$, and produces an equal-cost diversion option for the k*'th shipment. This allows the k*'th shipment to be split, if necessary to exactly match the available capacity of segment "s" while rerouting the remaining cars to another path.

Recalling equation 4.2.2, we have a lower bound on the optimal objective function:

$$Z_{DU} = \underset{k \in K}{\text{Max}} \left\{ \underset{(i,j) \in A}{\text{Min}} \sum_{s \in S} \sum \varphi_{ij}^k x_{ij}^k + \sum u_s C_s \right\} \qquad (4.2.2)$$

where:

C_s = Capacity of train segment $s \in S$, in cars.

u_s = Dual variable associated with segment $s \in S$.

x_{ij}^k = Flow volume on link $(i,j) \in A$ for commodity $k \in K$. In the following discussion, without a loss of generality, each individual railcar comprises a single shipment, so $x_{ij}^k \in \{0,1\}$

c_{ij}^k = Actual cost of car $k \in K$ moving over link $(i,j) \in A$.

φ_{ij}^{k} = Adjusted cost of car k $\in$ K moving over link (i,j) $\in$ A.

$$\varphi_{ij}^{k} = c_{ij}^{k} + \sum_{s \in S_{ij}} u_s \text{ for all links (i,j)} \in A.$$

If over capacity segment "v" (for "violated") has been identified, v $\in$ S, the dual variable u_v is increased by Δu_v such that remaining flow after diversion exactly matches segment capacity, thus:

$$\sum_{k \in K} \sum_{(i,j) \in A_v} x_{ij}^{k} = C_V \quad \text{(After } \Delta u_v \text{ applied)} \qquad (4.4.1)$$

Only one segment at a time is adjusted, thus for all links (i,j) $\in$ A:

$$\varphi_{ij}^{k}{}_{(Before)} = \varphi_{ij}^{k}{}_{(After)} \qquad (i,j) \notin A_v \qquad (4.4.2)$$

and

$$\varphi_{ij}^{k}{}_{(Before)} + \Delta u_v = \varphi_{ij}^{k}{}_{(After)} \quad (i,j) \in A_v \qquad (4.4.3)$$

where A_v = The set of arcs (i,j) spanning segment v $\in$ S.

Our objective will be to show that increasing the dual variable in this manner will cause the lower bound will improve, under certain conditions. Define the improvement in the lower bound as:

$$\Delta Z_{DU} = Z_{DU\ (After)} - Z_{DU\ (Before)} \qquad (4.4.4)$$

Define the diversion cost for shipment "k" diverted from segment "v" as δ_v^k, as before, where the adjusted cost $\varphi_{ij}^{k}{}_{(Less\ "v")}$ of all links (i,j) $\in$ A_v is temporarily reset to infinity. The adjusted cost of other links (i,j) $\notin$ A_v are not touched, therefore, as in

equation 4.4.2, $\varphi_{ij}^{k}{}_{(\text{Less "v"})} = \varphi_{ij}^{k}{}_{(\text{Before})} = \varphi_{ij}^{k}{}_{(\text{After})}$ for $(i,j) \notin A_v$.

$$\delta_v^k = \sum_{(i,j)\in A} \left(\varphi_{ij}^{k}{}_{(\text{Less "v"})} x_{ij}^{k}{}_{(\text{After})} - \varphi_{ij}^{k}{}_{(\text{Before})} x_{ij}^{k}{}_{(\text{Before})} \right) \quad (4.4.5)$$

Define the set of shipments utilizing segment "v" as $K_v \subset K$ where:

$$K_v = k \in \{ K \mid x_{ij}^k > 0, (i,j) \in A_v \}$$

Partition K_v into two disjoint subsets based on δ_v^k:

$$K_v^1 = k \in \{ K_v \mid \delta_v^k < \Delta u_v \}, \; K_v^2 = k \in \{ K_v \mid \delta_v^k \geq \Delta u_v \}$$

Any shipment $k \in K_v^1$ having $\delta_v^k < \Delta u_v$ will be positively driven off of segment "v" and any associated links A_v. However, since segment "v" is the *only* segment which has had its cost adjusted this iteration, it will be true that:

$$\varphi_{ij}^{k}{}_{(\text{Before})} = \varphi_{ij}^{k}{}_{(\text{After})} \qquad k \in K_v^1 \qquad (4.4.6)$$

Any shipment $k \in K_v^2$ having $\delta_v^k \geq \Delta u_v$ will remain on its original path, thus:

$$x_{ij}^{k}{}_{(\text{Before})} = x_{ij}^{k}{}_{(\text{After})} \qquad k \in K_v^2 \qquad (4.4.7)$$

Substituting identities (4.4.6) and (4.4.7) into the definition of ΔZ_{DU} gives:

$$\Delta Z_{DU} = \sum_{k \in K_v^2} \sum_{(i,j)\in A} \varphi_{ij}^{k}{}_{(\text{After})} x_{ij}^{k}{}_{(\text{Before})} + \sum_{k \in K_v^1} \sum_{(i,j)\in A} \varphi_{ij}^{k}{}_{(\text{Before})} x_{ij}^{k}{}_{(\text{After})}$$

$$- \sum_{s\in S} u_{S(\text{After})} C_S - \sum_{k \in K_v^2} \sum_{(i,j)\in A} \varphi_{ij}^{k}{}_{(\text{Before})} x_{ij}^{k}{}_{(\text{Before})}$$

$$- \sum_{k \in K_v^1} \sum_{(i,j)\in A} \varphi_{ij}^{k}{}_{(\text{Before})} x_{ij}^{k}{}_{(\text{Before})} + \sum_{s\in S} u_{S(\text{Before})} C_S$$

Reorganizing terms:

$$\Delta Z_{DU} = \sum_{k \in K_{V2}} \sum_{(i,j) \in A} \left(\varphi_{ij}^{k}{}_{(After)} - \varphi_{ij}^{k}{}_{(Before)} \right) x_{ij}^{k}{}_{(Before)}$$

$$+ \sum_{k \in K_{V1}} \sum_{(i,j) \in A} \varphi_{ij}^{k}{}_{(Before)} \left(x_{ij}^{k}{}_{(After)} - x_{ij}^{k}{}_{(Before)} \right)$$

$$- \sum_{s \in S} \left(u_{s(After)} - u_{s(Before)} \right) C_s \qquad (4.4.8)$$

Examining the first term of (4.4.8), recall that:

$$\varphi_{ij}^{k}{}_{(Before)} = \varphi_{ij}^{k}{}_{(After)} \qquad (4.4.2)$$

everywhere except where $(i,j) \in A_v$, then:

$$\varphi_{ij}^{k}{}_{(After)} - \varphi_{ij}^{k}{}_{(Before)} = \Delta u_v \qquad (4.4.3)$$

So all terms of the summation cancel out except where $(i,j) \in A_v$, leaving:

$$\sum_{k \in K_{V2}} \sum_{(i,j) \in A_v} \Delta u_v \; x_{ij}^{k}{}_{(Before)}$$

If all the excess cars are diverted, without "overcorrecting" by diverting too many cars, then remaining flow after diversion will exactly match segment capacity and identity (4.4.1) can be applied. This first term then further reduces to simply:

$$+ \Delta u_v \; C_v$$

The second term of (4.4.8) includes only those flows which are diverted off the overcapacity segment "v", $k \in \{ K_v \mid \delta_v^k < \Delta u_v \}$, for this subset $\varphi_{ij}^{k}{}_{(Before)} = \varphi_{ij}^{k}{}_{(After)}$

holds, so the definition of diversion cost δ_v^k given in (4.4.5) can be applied. The second term reduces to:

$$\sum_{k \in K_{v1}} \delta_v^k$$

Examining the third, and final term of (4.4.8), the only dual variable which has been adjusted is the one which pertains to segment "v", so all the other terms of the summation cancel out. The third term reduces to:

$$- \Delta u_v \, C_v$$

Reassembling the three terms, the first and third term of (4.4.8) cancel out, giving:

$$\Delta Z_{DU} = \sum_{k \in \{ K_v \mid \delta_v^k < \Delta u_v \}} \delta_v^k \qquad (4.4.9)$$

Define the actual flow assigned to segment $s \in S$:

$$F_S = \sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k \text{ (After)} \qquad (4.4.10)$$

Then, more generally, if $F_v \neq C_v$ (condition 4.4.1 does *not* have to hold true):

$$\Delta Z_{DU} = \sum_{k \in \{ K_v \mid \delta_v^k < \Delta u_v \}} \delta_v^k + \Delta u_v \, (F_v - C_v) \qquad (4.4.11)$$

As diversion cost δ_v^k is defined in (4.4.5), it must be a nonnegative number, assuming all the shipments are correctly assigned to their true shortest paths to begin with. If $F_v \geq C_v$ and $\Delta u_v > 0$,this establishes the proposition that applying the dual adjustment

rule will produce a monotonic improvement in the lower bound, under the following conditions:

1. At least one $\delta_v^k > 0$ resulting in $\Delta u_v = \delta_v^{k*} > 0$; it must not be a "degenerate" or zero cost diversion to an alternate optimal path. A diversion having $\delta_v^k = 0$ might improve primal feasibility, but will not improve the lower bound.
2. The lower bound will be improved so long as

$$F_V = \sum_{k \in K_V} \sum_{(i,j) \in A_V} x_{ij}^k \geq C_V \text{ (After } \Delta u_v \text{ applied)} \qquad (4.4.12)$$

   if Δu_v is "overcorrected" and too many cars are driven off ,the lower bound will be reduced by Δu_v for each car by which $F_v < C_v$. Since necessarily $\Delta u_v > \delta_v^k$ for every diverted flow "k", the net impact of "overcorrection" on the lower bound must be negative.
3. All the shipments must have been correctly assigned on a shortest path basis to begin with, and this shortest path assignment must be maintained at the conclusion of every iteration. Otherwise, the diversion cost calculation might yield a negative cost. Besides, it is a requirement of the Lagrangian Relaxation lower bound calculation that all shortest path subproblems must be solved to optimality upon the conclusion of every iteration.

As a practical matter, the Dynamic Car Scheduling algorithm gains efficiency by reprocessing flows on an exception basis rather than having to re-calculate every shortest path subproblem at each iteration. By limiting cost adjustments Δu_v to *increases only* and

never reducing dual prices, only flows currently assigned onto the current overcapacity segment "v", and having a diversion cost $\delta_v^k < \Delta u_v$ must be reexamined to satisfy condition (3). These flows are identified at the same time the dual price increase Δu_v is found. This is because an increase to a link's cost can never attract flow not already on the link; it can only drive off traffic.

Given this limitation on price adjustments $\Delta u_v \geq 0$, it is critical to avoid overcorrecting the dual variables u_s. Once a dual variable has been overcorrected, the algorithm has no way to recover its mistake, since cost reductions are not permitted. Overcorrection is not a fatal error, but might lead to a larger than necessary duality gap.
The tabu search procedure seems to find a "happy medium" between too-small corrections, which can lead to looping or very long solution times; and too-large corrections, which may lead to an excessively large duality gap, but it may temporarily violate conditions (2) and (3) during the process.

The dynamic car scheduling procedure continues to increase segment prices and split flows until a primal feasible solution is obtained, or until a maximum iteration limit is reached. This primal feasible solution also satisfies the Lagrangian Relaxation requirement that all subproblems be solved to optimality, so both the primal objective function and lower bound calculations are valid for this solution. A separate Lagrangian Heuristic is not needed to obtain a primal feasible solution. The restriction on price adjustments $\Delta u_v \geq 0$ may cause this algorithm to converge to a point other than the optimal solution. However, the lower bound can be used to measure how far any given solution is from optimality.

4.4.2 Simple Examples of Dual Adjustment Procedure

Figures 4.2, 4.3 and 4.4 introduce the dual adjustment heuristic, and demonstrate the function of the tabu list in diversion cost calculations and flow assignments. The

procedure starts by assigning all traffic on a shortest path basis, then each overcapacity train segment is processed in turn, increasing the dual variables to drive off excess flow.

Given that a shipment has previously been driven off an overcapacity segment, it is highly unlikely (but not impossible) that this same assignment will be part of an optimal solution. The tabu list maintains a list of all previous overcapacity segments utilized by each shipment, and prevents the consideration of those segments in any future diversion cost calculations. Without the tabu list, once path cost has been equilibrated on two or more routes, subsequent assignments would simply alternate among equal cost paths, each time computing a diversion cost of zero, but never resolving the overcapacity condition. The tabu list is used as a method of calculating a positive diversion cost for a shipment and increasing the dual variables, under circumstances when no cost increase would normally be applied. However, in actual flow assignment, the tabu list is only used as a tie-breaker among equal cost routes.

In all three examples, the following conditions hold:

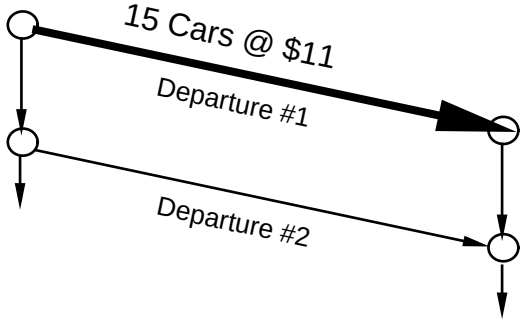
- Each train movement link has a maximum capacity of 10 cars. Yard inventory links are uncapacitated.
- Transit time is 11 hours. Trains depart once a day.
- Equipment is valued at $1 per hour, or $24/day. Train operating costs are the same regardless of which departure is used, so the objective function is calculated below based only on the hourly equipment cost component.

Figure 4.2 shows a case where 15 cars must be moved, but train capacity is only 10 cars. All shipments are initially assigned on a shortest path basis, producing an infeasible solution, but a valid lower bound Z_{DU} = $165. By inspecting shipments using departure #1, we find an increase of $24 to the cost of departure #1 would be sufficient to drive off the excess 5 cars, by equilibrating the path cost with departure #2. Using equation 4.4.9,

Fig 4.2: Simple Example of Dual Adjustment Heuristic

Infeasible Starting Solution:

Z_{DU} = $ 165

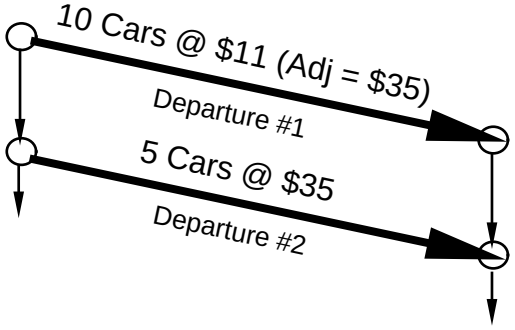


ITERATION 1

Increase the Link Cost of Departure #1 by $24.

Then split the shipment, reassigning the last 5 cars onto Departure #2.

Feasible Solution



Calculations:

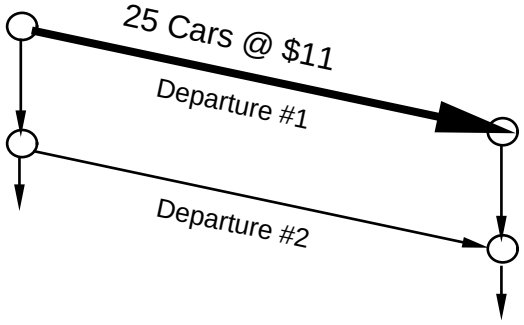
Z* = 10 ($11) + 5 ($35) = $285
Z_{DU}= 10 ($35) + 5 ($35) - 10 ($24) = $285

Gap = 0.0 %

Fig 4.3: More Difficult Example, with "Pooling"

Infeasible Starting Solution:

Z_{DU} = $ 275



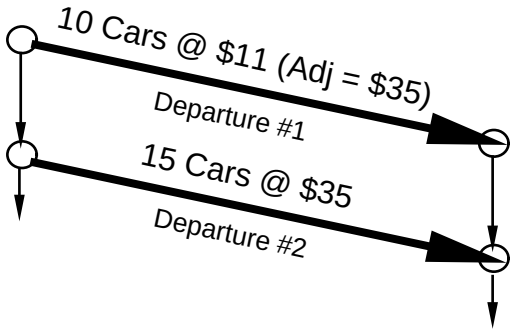
ITERATION 1

Increase the Link Cost of Departure #1 by $24.

Then split the shipment, reassigning the last 15 cars onto Departure #2.

Z_{DU}= $635

= 25 ($35) - 10 ($24)



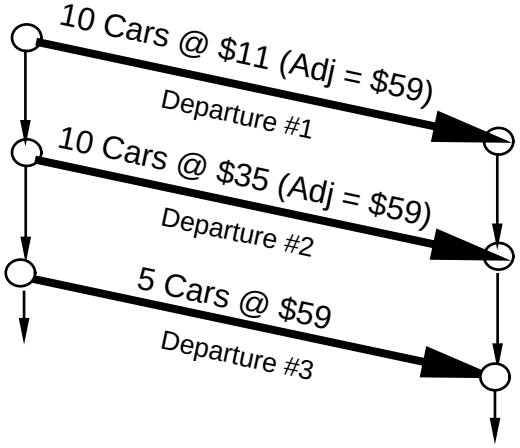
ITERATION 2

Departure #2 overflows, so simultaneously increase the cost of departures #1 and #2 by $24.

Then split the shipment again, reassigning the last 5 cars onto Departure #3.

Fig 4.3 (ctd)

Feasible Solution:



Calculations:

Z* = 10 ($11) + 10 ($35) + 5 ($59) = $755
Z_{DU}= 25 ($59) - 10 ($48) - 10 ($24) = $755

Gap = 0.0 %

Fig 4.4: Difficult Example, with "Tabus"

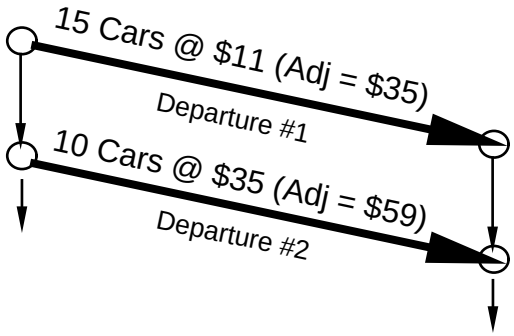
The Initial Solution and Iteration 1 are the same as in Fig 4.3.

ITERATION 2

Apply a cost increase of $24 on departure #2 to equilibrate path costs with departure #3. Then reassign only the 5 overflow cars on a *true shortest path basis*. These five cars now return to their *original* path via Departure #1, not utilize Departure #3 as intended.

Fig 4.4 (ctd)

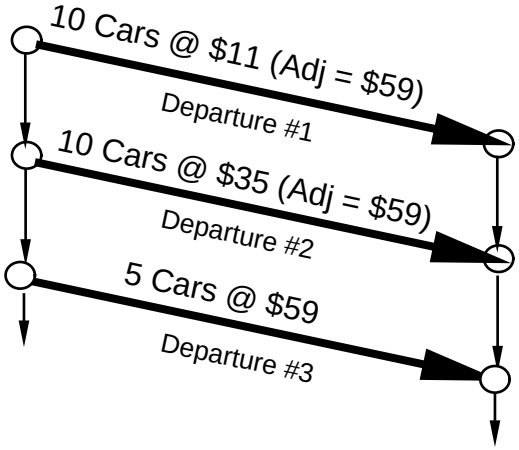
Dual Calculation is Invalid, solution is still infeasible



ITERATION 3

There are now 5 cars excess on departure #1, with tabus on both departures #1 and #2. Apply a cost increase of $24 on departure #1 to equilibrate costs with departure #3. Using tabus on both departures #1 and #2 as a tie-breaker, reassign the 5 cars excess to departure #3 on a *true shortest path basis*.. This produces a feasible, optimal solution, the same as obtained in Figure 4.3.

Feasible Solution with Valid Lower Bound



Calculations:

Z* = 10 ($11) + 10 ($35) + 5 ($59) = $755
Z_{DU}= 25 ($59) - 10 ($48) - 10 ($24) = $755

Gap = 0.0 %

this adjustment should improve the lower bound by (5 units)($24/unit) = $120. It produces a feasible, optimal solution having $Z^* = Z_{DU} = \$285$.

Figure 4.3 shows a more difficult example where 25 cars must be moved. Initial shortest path assignment produces $Z_{DU} = \$275$. In the first iteration, increasing the cost of departure #1 by $24 equilibrates costs with departure #2 and improves Z_{DU} by (15 units)($24/unit) = $360. The new $Z_{DU} = \$635$. Diversion of the 15 excess cars repairs departure #1, but now causes departure #2 to overflow.

By inspection, the obvious solution is to allow these excess 5 cars to overflow onto departure #3. To accomplish this requires a *simultaneous increase* of $24 applied to the dual variables on both departures #1 and #2. This three-way equilibration of path cost in the second iteration further improves the lower bound by (5 units)($24/unit) = $120 to $755, yielding a feasible, optimal solution, and a monotonically improving lower bound at each iteration.

Unfortunately, in real-world problems, the required simultaneous adjustments are not so simple to identify. Conceptually, however, the proposed tabu search procedure could be considered an "artificial intelligence" approach to identification of those segments which need to have simultaneous cost adjustments applied.

However, the tabu search procedure actually applies adjustments only on a one-segment-at-a-time basis. The excess flow, when reassigned after each adjustment, naturally tends to seek out and identify the next segment requiring a simultaneous cost adjustment. This means that some flow assignments may be temporarily "out of equilibrium" while other dual variables continue to be adjusted. These flows are *not* reassigned immediately, because the algorithm anticipates that subsequent dual variable adjustments will probably correct this "out of equilibrium" condition before the algorithm terminates. However, such flows are marked as "revisit" candidates, and are reinspected

at the *end* of the procedure, to either verify they are already on shortest path assignments, or else to return the flows to true shortest path assignments. Each segment adjustment corresponds to an "inside iteration", and this process continues until either a feasible solution is attained or the maximum "inside iteration" limit is reached.

If, at the end of the procedure, it is necessary to correct some "out of equilibrium" assignments, and this correction produces new segment capacity constraints violations, the dual adjustment procedure is called again. Each major cycle through the dual adjustment procedure is called an "outside iteration." The process continues until either a feasible solution is attained with all flows on valid shortest path assignments, or until the "outside iteration" limit is reached.

Figure 4.4 shows an example of the tabu search procedure at work. The initial shortest path assignment, and first iteration of the algorithm are identical to Figure 4.3, producing an assignment of 10 cars to departure #1 and 15 cars to departure #2, having Z_{DU} = \$635. Departure #2 is now over capacity by five cars. At this point, a "normal" diversion cost calculation would identify a zero cost opportunity to return the excess five cars back to departure #1, and the algorithm would enter an infinite loop.

However, since those five cars were previously *diverted from* departure #1, the tabu list would disallow the use of that path in the diversion cost calculation. The next-best opportunity would be to reroute the flow via departure #3, which would require a cost increase of \$24 on departure #2 to equilibrate path costs.

Applying this \$24 increase in iteration 3, the 5 cars overflow is now reassigned on a *true shortest path* basis. But since the cost of departure #1 has not yet been increased, these five cars will now return to departure #1, not to departure #3 a intended. Also, note the 10 cars still remaining on departure #2 are now "out of equilibrium" because departure

#1 has a lower current cost. Therefore, the lower bound calculation is no longer valid as long as this "out of equilibrium" condition exists.

At the end of iteration 3, there is a 5 car excess on departure #1, with tabus still in effect on both departures #1 and #2. The diversion cost calculation is then based on the use of departure #3 resulting in a $24 increase being applied to departure #1 in iteration 4. Now reassigning the 5 excess cars on a *true shortest path* basis, all three departures have identical path costs. However, the tabu list can be used as a tie-breaker, resulting in the selection of departure #3 for this assignment. This produces a feasible solution at the end of iteration #4.

Revisiting the 10 cars on departure #2 which were labeled as "out of equilibrium" during iteration #3, we now verify (after the $24 increase to departure #1) these cars are once again on a shortest path assignment, so no further action is necessary. The lower bound calculation establishes this is an optimal solution having $Z^* = Z_{DU} = \$755$, the same result obtained in Figure 4.3.

4.4.3 Avoiding Overcorrection of the Dual Variables

"Overcorrection" is the condition where in a feasible solution, for any segment $s \in S$, $u_s > 0$ while simultaneously $\sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k < C_s$.

This violates linear programming "complementary slackness" optimality conditions requiring either that $u_s = 0$ if $\sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k < C_S$, or else if $\sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k = C_S$, then $u_s \leq 0$.

As the examples of the previous section show, price adjustments on parallel "side by side" links can display a considerable degree of interaction with each other. However,

adjustments on "upstream" and "downstream" segments can also interact. This reflects another form of degeneracy in the mathematical formulation of this problem.

The search strategy employed by the dual adjustment heuristic is to "sweep up" making corrections in the latest time periods first. For example, in Figure 4.5, segment "B_1" will be visited first, then "A_1". To counter the myopic tendencies of this unidirectional "sweep" algorithm, a "look ahead" feature has been installed. One might not want to divert *all* the excess cars on a "downstream" link, because a subsequent adjustment to an "up-stream" overcapacity link could possibly repair the violation. Describing this procedure, as well as exploring the effect of different sweep strategies, will be the main focus of this section.

Figure 4.5 shows how, once an overcapacity segment "v" has been identified, each shipment using that segment is traced back to its origin (or current actual location) to determine how much of the overflow is *jointly* involved with an upstream over capacity segment "s". The strategy is to divert only as much flow from the downstream segment as is "safe." The maximum violation to segment "v" that can be repaired by the upstream correction to segment "s" is the smaller of the total violation on the upstream segment "s", or the total number of shipments jointly involved with both "v" and "s".

Define:

Ovf_s = The number of cars by which segment $s \in S$ has overflowed:

$$Ovf_s = C_S - \sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k \qquad (\text{when } C_S > \sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^k)$$

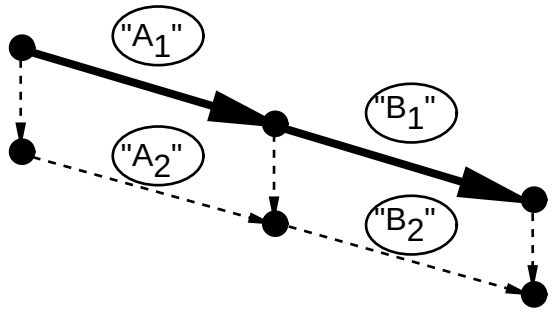
$$= 0 \qquad \text{Otherwise}$$

K_S = The set of shipments utilizing segment "s" as $K_S \subset K$, as before:

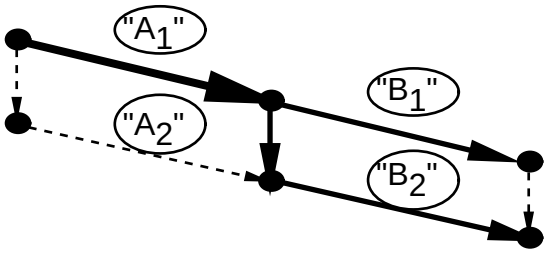
$$K_S = \{ k \in K \mid x_{ij}^k > 0, (i,j) \in A_S \}$$

Fig. 4.5: Avoiding an Overcorrection

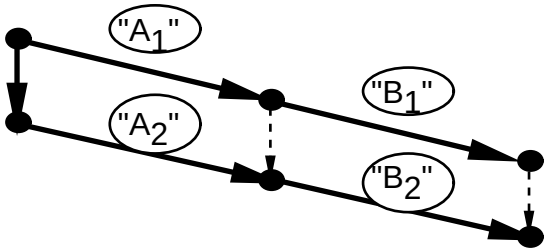
Original Assignment: Both "A_1" and "B_1" Over Capy



Price increase to downstream segment "B_1" drives excess flow to "B_2" but leaves "A_1" still over capacity



Applying increase to upstream segment "A_1" fixes both "A_1" and "B_1." Increase to "B_1" should have been deferred.



Jt_{sv} = The subset of cars using both upstream segment "s" and the current targeted overcapacity segment "v":

$$Jt_{sv} = K_s \cap K_v$$

Then the potentially "repairable" violation on segment "v" is:

$$Ovf_v^{(repairable)} = \min \{Ovf_s , \sum_{k \in Jt_{sv}} \sum_{(i,j) \in A_v} x_{ij}^k \} \qquad (4.4.13)$$

This “repairable” violation is subtracted from the total violation to yield the target correction amount. The remaining “non repairable” violation on segment “v”, which should be corrected at once, is:

$$Ovf_v^{(\text{non repairable})} = Ovf_v - Ovf_v^{(\text{repairable})} \qquad (4.4.14)$$

This diversion still satisfies required condition (4.4.12) to guarantee an improving lower bound at each iteration, that is, since

$$Ovf_v^{(\text{non repairable})} \leq Ovf_v$$

reducing the overflow amount by $Ovf_v^{(\text{non repairable})}$ will not “overcorrect” the dual variable u_v. Having reduced flow on “v” by $Ovf_v^{(\text{non repairable})}$, the algorithm’s strategy is then to wait and see if subsequent adjustment to the upstream segment “s” will also take care of the remaining overcapacity problem on “v.” If, after “s” is repaired, the overcapacity condition on “v” still persists, then “v” can be revisited and the remaining cars diverted directly. Otherwise, the procedure risks diverting too many cars, leaving “v” overcorrected in the final solution. A more formal statement of the look ahead algorithm will be provided in the next section.

For example, in Figure 4.5, suppose that only 3 cars are involved with upstream segment “A_1”, but that “B_1” has overflowed by ten cars. It is safe to immediately divert up to 10 - 3 = 7 cars from B_1, but then wait and see if the remaining 3 cars overflow will be repaired by the subsequent adjustment to A_1.

The assumption in Figure 4.5 is that the two overcapacity segments “A_1” and “B_1” belong to different trains. In this case, the shipment must be handled at the intermediate terminal, regardless of whether the outbound connection is today’s or tommorrow’s train.

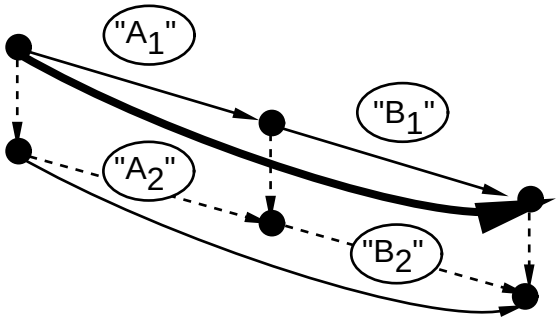
Then tie-breaking logic in the shortest path algorithm will favor the earliest possible departure from the origin terminal — that is, path A_1-B_2 will be utilized provided:

- B_1 is full
- Sufficient capacity exists on A_1 to accomodate the shipment
- The dual price on A_1 has not been increased

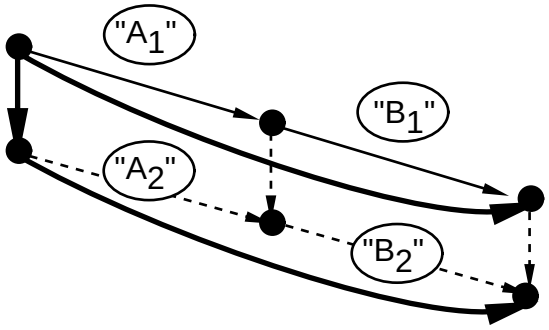
If both segments are part of the same train, as in Figure 4.6, the shipment will be held at the origin terminal so it can be forwarded on the “thru” block on the next day’s train. This “thru” block bypasses the handling at the intermediate terminal. If the cost of segment B_1 is increased, the shipment will divert immediately to path A_2-B_2, and the hybrid path A_1-B_2 would be utilized only in unusual conditions.

Fig. 4.6: Diverting with Thru Blocks

Original Assignment: Both "A_1" and "B_1" Over Capacity

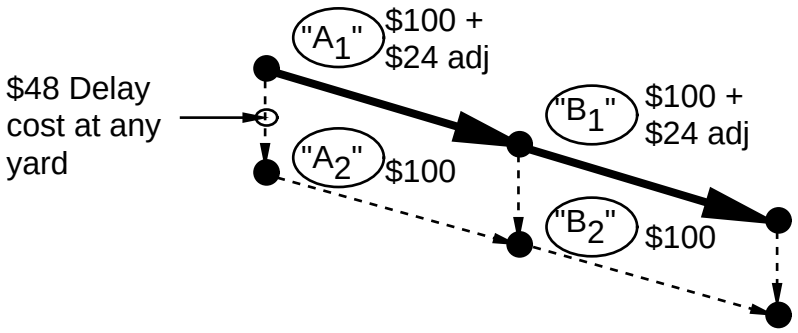


Applying increase to downstream segment "B_1" drives excess flow to the next day's train, holding cars at origin terminal in order to utilize the "thru" block.



Even where two separate trains are used, it is frequently the case that "last train out" assignment will be employed. This is a result of the algorithm's requirement (so the lower bound remains valid) that all assignments must remain optimal with respect to "adjusted" path costs. The reason for this is shown in Figure 4.7.

Fig. 4.7: Original vs Adjusted Path Costs



Path	Original	Adjusted
A1-B1	$200	$248
A1-B2	$248	$272
A2-B2	$248	$248

In Figure 4.7, even though path costs A_1-B_1 and A_2-B_2 are equilibrated, A_1-B_2 costs more on an adjusted cost basis, so it will not be used. This behavior occurs as a result of applying the $24 price increase onto link A_1. If the total $48 increase had been applied onto B_1 instead, then all three paths would have had the same adjusted cost, so A_1-B_2 could be used. Appreciating this "last train out" behavior is critical to comparing the performance of various "sweep direction" and "look ahead" alternatives, which will be further explored in Figures A.1-A.4 in the Appendix. All four possible permutatations of these approaches will be explored, assuming that flow diverts as described above when segment dual prices are increased.

In the “Sweep Up” approach, the deepest or latest train segments are visited first; in “Sweep Down”, the shallowest or earliest train segments are visited first. However, as soon as any diversion is made, the “sweep” immediately restarts at the beginning, so that if a train segment was bypassed earlier, that segment is immediately reexamined.

Equations 4.4.13 and 4.4.14 are applied to limit the initial flow diversion when “look ahead” is turned on. Look ahead always operates in the same direction as the sweep. If “Sweep Down” is used, then “downstream” segments are used to limit diversion instead of “upstream” segments (which would have already been visited and corrected.)

The test problems presented in Figures A.1-A.4 in the Appendix incorporate asymmetric traffic volumes and diversion costs, and are constructed by swapping flow volumes and diversion costs to present them in different combinations. The problems examine the performance of each of the four sweep direction and look ahead combinations, under varying circumstances likely to be encountered in actual practice. These results, summarized in Figure 4.8, show an “X” in each cell where the optimal solution was found.

Figure 4.8 - Car Scheduling Algorithm : Performance Comparison

Figure #	Sweep Up	Sweep Up w/Look Ahd	Sweep Dn	Sweep Dn w/Look Ahd
A.1		X		
A.2				X
A.3	X	X		X
A.4		X	X	X

This analysis clearly shows the positive impact of including the look ahead logic which solves 3 out of 4 problems correctly, as compared to only 1 out of 4 without look ahead. However, in this "static" problem, the outcome with respect to sweep direction is ambiguous, since either direction performs the same. The impacts of sweep direction and look ahead will be further explored in Section 4.4.7 in the context of rolling horizon simulation testing.

4.4.4 Formal Definition of the Dynamic Car Scheduling Algorithm

Initially, all shipments $k \in K$ are assigned onto the train schedule network on a shortest path basis. When new shipments are called in, these are also initially assigned on a shortest path basis with respect to the current adjusted link prices, φ_{ij}^{k}. The current trip plan for each shipment is stored in $Path_k$, consisting of a set of links from shipment origin to destination, in route sequence order. When a new shipment is added, $Path_k$ is immediately scanned to determine if any segment capacities have been violated. If any overcapacity segment "v" is found, the Dynamic Car Scheduling algorithm is called to restore feasibility.

Each shipment $k \in K$ has a tabu list $Tabu_k$. Each time a shipment is diverted from an overcapacity train segment "v", an audit trail is maintained by adding all links $(i,j) \in A_v$ traversing segment "v" into $Tabu_k$. The tabu list stores network links rather than segments themselves for efficiency reasons: tabus can be directly matched to network links prior to running the shortest path routine, which avoids having to expand one-to-many associations $(i,j) \in A_v$ each time the shortest path subroutine is called. This tabu list serves as an anti-cycling mechanism to help prevent the algorithm from entering a looping condition.

Each shipment $k \in K$ also has an indicator variable $revisit_k$, which indicates whether that shipment (as a result of tabu links being excluded from a diversion cost calculation) may not be on a true shortest path assignment with respect to the current dual

variables. Initially, $revisit_k$ is set to 0 indicating the shortest path assignment is valid, but $revisit_k = 1$ indicates the assignment should be reexamined in the next "Outside Iteration."

Main "Outside Iteration" Loop

Step 1:

Set Outside Iteration Counter: Outside_Iter = 0;

Step 2:

Increment Outside_Iter by 1; If Outside_Iter exceeds the outside iteration limit, then terminate. Note that the previous feasible solution is still valid and the rolling horizon simulation can still proceed; only the lower bound would be invalid, so the duality gap could not be measured.

Call the "Inside Iteration" Loop to restore a feasible solution.

Step 3:

For all flows $\{k \in K \mid revisit_k = 1\}$ recalculate the shortest path and reassign flow "k", then go to Step 2. If all $revisit_k = 0$, then terminate with a primal feasible solution and a valid lower bound.

Main "Inside Iteration" Loop

Step 1:

Set Inside Iteration Counter: Inside_Iter = 0;

Step 2:

Increment Inside_Iter by 1; If the maximum Inside Iteration limit has been exceeded, report "Infeasible Solution" and terminate.

Find the next segment "v" to visit, having a violated train capacity constraint.
For a "Sweep Up" direction, start at the segment having the latest departure time and walk "up" the sorted linked list of segments in reverse chronological order. For a "Sweep Down" direction, start at the earliest segment and walk "down" the list until the first segment "v" is encountered having:

$$Ovf_v > 0, \text{ where } Ovf_v = C_V - \sum_{k \in K} \sum_{(i,j) \in A_v} x_{ij}^k \quad (\text{when } C_V > \sum_{k \in K} \sum_{(i,j) \in A_v} x_{ij}^k)$$

$$= 0 \qquad \text{Otherwise}$$

If the overcapacity segment identified in the previous Inside Iteration had not been fully corrected (due to an "Excess Reduction" having been performed) then hold the segment pointer at that same segment rather restarting the scan at the top or bottom of the list. In many instances, returning to this same segment will allow the algorithm to complete the partial correction which it had started in the previous iteration, rather than starting from scratch each time and possibly "undoing" some of its previous work.

If no violated capacity constraints are found, report the duality gap and exit the inside iteration loop; otherwise, if a violated capacity constraint is found, determine the number of units to be diverted using equation 4.4.14. Perform:

$$Ovf_v^{(\text{non repairable})} = \text{Excess\_Reduction}(v)$$

If $Ovf_v^{(\text{non repairable})} > 0$ then take positive action now to reduce the volume on segment "v" by an amount $Ovf_v^{(\text{non repairable})}$ — go to Step 3. Otherwise, if $Ovf_v^{(\text{non repairable})} = 0$ then continue searching in the same sweep direction looking for the next overcapacity segment. Note that when the procedure returns from Step 3 back to the beginning of Step 2, the sweep restarts from the top or bottom of the train segment list, it does not continue where it left off.

Step 3:

For segment "v" Perform:

Δu_v = Cost_Adjustment(v);

Update link prices for all arcs (i,j) $\in A_v$:

$$\varphi_{ij}^{k}{}_{(new)} = \varphi_{ij}^{k}{}_{(old)} + \Delta u_v .$$

Flow_Diversion(v,Δu_v);

and return to the beginning of Step 2.

Excess_Reduction(v): returns Ovf_v(non repairable)

Step 1:

The first step is to calculate capacity overflows on upstream segments (or downstream, for "Sweep Down"). Walk through the linked list of segments, starting at the next segment in the current sweep direction past the current overcapacity segment "v", and compute for each upstream (or downstream) segment "s":

$$Ovf_s = C_S - \sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^{k} \qquad (\text{when } C_S > \sum_{k \in K} \sum_{(i,j) \in A_s} x_{ij}^{k})$$

$$= 0 \qquad \text{Otherwise}$$

It is not necessary to perform this calculation for the segments in the downstream (or upstream, for "Sweep Down") direction since those segments won't be scanned by the subsequent look ahead logic.

Initialize $Ovf_v^{(repairable)} = 0$.

Step 2:

The next step is to identify each flow currently assigned to overcapacity segment "v" and trace it in the Sweep direction to identify any interactions with other overcapacity segments. If any such interactions are identified, the volume of flow to be diverted from segment "v" is reduced by the number of cars in shipment "k", but not to exceed the overflow on segment "s".

If the end of the list has been reached and no more flows exist, go to Step 4.

For each flow $k \in K_v$ = The set of shipments utilizing segment "v" as $K_v \subset K$:

$$K_v = \{ k \in K \mid x_{ij}^k > 0, (i,j) \in A_v \}$$

Each shipment "k" has a "trip plan" which consists of a set of links to which each shipment is currently scheduled. Define, for each shipment $k \in K_v$:

$Path_k \subset A = \{ (i,j) \mid x_{ij}^k > 0 \}$ Ordered in route sequence, by increasing time

If the sweep direction is "Up", then for each shipment $k \in K_v$, starting at the first scheduled link (current position) of the shipment , scan the path until the first overcapacity segment "s" is encountered. For each arc $(i,j) \in Path_k$, check all segments $s \in S_{ij}$ (as defined in Section 3.3) to see if any underlying segments are over capacity. If any overcapacity segments $s \in S_{ij}$ having $Ovf_s > 0$ are found in $Path_k$, then set:

s = The first overcapacity segment $s \in S_{ij}$ encountered, having $Ovf_s > 0$

This scan identifies the first joint overcapacity segment which is *upstream* in shipment "k's" path from target segment "v". If arc s = v is encountered along $Path_k$, then the target overcapacity segment "v" has been reached, the scan terminates and no reduction related to this flow need be performed.

If the sweep direction is “Down” then the same process can be “jump started” by scanning the current path until current arc s = v is encountered. The pointer to current arc $(i,j) \in A_v$ is repositioned to the next arc *after* the arc which includes segment “v”, and the scan continues from that point downward to the end of $Path_k$. If no further overcapacity segments are found, then no reduction related to this flow need be performed.

Step 3:

If Step 3 is reached, then target segment “v”, joint segment “s” and shipment “k” have been identified which traverses both overcapacity segments. For each shipment “k” and joint segment “s”, the smaller of the flow volume of shipment “k” across segment “s” $\sum_{(i,j)\in A_s} x_{ij}^k$, or remaining overflow Ovf_s of segment “s” is considered repairable:

Increment $Ovf_v^{(repairable)}$ by $\min \{Ovf_s, \sum_{(i,j)\in A_s} x_{ij}^k\}$

Then since a given overflow can only be corrected once, eliminate double counting:

Decrement Ovf_s by $\min \{Ovf_s, \sum_{(i,j)\in A_s} x_{ij}^k\}$

Return to Step 2 and select the next flow.

Step 4:

Equation 4.4.14 can be applied to compute $Ovf_v^{(non\ repairable)}$:

$$Ovf_v^{(non\ repairable)} = Ovf_v - Ovf_v^{(repairable)} \qquad (4.4.14)$$

Return to the main “Inside Iteration” loop with the value of $Ovf_v^{(non\ repairable)}$

Cost_Adjustment(v): returns Δu_V

Step 1:

For each flow $k \in K_v$, do:

(Preventing use of overcapacity segment "v", also preventing use of any segments contained in the "tabu list" for shipment "k": recalculate the shortest path for each shipment to compute the diversion cost δ_V^k as the difference between current and revised path costs.)

Step 2:

Rank order all flows $k \in K_v$ from lowest to highest δ_V^k.

Step 3:

Partition K_v into two disjoint subsets K_v^1 and K_v^2 by counting down $Ovf_v^{(\text{non repairable})}$ cars. Move the first $Ovf_v^{(\text{non repairable})}$ cars into K_v^1, then all remaining cars into K_v^2. It might be necessary to "split" a shipment which falls on the boundary of K_v^1 and K_v^2 into two separate shipments in order to get the car counts to match precisely.

Set $revisit_k = 1$ for all $k \in K_v^2$ having $Tabu_k \neq \varnothing$

Step 4:

Now determine the required cost adjustment:

$$\Delta u_V = \delta_V^k \text{ where } k = \text{the last shipment moved into } K_v^1$$

Setting Δu_V in this manner partitions K_v into K_v^1 and K_v^2 based on Δu_V, so that it now conforms to the following definition:

$$K_v^1 = k \in \{ K_v \mid \delta_v^k \leq \Delta u_v \}, \; K_v^2 = k \in \{ K_v \mid \delta_v^k \geq \Delta u_v \}$$

This definition is nearly the same as the one proposed in Section 4.4.1, with the exception

that either set K_v1 or K_v2 may now contain "split" flows having precisely $\Delta u_V = \delta_V{}^k$. This does not affect the lower bound calculation, but "splitting" a flow may be required to attain feasibility and to satisfy the LP complementary slackness conditions which require that if $u_S \geq 0$, then $\sum_{k \in K} \sum_{(i,j) \in A_S} x_{ij}{}^k = C_S$ precisely.

Return to the main "Inside Iteration" loop with the value of Δu_V .

Flow_Diversion(v, Δu_V)

Step 1:

For all flows $k \in K_v2$ (having $\delta_v{}^k \geq \Delta u_v$) which will *remain* on their current assignment, increase the current stored path cost by Δu_V .

Step 2:

For all flows $k \in K_v1$ (having $\delta_v{}^k \leq \Delta u_v$) which will be *diverted from* their current assignment, calculate a new diversion path using the shortest path routine. This routine also includes several "tie breaking" rules which are critical to the performance of the algorithm:

- Prevent use of the overcapacity segment "v". (Segment "v" will not be used anyway when $\delta_v{}^k < \Delta u_v$, however, in cases where $\delta_v{}^k = \Delta u_v$, blocking the link is necessary in order to positively divert the flow.)
- Among valid diversion paths having equal cost, select the path which minimizes the number of "tabu" links traversed.
- Among paths with equal cost and equal number of tabu links, select the earliest available train departure from each node.

Remove the flow from its current path. Then reassign each flow $k \in K_v1$ to its new shortest path, update the adjusted path cost, and reset $revisit_k = 0$.

4.4.5 Two Possible Future Efficiency Improvements

First, if a diversion cost δ_v^k has been previously calculated, and if flow $k \in K_v$ still remains on its original path with no link costs changed, then the old diversion cost $\delta_v^{k(old)}$ represents a lower bound on the new diversion cost $\delta_v^{k(new)}$. If the cost of alternative paths has been increased, while the cost of the current path remains the same, then the diversion cost of a flow could only have increased since the last iteration. If the previously calculated $\delta_v^{k(old)} > \Delta u_V$, it is not necessary to recalculate the diversion cost for that shipment.

Second, a review of the literature suggests two possible ways of calculating diversion costs without having to redo the shortest path calculations for each flow.

Shier [1979] describes algorithms to compute the "k" shortest paths in a network. In Shier's approach, a vector of the best "k" node costs so far is retained instead of overlaying and destroying this information when a new path is found. Shier's method could be used to calculate flow diversion costs in case of an overcapacity route segment, however, this approach may not find a diversion cost if all "k" paths happen to use the same overcapacity segment. It would require considerable effort to trace each of the "k" paths back to their origin to find the least cost path which bypasses the overcapacity link. This approach would compute and store many paths for each O/D flow, only a small fraction of which would ever be needed to support diversion cost calculations.

Halder [1970] devised a method of quickly calculating the flow diversion associated with an increase of cost on a specific link (u,v). One can construct a cut through the network, dividing the set of nodes *N* into two disjoint subsets X and X'. This cut is defined by:

$D(i,u) < D(i,v) (i \in X)$

$D(i,u) > D(i,v) (i \in X')$

where D(i,u) is the cumulative distance from root node i to u. If D(i,u) = D(i,v) then

i may be assigned to either set without error, as long as the distance between node u and node v, d(u,v) > 0. With this proviso, Halder shows the cut (X, X') is independent of d(u,v). Note that, for (u,v) to lie on a shortest route from i to j we must have:

$$D(i,j) = D(i,u) + d(u,v) + D(v,j)$$

Halder has thus identified the set of node pairs that are of interest in studying the effect of changes in the length of of (u,v). Moreover, the shortest path between any two nodes in the disjoint set (X, X') must cross the cut (X, X'). Thus, for any change in the length of (u,v), the shortest path for such a pair of nodes must pass through one, and only one, of the links in the cut-set. Some information has therefore been acquired about the alternative shortest paths.

Implementing Halder's algorithm would require the generation of two shortest path trees: to compute D(i,u), a tree forward from each origin, "i", second, to compute D(v,j), a tree backward from each destination "j". This would allows the identification of the candidate set of "competing links", the determination of the best diversion path, and calculation of the new path cost using D(i,j) = D(i,u) + d(u,v) + D(v,j). Halder reports that this approach is considerably more efficient than simply blocking the affected link and recalculating the shortest path each time.

4.4.6 Convergence of the Dynamic Car Scheduling Algorithm

The convergence of the DCS algorithm is a critical design consideration. As defined here, "convergence" means the algorithm will find a feasible solution if one exists. For an optimal linear programming-based algorithm, suppose we knew at the start what the LP optimal dual prices were, and simultaneously increased all dual prices to those values. This should drive off just enough flow from overcapacity segments so that the complementary

slackness conditions would be satisfied. If sufficient capacity is provided, an LP-based algorithm will find an optimal, feasible solution.

Although the dual adjustment heuristic is not guaranteed to find an optimal solution, the provision of sufficient capacity is still a *necessary* condition to assure convergence. Otherwise, no feasible solution exists and by complementary slackness the duals must be unbounded. The DCS algorithm would just keep increasing the dual variables until the iteration limit is reached, never able to drive off enough flow to attain feasibility. It is easy to provide sufficient capacity simply by extending the length of the planning horizon, for example, by assigning 7 days' traffic to 12 days' trains. This works because DCS trip plans are developed only for traffic currently moving on the railroad. Excess capacity can be provided at the end of the planning horizon, but capacity in the first part of the planning horizon may still be very tightly constrained.

Suppose that a problem is so tightly constrained that only one feasible solution exists. Then this single solution would also be optimal and any LP-based algorithm would be guaranteed to find it. A suboptimal heuristic might overlook this possible solution and enter a looping condition instead. However, as capacity constraints are relaxed, the set of feasible solutions expands such that it becomes very likely that the algorithm can find at least one feasible solution.

The central argument used in the proof of convergence of the LP simplex algorithm is that there exist a finite number of extreme points, and since the objective function is nondecreasing, the same extreme point cannot be revisited twice. If the objective function does not improve at every iteration, then Dantzig, Orden and Wolfe's [1955] "lexicographic simplex" modification provides an anti-cycling mechanism to assure convergence even in highly degenerate problems. Therefore the simplex method must converge to an optimal solution within a finite (although possibly very large) number of iterations. If the simplex

algorithm ever were to revisit the same extreme point twice it would be in a looping condition and might never terminate.

For the dynamic car scheduling algorithm, a similar convergence argument can be made. Since price decreases are not allowed, any time a dual price is increased the same combination of dual prices can never occur again. Each time a flow is diverted, additional tabus are stored. Tabus in the Dynamic Car Scheduling procedure serve essentially the same purpose as the “lexicographic” modification to the Simplex Method, referred to in the previous paragraph. Thus, the same combination of dual prices, flow assignments and tabus can never be revisited twice. Combined with the conditions established in Section 4.4.1 for an improving lower bound, this parallels the LP convergence proof, establishing at least that the algorithm cannot enter a closed looping condition.

However, this still falls short of guaranteeing that the dynamic car scheduling procedure will always find a feasible solution if one exists. The primal simplex method operates at all times in the feasible region. It can be terminated prematurely (before true optimality is reached) and still have a feasible, but suboptimal solution. In contrast, this dual adjustment procedure, like the dual simplex method, operates in the infeasible region. These dual methods cannot be terminated prematurely and still have a feasible solution.

Even if an absolute mathematical proof of convergence were to be developed, it would likely still fall short of guaranteeing acceptable computational performance in a practical implementation. The only way to really ensure acceptable performance is through extensive computational testing. The DCS procedure has been tested thousands of times embedded within the rolling horizon simulation model, and has never failed to find a feasible solution to a practical problem. As well as demonstrating the steps of the algorithm, its ability to solve very tightly constrained problems will be demonstrated by

example in Figures A.10-A.13 in the Appendix. The train service network, yard costs and traffic demands shown in Figures A.5-A.9 in the Appendix are used in all examples.

The first example in Figures A.10-A.11 represents a moderately constrained problem, significantly more difficult than most real-world problems we usually expect to solve. Each problem is solved once in each sweep direction. The "Sweep Up with Look Ahead" method requires one fewer iteration, but the two approaches seem to be roughly equivalent in their ability to solve this particular test problem. In either direction the algorithm is able to solve the problem to optimality while honoring the integral flow requirement.

The second, more difficult problem in Figures A.12-A.13 uses the same demand and train schedules but slashes train capacities to only six cars instead of ten, representing a severe stress test of the algorithm. In this extremely stressed scenario, the Sweep Up approach at first appears to perform better attaining both a faster convergence rate and also a tighter duality gap. However, upon closer examination, it can be seen that Sweep Down with Look Ahead actually produces a better objective function value in the third iteration, although the lower bound calculation is not as tight.

Either approach appears to be capable of finding a feasible solution; however, the issue again becomes how the *distribution* of adjustments ultimately affects performance. Sweep Up with Look Ahead tends to place heavier corrections in the earlier part of the planning horizon; whereas, Sweep Down with Look Ahead biases those corrections to segments deeper down. Recall (from Section 4.4.4) that a "tabu" list of previous assignments is maintained for each shipment. If a shipment having "tabu" links is included in a diversion cost calculation, that shipment is marked with $revisit_k = 1$. Then during the next "outside" iteration that flow will be inspected and, if necessary, recalculated and restored to a true shortest path assignment. This process continues until a solution is obtained

where the assignment satisfies *both* conditions: (1) All train segment capacity constraints are respected, and (2) All shipments are routed via "true" shortest path assignments. This allows both a primal objective function value and valid lower bound to be derived from the same solution.

In Figure A.13, the severely stressed "Sweep Down with Look Ahead" a large number of "outside" iterations are required to attain convergence on both criteria, furthermore, after the third iteration (attaining $9680), the primal objective function value starts to degrade. By the 13'th iteration the objective function has been severely degraded to a final value of $12,393. What happened?

Unfortunately, this process of restoring flows to shortest path assignments may also destroy feasibility. Then the DCS algorithm must be called again, triggering another "Outside Iteration" to restore feasibility. Given the above two-part requirement, when any shipments are restored to shortest path routings, applying the heavier corrections in the earlier part of the planning horizon tends to drive traffic *down* to later time periods, *away* from the original overcapacity segments. If, however, the corrections are applied to segments deeper down, those shipments tend to be driven back *up* to earlier time periods, that is, returned to their original overcapacity assignments.

This bias explains why Sweep Up with Look Ahead converges after only three outside iterations, while Sweep Down with Look Ahead requires 11 outside iterations to satisfy both conditions (1) and (2). During these additional iterations, the dual variables (particularly in the earlier time periods) were seriously overcorrected and both the objective function and lower bound were seriously degraded. Normally, this effect is not so extreme, but this example illustrates how too many outside iterations can degrade the final result.

In fact, the requirement to obtain both a valid lower bound and feasible solution from the *same* traffic assignment, while convenient, is overly restrictive. For those flows

not on shortest path assignments, it's only necessary to calculate the shortest path *cost* and use that cost in place of the actual path cost to obtain a valid lower bound calculation. It's not actually necessary to physically reassign the flows. This adjustment produces the same lower bound which would have been obtained had the flows actually been restored to shortest path assignments, without destroying feasibility or needing to trigger additional outside iterations. The effect of turning the outside iterations "off" will be further explored in the next section.

4.4.7 Rolling Horizon Testing of Dynamic Car Scheduling

The focus of this section will be to assess the computational performance of several variations of the Dynamic Car Scheduling (DCS) algorithm in terms of such *technical* criteria as duality gap, required CPU time and number of iterations required to attain a solution. The *business* impacts of implementing dynamic car scheduling on such measures as service reliability are the focus of Chapter 5 and will not be examined here. Rolling horizon testing will examine the effects of varying:

(1) Sweep Direction "Up" or "Down"

(2) Look Ahead "On" or "Off"

(3) Implementation "Stand Alone" or in conjunction with the Train Segment Pricing algorithm

This yields a total of eight scenarios to be tested. These simulation testing results will be presented in Tables 4.1 and 4.2. Simulation testing will also examine the effects of turning the outside iterations "off" as suggested by the findings of the previous section.

The DCS algorithm has been embedded within a rolling horizon simulation model. A full description of this model will be presented in Chapter 5, but for the purpose of this section, the simulation simply serves as a "test problem generator." The simulation does,

however, add an additional dimension to the problem, by "locking" the first leg of the current trip plan if a shipment has already been classified in a yard. A certain number of hours after arrival, cars are sorted into outbound departure blocks and trains based on each shipment's current trip plan. At this point, the outbound departure block and train are "locked" by the model and cannot be changed, even if a higher priority car calls in later.

The effect of "locking" is shown in Figure 4.9, where a low priority car having an hourly cost of $1 has already been processed before a high priority shipment costing $2 is called in. To keep things simple, each segment only has 1 car capacity. Since the train hasn't departed the origin yard yet, network link A_1 still exists, so the high priority car is initially assigned to A_1-B_1, causing both segments to exceed capacity. The dynamic car scheduling algorithm is called to restore feasibility. What happens? As shown in Figure 4.9, the answer depends on which sweep direction is used. "Sweep Down" or "Sweep Up with Look Ahead" will apply the total dual variable increase to the *earlier* segment A_1. "Sweep Up" or "Sweep Down with Look Ahead" spreads the increase across both segments. Although the same total price increase of $48 is required, the distribution of the adjustments is different.

Ultimately, "Sweep Down with Look Ahead" equilibrates path cost for the low cost shipment on either segments B_1 or B_2, so if another shipment calls in later for B_1, a smaller price increase to B_1 would subsequently be required. In contrast, if the total $48 is initially charged to A_1, then a larger increase may be subsequently required to B_1 if additional traffic calls in later. This result has been confirmed in computational testing in Tables 4.1 and 4.3, which show that "Sweep Down with Look Ahead" generally produces smaller total price adjustments, lower average objective function values, tighter duality gaps and requires less computational effort, as compared with any other approach.

Figure 4.9 - Effect of Leg Locking

- Each flow is 1 car
- Each segment capacity is 1 car
- The $1/hr flow is "locked" onto segment A_1

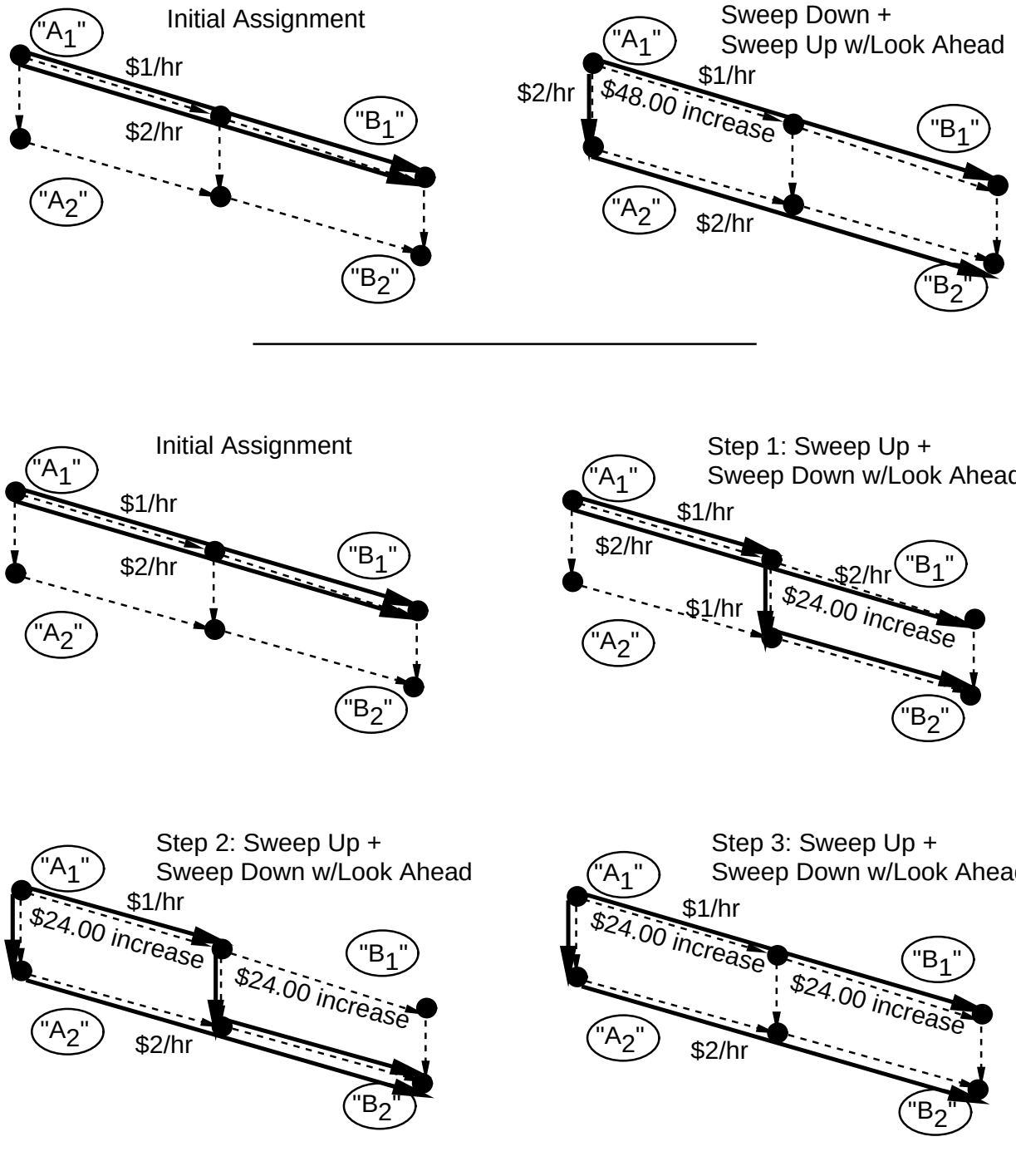


Table 4.1 presents the results of the first series of rolling horizon simulation model tests for 500 simulated hours each. These detail the results from the eight combinations of sweep direction, look ahead and operation either stand alone or in conjunction with the Train Segment Pricing model. CPU timings reported in Table 4.1 were obtained on a Macintosh Performa 6200CD computer, which has a Motorola 603 PowerPC chip running at 75 MHz. The program is implemented in C++ using a Symantec compiler. Executing this software on a faster machine such as a RISC/6000 would probably produce at least an order of magnitude speedup. Also, both the TSP and DCS algorithms are well suited to parallel processing of shortest path subproblems (resulting from the Lagrangian Relaxation of the capacity coupling constraints), which could result in substantial additional speedups if implemented on a multi-processor machine.

Table 4.1 - Performance of Dynamic Car Scheduling Algorithm

	Total Calls	CPU Sec/Call	Total Iterations	Iter to 1st Feas	Outside Iterations	% Gap
Sweep Up w/LK Alone	967	8.76	3.73	2.23	1.42	0.93
Sweep Up w/LK TSP	519	7.67	2.84	1.95	1.31	2.11
Sweep Up NO LK Alone	913	8.49	3.11	2.1	1.42	1.40
Sweep Up NO LK TSP	497	7.61	2.94	1.87	1.32	2.86
Sweep Dn w/LK Alone	910	8.04	3.08	2.20	1.38	0.77
Sweep Dn w/LK TSP	493	5.78	2.21	1.75	1.26	1.36
Sweep Dn NOLK Alone	940	8.26	3.02	2.08	1.40	1.00
Sweep Dn NOLK TSP	513	7.06	2.48	1.84	1.35	2.45

As shown in Table 4.1, DCS duality gaps tend to be very good. When DCS is operated stand alone, with look ahead in either sweep direction the duality gap averages less than 1%.

When DCS is operated in conjunction with Train Segment Pricing (TSP), penalty costs for late deliveries tend to amplify the duality gap. Yet overall DCS performance is still quite good yielding an average duality gap of 1.36% for the Sweep Down with Look Ahead approach. Significantly, the required number of calls to DCS is cut nearly in half and also the average time required per DCS iteration is reduced, regardless of sweep direction or look ahead. This appears to be the result of the introduction of delivery commitments on each shipment, along with penalty costs for late delivery in the objective function, which provides a clearer prioritization mechanism. To a lesser degree, this performance improvement may also be a result of TSP shedding a small amount of traffic volume, which consists either of unprofitable shipments rejected by the TSP algorithm, or a few cases when the customer rejected the TSP service offer.

Table 4.2 gives a comparison of results obtained if the DCS algorithm is stopped at the *first feasible solution*, as compared to the case where all assignments must be via "true" shortest paths must also be satisfied, requiring additional "outside iterations." With respect

Table 4.2- Comparison of First, Last and Best Iteration

	% Iter Better	% Gap Improved	Z* First	Z* Last	u First	u Last
Sweep Up w/LK Alone	14	.048	467842	467874	2528	2537
Sweep Up w/LK TSP	8	.118	486687	486825	4977	5004
Sweep Up NO LK Alone	13	.043	461386	461444	2538	2548
Sweep Up NO LK TSP	10	.119	471903	472113	4798	4827
Sweep Dn w/LK Alone	11	.055	461361	461378	2453	2463
Sweep Dn w/LK TSP	5	.116	469601	469700	4918	4938
Sweep Dn NOLK Alone	10	.046	487623	477704	2744	2756
Sweep Dn NOLK TSP	8	.126	479218	479343	4738	4760

to additional “outside” iterations, a better feasible solution than the first one is only found 5-14% of the time, and when found, generally results in only small improvement in the objective function value. *On the average, outside iterations degrade the objective function for every scenario.* Furthermore, dual price increases and excessive tabus applied during the outside iterations worsen the “starting” point for subsequent DCS iterations.

Based on these findings, confirming the results presented in Figure A.13, the recommended DCS operating mode will be to simply drive to the first feasible solution and stop. Additional “outside” iterations should not be performed. Once a feasible solution is obtained, a lower bound can be calculated by simply reexamining the flows marked with $revisit_k = 1$, without having to actually change the physical assignments or disrupt the current feasible solution. As a result of this change to the DCS operating mode, it was necessary to repeat the simulations previously performed. Table 4.3 shows the results of a repeated test of the “With Look Ahead” cases where the new operating mode is to simply drive to the first feasible solution and stop.

As in Table 4.1, “Sweep Down with Look Ahead” gives the best overall performance, producing the lowest average objective function values at the first feasible

Table 4.3 - Performance of Dynamic Car Scheduling Algorithm with "Outside Iterations" turned "Off"

	Total Calls	CPU Sec/Call	Total Iterations	% Gap	Z*	Flows Revisited
Sweep Up w/LK Alone	975	7.91	2.46	0.86	461765	12.89
Sweep Up w/LK TSP	545	5.22	1.97	1.48	478505	4.00
Sweep Dn w/LK Alone	954	7.37	2.34	0.7	461257	11.81
Sweep Dn w/LK TSP	548	5.90	2.10	1.06	473246	4.83

Figure 4.10 - DCS Duality Gap by Time

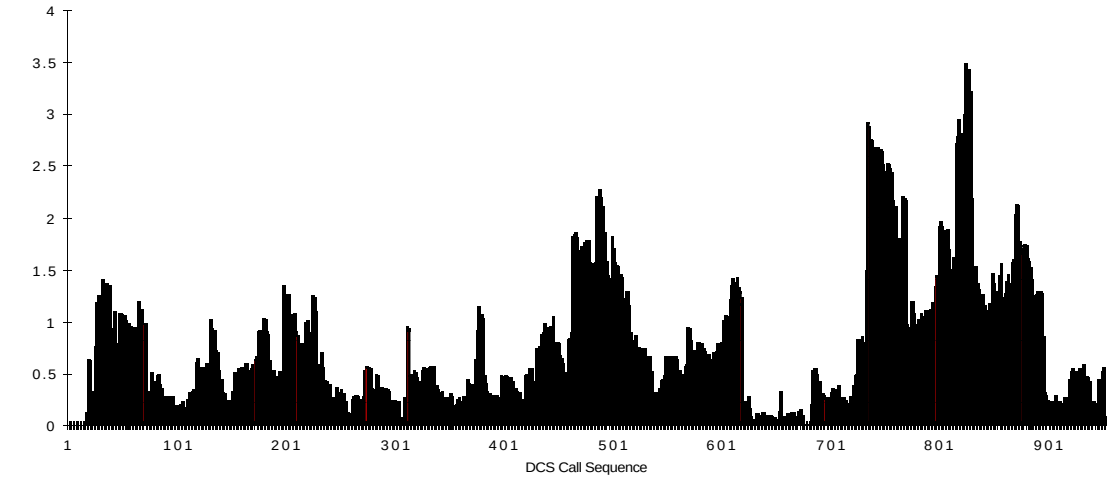


Fig 4.11 - Required Number of Iterations

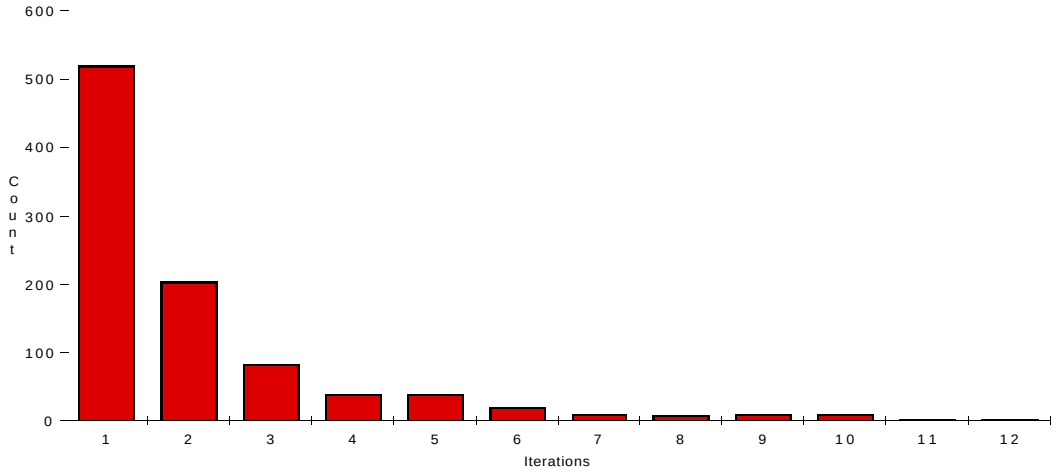
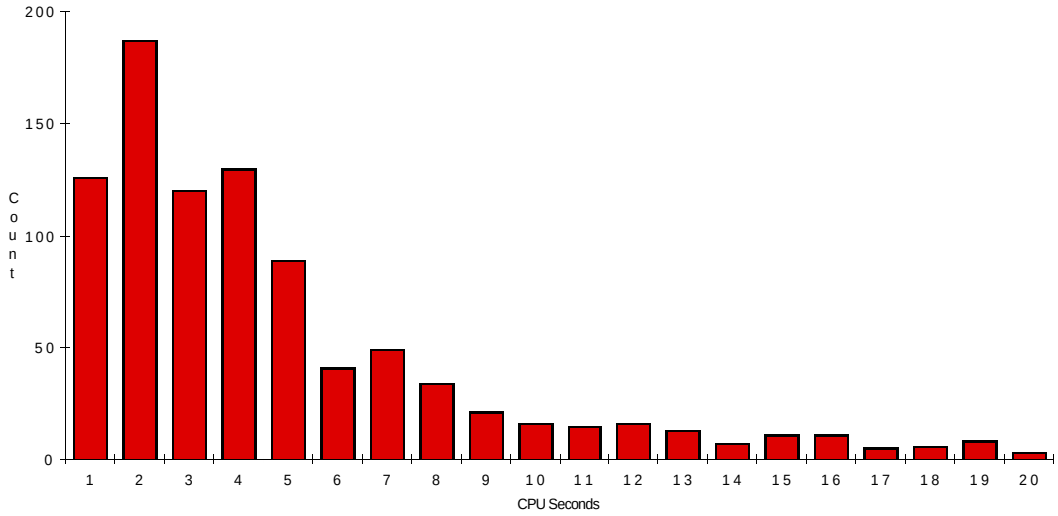


Fig. 4.12 - Required CPU Time



solution. When outside iterations are turned “off”, computational results still compare favorably with those given in Table 4.1. The “average number of flows revisited” column shows that, in this scenario, the number of flows remaining on non-optimal assignments are not very significant considering that over 1200 shipments are included each week within the simulation model.

Figure 4.10 shows the DCS Duality Gap by time from the “Sweep Down with Look Ahead” scenario given in Table 4.3. It shows some highly stressed time periods where the duality gap may reach as high as 3%, but rapidly recovers averaging 1.06% in 954 times DCS was called during the course of the simulation.

Figure 4.11 shows the distribution of total iterations required, and Figure 4.12 shows required CPU time.Both distributions are highly skewed with a long tail. Figure 4.11 extends out to 33 iterations; Figure 4.12 extends to 129 cpu seconds. Most problems can be solved with much less effort than the averages given in Table 4.1, but occasionally a very difficult problem is encountered which requires much more effort to solve.

The final recommendation will be to operate DCS using Sweep Down with Look Ahead, and Outside Iterations turned “Off”. This standard operating mode will be used in all the rolling horizon simulations of Chapter 5.

4.5 Train Segment Pricing Model

The train segment pricing model (TSP) assigns a 7-10 day O-D demand forecast to a planned train service network. Either actual or booking limit capacities can be used. Dual prices for overcapacity train segments are estimated using a standard subgradient step size procedure along the lines of Geoffrion [1974] and Fisher [1981]. The subgradient “step size adjustment” procedure adjusts dual costs using a simple formula:

$$\text{ADJUSTMENT} = \text{STEPSIZE} * \left(C_S - \sum_{k \in K} \sum_{(i,j) \in A_S} x_{ij}^k \right)$$

Unlike the dynamic car scheduling algorithm, no shipment-specific "diversion costs" must be calculated in this process. Neither are price adjustments limited to increases only — decreases are also allowed. However, the effect of constraining price reductions will be examined in Section 4.5.3.

The subgradient algorithm is used here primarily because the problem size is much larger than in Dynamic Car Scheduling, including a full weeks' forecast demand, and the main requirement of TSP is only to determine the dual prices, not a feasible solution. Geoffrion [1974] established conditions under which the subgradient algorithm will converge to an optimal solution, and since the TSP need not be solved in real time, this performance guarantee makes the subgradient algorithm an extremely attractive approach.

The subgradient algorithm is unlikely to equilibrate path costs among multiple paths. Costs would only be matched by accident or in the limit approaching infinity. Thus, the basis of the algorithm is all-or-nothing assignment to the most profitable path. No attempt is made to split flows across multiple paths. This means that convergence of the subgradient algorithm is essentially unaffected by the degeneracy that plagues LP-based solution methods. As shown in Figure A.14 in the Appendix, the subgradient procedure cycles flows back and forth without ever converging to a feasible solution. However, upon closer inspection, it can be seen that a tight lower bound is still attained in spite of this apparent non-convergence. A feasible solution, if desired, can be readily obtained through the application of a simple Lagrangian heuristic.

Fukushima [1984] tested a standard subgradient approach to a single commodity, *nonlinear* highway traffic assignment problem and found lower bounds within 1-2% of optimal, solving in half the time required by the Frank-Wolfe algorithm to attain a similar solution quality. Fukushima remarked that all-or-nothing assignment would always provide a

lower bound on the optimal solution, and made no attempt to split flows over more than one route.

4.5.1 Subgradient Solution Procedure

The following gives a formal statement of the subgradient algorithm used to solve the train segment pricing model. The procedure adjusts dual prices on each train segment proportional to the amount of traffic overflow or underflow on that segment. The adjustment is determined by multiplying the overflow or underflow by a "step size" parameter which must be calibrated to give best results.

Step 1 - Initialization

Set $\kappa = 0$, where κ is the current iteration number. Set all dual variables $u_{ij} = 0$ for the first time the subgradient procedure is called. However, if embedded in a rolling horizon simulation, simply carry forward the final dual variable values from the previous day's run and use those duals to "jump start" this procedure.

Step 2 - Maximum Iteration Test

Set $\kappa = \kappa + 1$;

If $\kappa >$ Maximum Iteration Limit then Stop

Invoke the Primal Heuristic (Section 4.5.2) to get a feasible solution and allow measurement of the duality gap, if the Primal Heuristic has not already been turned on.

Step 3 - Shortest Path Assignment

For each commodity "k" and link "i-j", adjust link costs by adding the dual (Section 3.3):

$$\varphi_{ij}^{k} = c_{ij}^{k} - \sum_{s \in S_{ij}} u_s$$

S_{ij} segments associated with link (i,j)

Then solve "Modified Shortest Path" problems as described in Section 4.2.1 for each shipment, to determine the optimal delivery time. Each shipment is assigned to the shortest path between origin node in the space/time network and node corresponding to the optimal delivery time.

The expected volume loaded onto the network for each shipment "k" is:

$$\Lambda_k = \left(\sum_{n \in T_k} P_{nk} M_{nk} \right) \xi_k$$

where:

T_k = The set of "termination" nodes from which shipment "k" can be delivered to the consignee, $T_k \subset N$.

P_{ik} = The probability customer of commodity "k" accepts the service offer for shipment delivery at node $i \in T_k$.

M_{ik} = 1 if shipment "k" delivered from node $i \in T_k$; else 0.

ξ_k = Expected demand for commodity "k".

Λ_k = The total flow to be assigned taking customer acceptance probability into account.

Step 4 - Compute $Z_D(u^\kappa)$; if no improvement after five iterations, cut the Step Size Multiplier λ_κ in half.

Based on the assignment in Step 3, compute the new dual objective function value $Z_D(u^\kappa)$:

$$Z_D(u^\kappa) = \sum_{i,j,k} \varphi_{ij} x_{ij}^k - \sum_S u_S \left(\sum_k \sum_{(i,j) \in A_S} x_{ij}^k - B_S \right)$$

where:

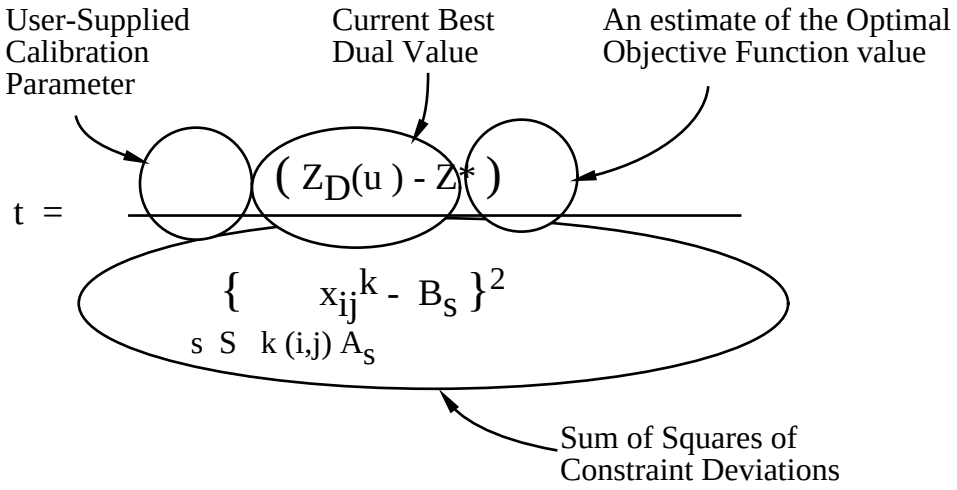
A_s = The set of links (i,j) associated with segment "s", defined the same as in Section 3.3.

B_S = A user-supplied booking limit, which can be greater or less than C_S, the actual capacity of train segment $s \in S$.

If $Z_D(u^\kappa)$ is less than the best previous value, store the value of $Z_D(u^{best})$. If $Z_D(u^{best})$ has not improved after 5 iterations, set $\lambda_{\kappa+1} = \lambda_\kappa / 2$.

Step 5 - Determine Step Size

The following formula suggested in Fisher [1985] is used for computing the step size "t_k" in each iteration κ:



The parameter λ_κ is supplied by the user and reduced after five iterations if the algorithm fails to make progress. A result given in Held, Wolfe and Crowder [1974] states conditions on t_κ under which the subgradient algorithm will converge to the optimal dual value Z_D. In summary, t_κ should converge to zero, but not too quickly, or else (as shown in Fisher [1985]) the subgradient algorithm may converge to a point other than the optimal solution. The parameter Z* is an estimate of the optimal primal objective function value.

For this application, *Z* has just been set equal to zero,* since this avoids any requirement to execute the primal heuristic at every iteration (see Section 4.5.2), and computational testing with Z* = 0 has yielded satisfactory results.

Step 6 - Update the Dual Variables

For each segment S, the dual variables are updated using:

$$u_S^{+1} = \max \{ 0, u_S - t \quad (\quad x_{ij}^k - B_S) \}$$
$$k\ (i,j)\ A_S$$

where:

u_S^κ = Value assumed by the dual variable on segment S during iteration "κ" of the subgradient algorithm. u_S^κ is not allowed to go negative.

α_κ = User supplied parameter α, $0 \leq \alpha \leq 1$;

$\alpha_\kappa = \alpha$ if $(\sum_{k} \sum_{(i,j) \in A_S} x_{ij}^k - B_S) < 0$;

$\alpha_\kappa = 1$ otherwise

The α parameter has been used to perform computational experiments in an attempt to reduce the infeasibility remaining in the dual problem. The results of this experiment will be described in Section 4.5.3.

Other terms are as defined above and in Section 3.5.

Then Go to Step 2.

4.5.2 Obtaining a Feasible Solution and Integration with Dynamic Car Scheduling

Using the Lagrangian primal heuristic is strictly optional since its only use here is to measure the duality gap. Only the dual prices are required to support the reservations and booking process, and these can be obtained from the standard subgradient algorithm. However, the output is still quite desirable for operations planning purposes, since the heuristic develops a "forward workload projection" based on the demand forecast. This includes both cars currently moving on the railroad as well as anticipated future demands.

The Lagrangian heuristic is a greedy assignment algorithm which obtains a feasible solution by rank ordering all flows based on priority, then sequentially assigning the flows on a shortest path basis. Cars currently moving on the railroad are assigned first, according to their current trip plans. Once underlying segments become saturated, associated network links are blocked, preventing those links from receiving any further traffic assignments. Subsequent assignments bypass saturated links. The heuristic is similar to an approach which AADT developed for the Santa Fe Railway (Gorman [1994]). *However, Santa Fe's heuristic is improved by using adjusted dual prices developed by the subgradient algorithm.* Much excess flow is priced off, leaving only a small residual. Computational testing of this procedure generally produces optimality gaps of less than 1%.

It should be pointed out that each shipment may be required to be assigned *twice* using the heuristic. The first assignment is performed on a shortest path basis by the subgradient algorithm. Then:

- If saturated segments are encountered along the original path, the flow must be reassigned with saturated segments blocked, until a path with some available capacity is found. If the available capacity is not sufficient to hold the entire shipment, then the shipment is split into two parts, saturating the first path and reassigning only the overflow.

• If the first path *does* have available capacity, it is not necessary to perform a second traffic assignment. Performing both dual and primal traffic assignments at the same time allows the algorithm to take advantage of this potential processing efficiency. Assigned link and segment volumes are tracked separately for the primal and dual problems.

Since the Train Segment Pricing problem formulation includes probabilities, the expected value of flow is likely to be non integral even if the underlying probability distribution is discrete. Because of this, the integral flow requirement applies only to shipments which have already been accepted and are already moving on the railroad. For expected future shipments, the integral flow requirement is relaxed, and the first path is completely filled up before diverting flow to a second path, even if that involves splitting the flow on a fractional basis. This does not create any problems since most of those future "expected" flows are fractional to begin with. This acknowledges the fact that the exact number of units shipped or the routing they will take, cannot be predicted with certainty.

To avoid a non-integral split of shipments already moving on the railroad, these must be assigned by the Lagrangian heuristic first. Since segment capacities are integral, processing these shipments first would be sufficient to ensure that no fractional splits are needed. However, all these shipments already have trip plans developed by the Dynamic Car Scheduling algorithm. Rather than re-routing these shipments on a shortest path basis, the approach taken here is simply to reuse the same trip plan already developed by the Dynamic Car Scheduling algorithm. Reusing this information results in considerable processing efficiency, maintains consistency between the two approaches, and as will be demonstrated in the next section, still produces a tight duality gap.

The Dynamic Car Scheduling trip plan is reused *only* in the primal heuristic. This trip plan cannot be reused in the subgradient algorithm itself because of the requirement that all subproblems must be solved to optimality. In the subgradient algorithm, current traffic as

well as predicted future flows are all reassigned on a shortest path basis to ensure the validity of the upper bound calculation.

Prior to running the heuristic, flows are presorted into two groups: current "real" and predicted future flows; within each group, the shipments are rank ordered in order of priority. The rank ordering criteria has a significant effect on the performance of the primal heuristic, but the order in which flows are processed has no effect on the subgradient algorithm.

The following is a formal statement of the Lagrangian heuristic:

Define:

Ψ^k	=	Customer acceptance probability if shipment "k" is delayed 24 hours (computed from logit function).
R^k	=	Shipment "k" revenue if customer accepts offer.
C_B^k	=	"Base Cost," the cost of the shortest path assignment with all dual variables set to zero.
ϖ^k	=	The hourly operating cost including car hire, terminal track space cost, etc.
PSegs	=	The set of pooled segments to be excluded from the shortest path calculation, $PSegs \subset S$

Step 1 - Rank Order Flows

Rank order flows within each subgrouping "Current" or "Future" flows, by expected impact of a 24-hour delay ϑ^k using:

$$\vartheta^k = (1 - \Psi^k)(R^k - C_B^k) + 24\,\Psi^k\,\varpi^k$$

The first term of this expression, $(1- \Psi^k)(R^k - C_B^k)$, corresponds to the expected net revenue loss from a 24 hour delay, while the second term $24\ \Psi^k\ \varpi^k$ is the expected hourly operating cost increase.

Process all "Current" flows first, followed by "Future" flows to avoid the occurance of fractional splits on the "Current" flows, which would violate the integral flow requirement on these flows.

Step 2 - Carry Forward Dual Variables

Use the same dual variables and adjusted link costs φ_{ij}^k used in the current iteration of the subgradient procedure. Update the individual link costs to reflect the price adjustment on the underlying segments:

$$\varphi_{ij}^k = c_{ij}^k + \sum_{s \in S_{ij}} u_s$$

segments associated with link (i,j)

Step 3 - Solve "Modified Shortest Path" subproblems for each flow, blocking saturated links

Select the next shipment "k" to assign until all flows have been assigned. Once all flows have been assigned, report the duality gap and perform another subgradient iteration until the iteration limit is reached. If the flow is currently moving on the railroad, assign it according to the trip plan developed by the Dynamic Car Scheduling algorithm. Otherwise, for all "future" flows:

Initialize *PSegs* $= \varnothing$.

Using adjusted link costs φ_{ij}^{k}, solve modified shortest path problems to determine both path taken and the customer acceptance probability. Determine the traffic volume to be assigned Λ_k using:

$$\Lambda_k = \left(\sum_{n \in T_k} P_{nk} M_{nk} \right) \xi_k$$

For each link along the shortest path, determine the smallest remaining capacity ζ of any train segment traversed. This defines the maximum quantity of flow which can be assigned without over- saturating the path.

If: (3a) $\zeta = 0$, then the path is completely saturated. Identify the first saturated link (with no remaining capacity), add it to the segment pool *PSegs* to prevent the link being utilized in the shortest path calculation, and recompute the Prize Collecting subproblem.

(3b) $0 < \zeta < \Lambda_k$ then split the flow into two parts $\Lambda_{k1} = \zeta$ and $\Lambda_{k2} = \Lambda_k - \zeta$.

Compute:

$$\xi_{k1} = \Lambda_{k1} \Big/ \left(\sum_{n \in S} P_{nk} M_{nk} \right)$$

$$\xi_{k2} = \Lambda_{k2} \Big/ \left(\sum_{n \in S} P_{nk} M_{nk} \right)$$

Assign ξ_{k1} and go to (3a) with the remaining ξ_{k2} .

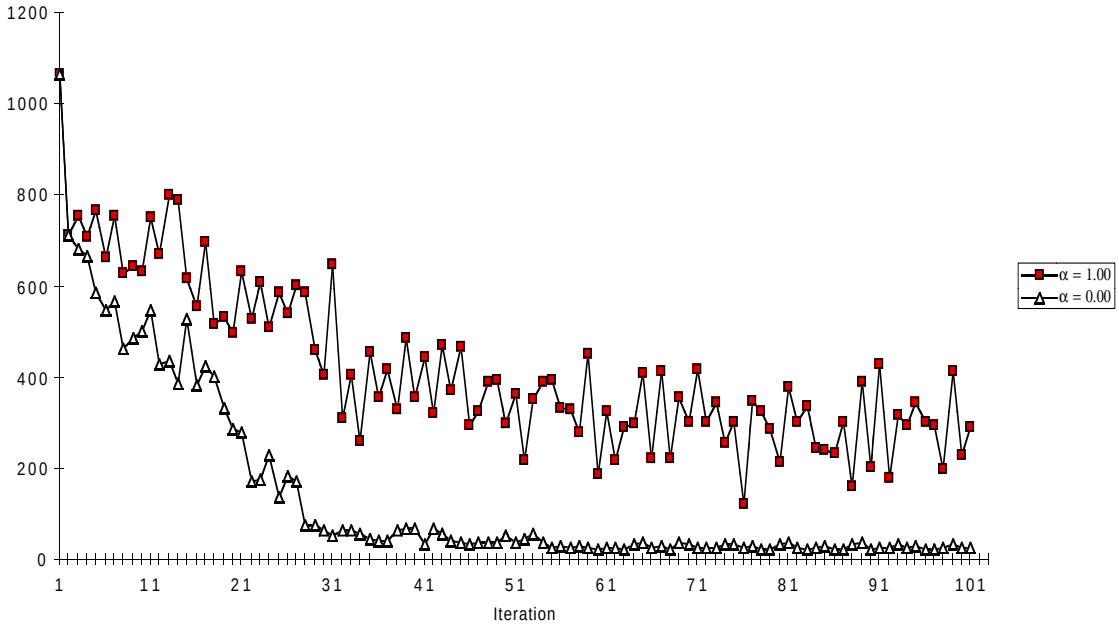
(3c) $\zeta \geq \Lambda_k$ then there is ample capacity remaining along the entire path; assign commodity “k”, go to the beginning of Step 3 and process the next flow.

4.5.3 Algorithmic Performance

The algorithm was tested on a problem adapted from Kwon [1994] containing 1250 traffic flows. Cost, revenue and service sensitivities were randomly generated for each flow. The traffic dataset and rail service network description will be found in an Appendix to this dissertation. The algorithm was tested using a parameter $0 \le \alpha \le 1$ which multiplies any negative adjustments. If $\alpha = 1$, we have the standard step size procedure with equal price adjustments in either direction. If $\alpha = 0$, negative cost adjustments are disallowed.

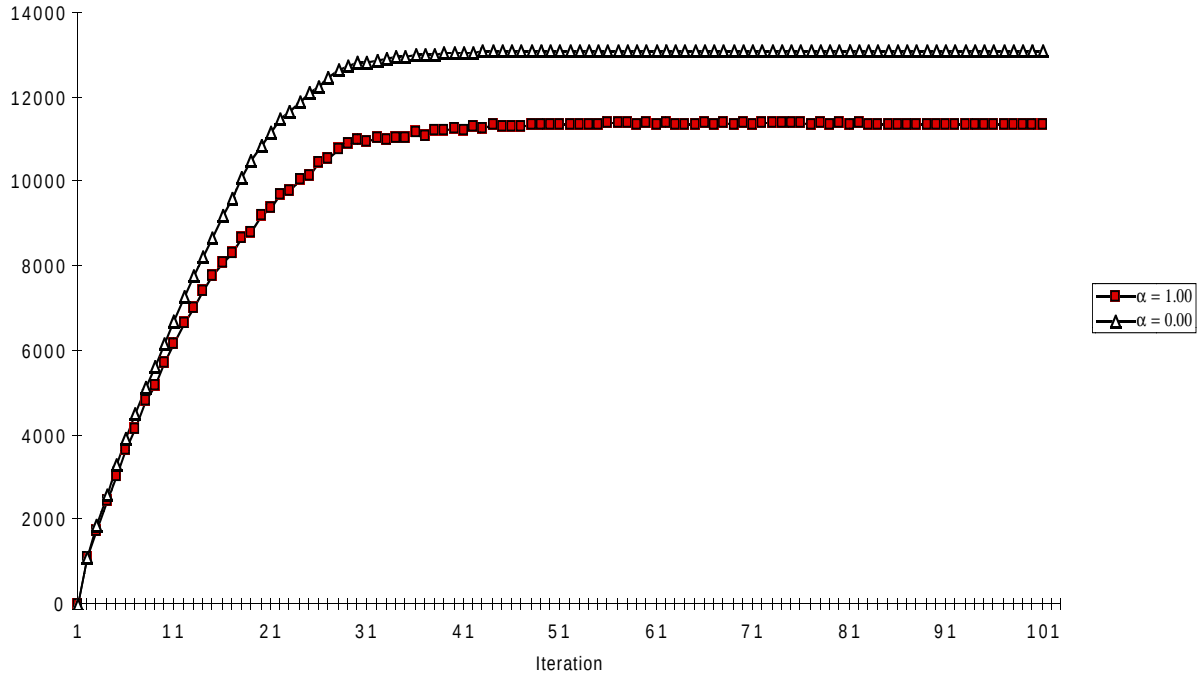
As shown in Figure 4.13, the algorithm drives out the most infeasibility if negative cost adjustments are disallowed. In the test problem with α= 0.00, total infeasibility was reduced from 1067 to 24 cars, a 97% reduction. The standard subgradient procedure with α= 1.00 does not drive out quite so much infeasibility, reaching at best an excess of 124 cars, but averaging about 300 cars residual infeasibility once the result stabilizes after 50 iterations. Results with α= 0.25, 0.50 and 0.75 are between these two extremes.

Fig 4.13 - Remaining Infeasibility



The performance of the primal heuristic depends more on the accuracy of the dual prices than it does on the infeasibility remaining in the dual problem. With α=0.00, most of the infeasibility has been driven out of the dual solution, so one might think this would represent an excellent starting position for the primal heuristic. Yet with α=1.00, the primal objective function value was better in spite of more infeasibility in the starting solution. α=1.00 produced a 0.57% gap. The best gap attained with α=0.00 was 1.24%.This result does not appear to be an aberration, since the best dual value monotonically decreases, and the best primal value monotonically increases as α approaches 1.00.

Fig 4.14 - Cumulative Correction



The sum total of correction applied, Σu_S is shown in Figure 4.14. With α= 0.00 this reaches 13,124 versus 11,370 with α = 1.00, indicating that the dual variables are overcorrected with α= 0.00. Results with α= 0.25, 0.50 and 0.75 are between these extremes.

Fig 4.15 - Upper Bound and Primal Solution

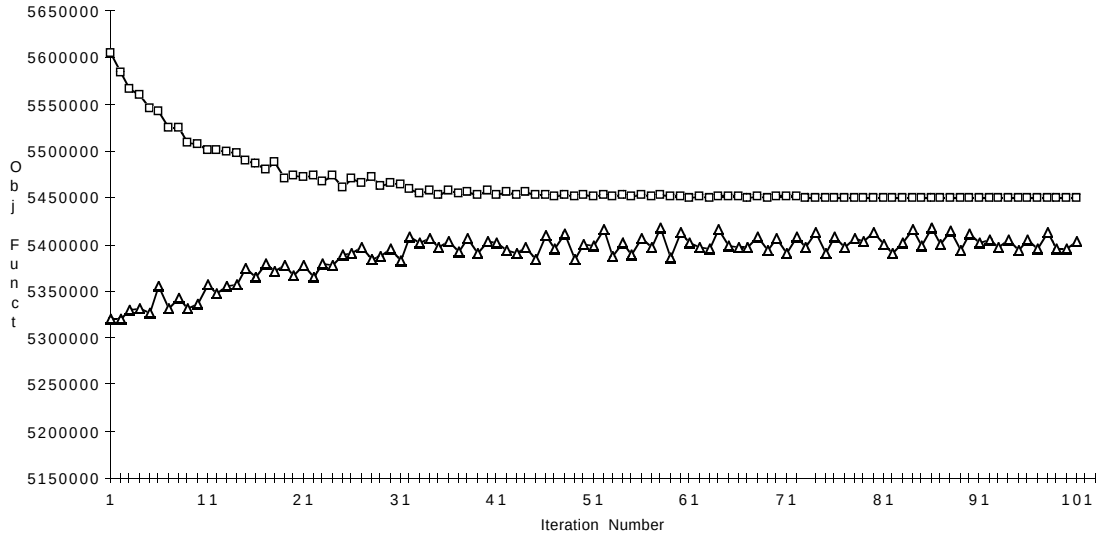


Figure 4.15 shows that the primal heuristic definitely performs better when started with dual values from the subgradient procedure, as opposed to working directly on the problem with all u_{ij}'s = 0 (as the AADT algorithm does in Gorman [1994]). Initially, with all u_{ij}'s = 0, the objective function starts at only 5.32 million, but improves to 5.42 million if dual prices from the subgradient algorithm are used. A 1% gap is attained by the end of iteration #33.

Note also that a tighter gap will be computed if the primal heuristic is applied on *every* subgradient iteration versus only on the final iteration. In Figure 4.15, the value of the objective function at the *last* iteration is 5,403,984 versus the *best* value attained of 5,419,164. The gap measured at the *last* iteration is 0.856% versus 0.574% if measured based on the *best* objective function value. *In this example, executing the heuristic at every iteration, rather than only at the last, increases the required CPU time for the Train Segment*

Pricing algorithm by 23%. This increase would be greater for more tightly constrained problems, and correspondingly less for problems with plenty of slack capacity.

In view of the fact that by either measurement the gap is less than 1%, it hardly seems worthwhile to expend the additional computational effort to execute the heuristic at every subgradient iteration. Figure 4.16 shows that in the rolling horizon simulation, the vast majority of CPU time is consumed by the Train Segment Pricing algorithm, so it is undesirable to further increase this resource consumption by unnecessarily executing the primal heuristic.

Figure 4.16 - CPU Time Distribution

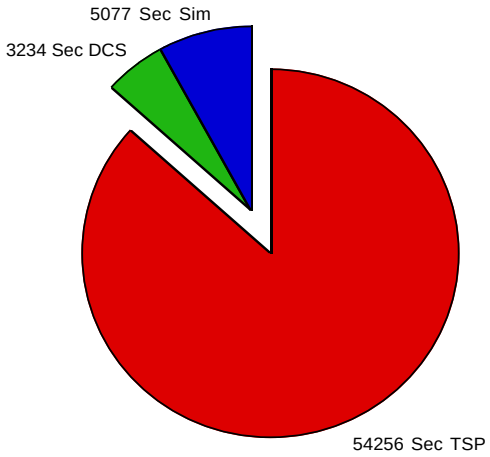
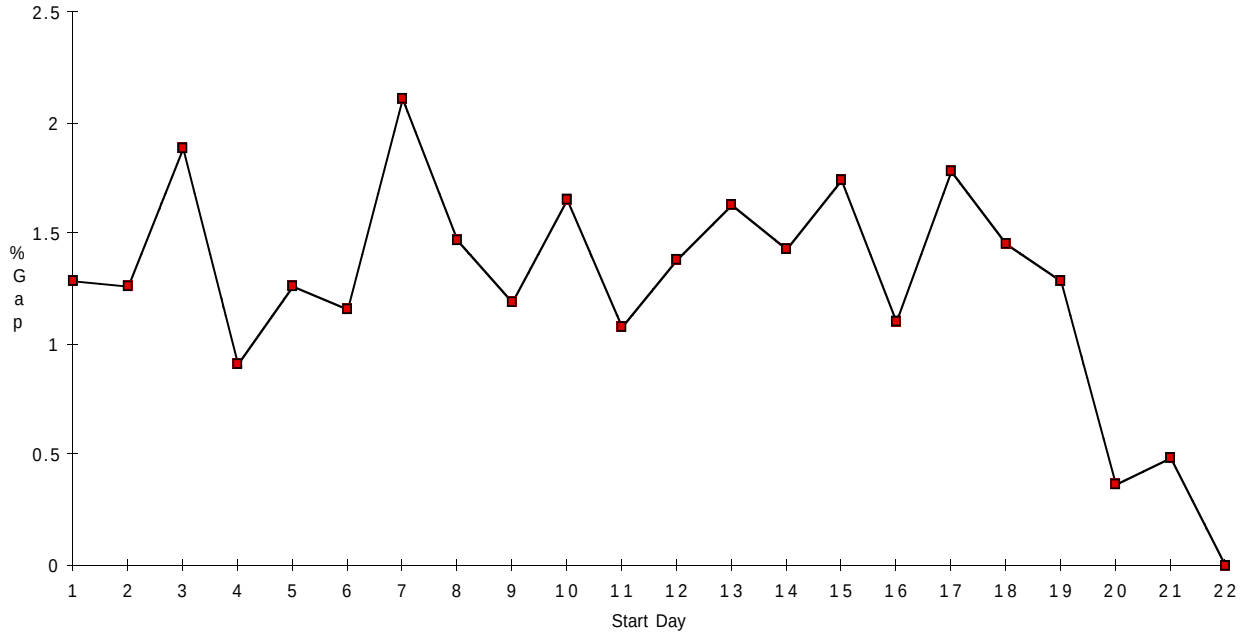


Figure 4.17 shows the TSP duality gap as a function of time. In all iterations except the first where no shipments have been originated yet, the primal heuristic assigns shipments already moving on the railroad based on their DCS trip plans. (These DCS trip plans are developed in the “Sweep Down with Look Ahead, Outside Iterations Off” mode.) The average TSP gap for the time interval 30 to 410 hours is 1.45%. After the start of day

19 (432 hours), the simulated TSP gap declines, because no addional traffic is added into the simulation past the cutoff of 500 hours. Note that since the primal objective function is calculated only based on the last TSP iteration, gaps reported in Figure 4.17 are only conservative estimates of the true gap. Actual gaps must actually be tighter than reported.

Fig 4.17 - TSP Gap by Day



4.6 Regulating the Application of Penalty Costs

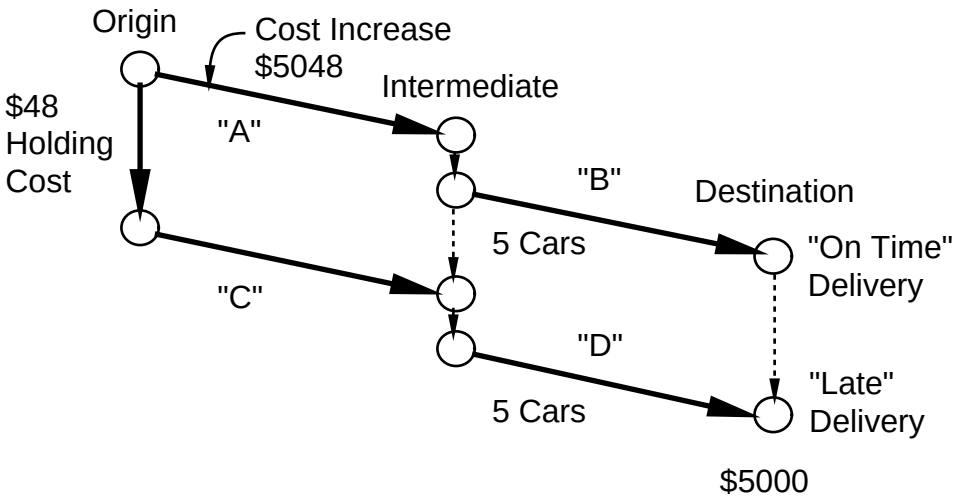
In the rolling horizon testing, although the vast majority of cars were delivered on time, some high priority shipments were still delivered late. This problem had not been apparent in any of the stand-alone tests of the car scheduling algorithms, but was only seen in the context of rolling horizon testing. The "brute force" approach of simply increasing the penalty cost did not significantly reduce the number of late deliveries. Ultimately, as shown in Figure 4.18, the root cause of the problem was traced to some very large adjustments to train segment dual prices made by the Dynamic Car Scheduling process *at the time the shipment was originally called in.*

These price adjustments cancel the effect of the penalty cost, neutralizing the penalty cost's ability to hold a high priority shipment on schedule. No matter how large the penalty cost, it could still be cancelled by a matching increase to the dual variable associated with an overcapacity train segment. Consider the example shown in Figure 4.18. Assume train "A" has only 5 cars remaining capacity. Now a large high priority shipment of 10 cars calls in having *no schedule slack* in the quotation based on TSP dual prices, and a $5000/car penalty cost for late delivery. This can happen because of the "open ended" nature of bid-price booking controls as reported by Williamson [1992] (pp 90-92), but the Dynamic Car Scheduling process will immediately detect if this causes a train capacity overflow and allow an appropriate management response, as will be further described here.

If no cars can be displaced from train "A", the Dynamic Car Scheduling algorithm must split this shipment keeping 5 cars on the first train "A" and displacing the remaining 5 high priority cars onto the next days' train "C". Given an hourly holding cost of $2, the diversion cost per car will be $48 holding cost plus the $5000 penalty. This gives a total $5048 price increase which must be applied to train segment "A" in order to *equilibrate path costs* and split the flow. After the $5048 price adjustment, the first five cars have a *zero*

diversion cost because path costs are now equilibrated via either trains “A” or “C.” If another shipment subsequently calls in causing train “B” to overflow, even a tiny increase to the cost of train “B” would be sufficient to drive these 5 cars off “A” onto train “C”, which would make all 10 cars late.

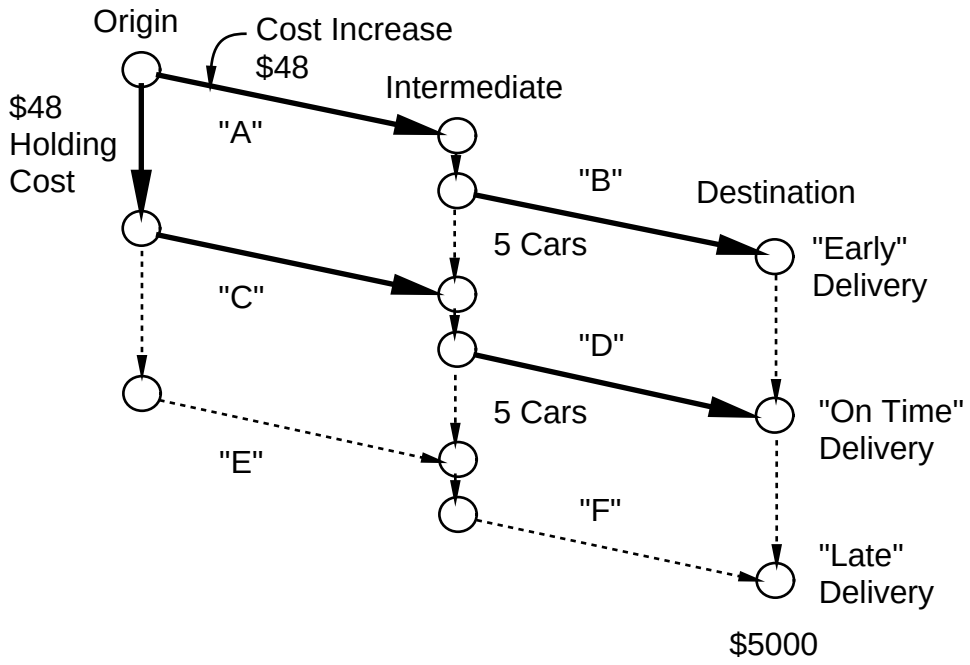
Figure 4.18 - Splitting A Flow with a Penalty Cost In a No-Slack Condition



The large price adjustment to segment “A”, caused by including the penalty cost as well as the hourly holding cost in the diversion cost calculation, sets up conditions allowing the five cars using “A” to subsequently be easily driven off schedule. There is nothing to prevent the remaining 5 cars currently assigned to “A” from also being diverted to “C”, because path costs have already been equilibrated via “A” and “C”. Furthermore, all 10 cars could easily be driven to an *even later* train (such as train “E” in Figure 4.19) — the additional diversion cost via train “E” would only be a modest $48 per car, because the fixed penalty cost has already been incurred, and there is no further penalty cost increase after the delivery commitment has already been missed.

By comparison, consider the case where the initial schedule quotation includes one days' slack time, as shown in Figure 4.19. Then an increase of only $48 to the cost of Train "A" would be sufficient to equilibrate the path cost. The $5000 penalty cost remains an effective barrier and is still able to prevent any cars being diverted to Train "E". In the test run of the Dynamic Car Scheduling algorithm (defined as "Scenario 3" in Chapter 5) presented in Figure 4.20, about half the split shipments were described by Figure 4.18, the other half were of the Figure 4.19 type.

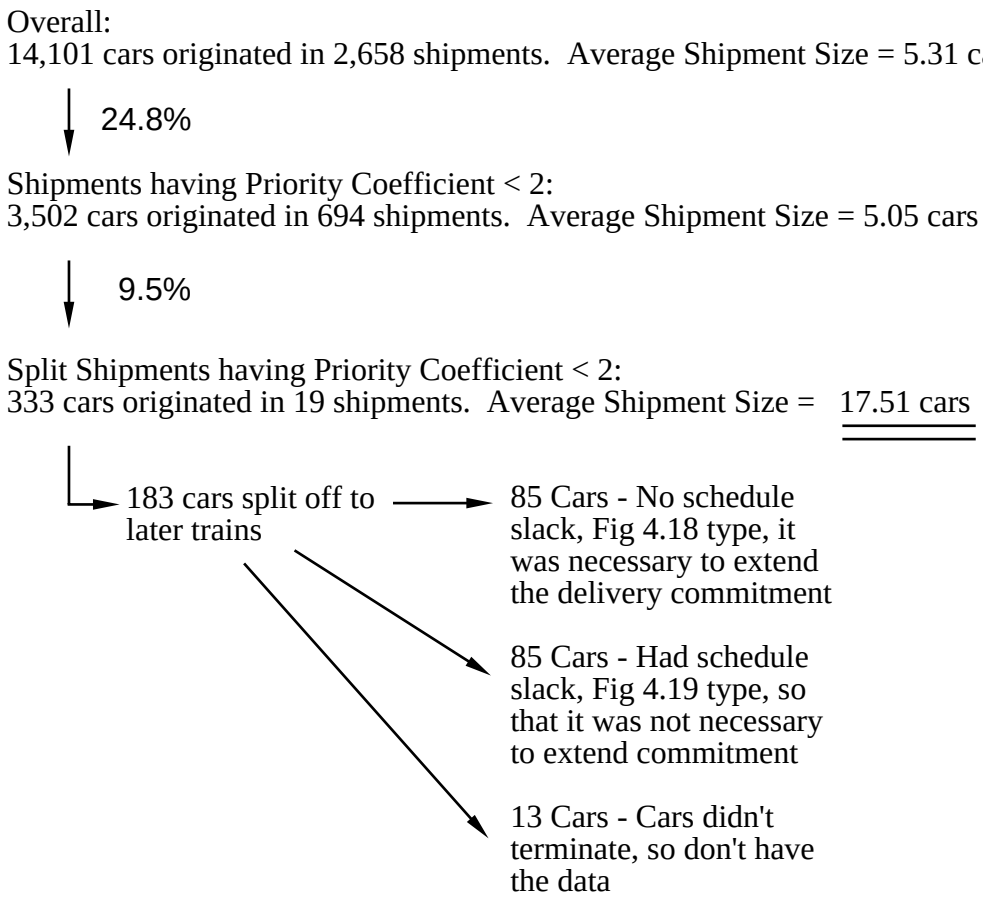
Figure 4.19 - Splitting A Flow with a Penalty Cost
One Days' Schedule Slack



It's important to note that this splitting of the shipment takes place *when the shipment is originally called in* . Probably the first option should be to determine if train capacity can be increased to accomodate the entire shipment without having to split it. If

increasing capacity is not an option, the customer should immediately be notified of the need to split the shipment, including planned delivery times for each part of the shipment. The customer could then decide whether or not to accept the rail carriers' service offer.

Fig. 4.20 - Split Shipments in Scenario 3 Test Run *



* In this special run of Scenario 3, any car having a logit priority coefficient less than 2 was considered a priority car and assigned a $5,000/car penalty cost for late delivery.

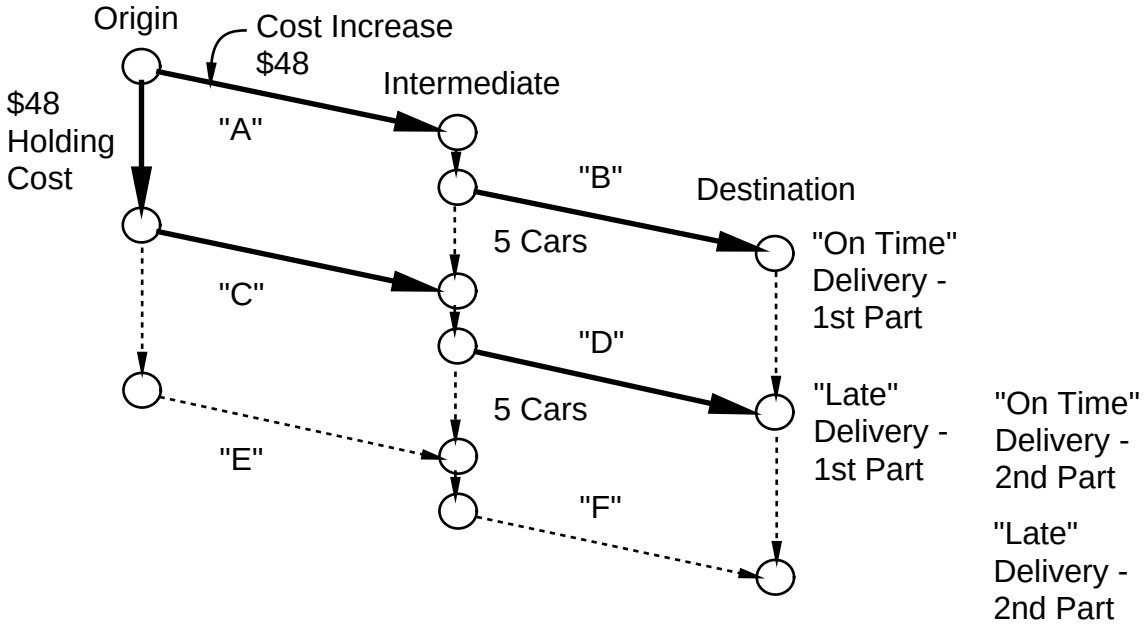
In the simulation when it was necessary to split a high priority shipment, average shipment size (17.51 cars) was much larger than the overall average of 5.31 cars. Clearly if a large number of cars are moving together as a unit shipment, to minimize switching the rail carrier would prefer to handle the movement as a single "block" and not split the shipment. Opportunities to adjust capacity, identified in the reservations process, should trigger a manual intervention if necessary to evaluate whether the rail carrier should adjust operations to meet demand, rather than automatically splitting the shipment and risking customer rejection of the service offer.

The reservations process can clearly serve as a "trigger mechanism" to drive a flexible train capacity management process. However, detailed simulation of this flexible capacity operation is beyond the scope of this dissertation. Also, in the event capacity *cannot* be increased, the Dynamic Car Scheduling (DCS) process still needs the ability to split priority shipments appropriately, without defeating the purpose of the penalty costs.

In the simulation, for newly called-in shipments the target delivery time constraint will be relaxed when necessary to eliminate the penalty cost component from the shipment diversion cost. The shipment can be split into two parts having *separate delivery commitments* for each part. This is permissible because the initial trip plan assignment is performed as part of the service quotation process, so the delivery commitment is not yet "locked in."

As shown in Figure 4.21, the first part of the shipment would still be scheduled onto train "B" while the second portion would be planned for delivery by train "D". Given these revised delivery commitments, only a $48 cost increase to train "A" is needed to equilibrate path costs and allow the overflow cars to be split off. The penalty costs remain effective in maintaining on-time delivery of *both portions* of the split shipment.

Figure 4.21 - Splitting A Flow with a Penalty Cost Extending the Service Commitment



A strategy of relaxing delivery targets to avoid excessive dual price adjustments is preferable to the alternative of maintaining the original unachievable delivery target, and applying the full penalty-cost driven increase to the dual variables. No purpose would be served by establishing an unachievable delivery target before the shipment even turns a wheel. Applying the full price increase would not permit any cars to be delivered sooner, but would only neutralize the penalty cost's ability to hold the shipment on schedule.

In a real world implementation, the carrier should assign the shipment to the train network using the Dynamic Car Scheduling process *before* making a service offer to the customer. This would confirm the feasibility of the service offer by ensuring adequate capacity is available to handle large multi-car shipments. It would identify if there is a need to either split such shipments and possibly extend the service commitment, or else if more

train capacity should be added to accomodate a large shipment without needing to displace other customers' loads.

However, in the rolling horizon model, performing the DCS assignment first would require programming the capability to "roll back" a DCS assignment, restoring the original system state, should a customer choose to reject the service offer. This would greatly increase the complexity of the simulation model code for only a very small gain in the accuracy of the simulation. As shown in Figure 4.20, only 85 high priority cars were split in such a manner as to extend the service commitment, out of 14,101 cars total in the simulation, or 0.6% of the total traffic. It is unlikely that more accurate simulation of this aspect of the service commitment process would have a material effect on the results of the analysis presented in Chapter 5.

CHAPTER 5

Rolling Horizon Simulation Testing

5.1 Chapter Organization

The Train Segment Pricing and Dynamic Car Scheduling algorithms, defined in the previous chapter, have been embedded into an event-oriented, rolling horizon simulation model *to assess their performance in the real-time control setting for which they were designed*. The model simulates the arrival and departure of road and local trains, pickup and setoff of cars at terminals, car classification within terminals, four variants of the trip planning process, and includes a highly detailed representation of the "booking" process whereby demand randomly materializes throughout the day. A test problem was adapted from Kwon [1994]; required modifications to Kwon's original problem are described in the Appendix.

In Section 5.2, each of the four trip planning scenarios are described. Section 5.3 describes how the model responds under each trip planning option. Section 5.4 proposes economic evaluation criteria to assess the profitability of each scenario, including the role and definition of penalty costs. Section 5.4 also defines a statistical approach by which two simulation outputs can be compared to estimate confidence intervals and determine the statistical significance of the results.

Section 5.5 presents testing results which compare the *operational* performance of the four different car scheduling scenarios. The following measures are reported: system throughput based on number of cars terminated; system stability based on enroute car inventory; train segment load factors; and transit time distribution relative to both the fastest possible time and to commitment, as a function of shipment priority. Our results will show that Dynamic Car Scheduling improves the operational stability of the system, increases throughput and improves train load factors; and for high priority traffic, it reduces transit time and reduces transit time variability.

Section 5.6 focuses on *economic* evaluation to assess the benefits of each implementation phase on an incremental basis. This section reports total contribution of each scenario; statistical convergence of contribution per day; confidence intervals and statistical significance of differences across scenarios. Our results will show that Dynamic Car Scheduling protects capacity for high contribution traffic, preventing displacement of highly profitable, service sensitive business by low-rated traffic, and improves overall throughput, thus improving the financial performance of the rail carrier.

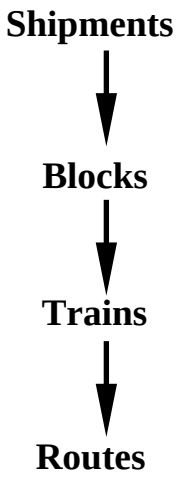
Finally, Section 5.7 presents two special scenarios. The first scenario examines the effect of the over or underbooking (in Scenario 4) on total system profitability. It is found that adjusting the "α" overbooking coefficient, as proposed in Section 3.6, has essentially no statistically significant effect on total system contribution. The second scenario explores the impact of incorporating penalty cost terms into the Dynamic Car Scheduling process when implemented stand-alone, as in Scenario 3. It is shown that incorporating penalty cost terms into DCS can be a very effective means of controlling service quality, in other words, a full blown implementation of TSP is not required in order to gain significant benefits from implementing DCS.

5.2 Definition of Test Scenarios

Any railroad network simulation model should address natural aggregation levels of rail traffic as shown in Figure 5.1. Typically, individual cars do not move directly over the tracks. First, cars are aggregated into blocks. Then blocks are further aggregated into trains. Trains may not follow the shortest path from origin to destination; instead, they might visit intermediate nodes to pick up or set off cars at enroute points.

In the railroad industry's current practice, cars are assigned to blocks without regard to whether the train is planned to operate on a given day or whether the train happens to be full. The primary objective of the dynamic car scheduling process is to couple this decision making by addressing all four levels of aggregation simultaneously and within a single model.

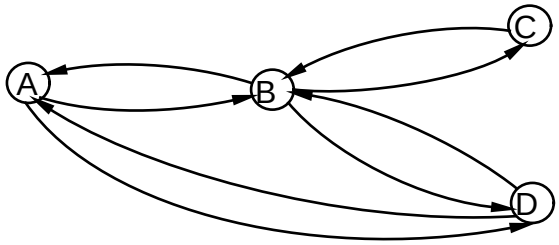
Figure 5.1: Levels of Aggregation in Rail Operations



Scenario 1: Current Business Practice

In this scenario, car to block aggregation is performed using a “flat blocking network” which does not include any time dimension. As shown in Figure 5.2, each yard classifies cars only to the next yard, except yard “A” builds a block directly to yard “D”, bypassing “B”, and vice versa. This approach is followed in ALK’s Automated Blocking Model, a planning model designed to assist in service design (Kornhauser and Mayewski [1983], Van Dyke [1986]).

Figure 5.2: "Flat" Blocking Network



Traditionally, car scheduling systems have used fixed table-driven car to block assignment. A table is required which lists all destinations assigned to that block. Replacing tables with a flat blocking network does not change this functionality at all, since the model is calibrated to replicate past table-based blocking policies as closely as possible. The main improvement is in data base maintenance, since it is no longer necessary to manually maintain the “tag table” or list of destinations assigned to each block. For Scenario 1, the simulation model will use a fixed blocking table input from an external file. For maximum consistency with Scenarios 2-4, this blocking table is generated from the space/time network

based on the fastest routing available at least 4 days per week. The optimization codes are not directly used in Scenario 1.

Block to train assignment is performed after the block has been determined, based on a "pick list" of blocks each train can carry. Cars are scheduled to the first available train which carries their assigned block, subject to minimum connection time criteria. Blocks are added to trains in a priority sequence (longest distance block first) until train capacity is reached; any overflow cars are rescheduled onto later trains. This represents the current state of the practice in the rail industry, as implemented in Norfolk Southern's "ABC" car scheduling system (Baugher [1993]).

Scenario 2: Uncapacitated Dynamic Network

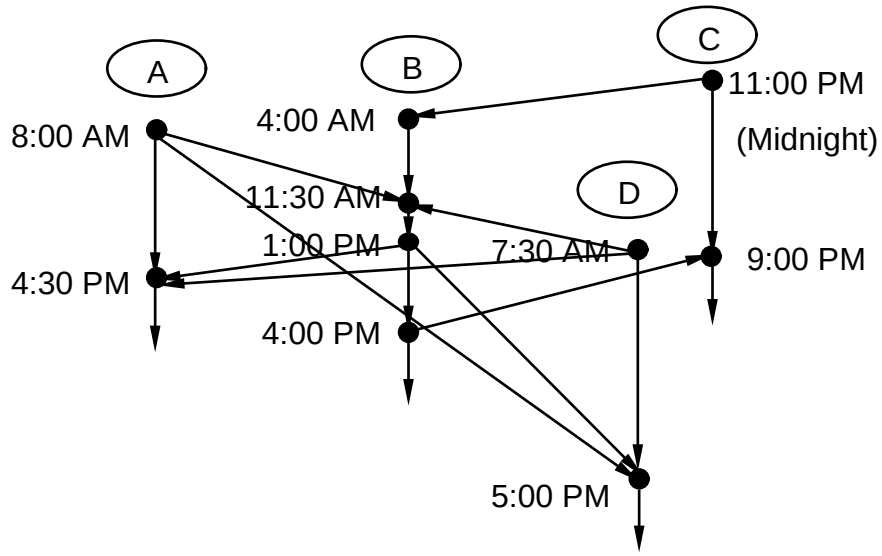
In a space/time or "dynamic" network, diagonal links represent planned train trajectories in space and time, connecting origin and destination nodes according to the blocks which are carried on the train. It is a three dimensional blocking network. Complete trip plans can be developed using a shortest path algorithm. This eliminates the need for a supplementary process to perform block to train assignment outside the network model.

Figure 5.3 shows a simple space/time network which can develop both block and train assignments within a single model. Vertical links represent cars waiting in yards for connections. Each node represents a specific time when some event occurs, such as the arrival or departure of a road or local train.

For example, location "B" has nodes at 4:00 AM, 11:30 AM, 1:00 PM and 4:00 PM. This pattern could be replicated to generate as many days' activity as desired, varied by day-of-week, or even customized to each day.

Each node may have zero, one or many over-the-road links associated with it, and has one vertical "holding" link connecting to the next node at the same location. A node

Figure 5.3: Dynamic Blocking & Train Schedule Network



having no associated over-the-road links marks the arrival or departure time of a local train, and is typically used as an entrance or exit point for traffic assignment onto the network.

To handle shipments of varying characteristics, different link cost coefficients based on mileage, time or yard handling cost can be assigned, or use of some links might be prohibited entirely, as for high/wide loads. Although the costs may vary, the basic structure of the network is determined solely by the train schedules and always remains the same for all shipments. Depending on the day of week, the model might choose to not only utilize different trains, but also route shipments using different blocks and through different intermediate yards. At this stage, however, train capacity constraints are still not enforced.

The ability to customize the train schedules on a daily basis, and to route shipments according to the train schedules, is what makes a dynamic network so attractive. A model based on a dynamic network would be much more flexible and responsive to day-to-day operating conditions than today's fixed table-based approach. In Scenario 2, shipment

routings are determined taking available connections into account, but individual train capacity constraints are still not enforced.

Scenario 3: Capacitated Dynamic Network

Current car scheduling systems can "overschedule" available train capacity such that the trip plans cannot all be achievable. Enforcement of train capacity constraints raises a subtle question: how does a car scheduling program recommend which shipments should be "bumped" off an overcapacity train? Implementation of these trip plans would require significant improvements to operational planning systems within rail terminals, so that trip plans can be used to actively drive the car classification process, rather than passively trying to "predict" which outbound connection a shipment might be likely to make.

Selection of priority shipments is not so difficult for intermodal, but requires more advance planning for railcars. Since selecting specific shipments for each outbound train may increase the complexity of terminal operations, there must exist a clear justification for asking a terminal to depart from the standing order sequence in which cars would naturally fall in the classification tracks. A justifiable basis for establishing terminal handling priorities would be based on meeting customer service delivery commitments. It doesn't matter whether commitments were determined by a real time quotation process or were contractually established long in advance.

While priority-based switching is complex, with proper planning and resource allocation within the terminal, it can be accomplished (Union Pacific Railroad [1995]). Numerous other benefits, not directly related to priority car switching, are also obtained within the scope of a comprehensive terminal decision support system. However, a detailed analysis of switching strategies is beyond the scope of this dissertation.

Scenario 4: Real Time Service Commitment Process

Contractually based service commitments work fine as long as the percentage of priority traffic remains low. But as the percentage of priority traffic increases, it once again becomes impossible to guarantee service reliability under a master scheduling approach. Since operating resources and capacity in the short term are fixed, a means of both limiting the volume of high priority traffic accepted and extending delivery times on low priority traffic are needed to maintain plan feasibility and to avoid service failures. The main impact of this process should be on those market segments having low service sensitivity. Highly sensitive traffic tends to receive a constant transit time quotation regardless of traffic conditions. This will be demonstrated through computational testing in this Chapter.

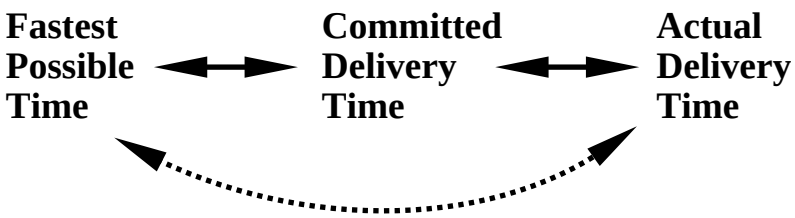
Conversely, under light traffic conditions, service quotations should be "tightened up" to take advantage of available capacity in the system. If the carrier is confident that an aggressive delivery schedule can be met, this would generally be in the carriers' interest so that the equipment can be unloaded sooner and released for another customer's use. The modified shortest path approach of Section 4.2.1 will suggest a delivery appointment time for each shipment. However, if this delivery time isn't convenient for the customer, under standard bid price criteria any delivery appointment time that yields a positive contribution should be accepted. This allows precision scheduling of shipment deliveries so that the customer can receive the goods exactly when and where they are needed.

Scenario 4 introduces a slack-based priority concept whereby penalty costs are assessed for late delivery. Lower priority traffic may receive a longer transit time quotation, but once a delivery commitment has been made, a penalty cost will be assessed to prevent late delivery. To avoid this penalty cost, a "low" priority shipment with no remaining schedule slack may sometimes take priority over a "high" priority shipment having remaining schedule slack relative to its delivery commitment.

This is better for customers than the fixed priority-based system studied in Kraft [1995], which does not develop any service guarantee at all for the lower-rated traffic classes. Under a fixed priority system, as shown by Kwon [1994] and Kraft [1995], service to lower priority traffic can be severely degraded if the percentage of high priority traffic is too high. Under a slack based priority system, service commitments can vary, but once a delivery appointment is accepted, this commitment is respected regardless of traffic "priority."

The performance evaluation of this scenario is more complex than the first three, since it requires a three-way comparison between fastest possible, offered and actual delivery time for each shipment as shown in Figure 5.4. In addition, a sensitivity analysis on the effect of booking limits in the Train Segment Pricing model will be performed. This allows for systematic "over" or "under" booking to see what would be the effect on service reliability, average load factor and overall system profitability.

Figure 5.4: Key Comparisons in Simulation Analysis



5.3 Simulation Model Description

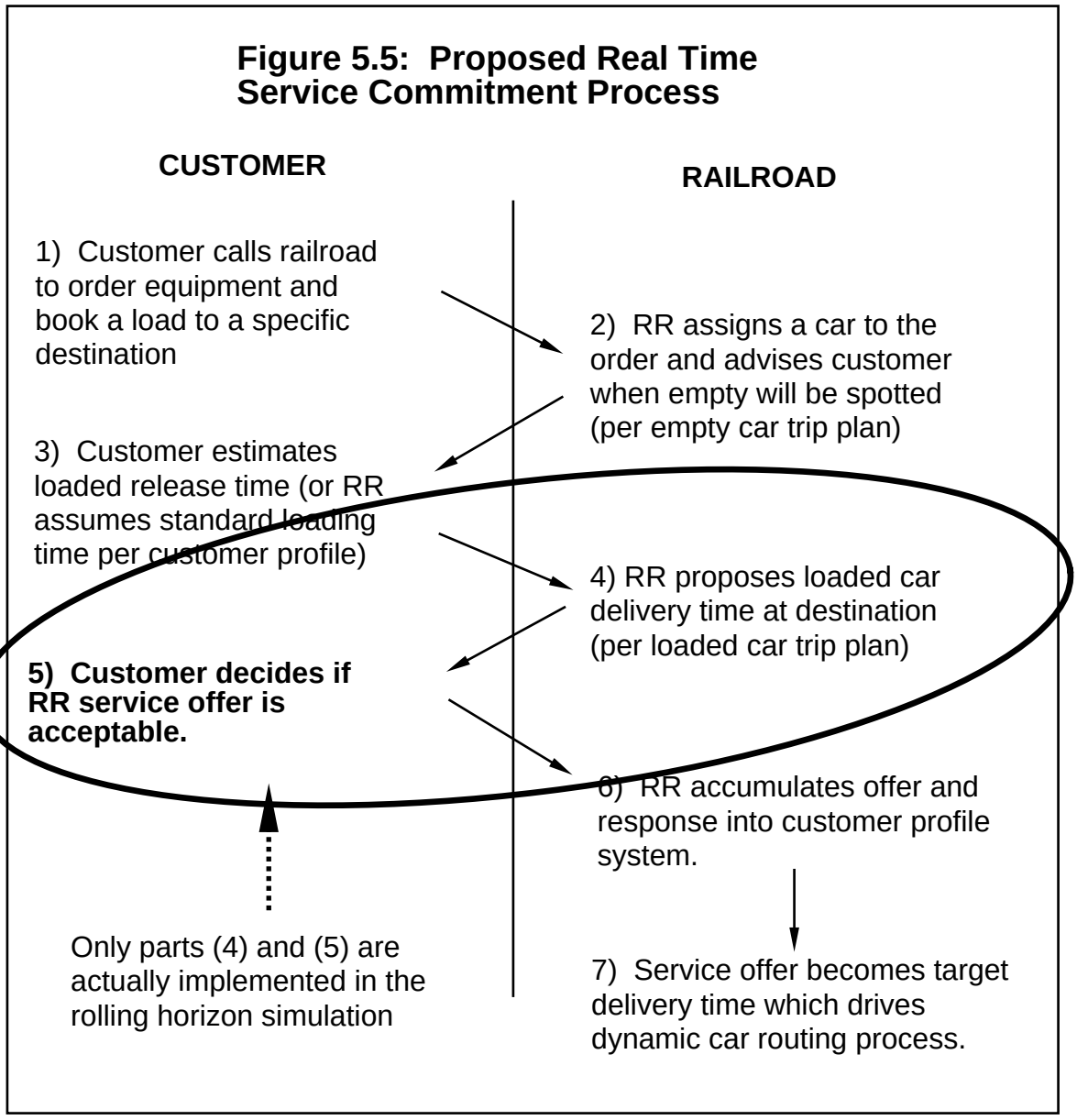
The simulation begins with random generation of traffic called in during the period 8 AM to 5 PM each day. The carrier has a forecast of *expected* origin-destination demand (which may include fractional values), but does not know the exact number of cars until

after each customer has called in their loads for the day. Based on this forecast, in Scenario 4 the Train Segment Pricing (TSP) model establishes dual prices for each train route segment at the beginning of each day.

These TSP dual prices are used to develop service commitments throughout the entire day. Although the demand forecast and TSP dual prices are not adjusted during the day, Dynamic Car Scheduling (DCS) dual prices are adjusted, if necessary, after each load calls in. If the current DCS link price rises above the value originally predicted by TSP, then the higher DCS price will be used. This partially addresses the requirement pointed out by Williamson [1992] that bid-prices need to be updated on a real time basis in order for implementation to be successful; although it is clearly not as good as adjusting the demand forecast and rerunning TSP, the DCS dual prices come practically "for free" so this approach can be implemented with little additional computational effort. Although DCS dual prices may influence the service quotation process, this doesn't hold true in reverse: TSP dual prices are not currently used in the DCS process.

The process for generating service commitments in the rolling horizon simulation model follows the general outline first proposed in Figure 1.2, reproduced as Figure 5.5 on the next page. The model implements the proposed process for performing Steps (4) and (5) in Figure 5.5; demand forecasting and empty car distribution are not addressed here.

In Scenarios 1-3, there is no provision for advising the customer ahead of time what the expected delivery time will be, except based on the shipment's current trip plan. In Scenarios 1-2, no reduced cost information exists; in Scenario 3, only DCS reduced costs are available. Without full knowledge of the reduced costs associated with overcapacity segments, the rail carrier cannot assess the profitability of offered shipments. Therefore, all offered shipments are accepted in Scenarios 1-3 whether profitable or not, or whether they can be delivered on time or not.



In Scenario 4, a *committed delivery target time* is offered for each shipment *at the time the shipment is first called in.* Delivery appointments to maximize profit are proposed using a modified shortest path algorithm. This takes the probability the customer will accept the service offer into account, implementing the procedures described in Sections 3.4 and

4.2.1 of this dissertation. For each link in the space/time network, the highest dual price from either the TSP or DCS current solutions is used, although usually the TSP prices dominate. Once a service offer has been formulated, if the offered delivery time is slower than the "base" time (the "normal" transit time developed with all dual variables set to zero), a random number generator is called to determine whether the customer accepts the service offer. If no delivery offer is profitable, the carrier will reject the load.

After a shipment has been "booked", a DCS trip plan is developed. If any train segment capacities have been exceeded, additional iterations of the DCS algorithm are performed to restore feasibility. If as a result of developing this trip plan, a newly called-in shipment must be split violating its committed delivery target time, the target time for each portion of the split shipment is adjusted following the procedures of Section 4.6.

Trip plans in the simulation model are developed as follows:

- In scenario 1, trip plans are based on a fixed blocking table; the first outbound train which carries each block after a fixed minimum processing time is used. Current industry trip planning processes modeled in this scenario are not sensitive to and do not require train-specific link costs as input.
- In scenario 2, the trip plan is developed using an uncapacitated, cost minimizing shortest path algorithm on a space/time network of blocks and trains.
- In scenario 3, trip plans are developed by the Dynamic Car Scheduling process, enforcing train capacities. Since delivery target times have not been established, no penalty costs for late delivery can be assessed. Shipment priorities are essentially determined by their hourly cost component; the minimization of total cost would tend to give priority to the most expensive car types and commodities.

• In scenario 4, the Dynamic Car Scheduling algorithm uses the TSP delivery time commitment to assess penalty costs for late delivery. Avoidance of penalty costs largely determines the shipment priorities. Slack time might be provided in a service quotation based on the *expectation* a higher priority shipment will call in later, but the initial trip plan still assigns the shipment to the most expeditious routing currently available. If a higher priority shipment calls in later, lower priority cars can be "bumped" or rescheduled at that time, if necessary, provided they haven't already been classified. This doesn't cause any problems provided sufficient schedule slack has been built into the service commitment, but avoids unnecessary delay in the event the higher priority cars don't materialize.

Once a delivery appointment time has been established and the initial trip plan developed, the penalty cost is *tripled* for the remainder of the life of the shipment. This increased penalty cost represents lost customer goodwill associated with not delivering a shipment on time after a firm commitment has been made.

Next, local trains bring the traffic into origin yards. After a fixed processing delay, the model simulates the resorting of cars into the appropriate outbound blocks. At the scheduled make-up time of the outbound train, cars are removed from yard inventory. The exact method of doing this depends whether the unconstrained scenarios (1) and (2) or constrained scenarios (3) and (4) are in effect:

• In scenarios (1) and (2) the trip plan is used only to control which *block* the shipment is classified into. When a train picks up cars, it first picks up the block which travels the longest distance, until either all available cars have been picked up or train capacity across some downstream segment is exceeded. Within each block, cars are picked up in a first-in-first-out sequence. After the first block is picked up, the process is repeated for all remaining blocks for the train at that location. If a train fills up, any cars missing planned connections must be "force rescheduled" to later trains. Block-level "locking" is

enforced: after a shipment has been classified, the block can not be changed, although a later train might be used if necessary.

- In scenarios (3) and (4), *both the block and outbound train* are determined by the trip plan. When an outbound train picks up cars, all the cars scheduled to that train are taken. Train capacity is never exceeded at pick up time, nor should there be any left over cars. Rescheduling occurs as soon as a higher priority shipment is called in, rather than waiting for a shipment to actually miss its outbound connection. Once a shipment has been classified, neither the block nor train can be changed.

The main purpose of locking the first leg of the trip plan in Scenarios (3) and (4) is to avoid the need to reclassify or "cherry pick" cars for specific outbound trains. Decisions on car scheduling must be made *before* an inbound train is brought to the "hump crest" and the switch list generated. After this, when cars are already switched into the classification tracks, it becomes prohibitively costly to attempt to change the decision. This assumes that yards will be able to build blocks on an outbound-train specific basis and separate cars for different outbound trains at the hump crest, if necessary. Some fairly simple strategies exist for doing this, such as sending cars scheduled to later trains to rehump tracks in order to keep the classification "clean" for one outbound train at a time. A more sophisticated strategy would manage block to track assignments in the bowl on a dynamic basis, to enable a greater number of blocks to be built in the same yard, and minimize the number of rehump cars. Or, additional constraints might be added into the DCS process to limit the number of blocks simultaneously active within a single yard.

The current "locking" assumptions might be overly restrictive. If a car is classified to a rehump track, then the locking need not apply to that car. However, the current locking rule provides an important secondary benefit: it guarantees positive space on the outbound train once a car has been classified.

This positive guarantee of train capacity has its primary benefit at intermediate terminals where a mainline train stops to pick up cars, or on short distance blocks which would be the last to be picked up. In scenarios (1) and (2), if a train fills up at the origin terminal, there may never be enough remaining capacity to pick up additional cars at an intermediate point, and shipments can become "stranded." Train locking overcomes this problem by reducing the number of cars scheduled to depart from the origin terminal and preserving just enough space to allow planned intermediate pickups.

The process repeats itself when a mainline train arrives at its destination, and its cars are set off into the rail yard. After a fixed processing delay, the cars are resorted into the appropriate outbound blocks. A considerable amount of simulation model code is devoted to "housekeeping" to simply maintain all data synchronized with current operating events. After a train arrives, the "used up" legs of shipment trip plans are written to a log file, and the memory associated with those legs is released. As well, any DCS tabu entries associated with the completed train leg can also be purged.

When the simulated event clock passes midnight, one additional days' activity is added onto the end of the network. Simulated events, such as arrivals and departures of road and local trains are scheduled. While the network is only extended once a day, deletion of old information is event-driven and occurs in real time. After all, once a train has terminated we do not want that train to be able to receive any more scheduled cars. Processing deletions on a real time basis frees up memory storage sooner and ensures that such illogical assignments cannot occur.To implement this code in a real world application would only require triggering network updates based on actual operating reportings rather than depending on the simulation timing routine to generate these events.

When a car reaches its destination, it is classified into a "local" block and picked up by an outbound local train at the scheduled departure time. At this point, the shipment

terminates, trip statistics are written to a log file, and all memory storage associated with the shipment is released.

As compared to Kwon [1994], this operational simulation is less detailed since it does not include switch engine assignments within terminals, delays due to unavailability of crews or power, or random delays to trains enroute causing missed connections. In this simulation, train arrivals and departures and car classification operations take place as scheduled. Any missed connections or late deliveries are solely due to capacity overflows and/or the operation of the Dynamic Car Scheduling process. This allows a controlled measurement of the effect of different car scheduling strategies apart from other factors which also influence service reliability. However, the modeling of the car scheduling and booking processes are much more detailed here than in Kwon's model.

5.4 Economic Performance Evaluation

The economic analysis developed here is somewhat restricted, as it is directly tied to the costs and revenues used in the Train Segment Pricing and Dynamic Car Scheduling optimization. A full analysis would best be developed using "real" rather than randomly generated input based on the actual network, traffic flows, required service levels, revenues and costs of a specific railroad.

Recall from Chapter 3 that the objective function of the Dynamic Car Scheduling model minimizes total cost, including penalty cost for late delivery:

$$\text{Min} \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \quad \text{for all } k \in K, (i,j) \in A. \qquad (3.3.2)$$

whereas the Train Segment Pricing model maximizes total expected profit:

$$\text{Max} \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \quad \text{for all } k \in K, (i,j) \in A. \qquad (3.4.1)$$

In the Dynamic Car Scheduling formulation (3.3.2) all link costs c_{ij}^k are positive, except c_{ij}^k coefficients on links incident to the super sink are nonnegative representing the penalty cost of late delivery. In the Train Segment Pricing formulation (3.4.1) c_{ij}^k's are all negative, except positive c_{ij}^k coefficients on super sink links represent revenues earned.

The two formulations use identical network structure and c_{ij}^k coefficients, except the signs on the c_{ij}^k coefficients are reversed. The main difference lies in the links connecting the shipment termination nodes with the super sink, where the Dynamic Car Scheduling formulation recognizes penalty cost, but the Train Segment Pricing model uses revenue along with the probability of customer acceptance of the service offer. Currently, penalty costs are only charged in the event a shipment is delivered late, although the mathematical formulation for Dynamic Car Scheduling does permit penalty costs for early delivery as well.

Penalties for late delivery are not used in Scenarios 1-2 because they would not influence shipment priority or routing in these uncapacitated car scheduling processes. In Scenario 3, no explicit delivery appointments have been established, so penalties are normally not used. However, two special runs of Scenario 3 will use penalty costs to implement a two tier priority scheme based on a high priority traffic subset. An arbitrarily large penalty cost ($5,000 per car) enforces compliance with the trip plan on the top 20% to 25% highest priority cars.

In Scenario 4, a sliding-scale penalty cost is calculated for *every* shipment based on its delivery appointment, profitability and on the service sensitivity of the customer. The penalty cost is calculated using:

$$\text{Penalty Cost}^k = (1 - \Psi^k)(R^k - C_B^k). \qquad (5.1)$$

where:

Ψ^k = Customer acceptance probability if shipment "k" is delayed 24 hours (computed from logit function) as compared with the "normal" transit time.

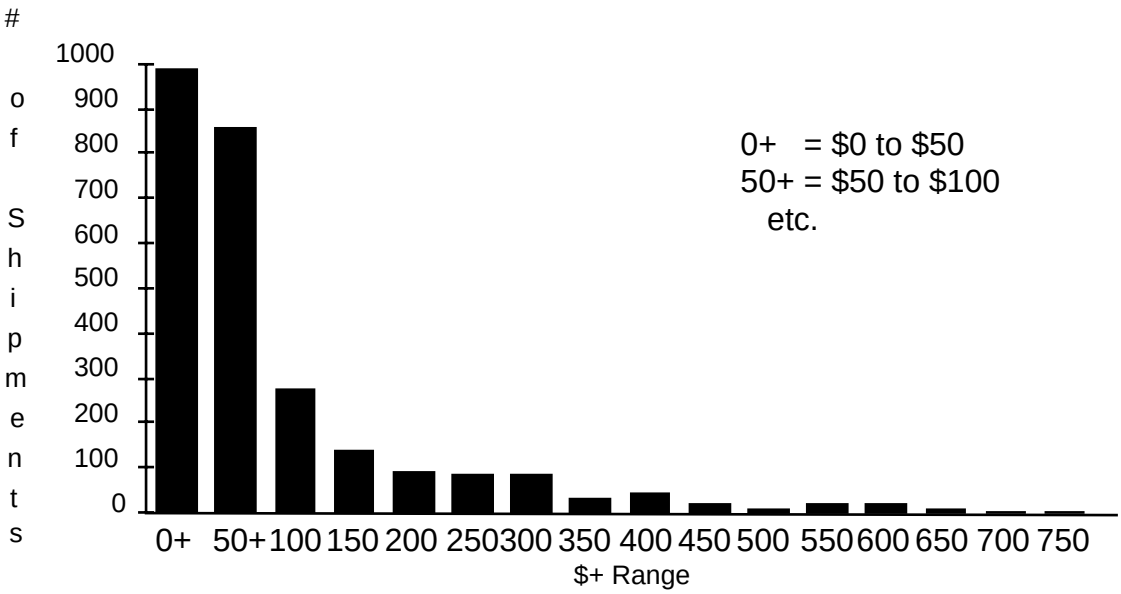
R^k = Revenue of shipment "k" if customer accepts offer.

C_B^k = "Base Cost," the cost of the shortest path assignment with all dual variables set to zero.

Equation 5.1 is the same as the first term of the "Expected Impact of a 24-hour delay" formula, already given in Step 1 of Section 4.5.2. It calculates the expected contribution loss if the service offer is extended by one day during the reservations process. Section 4.5.2 also proposes a second term which adds the hourly cost component of a 24 hour delay. This is already included in the shortest path calculation, so to avoid double counting of the hourly cost, only the first term is used here.

Using Equation 5.1, penalty costs in Scenario 4 average only $110, and the largest penalty is $741. The calculated distribution of penalty costs is shown in Figure 5.6. Once a shipment has been accepted and starts moving, this penalty is *tripled* to account for the lost goodwill associated with offering a customer commitment and then not meeting it. Still, this produces a much lower penalty than the $5000/car cost used in the Scenario 3 special run, so the model in Scenario 4 can actually *make an economic tradeoff comparing the benefits versus cost of violating any given service commitment.* This means that the formulation treats the delivery time commitment as a "soft" constraint which can be violated for a price.

Fig 5.6 - Distribution of Penalty Cost



Of course, if a carriers' policy is to never violate a delivery commitment, this can be accomplished by arbitrarily establishing an extremely high price: as is done in the Scenario 3 special run, where there is *no intention to make an economic tradeoff* — the extremely high cost essentially guarantees that high priority cars can never be displaced.

While the proposed approach is certainly not the only way to assess penalty cost, it does take into account both the profitability of the load and the service sensitivity of the customer. Currently, a step function penalty cost is assessed in the model. No penalty is assessed for early or on time delivery, but a fixed charge is made for any late delivery. Once a delivery commitment has been missed, this causes the model to assign a higher priority to new loads which still have an opportunity to arrive their destinations on-time. A different functional form, such as a linearly increasing penalty cost as a function of lateness, could also be used. An increasing penalty cost would tend to advance the older shipments first even though they have already missed their delivery commitments, often at

the expense of later arriving shipments which may still have a chance for on time delivery. The mathematical formulation of the Dynamic Car Scheduling model is very flexible, and can handle any kind of penalty cost function. The precise form of the penalty function can be determined by policy and is easily adjusted to suit the needs of a specific carrier.

Given that each shipment has a movement cost across each link, a holding cost at each yard, and penalty costs computed using equation 5.1, each of these variable costs excluding penalty cost are accumulated by the simulation model based on links and yards actually traversed. The final result is that revenue minus direct variable costs is reported as the contribution of each scenario. *Penalty costs are excluded from the economic evaluation*: this analysis reports "real" direct revenue or cost changes only. Since the train service network is the same across all scenarios, costs such as train crew expense are fixed and do not enter explicitly into the car scheduling model formulation. These costs could be accounted for by adding a fixed dollar amount into each simulation result. Since our objective here is to identify the *differences* across scenarios, such fixed costs would cancel out and therefore do not need to be considered here.

Since identical traffic data are used in each model run, following Kraft [1988] (pg. 153), simulation results can be directly compared for each shipment. Comparing the means of "n" paired observations is a standard statistical problem. Define:

a_k = Contribution of shipment "k" from run "A"

b_k = Contribution of shipment "k" from run "B"

Then take the difference in contribution, delay, or any other comparable measure before and after, for each shipment "k":

$$d_k = a_k - b_k,\ k = 1, \ldots n$$

Compute the average change in the measurement per shipment:

$$\overline{D} = \sum_{k=1}^{n} d_k / n$$

Estimate the Variance using:

$$VAR(A-B) = \sum_{k=1}^{n} (d_k - \overline{D})^2 / (n-1)$$

and $$_d = \sqrt{VAR(A-B)}$$

Finally, construct a confidence interval for:

$$\overline{D} = \overline{A} - \overline{B}$$

using the t-statistic having (n-1) degrees of freedom. If n > 50, “c” values derived from the Normal distribution can be used, as given in Table 5.1.

Confidence Interval Range $\overline{D}$ +/- c $_d$ $\sqrt{n}$

Table 5.1 - "C" values from Normal Distribution

Level of Confidence	Value for "c"
85%	1.44
95%	1.96
99%	2.58

5.5 Simulation Test Results: Operating Performance

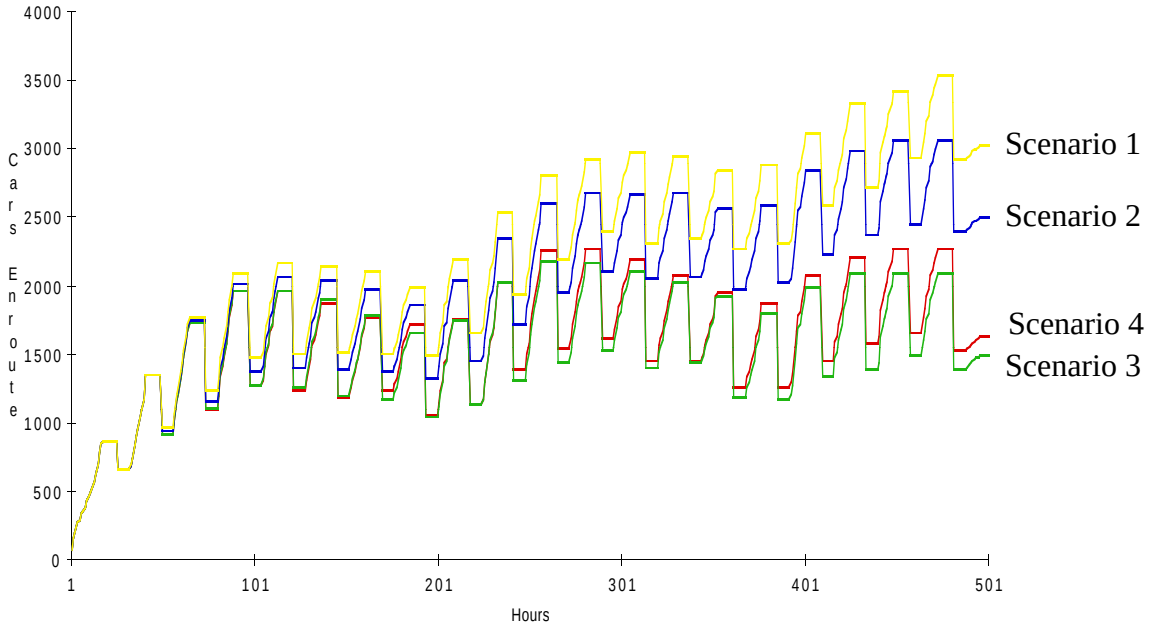
This section compares the performance of four different car scheduling strategies defined in Section 5.2. System stability, throughput, train segment load factors, and transit time distributions as a function of shipment priority will be compared across all four scenarios. For consistency, all four simulations measure service performance relative to the same “base” delivery time (developed using a shortest path calculation with all dual

variables set to zero), using a MIT traffic dataset derived from Kwon [1994] (as described in Appendix B) with normal train capacities and no special run options.

Following this cross-scenario comparison, train capacities will be varied within each scenario, and the effect of special run options in Scenarios 3 and 4 will be evaluated. In scenario 3, the DCS algorithm can be used on a stand-alone basis to expedite service to a high priority traffic subset. In scenario 4, service can be measured relative to the real time commitment as well as relative to the "base" delivery time.

Figure 5.7 shows the total enroute inventory of each of the scheduling scenarios as a function of time. The dynamic car scheduling Scenarios 3 and 4 reach a steady state inventory level after about three days, averaging between 1600-1700 cars. A weekly traffic pattern is readily apparent along with spikes representing the daily arrivals and departures of cars. Shipments call in during the interval 8 AM to 5 PM each day, but terminate on local

Figure 5.7 - Enroute Inventory by Scenario



trains at 1 AM. Since cars scheduled to originate after the simulation shut-down are not added to on-line inventory, this explains why the last daily traffic "spike" appears cut off.

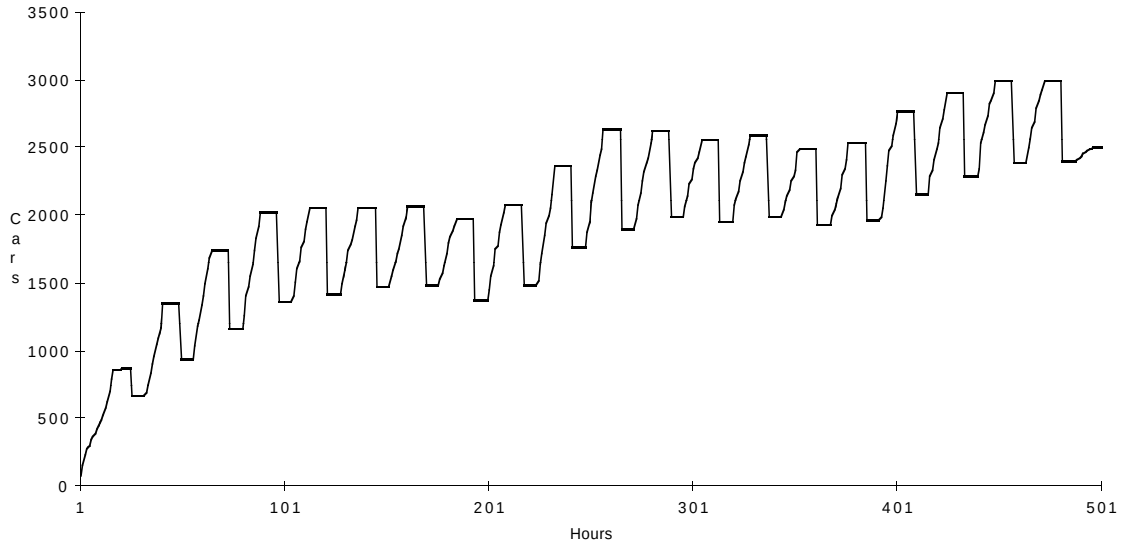
In contrast, in Scenarios 1 and 2, inventory continues to accumulate, with peak inventory exceeding 3500 cars in Scenario 1 and continuing to grow. Scenarios 1 and 2 have already exceeded capacity, whereas the Dynamic Car Scheduling process in Scenarios 3 and 4 is able to dynamically reroute traffic to keep all the cars moving towards their destination on a priority basis, and avoid accumulation of excessive inventory in terminals.
The accumulation of inventory in Figure 5.7 in Scenarios 1 and 2 largely results from localized bottlenecks, where certain train segments have exceeded capacity, although other trains or blocks might still be underutilized.

Fixed table based blocking is not able to adapt car routings to take advantage of available capacity elsewhere in the system. As a result, in real world rail systems, frequent manual interventions are required to add capacity to clear out excessive accumulations of cars, or else excess capacity must be built into the operating plan to begin with. In contrast, the Dynamic Car Scheduling process automatically reroutes cars to avoid capacity bottlenecks, better utilizing capacity already in the system. Dynamic Car Scheduling greatly reduces the need for manual intervention to keep the system fluid and improves overall capacity utilization, reducing the need to add extra trains or second sections.

The MIT traffic dataset described in the Appendix represents an extremely capacity constrained, challenging test problem. A 15% capacity reduction as in Figure B.3 in the Appendix would be sufficient to throw the Dynamic Car Scheduling simulation out of equilibrium, causing a slow accumulation of enroute inventory shown in Figure 5.8.

If capacity is reduced 15% across-the-board, the increase in train load factor is not proportional to the reduction of capacity. Some critical links already near capacity in the base case "meter" traffic to downstream trains, preventing them from attaining full capacity

Figure 5.8 - DCS Inventory in 15% Reduced Capacity Case



utilization. Comparing Figures B.2 versus B.3, utilization of segment 5-4 actually *declines* from 54.3% to 52.2%, because higher priority cars are available to be moved from location 4, preventing some low priority cars from departing terminal 5.

Another consequence of flow "metering" by high load factor segments is an increase in the number of shipments stranded enroute at the end of the 500 hour simulation. Out of 14,101 cars originated,1,499 cars remained on line at the end of the base case simulation, increasing to 2,503 cars after train capacity was reduced by 15%. This is consistent with the increase of enroute inventory shown in Figure 5.8.

At extremely high load factors, metering occurs despite the ability of the Dynamic Car Scheduling algorithm to flexibly route traffic. If all possible paths are saturated (as they are between locations 3-2 in this example) the Dynamic Car Scheduling process still ensures that the *highest priority* shipments are the ones moved. If the test network were more highly interconnected, the algorithm would find alternative paths for the lower

priority cars. But since no alternative paths exist in this test network, low priority shipments remain stranded.

Figure 5.9 gives the number of cars stranded and average cost per hour as a function of when the shipment was originally called in. The oldest stranded shipments were called in during the 320-370 hour interval having a very low average cost of approximately $3 per hour. All the high priority shipments called in during this time interval have already reached their destinations. Near the end of the simulation, the average cost rises into the $7 range because some high priority shipments simply haven't had sufficient time to reach their destinations before the simulation abruptly shut down. During the 440-460 hour range, the algorithm gives priority to the most expensive cars, causing the weighted average cost to dip slightly during that time interval. On the last simulated day, cars scheduled to originate after the simulation shut-down are not added to on-line inventory. This explains why not many cars are stranded originating in the 470-490 hour time interval.

Figure 5.9 - Stranded Shipments

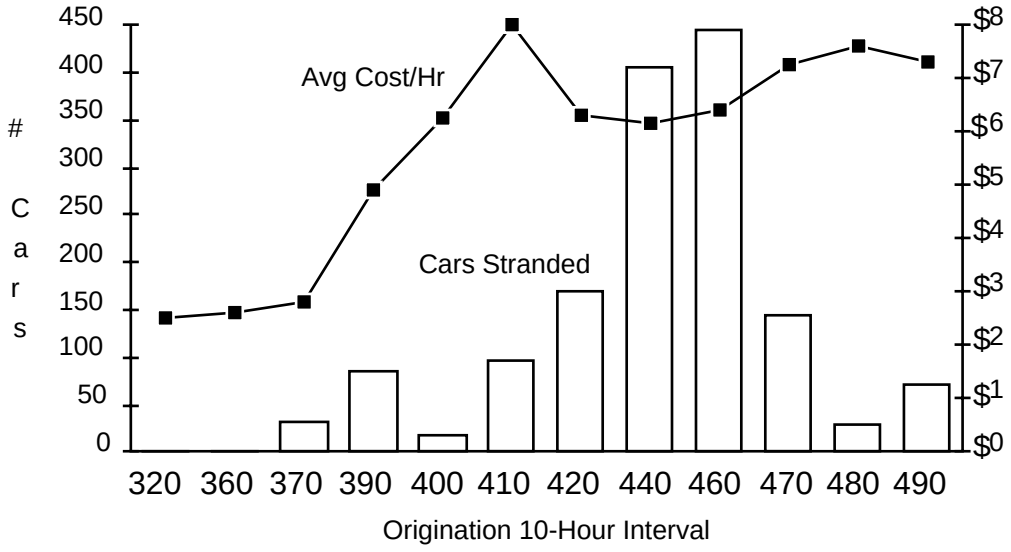
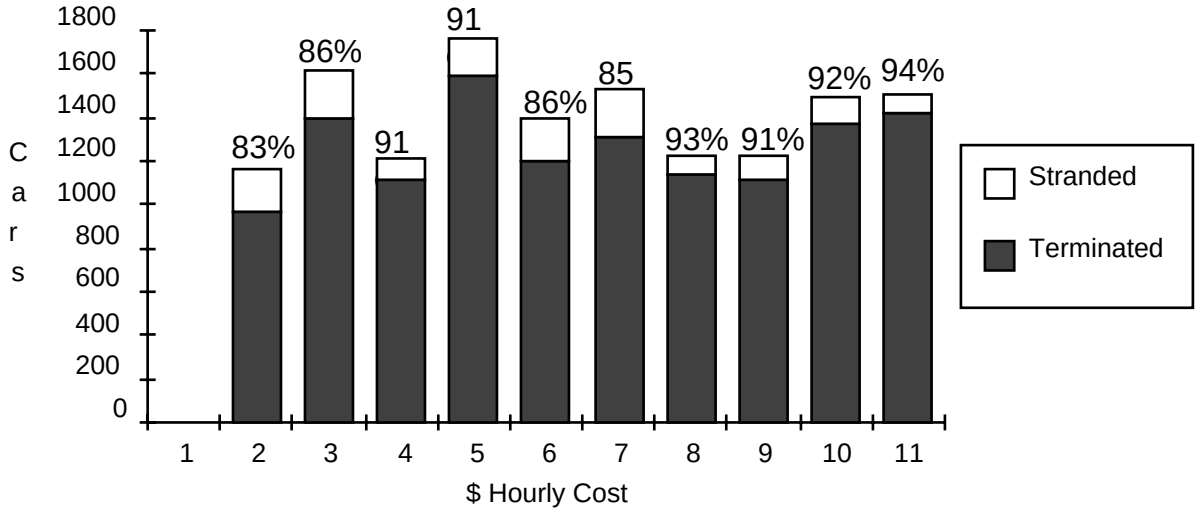


Figure 5.10 shows the number of cars originated versus terminated in the base case, as a function of cost per hour. A higher percentage of expensive cars reached their destinations within the simulated 500 hour time period, indicating these shipments received preferential treatment.

Figure 5.10 - Cars Originated vs Terminated

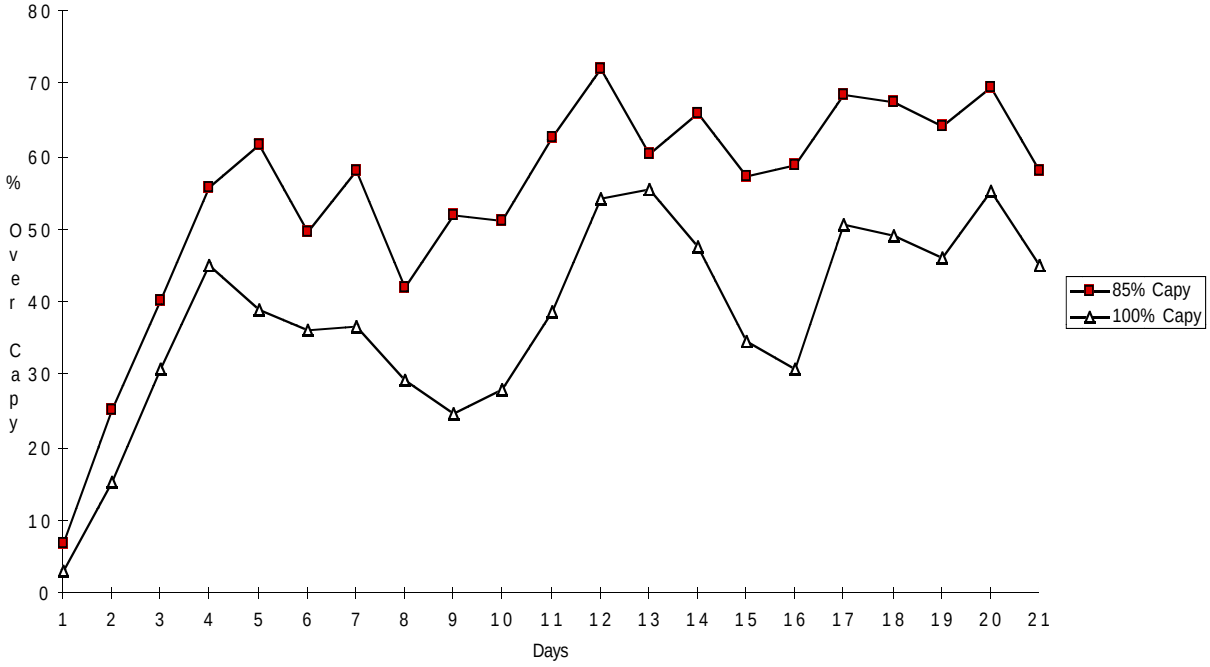


Another measure of the difficulty of a scenario is how many times the initial shortest path trip plan assignment exceeds available train capacity. Whenever the initial assignment produces a train capacity overflow, the Dynamic Car Scheduling process must be called to restore feasibility. As shown in Figure 5.11, in the base case, the Dynamic Car Scheduling process was called 975 times for 2658 shipments, or 37% of the time. On average, the reduced capacity scenario called the Dynamic Car Scheduling process 53% of the time, but needed it as much as 72% of the time on day 12. The weekly cycle of demand is readily apparent in Figure 5.11.

This comparison shows there is not much room to reduce capacity in the base case scenario, and still have a statistically meaningful performance evaluation. The base case

scenario is very close to maximum network capacity and represents a very challenging test for the rolling horizon model.

Figure 5.11 - DCS Call % by Day



5.5.1 Transit Time Comparisons versus "Base ETA"

Determining the impact of Dynamic Car Scheduling on service reliability is complicated by the fact there are two possible bases of comparison. The "Base" Estimated Time of Arrival (ETA) provides the most consistent comparison, since the same "Base" ETA applies to any given shipment regardless of which scheduling option is currently in effect. The "Base" ETA is calculated for each shipment based on a shortest path assignment with all dual prices set to zero, which usually gives the fastest possible transit time. This is the same as the original trip plan developed under Scenario 2, the uncapacitated dynamic network, in which train segment dual prices are never adjusted.

A second possible basis of comparison is relative to the shipment's *committed delivery target time* :

- In Scenarios 1 and 2, the committed delivery target time does not exist, so transit time comparisons are only performed relative to "Base" ETA's.
- In Scenario 3, the original trip plan ETA is captured after DCS has restored feasibility. Although the same dynamic network is used in Scenario 2, segment dual prices are adjusted and train capacity constraints enforced in Scenario 3: so the committed delivery target time can be different from the Base ETA.
- In Scenario 4, the committed delivery target time is based on either TSP dual prices, or the original DCS ETA, whichever is later, as described in Section 4.6.

Performance comparisons relative to the committed delivery target time will be performed in Section 5.5.2.

Figure 5.12 shows the transit time distribution relative to "base" transit time for each of the four car scheduling scenarios. *This shows Scenario 4 performs best* by terminating 10,069 cars at their base transit times, versus only 7,365 cars in Scenario 1. The Scenario 4 simulation also produced the "tightest" transit time distribution yielding a maximum three day spread versus base ETA.

- The transit time distribution from Scenario 1 has a long tail of seven days, resulting from the fixed table-based blocking algorithm's inability to find alternative paths around train segment capacity constraints (see Table C.1 in the Appendix.)
- Mixed results were obtained from Scenario 2, where transit time improved for most shipments, but some shipments suffered where excessive traffic was concentrated onto a few links without regard to train capacity. If a particular path offers the most rapid transit time, all cars will be routed that way, regardless of volume or train segment capacity. If the path is already saturated, more cars only worsen the train capacity overflow and increase

the delay time. Uncapacitated shortest path routing in Scenario 2 can be even more "myopic" than fixed table based blocking, since the transit time distribution has an 11 day tail, the widest spread of any scenario (see Table C.2 in the Appendix.)

- The transit time distribution for Dynamic Car Scheduling in Scenario 3 is strongly dependent on the hourly equipment cost. The worst case spread is a 5 day delay but the distribution is very "tight" for the more expensive equipment types (see Table C.3.)
- The Scenario 4 transit time distribution is very "tight" with a maximum delay of only 3 days beyond the base, which is strongly influenced by penalty cost, but not by hourly equipment cost. (see Tables C.4 and C.5 in the Appendix.)

Figure 5.12 - Transit Time vs Base Delivery

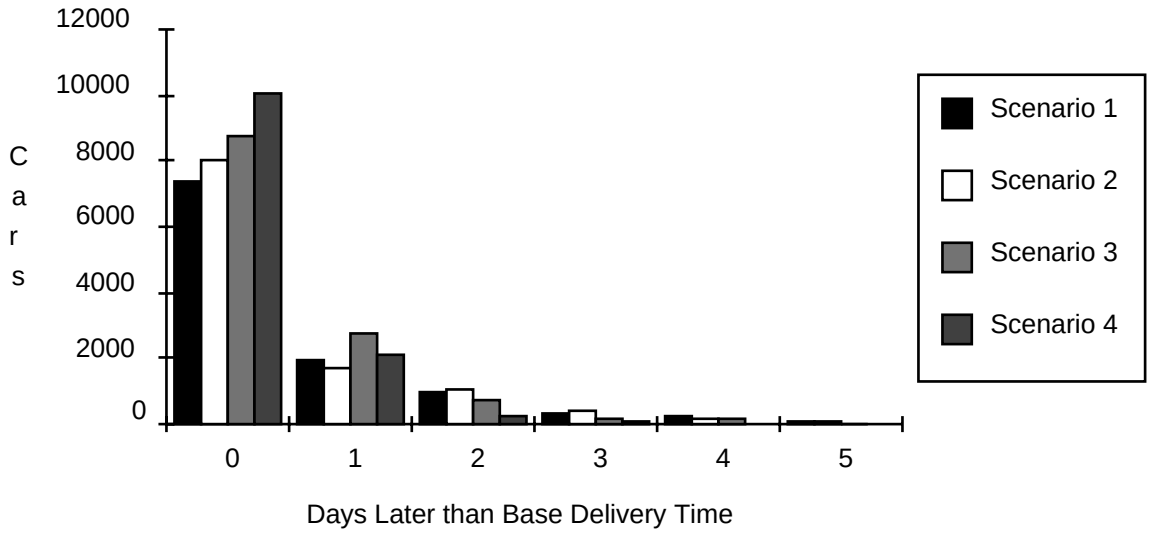


Figure 5.13 shows the average hourly cost of shipments as a function of days delivered later than base ETA. In uncapacitated Scenarios 1 and 2, transit time is essentially independent of hourly cost, but in capacitated Scenarios 3 and 4, the distribution is skewed to produce more rapid transit times for expensive shipments, and longer times for lower cost shipments. The delayed shipments have a very low average cost per hour.

Figure 5.13 - Days Later than Base vs Average Hourly Cost

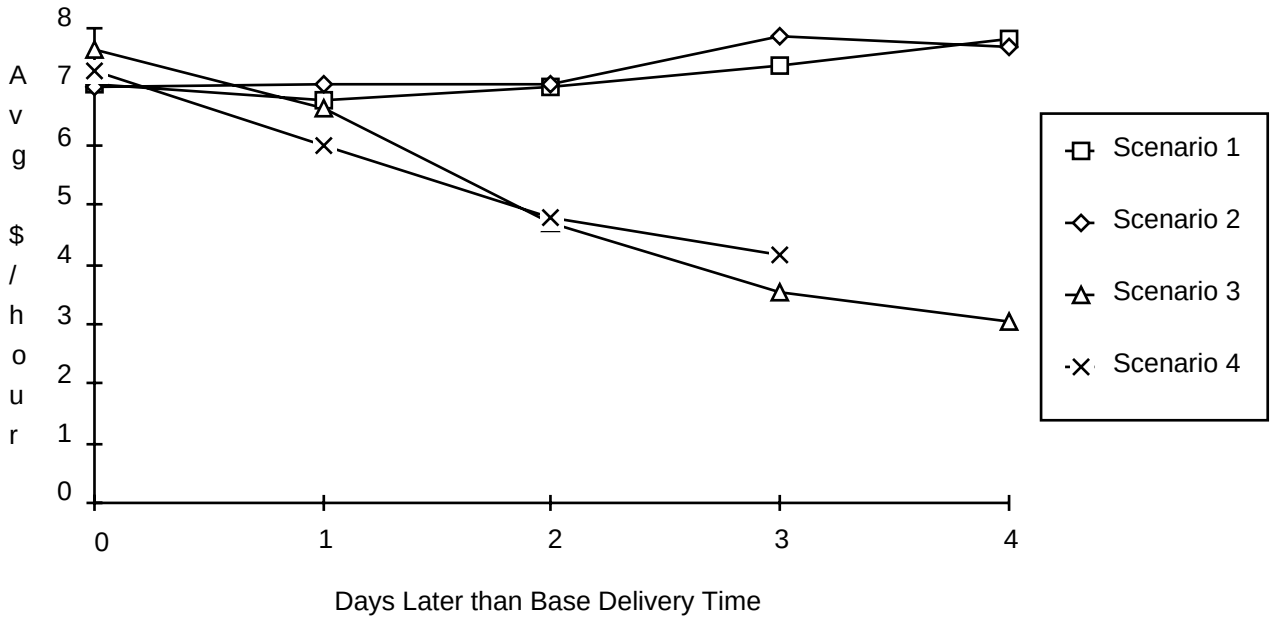


Table C.4 in the Appendix shows that transit time distribution in Scenario 4 is not as strongly correlated with hourly cost compared to Scenario 3, since some inexpensive cars are treated as priority shipments, although some influence is still clearly apparent. The correlation in Scenario 4 is much stronger with penalty costs, as shown in Table C.5. Figure 5.14 shows how a higher proportion of shipments with high penalty costs are delivered by the Base ETA. The number of cars with high penalty costs is a relatively small proportion of the total, but service reliability on these shipments is excellent. As shown in Table C.5, no shipment having a penalty cost greater than $500 was delivered later than its base time. The vast majority of cars delivered later than base had penalty costs under $100.

Fig 5.14 - Scenario 4 Deliveries

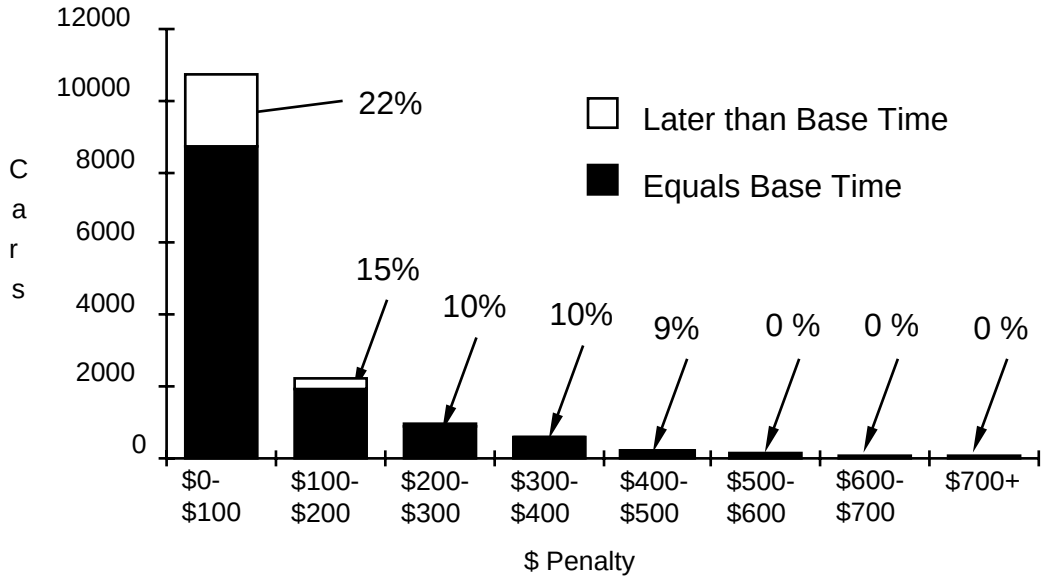


Fig 5.15 - Days Later than Base vs Penalty Cost

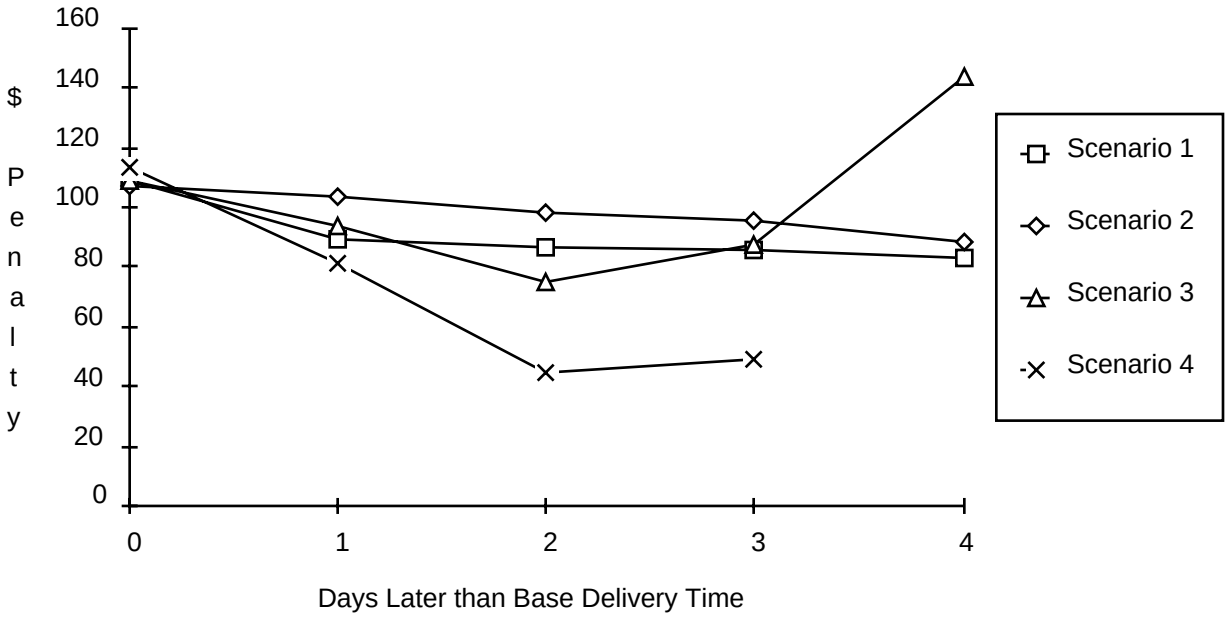


Figure 5.15 shows the average penalty cost for each scenario as a function of the number of days delivered later than base time. In Scenario 4, cars delivered later tend to have lower penalty costs. In the other scenarios, delivery time is independent of penalty cost. The average cost increase on day 4 in Scenario 3 appears to be a statistical aberration resulting from a very small sample size. These 183 cars have a very low average cost per hour of only $3. The penalty cost is not used in Scenarios 1,2 or 3.

5.5.2 Transit Time Comparisons versus "Commitment ETA"

Using a "Commitment ETA," performance is compared to a customized delivery time for each shipment. Thus, if the rail carrier anticipates a delay as part of the initial booking process, and the customer agrees to it, the delivery commitment for that particular shipment can be extended. This measurement is only developed for scenarios 3 and 4.

The carrier would only propose postponing a delivery appointment time if the shipment is not exceptionally profitable and if the carrier believes the customer has a high probability of accepting the delay, as described in Section 3.4. Table C.7 shows the number of days by which the transit time was extended in Scenario 4 (relative to base time) when a new shipment called in. Very few cars with penalty cost greater than $300 had any time added to their service offer. No shipment had more than 3 days added. In the vast majority of cases, cars with extended delivery commitments had penalty cost less than $100, and the Base ETA was only extended by one day.

In Scenario 3, often the Dynamic Car Scheduling process can anticipate a shipment delay at the time of original booking. In Table C.6, 10,568 cars were delivered on or before their initial trip plan ETA's, versus 8,732 cars delivered by their base transit times in Table C.3. This improvement of 1,836 cars represents cases where the delayed delivery was anticipated up-front by DCS, and the initial trip plan ETA was captured at the time when the shipment first called in. This produces an 84% overall on time delivery

rate, improved from 69% when delivery time is predicted just using Base ETA. The on time delivery percentage improves because the DCS planning algorithm can predict capacity-caused delays which are not seen by current car scheduling processes.

Table 5.2 shows service performance *relative to TSP commitment* in the Scenario 4 simulation. It should be apparent that the simulated transit time distribution is extremely "tight." Almost all transit time variability was incurred by cars having low penalties less than $100. Large penalty costs are not required: using the improved DCS logic described in Section 5.6, small penalties on the order of $100 are generally sufficient to hold a shipment on schedule.

Table 5.2 - Performance vs Commitment by Penalty Cost for Scenario 4

		Days Early (-) or Late (+) vs Commitment					
Penalty Cost	Row Summary	-2	-1	0	1	2	3
$0-$100	8756	19	280	8006	356	84	11
$100-$200	1907		1	1900	6		
$200-$300	854		7	847			
$300-$400	544			544			
$400-$500	205			205			
$500-600	122			122			
$600-700	55			55			
$700+	22			22			
Total Cars	12465	19	288	11701	362	84	11

5.6 Simulation Test Results: Economic Performance

Table 5.3 gives the total revenues, contribution and standard errors associated with each scenario. Revenues and costs are accumulated from all cars reaching their destinations before the end of the 500 hour simulation. Revenues, of course, are given directly for each

shipment; costs are accumulated based on the sum of c_{ij}^{k} costs defined for every link actually traversed. Any link cost adjustments due to dual variables, and penalty costs are excluded, so that the cost represents the true variable cost of links traversed. This cost can include a time and mileage cost for each car, a fixed cost per car for each classification, and an hourly cost per car for yard track occupancy. Contribution is the difference between Revenues and Costs for each terminated car.

Since the same traffic database is used for every scenario, use of these pairwise correlated observations reduces the error of the difference, as described in Section 5.4. All contribution differences reported in Table 5.3 are statistically significant to higher than 99% confidence. Figure 5.16 addresses the question of whether the model has been run for a sufficiently long duration. Average daily contribution of terminated traffic is plotted as a

Table 5.3 - Contribution* by Scenario

	Revenue	Contribution	Contribution Std Err	Contribution Difference	Difference Std Err
Scenario 1	$16,523,980	$12,265,600	$65,044		
				$319,070	$45,946
Scenario 2	$17,460,490	$12,584,670	$62,768		
				$955,150	$44,194
Scenario 3	$18,885,560	$13,539,820	$55,812		
				$450,710	$27,078
Scenario 4	$18,825,810	$13,990,530	$56,401		

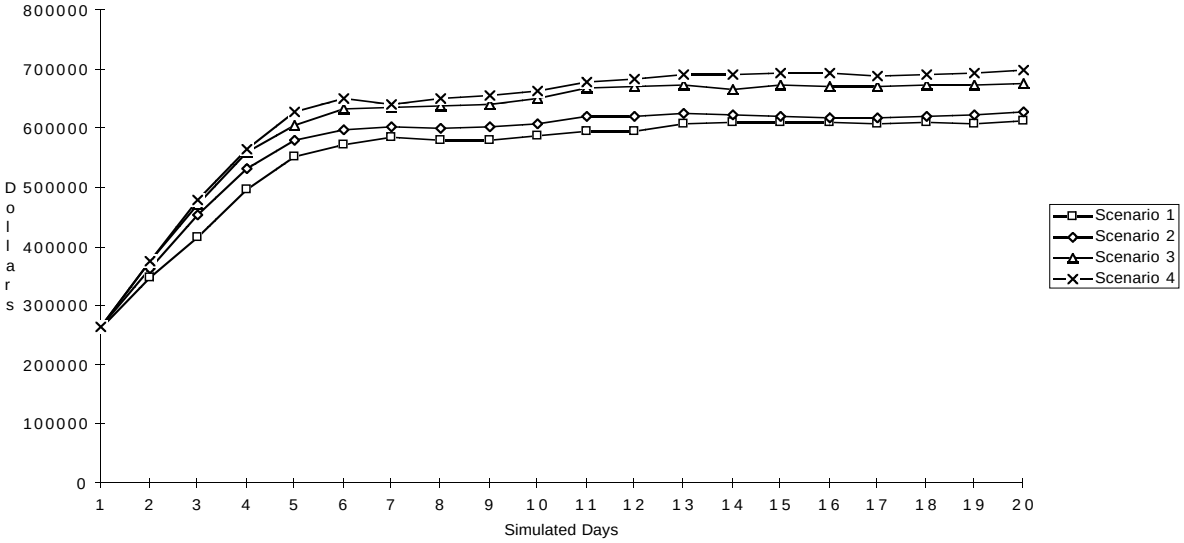
* Based on 14,101 originated shipments in a 500 hour simulation, with normal train capacities

function of the run duration in days. Given: Π_k = daily contribution on day "k" of a "K" day simulation, Figure 5.24 graphs:

$$\text{Average Contribution}_K = \sum_{k=1}^{K} \Pi_k \; / \; K$$

Although Figure 5.16 still exhibits a very slight upward trend in average daily contribution, the average daily contribution value stabilizes as early as seven days into the simulation, so that further increasing the duration of the simulation beyond 500 hours is unlikely to significantly affect the result.

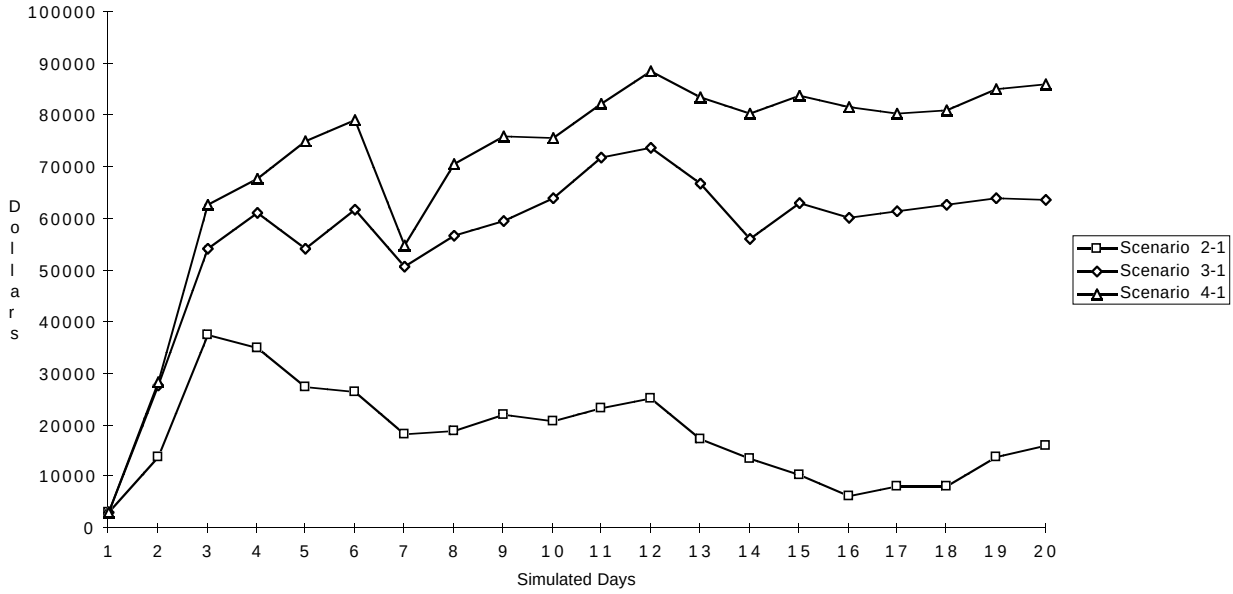
Fig. 5.16 - Average Daily Cont



A more specific measure of statistical convergence focuses on whether the *differences* in contribution have attained steady state values. Figure 5.17 shows that these differences also have attained a steady state: the daily contribution difference associated

with Scenarios 3 and 4 is reasonably constant beyond simulated day 10. Again, it appears that extending the simulation would not significantly affect the result.

Fig 5.17 - Average Daily Contribution D



One might ask how can these results reflect statistical stability when, according to Figure 5.7, capacity has been exceeded and enroute inventory continues to accumulate in Scenarios 1 and 2. The answer lies in the fact that the performance is being measured based on cars *terminated* per unit time. Consider the classical queueing problem where the arrival rate, λ, exceeds the service rate, μ. Queue length, represented by enroute inventory, will continue to increase. However, the *output* of this queueing system is in equilibrium since cars reach their destinations at a constant rate of μ cars/day.

Economic results reported here are primarily driven by the increased effective capacity of the system. From Tables C.1-C.4, the total number of cars reaching their destinations is 11,071 in Scenario 1, increasing to 11,595 and 12,602 cars in Scenarios 2

and 3, respectively. Average revenue and contribution per car, reported in Table 5.4 fluctuates slightly but does not show major variation across scenarios.

In Scenario 4 the number of terminated cars declines slightly to 12,465, but average revenue and contribution per car improves because some unprofitable loads are rejected. Although load rejection eliminates some revenues, the cost is reduced even more so that net contribution is maximized in Scenario 4. It is important to remember that the costs used in this network model do not include fixed cost, or even many items commonly regarded as incremental costs, such as crew expense. From the perspective of the model, crew costs would be a fixed charge associated with the existence of a train which would not change as a function of the number of cars on the train.

Table 5.4 - Revenue Ratios by Scenario

	Terminating Cars	Revenue	Contribution	Rev/Car	Cont/Car
Scenario 1	11,071	$16,523,980	$12,265,600	$1,493	$1,108
Scenario 2	11,595	$17,460,490	$12,584,670	$1,506	$1,085
Scenario 3	12,602	$18,885,560	$13,539,820	$1,497	$1,074
Scenario 4	12,465	$18,825,810	$13,990,530	$1,510	$1,122

Many railways measure their profitability based on the “Operating Ratio”:

Operating Ratio = Costs / Revenues

To put these model results in perspective, the contribution improvement from Scenario 1 to Scenario 4 is $13,990,530 - $12,265,600 or $1,724,930. Using Scenario 1 revenue of $16,523,980, a full implementation of the Dynamic Car Scheduling process with service commitments would produce *a better than 10 point improvement* in a rail carrier's operating ratio.

Of course, the Scenario 1 result might be made to look better through the provision of additional train capacity to increase the number of cars reaching their destinations, which might cost less than the additional revenues gained. However, offsetting this, potential revenue increases and traffic gains for improved service quality are not included in the benefits of scenarios 3 and 4; nor are related cost savings included from operating a capacity balanced, scheduled railroad: such as better crew and asset utilization, and a reduction of unplanned extra and second section train starts.

Figure 5.18 shows the effect of varying train capacity on the contribution of each scenario. Train capacity, rather than traffic, was varied so that the traffic data base would remain identical across all scenarios. This allows performance comparison using "pairwise correlated" observations to reduce statistical sampling errors. Train capacity on every segment is scaled up or down by a uniform percentage across the board; this allows a sensitivity analysis to be performed on the effect of varying train load factors, while still maximizing statistical comparability across scenarios. The experimental results seemed "well behaved" since within each scenario, as capacity increased, contribution also increased. Comparisons across scenarios also turned out as expected, with the current railroad "Base Case" Scenario 1 performing worst, and Scenario 4 performing best at all capacity levels, with other scenarios exhibiting performance intermediate to these two.

Fig 5.18 - Contribution vs Ca

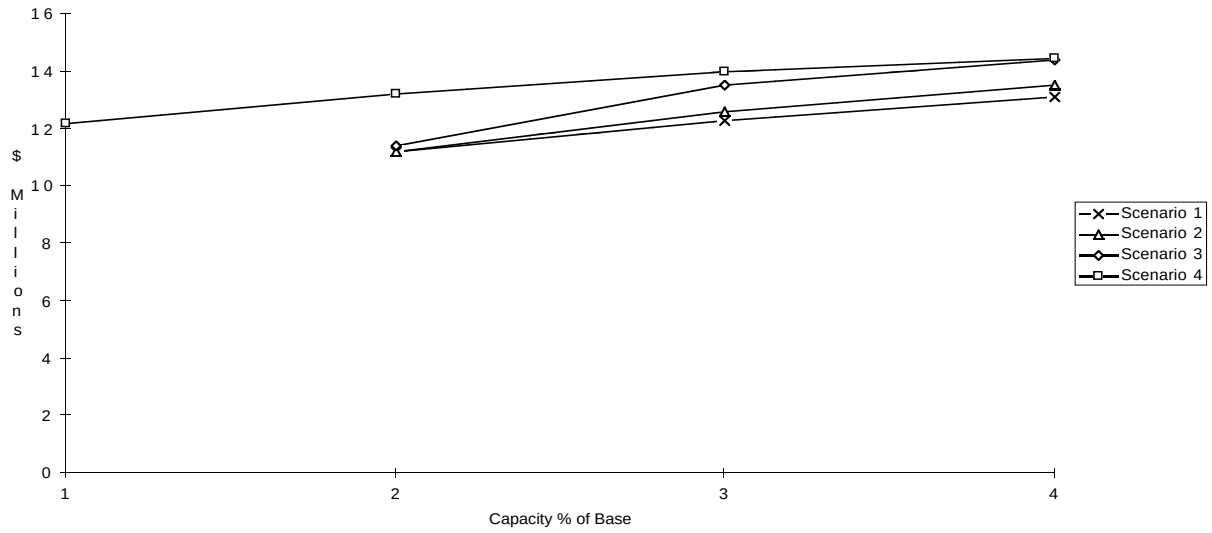


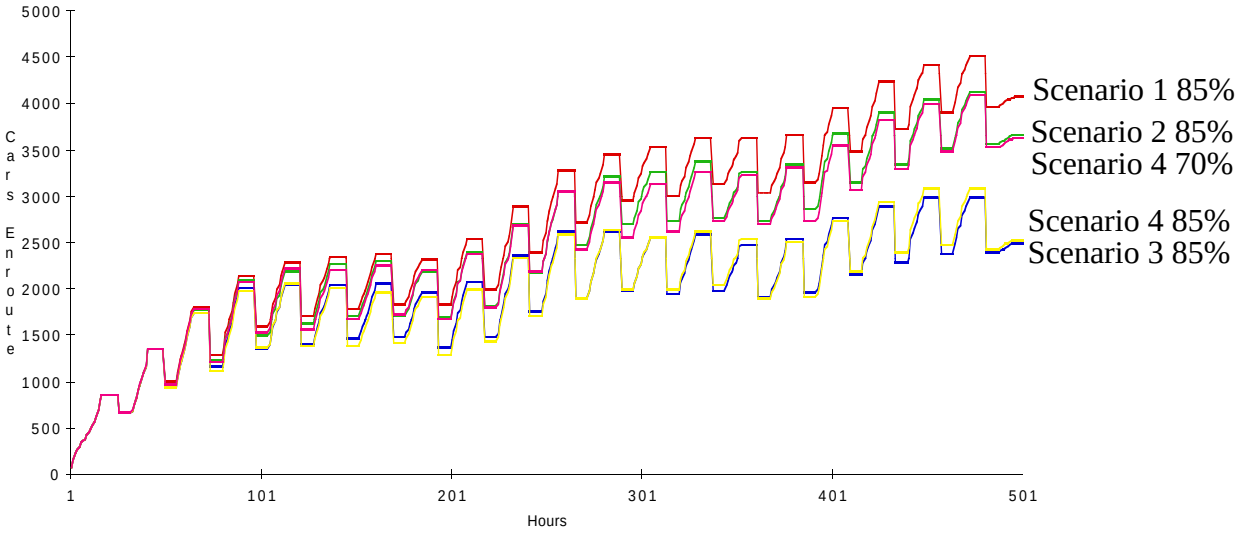
Figure 5.19 shows the enroute inventory for the five reduced capacity scenarios. Scenarios 1 and 2 have already exceeded capacity in the basic 100% capacity case, so no purpose would be served by reducing capacity by more than 15%. In Scenario 3 at the 85% capacity level, the required CPU time in Table 5.5 increased so dramatically that no further

Table 5.5 - Capacity versus CPU Seconds*

% Capacity	Scenario 3 DCS	Scenario 4 DCS	Scenario 4 TSP
115%	2518	2272	40204
100%	7627	3713	47771
85%	65142	5640	54847
70%	N/A	6909	60754

* CPU Seconds do not match exactly those previously reported in Chapter 4 due to the following: (1) Chapter 5 runs were performed on a different machine (2) Minor DCS enhancements were performed during Chapter 5 testing (3) In Chapter 5, TSP was run with Primal Heuristic turned "off."

Fig 5.19 - Enroute Inventory in Reduced



capacity reductions were attempted, although enroute inventory was still stable. In Scenarios 1-3, three simulations each were performed using multiplier factors of 85%, 100% and 115%. Since traffic can be rejected in Scenario 4, a larger capacity reduction could be attempted, using multiplier factors of 70%, 85%, 100% and 115% in Scenario 4.

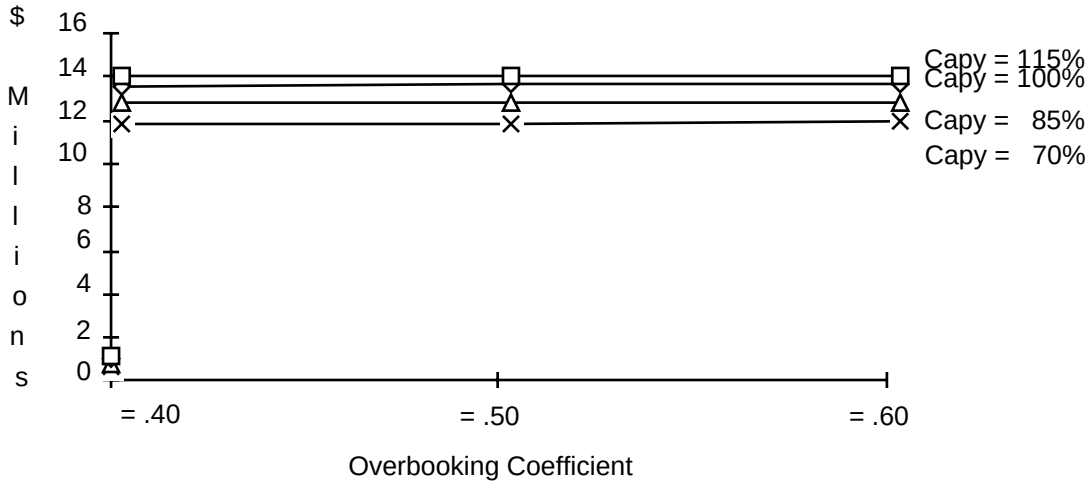
Summarizing these results, Scenario 4 dominates all the others by producing the greatest profit at any train load factor. Utilizing TSP delivery target times and penalty costs seems to improve the computational performance of DCS. The combined TSP-DCS process generates the most profitable shipment routing, improves the efficiency of the real time DCS process and operates more reliably across a wider range of train load factors than does DCS by itself. In Figure 5.19, increasing enroute inventory in the 70% Scenario 4 run seems to suggest that this simulation has not attained a steady state. As traffic congestion continues to build, the TSP algorithm will eventually begin to reject more traffic so that this Scenaro 4 simulation will, once again, eventually attain a steady state.

5.7 Two Special Rolling Horizon Scenarios

Two special run options will be presented in this section. First, the impact of implementing an "α" overbooking coefficient in the Train Segment Pricing model will be explored. Second, a set of simulation results will be presented to develop the possibility of utilizing the Dynamic Car Scheduling algorithm on a stand-alone basis (without TSP) to implement a two tier high /low priority traffic scheme.

Figure 5.20 shows the results of a series of simulations designed to test the impact of using an "α" overbooking coefficient, as proposed in Section 3.6, in the Train Segment Pricing model. The "α" coefficient was varied in the range α=.40, .50, .60 and across the full range of train capacity scaling coefficients .70, .85, 1.00 and 1.15 to see if the impact might be affected by train load factor. As Figure 5.20 shows, varying the value of the "α" coefficient had essentially *no impact* on the result of the rolling horizon simulation with differences in most cases statistically insignificant at the 85% confidence level. A few comparisons had statistically significant, but small differences. These comparisons

Figure 5.20 - Overbooking Coefficient vs Contribution



suggested that the optimal alpha value was in the neighborhood of $\alpha = .50$, which produces essentially the same result as using actual train capacity directly in the Train Segment Pricing model without any adjustment.

Much stronger results were obtained from a test of the Dynamic Car Scheduling algorithm, where penalty costs were added to improve the performance of a priority subset of traffic. These test results show that substantial benefits can be obtained by simply implementing the Dynamic Car Scheduling process by itself. It is not necessary to fully implement a real time reservations process to get some of the benefits of Dynamic Car Scheduling.

The "stand alone" tests (Scenario 3) of DCS in the previous section did not include any penalty costs. In those tests, the DCS program determined shipment priority based solely on hourly cost. Table C.6 shows that nearly all the expensive shipments were delivered on time anyway, but performance is not perfect: 52 cars in the $10-11 range were delivered one day late, as were 61 cars in the $11-12 range.

The question is *whether these late deliveries could be eliminated?* Table C.8 confirms that it is indeed possible to eliminate all the late deliveries of expensive cars, through application of penalty costs to all cars costing more than $10 per hour. This use of penalty costs reinforces the Dynamic Car Scheduling algorithm's natural tendency to favor expensive car types.

A more difficult challenge is to induce the DCS algorithm to prioritize shipments based on some criteria *other* than hourly cost. The logit priority coefficient defined in Section 3.4 was selected as a direct measure of the service sensitivity of the customer. Table C.9 shows that, in the Scenario 3 base case, performance is essentially uncorrelated with the logit priority coefficient.

The objective of the test was to see whether the DCS algorithm could be induced to eliminate late deliveries to the most highly service sensitive customers, regardless of shipment hourly cost. *Table C.10 shows that, after application of penalty costs, all the late deliveries on high priority shipments (having logit coefficient less than 2) were completely eliminated.* This use of penalty costs forces the Dynamic Car Scheduling process to prioritize shipments along a dimension which is completely independent of its normal inclination, which is to prioritize on the basis of hourly cost. The result is significant since it shows that transit time on *any* shipment can be controlled in the Dynamic Car Scheduling process, through application of the appropriate penalty cost.

CHAPTER 6

Summary and Conclusions

6.1 Evaluating the Research Contribution

This research develops a novel method of managing day to day railroad network operations. It develops two new models and solution algorithms for implementing a reservations-based, capacity constrained car scheduling process on freight railroads.

Relatively few applications of Lagrangian Relaxation to multi-commodity network flow problems were found in our literature review. The development of an LR-based dual ascent heuristic for a specialized MCNF problem, using a “tabu search” approach to combat degeneracy, represents a contribution to the mathematical programming literature as well as to the related transportation literature.

While “gain” coefficients are often found in mathematical formulations of electrical and communications network problems, their use is less common in transportation network applications. The primary exception might be certain types of manufacturing logistics problems where gain coefficients could be used to represent transformation of material flows within the model. This formulation of a rail freight yield management problem using gain coefficients to represent customer acceptance probabilities is a novel contribution to the transportation literature.

6.2 Usability of the Research and its Value to Railroads

The research develops practical solution algorithms for very large scale, real world railroad shipment routing problems. A key motivation of the design of these algorithms is to make them readily adaptable to large scale parallel processing architectures. For example, the Train Segment Pricing formulation decomposes the problem into literally thousands of independent subproblems which could each be assigned to their own processor. Parallel processing can also be employed in the Dynamic Car Scheduling algorithm in diversion cost calculations and in the reassignment of diverted flows. This ability to utilize large scale parallel processing ensures the scalability of the mathematical approach, to encompass the full networks of the large mega-carriers resulting from the recent round of United States railway mergers.

Rolling horizon testing of the two proposed algorithms has established not only the technical feasibility of implementing an advanced car scheduling system, but projected that full implementation of this system might improve a rail carriers' operating ratio by more than 10 points. This estimate was developed on the basis of direct operating savings and revenue increase through improved train capacity utilization, not taking any "soft" revenue or market share gains related to improved transit time reliability into account. However, it should be clear that the primary motivation of this research has not been solely to reduce operating cost, but to develop a better management tool which allows railroads to directly control service reliability and match both transit time and reliability to the requirements of the individual customer. The goal is to allow railroads to be more competitive in attracting high revenue, service sensitive traffic without requiring the very high traffic volumes needed to establish a dedicated train service network.

It should also be pointed out that the projected 10 point operating ratio improvement is not due solely to the implementation of this car scheduling procedure by itself. Implicitly assumed here is the ability to run trains on time, and that yards can make connections as scheduled. This is not entirely an unreasonable assumption and would be facilitated by moving towards a preplanned, scheduled train operation and discontinuing certain practices disruptive to terminal operations, for example, annuling a train after it has already been built. The operational framework proposed here allows for a considerable degree of operational flexibility but changes must be implemented on a preplanned basis, preferably with 24-36 hours lead time for manifest train service. Somewhat less lead time would be required in intermodal because terminal operations are more flexible.

The primal heuristic proposed as part of the TSP process can be used to calculate a forward workload projection to form the basis of a flexible operations planning tool. However, to achieve the full 10 percent improvement will also require investment in related planning and control systems including demand forecasting, schedule-adherance based train dispatching, and improved terminal planning and control systems. With investment in comprehensive decision support systems, the projected 10 percent improvement does not seem unreasonable, and even larger gains might be possible.

Rail carriers should consider implementing the DCS and TSP algorithms as an integrated package rather than DCS by itself. Computational testing has shown that DCS both runs faster and tends to produce a higher quality solution when used in conjunction with TSP target times and penalty costs. Initially, TSP delivery times can serve as internal delivery targets only, until such time that a carrier gains sufficient confidence to begin sharing these ETA projections with customers.

Over a longer term, a service commitment process could be established with the objective of being able to give customers firm delivery appointments on most traffic. Scheduling of pickup and delivery appointments is already standard practice in the trucking industry, and this systems improvement would allow railroads to do business in the same manner. Real time service commitment should be integrated into the rail carriers' capacity management system. If a large shipment calls in which was not predicted, or conversely a predicted large shipment fails to materialize this event should be intercepted by the booking program and routed for manual intervention. This manual review should consider the possibility of adding additional train capacity to accommodate the demand without having to raise the TSP segment price too high, and end up rejecting other shipments later.

6.3 Future Research Possibilities

A primary objective of this research has been to develop a dynamic car scheduling capability beyond the conceptual stage into an operational prototype system. However, the concept still requires more incubation before it is ready for full scale implementation. There are opportunities for further technical improvements to the car scheduling process, also, more research is needed to address systems integration issues and to develop the true costs and benefits of implementation. Projected economics, however, will likely be specific to each individual rail carrier, and will depend on that particular carriers' past and current operating practices as well as its regulatory and marketplace environment.

A priority area for research should focus on a more accurate representation of the effect of dynamic capacity management on system performance. A fixed capacity network was assumed here, primarily because the rules for varying train capacity are likely to be very complex, are difficult to concisely code into a computer program and may also vary across the four car scheduling scenarios. Fixed table based blocking cannot adjust car

routing patterns in response to stochastic demand fluctuations, so this system requires frequent manual intervention to better adjust train operations to demand, or else a lot of excess capacity must be built into the system.

These interventions exact a price in terms of increased operating cost and also degrade service reliability — particularly if intervention is performed on a reactive rather than proactive basis. While the economics of the current operation might be more accurately assessed by a more detailed simulation of flexible operating policies, the Dynamic Car Scheduling scenarios would also benefit from the ability to make flexible capacity adjustments — although specific procedures for triggering adjustments to the operating plan would likely be much different than today's.

One possible research direction could explore demand forecasting, and demand-responsive operation of flexible capacity networks utilizing advanced car scheduling approaches. It would explore the impact of implementing alternative strategies for operation of extra trains, second sections, bypass trains, annulments, or consolidations, and their impact on utilization of locomotives, crews and track capacity slots in real time. This research might include the development of a real time "gaming" simulation, which could be used to explore human factors issues and also as a training environment for operations center personnel.

A second research direction would be to improve the optimization algorithms themselves. This might include simultaneous optimization of empty and loaded movements on a space/time network, and the adoption of a true stochastic network model formulation. This would be a highly mathematical and computationally intensive line of research.

A third direction would be the development of improved management systems for intermodal and rail terminal operations. The only known published terminal operations

model compatible with the kind of dynamic car routing proposed here is the work by Kraft and Spielberg [1993]. This model was further developed on a proprietary basis by Kraft for the Union Pacific Railroad [1995].

A fourth and final direction for research would be to revisit the literature on Service Design techniques. Nearly all of this literature assumes a fixed, tag-table based shipment routing strategy. However, under a dynamic routing scenario, block and train volumes might behave quite differently than these fixed table-based blocking models assume. A new set of tools may need to be developed to optimize the Service Design function in a dynamic car scheduling environment.

Bibliography

Aashtiani, H. Z. and T. L. Magnanti, “Equilibria on a Congested Transportation Network,” SIAM Journal on Algebraic and Discrete Methods Volume 2, No 3, pp 213-226, 1981.

Ali, A.I., J.L. Kennington and B. Shetty, The Equal Flow Problem, European Journal of Operational Research, Vol. 36, pp.107-115, 1988.

Allen, M. M. and W. J. Rennicke, “Terminal Sequencing System,” Transportation Research Forum Proceedings, Volume 18, No 1, pp 414-420, 1977

Anderson, S. P., A. DePalma and J.F. Thisse, “A Representative Consumer Theory of the Logit Model,” International Economic Review, Volume 29, No 3, pp 461-466, August 1988.

Assad, A. A., “Multicommodity Network Flows - A Survey,” Networks, Vol. 8, pp. 37-91, 1978.

Assad, A. A. , “Models for Rail Transportation,” Transportation Research 14A, pp. 205-220, 1980a.

Assad, A. A., “Modelling of Rail Networks: Toward a Routing/ Makeup Model,” Transportation Research 14B, pp. 101-114, 1980b.

Borsch-Supan, A., “On the Compatibility of Nested Logit Models with Utility Maximization,” Journal of Econometrics, Volume 43, No 3, pp 373-388, March 1990.

Bailey, A. , The Status of Car Scheduling: A Railroad Industry Review, presented at INFORMS Rail Special Interest Group Spring Roundtable, Los Angeles, CA, April 1, 1995.

Barnhart, C. and Y. Sheffi, “A Network-Based Primal-Dual Heuristic for the Solution of Multicommodity Network Flow Problems,” Transportation Science, Vol 27, No. 2, May 1993, pp 102-117.

Barr, R. S. , F. Glover and D. Klingman, “An Improved Version of the Out-of-Kilter Method and a Comparative Study of Computer Codes,” Mathematical Programming, Vol. 7, pp. 60-86, 1974.

Baugher, R., Norfolk Southern Integrated Transportation Management System: “Seamless” Service Management, presented at Transportation Research Forum New York City meeting, 1993.

Belobaba, P. P., “Airline Yield Management: An Overview of Seat Inventory Control,” Transportation Science 21, pp 63-73, 1987.

Belobaba, P. P., “Application of a Probabilistic Decision Model to Airline Seat Inventory Control,” Operations Research 37(2) pp. 183-197, 1989.

Bertsekas, D.P. and P. Tseng. "Relaxation methods for minimum cost ordinary and generalized network flow problems," Operations Research, Vol. 36, No. 1, pp 93-114, 1988.

Bodin, L. D., B. L. Golden , A. D. Schuster and W. Romig, "A Model for the Blocking of Trains," Transportation Research 14B, pp. 115-120, 1980.

Bradley, S. P. , A. C. Hax and T. L. Magnanti, Applied Mathematical Programming, Addison-Wesley Publishing Company, pp 474-477, 1977.

Braklow, J. W., W. W. Graham, S. M.. Hassler, K. E. Peck, W. B. Powell, "Interactive Optimization Improves Service and Performance for Yellow Freight System," Interfaces, v22n1, p. 147-172, Jan/Feb 1992.

Brannlund, U. , P. O. Lindberg, J. E. Nilsson, and A. Nou, "Allocation of Scarce Track Capacity Using Lagrangian Relaxation," Working Paper, January 1994.

Brown, G. G. and R. D. McBride, "Solving Generalized Networks," Management Science, Vol 30, No 12, pp 1497-1523, December 1984.

Bryson, N. , "Parametric Programming and Lagrangian Relaxation: The Case of the Network Problem with a Single Side Constraint," Computers and Operations Research, Volume 18, No. 2, pp. 129-140, 1991.

Campbell, K. C. and E. K. Morlok, "Rail Freight Service Flexibility and Yield Management," Transportation Research Forum Proceedings, 36'th Annual Meeting, Volume 2, pp 529-548, 1994

Campbell, K. C., Booking and Revenue Management for Rail Intermodal Services, Ph. D. Dissertation, Department of Systems Engineering, University of Pennsylvania, 1996.

Chajakis, E. D., Scheduling and Lagrangean Approximations, Ph. D. Dissertation, the Department of Operations and Information Management, University of Pennsylvania, 1993.

Charnes, A. and W. W. Cooper, "Deterministic Equivalents for Optimizing and Satisficing Under Chance Constraints," Operations Research, Vol. 11, pp 18-39, 1963.

Chen, B. and P. T. Harker, "Two Moments Estimation of the Delay on Single-Track Rail Lines with Scheduled Traffic," Transportation Science, Vol 24, No. 4, pp. 261-275, November 1990.

Chen, S. and R. Saigal, "A Primal Algorithm for Solving a Capacitated Network Flow Problem with Additional Linear Constraints," Networks 7, pp 59-79, 1977.

Chen, C. J. and M. Engquist, "A Primal Simplex Approach to Pure Processing Networks," Management Science, Vol 32, No 12, December 1986.

Cooper, L. and L. J. LeBlanc, "Stochastic Transportation Problems and Other Network Related Convex Problems," Naval Research Logistics Quarterly, Vol 24, pp 327-337, 1977.

Crainic, T. G., J. A. Ferland and J. Rousseau, "A Tactical Planning Model for Rail Freight Transportation," Transportation Science, Vol 18, #2, pp 165-184, May 1984.

Crainic, T. G. and Michael Gendreau, "Approximate Formulas for the Computation of Connection Delays Under Capacity Restrictions in Rail Freight Transportation," Center for Transportation Research, University of Montreal, Publication #438, November 1985.

Crainic, T. G. and J. Rousseau, “Multicommodity, Multimode Freight Transportation: A General Modeling and Algorithmic Framework for the Service Network Design Problem,” Transportation Research B, Vol 20B, No. 3, pp 245-242, 1986.

Crainic, T. G., M. Florian and J. Leal, “A Model for the Strategic Planning of National Freight Transportation by Rail,” University of Montreal, Center for Transportation Research, Publication #518, June 1987.

CSX Transportation Presentation, ORSA/TIMS Rail Special Interest Group meeting, session on Rail Terminal Modeling, April 24-27, Boston MA, 1994.

Dantzig, G. B. , A. Orden and P. Wolfe, “The Generalized Simplex Method for Minimizing a Linear Form under Linear Inequality Restraints,” Pacific Journal of Mathematics 5, 183-195, 1955.

Dejax, P. and T. G. Crainic, “A Review of Empty Flows and Fleet Management Models in Freight Transportation,” Transportation Science, Vol. 21, No. 4, pp 227-247, Nov. 1987.

Dejax, P. and J. H. Bookbinder, “Goods Transportation by the French National Railway (SNCF): the Measurement and Marketing of Reliability,” Transportation Research, Vol 25A, #4, pp 219-225, 1991.

Dial, R., F. Glover, D. Karney and D. Klingman, “A Computational Analysis of Alternative Algorithms and Labeling Techniques for Finding Shortest Path Trees,” Networks, Vol. 9, pp 251-248, 1979.

Dijkstra, E. W., “A note on two problems in connexion with graphs,” Numer. Math., Vol 1., pp. 269-271, 1959.

Dingle, A. D., An Evaluation of CACTUS, A Dynamic Track Assignment Program Pilot Installation at Southern Pacific's West Colton Yard, prepared for the AAR Freight Car Utilization Research-Demonstration program, February 15, 1984.

Divoky, J. J. and M. S. Hung, "Performance of Shortest Path Algorithms in Network Flow Problems," Management Science, Vol. 36, No. 6, June 1990.

Dong, Y. ,"Using the Task Assignment Technique in Rail Yard Operations," AAR Affiliated Laboratories at M.I.T., Presentation to Railroad Industry Representatives, April 22, 1994.

Duffy, M. ,"Statistical Process Control Applied to Rail Freight Terminal Performance: A Case Study of CSX's Radnor Yard," AAR Affiliated Laboratories at M.I.T., Presentation to Railroad Industry Representatives, April 22, 1994.

Elam, J. , F. Glover, and D. Klingman, "A Strongly Convergent Primal Simplex Algorithm for Generalized Networks," Mathematics of Operations Research, Vol. 4, No. 1, pp 39-59, February 1979.

Elkins, S. ,"The Basics of Yield Management by IDeas," Integrated Decisions and Systems, Inc., 3500 Yankee Drive, Suite 350, Eagan, MN 55121, September 1991.

Engelberg, G. P. and C. R. Yagar, "New Approaches to Yard Planning Using Computer Simulation," Proceedings of the Transportation Research Forum, Vol 23, No. 1, pp. 185-194, 1982.

Etcheberry, J. , "The Set Covering Problem: A New Implicit Enumeration Algorithm," Operations Research, Vol 25., No 5, pp 760-772, Sept-Oct 1977.

Farvolden, J.M. , W.B. Powell and I. J. Lustig, “A Primal Partitioning Solution for the Arc-Chain Formulation of a Multicommodity Network Flow Problem,” Operations Research, Vol 41., No. 4, pp 669-693, July-August 1993.

Fisher, M.L. and D.S. Hochbaum, “Database Location in Computer Networks,” Journal of the Association for Computing Machinery, Vol 27, No. 4, pp. 718-735, October 1980.

Fisher, M.L., “The Lagrangian Relaxation Method for Solving Integer Programming Problems,” Management Science, Vol 27, No 1, pp. 1-18, January 1981.

Fisher, M.L., “An Applications Oriented Guide to Lagrangian Relaxation,” Interfaces 15:2, pp. 10-21, March-April 1985.

Fisher, M. L. , R. Jaikumar and L. N. Van Wassenhove, “A Multiplier Adjustment Method for the Generalized Assignment Problem,” Management Science, Vol 32, No. 9, pp 1095-1103, Sept 1986.

Florian, M., S. Nguyen and S. Pallottino, “A Dual Simplex Algorithm for Finding all Shortest Paths,” Networks, Vol 11, pp 367-378, 1981.

Frantzeskakis, L. F. and W. B. Powell, “A Successive Linear Approximation Procedure for Stochastic, Dynamic Vehicle Allocation Problems,” Transportation Science, Vol 24, No. 1, pp 40-57, February 1990.

Frederickson, G. N., “Fast Algorithms for Shortest Paths in Planar Graphs, With Applications,” SIAM Journal of Computing, Vol 16, No. 6, pp 1004-1022, 1987

Fukushima, M., “On the Dual Approach to the Traffic Assignment Problem,” Transportation Research-B, Vol 18B, No. 3, pp. 235-245, 1984.

Gallego, G. and G. van Ryzin, “Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons,” Management Science, Vol. 40, No. 8, pp 999-1020, 1994a.

Gallego, G. and G. van Ryzin, “A Multi-Product Dynamic Pricing Problem and Its Applications to Network Yield Management,” Department of Industrial Engineering and Operations Research, Working Paper, Columbia University, 1994b.

Geoffrion, A. M., “Lagrangian Relaxation and its Uses in Integer Programming,” Math. Programming Study, Vol 2, pp. 82-114, 1974.

Glickman, T. S. and H. D. Sherali, “Large Scale Network Distribution of Pooled Empty Freight Cars over Time, with Limited Substitution and Equitable Benefits,” Transportation Research, Vol 19B, #2, pp 85-94, 1985.

Glover, F. and D. Klingman, “ The Simplex SON Algorithm for LP/Embedded Network Problems,” Mathematical Programming Study 15, pp. 148-176, 1981.

Gohring, K. W. , “Application of Network Flow Theory to the Distribution of Locomotives and Cabooses,” Operations Research Section, Southern Railway, presented at the Institute of Management Sciences, Southeastern Chapter Winter Symposium, March 3-4, 1971.

Gorman, M. F., Presentation at the Transportation Research Forum Daytona Beach meeting, 1994.

Gorman, M. F., An Application of Genetic and Tabu Searches to the Freight Railroad Operating Plan Problem, ATSF Railway, presented at the Los Angeles INFORMS session, April 1995.

Grigoriadis, M. D. and W. W. White, “A Partitioning Algorithm for the Multicommodity Network Flow Problem,” Mathematical Programming, Vol. 3, pp.157-177l, 1972.

Guignard, M. and M. B. Rosenwein, “An Improved Dual Based Algorithm for the Generalized Assignment Problem,” Operations Research, Vol 37, No. 4, pp 658-663, July-August 1989a.

Guignard M. and M. B. Rosenwein, “An application-oriented guide for designing Lagrangean dual ascent algorithms,” European Journal of Operations Research 43, pp 197-205, 1989b.

Haghani, A. E., “Formulation and Solution of a Combined Train Routing and Makeup, and Empty Car Distribution Model,” Transportation Research 23B, pp. 433-452, 1989.

Hajek, B. and R. G. Ogier, “Optimal Dynamic Routing in Communication Networks with Continuous Traffic,” Networks, Vol 14, pp 457-487, 1984.

Halder, A.K., “The Method of Competing Links”, Transportation Science, Vol. 4, pp 36-51, 1970.

Hallowell, S. F. , Optimal Dispatching Under Uncertainty: With Application to Railroad Scheduling, Ph. D. Dissertation, the Department of Operations and Information Management, University of Pennsylvania, 1993.

Harker, P. T., “Services and Technology: Reengineering the Railroads,” Interfaces 25:3, pp. 72-80, May-June 1995.

Harker, P. T. and S. Hong, “Two moments estimation of the delay on a partially double-track rail line with scheduled traffic,” Journal of the Transportation Research Forum Vol 31, No. 1, pp. 38-49, 1990.

Harker, P. T. and S. Hong, Pricing of Track Time in Railroad Operations: An Internal Market Approach, University of Pennsylvania, School of Engineering and Applied Science, Department of Systems, Working Paper, July 1993.

Hartman, J. K. and L. S. Lasdon, "A Generalized Upper Bounding Algorithm for Multicommodity Network Flow Problems," Networks 1, pp. 333-354, 1972.

Held, M. and R. M. Karp, "The Traveling Salesman Problem and Minimum Spanning Trees," Operations Research, Vol. 18, pp 1138-1162, 1970.

Held, M. and R. M. Karp, "The Traveling Salesman Problem and Minimum Spanning Trees: Part II," Mathematical Programming, Vol. 1, pp. 6-25, 1971.

Held, M. H., P. Wolfe and H. D. Crowder, "Validation of subgradient optimization," Mathematical Programming, Vol. 6, No. 1, pp. 62-88, 1974.

Higgins, A. , E. Kozan and L. Ferreira, "Optimal Scheduling of Trains on a Single Line Track," Faculty of Science working paper, Queensland University of Technology, May 1995.

Ho, Y. C., ed., Discrete Event Dynamic Systems: Analyzing Complexity and Performance in the Modern World, IEEE Press, 1992.

Holmberg, K, Joborn, M. and J. T. Lundgren, "A Model for Distribution of Empty Freight Cars," Report LiTH-MAT-R-1996-11, Linkoping Institute of Technology, Department of Mathematics, S-581 83 Linkoping, Sweden, May 1996.

Hong, S. and P. T. Harker, "Tactical Scheduling of Freight Railroad Operations," Fishman-Davidson Center, The Wharton School, University of Pennsylvania, October 1990.

Green, P. E. and A. M. Krieger, "Choice Rules and Sensitivity Analysis in Conjoint Simulators," Journal of the Academy of Marketing Science, Volume 16, No 1, pp 114-127, Spring 1988.

Hu, T. C., "Multi-Commodity Network Flows," Operations Research, Vol. 11, pp. 344-360, 1963.

Huntley, C. L., D. E. Brown, D. E. Sappington, and B. P. Markowicz, "Freight Routing and Scheduling at CSX Transportation," Working Paper, The Institute for Parallel Computation and the Department of Systems Engineering, University of Virginia, Charlottesville, VA, December 1993.

Jones, K.L., I.J. Lustig, J.M. Farvolden and W.B. Powell, "Multicommodity network flows: The Impact of formulation on decomposition," Mathematical Programming, Vol. 62, pp 95-117, 1993.

Jordan, William C and Mark A Turnquist, "A Stochastic, Dynamic Network Model for Railroad Car Distribution," Transportation Science, Vol 17, #2, pp 123-145, May 1983.

Jovanovic, D. and Patrick T. Harker, "A Decision Support System for Train Dispatching: An Optimization-Based Methodology," Journal of the Transportation Research Forum, Vol 30, #1, 1990.

Keaton, M. H., "Designing Optimal Railroad Operating Plans: Lagrangian Relaxation and Heuristic Approaches," Transportation Research 23B, pp. 415-431, 1989.

Keaton, M. H.,"Service-Cost Tradeoffs for Carload Freight Traffic in the U.S. Rail Industry," Transportation Research 25A, #6, pg 363, November 1991.

Keaton, M. H.,"The Impact of Train Timetables on Average Car Time in Rail Classification Yards," Journal of the Transportation Research Forum, Volume 32, #2, 1992.

Keaton, M. H., "Economies of Density and the Structure of the Less-than-Truckload Motor Carrier Industry Since Deregulation," Proceedings, Transportation Research Forum Daytona Beach meeting, Volume 1, pp 59-74, 1994.

Kedia, P. K., Multiplier Adjustment Method for Solving Certain Combinatorial Optimization Problems, Ph. D. Dissertation, the Department of Operations and Information Management, University of Pennsylvania, 1985.

Kennington, J.L. and M. Shalaby. An effective subgradient procedure for minimal cost multicommodity flow problems. Management Science, Vol. 23, pp 994-1004, 1977.

Kennington, J. L., "A Survey of Linear Cost Multicommodity Network Flows," Operations Research, Vol 26, No. 2, pp. 209-236, March-April 1978.

Klein, R. S., H. Luss and D. R. Smith, "A Lexicographic Minimax Algorithm for Multiperiod Resource Allocation," Mathematical Programming, Vol. 55, No. 2, Series A, pp 213-234, 1992.

Koning, R. H and G. Ridder, "On the compabibility of nested logit models with Utility Maximization," Journal of Econometrics, Volume 63, No 2, pp 389-396, August 1994.

Kornhauser, Alain L and Peter Mayewski,"A Heuristic Approach to the Generation of Network Oriented Blocking Plans for Railroad Operations," Proceedings of the 24th Annual Meeting, Transportation Research Forum, #2, pg 162, 1983.

Kraay, D. R. and P. T. Harker, Real-Time Scheduling of Freight Railroads, University of Pennsylvania, School of Engineering and Applied Science, Department of Systems, Working Paper 94-03, October 1994, Transportation Research B, forthcoming.

Kraft, D. J. , T. H. Oum, and M. W. Tretheway, “Airline Seat Management,” Proceedings of the Transportation Research Forum, Vol 27, No. 1, pp. 340-348, 1986.

Kraft, E. R., “Jam Capacity of Single Track Rail Lines,” Proceedings of the Transportation Research Forum, Vol 23, No. 1, pp. 461-471, 1982.

Kraft, E. R. , “A Branch and Bound Procedure for Optimal Train Dispatching,” Journal of the Transportation Research Forum, Vol 28, No. 1, pp. 263-276, 1987.

Kraft, E. R.,”Analytical Models for Rail Line Capacity Analysis,” Journal of the Transportation Research Forum, Vol 29, No. 1, pp. 153-162, 1988.

Kraft, E. R. and M. Guignard-Spielberg, “A Mixed Integer Optimization Model to Improve Freight Car Classification in Railroad Yards,” Report 93-06-06, Department of Operations and Information Management, The Wharton School, University of Pennsylvania, 1993.

Kraft, E. R. “The Link Between Demand Variability and Railroad Service Reliability,” Journal of the Transportation Research Forum, Vol. 34, No. 2, pp. 27-43, 1995.

Kwon, O. K. and C. D. Martland, “Simulating Rail Network Reliability: The Effects of Traffic Variability, Line Reliability, and Controls on Dispatching Trains from Terminals,” M.I.T. Working Paper, September, 1992.

Kwon, O.K. , Managing Heterogeneous Traffic on Rail Freight Network Incorporating the Logistics Needs of Market Segments, Ph. D. Dissertation, Dept of Civil and Environmental Engineering, Massachussetts Institute of Technology, 1994.

Lee, T. C. and M. Hersh, "A Model for Dynamic Airline Seat Inventory Control with Multiple Seat Bookings," Transportation Science 27, pp 252-265, 1993.

Little, P., J. M. Sussman and C. D. Martland, "Alternative Freight Car Maintenance Policies with Attractive Reliability/ Cost Relationships," Journal of the Transportation Research Forum, Vol 31, #1, 1991.

Luss, H. and D. R. Smith, "Resource Allocation among Competing Activities: A Lexicographic Minimax Approach," Operations Research Letters, Vol. 5, No. 5, pp 227-231, November 1986.

Luss, H. , "An Algorithm for Separable Non-Linear Minimax Problems," Operations Research Letters, Vol. 6, No. 4, pp 159-162, September 1987.

Manrai, A. K., "Mathematical models of Brand Choice Behavior," European Journal of Operational Research, Volume 82, No 1, pp 1-17, April 1995.

Markowicz, B. P. and M. A. Turnquist, Applying the LP Solution to the Daily Distribution of Freight Cars, presented at the TIMS/ORSA National Meeting, Las Vegas, NV, April 1990.

Marsten, R., R. Subramanian, I. Lustig and D. Shanno, "Interior Point Methods for Linear Programming: Just Call Newton, Lagrange, and Fiacco and McCormick!", Interfaces, Vol. 20, No. 4, pp. 105-116, 1990.

Martland, C. D. and M. E. Smith, “Estimating the Impact of Advanced Dispatching Systems on Terminal Performance,” Journal of the Transportation Research Forum, Vol 30, #2, pg. 286, 1990.

Martland, C. D., P. Little and J. M. Sussman, “Service Management in the Railroad Industry,” in Railroad Freight Transportation Research Needs, Conference Proceedings 2, of the joint AAR/FRA/TRB conference in Bethesda, Md, July 12-14, 1993, National Academy Press, Washington, D.C. 1994, pp. 89-103.

McBride, R. D. ,”Solving Embedded Generalized Network Problems,” European Journal of Operational Research, Vol. 21, pp 82-92, 1985.

McBride, R. D. and J. W. Mamer, “Solving Multicommodity Flow Problems with a Primal Embedded Network Simplex Algorithm,” presented at ORSA/TIMS conference, Phoenix AZ, November 1993.

McCarren, J. Reilly and Carl D. Martland, The MIT Service Planning Model, Studies in Railroad Operations and Economics, Vol 31, MIT, 1980.

Mendiratta, V. B. and M. A. Turnquist, “Model for Management of Empty Cars,” Transportation Research Record 838, pp 50-55, 1981.

Mercer Management Consulting, “The Service Quality Challenge for the 1990s,” The 5th American Railroad Conference, November 1991.

Miliotis, P. ,”Integer Programming Approaches to the Traveling Salesman Problem,” Mathematical Programming, Vol. 10, pp 367-378, 1976.

Missouri Pacific Railroad, Missouri Pacific's Computerized Freight Car Scheduling System: Functional Requirements, Report number FRA/OPPD-77/10 Final Rpt, NTIS Order Number PB-2754239/8ST DOTL NITS, 1977.

Morlok, E. K. and R. B. Peterson, Railroad Freight Train Scheduling: A Mathematical Programming Formulation, Northwestern University, 1970a.

Morlok, E. K. and R. B. Peterson, "A Final Report on a Development of a Geographic Transportaton Network Generation and Evaluation Model," Proceedings, Eleventh Annual Meeting of the Transportation Research Forum, Volume 11, pp. 71-105, 1970b.

Moss, F. H. and A. Segall, An Optimal control approach to dynamic routing in data communication networks, Part I: Principles; Part II, Geometrical interpretation, EE Publs. 312 and 319, Technion- Israel Institute of Technology (1977,1978).

Nagamochi, H. and T. Ibaraki, "On max-flow min-cut and integral flow properties for multicommodity flows in directed networks," Information Processing Letters, Vol. 31, pp. 279-285, 1989.

Nagamochi, H. and T. Ibaraki, "Multicommodity flows in certain planar directed networks," Discrete Applied Math., Vol. 27, pp.125-145, 1990.

Norfolk Southern Railway discussion, Trip Planning and Seamless Transportation Session, Proceedings of the Transportation Research Forum Meetings, St. Louis, pg 262, 1992.

Nozick, L. K., A Model of Intermodal Rail-Truck Service for Operations Management, Investment Planning and Costing, Ph. D. Dissertation, the Department of Systems, University of Pennsylvania, 1992.

Organisation for Economic Co-operation and Development (OECD), Advanced Logistics and Road Freight Transport, 1992.

Papastavrou, J. D., S. Rajagopalan and A. J. Kleywegt, “The Dynamic and Stochastic Knapsack Problem with Deadlines,” Management Science, Vol. 42, No. 12, pp 1706-1718, December 1996.

Pape, U., “Implementation and Efficiency of Moore-Algorithms for the Shortest Route Problem,” Mathematical Programming 7, pp 212-222, 1974.

Peat, Marwick, Mitchell, Inc., “Parametric Analysis of Railway Line Capacity,” prepared for the Federal Railroad Administration, DOT-FR-4-5014-2, 1975.

Petersen, E. R. ,”Over-the-Road Transit Time for a Single Track Railway,” Transportation Science 8, pp 65-74, 1974.

Petersen, E. R. ,”Railyard Modeling; Part I. Prediction of Put-Through Time,” Transportation Science 11, pp 37-49 (1977a).

Petersen, E. R. , “Railyard Modeling; Part II. The effect of Yard Facilities on Congestion,” Transportation Science 11, 50-59 (1977b).

Petersen, E. R. and A. J. Taylor, “A Structured Model for Rail Line Simulation and Optimization,” Transportation Science, Vol 16, No. 2, May 1982.

Petersen, E. R., and A. J. Taylor, “Line Block Prevention in Rail Line Dispatch and Simulation Models,” INFOR journal Vol 21, no. 1, pp 46-51, February 1983.

Philip, C.E. and J. M. Sussman,”Inventory Model of the Railroad Empty Car Distribution Process,,” Transportation Research Record 656 pp. 52-60, 1977.

Pinar, M.C. and S.A. Zenios, “Solving Nonlinear Programs with Embedded Network Structures,” in Network Optimization Problems, edited by D.Z. Du and P. M. Pardalos, World Scientific Publishing Company, pp 177-202, 1993.

Powell, W. B., “A Stochastic Model of the Dynamic Vehicle Allocation Problem,” Transportation Science, Vol. 20, No. 2, p 117-129, 1986a.

Powell, W. B., “Iterative Algorithms for Bulk Arrival, Bulk Service Queues with Poisson and Non Poisson Arrivals,” Transportation Science, Vol 20, #2, pg 69, 1986b.

Powell, W. B., “A Local Improvement Heuristic for the Design of Less-than-Truckload Motor Carrier Networks,” Transportation Science, Vol. 20, pp 246-257, 1986c.

Powell, W. B., “An Operational Planning Model for the Dynamic Vehicle Allocation Problem with Uncertain Demands,” Transportation Research B, Vol 21, No. 3, pp. 217-232, 1987.

Powell, W. B., “A Comparative Review of Alternative Algorithms for the Dynamic Vehicle Allocation Problem,” in Vehicle Routing: Methods and Studies, B. I. Golden and A. A. Assad (eds) , North-Holland, Amsterdam, 1988.

Powell, W. B. , Y. Sheffi, K. S. Nickerson, K. Butterbaugh, and S. Atherton, “Maximizing Profits for North American Van Lines’ Truckload Division: A New Framework for Pricing and Operations,” Interfaces 18:1 pp 21-41, January-February 1988.

Powell, W. B. and Y. Sheffi, “Design and Implementation of an Interactive Optimization System for Network Design in the Motor Carrier Industry,” Operations Research, Vol. 37, No. 1, pp 12-27, Jan.-Feb. 1989.

Powell, W.B. “A Review of Sensitivity Results for Linear Networks and a New Approximation to Reduce the Effects of Degeneracy,” Transportation Science, Vol. 23, No. 4, pp 231-243, November 1989.

Powell, W. B. , “Optimization Methods and Algorithms: An Emerging Technology for the Motor Carrier Industry,” IEEE Transactions on Vehicular Technology, Vol. 40, No. 1, February 1991.

Powell, W. B. and R. K. Cheung, “An Algorithm for Multistage Dynamic Networks with Random Arc Capacities, with an Application to Dynamic Fleet Management,” Princeton University, School and Engineering and Applied Science, Department of Civil Engineering and Operations Research, Technical Report SOR-92-11, August 1992.

Powell, W. B. , P. Jaillet, and A. Odoni, Stochastic and Dynamic Networks and Routing, to appear in the Handbook in Operations Research and Management Sciences, Princeton University, School and Engineering and Applied Science, Department of Civil Engineering and Operations Research, Technical Report SOR-93-19, July 1993.

Powell, W. B., “Making the Solution Fit the Problem,” in the INFORMS Rail Special Interest Group newsletter, Vol. 2, No. 1, pp 6-9, Spring 1995.

Quanshou, Z. and X. Yuanjin, “A Marshalling Yard Decision Support System,” Rail International, May 1992.

Railway Age, “What’s Needed to Keep Intermodal Growing,” pg 62, October 1993.

Roberts, Paul O., “Factors Influencing the Demand for Goods Movement,” Working Paper No. 1, Center for Transportation Studies, MIT, Cambridge, MA, 1975.

Rosenwein, M. B., Design and Application of Solution Methodologies to Optimize Problems in Transportation Logistics, Ph. D. Dissertation, the Department of Operations and Information Management, University of Pennsylvania, 1986.

Rosenwein, M. B., “An Improved Bounding Procedure for the Constrained Assignment Problem,” Computers and Operations Research, Vol 18, No. 6, pp 531-535, 1991.

Ryu, C., Capacity-Oriented Planning and Scheduling in Production and Distribution Systems, Ph. D. Dissertation, the Department of Operations and Information Management, University of Pennsylvania, 1993.

Santa Fe Railway, Personal Visit, April 18, 1994.

Sauder, R. L. and W. M. Westerman, “Computer aided train dispatching: decision support through optimization,” Interfaces, Vol 13, pp 24-37, 1983.

Schlenker, M. A., “Service Reliability Research Project,” AAR Affiliated Laboratories at M.I.T., Presentation to Railroad Industry Representatives, April 22, 1994.

Schrage, L. ,”Implicit Representation of Variable Upper Bounds in Linear Programming,” Mathematical Programming Study 4, pp 118-132, 1975.

Schultz, G.L. and R.R. Meyer, “An Interior Point Method for Block Angular Optimization,” SIAM Journal on Optimization, Vol. 1, 1991.

Shaeffer, W. W. and George L. Stern, “The Influence of Locomotive Reliability upon Locomotive Acquisition and Maintenance Policies,” Proceedings of the 27th Annual Meeting, Transportation Research Forum, #1, pp 113, 114, 1986.

Shapiro, J. F., “A Note on the Primal-Dual and Out-of-Kilter Algorithms for Network Optimization Problems,” Networks, Vol 7, pp 81-88, 1977.

Shier, D. R., “On Algorithms for Finding the K Shortest Paths in a Network,” Networks, Vol. 9, pp 195-214, 1979.

Smith, B. C., J. F. Leimkuhler, R. M. Darrow, “Yield Management at American Airlines,” Interfaces 22: 1, pp 8-31, January-February 1992.

SRI International, User Description of a Dynamic Track Assignment Program: Cactus, Final Report, June 1983.

Strasser, S. ,” The Effect of Railroad Scheduling on Shipper Modal Selection: A Simulation,” Journal of Business Logistics, Vol 13, No. 2, 1993.

Stone, R. and M. Diamond, “Optimal Inventory Control for a Single Flight Leg,” Northwest Airlines, Operations Research Division, Working Paper, December 6, 1992.

Talluri , K. and G. Van Ryzin, “An Analysis of Bid-Price Controls for Network Revenue Management,” Columbia University Working Paper, October 31, 1997 (to appear in Management Science).

Tobin, R. L. and Friesz, T. L., “Sensitivity Analysis for Equilibrium Network Flow,” Transportation Science, Volume 22, Number 4, pp 242-250, 1988.

Turnquist, M. A. and M. S. Daskin, “Queueing Models of Classification and Connection Delay in Railyards,” Transportation Science, Volume 16, No. 2, pp. 207-230, May 1982.

Turnquist, M. A. and B. P. Markowicz, An Interactive Microcomputer-Based Planning Model for Railroad Car Distribution, presented at the CORS/ORSA/TIMS National Meeting, Vancouver, BC, Canada, May 1989.

Union Pacific Railroad, New Routes to Excellence, Employee newsletter, February 1995.

Van Dyke, C. D.,"The Automated Blocking Model: A Practical Approach to Freight Railroad Blocking Plan Development," Proceedings of the 27th Annual Meeting, Transportation Research Forum, 1986, #1.

Van Dyke, C. D. and L. C. Davis, "Railroad Capacity Planning: A Case Study of the Bejing-Shanghai Rail Corridor," Journal of the Transportation Research Forum, Vol 32, No. 1, pp. 86-102, 1991.

Wardrop, J. G., "Some Theoretical Aspects of Road Traffic Research," Proceedings, Institute of Civil Engineers, Part II (1), pp 325-378, 1952.

Welch, N. and J. Gussow, "Expansion of Canadian National Railway's Line Capacity", Interfaces 16:1, pp.51-64, 1986.

White, W. W. and E. Wrathall, A System For Railroad Traffic Scheduling, IBM Corp, August 1970.

Williamson, E. L., Airline Network Seat Inventory Control: Methodologies and Revenue Impacts, Ph. D. Dissertation, Department of Aeronautics and Astronautics, Flight Transportation Laboratory, Massachusetts Institute of Technology, 1992.

Wong, P. J. , C. V. Elliott and M. R. Hathorne, Demonstration of a Dynamic Track Assignment Program, Phase I Report, prepared for the AAR Freight Car Utilization Research-Demonstration program, June 1979.

Yagar, S. and F. F. Saccomanno, “An Efficient Sequencing Model for Humping in a Rail Yard,” Transportation Research A, Vol 17A, No 4, pp. 251-262, 1983.

Appendix A

Example Problems of Dynamic Car Scheduling and Train Segment Pricing, from Chapter 4

Appendix A gives several test examples pertaining to discussion in Chapter 4. The first four examples are very simple test problems used in the construction of Figure 4.8. These examples compare the performance of various combinations of Sweep Up/Sweep Down and Look Ahead/No Look ahead logic. Figure 4.8 shows that the Look Ahead definitely improves the performance of the Dynamic Car Scheduling algorithm, but is unable to conclude that any sweep direction is better than the other. The resolution of this issue will have to await rolling horizon simulation testing in Section 4.4.7.

Figures A.5 through A.9 present a very simple "toy" test problem subsequently used in figures A.10 through A.13 to demonstrate the steps of the Dynamic Car Scheduling algorithm. Figures 5.2 and 5.3 in Chapter 5 also pertain to this same test problem.

Figure A.14 demonstrates how the Subgradient Step Size procedure flips traffic flows back and forth without ever converging to a feasible solution. Surprisingly, in spite of this behavior the Step Size procedure is still able to adjust the dual variables to derive a tight lower bound. In this example we terminated the procedure after 10 iterations, by which time the duality gap had tightened to less than 1%.

The space/time network of Figure 4.6 is utilized in all four Figures A.1- A.4. Each segment has a capacity of 10 cars, which includes “overhead” A-C traffic carried on the thru block. These examples are referenced from Section 4.4.3 of the dissertation.

The train service network, yard costs and origin-destination demands shown in the following Figures A.5-A.9 are used in test examples A.10 - A.13.

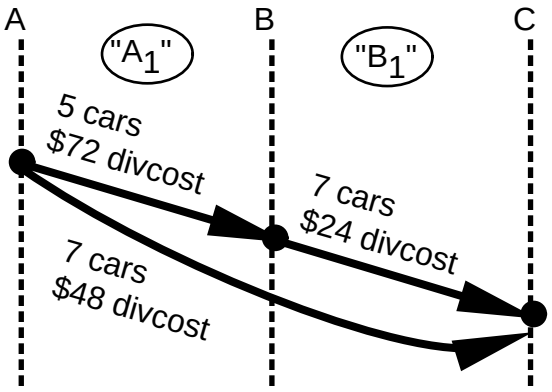
In Figures A.5 and A.6, Trains R123 (eastbound) and R134 (westbound) operate between stations #1 and #4, with an intermediate pickup and setout at location #2, bypassing location #3. Trains R556 (eastbound) and R557 (westbound) shuttle between locations #4 and #5. Location #4 is an intermediate classification yard where a connection must be made. The XPRS train is never used.

Local trains are specified by their symbols, locations, departure and arrival times as in Figure A.7. The main distinction is that local trains only serve customers at one location. Currently, only local trains can originate and terminate traffic. However, it would not be difficult to enhance the traffic input process to allow road trains to serve customers directly, or to allow more than one local train to serve the same customer.

The Yard Cost file, shown in Figure A.8, gives the fixed and variable cost per hour associated with terminal handling costs at each yard.

The Raw Demand file in Figure A.9 lists all origin to destination flows. For clarity, only eastbound flows are included in the test examples. The origin node is specified by location, originating local symbol and time. The destination node is specified only by location and terminating local symbol: the Dynamic Car Scheduling algorithm is permitted to choose the delivery time. The hourly cost depends on the car type used and the value of the commodity shipped, so it must be separately specified for every flow. Finally the number of cars in the shipment is listed. Length and tonnage of each individual shipment are planned for a future enhancement.

Fig. A.1: First Look Ahead Example



(A) Sweep Up.

(1) Apply \$24 increase to B_1, drive off 4 cars excess from local B-C block
(2) Apply \$24 increase to A_1, drive off 2 cars excess from thru A-C block

In this feasible solution, B_1 has been "overcorrected" by 2 cars

(B) Sweep Up with Look Ahead.

(1) Apply \$24 increase to B_1, drive off only 2 cars from local B-C block
(2) Apply \$24 increase to A_1, drive off 2 cars from thru A-C block

Produces a feasible, optimal solution

(C) Sweep Down

(1) Apply \$48 increase to A_1, driving off 2 cars from thru A-C block
(2) To repair segment B_1, a further two cars can be driven from A-C block without requiring any further price adjustments

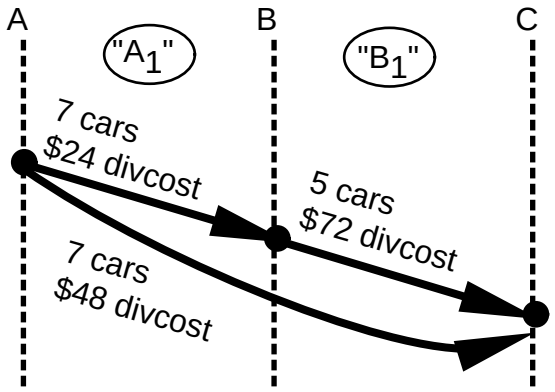
This solution leaves A_1 overcorrected by 2 cars.

(4) Sweep Down with Look Ahead

(1) Bypass A_1, because joint traffic with downstream B_1 exceeds A_1 overflow.
(2) Apply \$24 increase to B_1, drive off 4 cars excess from local B-C block
(3) Apply \$24 increase to A_1, drive off 2 cars excess from thru A-C block

In this feasible solution, B_1 has been "overcorrected" by 2 cars

Fig. A.2: Second Look Ahead Example



(A) Sweep Up.

(1) Apply \$48 increase to B_1, drive off 2 cars excess from thru A-C block
(2) To repair segment A_1, a further two cars can be driven from A-C block without requiring any further price adjustments

In this feasible solution, B_1 has been "overcorrected" by 2 cars

(B) Sweep Up with Look Ahead.

(1) Bypass B_1, because joint traffic with upstream A_1 exceeds B_1overflow.
(2) Apply \$24 increase to A_1, drive off 4 cars excess from local A-B block
(3) Apply \$24 increase to B_1, drive off 2 cars excess from thru A-C block

In this feasible solution, A_1 has been "overcorrected" by 2 cars

(C) Sweep Down

(1) Apply \$24 increase to A_1, drive off 4 cars excess from local A-B block
(2) Apply \$24 increase to B_1, drive off 2 cars excess from thru A-C block

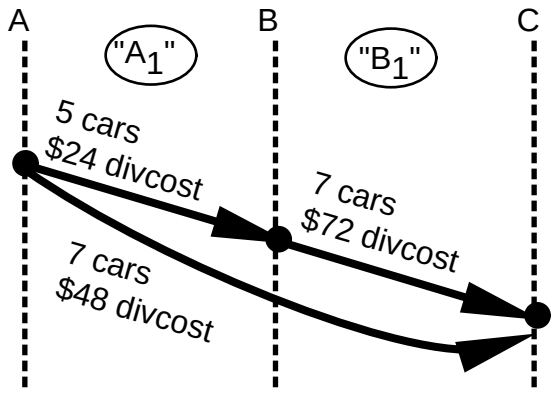
This solution leaves A_1 "overcorrected" by 2 cars

(4) Sweep Down with Look Ahead

(1) Apply \$24 increase to A_1, drive off only 2 cars from local A-B block
(2) Apply \$24 increase to B_1, drive off 2 cars from thru A-C block

Produces a feasible, optimal solution

Fig. A.3: Third Look Ahead Example



(A) Sweep Up.

(1) Apply $48 increase to B_1, drive off 4 cars excess from thru A-C block
(2) Segment A_1 is "fixed" and does not require any further adjustments

Produces a feasible, optimal solution

(B) Sweep Up with Look Ahead.

(1) Apply a $48 increase to B_1, but only divert 2 cars.
(2) To repair segment A_1, a further two cars can be driven from A-C block without requiring any further price adjustments

Produces a feasible, optimal solution

(C) Sweep Down

(1) Apply $24 increase to A_1, drive off 2 cars excess from local A-B block
(2) Apply $24 increase to B_1, drive off 4 cars excess from thru A-C block

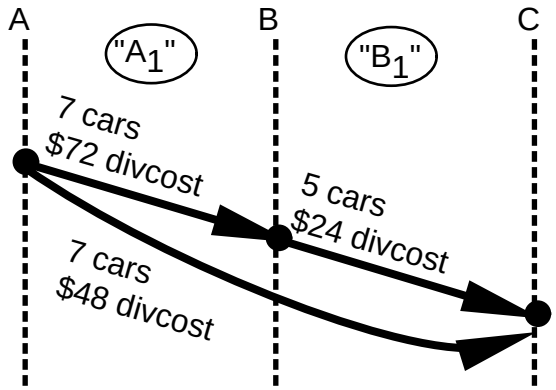
This solution leaves A_1 "overcorrected" by 2 cars

(4) Sweep Down with Look Ahead

(1) Bypass A_1, because joint traffic with downstream B_1 exceeds A_1overflow.
(2) Apply $48 increase to B_1, drive off 4 cars excess from thru A-C block
(4) Segment A_1 is "fixed" and does not require any further adjustments

Produces a feasible, optimal solution

Fig. A.4: Fourth Look Ahead Example



(A) Sweep Up.

(1) Apply \$24 increase to B_1, drive off 2 cars excess from local B-C block
(2) Apply \$24 increase to A_1, drive off 4 cars excess from thru A-C block

This solution leaves B_1 "overcorrected" by 2 cars

(B) Sweep Up with Look Ahead

(1) Bypass B_1, because joint traffic with upsttream A_1 exceeds B_1overflow.
(2) Apply \$48 increase to A_1, drive off 4 cars excess from thru A-C block
(3) Segment B_1 is "fixed" and does not require any further adjustments

Produces a feasible, optimal solution

(C) Sweep Down.

(1) Apply \$48 increase to A_1, drive off 4 cars excess from thru A-C block
(2) Segment B_1 is "fixed" and does not require any further adjustments

Produces a feasible, optimal solution

(D) Sweep Down with Look Ahead.

(1) Apply a \$48 increase to A_1, but only divert 2 cars.
(2) To repair segment B_1, a further two cars can be driven from A-C block without requiring any further price adjustments

Produces a feasible, optimal solution

Figure A.5: Train Service Network used in Test Examples

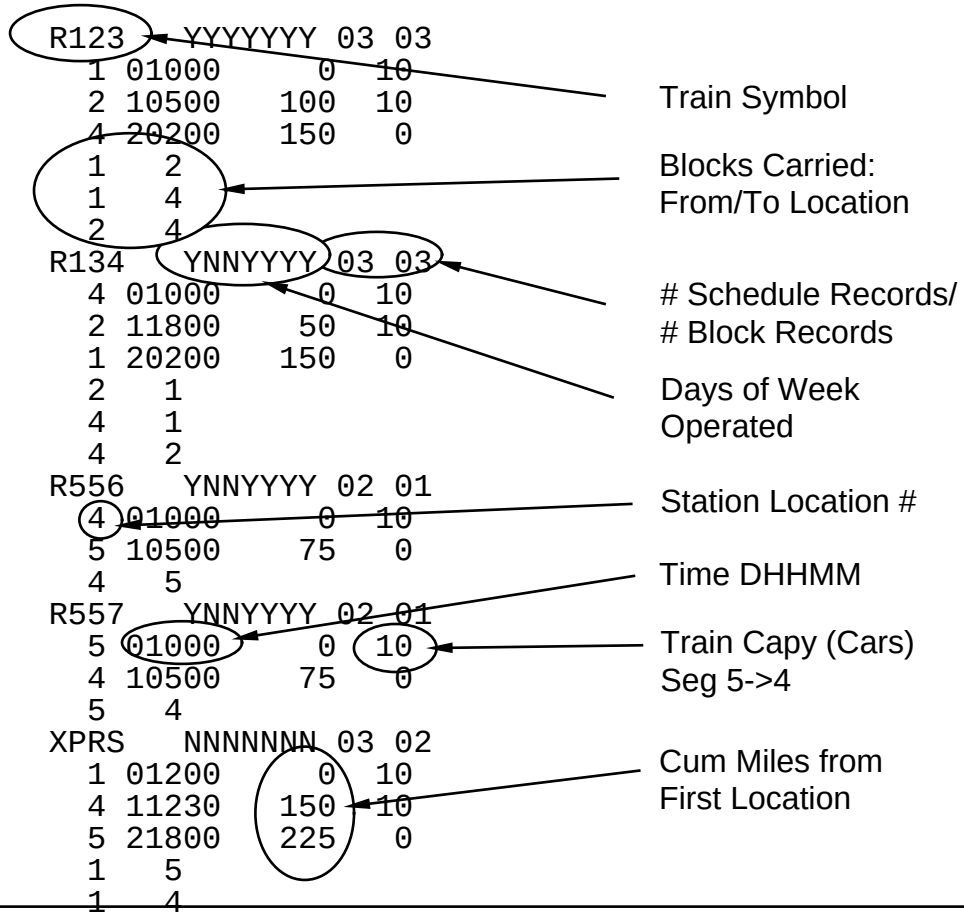


Figure A.6: Train Service Blocking Network

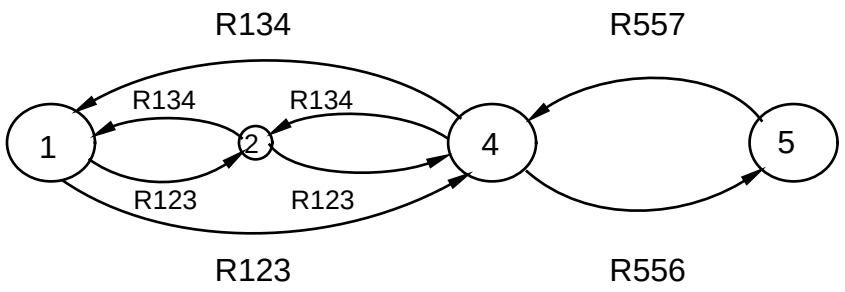


Figure A.7: Local Trains for Test Examples

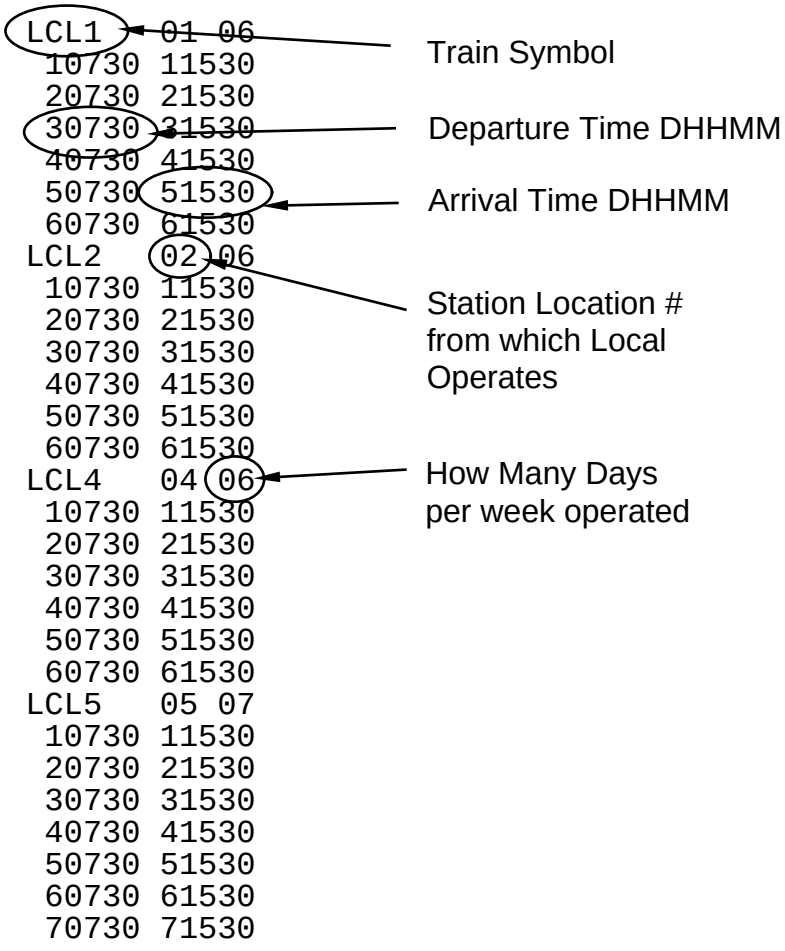


Figure A.8: Yard Cost File

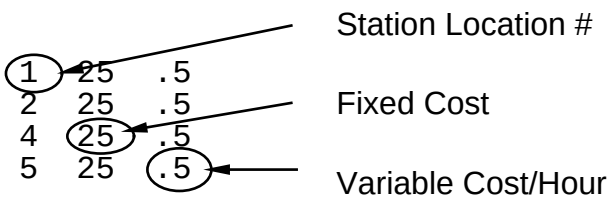


Figure A.9: Raw Demand File

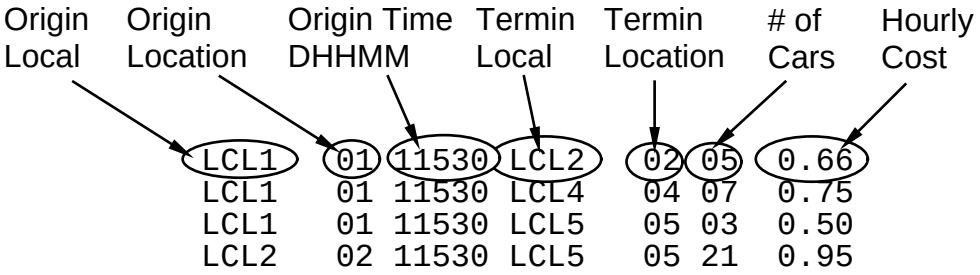
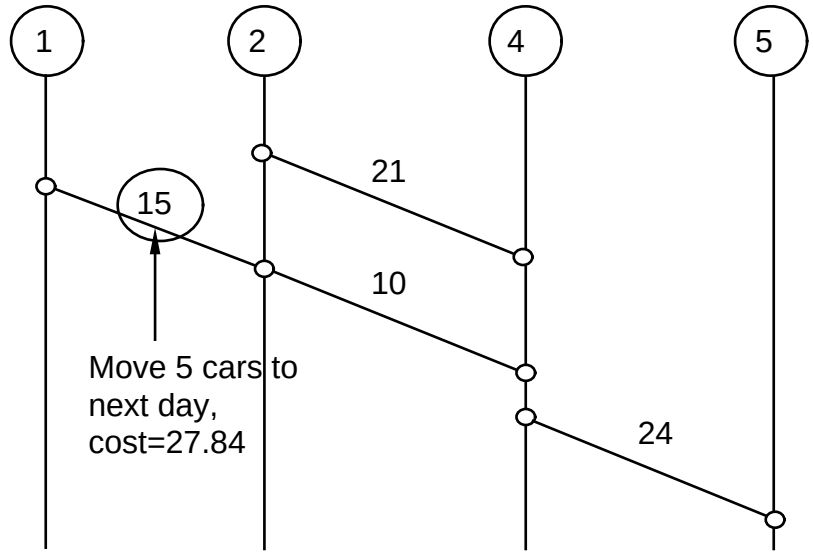


Fig A.10: Sweep Up with Look Ahead:
10 Cars Capacity

Initial Shortest Path Assignment



Iteration #1

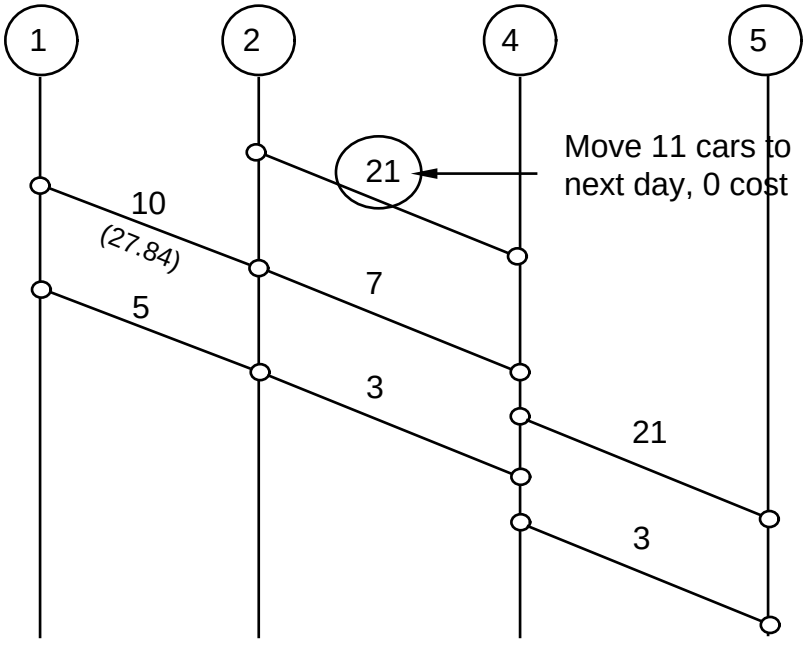
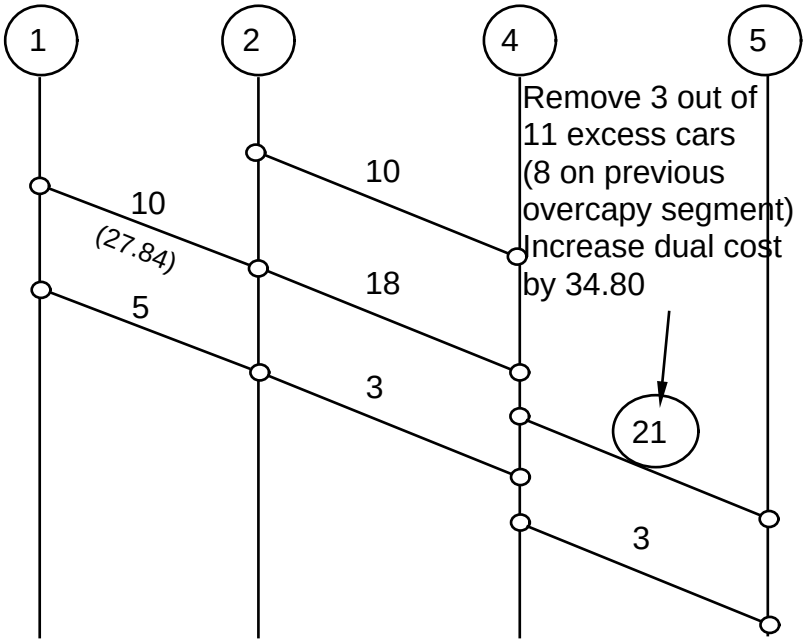


Fig A.10
(ctd)

Iteration #2



Iteration #3

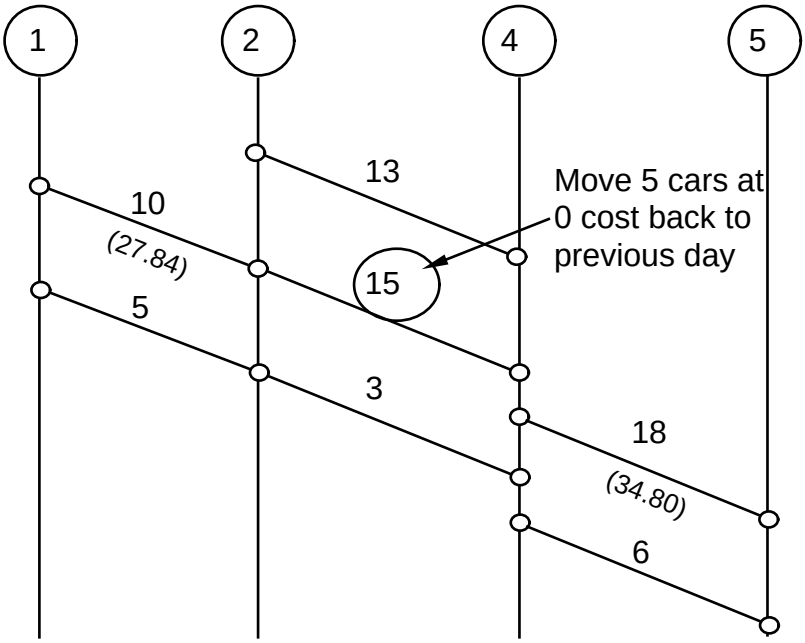
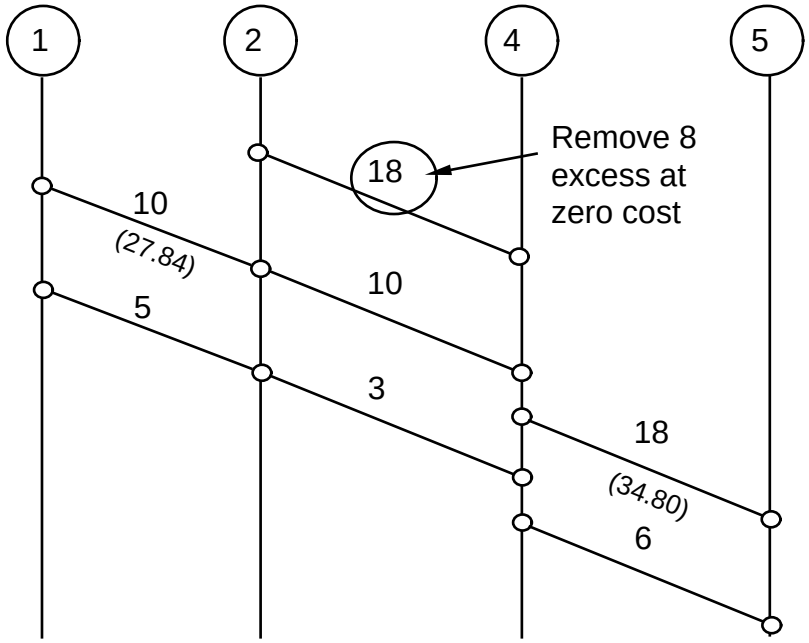


Fig A.10 (ctd)

Iteration #4



Iteration #5

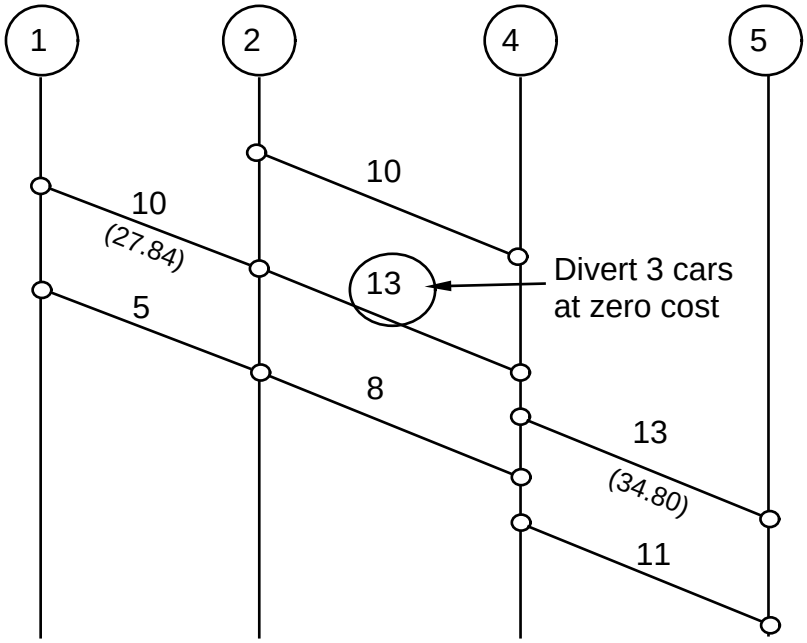
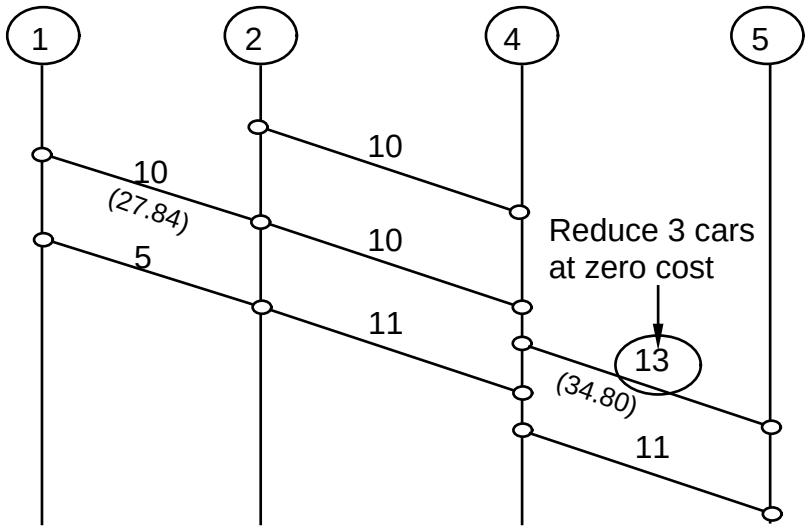


Fig A.10 (ctd)

Iteration #6



Iteration #7

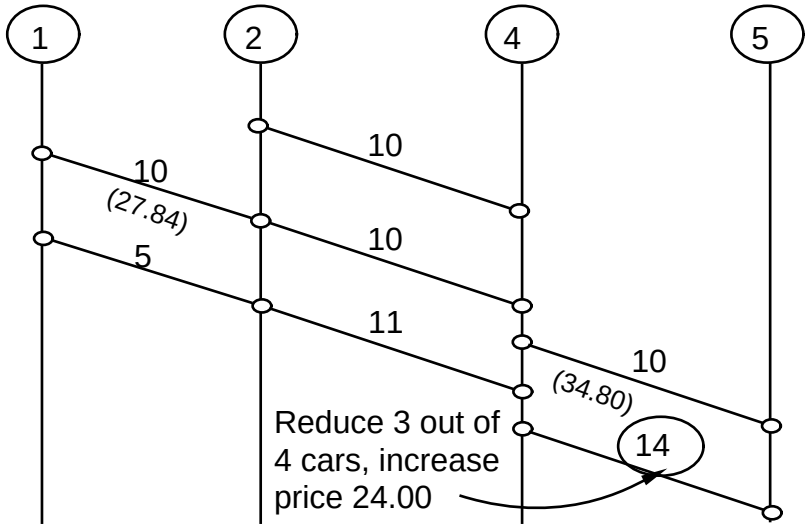
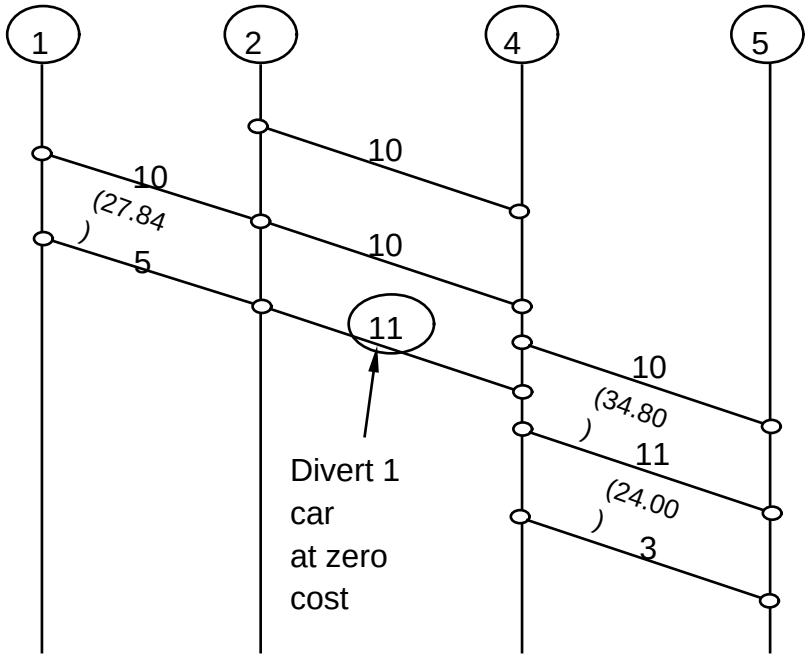


Fig A.10 (ctd)

Iteration #8



Iteration #9

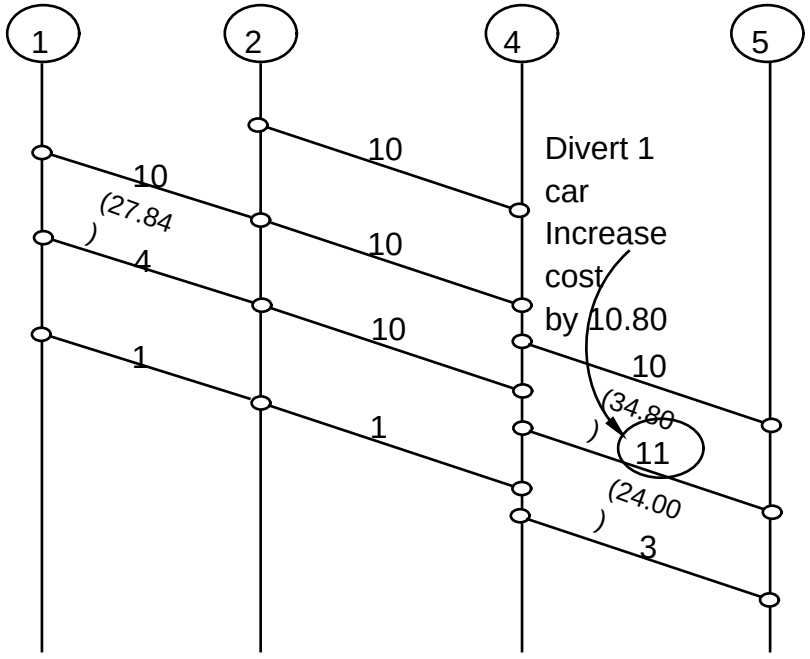
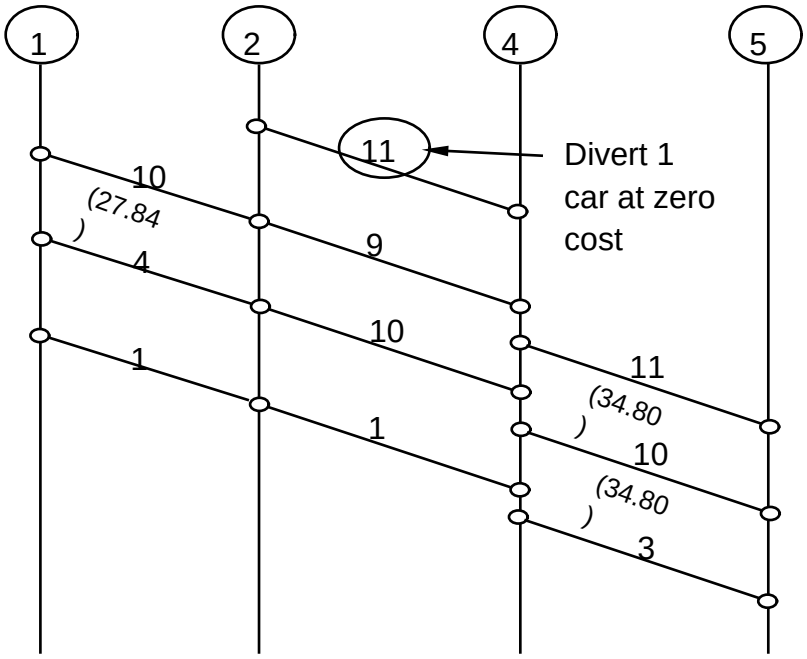


Fig A.10 (ctd)

Iteration #10



Iteration #11

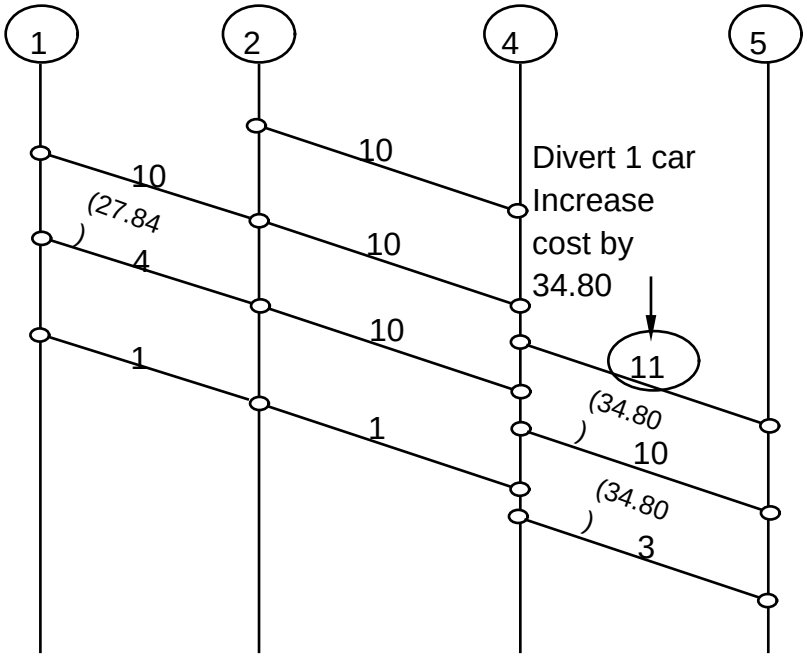
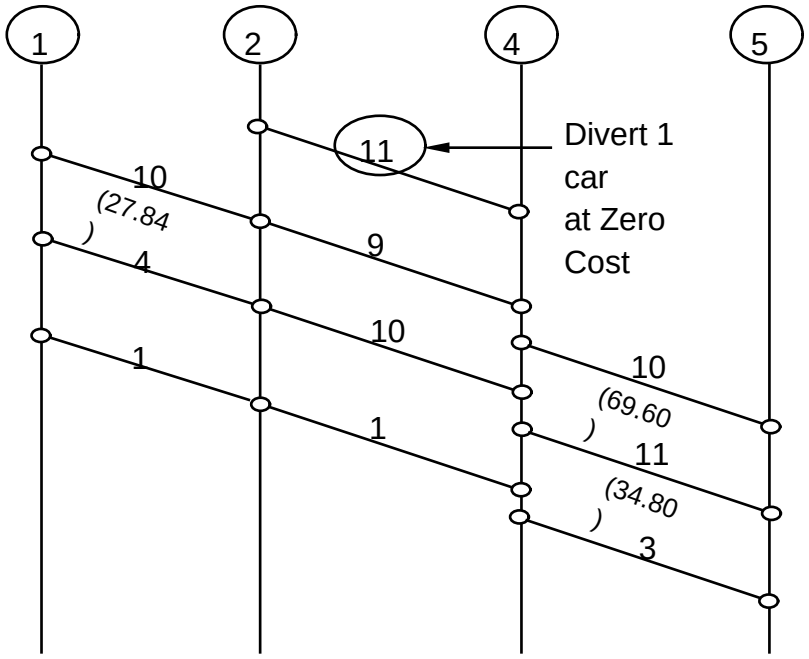


Fig A.10 (ctd)

Iteration #12



Iteration #13

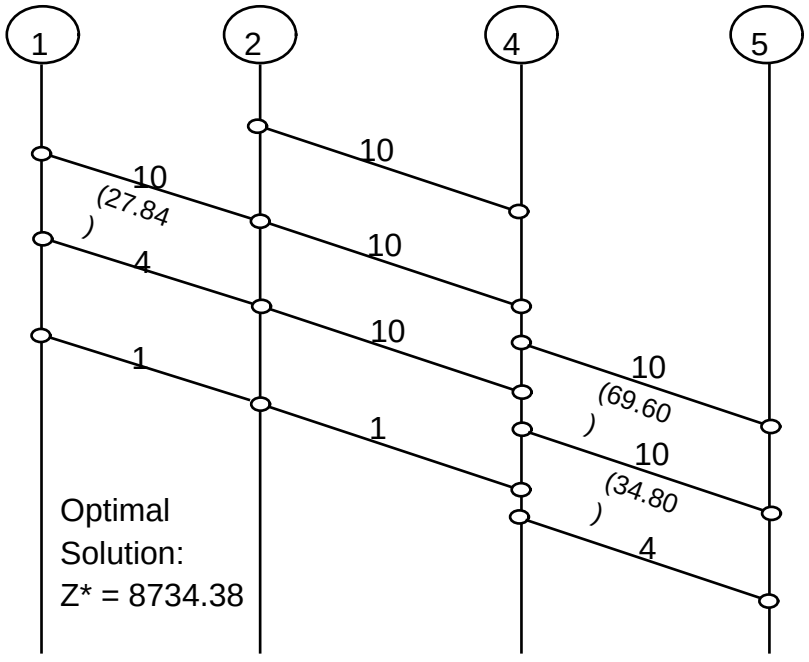
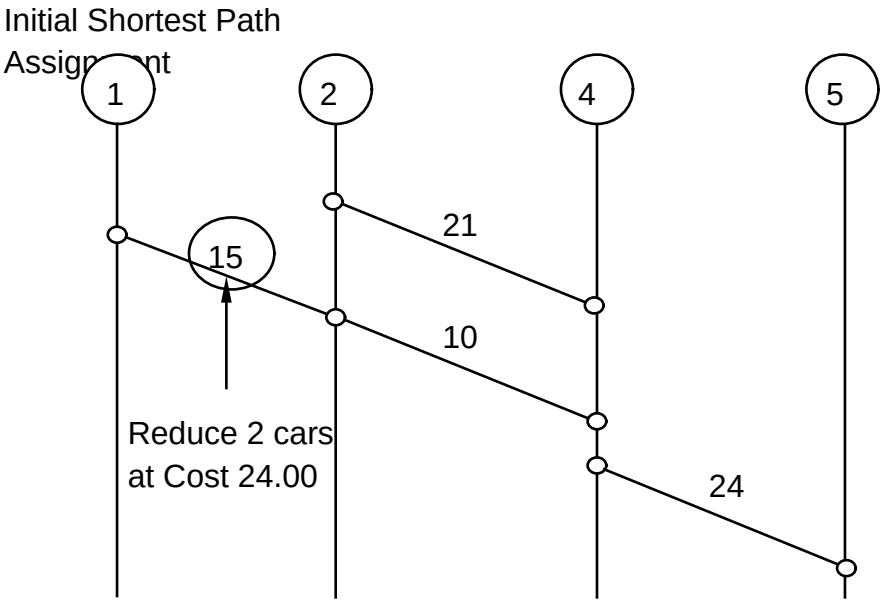


Fig A.11: Sweep Down with Look Ahead:
10 Cars Capacity



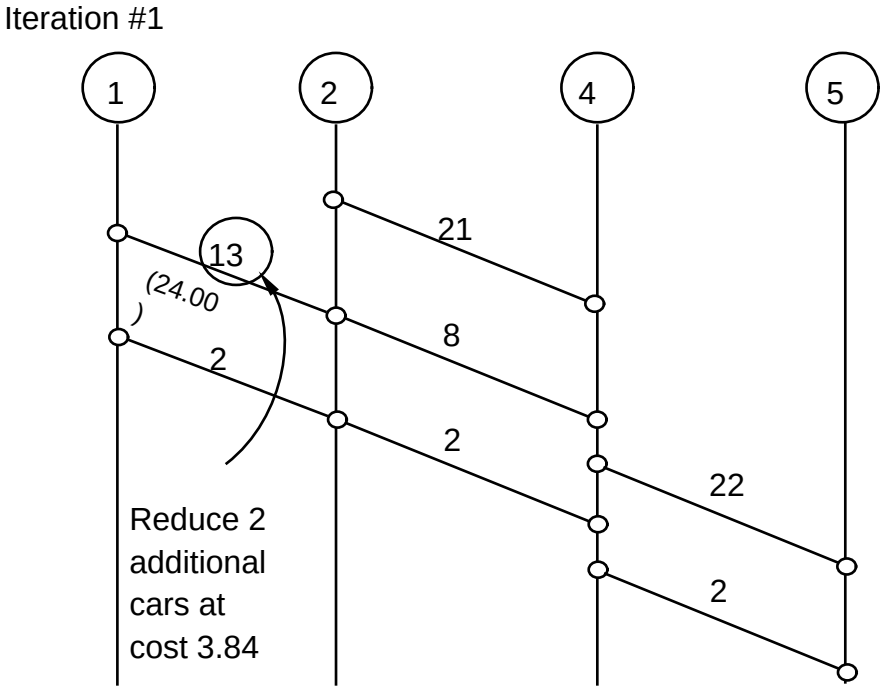
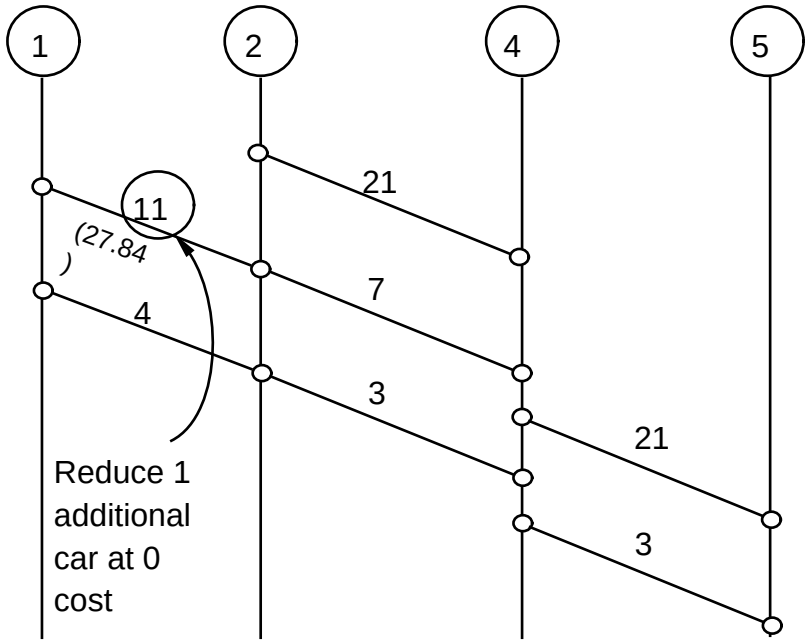


Fig A.11 (ctd)

Iteration #2



Iteration #3

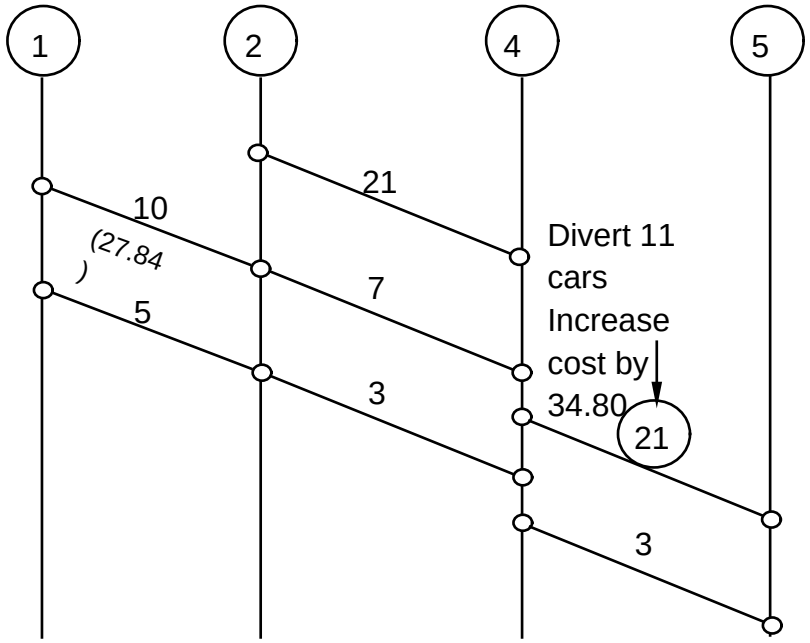
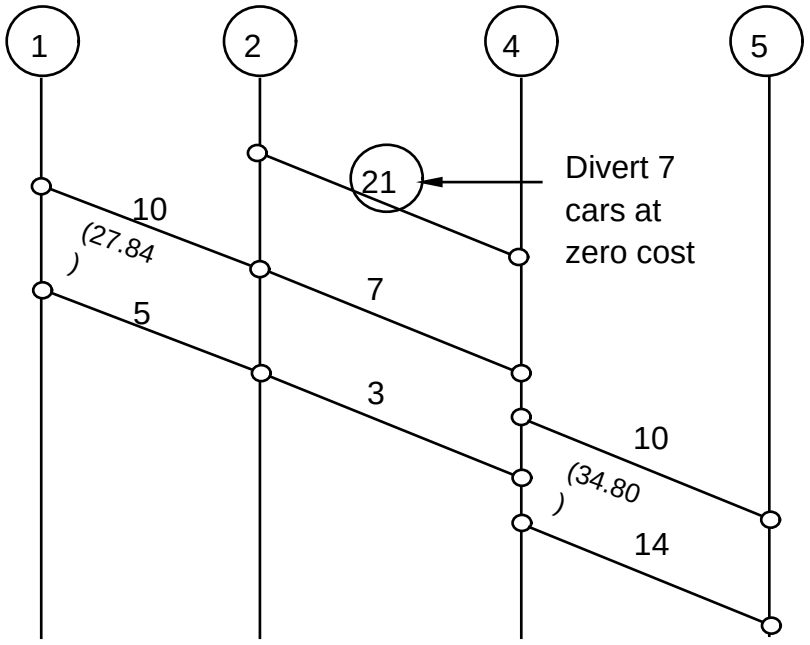


Fig A.11 (ctd)

Iteration #4



Iteration #5

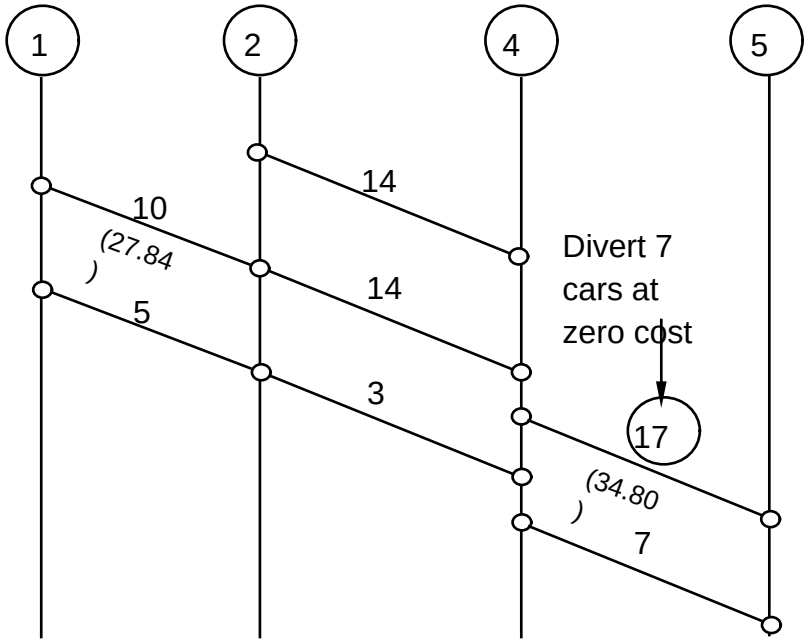
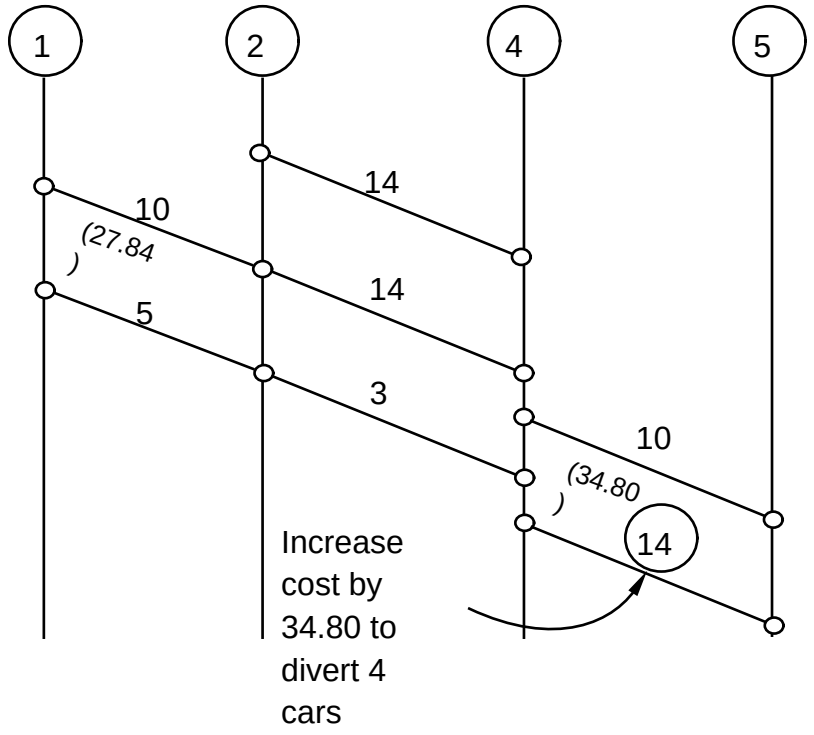


Fig A.11 (ctd)

Iteration #6



Iteration #7

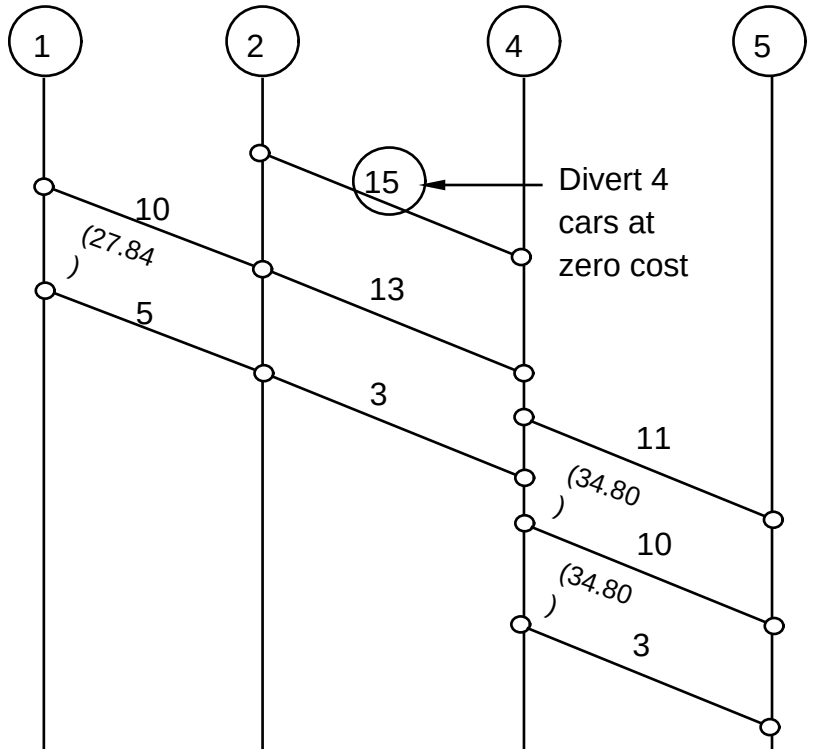
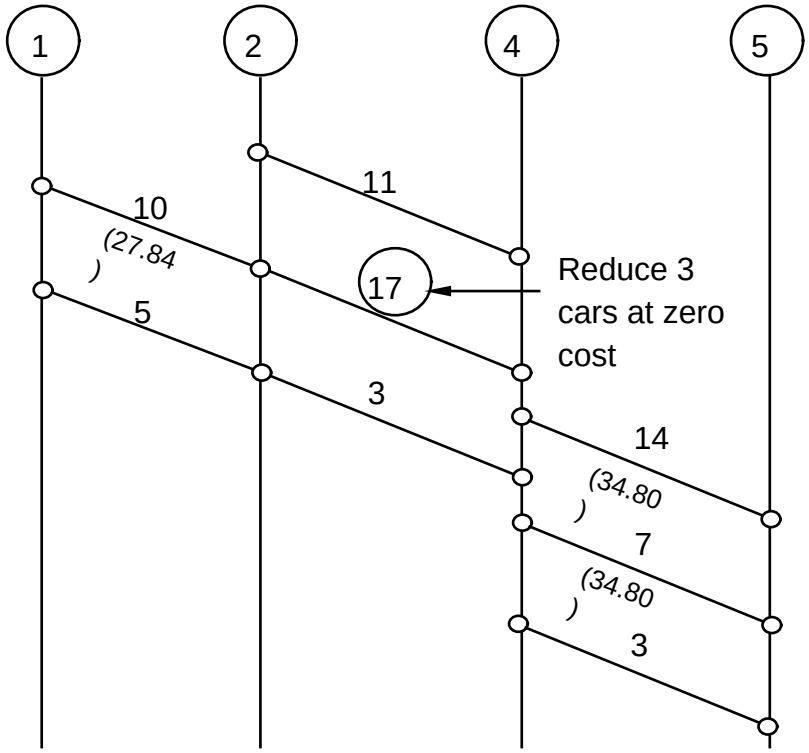


Fig A.11 (ctd)

Iteration #8



Iteration #9

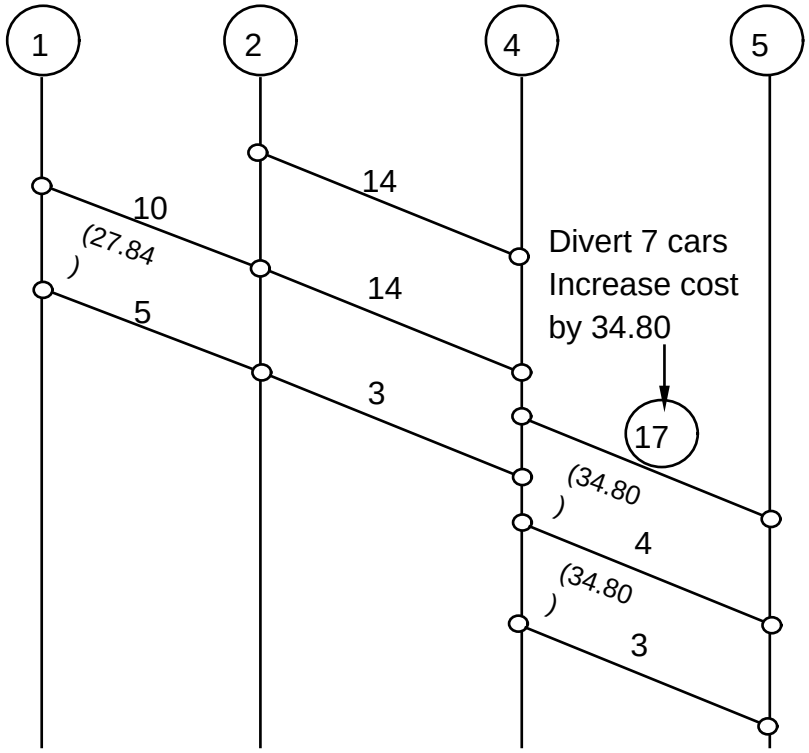
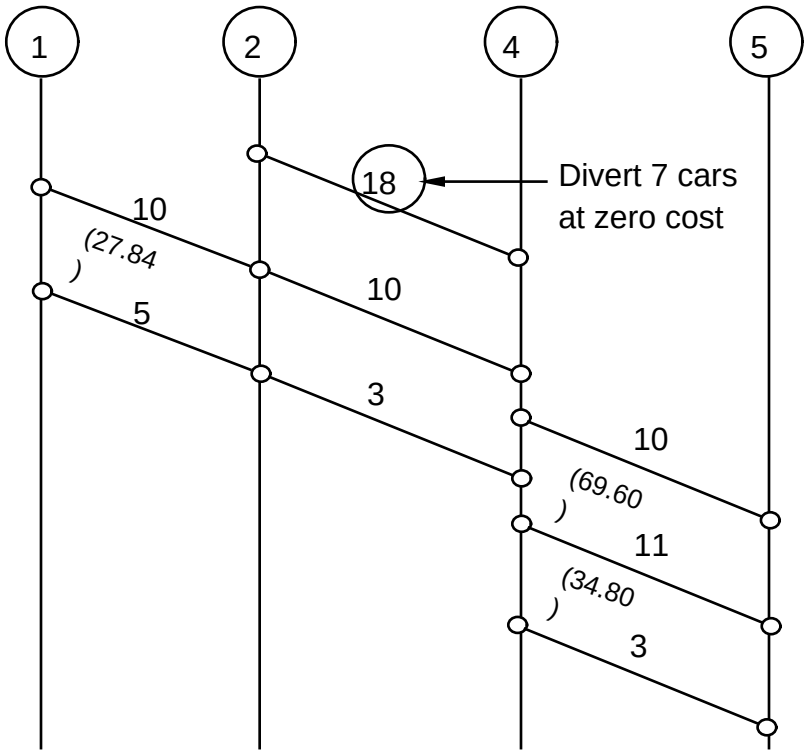


Fig A.11 (ctd)

Iteration #10



Iteration #11

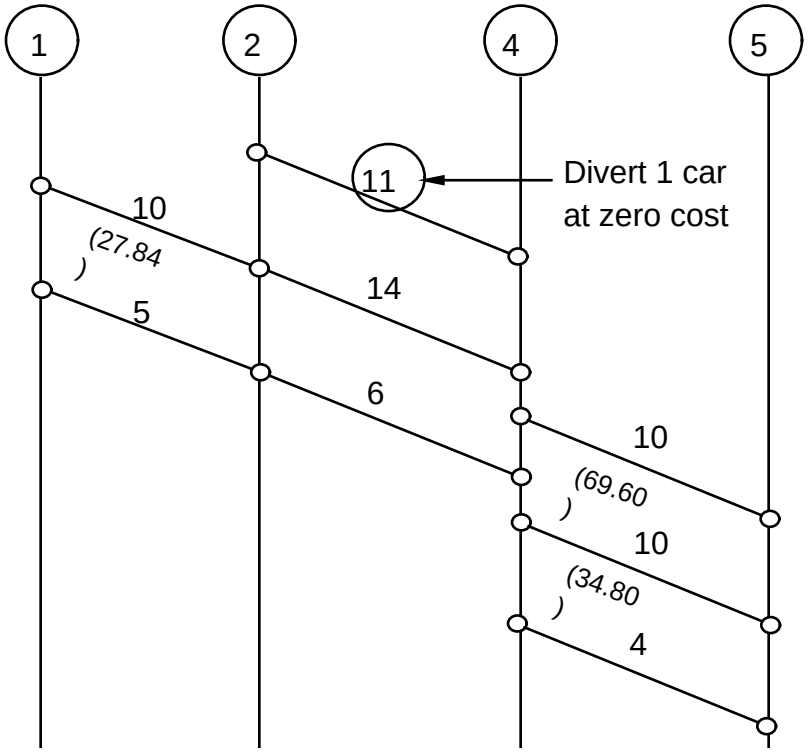
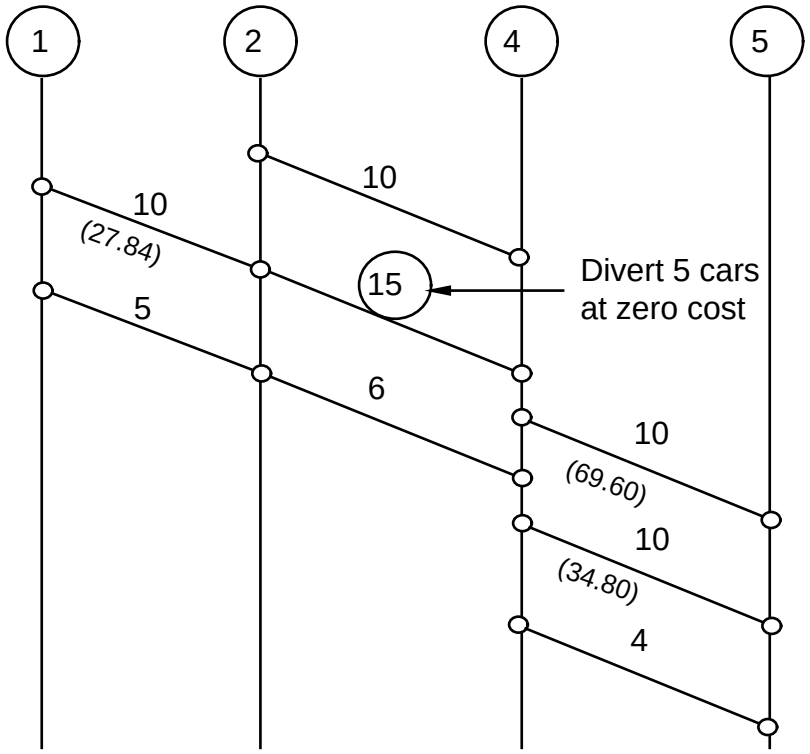


Fig A.11 (ctd)

Iteration #12



Iteration #13

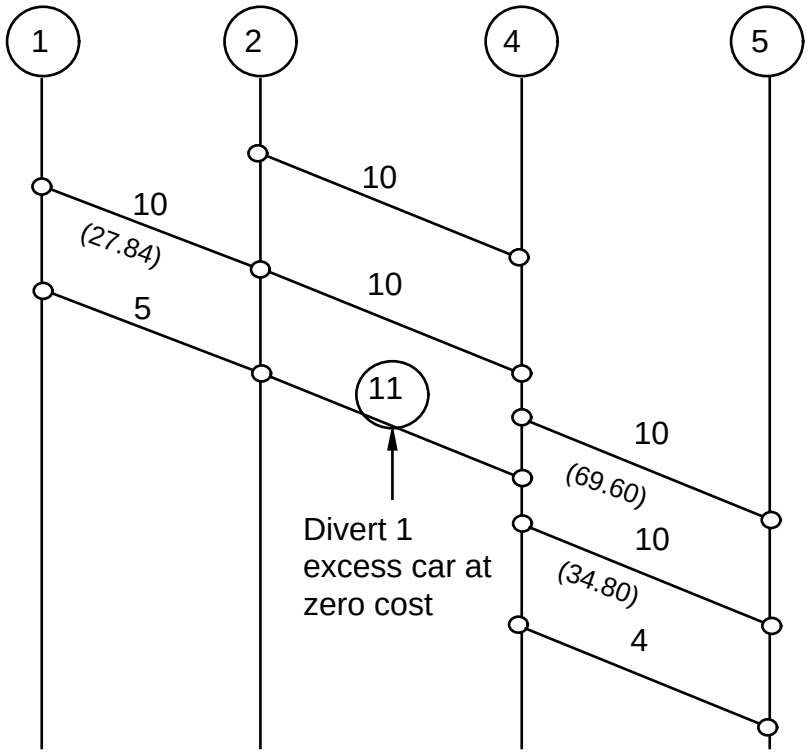


Fig A.11 (ctd)

Iteration #14

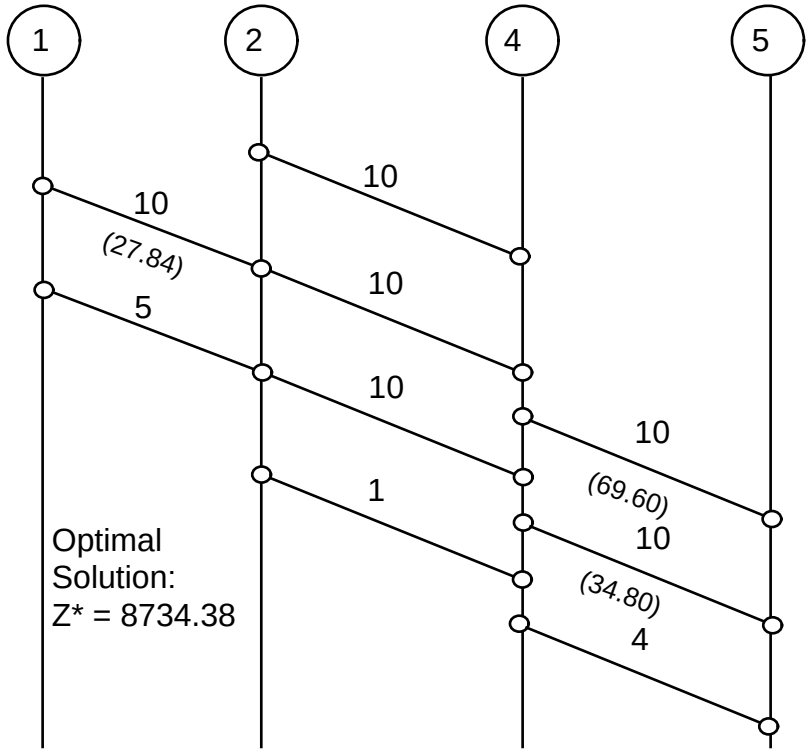
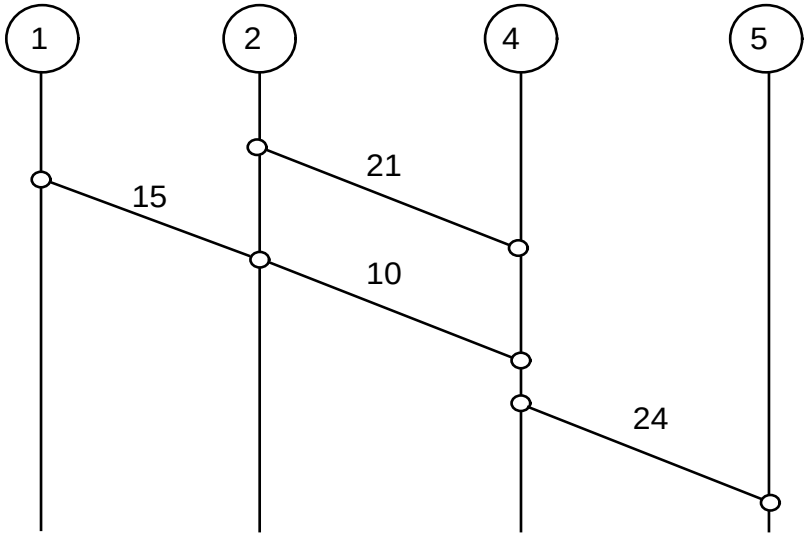


Fig A.12: Sweep Up with Look Ahead:
6 Cars Capacity

Initial Shortest Path Assignment



First Feasible Solution

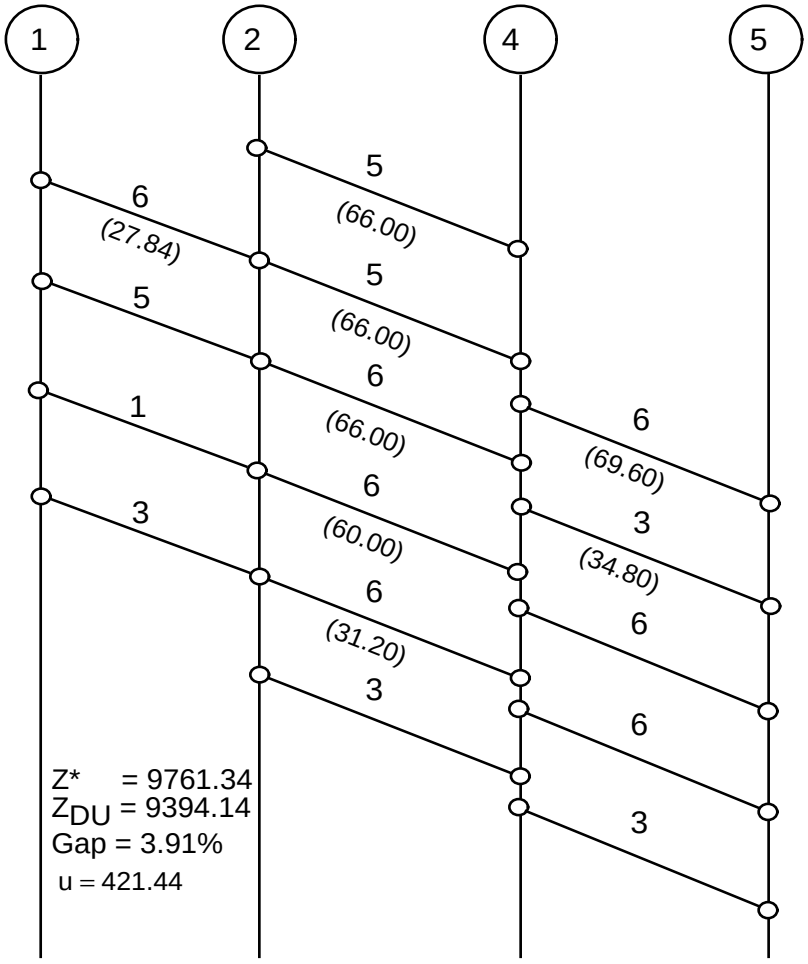


Fig A.12 (ctd)

Second Feasible Solution

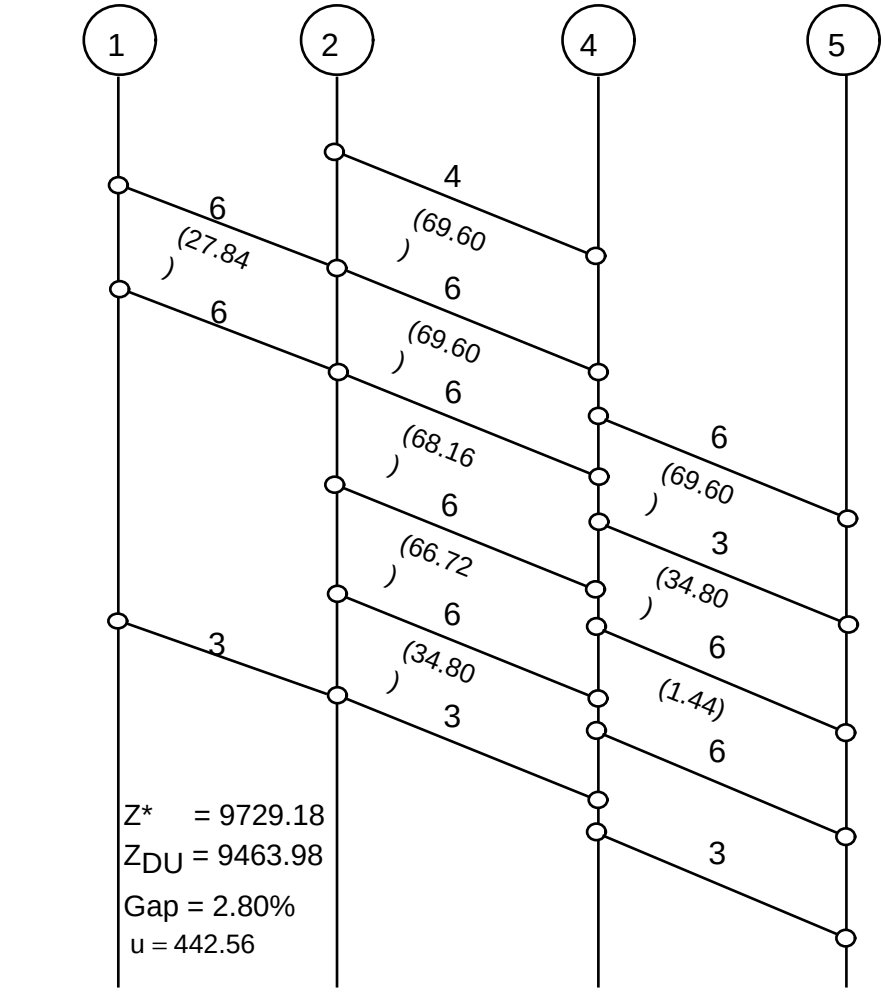


Fig A.12 (ctd)

Third Feasible Solution

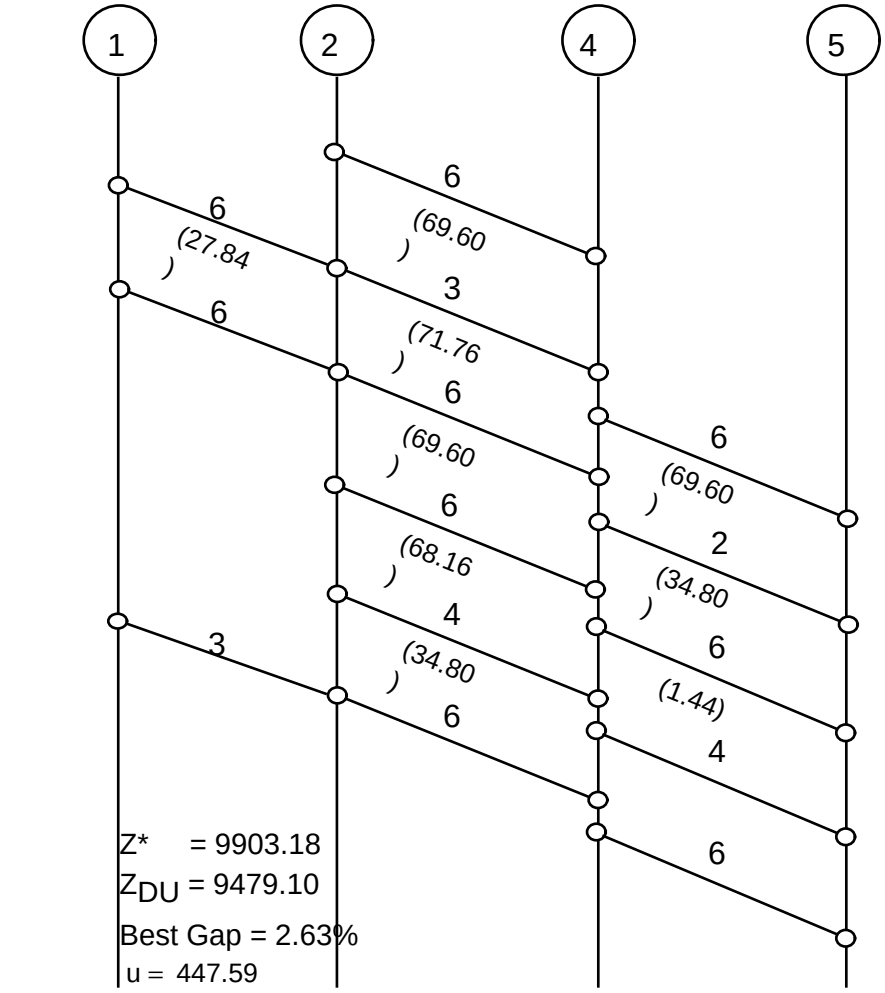
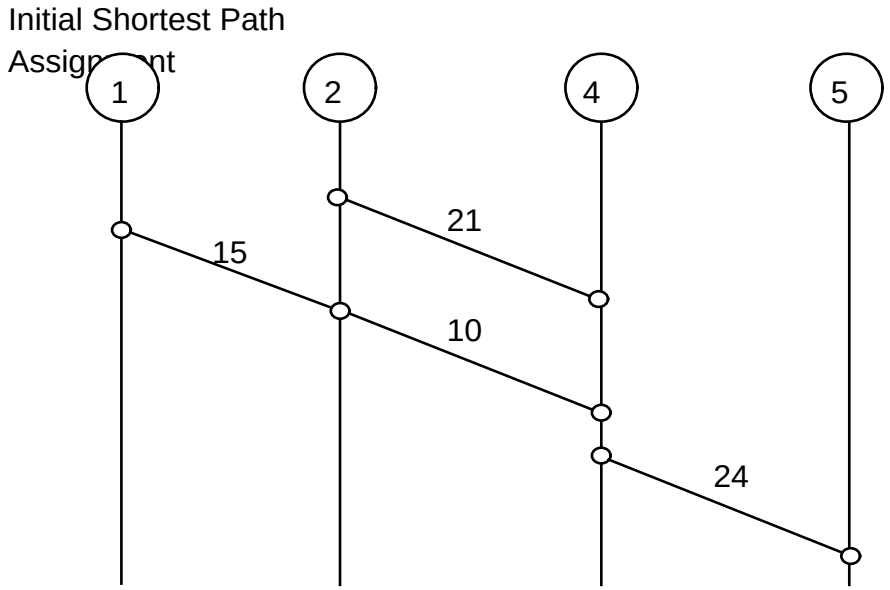


Fig A.13: Sweep Down with Look Ahead:
6 Cars Capacity



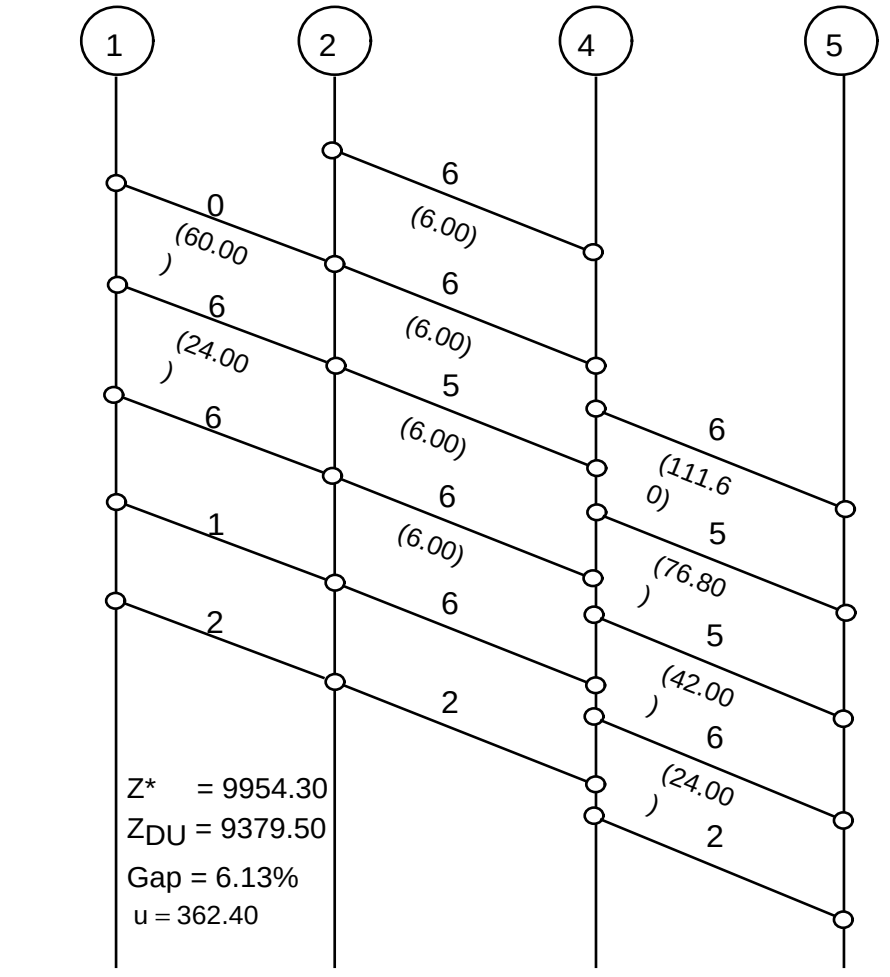


Fig A.13 (ctd)

Second Feasible Solution

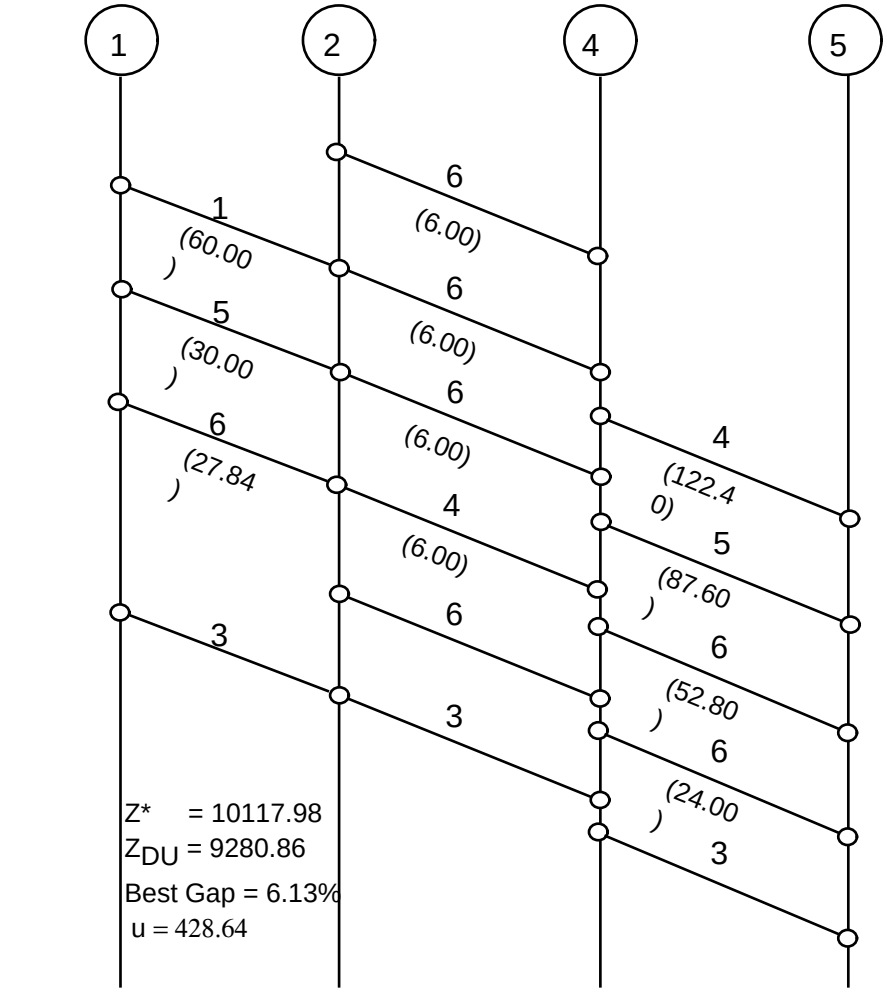


Fig A.13 (ctd)

Third Feasible Solution

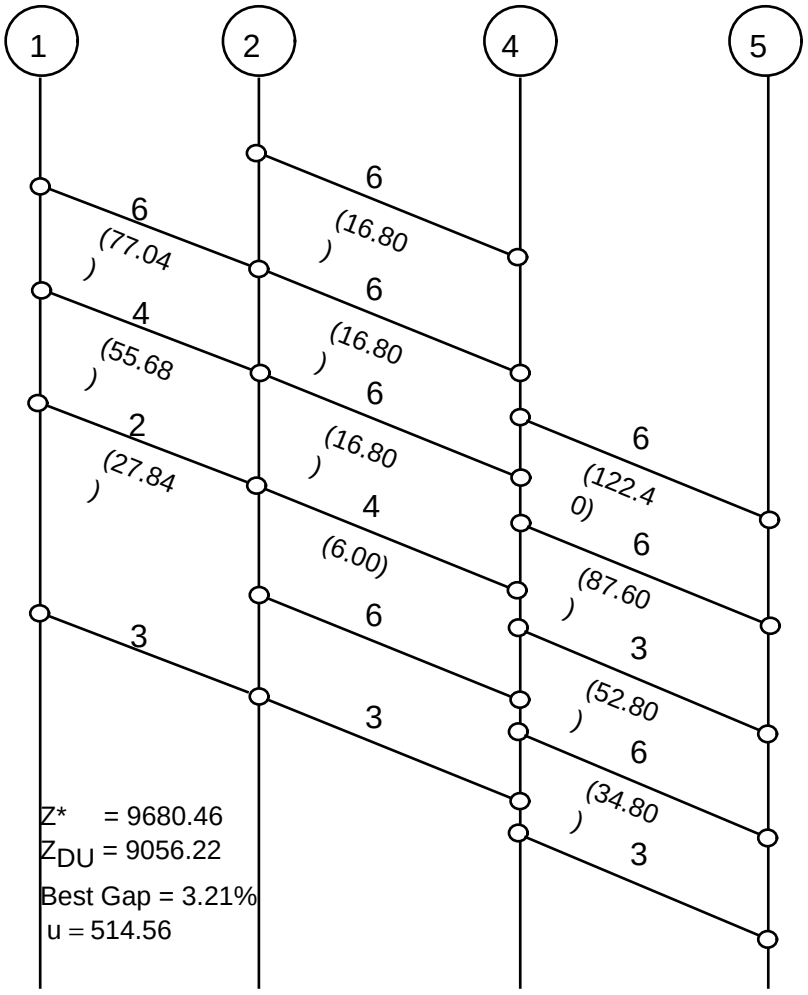


Fig A.13 (ctd)

Eleventh Feasible Solution

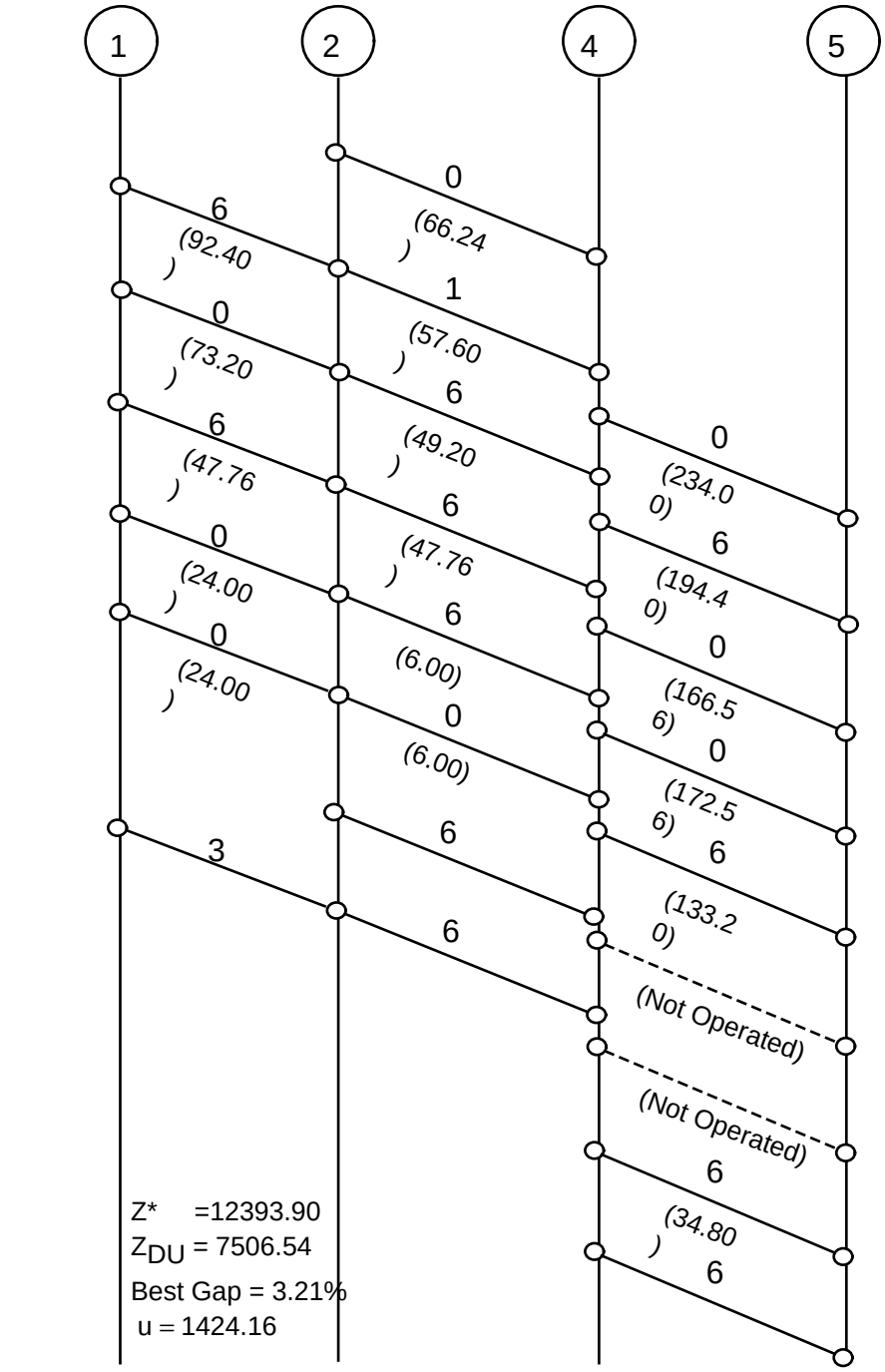
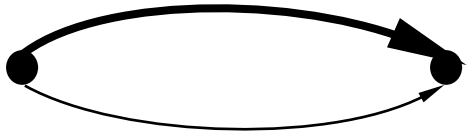


Fig. A.14: Standard Step Size Procedure Fails to Solve Linear Network Flow Problems But Can Still Produce a Tight Lower Bound

Initial Solution

Top Route: Cost $10
Flow 15 units Capy 10 units

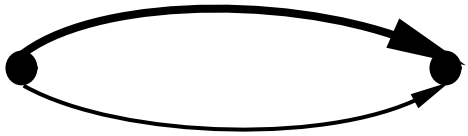


Z_{DU} = (15)(10) = $150

Bottom Route: Cost $12
Flow 0 units Capy 10 units

Optimal Solution will not be reached following stepsize procedure

Top Route:
Cost $10 + $ 2 adjustment = $12
Flow 10 units Capy 10 units

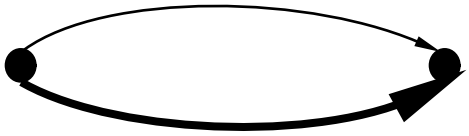


Z* = (10)(10) + (12)(5) = 160
Gap = (160 - 150) /160 = 6.25%

Bottom Route: Cost $12
Flow 5 units Capy 10 units

Cost Adj #1 w Stepsize = $0.50

Top Route:
Cost $10 + (5)(.50) = $12.50
Flow 0 units Capy 10 units



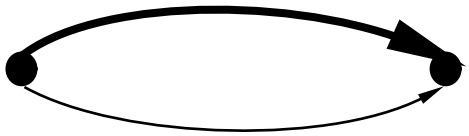
Z_{DU} = (15)(12) - (10)(2.50) = 155
Gap = (160 - 155) /160 = 3.125%

Bottom Route: Cost $12
Flow 15 units Capy 10 units

Fig. A.14 (ctd)

Cost Adj #2 w Stepsize = $0.50

Top Route:
Cost $12.50 - $2.50 = $ 10
($2.50 is the most that can be subtracted due to neg deviation)
Flow 15 units Capy 10 units

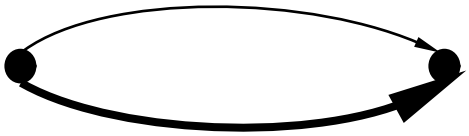


Z_{DU} = (15)(10) - (10)(2.50) = 125

Bottom Route:
Cost $12 + (5)(.50) = $14.50
Flow 0 units Capy 10 units

Cost Adj #3 w Stepsize = $0.50

Top Route:
Cost $10 + (5)(.50) = $12.50
Flow 0 units Capy 10 units



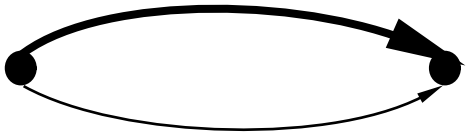
Z_{DU} = (15)(12) - (10)(2.50) = 155

Bottom Route:
Cost $14.50 - $2.50 = $12
($2.50 is the most that can be subtracted)
Flow 15 units Capy 10 units

Note this is the same outcome as Step #1, reduce step size to $0.25

Cost Adj #4 w Stepsize = $0.25

Top Route:
Cost $12.50 - (10)(.25) = $10.00
Flow 15 units Capy 10 units



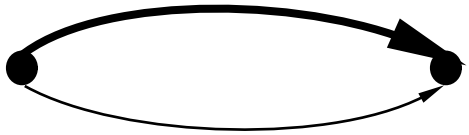
Z_{DU} = (15)(10) - (10)(1.25) = 137.5

Bottom Route:
Cost $12.00 + (5)(.25) = $13.25
Flow 0 units Capy 10 units

Fig. A.14 (ctd)

Cost Adj #5 w/Stepsize = $0.25

Top Route:
Cost $10.00 + (5)(.25) = $11.25
Flow 15 units Capy 10 units

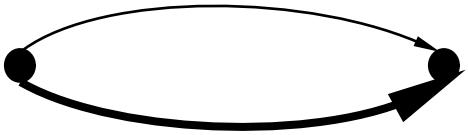


Z_{DU} = (15)(11.25) - (10)(1.25) = 156.25
Gap = (160 - 156.25) /160 = 2.34 %

Bottom Route:
Cost $13.25 - $1.25 = $12.00
($1.25 is the maximum reduction)
Flow 0 units Capy 10 units

Cost Adj #6 w Stepsize = $0.25

Top Route:
Cost $11.25 + (5)(.25) = $12.50
Flow 0 units Capy 10 units

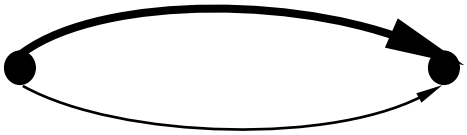


Z_{DU} = (15)(12) - (10)(2.50) = 160

Bottom Route:
Cost $12.00
Flow 15 units Capy 10 units

Cost Adj #7 w Stepsize = $0.25

Top Route:
Cost $12.50 - (10)(.25) = $10.00
Flow 15 units Capy 10 units



Z_{DU} = (15)(10) - (10)(1.25) = 137.5

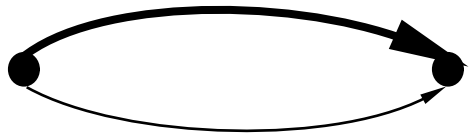
Bottom Route:
Cost $12.00 + (5)(.25) = $13.25
Flow 0 units Capy 10 units

Note this is the same outcome as Step #4, reduce step size to $0.125

Fig. A.14 (ctd)

Cost Adj #8 w Stepsize = $0.125

Top Route:
Cost $10.00 + (5)(.125) = $10.625
Flow 15 units Capy 10 units

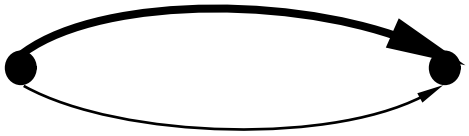


Z_{DU} = (15)(10.625) - (10)(.625) = 153.125

Bottom Route:
Cost $13.25 - (10)(.125) = $12.00
Flow 0 units Capy 10 units

Cost Adj #9 w Stepsize = $0.125

Top Route:
Cost $10.625 + (5)(.125) = $11.25
Flow 15 units Capy 10 units

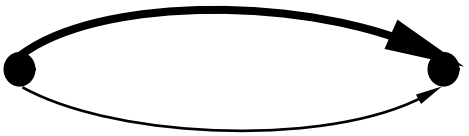


Z_{DU} = (15)(11.25) - (10)(1.25) = 156.25

Bottom Route:
Cost $12.00
Flow 0 units Capy 10 units

Cost Adj #10 w Stepsize = $0.125

Top Route:
Cost $11.25 + (5)(.125) = $11.875
Flow 15 units Capy 10 units



Z_{DU} = (15)(11.875) - (10)(1.875) = 159.375
Gap = (160 - 159.375) /160 = 0.39 %

Bottom Route:
Cost $12.00
Flow 0 units Capy 10 units

Appendix B

Adapting Kwon's [1994] Test Problem for the Rolling Horizon Simulation

This Appendix describes how Kwon's problem was adapted for the Rolling Horizon simulation testing of Chapter 5.

Appendix B of Kwon's [1994] dissertation gives a test problem containing 12 terminals and one weeks' traffic consisting of 1250 shipments. Several inconsistencies were noted in Kwon's data set:

- Some trains had extra stations where they neither picked up nor set off cars. To avoid creating unnecessary train route segments, these extra stations were removed from the input train schedules.
- A few blocks were assigned to trains not serving those locations. In particular, no schedules were given for train 809 between stations 3-1, nor for train 908 between 9-10. Any blocks carried on 809 or 908 between these points were dropped from the input file.
- There are some inconsistencies between Kwon's traffic data base and his reported train load factors. Kwon reported train load factors between 5%-40% between stations 4-5 on train 261, however, the traffic data base has no shipments terminating at station 5. Since

station 5 is at the end of the line, our simulation gives a zero load factor for this train. It is impossible to know if any other shipments are missing from the published traffic data set.

Several input data items required by our model are not available in Kwon's test data. Cost per hour, revenue per car and priority coefficient were randomly generated for each shipment by sampling from a uniform distribution:

- Cost per hour = U($2,$12)
- Revenue = U($1000,$2000)
- Logit Coefficient for Acceptance Function = U(1,5)

In Kwon's problem, traffic is always specified as an integral number of cars, and the same traffic pattern is repeated across many different origin destination pairs. In a real problem, such uniformity would be unlikely. To make the test problem more realistic, Kwon's original traffic flow values were randomized by selecting new values from a uniform distribution, where "N" is the number of cars specified in Kwon's original test problem and $\mu = U(0,N)$. This μ replaces Kwon's original traffic as the expected number of cars for each shipment. In the simulation, it is used as input to a random number generator to determine the actual number of cars each time a new shipment is called in.

Since on average this transformation cuts traffic in half, our "Base Case" train capacities had to be correspondingly reduced to 60% of Kwon's original value. A full 50% reduction would have produced a more difficult problem than Kwon's original test scenario. If capacity were reduced by the full 50%, during peak days, more traffic would need to be moved than in Kwon's problem, but a full 50% capacity reduction would not provide any slack capacity for recovering from such days. As well, because portions of trains 809 and 908 had to be dropped, there is good reason to believe that this test problem may actually be more difficult than Kwon's.

Figure B.1, reproduced from Kwon's [1994] dissertation (his Figure 6.13, page 181) shows the train segment utilization he obtained. Kwon reported minimum and maximum, but not average load factors which makes direct comparison difficult. Figure B.2 shows our result. In Kwon's analysis, 17 train segments were fully utilized, at least on certain peak days. In our base case Dynamic Car Scheduling scenario, our closest analog to Kwon's model, 19 segments were fully utilized on at least some days, some at very high average load factors. This supports our assertion that this problem is comparable to, possibly slightly more difficult than Kwon's original problem.

Individual train load factors for Figure B.2 are reported in Table B.1. At this detailed level, additional bottlenecks become apparent. Inbound to location 6, train 722 has a 99% load factor even though the overall load factor for segment 7-6 is only 64.1%. Likewise, train 227 has a 97% load factor although overall loading for segment 3-6 is only 49.1%. Only trains 722 and 227 stop at location 6, the others bypass this location.

Figure B.3 shows the result of reducing all train segment capacities by 15%. The effect of this reduction is discussed in Chapter 5 of the dissertation. Train load factors related to Figure B.3 are reported in Table B.2.

Figure B.1 - Train Segment Utilization from Kwon [1994]

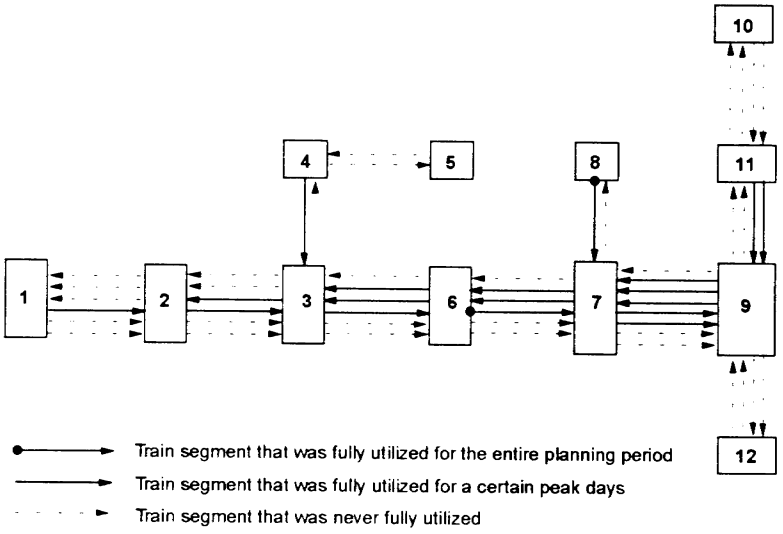


Figure B.2 - Train Capacity Utilization from Dynamic Car Scheduling Scenario #3 Base Case

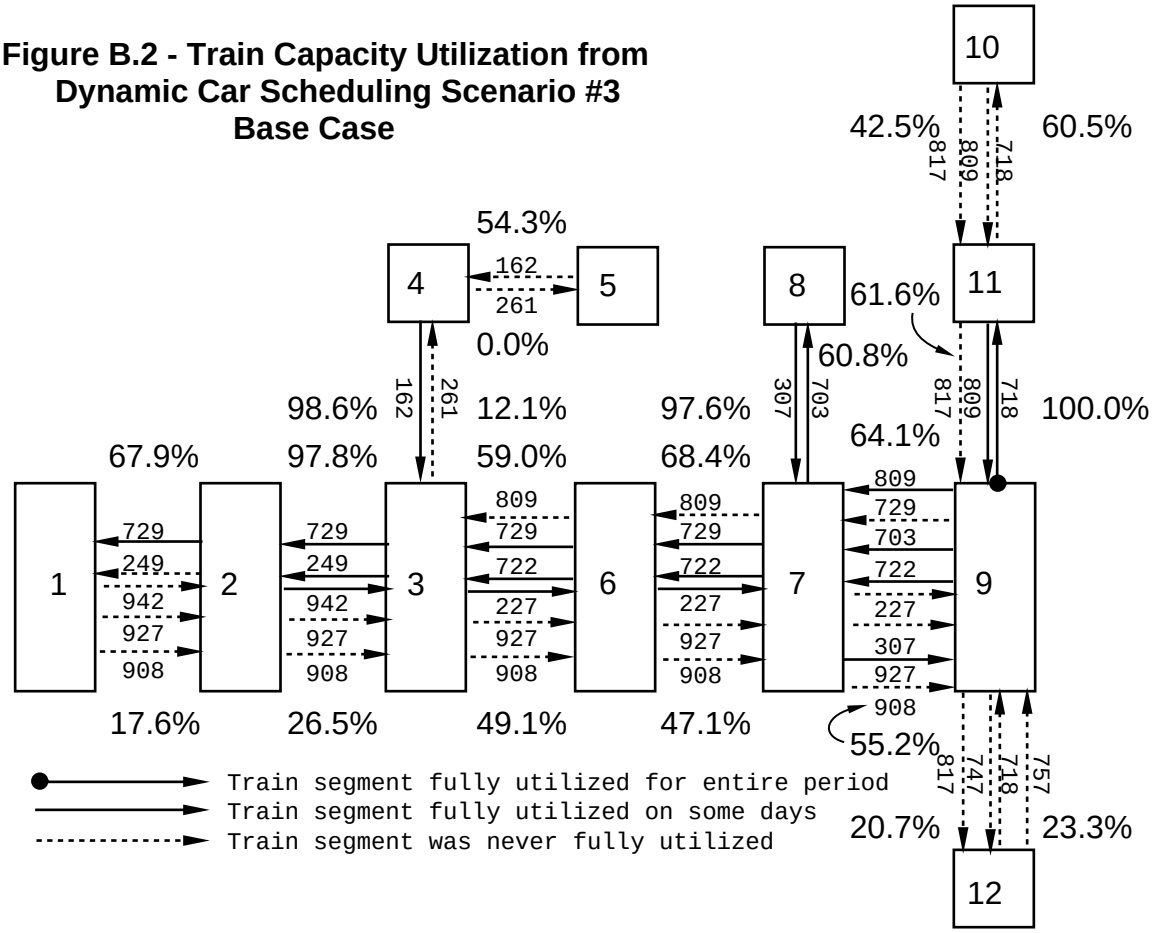


Figure B.3 - Train Capacity Utilization from Dynamic Car Scheduling Scenario #3 Base Capacity Reduced by 15%

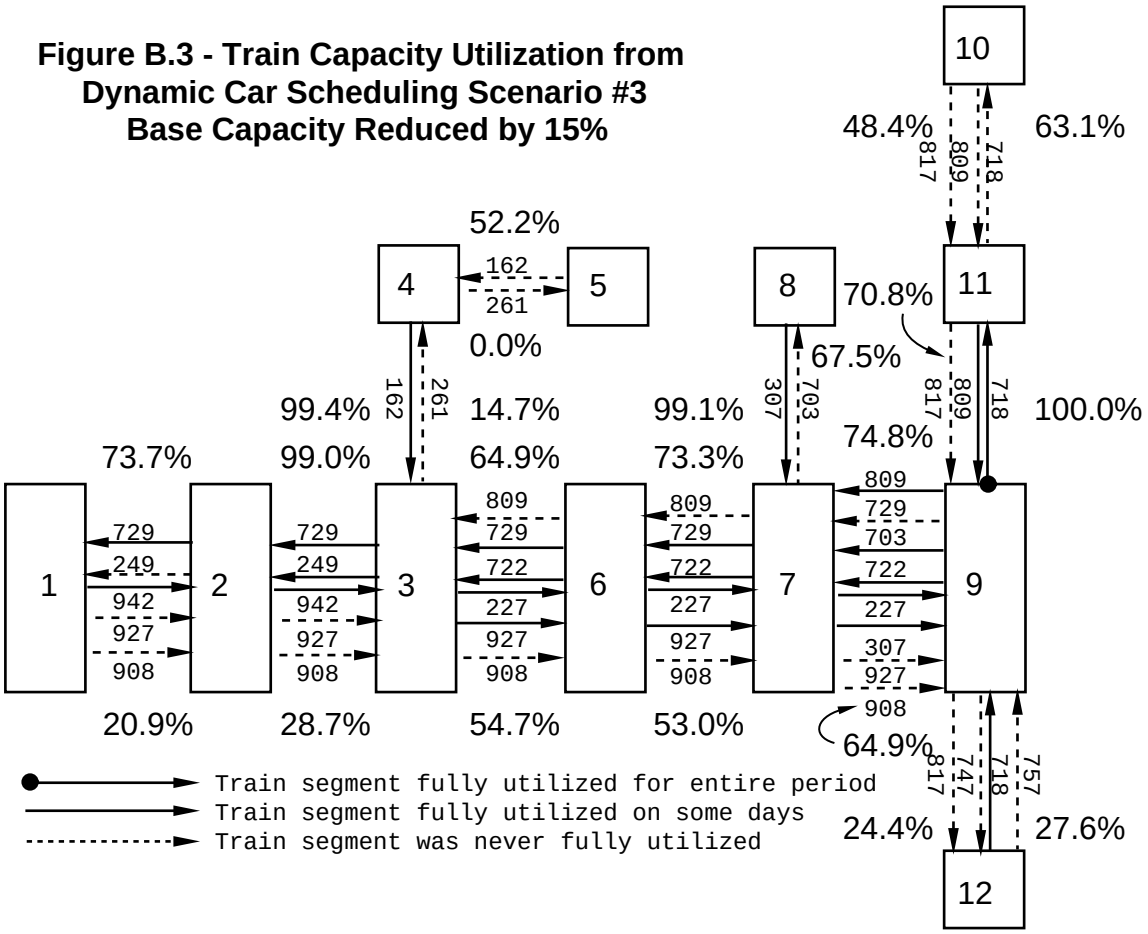


Table B.1 - Dynamic Car Scheduling - Base Case
Individual Train Load Factors

	A	B	C	D	E	F	G
1	Symbol	From	To	Max Capy	Act Cars	# Trns	# Full
2	R162	4	3	1207	1190	17	12
3	R162	5	4	1207	655	17	
4	R227	3	6	799	778	17	14
5	R227	6	7	799	688	17	10
6	R227	7	9	799	458	17	
7	R249	2	1	799	52	17	
8	R249	3	2	799	773	17	13
9	R261	3	4	1615	196	17	
10	R261	4	5	1615	0	17	
11	R307	7	9	1207	584	17	
12	R307	8	7	1207	1178	17	11
13	R703	7	8	1207	734	17	2
14	R703	9	7	1207	804	17	4
15	R718	9	11	1615	1615	17	17
16	R718	11	10	1615	977	17	
17	R718	12	9	1615	674	17	
18	R722	6	3	799	357	17	2
19	R722	7	6	799	773	17	9
20	R722	9	7	799	784	17	12
21	R729	3	1	1615	1588	17	12
22	R729	7	3	1615	1377	17	4
23	R729	9	7	1615	198	17	
24	R747	9	12	1615	0	17	
25	R757	12	9	1615	77	17	
26	R809	7	3	2023	883	17	
27	R809	9	7	2023	1832	17	3
28	R809	10	11	2023	1303	17	
29	R809	11	9	2023	1667	17	2
30	R817	9	12	1615	667	17	
31	R817	10	11	1615	244	17	
32	R817	11	9	1615	575	17	
33	R908	1	3	2023	192	17	
34	R908	3	7	2023	341	17	
35	R908	7	9	2023	528	17	
36	R927	1	3	1615	207	17	
37	R927	3	7	1615	1060	17	
38	R927	7	9	1615	1544	17	8
39	R942	1	2	799	381	17	
40	R942	2	3	799	776	17	11

Table B.2 - Dynamic Car Scheduling - Reduced Capacity Scenario
Individual Train Load Factors

	A	B	C	D	E	F	G
1	Symbol	From	To	Max Capy	Act Cars	# Trns	# Full
2	R162	4	3	1020	1014	17	15
3	R162	5	4	1020	532	17	
4	R227	3	6	663	654	17	16
5	R227	6	7	663	589	17	12
6	R227	7	9	663	470	17	4
7	R249	2	1	663	147	17	
8	R249	3	2	663	659	17	15
9	R261	3	4	1360	200	17	
10	R261	4	5	1360	0	17	
11	R307	7	9	1020	519	17	
12	R307	8	7	1020	1011	17	16
13	R703	7	8	1020	689	17	
14	R703	9	7	1020	937	17	11
15	R718	9	11	1360	1360	17	17
16	R718	11	10	1360	858	17	
17	R718	12	9	1360	714	17	1
18	R722	6	3	663	345	17	3
19	R722	7	6	663	660	17	14
20	R722	9	7	663	642	17	14
21	R729	3	1	1360	1343	17	12
22	R729	7	3	1360	1277	17	6
23	R729	9	7	1360	289	17	
24	R747	9	12	1360	0	17	
25	R757	12	9	1360	37	17	
26	R809	7	3	1717	806	17	
27	R809	9	7	1717	1693	17	13
28	R809	10	11	1717	1240	17	
29	R809	11	9	1717	1593	17	7
30	R817	9	12	1360	664	17	
31	R817	10	11	1360	249	17	
32	R817	11	9	1360	585	17	
33	R908	1	3	1717	240	17	
34	R908	3	7	1717	452	17	
35	R908	7	9	1717	797	17	
36	R927	1	3	1360	174	17	
37	R927	3	7	1360	941	17	1
38	R927	7	9	1360	1302	17	5
39	R942	1	2	663	368	17	1
40	R942	2	3	663	659	17	16

Appendix C

Rolling Horizon Simulation Test Results

The following tables present the results of rolling horizon simulation testing of Chapter 5. This Appendix includes all the output tables except for Table 5.2, which result was so significant that we chose to present it in-line with the text of Chapter 5.

	A	B	C	D	E	F
1	**Table C.1 - Transit Time Distribution by Hourly Cost for Scenario 1**					
2						
3			Days Later than Base ETA			
4	Cost/Hour	Row Summary	0	1	2	3
5	2	943	661	189	57	16
6	3	1271	840	289	86	19
7	4	956	663	139	92	23
8	5	1383	831	298	157	67
9	6	1032	640	148	158	27
10	7	1179	757	175	139	30
11	8	1054	797	160	47	23
12	9	979	608	195	81	26
13	10	1121	824	151	77	36
14	11	1153	744	208	106	41
15	Total Cars	11071	7365	1952	1000	308
16						
17						
18	Table C.1 - ctd		Days Later than Base ETA			
19	Cost/Hour	4	5	6	7	
20	2	2	2	16		
21	3	23	14			
22	4	28	1	10		
23	5	29	1			
24	6	28	10	1	20	
25	7	35	33	10		
26	8	4	19	1	3	
27	9	56	5	8		
28	10	11	20	2		
29	11	47	2	5		
30	Total Cars	263	107	53	23	

	A	B	C	D	E	F
1	**Table C.2 - Transit Time Distribution by Hourly Cost for Scenario 2**					
2						
3			Days Later than Base ETA			
4	Cost/Hour	Row Summary	0	1	2	3
5	2	1010	774	96	77	4
6	3	1275	886	234	89	22
7	4	1014	745	129	110	12
8	5	1405	856	219	214	68
9	6	1100	701	210	100	48
10	7	1280	854	202	111	58
11	8	1088	893	109	50	9
12	9	996	653	153	81	49
13	10	1207	839	179	112	46
14	11	1220	802	182	136	48
15	Total Cars	11595	8003	1713	1080	364
16						
17	Table C.2 -ctd		Days Later than Base ETA			
18	Cost/Hour	4	5	6	7	8
19	2	8	6			
20	3	11	14	2	1	
21	4	6	9			
22	5	33	15			
23	6	14	6	3	1	
24	7	22	9	4		
25	8	18	6	3		
26	9	49	6	5		
27	10	2	9	8		8
28	11	22	21	1		
29	Total Cars	185	101	26	2	8
30						
31	Table C.2 -ctd		Days Later than Base ETA			
32	Cost/Hour	9	10	11		
33	2	3	24	18		
34	3	16				
35	4	3				
36	5					
37	6	11	6			
38	7	5	15			
39	8					
40	9					
41	10	4				
42	11		8			
43	Total Cars	42	53	18		

	A	B	C	D	E	F	G	H
1	**Table C.3 - Transit Time Distribution by Hourly Cost for Scenario 3**							
2								
3			Days Later than Base ETA					
4	Cost/Hour	Row Summary	0	1	2	3	4	5
5	2	965	476	228	117	68	62	14
6	3	1391	670	421	129	58	113	
7	4	1107	683	219	164	33	8	
8	5	1595	1049	376	150	20		
9	6	1196	770	327	99			
10	7	1308	970	285	52	1		
11	8	1138	815	318	5			
12	9	1114	882	230	2			
13	10	1372	1252	120				
14	11	1416	1165	251				
15	Total Cars	12602	8732	2775	718	180	183	14

	A	B	C	D	E	F
1	**Table C.4 - Transit Time Distribution by Hourly Cost for Scenario 4**					
2						
3			Days Later than Base ETA			
4	Cost/Hour	Row Summary	0	1	2	3
5	2	1039	702	282	44	11
6	3	1413	977	358	70	8
7	4	1125	860	193	41	31
8	5	1607	1266	279	61	1
9	6	1231	926	289	15	1
10	7	1295	1088	182	24	1
11	8	1091	966	119	6	
12	9	1033	940	89	4	
13	10	1264	1158	98	6	2
14	11	1367	1186	180	1	
15		12465	10069	2069	272	55

Table C.5 - Transit Time Distribution by Penalty Cost for Scenario 4

				Days Later than Base ETA			
Penalty Cost	Row Summary	# Later	% Later	0	1	2	3
$0-$100	8756	1956	22.3	6800	1632	269	55
$100-$200	1907	278	14.6	1629	275	3	
$200-$300	854	88	10.3	766	88		
$300-$400	544	56	10.3	488	56		
$400-$500	205	18	8.8	187	18		
$500-600	122	0	0.0	122			
$600-700	55	0	0.0	55			
$700+	22	0	0.0	22			
Total Cars	12465	2396	19.2	10069	2069	272	55

Table C.6 - Transit Time Distribution by Hourly Cost for Scenario 3

		Days Later than Original Trip Plan ETA				
Cost/Hour	Row Summary	- 1	0	1	2	3
2	965	1	576	193	90	71
3	1391	6	922	274	105	68
4	1107	3	814	235	44	11
5	1595	17	1264	220	80	14
6	1196	12	965	200	19	
7	1308	4	1130	161	12	1
8	1138		1078	60		
9	1114	16	1085	12	1	
10	1372	1	1319	52		
11	1416	8	1347	61		
Total Cars	12602	68	10500	1468	351	165

Table C.6 - ctd		Days Later than Original Trip Plan ETA	
Cost/Hour		4	5
2		24	10
3		16	
4			
5			
6			
7			
8			
9			
10			
11			
Total Cars		40	10

Table C.7 - Slack Time Added by Penalty Cost for Scenario 4

		Days Slack Time Added vs Base ETA			
Penalty Cost	Row Summary	0	1	2	3
$0-$100	8756	7468	1071	156	61
$100-$200	1907	1706	201		
$200-$300	854	763	91		
$300-$400	544	534	10		
$400-$500	205	195	10		
$500-600	122	122			
$600-700	55	55			
$700+	22	22			
Total Cars	12465	10865	1383	156	61

	A	B	C	D	E	F	G
1	**Table C.8 - Transit Time Distribution by Hourly Cost**						
2	**with Penalty Costs on Shipments costing more than $10/hour**						
3							
4			Days Later than Original Trip Plan ETA				
5	Cost/Hour	Row Summary	- 1	0	1	2	3
6	2	965	10	569	203	90	59
7	3	1387	6	910	287	102	65
8	4	1104	3	834	200	64	3
9	5	1578	19	1253	226	68	12
10	6	1180	11	982	172	15	
11	7	1316	2	1168	144	2	
12	8	1135	10	1049	76		
13	9	1114		1087	27		
14	10	1384	2	1382			
15	11	1419	9	1410			
16	Total Cars	12582	72	10644	1335	341	139
17							
18							
19	Table C.8 ctd	Days Later than Original Trip Plan ETA					
20	Cost/Hour	4	5				
21	2	24	10				
22	3	17					
23	4						
24	5						
25	6						
26	7						
27	8						
28	9						
29	10						
30	11						
31	Total Cars	41	10				

	A	B	C	D	E	F	G
1	**Table C.9 - Transit Time Distribution by Logit Priority Coefficient**						
2							
3			Days Later than Original Trip Plan ETA				
4	Priority Coeff	Row Summary	- 1	0	1	2	3
5	1	3133	25	2767	213	67	32
6	2	3326	20	2758	443	71	34
7	3	3137	8	2539	393	130	65
8	4	3006	15	2436	419	83	34
9	Total Cars	12602	68	10500	1468	351	165
10							
11							
12	Table C.9 ctd	Days Later than Original Trip Plan ETA					
13	Priority Coeff	4	5				
14	1	19	10				
15	2						
16	3	2					
17	4	19					
18	Total Cars	40	10				

Table C.10 - Transit Time Distribution by Logit Priority Coefficient with Penalty Costs on Shipments having Priority Coefficient < 2

		Days Later than Original Trip Plan ETA				
Priority Coeff	Row Summary	- 1	0	1	2	3
1	3222	25	3197			
2	3299	16	2669	501	75	31
3	3090	20	2522	338	138	65
4	2995	21	2343	452	97	67
Total Cars	12606	82	10731	1291	310	163

Table C.10 ctd	Days Later than Original Trip Plan ETA
Priority Coeff	4
1	
2	7
3	7
4	15
	29