© Elsevier Science Publishers B.V., 1988

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owner.

ISBN: 0 444 70407 8

Published by:
Elsevier Science Publishers B.V.
P.O. Box 1991
1000 BZ Amsterdam
The Netherlands

Sole distributors for the U.S.A. and Canada:
Elsevier Science Publishing Company, Inc.
52 Vanderbilt Avenue
New York, N.Y. 10017
U.S.A.

PRINTED IN THE NETHERLANDS

PREFACE

The vehicle routing problem (VRP), like its well-known traveling salesman cousin, is a fascinating problem. It is easy to describe, but difficult to solve. One might say that such problems belong to the class of easy-NP-hard!

A further fascinating feature of vehicle routing is that the basic problem can be extended into an untold number of variations that are not just mathematical diversions; they do occur in the real-world problem-solving arena. Thus, it is no wonder that researchers from the fields of operations research, mathematics, transportation, and computer science, such as those who have contributed papers to this volume, have found this problem most challenging.

Although difficult to solve in an optimizing sense, VRPs are solved in an operational sense. The world's economies could not operate except for the fact that VRPs and their extensions have readily available "practical" solutions. Things are delivered and picked up, and customer demands are more or less satisfied on time without too much pain. But competition and the desire to improve profits call for better solutions. To their credit, VRP researchers have not been lured to the rocks by the siren of optimality. They recognized early that important improvements in vehicle routing could be made by non-optimal, directed investigations into the mathematical and computational structures that describe VRPs. Such efforts have helped to form the theoretical field of heuristics, as well as aiding in the development of heuristic solution procedures. The resulting heuristic algorithms go way beyond the replication of how the experts run their operational systems; these algorithms seek and find improved solutions that can be implemented.

The papers in this volume not only summarize the past developments in VRP technology, but also describe many new advances. One can only be impressed by the abilities of the authors as they attack and resolve a diverse set of important problems. Researchers and practitioners will find much here that is new and rewarding.

A closing thought. The original and classical VRP can be traced back in time many hundreds of years. It is a problem that arises every year at about this time; that it gets solved each time has always, I am sure, amazed us all. Our wonderment in how the solution is obtained is an unconscious force (in a psychological sense), stemming from our childhood, that motivates our search for algorithms to solve the VRP. The problem: How does Santa Claus do it? Santa has a single vehicle with finite capacity that leaves from a single depot; millions of

Lin, S., and B.W. Kernighan, 1973. An Effective Heuristic Algorithm for the Traveling Salesman Problem. Operations Research 21, 498-516.

Minkoff, A.S., 1985. Real-Time Dispatching of Delivery Vehicles. PhD Thesis, Operations Research Center, MIT.

Or, I., 1976. Traveling-Salesman-type Combinatorial Problems and Their Relation to the Logistics of Blood Banking. PhD Thesis, Dept. of Industrial Engineering and Management Sciences, Northwestern University.

Powell, W., 1985. An Operational Planning Model for the Dynamic Vehicle Allocation Problem with Uncertain Demands. Department of Civil Engineering Working Paper EES-85-15, Princeton University.

Psaraftis, H.N., 1980. A Dynamic Programming Solution to the Single-Vehicle Man-to-Many Immediate Request Dial-A-Ride Problem. Transportation Science 14, 130-154.

Psaraftis, H.N. 1983. k-Interchange Procedures for Local Search in a Precedence-Constrained Routing Problem. European Journal of Operational Research 13, 391-402.

Psaraftis, H.N., J.B. Orlin, D. Bienstock, and P.M. Thompson, 1985. Analysis and Solution Algorithms of Sealift Routing and Scheduling Problems: Final Report. Working Paper No. 1700-85, Sloan School of Management, MIT.

Savelsbergh, M.W.P., 1985. Local Search in Routing Problems with Time Windows. Annals of Operations Research 4, 285-305.

Tharakan, G.G., and H.N. Psaraftis, 1981. An Exact Algorithm for the Exponential Disutility Dial-A-Ride Problem. Working Paper, MIT.

Wilson, N.H.M., J.M. Sussman, H.K. Wang, and B.T. Higonett, 1971. Scheduling Algorithms for Dial-A-Ride systems. Urban Systems Laboratory Report USL TR-70-13, MIT.

Wilson, N.H.M., and H. Weissberg, 1976. Advanced Dial-A-Ride Algorithms Research Project: Final Report. Report R76-20, Dept. of Civil Engineering, MIT.

Wilson, N.H.M., and N.H. Colvin, 1977. Computer Control of the Rochester Dial-A-Ride System. Report R77-31, Dept. of Civil Engineering, MIT.

Vehicle Routing: Methods and Studies
B.L. Golden and A.A. Assad (Editors)
© Elsevier Science Publishers B.V. (North-Holland), 1988

A COMPARATIVE REVIEW OF ALTERNATIVE ALGORITHMS FOR THE DYNAMIC VEHICLE ALLOCATION PROBLEM †

Warren B. POWELL

Princeton University, Department of Civil Engineering and Operations Research, Princeton, New Jersey 08544

The dynamic vehicle allocation problem involves managing a generally large fleet of vehicles over time to maximize total profits. The problem is reviewed in the context of truckload trucking with special attention given to dispatching and repositioning trucks in anticipation of forecasted future demands. Four different methodological approaches are reviewed: deterministic transshipment networks, stochastic/nonlinear networks, Markov decision processes and stochastic programming. The methods are contrasted in terms of their formulation of the objective function and decision variables, the degree to which actual practices can be represented, and computational requirements. The paper provides an example of how a particular problem can be approached from significantly different perspectives.

1. INTRODUCTION

The dynamic vehicle allocation problem arises in industries where a fleet of vehicles must be managed over time responding to known or forecasted demands for capacity. Motor carriers, railroads, container shipping lines, and auto or truck rental companies are immediate examples of this problem. Different industries, however, exhibit unique characteristics which lend themselves to different modelling approximations. For this reason, the discussion here uses truckload trucking as the industry context, although the basic concepts and algorithms are more general. Truckload trucking is also one of the simplest of these modes and as a result poses the least overhead in terms of industry minutiae.

Briefly, and somewhat simplistically, the problem faced by truckload motor carriers can be described as follows. A shipper will call a carrier with a load going from city A to city B. The carrier must deadhead a truck to the shipper where the trailer is loaded and then run to city B where the delivery is made. The carrier must then decide what to do with the truck once it arrives in B. At any point in time, a truck must either be assigned to a load, repositioned empty to a city in anticipation of loads to be called in later, or simply held at its current location. It is important to realize that there is no consolidation function, as typically arises in vehicle routing problems. This property significantly simplifies the problem and focuses attention on the fleet allocation side of the problem.

As with auto rental companies, truckload trucking is characterized by a high level of competition in the marketplace compounded by a high degree of uncertainty regarding future demands. Typically, 60 percent of the loads called in are for pickup the same day, implying

† This research was supported by the National Science Foundation under a Presidential Young Investigator Award under Grant ECE-8408044.

that at the beginning of the day the carrier will know only 40 percent of the loads that will be carried that day. At the same time, if a carrier does not have a truck close by when a load is called in, that load will generally be given to a competitor. Since the largest company captures only 1 percent of the market, shippers regularly work with several carriers. For this reason, a carrier also has some freedom to turn down freight and as a result must learn to be selective about the freight it carries. This process, known as load acceptance, is an important part of the fleet management problem.

The fleet dispatching process at most truckload carriers works as follows. At any given point in time, there will be a set of unassigned drivers and unassigned loads. Dispatchers will typically have access to a computer terminal that might show, for example, all the loads within 80 miles of a given driver. The dispatcher must then assign the driver to one of these loads, taking care to match equipment types. This decision is often made while the driver is waiting on the telephone, and it is not unusual for the driver to refuse to pull a load, preferring another load moving in a desired direction (the driver may be trying to get home). At other times, the dispatcher may simply ask the driver to call back after 30 minutes if no load is available. If it is unlikely that a load will become available in his area, the dispatcher may tell the driver to move empty to a better location and call back when he arrives.

The process of assigning a driver to a load generally focuses on minimizing empty miles, which is often interpreted to mean that a driver should be assigned to the nearest load for which the driver has the correct equipment. Only one carrier, to this author's knowledge, actually solves a transportation problem to assign drivers to loads optimally. Separate from the dispatching process is the planning function which determines how to reposition empty trucks. This is a largely heuristic process where experienced planners will move trucks from traditional surplus regions to traditional deficit regions. The difficulty is in forecasting the number of loads yet to be booked out of a region which must be balanced against the profitability of loads out of each region. For the most part, dispatchers and planners respond to activities like trucks and loads, and have considerably more difficulty comparing the profitability of different actions.

The last important step in fleet management is load acceptance and load solicitation. Unlike classical vehicle routing problems, where the objective is to carry all the freight at least cost, the highly competitive truckload market allows carriers to refuse freight occasionally which does not appear profitable. In addition, the sales force may attempt to solicit freight to help fill empty backhaul movements. Almost all carriers approach load acceptance and load solicitation by trying to attract all the freight they can and accepting everything they have the capacity to handle. However, most carriers have a good sense that a load paying $1.30 per mile in one direction may be good while another load paying $0.90 in a backhaul lane may be even better. Just the same, under the philosophy that a bird in the hand is worth two in the bush, the usual rule is to accept loads until they run out of capacity. Load solicitation, on the other hand, is a process that varies widely among carriers. It is likely that most carriers evaluate their sales force in terms of total sales (or, in some cases, total loads) which gives little incentive to attract freight in profitable lanes either on a long term or a short term basis. One carrier solves a network transshipment model each night. The network model heuristically turns forecasts of freight into fictitious loads (this process is described below) which might be accepted or rejected. The fictitious loads which are "accepted" by the model become recommendations of places where the sales force should solicit freight.

Fortunately from a planning perspective, most carriers actively collect and maintain all the data that is necessary to feed an optimization model. When a shipper calls in a load, the carrier immediately keys in the exact origin and destination of the load, its pickup time and date, its delivery time, the equipment type required and other special handling costs. Generally, the carrier already has in the computer a previously agreed on tariff for the load (which may vary between shippers and between traffic lanes). At the same time, each driver, after dropping off a load, calls in giving his exact location and status. This way, it is possible to calculate accurately the deadhead miles from each driver to each load.

Separate from real-time data on each driver and each load, it is relatively straightforward to develop detailed forecasts of future activities from extensive records of past activities. Generally each loaded and empty move is recorded on the computer. This is necessary since all drivers are paid by the mile, and there are usually different rates paid for loaded and empty moves. Thus, records of activities must be kept in order to pay the driver.

The dynamic vehicle allocation problem (DVA) has been studied by a number of authors. An excellent review of this work is given by Dejax and Crainic [9]. The earliest efforts in this problem were applied to the repositioning of empty rail freight cars (Misra, [18], Ouimet, [19] and Baker, [1]). Misra used a simple static transportation formulation, even though White and Bomberault [30] had already formulated the dynamic vehicle allocation problem in the now classical time-space framework. Uncertainty in demand forecasts has been incorporated by posing the problem as a spatially separated inventory problem (Philip and Sussman, [20]). Other contributions are reviewed during the presentation.

The primary goal of this research is to review and contrast alternative modelling and solution approaches with particular attention given to the handling of forecasting uncertainties. This review documents a transition from a purely network based approach to stochastic optimization techniques using Markov decision processes and other techniques. The presentation is organized as follows. Section 2 describes deterministic formulations, covering both single stage and dynamic models. The deterministic assumption applies to the modelling of forecasting uncertainties surrounding loads that will have to be carried in the future. Section 3 describes two approaches for extending the basic network formulation to handle forecasting uncertainties, including one formulation where flows on links are handled explicitly as random variables. Section 4 formulates the same problem as a Markov decision process with a very large state space, serving primarily as an approach for developing a better understanding of the problem structure. Section 5 combines insights from Markov decision processes and the classical network formulations to form a hybrid model that reduces to a linear network. Finally, Section 6 provides an overview of alternative models and discusses implementation issues.

2. DETERMINISTIC MODELS

Deterministic models have played an important historical role in the dynamic vehicle allocation problem. At the same time, while highly simplistic in the assumptions these models impose, their inherent simplicity suggests that they will continue to serve as valuable tools in practice. For the purposes of the discussion here deterministic models are divided between single stage models, which simply assign trucks to loads with little or no forecasting, and dynamic models which explicitly track trucks and loads over a given planning horizon.

It is common when modelling flows over a continuum as large as the United States to divide the country into 60 to 100 discrete regions. Let $\mathbf{R} = \{1, ..., R\}$ denote the set of regions. Then define:

$x_{i,j}(t)$ = flow of trucks moving loaded from region i to region j, departing from i in period t,

$y_{i,j}(t)$ = flow of trucks moving empty from region i to region j,

$t_{i,j}$ = travel time in integer time units to travel from i to j (for simplicity, travel times for moving loaded and empty are assumed to be equal),

$r_{i,j}$ = average contribution (revenue minus direct operating cost) for pulling a load from i to j,

$c_{i,j}$ = cost of moving empty from i to j,

$F_{i,j}(t)$ = random variable denoting the number of loads that will be called in from i to j to be picked up at time t,

$f_{ij}(t)$ = $E[F_{ij}(t)]$,

= expected number of forecasted loads from i to j departing at time t,

$L_{i,j}(t)$ = actual number of loads known at time t=0 to be available moving from i to j at time t,

$T_i(t)$ = number of drivers becoming available for the first time in region i at time t,

P = length of the planning horizon.

Using this notation it is possible to describe single stage and dynamic deterministic network models.

2.1 Single stage deterministic models

The simplest single stage model is a transportation problem assigning available drivers to available loads, as depicted in Figure 1. In this figure,

$D_i(t)$ = total outbound loads from region i at time t,

$$= \sum_{j \in \mathbf{R}} L_{ij}(t) ,$$

E_r = extra trucks needed to satisfy demand,

$$= \max \{ \sum_{i \in \mathbf{R}} [D_i(1) - T_i(1)] , 0 \} ,$$

E_D = extra demand needed to absorb excess trucks,

$$= \max \{ \sum_{i \in \mathbf{R}} [T_i(1) - D_i(1)] , 0 \} .$$

This model does nothing more than assign available trucks to available loads and is unable to plan for the future. Any trucks that are assigned to the dummy demand node are simply held in the region. It is possible, of course, to simply augment the demands by adding the number of known loads with forecasted loads, but such a model runs the risk of refusing a known load in one region over a forecasted load in another region. This problem can be alleviated by providing very simple forecasting capabilities. Let

$p_i(t)$ = the average return of a truck in region i on day t until the end of the planning horizon.

The factors $\{p_i(t)\}$ are termed *salvage values*, and a simple approach for calculating them based on historical data is given in the appendix.

Using these salvage values, Figure 2 presents an alternative single stage dispatch model. Here, the demand nodes for known loads have been augmented by separate nodes for forecasted loads. A truck repositioned empty from i to j will arrive on day t_{ij}, at which point it will receive expected profits of $p_j(t_{ij})$. It is necessary to bound the number of vehicles repositioned to region j arriving on day t_{ij} . Let

$$f_{i\cdot}(t) = \sum_{j \in R} f_{ij}(t) \tag{1}$$

be the total forecasted demand out of i. Since $f_{i\cdot}(t)$ will usually be noninteger, we let $\hat{f}_{i\cdot}(t)$ be the upper bound and set

$$\hat{f}_{i\cdot}(t) = \left\lfloor f_{i\cdot}(t) + \gamma \right\rfloor \tag{2}$$

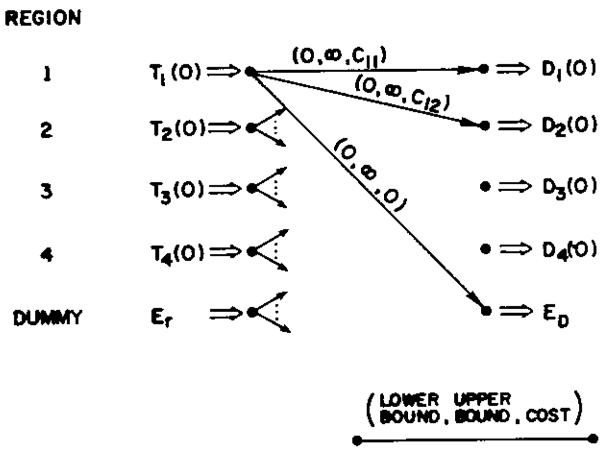

Figure 1
Single stage transportation formulation with overflow nodes

where $\lfloor x \rfloor$ is the largest integer less than or equal to x and $0 \le \gamma < 1$ is a specified parameter. $\gamma = .5$ produces conventional rounding while $\gamma = .8$ tends to round more forecasts up.

The real value of simple assignment models is their ability to handle high levels of detail. Consider the network in Figure 1 where each supply node is a single driver and each demand node is a single load. In this case a link (i,j) represents an assignment of driver i to load j. The cost coefficient c_{ij} would then represent the empty cost from the precise location of driver i to the precise location of load j, thus avoiding the need to aggregate the country into regions. In addition, if truck i represents an equipment type that is incompatible with load j, this link would not be generated. The biggest advantage of this model, then, is its ability to incorporate a high level of detail about a driver's location and characteristics. This more detailed formulation can also be augmented by the forecast nodes given in Figure 2, allowing the model to recommend empty repositioning moves. The only, and major, weakness of the model is that its relatively simplistic representation of the downstream effects limits its ability to reposition empties between regions accurately. Despite this weakness, however, it is quite possible that the ability of the model to choose the best driver for a load may measurably outperform a dispatcher's performance in the same task, overcoming the model's other weaknesses.

Another feature of the transportation formulation is its small size and the speed with which it can be solved. Assume every driver and load were represented explicitly. Typical problem sizes range from 100 to 1000 drivers being assigned to a comparable number of loads. All

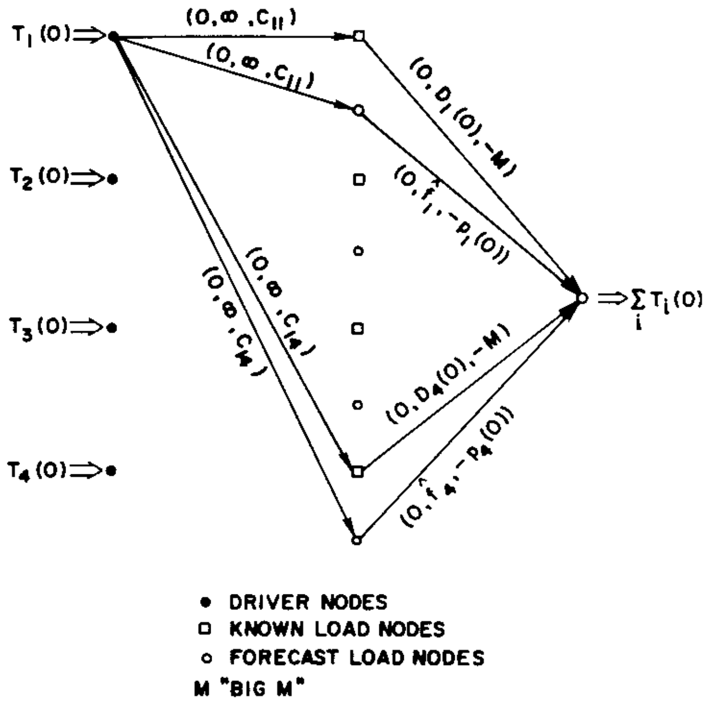

Figure 2
Single stage transportation formulation with forecast load nodes

these models use pruning rules which limit assignments of drivers to the nearest 5 to 10 loads, giving network sizes with 500 to 10,000 links. On a large mainframe, such problems can be optimized from scratch in a few seconds. Furthermore, these codes can be reoptimized from a previous basis following a change in the data usually in one tenth the time.

2.2 Dynamic deterministic models

The biggest limitation of static models is the simple way in which future activities are forecasted. Ideally, the simple linear salvage values used to summarize the value of an additional truck in each region should be replaced with a nonlinear, nonseparable function of the vector of supplies in each region. A straightforward solution to this problem, which is now considered the classical approach to the dynamic vehicle allocation problem, is to form a time-space diagram where each node represents a region at a particular point in time. Between any pair of nodes are three types of links: known loads, where the cost coefficient is minus the direct contribution for that load; empty movements, with a cost equal to the region to region empty repositioning costs; and forecasted loads, with a cost coefficient equal to minus the historical average direct contribution. Typically, loaded movements between the same regions on the same day are modelled as separate links which then would each have an upper bound of one. This allows known loads to be modelled with more detailed information on costs, revenues and travel times. Empty movement links of course have no upper bound.

The data requirements for a dynamic model are relatively straightforward. First is the real-time information on loads and drivers. It is necessary to know when and where each driver will first become available (it is not possible to take equipment types into consideration without sacrificing the pure network structure). Next we need to know the pickup and delivery place and time, and the contribution (revenue minus direct operating cost), of each pending load in the system. Empty movements can be generated by calculating the average empty distance among empty moves actually made between a given pair of regions. It is also possible to calculate distances using the coordinates of zip codes within a region. Normally the list of possible empty moves is restricted to moves within a given radius.

The last and most difficult input to any dynamic model are the forecasts of future loads, represented by the $f_{ij}(t)$. It is beyond the scope of this discussion to discuss this step in detail, but the essence is that the $f_{ij}(t)$ are derived from time series forecasting models. These models might be built from six months or more of past loads, taking into account seasonal and other trends as well as recent activities.

Two important issues arise in the use of deterministic, dynamic network models. First, a method must be developed for representing forecasted loads as links in the network. Second, it is necessary to choose the length of the planning horizon and to develop a procedure for truncating the network. A difficulty arises again with setting the upper bounds on the forecasted loads. Ninety percent of the demand forecasts will normally fall between 0 and 1, with many below .5. Three approaches may be used:

(1) "Integerize" the upper bounds using heuristic rounding rules. The upper bound might be given by $\hat{f}_{ij}(t) = \lfloor f_{ij}(t) + \gamma \rfloor$ as we did in Equation 2. γ should be chosen so that $\sum_{j \in R} \hat{f}_{ij}(t) \approx \sum_{j \in R} f_{ij}(t)$.

(2) Randomly sample forecasted loads by using the mean, $f_{ij}(t)$, to fit a distribution (such as the Poisson) and then sample from this distribution. Each randomly sampled load would then be represented as a link with an upper bound of one.

(3) Use fractional upper bounds. Typically this is handled by changing the units from trucks to tenths or hundredths of trucks.

Since the last approach produces fractional solutions, it will be necessary to "integerize" the final solution. The second approach suffers from the randomness introduced by the sampling process. The first approach suffers from biases introduced by the rounding process, resulting in significantly higher or lower forecasts.

Separate from the issue of dealing with fractional demand forecasts is the serious problem of truncating the planning horizon. Researchers actively applying this approach to problems in rail and trucking have, in private conversations, reported serious distortions when the truncation is not handled properly even when reasonably long planning horizons are used. At some point we are forced to ignore the future impact of a decision, and this often has an amazing ability to ripple back to the beginning of the planning horizon. The simplest example is a network with regions where trucks generally cannot move out loaded (such as Montana). Normally the price to carry a load into such a region is quite high to cover the cost of the empty backhaul. Near the end of the planning horizon, the network model will tend to push trucks into Montana to get the high price since the cost of moving out empty is being ignored.

The deterministic, dynamic network model is simply a special example of an infinite stage linear program. For notational simplicity assume that each time period t, $t = 0, 1, 2, \ldots$, is one day and that all travel times between regions are exactly one day (travel times other than one day are easy to handle but complicate the presentation without contributing any insights). In addition, we will aggregate over a single link all loaded flows from region i on day t to region j, where in practice we would generate different links for each known load and all forecasted loads. To discuss the issues associated with truncation define the following:

$U_{ij}(t)$ = total number of loads expected to be available from i to j leaving on day t,

$= L_{ij}(t) + f_{ij}(t)$,

$S_i(t)$ = supply of trucks at i on day t,

$= \sum_{k \in R} [x_{ki}(t-1) + y_{ki}(t-1)] + T_i(t)$,

$\mathbf{S}(t)$ = $\{S_1(t), \ldots, S_R(t)\}$,

α = a "discount" factor where α^t is the value of a dollar today spent in period t.

Taking advantage of the assumption of one day travel times, we have that $\sum_{i \in R} S_i(t) = \sum_{i \in R} T_i(0) =$ the fleet size, and $T_i(t) = 0$ for $t \geq 1$. This implies that the state of the system at time t is given completely by the vector $\mathbf{S}(t)$. The problem is one of maximizing total profits over time, where we adopt the approach of maximizing total discounted profits with a daily "discount" factor α. The problem then is to solve the following optimization problem:

$$\max_{\{x,y\}} \sum_{t=0}^{\infty} \sum_{i \in R} \sum_{j \in R} [r_{ij} x_{ij}(t) - c_{ij} y_{ij}(t)] \alpha^t, \tag{3}$$

subject to:

$$\sum_{j\in\mathcal{R}} [\, x_{ij}(0) + y_{ij}(0)\,] = T_i(0) \qquad \forall\, i \tag{3a}$$

$$\sum_{j\in\mathcal{R}} [\, x_{ji}(t-1) + y_{ji}(t-1)\,] = \sum_{k\in\mathcal{R}} [\, x_{ik}(t) + y_{ik}(t)\,] \quad t = 1, 2, ..., \qquad \forall\, i \tag{3b}$$

$$x_{ij}(t) \le U_{ij}(t) \qquad \forall\, i, j, t, \tag{3c}$$

$$x_{ij}(t), y_{ij}(t) \ge 0 \qquad \forall\, i, j, t\,. \tag{3d}$$

In practice we have to maximize profits over a given planning horizon. Equation 3 can always be rewritten using standard dynamic programming concepts as follows:

$$\max_{\{x,y\}} \sum_{t=0}^{P} \sum_{i\in\mathcal{R}} \sum_{j\in\mathcal{R}} [\, r_{ij}\, x_{ij}(t) - c_{ij}\, y_{ij}(t)\,]\alpha^t + \Psi_{P+1}(S(P+1)) \tag{4}$$

where

$$\Psi_{P+1}(S(P+1)) = \max_{\{x,y\}} \sum_{t=P+1}^{\infty} \sum_{i\in\mathcal{R}} \sum_{j\in\mathcal{R}} [\, r_{ij}\, x_{ij}(t) - c_{ij}\, y_{ij}(t)\,]\, \alpha^t \tag{5}$$

subject to:

$$\sum_{j\in\mathcal{R}} [\, x_{ij}(P+1) + y_{ij}(P+1)\,] = S_i(P+1) \tag{6a}$$

$$\sum_{j\in\mathcal{R}} [\, x_{ji}(t-1) + y_{ji}(t-1)\,] = \sum_{k\in\mathcal{R}} [\, x_{ik}(t) + y_{ik}(t)\,] \quad t = 1, 2, ..., \qquad \forall\, i \tag{6b}$$

$$x_{ij}(t) \le U_{ij}(t) \qquad \forall\, i, j, t, \tag{6c}$$

$$x_{ij}(t), y_{ij}(t) \ge 0 \qquad \forall\, i, j, t\,. \tag{6d}$$

The problem with (4) is that the function $\Psi_{P+1}(S(P+1))$ is a very complex, nonlinear, nonseparable function ($\Psi_{P+1}(\cdot)$ must also incorporate the effects of flow conservation constraints and upper bounds). The challenge is to replace $\Psi_{P+1}(S(P+1))$ with an easier to estimate function $\hat{\Psi}_{P+1}(S(P+1))$. This problem has been studied in a network context by Hughes and Powell [15] based on more general work by Grinold [10, 11, 12] and Grinold and Hopkins [13]. Three methods are reviewed briefly here, the first two resulting in simple pure network formulations

and the last one resulting in a generalized network.

The naive approach

This approximation consists simply of

$$\hat{\Psi}_{P+1}\ (S(P+1)) = 0\,. \tag{7}$$

End effects are just ignored, generally producing serious distortions in the optimal flow pattern.

The salvage method

This method puts a price (the salvage value) on a truck left in a region at the end of the planning horizon. As before, let $p_i(t)$ be the value of an additional vehicle in region i on day t, whose calculation is described in the appendix. Then

$$\hat{\Psi}_{P+1}\ (S(P+1)) = \sum_{i\in\mathcal{R}} S_i(P+1)\, p_i(P+1)\,. \tag{8}$$

This linear approximation lends itself easily to a pure network formulation. Both the naive and salvage approximations involve links from each node in time period $P+1$ to a supersink. The naive approach puts a cost of zero while the salvage method puts a cost of $p_i(P+1)$ from the node for region i. The salvage values $p_i(P+1)$ capture the expected contribution from period $P+1$ out to the end of a second planning horizon P_S. Trucks are assumed to follow historical trajectories in the periods $P+1$, $P+2$, ..., P_S (this is explained more thoroughly in the appendix).

A way to improve the accuracy of $\hat{\Psi}_{P+1}(\cdot)$ in both the naive and salvage methods is to add upper bounds on the flows that terminate in each region. That is

$$S_i(P+1) \le \hat{u}_i(P+1) \tag{9}$$

This constraint helps to mitigate somewhat the nonlinear properties of $\hat{\Psi}_{P+1}(\cdot)$. A common choice of $\hat{u}_i$ is:

$$\hat{u}_i\,(P+1) = \left\lfloor \sum_{j\in\mathcal{R}} f_{ij}\,(P+1) + \gamma \right\rfloor, \tag{10}$$

where again γ is our rounding factor.

Deterministic transshipment networks using the salvage method to mitigate end effects is the most widely used approach for solving dynamic vehicle allocation problems (see, for example, Chih [7] and Shan [25]). At this time, relatively little attention has been given to developing formal methods for calculating salvage values or evaluating their performance. The attraction is that the problems can be efficiently solved using standard network codes. A standard problem will use 60-80 regions with a planning horizon of 10 days, resulting in a network with 20,000 to 60,000 links which can be optimized in less than a minute on a large mainframe.

The dual equilibrium method

Assume, again for simplicity, that while there may be information about loads available for pickup today, nothing is known about tomorrow and, in addition, the forecasts of loads available for days $t = 1, 2, \ldots$, are the same. Thus we may let:

$$U(0) \;=\; \text{vector of upper bounds for today}$$

$$U(1) \;=\; \text{vector of upper bounds for tomorrow}$$

$$= \; U(t), \;\; t \geq 2\,.$$

In Grinold's terminology, day 0 is the transient stage and day 1 is the stationary stage which repeats itself indefinitely. Both the transient and stationary stages may in principle consist of one or more days. The transient stage might reasonably consist of the first 14 days while the first stationary stage might start on day 15 and consist of one week which repeats itself indefinitely.

The dual equilibrium method approximates the flows in the stationary stages as all being equal (that is, in a kind of stationary equilibrium). Using this notion, we may aggregate the flows starting in time period 1 using:

$$\hat{x}_{ij}(\alpha) \;=\; (1-\alpha)\sum_{t=1}^{\infty}\alpha^{t-1}\, x_{ij}(t)\,. \tag{11}$$

If in fact $x_{ij}(1) = x_{ij}(2) = \ldots = x_{ij}(t)$, then $\hat{x}_{ij}(\alpha) = x_{ij}(1)$. Next we aggregate the other network constraints (3b), (3c) and (3d) by multiplying both sides by $(1-\alpha)\alpha^{t-1}$ and summing over all t, giving:

$$\sum_{j\in R} [\,(1-\alpha)\,x_{ij}(0) + \alpha\,\hat{x}_{ij}(\alpha) + (1-\alpha)\,y_{ij}(0) + \alpha\,\hat{y}_{ij}(\alpha)\,] \;=\; \sum_{k\in R} [\,\hat{x}_{ki}(\alpha) + \hat{y}_{ik}(\alpha)\,] \tag{12}$$

$$\hat{x}_{ij}(\alpha) \;\leq\; u_{ij} \tag{13}$$

$$\hat{x}_{ij}(\alpha)\,,\hat{y}_{ij}(\alpha) \;\geq\; 0 \tag{14}$$

where $u_{ij} = u_{ij}(1) = u_{ij}(2) = \ldots = u_{ij}(t)$.

Finally, the objective function becomes, after a few manipulations:

$$\min_{\{x,y\}} \sum_{i\in R}\sum_{j\in R} [\, r_{ij}\,(x_{ij}(0) + \frac{\alpha}{1-\alpha}\,\hat{x}_{ij}(\alpha)) \;-\; c_{ij}(y_{ij}(0) + \frac{\alpha}{1-\alpha}\,\hat{y}_{ij}(\alpha))\,] \tag{15}$$

subject to (12), (13) and (14).

The key to this approach is the realization that Equation 12 represents the constraints for a generalized network. The graph for a two region network is depicted in Figure 3. Flows from the transient stage are first factored down by $1-\alpha$ when entering the stationary stage. Since each stage consists of only one time period, the links in the stationary stage loop back on themselves, factored down by α to represent the discounting from one stage to the next. In addition, the cost coefficients in the transient stage are factored by $\alpha/(1-\alpha)$.

The intuition behind Grinold's dual equilibrium approach is a little difficult to follow in this problem context. Hughes and Powell present the *generalized summation* method which is virtually equivalent to the dual equilibrium method. This method aggregates the flows using $\hat{x}(\alpha) = \sum_{t=1}^{\infty}\alpha^t\, x(t)$. In it, all flows making a transition from one period to the next are discounted by α. The arc coefficients are unaffected. The flows on the stationary arcs can now be interpreted as the total discounted *flows* and hence the upper bounds must be factored up by $\alpha/(1-\alpha)$. The resulting graph, shown in Figure 4, is much easier to understand intuitively since the arcs moving from one stage to the next always carry the discount factor.

Hughes and Powell report on a set of experiments on randomly generated test networks using discount factors α of .3 and .6. A "brute force" approach was used where a large network was generated covering N stages where N was chosen so that $\alpha^N \leq .05$ (resulting in networks with $N = 3$ and 8 stages, respectively) with 2 or 7 time periods within each stage. The solution to this brute force approach was then compared on the basis of optimal flows in the first time period alone as well as the flows throughout the transient stage. The results demonstrated that the dual equilibrium and generalized summation methods significantly outperformed the naive and salvage value methods.

What is most important about this line of research are the following observations:

(i) The assumption of a deterministic future actually complicates the problem if end effects are handled carefully, and

(ii)forecasting uncertainties must be handled in a highly heuristic fashion through the use of deflated "discount factors."

The Hughes and Powell experiments are limited since they do not actually simulate decision making under uncertainty over a planning horizon. Rather, they report only on a side-by-side

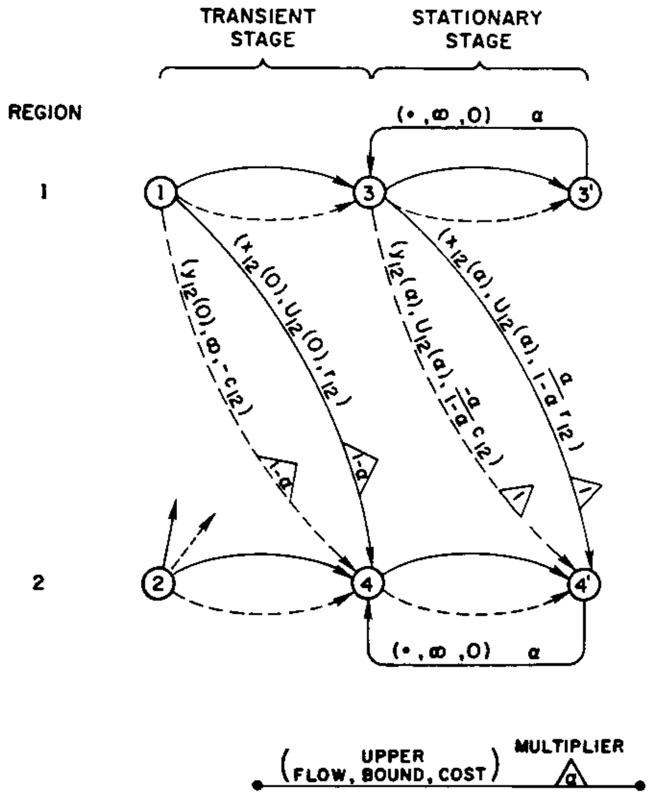

Figure 3

Generalized network for dual equilibrium method, one day per stage

comparison of flows from a single run of a network model. Second, there are several variations to the salvage value method in terms of how the salvage values are computed and the choice of upper bounds on the final link.

A question that often arises in the development of dynamic models is whether an accurate model of future activities is really needed given that the model will be solved repeatedly on a rolling horizon basis. This question must be addressed from two perspectives. First, from the perspective of dispatching trucks, the question is an empirical one that has yet to be carefully and rigorously investigated. Cape [6] compared a simple transportation formulation to a stochastic programming heuristic described below, showing a significant improvement for the latter model. The author is continuing to investigate this issue more thoroughly. As a general rule, good decisions today require good forecasts of future activities, even if we are not going to carry out all of the decisions we optimize for in the future.

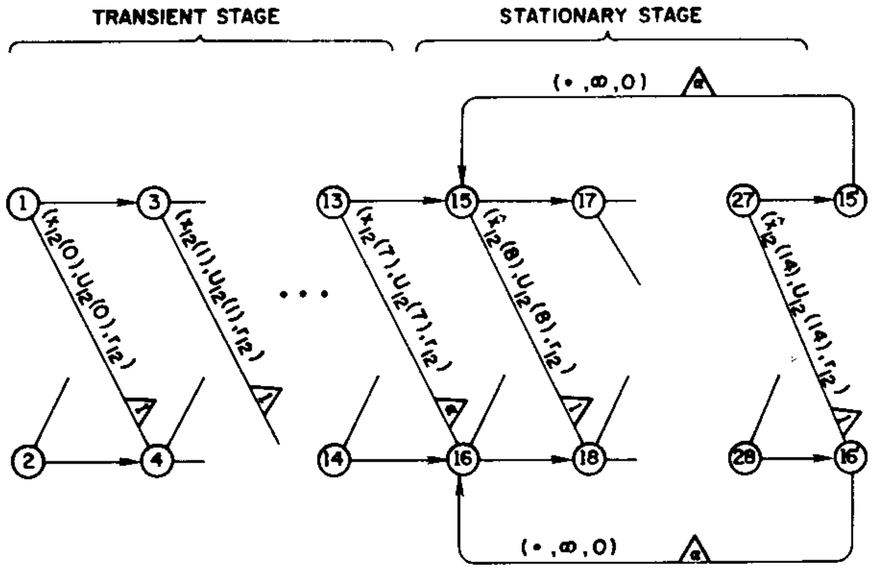

Figure 4

Generalized network for generalized summation method, seven days per stage

The second perspective which is generally ignored is that of pricing and load evaluation. An important but largely neglected aspect of optimization models is their ability to provide estimates of the marginal value of an activity. In the context of the DVA, we might ask what is the value of a load booked from region i, day 1 to region j, day 3. This requires knowing the marginal value of a truck in region j, day 3, which in turn requires that we make our best forecast of what we will be doing on day 3. Our research has shown that dispatching trucks is much easier than pricing their activities, and places much higher demands on the quality of the optimization in the future.

The remainder of this paper reviews different approaches to handling uncertainty in demand forecasts. The next section considers direct modifications to the network formulation, after which Section 4 turns to more classical stochastic optimization based approaches.

3. STOCHASTIC NETWORK FORMULATIONS

The literature on stochastic formulations of the DVA is quite thin. Two approaches have been attempted in the literature, the first resulting in a nonlinear network model with deterministic flows and the second producing a network-based nonlinear math program which represents flows as random variables. In order to build a common framework, the first approach is termed the stochastic DVA with *simple recourse* while the second is the stochastic DVA with *null recourse*, for reasons that are described below.

3.1 The stochastic DVA with simple recourse

Define:

$z_{ij}(t)$ = total flow of trucks assigned to move (loaded or empty) from i to j departing at time t,

$X_{ij}(t)$ = random variable denoting the number of loaded trucks moving from i to j at time t,

$Y_{ij}(t)$ = random variable denoting the number of empty trucks moving from i to j at time t.

From the definition of $z_{ij}(t), X_{ij}(t)$ and $Y_{ij}(t)$, it is clear that

$$X_{ij}(t) = \min[\, z_{ij}(t), F_{ij}(t)\,] \tag{16}$$

and

$$Y_{ij}(t) = z_{ij}(t) - X_{ij}(t) \tag{17}$$

where $F_{ij}(t)$, as before, is a random variable denoting the number of loads that will be available from i to j at time t. The probability distribution of $F_{ij}(t)$ is assumed known. We wish to find an optimal allocation of trucks $\{z\}$ to maximize total expected profits over a given planning horizon:

$$\max_{\{z\}} \sum_{t=0}^{P} \sum_{i\in\mathbf{R}} \sum_{j\in\mathbf{R}} [\, r_{ij}\, E[X_{ij}(t)] - c_{ij}\, E[Y_{ij}(t)]\,] \tag{18}$$

subject to flow conservation on the flows $\{z\}$ and where $E[X_{ij}(t)]$ and $E[Y_{ij}(t)]$ are derived using (16) and (17). Note that we must still choose a planning horizon P and devise a strategy to manage end effects. It is possible to rewrite (18) by defining a profit function $g_{ijt}(z_{ij}(t))$ for each link using:

$$g_{ijt}(z_{ij}(t)) = E[\, r_{ij}\, X_{ij}(t) - c_{ij}\, Y_{ij}(t)\,] \tag{19}$$

which produces the following objective function:

$$\max_{\{z\}} \sum_{t=0}^{P} \sum_{i\in\mathbf{R}} \sum_{j\in\mathbf{R}} g_{ijt}(z_{ij}(t)) \tag{20}$$

subject to flow conservation and nonnegativity constraints on the flows $\{z\}$. Equation (20) is a concave, separable nonlinear network problem which is easily solved using the Frank-Wolfe algorithm which produces a sequence of linear transshipment problems.

An equivalent formulation of the same problem, developed in Powell et al. [21] starts by defining the following:

$S_i(t)$ = total flow through region i at time t,

$\theta_{ij}(t)$ = fraction of the total supply in i at time t that is to be sent to j.

Clearly

$$z_{ij}(t) = \theta_{ij}(t) \cdot S_i(t) \quad \forall\, i, j, t\,. \tag{21}$$

The flow conservation constraints are now written

$$S_i(0) = T_i(0) \quad \forall\, i \tag{22}$$

$$S_i(t) = \sum_{k\in\mathbf{R}} \theta_{ki}(t-1) \cdot S_k(t-1) \quad \forall\, i, t = 1, 2, \ldots \tag{23}$$

$$\sum_{j\in\mathbf{R}} \theta_{ij}(t) = 1 \quad \forall\, i, t = 0, 1, 2, \ldots \tag{24}$$

$$\theta_{ij}(t) \geq 0 \quad \forall\, i, j, t\,. \tag{25}$$

The real decision variables are the flow allocation fractions $\{\theta\}$, implying that we may explicitly incorporate constraints (22) and (23) into supply functions $S_i(t,\theta)$ where the vector θ reflects the fact that the supply of vehicles at i at time t may depend on all flow allocation decisions made prior to t. Now the optimization problem may be written as:

$$\max_{\{\theta\}} G(\theta) = \sum_{t=0}^{P} \sum_{i\in\mathcal{R}} \sum_{j\in\mathcal{R}} g_{ij}\,(\theta_{ij}(t)\cdot S_i(t,\theta)) \tag{26}$$

subject to (24) and (25). Equation (26) is a concave, *nonseparable* objective function with a very simple constraint set. Taking advantage of the acyclic structure of the network, simple recursions can be developed for taking derivatives using equations of the form:

$$\frac{\partial G}{\partial\theta_{ij}(t)} = \frac{\partial g_{ij}(\theta_{ij}(t)\cdot S_i(t,\theta))}{\partial\theta_{ij}(t)} + \frac{\partial G}{\partial S_j(t+1)}\cdot\frac{\partial S_j(t+1)}{\partial\theta_{ij}(t)} \tag{27}$$

and

$$\frac{\partial G}{\partial S_i(t)} = \sum_{j\in\mathcal{R}} \frac{\partial g_{ij}(\theta_{ij}(t)\cdot S_i(t,\theta))}{\partial S_i(t)} + \frac{\partial G}{\partial S_j(t+1)}\;\frac{\partial S_j(t+1)}{\partial S_i(t)}\,. \tag{28}$$

Equations (27) and (28) are applied recursively going backwards in time. Using this approach, the Frank-Wolfe algorithm can be applied to (26) yielding a sequence of trivial subproblems. Powell et al. [21] showed that solving (26) using the decision variables $\{\theta\}$ was approximately five times faster than solving (20) using a more classical formulation based on link flows. The extra computational effort required to find the derivatives using (27) and (28) was more than offset by the speed with which the linearized subproblems could be solved. Particularly interesting is that the second formulation (26) was faster even in terms of iterations (that is, ignoring the time required to solve each iteration), requiring 7 iterations to reach the same objective function value the first approach required 18 iterations to reach.

Separate from the pure algorithmic issues, the use of the vector $\{\theta\}$ as decision variables provides a cleaner statement of the modelling assumptions. Specifically, we are asked to decide, in advance (that is, before we can see realizations of the demands $D_{ij}(t)$) the amount of flow that will be moved from i to j at time t. Then, when the demands are known, we are to move as many loads as we can. Excess vehicles move empty and excess loads are lost. The inability to reallocate excess trucks on one link to handle excess demand on another link defines a strategy with simple recourse and represents a very strong assumption. In the case of truckload trucking, motor carriers are highly responsive to real-time conditions. It is useful to consider, however, that while this model assumes a vehicle will move empty even if there is no load available,, the deterministic model further assumes that the load will be available (that is, that the forecast will come true). The next model represents an attempt to extend the model to provide for a more flexible response.

3.2 The stochastic DVA with null recourse

The model with null recourse is defined as the strategy where if a vehicle cannot be moved loaded from i to j then it will simply be held until time period $t+1$, a kind of null response for the vehicle. Under this assumption, it is necessary to control the flow of loaded and empty vehicles separately. Define

$\alpha_{ij}(t)$ = fraction of the supply of vehicles at region i at time t which is allocated to be moved loaded from i to j,

$\beta_{ij}(t)$ = fraction of the supply of vehicles at region i, time t which is to be moved empty from i to j.

In this case, the flows of loaded and empty vehicles are given by

$$X_{ij}(t) = \min[\,\alpha_{ij}(t)\cdot S_i(t), F_{ij}(t)\,] \tag{29}$$

$$Y_{ij}(t) = \beta_{ij}(t)\cdot S_i(t) \qquad j\neq i\,. \tag{30}$$

The fractions $\alpha_{ij}(t)$ and $\beta_{ij}(t)$ must satisfy

$$\sum_{j\in\mathcal{R}} [\,\alpha_{ij}(t)+\beta_{ij}(t)\,] = 1 \tag{31}$$

where generally $\alpha_{ii}(t)=0$. Let $Y_{ii}(t)$ be the flow of vehicles held at i until $t+1$. This is given by the number of vehicles that were intentionally held at i plus the overflow from links where demand fell short of the allocated capacity. Thus

$$Y_{ii}(t) = \beta_{ii}(t)\,S_i(t) + \sum_{\substack{j\in\mathcal{R}\\ j\neq i}} [\,\alpha_{ij}(t)\cdot S_i(t) - X_{ij}(t)\,]\,. \tag{32}$$

This simple recourse strategy implies that $X_{ij}(t)$ is a random variable (as is $X_{ij}(t)+Y_{ij}(t)$, unlike in the previous model), which in turn means that the supplies $S_i(t)$ are random. Equation 32 is the heart of the difference between the simple recourse model and the null recourse model. Under the simple recourse model, a truck might be "allocated" to move over a traffic lane with 30 percent probability of having a load, meaning a 70 percent probability of moving empty. There is no chance a carrier could operate profitably under such a strategy. The null recourse policy assumes that a truck allocated to move loaded will simply be held in a region if no load arises, thus avoiding the cost of moving empty. This is more accurate than the simple recourse approach but does not model the reality of possibly using a truck on one of multiple outbound traffic lanes, thereby increasing the probability it will move loaded somewhere.

The solution approach follows very closely that used to solve (26). This model was explored in depth by Powell [22]. This work was based in part on earlier work by Jordan [16] and Jordan and Turnquist [17] who considered the empty car distribution problem for railroads. While their model did not consider loaded movements, they did explicitly model stochastic supplies and the concept behind the allocation vectors $\{\alpha\}$ and $\{\beta\}$, while not explicitly stated, is implicit in their formulation.

Both the simple and null recourse models produce nonlinear programs that can be solved fairly efficiently even for large networks. Execution times are roughly 3 to 10 times that for comparable deterministic models (the execution time depends largely on how close to optimality the algorithm is run). Of course the models produce fractional solutions. The more serious problem is that even the null recourse formulation is not a realistic model of actual carrier behavior. Furthermore, while it can be optimized fairly efficiently, the null recourse model is already alarmingly complex due to the need to handle stochastic supplies of trucks. In terms of the mathematical formulation, there does not appear to be much room left for further relaxing the assumptions and still having a workable mathematical model. For this reason, the next section attempts to develop additional insights by formulating the problem as a classical Markov decision process.

4. THE DVA AS A MARKOV DECISION PROCESS

Having effectively run into a dead end with classical network formulations of the stochastic DVA, an alternative approach is to reformulate the problem as a Markov decision process (MDP). Here the simplifying assumptions, particularly that all travel times are one time period, significantly ease the presentation. Assume we have a fleet of K vehicles that may be distributed over R regions, and let $S_i(t)$ be the number of vehicles in region i at time t. If all the travel times between regions are one period, then $S(t) = \{S_1(t), ..., S_R(t)\}$ completely defines the state of the system at time t. The problem is now to optimize the transitions from one transition to the next to maximize the average reward per time period. That is, if $R(t)$ is the profits earned in the transition from t to $t+1$, then we wish maximize $\lim_{s\to\infty} \sum_{t=0}^{s} E[\ R(t)\ ]/s$. An appreciation of the structure of the problem is most easily developed for the case where the fleet $K = 1$ vehicle. After this, the approach is extended to the problem with $K > 1$. The purpose of this exercise is primarily to explore the structure of the decision variables under uncertainty. At the end of this section, we show how the flow splitting approach described in Section 3 can be viewed as a restricted form of the MDP structure.

4.1 The one vehicle MDP

With one vehicle, the state vector $S(t)$ can also be viewed equivalently as a vector giving the current location of the vehicle. Alternatively we may define a new state variable $\bar{S}(t) = i$ if the vehicle is in region i at time t. Now a transition from state i to state j is equivalent to a vehicle moving from region i to region j, earning a random "reward" of R_{ij} where

$$R_{ij} = \begin{cases} r_{ij} & \text{if the movement is made loaded} \\ -c_{ij} & \text{if the movement is made empty.} \end{cases}$$

The problem now is to find the structure of an optimal policy that will determine what a vehicle will do when faced with a set of realizations on the available loads. On a given day, when all the loads are known, the vehicle must accept one of these loads or reject all of them and move empty instead (holding in the region until the next time period is a special case of an empty move). Define:

Λ_{ij} = the decision to move loaded from i to j *if a load is available*, and

E_{ij} = the decision to move empty from i to j.

For a vehicle in region i, the set of all possible actions is given by $\{\Lambda_{ij}, E_{ij}, j \in R = 1, ..., R\}$. Let Δ_i be the set of all possible permutations of this set and let $\delta_i \in \Delta_i$. Thus, for a three region problem, we might have

$$\delta_1 = [\Lambda_{13}, \Lambda_{12}, E_{11}, \Lambda_{11}, E_{12}, E_{13}]. \tag{33}$$

δ_1 is a particular policy that says the vehicle (in region 1) should move loaded from 1 to 3, if there is a load available, otherwise it should move loaded from 1 to 2 and, failing this, it should move empty from 1 to 1. Since the vehicle can always do the empty option, this is the last option that need be considered.

For a given policy δ_i, there is a vector of rewards $\gamma_i = [\gamma_{i1}, \gamma_{i2}, ..., \gamma_{iN}]$, where $N = 2R$, that corresponds to the reward received if a particular option is used. Thus, for the example in (33) we would have

$$\gamma_1 = [r_{13}, r_{12}, -c_{11}, r_{11}, -c_{12}, -c_{13}]. \tag{34}$$

Next, there are probabilities associated with each option. Assume the options in δ_i are ordered from $n = 1, ..., 2R$, and let

$d_{in}(\delta_i(t))$ = probability the vehicle in i is dispatched on the n^{th} option given a policy $\delta(t)$.

These probabilities can be easily worked out if we know the probability that a loaded option is not available (that is, the probability the number of loads from i to j is zero). Again, for the example in (33) we might have

$$\mathbf{d}_i(\delta_i(t)) = [.26\ .33\ .41\ 0\ 0\ 0] \tag{35}$$

where the first empty option receives all the remaining probabilities.

If $R_i(t)$ is the reward earned by a vehicle in region i at time t, then

$$E[R_i(t)] = \sum_{n=1}^{2R} \gamma_{in}(\delta_i(t))\, d_{in}(\delta_i(t)). \tag{36}$$

Assume we are maximizing expected profits $\bar{R}(P)$ over a planning horizon of length P. Our problem is to find a *strategy* $\delta = [\delta(0), \delta(1), ..., \delta(P)]$ that solves

$$\max_{\{\delta\}} \sum_{t=0}^{P} \pi(0) \cdot \mathbf{P}^t \cdot E[\mathbf{R}(t)] \tag{37}$$

where $\pi(0)$ is the initial state vector, $\mathbf{P}$ is the matrix of transition probabilities and $\mathbf{R}(t)$ is the column vector of rewards for being in each state (the convention is used that probabilities are row vectors and rewards are column vectors). The matrix $\mathbf{P}$ is easily derived from the dispatch probabilities.

The MDP in (37) is normally solved through a standard dynamic programming recursion. Let $W_i(t)$ be the optimal expected reward from time t until the end of the planning horizon given the process is in state i.

$$W_i(t) = \max_{\{\delta_i\}} E[R_i(t, \delta(t))] + \sum_{j \in R} p_{ij}(\delta(t)) \cdot W_j(t+1). \tag{38}$$

The recursion in (38) is easily solved. Let $q_i(n)$ denote the *destination* that is implied by the n^{th} option in the vector δ_i, and define

$$w_{in}(t) = \gamma_{in}(\delta_i(t)) + W_{q_i(n)}(t+1). \tag{39}$$

thus $w_{in}(t)$ is the reward if option n is used, taking us to region $q_i(n)$. We can now rewrite (38) to be

$$W_i(t) = \max_{\{\delta_i\}} \sum_{n=1}^{2R} w_{in}(t) \cdot d_{in}(\delta(t)). \tag{40}$$

It is not hard to see that (40) is solved optimally by choosing a vector $\delta_i(t)$ which satisfies

$$w_{i1}(t) \geq w_{i2}(t) \geq ... \geq w_{iN}(t) \tag{41}$$

where as before $N = 2R$. Thus we need simply to rank the options in terms of their direct contribution plus optimal future earnings, thereby maximizing the probability the vehicle will be dispatched on the most lucrative options.

This section describes an optimal algorithm for the finite planning horizon problem for a single vehicle. Under certain conditions, as P becomes large the optimal policy in the first period becomes independent of P (see, for example, Heyman and Sobel [14, pp. 125-138] and Bean and Smith, [2]). Since the state space is so small, the algorithm is easily implementable. Most importantly, the discussion provides insights into the structure of optimal policies that are used in the design of a heuristic in Section 5. Next we consider briefly the problems in extending the classical MDP framework to the K vehicle problem.

4.2 The K vehicle MDP

With K vehicles the state of the system must now be given by

$$\mathbf{S}(t) = \{S_1(t), ..., S_R(t)\} \tag{42}$$

where $S_i(t)$, as before, is the number of vehicles in region i at time t.

This immediately introduces a problem in terms of the size of the state space. Let S be the set of all possible states. It can be shown that for a problem with K vehicles and R regions that

$$|S| = \binom{K+R-1}{R-1}. \tag{43}$$

For a small fleet we might have $K = 100$ and $R = 50$ giving $|S| = 6.7 \times 10^{39}$, while with a full sized fleet with $K = 2000$ and $R = 100$ we get, using Stirling's approximation $|S| \approx 10^{83}$. Even a toy problem with $K = 20$ vehicles and $R = 10$ gives $|S| = 10^7$. Problems smaller than this are not even interesting for testing purposes.

Aside from the size of the state space, the determination of strategies, rewards and transition probabilities, the problem becomes significantly more complex when $K > 1$. It is useful, nonetheless, to at least formulate the structure of a policy variable in this context. With one vehicle, $\delta_i(t)$ represents a ranking of options to be considered by a vehicle in region i at time t. With K vehicles, we need a policy for each vehicle in each region. Consider a system in *state* $s = \{S_1(t), S_2(t), ..., S_R(t)\}$ and a particular region i. We would then have a policy $\delta_{sik}(t)$ which is identical to (33) but would apply to the k^{th} vehicle in region i when the system is in state s, where $0 < k \le S_i(t)$. Now a vehicle moving from i to j no longer implies a transition from state i to j, and as a result the value of sending a truck from i to j, which earlier we could represent as $w_{in}(t)$ (Equation 39 above), is no longer so easily derived. This in turn takes away our optimal policy of ranking options so that (41) is satisfied.

If the MDP could be solved, the outputs would be identical to those from a network model in terms of instructions for the first time period. Given the deterministic supplies of trucks and deterministic opportunities, each dispatch probability vector will be of the form $d_{ikn}(1)$ = [1, 0, 0, ...] since there will always be a "best" option (given the policies for the other vehicles) for each truck, and this option will be available with probability 1. Thus there is a specific instruction for each vehicle in the first period and a much richer set of strategies in the future (these strategies, however, are not directly implemented).

The K vehicle problem quickly gives us a very large state space, a significantly larger space of possible policies and finally eliminates the special structure we used to determine an optimal policy. Further research may yet reveal structure that will yield a workable algorithm, but at this point this line of investigation does not look promising. The exercise does, however, yield insights into the structure of the problem. The strategy vectors $\delta_{sik}(t)$ for each vehicle k in region i with the system in state s is a relatively general mechanism for controlling the flow of vehicles between regions. A restriction of this formulation would use a policy vector of the form $\delta_i(t)$ as given by (33). In this case, we are forcing all vehicles to be dispatched under the same policy independent of the state or the number of vehicles in the system. Note that such a restriction implies that all empty vehicles move in the same direction. Let $d_{ikn}(\delta_i(t))$ be the probability the k^{th} vehicle in region i is dispatched on the n^{th} option. Let $\eta_{in}(x)$ be the fraction of vehicles dispatched on the n^{th} option given that there are $x = S_i(t)$ vehicles in the region, where

$$\eta_{in}(x) = \frac{1}{x} \sum_{k=1}^{x} d_{ikn}(\delta_i(t)) . \tag{44}$$

Clearly $\eta_{in}(x)$ is a nonlinear function of x. Contrast this with the flow splitting variables $\{\theta\}$ or $\{\alpha, \beta\}$ used earlier where the fraction of flow moved over a particular option is controlled directly as decision variables. The comparison is illustrated in Table 1. Assume we are using a fixed policy vector $\delta_i(t) = \{\Lambda_{i3}, \Lambda_{i2}, \Lambda_{i5}, \Lambda_{i1}, E_{ii}\}$ for all vehicles (we could also use a different vector $\delta_{ik}(t)$ for each vehicle k). Assume that the first four dispatch probability vectors d_{i1}, d_{i2}, d_{i3} and d_{i4} corresponding to $\delta_i(t)$ are as shown in Table 1. Using (44) the vectors $\eta_i(x)$ are also calculated for $x = 1, 2, 3$ and 4. Thus the fraction of vehicles dispatched loaded from region i to region 5 is 0.10 if there is $x = 1$ truck in region i. This changes to 0.11 if $x = 2$, 0.16 if $x = 3$ and 0.22 if $x = 4$. As one might expect, the fraction of trucks moving loaded to region 5 varies as a function of the number of trucks in the region. We may alter these probabilities by manipulating the policy vector $\delta_i(t)$ (or the individual policy vector $\delta_{ik}(t)$ for $k = 1,..., 4$). Under the simple recourse strategy, we may change θ_{i5}, thereby directly changing the *total* number of trucks moving from i to 5. Furthermore, this fraction is statistically independent of $S_i(t)$, the supply of trucks in i at time t. The fraction of trucks actually moving loaded is a nonlinear function of $S_i(t)$ which must be worked out from (16). This function, however, will be quite different from $\eta_i(x)$, particularly when one compares the fraction of trucks moving loaded somewhere as a function of x. Under the null recourse strategy, we may use d_{i5} to directly control the fraction of trucks allocated to move loaded from i to 5. This fraction is also independent of $S_i(t)$, although the fraction moving loaded will again be a (strictly decreasing) function of $S_i(t)$. The biggest difference between the simple and null recourse formulations and the MDP formulation is that the latter will yield a much higher probability that a truck will move outbound loaded to some destination.

Table 1
Illustrative dispatch probabilities for a fixed policy vector $\delta_i(t)$ and flow fractions $\eta_i(x)$

Option	Policy $\delta_i(t)$	d_{i1}	d_{i2}	d_{i3}	d_{i4}	$\eta_i(1)$	$\eta_i(2)$	$\eta_i(3)$	$\eta_i(4)$
1	Λ_{i3}	0.5	0.35	0.15	0.05	0.50	0.43	0.33	0.26
2	Λ_{i2}	0.4	0.50	0.45	0.30	0.40	0.45	0.45	0.41
3	Λ_{i5}	0.1	0.12	0.25	0.40	0.10	0.11	0.16	0.22
4	Λ_{i1}	0.0	0.0	0.10	0.15	0.0	0.0	0.03	0.06
5	E_{ii}	0.0	0.0	0.05	0.10	0.0	0.0	0.02	0.04

It is important in the design of an efficient algorithm that the inherent network structure of the problem be recognized. The enumeration of all possible states required by the MDP approach loses this structure, but it does at least provide a rigorous framework for formulating the problem. Most important is the structure of the policy variables used to handle decision making under uncertainty, although the problem now is to determine how to optimize over the set of possible strategies. In the next section we consider an approach that combines the basic ideas used in the dynamic network formulation in Section 2 with the recourse structure developed in this section.

5. STOCHASTIC PROGRAMMING FORMULATION OF THE DVA

A somewhat different literature has evolved under the general heading of stochastic programming with recourse. Although this literature is addressing the same types of problems as those solved using the MDP framework, the language and orientation is somewhat different. In this section, we present the stochastic DVA as a stochastic programming problem with recourse. We then show how the two time period problem (referred to in this literature as a two stage problem) can be solved optimally as a pure network, and from this we present a simple approximation for the n-stage problem which can also be solved as a pure network.

5.1 Background

Stochastic programming has enjoyed a fairly rich literature since the initial work by Dantzig [8], with important recent contributions by Birge and Wets [4], Birge and Wallace [5], and Wets [27, 28, 29], among others. Most recently Wallace [26] deals directly with stochastic programming problems arising in networks, which forms the basis for the discussion here.

As before, let $\mathbf{x}(t)$ and $\mathbf{y}(t)$ be vectors of loaded and empty flows for time period t, with coefficient vectors $\mathbf{r}$ and $\mathbf{c}$, and let $\mathbf{F}(t)$ be the vector of random demands between regions. We also let $\mathbf{L}(t)$ be the vector of the number of known loads moving between regions. For the discussion here let $\mathbf{L}(t)=0, t \geq 1$, meaning that we only know the loads in the first time period. Our problem can now be stated as:

$$\max\ \mathbf{r}^T\mathbf{x}(0) - \mathbf{c}^T\mathbf{y}(0) + Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \tag{45}$$

$$\sum_{j\in R} [\, x_{ij}(0) + y_{ij}(0)\,] = T_i(0) \qquad \forall\, i \tag{45a}$$

$$\sum_{j\in R} [\, x_{ji}(t-1) + y_{ji}(t-1)\,] = \sum_{k\in R} [\, x_{ik}(t) + y_{ik}(t)\,] \quad t = 1, 2, ..., \qquad \forall\, i \tag{45b}$$

$$x_{ij}(t) \leq U_{ij}(t) \qquad \forall\, i, j, t, \tag{45c}$$

$$x_{ij}(t), y_{ij}(t) \geq 0 \qquad \forall\, i, j, t\ . \tag{45d}$$

$Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}})$ gives total expected profits from time periods $t = 1, ..., P$ given the decisions $\overline{\mathbf{x}} = \mathbf{x}(0)$ and $\overline{\mathbf{y}} = \mathbf{y}(0)$ made in time period 0. $Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}})$ is the recourse problem and can be stated as follows. Let $\mathbf{F} = \{\mathbf{F}(1), \mathbf{F}(2), ..., \mathbf{F}(P)\}$ be the random vector of all loads in the future and let $Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}} \mid \mathbf{F})$ be the conditional expected profits given $\mathbf{F}$. Then $Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}} \mid \mathbf{F})$ is given by

$$Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}} \mid \mathbf{F}) = \max \sum_{t=1}^{P} [\, \mathbf{r}^T\mathbf{x}(t) - \mathbf{c}^T\mathbf{y}(t)\,] \tag{46}$$

subject to (45b), (45c) and (45d), where $U_{ij}(t) = F_{ij}(t)$ in 45c. Taking expectations gives:

$$Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}}) = E_{\mathbf{F}}\,\{\, Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}} \mid \mathbf{F})\,\}\ . \tag{47}$$

The difficulty with this approach is that the sample space for $\mathbf{F}$ is extremely large and each outcome involves solving the network transshipment problem in (46). For a two stage problem Wallace proposes replacing $Q\,(\overline{\mathbf{x}}, \overline{\mathbf{y}})$ with a series of cuts which would each successively produce an improved bound.

A standard technique for solving stochastic programming problems is to use a simple recourse strategy. To state this in more classical terms, let $z_{ij}(t) = x_{ij}(t) + y_{ij}(t)$ be the total flow from i to j as is done in Section 3. Now replace constraint (45c) with

$$z_{ij}(t) + z_{ij}^{+}(t) - z_{ij}^{-}(t) = F_{ij}(t) \tag{48a}$$

$$z_{ij}(t), z_{ij}^{+}(t), z_{ij}^{-}(t) \geq 0 \tag{48b}$$

where $z_{ij}^{+}(t)$ is the underage (the extent to which flow falls below demand) and $z_{ij}^{-}(t)$ is the overage, representing empty vehicles. Thus $x_{ij}(t) = z_{ij}(t) - z_{ij}^{-}(t)$ and $y_{ij}(t) = z_{ij}^{-}(t)$. In our problem there is no penalty for underage and the overage cost is the cost of moving empties. The notion of simple recourse is that we decide on the entire vector $\mathbf{z}(t), t = 0, 1, ..., P$, and then use the recourse variables $\mathbf{z}^{+}$ and $\mathbf{z}^{-}$ to respond to uncertainties. Defining the decision variables in terms of $\{z\}$, the recourse function becomes

$$Q\,(\mathbf{z} \mid \mathbf{F}) = \max_{\{\mathbf{z}^{-}(t), t\geq 1\}} \sum_{t=1}^{P} \mathbf{r}^T [\, \mathbf{z}(t) - \mathbf{z}^{-}(t)\,] - \mathbf{c}^T\mathbf{z}^{-}(t) \tag{49}$$

subject to (48a) and (48b). Since the vector $\{\mathbf{z}^{-}\}$ is not constrained by network constraints, we clearly wish to minimize $\mathbf{z}^{-}$ implying that (49) is solved by

$$z_{ij}^{-}(t \mid z_{ij}(t), F_{ij}(t)) = \max\,[\, z_{ij}(t) - F_{ij}(t), 0\,]\ . \tag{50}$$

Let

$$g_{ij}(z_{ij}(t)) = E\,[\, r_{ij}(z_{ij}(t) - z_{ij}^{-}(t)) - c_{ij}\, z_{ij}^{-}(t)\,] \tag{51}$$

where $z_{i\bar{j}}(t)$ is now given by (50). The recourse function is then

$$Q\ (\mathbf{z}) \ = \ \sum_{t=1}^{P} \sum_{i\in R} \sum_{j\in R} g_{ij}(z_{ij}(t))\,. \tag{52}$$

The function $g_{ij}(z_{ij}(t))$ is identical to that given in (19). Combining (52) and (45) gives the same nonlinear network formulation as in (20), although the functions in the first period are linear. In other words, stochastic programming with simple recourse in the context of the DVA is equivalent to deciding in advance how many vehicles to send from i to j at time t, and then sending that number anyway when the demand is known, incurring lost revenues or additional costs as required.

To gain an appreciation of how poor simple recourse performs as a model, consider the following situation typical of a large motor carrier. Assume from a given region i that the forecasted number of loads to each of 50 destinations is described by a Poisson distribution with a mean of .1 loads. If there is a single vehicle in region i and the decision is made to move it to region j, (that is, $z_{ij}=1$), then the probability the vehicle will move empty is $Prob\,[F_{ij}=0]=.1^0\,e^{-.1}\,/\,0!=.905$. In reality, the carrier might adapt a strategy to accept the best available load. Let $F_i=\sum_{j\in R} F_{ij}$ be the total outbound demand. In the latter strategy the probability that the vehicle will have to move empty (or hold) is $Prob\,[F_i=0]=5^0\,e^{-5}\,/\,5!=.000056$. This strategy can be represented using a multiple stage formulation where the decision maker is allowed to respond to the opportunities available to him.

In the next section we show how the two stage DVA can be solved optimally as a pure network.

5.2 The two stage DVA

Remembering that each stage of the DVA is just a transportation problem, it is easy to see that dispatching vehicles out of region i is independent of dispatching out of any other region. Note that this is purely a consequence of the two stage model since in an n stage model the downstream effects of dispatching vehicles implies that dispatching decisions out of region i must be coordinated with those out of all other regions. The two stage problem can be represented using (45) to (47) with $P=1$.

Begin with the recourse function $Q\,(\overline{\mathbf{x}},\overline{\mathbf{y}})$. The solution of (46) can be represented using the same approach used to formulate the problem as an MDP. For trucks in region i, the optimal policy is given by

$$\delta_i(1) \ = \ [\,\Delta_{ik_1}, \Delta_{ik_2}, \ldots, \Delta_{ik_R}, E_{ii}\,, \ldots\,] \tag{53}$$

where $r_{ik_1} \geq r_{ik_2} \geq \ldots \geq r_{ik_R}$, and where we assume that the cost of holding the vehicle in the region, c_{ii} , is less than the cost of moving it empty to any other region. The policy vector in (53) states that we will use a vehicle on the available load with the highest revenue and, if none are available, the vehicle will be held in that region. Since we are only considering a two stage problem, the policy in (53) is clearly optimal for all vehicles in region i, whereas in the P-stage problem we would require the much more complex formulation outlined in Section 4.2 for the N vehicle MDP.

Given the simple structure of the second stage problem, we now need to find $Q\,(\overline{\mathbf{x}},\overline{\mathbf{y}})$. Let

s_i = supply of vehicles in region i at the beginning of the second stage,

$= \sum_{k\in R} [\,x_{ki}(0) \,+\, y_{ki}(0)\,]$,

$Q_i(s_i)$ = expected profits from vehicles dispatched out of region i given a supply of s_i .

Clearly

$$Q\ (\overline{\mathbf{x}},\overline{\mathbf{y}}) \ = \ \sum_{i\in R} Q_i(s_i)\,. \tag{54}$$

Thus we have to find $Q_i(s_i)$. Recall that $d_{ikn}(\delta_i(t))$ is the probability the k^{th} vehicle in region i is dispatched on the n^{th} option, and that $\gamma_{in}(\delta_i(t))$ is the value of the n^{th} option. Define

$v_{ik}(\delta_i(t))$ = value of the k^{th} vehicle in region i.

Then

$$v_{ik}(\delta_i(t)) \ = \ \sum_{n=1}^{2R} d_{ikn}(\delta_i(t)) \ \cdot \ \gamma_{in}(\delta_i(t)) \tag{55}$$

and

$$Q_i(s_i) \ = \ \sum_{k=1}^{s_i} v_{ik}(\delta_i(t))\,. \tag{56}$$

All that is left is determining the dispatch probabilities.

Let

$$\hat{F}_{in} \ = \ \sum_{l=1}^{n} F_{ik_l} \tag{57}$$

where $\hat{F}_{in}$ is the cumulative total number of loads in the *best* n options, $1 \leq n \leq R$. The event that the k^{th} vehicle is dispatched on the n^{th} option is equivalent to the joint event that $\hat{F}_{i,n-1} < k$ *and* $\hat{F}_{in} \geq k$. Thus

$$d_{ikn}(\delta_i(t)) = Prob\,[\,\hat{F}_{i,n-1} < k, \hat{F}_{i,n} \geq k\,]$$

$$= Prob\,[\,\hat{F}_{i,n-1} < k\,] - Prob\,[\,\hat{F}_{i,n} < k\,]$$

where the second equality follows from the identity $P(A \cap B) = P(A) - P(\bar{B})$ when $\bar{B} \subset A$. Thus the dispatch probabilities reduce to the difference between two cumulative distributions. If a vehicle is not dispatched on one of the first n loaded options, then it is always moved on the first empty option which, following (53), means being "held" in region i until the next time period.

Having determined the recourse function, we can now consider the two stage optimization problem, given by (45). The recourse function $Q(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is separable in the variables s_i and from (56) we see that the function is piecewise linear. Thus (45) can be solved exactly as a pure network as indicated in Figure 5. The two stages are easily discerned in the network. The first stage, reflecting "known" loads and empty opportunities, forms a transportation problem and the second stage made up of "stochastic links" represents the value of each additional vehicle in a region.

It is useful to contrast the stochastic programming formulation with that based on Markov decision processes. Within the research literature, it is common to use one of the two approaches but apparently less common to compare the two directly. Stochastic programming uses traditional decision variables representing flows on vehicles. It also requires at least implicitly enumerating all possible outcomes of the random vector $\mathbf{F}$ and solving a network problem for each possible outcome F. The computational challenge has been finding the optimal first period flows without actually enumerating all the outcomes for the second period. The addition of a third stage appears to make the problem completely intractable since in principle every outcome of $\mathbf{F}$ in the second period still requires enumerating all possible outcomes of $\mathbf{F}$ in the third stage.

The Markov decision process formulation uses policies as decision variables where a single policy describes what decisions must be made for *all* possible outcomes of $\mathbf{F}$ in the second stage. Of course, a policy vector $\delta(t)$ is much more complex than a set of link flows $x(t)$ and $y(t)$. On the other hand, rather than enumerating all possible outcomes of $\mathbf{F}$, the MDP framework requires enumerating all possible states $\mathbf{S}(t)$. Since multiple outcomes of $\mathbf{F}$ can produce the same state $\mathbf{S}(t)$, the number of states is smaller (in fact, significantly so) than the number of outcomes of $\mathbf{F}$.

5.3 An approximation for the multistage DVP

This last section looks to combine ideas developed for the multistage *deterministic* DVA with the approach just presented for the two stage stochastic DVA presented in the previous section. The ideas in this section were developed in Powell [23] and are the foundation of a vehicle dispatching system implemented at North American Van Lines (Powell *et al.* [24]). The development here, however, is considerably different and serves to synthesize ideas from dynamic networks, MDP's and stochastic programming.

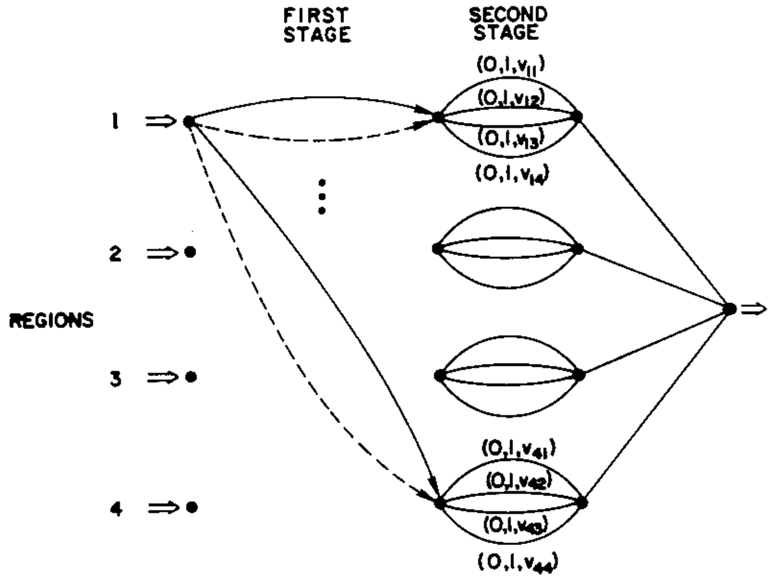

Figure 5
Pure network representation for two stage stochastic programming formulation

The two stage problem involves two sets of decision variables: the policy vector $\delta(1) = \{\delta_1(1), \delta_2(1),..., \delta_R(1)\}$ for the second stage, and flow variables $\mathbf{x}(0)$ and $\mathbf{y}(0)$ for the first stage. For the multistage problem, let $\mathbf{S}(t) = \{S_1(t), S_2(t),..., S_R(t)\}$ be the state of the system given decisions $\{\mathbf{x}(0), \mathbf{y}(0)\}$ in the first stage and policies $\delta(1), \delta(2),..., \delta(t-1)$ up until time t. Let $\Psi(t \mid \mathbf{S}(t))$ be the optimum expected profits from time t until the end of the planning horizon given a system starting in time t. Then

$$\Psi(t \mid S(t)) = \max_{\delta(t)} \{ \sum_{i \in R} \sum_{k=1}^{S_i(t)} \sum_{n=1}^{2R} d_{ikn}(\delta(t)) \gamma_{in}(\delta(t))$$

$$+ E_{S(t+1)} [\Psi(t+1 \mid S(t+1)) \mid S(t), \delta(t)] \} . \tag{61}$$

The expectation reflects the fact that $S(t+1)$ is a random variable, dependent on $S(t)$ and the policy $\delta(t)$. As we did with the deterministic DVA, we would like to replace $\Psi(t+1 \mid S(t+1))$ with a simpler approximation. As before, we will use

$$\hat{\Psi}(t+1 \mid S(t+1)) = \sum_{i \in R} p_i(t+1) S_i(t+1) . \tag{62}$$

where, as before, $p_i(t+1)$ is the salvage value giving the expected value of a vehicle in region i at time $t+1$ until the end of the planning horizon. Taking expectations of (62) gives

$$E [\hat{\Psi}(t+1 \mid S(t+1)) \mid S(t), \delta(t)] = \sum_{i \in R} p_i(t+1) \, E [S_i(t+1)] \tag{63}$$

where

$$E [S_i(t+1) \mid S(t)] = \sum_{i \in R} \sum_{k=1}^{S_i(t)} \sum_{n=1}^{2R} d_{ikn}(\delta(t)) . \tag{64}$$

Combining (61)-(64) yields

$$\Psi(t \mid S(t)) = \max_{\delta(t)} \left\{ \sum_{i \in R} \sum_{k=1}^{S_i(t)} \sum_{n=1}^{2R} d_{ikn}(\delta(t)) \, [\gamma_{in}(\delta(t)) + p_{q_i(n)}(t+1)] \right\}. \tag{65}$$

(65) is now very similar to (40) with $w_{in}(t) = \gamma_{in}(\delta(t)) + p_{q_i(n)}(t+1)$ and can be solved by choosing $\delta_i(t)$ for each region i so that (41) is satisfied. One important difference, however, is that we no longer can guarantee that all loaded moves will be ranked above all empty moves (also, the best empty move may not be to hold in a region until the next day). It is certainly intuitively more reasonable that some empty moves would be ranked above some loaded moves, but in our logic the highest ranked empty option within the vector $\delta_i(t)$ is the lowest option to receive any probability. This behavior is a direct consequence of the use of the linear approximation implicit in (62). We would like some vehicles to move empty to a given destination but we would like the model to recognize that there are declining marginal returns for each additional empty vehicle sent.

This problem can be mitigated heuristically. Let $U_{ij}(\delta(t))$ be a random variable denoting the maximum number of vehicles that we wish to allow to be used for the n^{th} option. The choice of $U_{in}(\delta(t))$ must satisfy

$$U_{ij} \le \begin{cases} F_{ij}(t) & \text{if option } n \text{ represents moving loaded region } j \\ \infty & \text{if option } n \text{ represents moving empty region } j \, . \end{cases}$$

If n is a loaded option, then it is natural (though not necessarily optimal) to use $U_{ij} = F_{ij}(t)$. If n is an empty option, one possible approximation is to use $U_{ij} = E_{ij}(t)$ where $E_{ij}(t)$ is a random variable denoting the *historical* number of empties that have been moved from i to j. This approach has been implemented and works quite well in practice though it imposes additional data requirements that can be hard to explain, as well as creating problems in certain situations.

Using the random variables $\{U_{in}(\delta(t))\}$, it is straightforward to work out the dispatch probabilities $\{d_{ikn}(\delta(t))\}$. Let the indices, $k_1, k_2, \ldots, k_{2R}$ rank the options W_{in} as in (41), and let

$$\hat{U}_{in} = \sum_{l=1}^{n} U_{ik_l} , \tag{66}$$

similar to (57). Then the same arguments leading to (60) gives

$$d_{ikn}(\delta(t)) = Prob [\hat{U}_{i,n-1} < k] - Prob [\hat{U}_{i,n} < k] . \tag{67}$$

The problem can now be solved in a manner similar to the two stage DVA. Let

$$v_{ik}(\delta_i(t)) = \text{the value of the } k^{th} \text{ vehicle in region } i \text{ under policy } \delta_i(t)$$

$$= \sum_{n=1}^{2R} d_{ikn}(\delta(t)) \cdot w_{in}(t) . \tag{68}$$

The recourse function $Q(\overline{\mathbf{x}}, \overline{\mathbf{y}})$ can now be approximated as a separable, piecewise linear function. Let $\hat{Q}(S(t))$ denote this approximation. Then

$$\hat{Q}(S(t)) = \sum_{i \in R} \hat{Q}_i(S_i(t)) \tag{69}$$

where

$$\hat{Q}_i(S_i(t)) = \sum_{k=1}^{S_i(t)} v_{ik}(\delta_i(t)) . \tag{70}$$

This approximation is only useful when the state vector $S(t)$ is known with certainty. The latest time period for which this is true is period $t = 1$. The complete optimization problem from the first stage onward is given by

$$\max\ \mathbf{r}^T\,\mathbf{x}(0) - \mathbf{c}^T\,\mathbf{y}(0) + \hat{Q}\,(S(1))\,. \tag{71}$$

This can be solved using the same pure network shown in Figure 5. The flows on links in the first stage are given by $\{x(0), y(0)\}$ while the flows on links starting in the second stage into the supersink are given by $S(1)$. The difference is that the coefficients on these links capture the approximate value of an additional vehicle starting at time $t = 1$ out to $t = P$.

A particularly important property of this formulation is that the salvage values $p_i(t)$ satisfy

$$p_i(t) \rightarrow \eta_i + \eta \cdot (P - t) \quad \text{as } P \rightarrow \infty \tag{72}$$

where η_i is a region specific factor (reflecting the relative value of a vehicle in one region over the other), and η is a system parameter giving the limiting expectation of the daily contribution of a vehicle. This result is a well known property of Markov reward processes (see, for example, Bhat [3]). Thus, for P sufficiently large, a further increase in P will increase the costs on all arcs leading into the supersink by a constant. Experiments reported in Powell [23] show that for a particular truckload motor carrier that (72) becomes quite accurate for $P > 10$ days. A simple recursion for calculating the values $\{p_i(s)\}$ is described in the appendix. The idea is that future activities will approximately follow recent history. Also, errors in the salvage values will be partially mitigated by the fact that the value of the k^{th} truck in region i, $v_{ik}(\delta(t))$, is formed by a weighted average of salvage values, reducing the effect of an error in any one value.

6. SUMMARY AND CONCLUSIONS

The principal goal of this paper has been to expose a range of modelling frameworks and approximations. It was not possible to describe every variation, and attention was focussed on models which recognized the dynamic nature of the problem and uncertainties in forecasting. The large majority of the vehicle routing literature assumes a deterministic problem, and stochastic routing problems incorporate uncertainty in customer demands but not in the tours.

Table 2 provides an overview of the different modelling approaches and their characteristics in terms of problem size, ability to handle details about specific drivers and loads, and the resulting optimization problem and solution algorithm. Several of the models (1, 3, 4, 6, 7, 9 and 10) are single commodity network flow problems which are unable to handle details about

TABLE 2

Comparison of alternative modelling formulations

FORMULATION	PROBLEM SIZE[1]	LEVEL OF DETAIL	FORECASTING	OBJECTIVE FUNCTION	MODEL OUTPUT	SOLUTION ALGORITHM
1: TRANSPORTATION	1,600 LINKS 122 NODES	SINGLE COMMODITY	NONE, OR SIMPLE SALVAGE VALUES	LINEAR NETWORK	INTEGER LINK FLOWS	NETWORK SIMPLEX
2: ASSIGNMENT	7,500 LINKS 660 NODES	DRIVERS AND LOADS MODELLED EXPLICITLY	NONE, OR SIMPLE SALVAGE VALUES	LINEAR NETWORK	INTEGER DRIVER ASSIGNMENTS	NETWORK SIMPLEX
3: TRANSSHIPMENT INTEGER FORECASTS	15,000 LINKS 1,200 NODES	SINGLE COMMODITY	HEURISTIC GENERATION OF FORECASTER	LINEAR NETWORK	INTEGER LINK FLOWS	NETWORK SIMPLEX
4: TRANSSHIPMENT FRACTIONAL FORECASTS	30,000 LINKS 1,200 NODES	SINGLE COMMODITY	DETERMINISTIC FORECASTS	LINEAR NETWORK	FRACTIONAL LINK FLOWS	NETWORK SIMPLEX
5: ASSIGNMENT/ TRANSSHIPMENT[2]	22,500 LINKS 1,800 NODES	DRIVERS/LOADS MODELLED EXPLICITLY IN FIRST TIME PERIOD	HEURISTIC GENERATION OF FORECASTED LOADS	LINEAR NETWORK	INTEGER DRIVER ASSIGNMENTS AND LINK FLOWS	NETWORK SIMPLEX
6: STOCHASTIC NETWORK-SIMPLE RECOURSE	30,000 LINKS 1,200 NODES	SINGLE COMMODITY	STOCHASTIC DEMANDS	NONLINEAR, SEPARABLE NETWORK	FRACTIONAL LINK FLOWS	FRANK-WOLFE
7: STOCHASTIC NETWORK NULL RECOURSE	30,000 LINKS 1,200 NODES	SINGLE COMMODITY	STOCHASTIC DEMANDS	NONLINEAR, NONSEPARABLE NETWORK	FRACTIONAL LINK FLOWS	FRANK-WOLFE
8: MARKOV DECISION	10^{85} STATES[3]	SINGLE COMMODITY	STOCHASTIC DEMANDS	NONLINEAR	INTEGER FLOWS/ DISPATCH POLICIES	DYNAMIC PROGRAMMING
9: STOCHASTIC PROGRAMMING	10,000 LINKS 500 NODES	SINGLE COMMODITY	STOCHASTIC DEMANDS-HISTORICAL ACTIVITIES IN THE FUTURE	LINEAR NETWORK	INTEGER LINK FLOWS	NETWORK SIMPLEX
10: ASSIGNMENT/STOCHASTIC PROGRAMMING (N-STAGE HEURISTIC)	17,500 LINKS 1,100 NODES	DRIVERS/LOADS MODELLED EXPLICITLY IN FIRST TIME PERIOD	STOCHASTIC DEMANDS-HISTORICAL ACTIVITIES IN THE FUTURE	LINEAR NETWORK	INTEGER DRIVER ASSIGNMENTS AND LINK FLOWS	NETWORK SIMPLEX
11: SET PARTITIONING	30,000 TOURS 2,000 CON-STRAINTS	DRIVERS/LOADS MODELLED EXPLICITLY THROUGHOUT	HEURISTIC GENERATION OF FORECASTED LOADS	LINEAR PROGRAM	DRIVER TOURS	LAGRANGIAN RELAXATION

1: Problem sizes are approximately estimated based on 60 regions, 10 day explicit planning horizon (for transshipment networks) and assuming there are 300 drivers and loads actively being assigned.

2: Characterized by an assignment problem front-end and a deterministic transshipment network with integer forecasts.

3: Based on a 600 vehicle fleet.

specific drivers and loads. These details include equipment type, the precise deadhead distance from a driver to a load, precise time of availability of a driver and the precise pickup and delivery times of each load. The assignment model (2) is the only network flow formulation where individual drivers and loads are represented explicitly, allowing most details to be incorporated. The key limitation is its inability to perform these assignments in the context of any forecasting. On the other hand, the inability of network flow models to handle driver and load details is a major impediment to their practical implementation in the real-world.

Cape [6] combined the assignment model and the network flow model into a single network where each driver and load is explicitly represented by a node in the first time period. From a driver node, links might extend to five or ten known loads (which are within a reasonable distance and which are compatible in terms of equipment type and driver arrival and pickup times). In addition, there will be a link from a driver node to a node associated with the driver's region from the first time period, out of which forecasted opportunities are modelled. Flow from a driver node to a known load node flow forward in time ending in a node corresponding for the region and time period where the load terminates. Thus, details about an individual driver and load are retained for the first assignment and are then lost as the flows move into the future. This combined assignment/transshipment model is represented by formulations 5 and 10 in Table 2.

The last formulation, based on set partitioning, was not discussed in this paper but was included in the table for completeness. This approach, which has received widespread attention both in the research literature and in practice, requires generating feasible tours for each driver and then choosing the best set of tours so that each load is covered by one driver. This approach allows for a very high level of detail in representing drivers and loads both in the present as well as the future. The difficulty here is that an extremely large number of tours must often be generated, resulting in a very large integer programming problem, generally restricting their use to smaller private fleet operations. In addition, this approach does not lend itself well to forecasting uncertainties.

Among the 11 formulations, the Markov decision process approach is at this time restricted to toy problems, as would be exact N-period stochastic programming formulations. Aside from the MDP approach and the set partitioning approach, which will comfortably handle fleets of several hundred trucks, all the remaining formulations will easily handle fleets of several thousand trucks. This is true even of the simple and null recourse models, although they are somewhat slower due to their nonlinear nature. However, the fractional solutions produced by the nonlinear models, as well as the transshipment model with fractional forecasts, require some method to "integerize" the solution prior to implementation.

The remainder of this section reviews implementation issues involved with DVA models in general and the state of implementation in the industry. Finally, we review major research issues still facing the dynamic vehicle allocation problem.

6.1 Implementation of DVA models

Real-world implementations of dynamic fleet dispatching models are still quite few. The models are still evolving in terms of their ability to properly handle forecasting while also coping with the high level of detail that is required to manage a real operation. There are two applications of DVA models. The first is in a batch mode which might be run once or twice a day to determine repositioning strategies. This mode places less emphasis on specific real-time details and instead focuses attention on the broader pattern of surpluses and deficits. The second mode is for real-time driver dispatching, which places a very high emphasis on driver and load details so as not to make infeasible assignments. Using a network model for real-time driver assignments places much higher demands on the carrier's MIS system, which must be up-to-date at all times, and on the network model, which must be capable of detecting changes in the drivers or loads and then reoptimizing from a previous optimal solution within a few seconds.

The data requirements for all the DVA models are effectively the same, with the only exception being the assignment or transportation formulations which may not require any forecast information. Regardless of whether the model is being run in batch or real-time, a network model requires real-time data on drivers, trucks and loads pending. In addition, there is a set of base files which are used to forecast future activities as well as to provide information for calculating empty distances. When the model is run, it is necessary first to extract the status of each driver, including his estimated time of arrival, his destination and his equipment type (in some applications it is necessary to know his domicile and recent dispatching history). If there are different compensation rates for drivers, this will also be needed. Information about current pending loads includes origin and destination, pickup and delivery dates, equipment restrictions and compensation rates.

Base historical files required for forecasting include region to region average empty and loaded distances and travel times, empty movement costs and average contribution per load. Also needed is a set of models for forecasting loads over a 10 or 20 day planning horizon. These models typically work on a one year base of data supplemented by recent activities. Historical loaded contributions (revenue minus direct operating costs) and travel times reflect actual delivery costs (including special handling charges and extra drop-off costs) and additional times resulting from multiple stops.

The biggest hurdle facing most carriers is the lack of an up to date MIS system that both retains the necessary historical data as well as being able to provide current extracts of drivers and loads. Second to this is the traditional hesitation of management to accept help in day to day operations. Just the same, three carriers (to this author's knowledge) are actively using network models for fleet management. The first to do so uses a dynamic transshipment model (formulation 3 in Table 2) each night to plan general fleet movements. Then, an assignment model is used in real-time to perform detailed assignments of drivers to loads. This application is not reported in the research literature and no quantified estimates of impacts are available. However, the system has been in use for over six years suggesting that management is quite happy with the results.

More recently, the author was involved in the installation of a package called LOADMAP, which is an algorithm based on the N-stage stochastic programming heuristic, at North American Van Lines' Commercial Transport division. This implementation is described in Powell *et al.* [24]. To estimate the value of the package, a simulated game was conducted where two teams of six dispatchers each, made up of upper management from North American Van Lines, competed against each other. The performance of these teams was then compared to the performance of the network model. The model outperformed the best of the two teams by 12 percent, with 43 percent fewer refused loads (loads the carrier was unable to carry), 15 percent fewer empty miles and 6 percent higher revenues. This package is being used by two motor carriers and is run in batch approximately three to six times per day.

6.2 Directions for further research

The challenge facing the dynamic vehicle allocation problem is one of developing a computationally feasible algorithm which incorporates planning uncertainties. For the most part stochastic considerations have been largely ignored within the vehicle routing literature. At the same time, the stochastic optimization literature has not progressed very far in terms of handling large problems.

Some of the research directions that are of highest priority include the following:

1. We do not have a rigorous formulation of the stochastic DVA. Sections 4 and 5 of this paper provide a foundation for the structure of the decision variables, but this presentation needs to be firmed up considerably.
2. Can the special structure of the problem be exploited to provide a computationally feasible, optimal solution to the MDP or stochastic programming formulations? The one-vehicle MDP and the two stage stochastic program with network recourse provide glimpses of what is possible here.
3. Does a planning horizon exist, where the optimal policy for time period $t=0$ based on a P period horizon is optimal as $P \to \infty$? Recent research in this area has established conditions for planning horizons, and these should be investigated.
4. Can bounds be developed to evaluate the efficiency of heuristics? It is likely that a bound for a medium to large problem will be more useful than an optimal solution for a very small problem.

Separate from basic theoretical issues are a range of more experimental research topics. These include:

5. The development of the software to rigorously test alternative heuristics. Research *is* progressing in this area and has exposed a variety of important experimental design questions.
6. The most obvious question is, of course, how well do the different formulations actually perform in a rigorous test environment? It is possible, for example, that the simple transportation network in Figure 2 will perform adequately. Actual performance may easily depend on the degree of uncertainty.

APPENDIX

Both the deterministic transshipment networks and the N-stage stochastic programming heuristic make use of regional salvage values, $p_i(t)$, giving the expected net contribution of a truck in region i at time t until the end of a secondary planning horizon, P_s. This can be accomplished through a simple backward recursion. Assume we have available the following:

$g_{ij}(t)$ = forecasted number of trucks moving loaded from i to j at time t,

$e_{ij}(t)$ = forecasted number of trucks moving empty from i to j at time t.

This information can be obtained in two ways. First it is possible to use six months of actual historical activities which are worked into a set of weekly averages. The weekly cycle is then assumed to repeat itself indefinitely. This approach has actually been applied in practice but suffers from some important limitations. First of course is the fact that a six month rolling average does not necessarily forecast the future. Second, and actually more significant, is that from a practical perspective a carrier's database on empty activities, $e_{ij}(t)$, can be of low quality. Often there is no record of activities of trucks holding in a region, $e_{ii}(t)$, which must then be inferred from flow conservation equations. The principal advantage of the use of historical activities is that they represent actual activities and as such may provide a better prediction of actual future costs and revenues.

The second method for estimating loaded and empty activities is to develop a deterministic transshipment network model with a planning horizon P_s' that is substantially longer than P_s. The activities $g_{ij}(t)$ and $e_{ij}(t)$ are then just the optimal loaded and empty flows off this network. Given that integer solutions are not really necessary here, it is best to use fractional upper bounds on the loaded movement links. The advantage of this approach is that the loaded and especially the empty activities become true forecasts. The disadvantage is that the network is very large, since $P_s > P$, and $P_s' > P_s$, and because the use of fractional forecasts greatly expands the number of links required. Also, one is never sure that the network model is actually predicting future activities. Note that since this network model cannot use any salvage values, it will be necessary to use $P_s' \gg P_s$ to mitigate truncation effects. Once the loaded and empty activities are estimated, salvage values can be calculated as follows. Let

$$p_i(P_s) = 0 \quad \forall i \, . \tag{A.1}$$

Then, beginning with $t = P_s - 1$ and working backward in time, let

$$P_i(t) = \sum_{j \in R} \theta_{ij}(t) \cdot w_{ij}(t) - \alpha_{ij}(t) \cdot \overline{w}_{ij}(t) \tag{A.2}$$

where

$$w_{ij}(t) = \begin{cases} r_{ij}(P_s - t)/t_{ij} & \text{if } P_s - t \leq t_{ij} \\ r_{ij} + p_j(t + t_{ij}) & \text{otherwise} \end{cases} \tag{A.3}$$

and where $\overline{w}_{ij}(t)$ is defined similarly using the empty cost c_{ij} instead of the load contribution. Note that we are allowing the travel times, t_{ij}, to differ from unity. The fractions θ_{ij} and α_{ij} are the fraction of trucks moving loaded and empty, given by

$$\theta_{ij}(t) = \frac{g_{ij}(t)}{\sum_{l \in R} (g_{ie}(t) + e_{ie}(t))} \tag{A.4}$$

and

$$\alpha_{ij}(t) = \frac{e_{ij}(t)}{\sum_{l \in R} (g_{il}(t) + e_{il}(t))} . \tag{A.5}$$

Equation A.2 defines a backward recursion that is exceptionally fast and provides salvage values that are fairly robust with respect to errors in the estimates of the activity variables.

REFERENCES

[1] Baker, L., "Overview of Computer Based Models Applicable to Freight Car Utilization", Report prepared for U.S. Department of Transportation, NTIS, Springfield, VA (1977).

[2] Bean, J. C. and Smith, R. L., "Conditions for the Existence of Planning Horizons", *Mathematics of Operations Research* 9 (1984) 391-401.

[3] Bhat, U. N. *Elements of Applied Stochastic Processes* (John Wiley and Sons, New York, 1984).

[4] Birge, J. R. and Wets, R. J-B., "Designing Approximation Schemes for Stochastic Optimization Problems, in Particular for Stochastic Programs with Recourse", *Mathematical Programming Study* 27 (1986) 54-102.

[5] Birge, J. R. and Wallace, S. W., "A Separable Piecewise Linear Upper Bound for Stochastic Linear Programs", University of Michigan, Department of Industrial and Operations Engineering Working Paper, (1987).

[6] Cape, D. J., "MICROMAP: A Dynamic Dispatch and Planning Model for Truckload Motor Carriers", Bachelor's thesis, Department of Civil Engineering, Princeton University, (1987).

[7] Chih, K. C-K., "A Real Time Dynamic Optimal Freight Car Management Simulation Model of the Multiple Railroad, Multicommodity Temporal Spatial Network Flow Problem", Ph.D. dissertation, Department of Civil Engineering and Operations Research, Princeton University, (1986).

[8] Dantzig, G. B., "Linear Programming Under Uncertainty", *Management Science* 1 (1955) 197-206.

[9] Dejax, P. J., and Crainic, T. G., "Models for Empty Freight Vehicle Transportation Logistics", *Transportation Science* to appear (1987).

[10] Grinold, R., "Finite Horizon Approximations of Infinite Horizon Linear Programs", *Mathematical Programming* 12 (1977) 1-17.

[11] Grinold, R., "Convex Infinite Horizon Programs", *Mathematical Programming* 15 (1983) 64-82.

[12] Grinold, R., "Model Building Techniques for the Correction of End Effects in Multistage Convex Programs", *Operations Research* 31 (1983) 407-431.

[13] Grinold, R. and Hopkins, D. S. P., "Computing Optimal Solutions for Infinite-Horizon Mathematical Programs with a Transient Stage", *Operations Research* 21 (1973) 179-187.

[14] Heyman, D. and Sobel, M. J., *Stochastic Models in Operations Research Vol. 11*, (McGraw-Hill, New York, 1984).

[15] Hughes, R. E. and Powell, W. B., "Mitigating End Effects in the Dynamic Vehicle Allocation Model", forthcoming in *Management Science* (1985).

[16] Jordan, W. C., "The Impact of Uncertain Demand and Supply on Empty Rail Car Distribution", Ph.D. dissertation, Cornell University, Ithaca, New York, (1982).

[17] Jordan, W. C. and Turnquist, M. A., A "Stochastic, Dynamic Model for Railroad Car Distribution", *Transportation Science* 17 (1983) 123-145.

[18] Misra, S. C., "Linear Programming of Empty Wagon Disposition", *Rail International* 3 (1972) 151-158.

[19] Ouimet, G. P., "Empty Freight Car Distribution", Master Thesis, Queen's University, Kingston, Ontario (1972).

[20] Philip, C. E. and Sussman, J. M., "Inventory Model of the Railroad Empty Car Distribution Process", *Transportation Research Record* 656 Transportation Research Board (1977) 52-60.

[21] Powell, W. B., Sheffi, Y. and Thiriez, S., "The Dynamic Vehicle Allocation Problem with Uncertain Demands", *Ninth International Symposium on Transportation and Traffic Theory*, (1984).

[22] Powell, W. B., "A Stochastic Model of the Dynamic Vehicle Allocation Problem", *Transportation Science* 20 (1986) 117-129.

[23] Powell, W. B., "An Operational Planning Model for the Dynamic Vehicle Allocation Problem with Uncertain Demands", *Transportation Research* 21B (1987) 217-232.

[24] Powell, W. B., Sheffi, Y., Nickerson, K., Butterbaugh, K. and Atherton, S., "Maximizing Profits for Truckload Motor Carriers: A New Framework for Pricing and Operations", *Interfaces* to appear (1987).

[25] Shan, Yen-Shwin, "A Dynamic Multicommodity Network Flow Model for Real Time Optimal Rail Freight Car Management", Ph.D. dissertation, Department of Civil Engineering and Operations Research, Princeton University, (1985).

[26] Wallace, S. W., "Solving Stochastic Programs with Network Recourse", *Networks* 16 (1986) 295-317.

[27] Wets, R. J-B., "Solving Stochastic Programs with Simple Recourse", *J. Stochastics* 10 (1983) 219-242.

[28] Wets, R. J-B., "Stochastic Programming: Solution Techniques and Approximation Schemes" in: Bachem, A., Grotschel M., and Kurte, B. (eds.), *Mathematical Programming: The State of the Art* (Springer-Verlag, Berlin, 1983) 565-603.

[29] Wets, R. J-B., "Large Scale Programming Techniques in Stochastic Programming", Working Paper WP-84-90, IIASA, Austria, in: Ermoliev Y., and Wets, R. J-B. (eds.), *Numerical Methods in Stochastic Programming*, (1984).

[30] White, W. W. and Bomberault, A. M., "A Network Algorithm for Empty Freight Car Allocation", *IBM System Journal* 8 (1969) 147-169.