A Stochastic Fleet Composition Problem

Ryan Loxton, Qun Lin, Kok Lay Teo

Department of Mathematics and Statistics, Curtin University, Perth, Western Australia

Abstract

In this paper, we consider the problem of forming a new vehicle fleet, consisting of multiple vehicle types, to cater for uncertain future requirements. The problem is to choose the number of vehicles of each type to purchase so that the total expected cost of operating the fleet is minimized. The total expected cost includes fixed and variable costs associated with the fleet, as well as hiring costs that are incurred whenever vehicle requirements exceed fleet capacity. We develop a novel algorithm, which combines dynamic programming and the golden section method, for determining the optimal fleet composition. Numerical results show that this algorithm is highly effective, and takes just seconds to solve large-scale problems involving hundreds of different vehicle types.

Keywords: Fleet Composition, Dynamic Programming, Golden Section Method, Convex Optimization

1. Introduction

Many companies rely on a private vehicle fleet to transport goods, equipment, and personnel. The size and composition of such a fleet must be carefully chosen to meet the company's operational requirements. If the fleet is too small and cannot meet the company's requirements, then many additional vehicles will need to be hired, resulting in excessive hiring costs. On the other hand, if the fleet is too large, then some vehicles will be idle for long periods of time, resulting in a significant opportunity cost. This tradeoff leads to an optimization problem—the so-called *fleet composition problem*—in which the number of vehicles in the fleet needs to be chosen to minimize total cost.

Kirby [6] was one of the first researchers to investigate the fleet composition problem. Kirby considered a very simple problem in which the vehicle fleet is *homogeneous*—that is, all vehicles are of the same type. More complicated models, some of which incorporate vehicle routing, have since been developed; see [4, 5, 10, 11] for more details.

Ghiani, Laporte, and Musmanno [3] have considered a homogeneous fleet composition model in which the cost function is the sum of fixed, variable, and hiring costs. We have extended this model to cater for heterogeneous fleets (fleets with multiple vehicle types) in our previous work [7]. Both the model in [3] and our extended model in [7] assume that future vehicle requirements are known—an unrealistic assumption. In practice, it is usually impossible to predict future vehicle requirements exactly.

In this paper, we extend the models in [3, 7] to the more realistic case where future vehicle requirements follow a given probability distribution. The resulting fleet composition problem is much more complicated than those in [3, 7]. We develop an efficient algorithm, which uses the golden section method to solve a dynamic programming recurrence equation, for determining the optimal fleet composition. As we show in Section 5, combining dynamic programming and golden section search in this way yields excellent results. This combined approach is a major improvement over the algorithm in [7].

2. Problem Formulation

Consider a company or organization that needs to purchase a vehicle fleet to operate over a future time horizon. Let m denote the number of vehicle types to be included in the fleet, and let n denote the number of periods in the time horizon.

Let q_{ij} denote the number of type-i vehicles required during period j. We assume that q_{ij} is a discrete random variable with probability function

$$P(q_{ij} = k) = \theta_{ijk}, \qquad k = 0, \dots, N_i,$$

where N_i is a given integer and θ_{ijk}, $k = 0, \ldots, N_i$ are given real numbers such that

$$\theta_{ijk} \geq 0, \qquad k = 0, \ldots, N_i,$$

and

$$\sum_{k=0}^{N_i} \theta_{ijk} = 1.$$

Let p_i be a decision variable representing the number of type-i vehicles to be included in the fleet. Note that p_i remains fixed throughout the planning horizon—we do not allow the fleet size to change from period to period. We impose the following constraint on the total fleet size:

$$\sum_{i=1}^{m} p_i \leq p_{\max}, \tag{1}$$

where $p_{\max}$ is a given integer.

Consider an arbitrary period j. If $p_i < q_{ij}$, then the number of type-i vehicles required during period j exceeds the number of type-i vehicles in the fleet, and thus an additional $q_{ij} - p_i$ type-i vehicles will need to be hired during this period. On the other hand, if $p_i \geq q_{ij}$, then there is no need to hire type-i vehicles during period j. Thus, the total number of type-i vehicles hired during period j will be equal to $\max(q_{ij} - p_i, 0)$. Let γ_i denote the cost of hiring a type-i vehicle for one period. Then the total hiring cost for type-i vehicles is

$$C_i^h(p_i) = \gamma_i \sum_{j=1}^{n} \max(q_{ij} - p_i, 0).$$

Let α_i denote the *fixed cost* per period of a type-i vehicle. Fixed costs include the initial purchase cost (minus the salvage value), registration fees, insurance premiums, and other costs that do not depend on how often the vehicle is used. The total fixed cost for type-i vehicles is

$$C_i^f(p_i) = n\alpha_i p_i.$$

Let β_i denote the *variable cost* incurred when a type-i vehicle is used for one period. Variable costs are generally due to maintenance. If $p_i \geq q_{ij}$, then only q_{ij} of the fleet's type-i vehicles will be used during period j, but if $p_i < q_{ij}$, then all of the fleet's type-i vehicles will be used during period j. Thus, the number of owned type-i vehicles used during period j will be equal to $\min(q_{ij}, p_i)$. It follows

that the total variable cost for type-i vehicles is

$$C_i^v(p_i) = \beta_i \sum_{j=1}^{n} \min(q_{ij}, p_i).$$

By summing the fixed, variable, and hiring costs, we obtain the total cost of purchasing p_i vehicles of type i:

$$C_i(p_i) = C_i^f(p_i) + C_i^v(p_i) + C_i^h(p_i)$$

$$= \underbrace{n\alpha_i p_i}_{\text{Fixed costs}} + \underbrace{\beta_i \sum_{j=1}^{n} \min(q_{ij}, p_i)}_{\text{Variable costs}}$$

$$+ \underbrace{\gamma_i \sum_{j=1}^{n} \max(q_{ij} - p_i, 0)}_{\text{Hiring costs}}.$$

Note that the cost of maintaining an owned vehicle for one period must be less than the cost of hiring the same vehicle for one period—otherwise, there would be no reason to purchase vehicles. Thus, we assume that β_i is less than γ_i. The key question that now arises is: what values of p_i, $i = 1, \ldots, m$ minimize the total expected cost subject to the fleet size constraint (1)? We formulate this question as the following stochastic programming problem:

$$\begin{aligned} \underset{p_1, \ldots, p_m}{\text{Minimize}} \quad & E\left\{ \sum_{i=1}^{m} C_i(p_i) \right\} \\ \text{subject to} \quad & \sum_{i=1}^{m} p_i \leq p_{\max} \\ & p_1, \ldots, p_m \in \mathbb{Z}^+ \cup \{0\} \end{aligned} \tag{2}$$

The parameters in Problem (2) are summarized in Table 1.

The expected cost function in Problem (2) can be simplified as follows. First, since expectation is a linear operator,

$$E\left\{ \sum_{i=1}^{m} C_i(p_i) \right\} = \sum_{i=1}^{m} E\{C_i(p_i)\}$$

and

$$\begin{aligned} E\{C_i(p_i)\} = n\alpha_i p_i & + \beta_i \sum_{j=1}^{n} E\{\min(q_{ij}, p_i)\} \\ & + \gamma_i \sum_{j=1}^{n} E\{\max(q_{ij} - p_i, 0)\}. \end{aligned} \tag{3}$$

m	Number of vehicle types
n	Number of periods in the time horizon
α_i	Fixed cost per period of a type-i vehicle
β_i	Variable cost per period of a type-i vehicle
γ_i	Hiring cost per period of a type-i vehicle
θ_{ijk}	Probability that k type-i vehicles will be required during period j
N_i	Maximum number of type-i vehicles required during a single period
$p_{\max}$	Maximum fleet size

Table 1: Parameters in Problem (2).

Furthermore, it follows from basic probability theory that

$$E\{\min(q_{ij}, p_i)\} = \sum_{k=0}^{N_i} \min(k, p_i) P(q_{ij} = k)$$
$$= \sum_{k=0}^{N_i} \theta_{ijk} \min(k, p_i)$$

and

$$E\{\max(q_{ij} - p_i, 0)\} = \sum_{k=0}^{N_i} \max(k - p_i, 0) P(q_{ij} = k)$$
$$= \sum_{k=0}^{N_i} \theta_{ijk} \max(k - p_i, 0).$$

Substituting these two equations into equation (3) gives

$$E\{C_i(p_i)\} = n\alpha_i p_i + \beta_i \sum_{j=1}^{n} \sum_{k=0}^{N_i} \theta_{ijk} \min(k, p_i) \\ + \gamma_i \sum_{j=1}^{n} \sum_{k=0}^{N_i} \theta_{ijk} \max(k - p_i, 0). \quad (4)$$

Thus, Problem (2) can be rewritten as follows:

$$\underset{p_1, \ldots, p_m}{\text{Minimize}} \quad \sum_{i=1}^{m} \left\{ n\alpha_i p_i + \beta_i \sum_{j=1}^{n} \sum_{k=0}^{N_i} \theta_{ijk} \min(k, p_i) \\ + \gamma_i \sum_{j=1}^{n} \sum_{k=0}^{N_i} \theta_{ijk} \max(k - p_i, 0) \right\}$$

$$\text{subject to} \quad \sum_{i=1}^{m} p_i \le p_{\max}$$
$$p_1, \ldots, p_m \in \mathbb{Z}^+ \cup \{0\}$$

For notational simplicity, we have assumed that the per period fixed, variable, and hiring costs are the same in each period. This is not a major restriction, as all of our

subsequent results can be easily extended to the case where α_i, β_i, and γ_i depend on j, provided of course that the variable cost in each period is less than the hiring cost.

3. Preliminary Results

The aim of this section is to show that $E\{C_i(\cdot)\}$ given by (4) is a convex function on $\mathbb{R}$. For simplicity, we will drop the i subscripts and write $E\{C_i(p_i)\}$ as

$$E\{C(p)\} = n\alpha p + \beta \sum_{j=1}^{n} \sum_{k=0}^{N} \theta_{jk} \min(k, p) \\ + \gamma \sum_{j=1}^{n} \sum_{k=0}^{N} \theta_{jk} \max(k - p, 0).$$

Our first result is given below.

Lemma 3.1. *Let $f : [0, \infty) \to \mathbb{R}$ be a function satisfying*

$$f(x) = f(\lfloor x \rfloor) + (x - \lfloor x \rfloor) \cdot \left\{ f(\lfloor x \rfloor + 1) - f(\lfloor x \rfloor) \right\}. \quad (5)$$

Then f is convex if and only if $\{f(\xi) - f(\xi - 1)\}_{\xi=1}^{\infty}$ is a non-decreasing sequence.

Proof. Suppose that f is convex. Then for each integer $\xi \ge 1$,

$$f(\xi) \le \tfrac{1}{2} f(\xi - 1) + \tfrac{1}{2} f(\xi + 1).$$

Thus,

$$f(\xi) - f(\xi - 1) \le f(\xi + 1) - f(\xi).$$

This shows that $\{f(\xi) - f(\xi - 1)\}_{\xi=1}^{\infty}$ is non-decreasing, as required.

To prove the opposite implication, suppose that the sequence $\{f(\xi) - f(\xi - 1)\}_{\xi=1}^{\infty}$ is non-decreasing. Let $x > 0$ and choose $\epsilon > 0$ sufficiently small so that $\lfloor x + \epsilon \rfloor = \lfloor x \rfloor$.

Then by equation (5),

$$f(x + \epsilon) = f(\lfloor x \rfloor) + (x + \epsilon - \lfloor x \rfloor) \cdot \{f(\lfloor x \rfloor + 1) - f(\lfloor x \rfloor)\}$$
$$= f(x) + \epsilon\{f(\lfloor x \rfloor + 1) - f(\lfloor x \rfloor)\}.$$

Hence,

$$\frac{f(x + \epsilon) - f(x)}{\epsilon} = f(\lfloor x \rfloor + 1) - f(\lfloor x \rfloor).$$

Taking the limit as $\epsilon \to 0+$ yields the right-derivative of f:

$$D_+ f(x) := \lim_{\epsilon \to 0+} \frac{f(x + \epsilon) - f(x)}{\epsilon} = f(\lfloor x \rfloor + 1) - f(\lfloor x \rfloor).$$

Now, suppose that $0 \leq y < z$. Then $0 \leq \lfloor y \rfloor \leq \lfloor z \rfloor$, and thus since $\{f(\xi) - f(\xi - 1)\}_{\xi=1}^{\infty}$ is non-decreasing,

$$D_+ f(y) = f(\lfloor y \rfloor + 1) - f(\lfloor y \rfloor)$$
$$\leq f(\lfloor z \rfloor + 1) - f(\lfloor z \rfloor) = D_+ f(z).$$

Hence, $D_+ f$ is non-decreasing. It's also clear from (5) that f is a continuous function. Thus, we have shown that f is continuous, right-differentiable, and its right-derivative is non-decreasing. These three conditions are sufficient for convexity (see Chapter 5 of [9]). $\square$

Equation (5) implies that f in Lemma 3.1 is continuous and piecewise linear. Note also that f can only change slope at the integer points $x = 1, 2, 3, \ldots$

Our goal is to use Lemma 3.1 to prove that $E\{C(\cdot)\}$ is a convex function. To do this, we need to show that $E\{C(\cdot)\}$ satisfies equation (5).

Lemma 3.2. *The expected cost $E\{C(\cdot)\}$ can be written as*

$$E\{C(p)\} = E\{C(\lfloor p \rfloor)\} + (p - \lfloor p \rfloor)$$
$$\times \{E\{C(\lfloor p \rfloor + 1)\} - E\{C(\lfloor p \rfloor)\}\},$$

for all $p \in [0, \infty)$.

Proof. Note that $\{0, \ldots, \min(N, \lfloor p \rfloor)\}$ and $\{\lfloor p \rfloor + 1, \ldots, N\}$ form a partition of $\{0, \ldots, N\}$. Hence, for each $p \in [0, \infty)$,

$$E\{C(p)\} = n\alpha p + \beta \sum_{j=1}^{n} \sum_{k=0}^{N} \theta_{jk} \min(k, p)$$
$$+ \gamma \sum_{j=1}^{n} \sum_{k=0}^{N} \theta_{jk} \max(k - p, 0).$$

Therefore,

$$E\{C(p)\} = n\alpha p + \beta \sum_{j=1}^{n} \sum_{k=0}^{\min(N, \lfloor p \rfloor)} \theta_{jk} k$$
$$+ \beta \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} p \qquad (6)$$
$$+ \gamma \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} (k - p).$$

It follows that

$$E\{C(\lfloor p \rfloor)\} = n\alpha \lfloor p \rfloor + \beta \sum_{j=1}^{n} \sum_{k=0}^{\min(N, \lfloor p \rfloor)} \theta_{jk} k$$
$$+ \beta \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} \lfloor p \rfloor \qquad (7)$$
$$+ \gamma \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} (k - \lfloor p \rfloor).$$

Replacing p in (6) by $\lfloor p \rfloor + p - \lfloor p \rfloor$ gives

$$E\{C(p)\} = n\alpha \lfloor p \rfloor + n\alpha(p - \lfloor p \rfloor) + \beta \sum_{j=1}^{n} \sum_{k=0}^{\min(N, \lfloor p \rfloor)} \theta_{jk} k$$
$$+ \beta \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} \lfloor p \rfloor$$
$$+ \beta \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} (p - \lfloor p \rfloor)$$
$$+ \gamma \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} (k - \lfloor p \rfloor)$$
$$- \gamma \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} (p - \lfloor p \rfloor).$$

Hence, by equation (7),

$$E\{C(p)\} = E\{C(\lfloor p \rfloor)\} + (p - \lfloor p \rfloor)$$
$$\times \left\{ n\alpha + (\beta - \gamma) \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk} \right\}. \qquad (8)$$

Furthermore,

$$E\{C(\lfloor p \rfloor + 1)\} = n\alpha + n\alpha\lfloor p \rfloor$$
$$+ \beta \sum_{j=1}^{n} \sum_{k=0}^{N} \theta_{jk} \min(k, \lfloor p \rfloor + 1)$$
$$+ \gamma \sum_{j=1}^{n} \sum_{k=0}^{N} \theta_{jk} \max(k - \lfloor p \rfloor - 1, 0),$$

from which we have

$$E\{C(\lfloor p \rfloor + 1)\}$$
$$= n\alpha + n\alpha\lfloor p \rfloor + \beta \sum_{j=1}^{n} \sum_{k=0}^{\min(N, \lfloor p \rfloor)} \theta_{jk} k$$
$$+ \beta \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk}(\lfloor p \rfloor + 1)$$
$$+ \gamma \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk}(k - \lfloor p \rfloor - 1)$$
$$= E\{C(\lfloor p \rfloor)\} + n\alpha + (\beta - \gamma) \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk}.$$

Thus,

$$E\{C(\lfloor p \rfloor + 1)\} - E\{C(\lfloor p \rfloor)\}$$
$$= n\alpha + (\beta - \gamma) \sum_{j=1}^{n} \sum_{k=\lfloor p \rfloor + 1}^{N} \theta_{jk}. \quad (9)$$

Combining (8) and (9) completes the proof. □

On the basis of Lemmas 3.1 and 3.2, we now show that $E\{C(\cdot)\}$ is a convex function.

Theorem 3.1. *The expected cost $E\{C(\cdot)\}$ is convex on the interval $[0, \infty)$.*

Proof. It follows from (9) in the proof of Lemma 3.2 that for each integer $\xi \geq 1$,

$$E\{C(\xi+1)\} - E\{C(\xi)\} = n\alpha + (\beta-\gamma) \sum_{j=1}^{n} \sum_{k=\xi+1}^{N} \theta_{jk} \quad (10)$$

and

$$E\{C(\xi)\} - E\{C(\xi-1)\} = n\alpha + (\beta-\gamma) \sum_{j=1}^{n} \sum_{k=\xi}^{N} \theta_{jk}. \quad (11)$$

Recall that the probabilities θ_{jk} are non-negative. Also recall that $\beta < \gamma$. Thus,

$$(\beta - \gamma) \sum_{j=1}^{n} \sum_{k=\xi}^{N} \theta_{jk} \leq (\beta - \gamma) \sum_{j=1}^{n} \sum_{k=\xi+1}^{N} \theta_{jk}.$$

Combining this inequality with (10) and (11) yields

$$E\{C(\xi)\} - E\{C(\xi-1)\} \leq E\{C(\xi+1)\} - E\{C(\xi)\}.$$

The convexity of $E\{C(\cdot)\}$ then follows immediately from Lemmas 3.1 and 3.2. □

4. Solving the Fleet Composition Problem

In this section, we will develop an algorithm for solving the fleet composition problem introduced in Section 2. This algorithm is based on a novel combination of two classical optimization tools—dynamic programming and golden section search.

For fixed $l \in \{1, \ldots, m\}$ and $\xi \in \mathbb{Z}^+ \cup \{0\}$, consider the following subproblem of Problem (2):

$$\begin{aligned} \underset{p_1, \ldots, p_l}{\text{Minimize}} \quad & \sum_{i=1}^{l} E\{C_i(p_i)\} \\ \text{subject to} \quad & \sum_{i=1}^{l} p_i \leq \xi \\ & p_1, \ldots, p_l \in \mathbb{Z}^+ \cup \{0\} \end{aligned} \quad (12)$$

Let $g_l(\xi)$ denote the optimal cost of Problem (12). Then clearly,

$$g_1(\xi) = \min_{p_1 \in \{0, \ldots, \xi\}} E\{C_1(p_1)\}. \quad (13)$$

If $l \geq 2$, then it follows from the well-known *principle of optimality* (see [2]) that

$$g_l(\xi) = \min_{\substack{p_1, \ldots, p_l \in \mathbb{Z}^+ \cup \{0\} \\ p_1 + \cdots + p_l \leq \xi}} \left\{ \sum_{i=1}^{l} E\{C_i(p_i)\} \right\}$$
$$= \min_{p_l \in \{0, \ldots, \xi\}} \min_{\substack{p_1, \ldots, p_{l-1} \in \mathbb{Z}^+ \cup \{0\} \\ p_1 + \cdots + p_{l-1} \leq \xi - p_l}} \left\{ \sum_{i=1}^{l} E\{C_i(p_i)\} \right\}$$
$$= \min_{p_l \in \{0, \ldots, \xi\}} \left\{ E\{C_l(p_l)\} + \min_{\substack{p_1, \ldots, p_{l-1} \in \mathbb{Z}^+ \cup \{0\} \\ p_1 + \cdots + p_{l-1} \leq \xi - p_l}} \sum_{i=1}^{l-1} E\{C_i(p_i)\} \right\}$$
$$= \min_{p_l \in \{0, \ldots, \xi\}} \left\{ E\{C_l(p_l)\} + g_{l-1}(\xi - p_l) \right\}. \quad (14)$$

5

We define $g_0 := 0$. Then equation (14) reduces to equation (13) when $l = 1$.

Starting with $g_0 = 0$, we can use equation (14) with $l = 1$ to calculate $g_1(\xi)$ for each $\xi = 0, \ldots, p_{\max}$. Then we can use (14) with $l = 2$ to calculate $g_2(\xi)$ for each $\xi = 0, \ldots, p_{\max}$. Continuing in this way, we will eventually obtain $g_m(p_{\max})$, the optimal cost of Problem (2). Our algorithm for solving Problem (2) is based on this idea. A key part of our algorithm is an efficient method for performing the minimization on the right-hand side of (14).

We now prove several fundamental results.

Theorem 4.1. *For each integer $l = 0, \ldots, m$, the sequence $\{g_l(\xi) - g_l(\xi - 1)\}_{\xi=1}^{\infty}$ is non-decreasing.*

Proof. Since $g_0 = 0$, the result is clearly true for $l = 0$.

Suppose now that $\{g_l(\xi) - g_l(\xi-1)\}_{\xi=1}^{\infty}$ is non-decreasing for some $l \in \{0, \ldots, m-1\}$. We will show that the sequence $\{g_{l+1}(\xi) - g_{l+1}(\xi - 1)\}_{\xi=1}^{\infty}$ is also non-decreasing.

Let $\xi \geq 1$ be a fixed integer. From the recurrence equation (14), we have

$$g_{l+1}(\xi - 1) = \min_{p \in \{0, \ldots, \xi-1\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi - 1 - p) \right\}.$$

Let $p^* \in \{0, \ldots, \xi - 1\}$ denote the minimizing element on the right-hand side of this equation. Then

$$g_{l+1}(\xi - 1) = E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*).$$

Hence,

$$
\begin{aligned}
g_{l+1}&(\xi + 1) + g_{l+1}(\xi - 1) \\
&= \min_{p \in \{0, \ldots, \xi+1\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi + 1 - p) \right\} \\
&\qquad + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \\
&= \min_{p \in \{0, \ldots, \xi+1\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi + 1 - p) \right. \\
&\qquad \left. + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \right\} \\
&= \min\{\Gamma_1, \Gamma_2, \Gamma_3\}, \quad\quad\quad\quad (15)
\end{aligned}
$$

where

$$
\begin{aligned}
\Gamma_1 &:= \min_{p \in \{0, \ldots, p^*\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi + 1 - p) \right. \\
&\qquad\qquad \left. + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \right\}, \\
\Gamma_2 &:= E\{C_{l+1}(p^* + 1)\} + g_l(\xi - p^*) \\
&\qquad\qquad + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*),
\end{aligned}
$$

$$
\begin{aligned}
\Gamma_3 &:= \min_{p \in \{p^*+2, \ldots, \xi+1\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi + 1 - p) \right. \\
&\qquad\qquad \left. + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \right\}.
\end{aligned}
$$

Now, for each $p = 0, \ldots, p^*$, we have $\xi + 1 - p > \xi - p^*$. Thus, by our inductive hypothesis,

$$g_l(\xi + 1 - p) - g_l(\xi - p) \geq g_l(\xi - p^*) - g_l(\xi - 1 - p^*),$$

for each $p = 0, \ldots, p^*$. Rearranging this inequality gives

$$g_l(\xi + 1 - p) + g_l(\xi - 1 - p^*) \geq g_l(\xi - p) + g_l(\xi - p^*).$$

Thus,

$$
\begin{aligned}
\Gamma_1 &= \min_{p \in \{0, \ldots, p^*\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi + 1 - p) \right. \\
&\qquad\qquad \left. + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \right\} \\
&\geq \min_{p \in \{0, \ldots, p^*\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \right. \\
&\qquad\qquad \left. + E\{C_{l+1}(p^*)\} + g_l(\xi - p^*) \right\} \\
&\geq \min_{p \in \{0, \ldots, \xi\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \right\} \\
&\qquad + \min_{p \in \{0, \ldots, \xi\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \right\} \\
&= 2g_{l+1}(\xi). \quad\quad\quad\quad (16)
\end{aligned}
$$

For Γ_2, we have

$$
\begin{aligned}
\Gamma_2 &= E\{C_{l+1}(p^* + 1)\} + g_l(\xi - p^*) \\
&\qquad + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \\
&= E\{C_{l+1}(p^* + 1)\} + g_l(\xi - (p^* + 1)) \\
&\qquad + E\{C_{l+1}(p^*)\} + g_l(\xi - p^*) \\
&\geq \min_{p \in \{0, \ldots, \xi\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \right\} \\
&\qquad + \min_{p \in \{0, \ldots, \xi\}} \left\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \right\} \\
&= 2g_{l+1}(\xi). \quad\quad\quad\quad (17)
\end{aligned}
$$

Recall from Theorem 3.1 that $E\{C_{l+1}(\cdot)\}$ is convex. Hence, it follows from Lemmas 3.1 and 3.2 that for each integer $p \geq 1$,

$$
\begin{aligned}
E\{C_{l+1}(p)\} &- E\{C_{l+1}(p - 1)\} \\
&\leq E\{C_{l+1}(p + 1)\} - E\{C_{l+1}(p)\}.
\end{aligned}
$$

Thus, if $p \geq p^* + 1$, then

$$
\begin{aligned}
E\{C_{l+1}(p^* + 1)\} &- E\{C_{l+1}(p^*)\} \\
&\leq E\{C_{l+1}(p)\} - E\{C_{l+1}(p-1)\}
\end{aligned}
$$

and so

$$
\begin{aligned}
E\{C_{l+1}(p)\} &+ E\{C_{l+1}(p^*)\} \\
&\geq E\{C_{l+1}(p-1)\} + E\{C_{l+1}(p^*+1)\}.
\end{aligned}
$$

Consequently,

$$
\begin{aligned}
\Gamma_3 = &\min_{p \in \{p^*+2,\ldots,\xi+1\}} \Big\{ E\{C_{l+1}(p)\} + g_l(\xi + 1 - p) \\
&\qquad\qquad + E\{C_{l+1}(p^*)\} + g_l(\xi - 1 - p^*) \Big\} \\
\geq &\min_{p \in \{p^*+2,\ldots,\xi+1\}} \Big\{ E\{C_{l+1}(p-1)\} + g_l(\xi + 1 - p) \\
&\qquad\qquad + E\{C_{l+1}(p^*+1)\} + g_l(\xi - 1 - p^*) \Big\} \\
= &\min_{p \in \{p^*+1,\ldots,\xi\}} \Big\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \\
&\qquad\qquad + E\{C_{l+1}(p^*+1)\} + g_l(\xi - (p^*+1)) \Big\} \\
\geq &\min_{p \in \{0,\ldots,\xi\}} \Big\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \Big\} \\
&\quad + \min_{p \in \{0,\ldots,\xi\}} \Big\{ E\{C_{l+1}(p)\} + g_l(\xi - p) \Big\} \\
= &\, 2g_{l+1}(\xi). \qquad\qquad\qquad\qquad\qquad\qquad (18)
\end{aligned}
$$

Combining inequalities (16)-(18) with (15) gives

$$
g_{l+1}(\xi + 1) + g_{l+1}(\xi - 1) \geq 2g_{l+1}(\xi),
$$

and so

$$
g_{l+1}(\xi) - g_{l+1}(\xi - 1) \leq g_{l+1}(\xi + 1) - g_{l+1}(\xi).
$$

Since $\xi \geq 1$ was chosen arbitrarily, this shows that the sequence $\{g_{l+1}(\xi) - g_{l+1}(\xi-1)\}_{\xi=1}^{\infty}$ is non-decreasing. The result then follows by induction. $\qquad \square$

We have already defined $g_l(\xi)$ for each non-negative integer ξ. We now extend the domain of g_l to $[0, \infty)$ as follows:

$$
g_l(x) = g_l(\lfloor x \rfloor) + (x - \lfloor x \rfloor) \cdot \{ g_l(\lfloor x \rfloor + 1) - g_l(\lfloor x \rfloor)\}, \quad (19)
$$

for each $x \in [0, \infty)$. This extended function is clearly piecewise linear and satisfies condition (5) in Lemma 3.1. We now show that it is also convex.

Theorem 4.2. *For each $l = 0, \ldots, m$, the extended function g_l is convex on $[0, \infty)$.*

Proof. Recall from Theorem 4.1 that $\{g_l(\xi) - g_l(\xi-1)\}_{\xi=1}^{\infty}$ is non-decreasing. Hence, it follows from equation (19) and Lemma 3.1 that g_l is convex. $\qquad \square$

Recall the recurrence equation (14):

$$
g_l(\xi) = \min_{p_l \in \{0,\ldots,\xi\}} \Big\{ E\{C_l(p_l)\} + g_{l-1}(\xi - p_l) \Big\}. \quad (20)
$$

To calculate $g_l(\xi)$ using this equation, we need to solve the minimization problem on the right-hand side. This can be done by simply evaluating $E\{C_l(p_l)\} + g_{l-1}(\xi - p_l)$ at each $p_l = 0, \ldots, \xi$ and then identifying the optimal solution. However, this brute force approach is very inefficient when ξ is large. We will now describe a superior approach.

Suppose that we relax the integer constraint on the right-hand side of equation (20). Then we obtain the following optimization problem:

$$
\begin{aligned}
&\text{Minimize} \quad h_l(p, \xi) := E\{C_l(p)\} + g_{l-1}(\xi - p) \\
&\quad\; p \\
&\text{subject to} \quad 0 \leq p \leq \xi
\end{aligned} \quad (21)
$$

We have the following important result.

Theorem 4.3. *Let $\xi \geq 0$ in Problem (21) be an integer. If p^* is a solution of Problem (21), then $\lfloor p^* \rfloor$ and $\lceil p^* \rceil$ are also solutions of Problem (21).*

Proof. If $p^* \in \mathbb{Z}$, then $p^* = \lfloor p^* \rfloor = \lceil p^* \rceil$ and the result follows immediately. Hence, we assume that $p^* \notin \mathbb{Z}$. This means that $p^* \in (\varsigma - 1, \varsigma)$, where $\varsigma = \lceil p^* \rceil = \lfloor p^* \rfloor + 1$ and $\varsigma - 1 = \lfloor p^* \rfloor$.

By Lemma 3.2, we know that for $p \in [\varsigma - 1, \varsigma]$,

$$
\begin{aligned}
E\{C_l(p)\} = E\{C_l(\varsigma - 1)\} &+ (p - \varsigma + 1) \\
&\times \{E\{C_l(\varsigma)\} - E\{C_l(\varsigma - 1)\}\}. \quad (22)
\end{aligned}
$$

If $p \in (\varsigma - 1, \varsigma)$, then $\xi - \varsigma < \xi - p < \xi - \varsigma + 1$, so $\lfloor \xi - p \rfloor = \xi - \varsigma$ (recall that ξ is an integer). Thus, by equation (19),

$$
\begin{aligned}
g_{l-1}(\xi - p) = g_{l-1}(\xi - \varsigma) &+ (\varsigma - p) \\
&\times \{g_{l-1}(\xi - \varsigma + 1) - g_{l-1}(\xi - \varsigma)\} \quad (23)
\end{aligned}
$$

for $p \in [\varsigma - 1, \varsigma]$. Combining equations (22) and (23) gives

$$
h_l(p, \xi) = E\{C_l(p)\} + g_{l-1}(\xi - p) = \mu p + \omega, \quad p \in [\varsigma - 1, \varsigma],
$$

7

where

$$\mu := E\{C_l(\varsigma)\} - E\{C_l(\varsigma - 1)\} - g_{l-1}(\xi - \varsigma + 1) + g_{l-1}(\xi - \varsigma)$$

and

$$\omega := E\{C_l(\varsigma)\} + g_{l-1}(\xi - \varsigma) + \varsigma \cdot \{E\{C_l(\varsigma - 1)\} \\ - E\{C_l(\varsigma)\} + g_{l-1}(\xi - \varsigma + 1) - g_{l-1}(\xi - \varsigma)\}.$$

This shows that $h_l(\cdot, \xi)$ is a linear function with slope μ on $[\varsigma - 1, \varsigma]$.

Now, since $p^* \in (\varsigma - 1, \varsigma)$ is a minimizer of $h_l(\cdot, \xi)$, we must have $\mu = 0$. Hence,

$$h_l(p^*, \xi) = h_l(\lfloor p^* \rfloor, \xi) = h_l(\lceil p^* \rceil, \xi) = \omega.$$

Since p^* is a solution of Problem (21), this shows that $\lfloor p^* \rfloor$ and $\lceil p^* \rceil$ are also solutions. $\square$

Theorem 4.3 ensures that Problem (21) has an integer solution whenever ξ is an integer. Hence, for each non-negative integer $\xi \geq 0$,

$$g_l(\xi) = \min_{p \in \{0,\ldots,\xi\}} \{E\{C_l(p)\} + g_{l-1}(\xi - p)\} \\ = \min_{p \in \{0,\ldots,\xi\}} h_l(p, \xi) = \min_{p \in [0,\xi]} h_l(p, \xi).$$

It follows that the integer constraint on the right-hand side of equation (20) is redundant, and thus we can compute $g_l(\xi)$ by solving Problem (21).

We now show that $h_l(\cdot, \xi)$ is a convex function. This is an important result that will enable us to solve Problem (21) in an efficient manner.

Theorem 4.4. *For each $l \in \{1, \ldots, m\}$ and $\xi \in \mathbb{Z}^+ \cup \{0\}$, the objective function $h_l(\cdot, \xi)$ in Problem (21) is convex on the interval $[0, \xi]$.*

Proof. Recall from Theorems 3.1 and 4.2 that $E\{C_l(\cdot)\}$ and $g_{l-1}(\cdot)$ are convex. Thus, if $y \in [0, \xi]$ and $z \in [0, \xi]$, then for each $\lambda \in [0, 1]$,

$$E\{C_l(\lambda y + (1 - \lambda)z)\} \leq \lambda E\{C_l(y)\} + (1 - \lambda)E\{C_l(z)\}$$

and

$$g_{l-1}(\xi - \lambda y - (1 - \lambda)z) = g_{l-1}(\lambda(\xi - y) + (1 - \lambda)(\xi - z)) \\ \leq \lambda g_{l-1}(\xi - y) + (1 - \lambda)g_{l-1}(\xi - z).$$

Therefore,

$$h_l(\lambda y + (1 - \lambda)z, \xi) \\ = E\{C_l(\lambda y + (1 - \lambda)z)\} + g_{l-1}(\xi - \lambda y - (1 - \lambda)z) \\ \leq \lambda E\{C_l(y)\} + (1 - \lambda)E\{C_l(z)\} \\ + \lambda g_{l-1}(\xi - y) + (1 - \lambda)g_{l-1}(\xi - z) \\ = \lambda h_l(y, \xi) + (1 - \lambda)h_l(z, \xi).$$

This shows that $h_l(\cdot, \xi)$ is convex on $[0, \xi]$, as required. $\square$

It follows from Theorem 4.4 that Problem (21) is a one-dimensional convex optimization problem. Such problems can be solved efficiently using the *golden section method* (see [1, 8]).

The golden section method works by gradually reducing the *interval of uncertainty*—a known interval that contains at least one optimal solution. This is done by evaluating the cost function at certain *test points* and then exploiting convexity.

Let $\mathcal{I} = [a, b]$ be a given interval of uncertainty for Problem (21) (initially, we have $a = 0$ and $b = \xi$). In the golden section method, we define two test points τ_1 and τ_2 as follows:

$$\tau_1 = b - r|\mathcal{I}| = b - r(b - a)$$

and

$$\tau_2 = a + r|\mathcal{I}| = a + r(b - a),$$

where r is the *golden ratio* defined by

$$r := \frac{\sqrt{5} - 1}{2} \approx 0.618.$$

Note that $\tau_1 < \tau_2$ and $r^2 + r = 1$.

There are two cases to consider:

- Case 1: $h_l(\tau_1, \xi) < h_l(\tau_2, \xi)$

- Case 2: $h_l(\tau_1, \xi) \geq h_l(\tau_2, \xi)$

Recall from Theorem 4.4 that $h_l(\cdot, \xi)$ is convex. Thus, if Case 1 occurs, then any optimal solution of Problem (21) must lie in the interval $[a, \tau_2]$. This means that the interval of uncertainty can be reduced to

$$\mathcal{I}' = [a', b'] = [a, \tau_2].$$

The test points for this new interval of uncertainty are

$$\tau_1' = b' - r|\mathcal{I}'| = \tau_2 - r(\tau_2 - a)$$

8

and

$$\tau_2' = a' + r|\mathcal{I}'| = a + r(\tau_2 - a) = a + r^2(b-a)$$
$$= a + (1-r)(b-a) = b - r(b-a) = \tau_1.$$

Since τ_2' coincides with τ_1, we only need to compute τ_1'.

Now, if Case 2 occurs, then the interval of uncertainty can be reduced to

$$\mathcal{I}' = [a', b'] = [\tau_1, b].$$

The new test points are

$$\tau_1' = b' - r|\mathcal{I}'| = b - r(b - \tau_1) = b - r^2(b-a)$$
$$= b - (1-r)(b-a) = a + r(b-a) = \tau_2$$

and

$$\tau_2' = a' + r|\mathcal{I}'| = \tau_1 + r(b - \tau_1).$$

In this case, τ_1' coincides with τ_2, so we only need to compute τ_2'.

By repeating these steps with $\mathcal{I}$ replaced with $\mathcal{I}'$, we can further reduce the interval of uncertainty. The golden section method keeps iterating in this way until the length of the interval of uncertainty is sufficiently small. The question that immediately arises is: How small is "sufficiently small"? That is, how many golden section iterations are needed to solve Problem (21)? The following result answers this question.

Theorem 4.5. *Let ξ in Problem* (21) *be a positive integer, and let M be any integer such that*

$$M > -\frac{\ln \xi}{\ln r}. \tag{24}$$

Furthermore, let $[a_M, b_M]$ denote the final interval of uncertainty after M golden section iterations are applied to Problem (21). *Then $\lceil a_M \rceil$ is a solution of Problem* (21).

Proof. It's easy to prove that

$$|\mathcal{I}'| = b' - a' = r(b-a).$$

Thus, each golden section iteration reduces the length of the interval of uncertainty by a factor of r. Since the initial interval is $[0, \xi]$,

$$b_M - a_M = r^M \xi.$$

From (24), we obtain

$$r^M < \frac{1}{\xi}.$$

Therefore,

$$b_M - a_M = r^M \xi < 1. \tag{25}$$

Now, let p^* denote a solution of Problem (21) in the interval of uncertainty $[a_M, b_M]$. We know from Theorem 4.3 that both $\lfloor p^* \rfloor$ and $\lceil p^* \rceil$ are also solutions of Problem (21). Thus, to complete the proof, we just need to show that either $\lceil a_M \rceil = \lfloor p^* \rfloor$ or $\lceil a_M \rceil = \lceil p^* \rceil$.

Since $a_M \le p^* \le b_M$, we have

$$\lceil a_M \rceil \le \lceil p^* \rceil.$$

Thus, either $\lceil a_M \rceil = \lceil p^* \rceil$ or $\lceil a_M \rceil < \lceil p^* \rceil$. If $\lceil a_M \rceil = \lceil p^* \rceil$, then the proof is complete. Therefore, we assume that $\lceil a_M \rceil < \lceil p^* \rceil$. By (25),

$$p^* \le b_M < a_M + 1 \le \lceil a_M \rceil + 1 \tag{26}$$

and so

$$\lceil a_M \rceil < \lceil p^* \rceil \le \lceil a_M \rceil + 1.$$

Hence, we must have

$$\lceil p^* \rceil = \lceil a_M \rceil + 1. \tag{27}$$

Now, if p^* is an integer, then

$$p^* = \lceil p^* \rceil = \lceil a_M \rceil + 1,$$

which contradicts (26). Thus, p^* cannot be an integer. This implies that

$$\lceil p^* \rceil = \lfloor p^* \rfloor + 1. \tag{28}$$

Combining (27) and (28) yields $\lceil a_M \rceil = \lfloor p^* \rfloor$. $\qquad\square$

Let

$$\sigma_l(\xi) := \operatorname*{arg\,min}_{p \in \{0,\dots,\xi\}} \left\{ E\{C_l(p)\} + g_{l-1}(\xi - p) \right\}$$
$$= \operatorname*{arg\,min}_{p \in \{0,\dots,\xi\}} h_l(p, \xi).$$

Then clearly,

$$g_l(\xi) = h_l(\sigma(\xi), \xi).$$

The following result is an immediate consequence of Theorems 4.3 and 4.5.

Corollary 4.1. *Let ξ in Problem (21) be a positive integer, and let M be any integer such that*

$$M > -\frac{\ln \xi}{\ln r}.$$

Furthermore, let $[a_M, b_M]$ denote the final interval of uncertainty after M golden section iterations are applied to Problem (21). Then $\sigma_l(\xi) = \lceil a_M \rceil$ and

$$g_l(\xi) = h_l(\lceil a_M \rceil, \xi).$$

Proof. It follows from Theorems 4.3 and 4.5 that

$$g_l(\xi) = \min_{p \in \{0, \ldots, \xi\}} h_l(p, \xi) = \min_{p \in [0, \xi]} h_l(p, \xi) = h_l(\lceil a_M \rceil, \xi).$$

Furthermore,

$$\sigma_l(\xi) = \arg\min_{p \in \{0, \ldots, \xi\}} h_l(p, \xi) = \lceil a_M \rceil.$$

This completes the proof. $\qquad\square$

Let $l \in \{1, \ldots, m\}$ and suppose that $g_{l-1}(\xi)$ is known for each $\xi = 0, \ldots, p_{\max}$. Then the value of g_{l-1} at non-integer points can be computed using equation (19). On the basis of Corollary 4.1, we propose the following algorithm for computing $g_l(\xi)$ and $\sigma_l(\xi)$.

Algorithm 4.1. (Computes $g_l(\xi)$ and $\sigma_l(\xi)$.)

1. If $\xi = 0$, then return $\sigma_l(\xi) = 0$ and

$$g_l(\xi) = h_l(0, 0) = E\{C_l(0)\} + g_{l-1}(0).$$

   Otherwise, go to Step 2.

2. Set
$$M = \left\lfloor -\frac{\ln \xi}{\ln r} \right\rfloor + 1. \tag{29}$$

3. Apply the golden section method to Problem (21) for M iterations. Let $[a_M, b_M]$ denote the final interval of uncertainty.

4. Return $\sigma_l(\xi) = \lceil a_M \rceil$ and

$$g_l(\xi) = h_l(\lceil a_M \rceil, \xi) = E\{C_l(\lceil a_M \rceil)\} + g_{l-1}(\xi - \lceil a_M \rceil).$$

ξ	$M + 2$
10	7
10^2	12
10^3	17
10^4	22
10^5	26

Table 2: When $\xi > 0$, Algorithm 4.1 performs $M + 2$ expected cost evaluations, where M is given by equation (29).

Step 3 of Algorithm 4.1 involves evaluating the cost function $h_l(\cdot, \xi)$ at various test points. Since $h_l(\cdot, \xi)$ depends on g_{l-1}, Algorithm 4.1 will only work correctly if $g_{l-1}(\xi)$ is known for each $\xi = 0, \ldots, p_{\max}$. Thus, Algorithm 4.1 must be applied recursively: first for $l = 1$, then for $l = 2$, and so on until $l = m$.

If $\xi = 0$, then Algorithm 4.1 only needs to compute $h_l(0, 0) = E\{C_l(0)\} + g_{l-1}(0)$. This requires a single evaluation of $E\{C_l(\cdot)\}$ at $p = 0$. In this case, we say that Algorithm 4.1 performs 1 *expected cost evaluation.*

If $\xi > 0$, then Algorithm 4.1 applies M golden section iterations to Problem (21). The first golden section iteration requires two expected cost evaluations, and every subsequent iteration requires one expected cost evaluation (recall that one of the new test points coincides with an old test point). Thus, Algorithm 4.1 will perform $M + 1$ expected cost evaluations in Step 3, and 1 expected cost evaluation in Step 4. This amounts to $M + 2$ expected cost evaluations in total. Note that $M + 2$ is small even when ξ is extremely large; see Table 2.

The next result shows that $g_l(\xi)$ can sometimes be computed without invoking Algorithm 4.1.

Theorem 4.6. *If $g_l(\xi') = g_l(\xi' - 1)$ for some positive integer $\xi' > 0$, then $g_l(\xi) = g_l(\xi')$ and $\sigma_l(\xi) = \sigma_l(\xi')$ for each integer $\xi > \xi'$.*

Proof. Let $l \in \{1, \ldots, m\}$ be fixed. It's clear that replacing ξ with $\xi+1$ in Problem (12) enlarges the feasible region. Thus,

$$g_l(\xi) = \min_{\substack{p_1, \ldots, p_l \in \mathbb{Z}^+ \cup \{0\} \\ p_1 + \cdots + p_l \leq \xi}} \left\{ \sum_{i=1}^{l} E\{C_i(p_i)\} \right\}$$

$$\geq \min_{\substack{p_1, \ldots, p_l \in \mathbb{Z}^+ \cup \{0\} \\ p_1 + \cdots + p_l \leq \xi+1}} \left\{ \sum_{i=1}^{l} E\{C_i(p_i)\} \right\} = g_l(\xi + 1).$$

This shows that the sequence $\{g_l(\xi)\}_{\xi=0}^{\infty}$ is non-increasing.

10

Now, suppose that $g_l(\xi') = g_l(\xi' - 1)$ for some integer $\xi' > 0$. Then

$$g_l(\xi') - g_l(\xi' - 1) = 0. \tag{30}$$

Furthermore, since $\{g_l(\xi)\}_{\xi=0}^{\infty}$ is non-increasing, we know that for each integer $\xi \geq 1$,

$$g_l(\xi) - g_l(\xi - 1) \leq 0. \tag{31}$$

Recall from Theorem 4.1 that $\{g_l(\xi) - g_l(\xi-1)\}_{\xi=1}^{\infty}$ is non-decreasing. Hence, it follows from (30) and (31) that for each integer $\xi > \xi'$,

$$0 = g_l(\xi') - g_l(\xi' - 1) \leq g_l(\xi) - g_l(\xi - 1) \leq 0.$$

Therefore, $g_l(\xi) = g_l(\xi - 1)$. Hence, $g_l(\xi) = g_l(\xi')$ for each integer $\xi > \xi'$. $\square$

Note that $g_m(p_{\max})$ is the optimal cost of Problem (2). Furthermore, by the principle of optimality, it follows that $p_m^* = \sigma_m(p_{\max})$ is the optimal number of type-m vehicles, $p_{m-1}^* = \sigma_{m-1}(p_{\max} - p_m^*)$ is the optimal number of type-$(m-1)$ vehicles, $p_{m-2}^* = \sigma_{m-2}(p_{\max} - p_m^* - p_{m-1}^*)$ is the optimal number of type-$(m-2)$ vehicles, and so on. Thus, computing $g_l(\xi)$ and $\sigma_l(\xi)$ is the key to solving the fleet composition problem.

The algorithm described below is based on this idea. The algorithm starts by invoking Algorithm 4.1 to compute $g_1(\xi)$ and $\sigma_1(\xi)$ for each $\xi = 0, \ldots, p_{\max}$. If at any stage the value of g_1 is the same at two consecutive integers, then the remaining values of g_1 and σ_1 are recorded immediately, as shown in Theorem 4.6. After computing g_1, the algorithm moves on to compute g_l for $l \geq 2$ in a similar manner. Finally, the algorithm backtracks using σ_l to determine the optimal number of vehicles for each vehicle type.

Algorithm 4.2. (Finds an optimal solution $p_1^*, \ldots, p_m^*$ of Problem (2).)

1. If $m > 1$, then set $0 \to \xi$ and $1 \to l$. Otherwise, set $p_{\max} \to \xi$ and $1 \to l$.

2. Use Algorithm 4.1 to compute $g_l(\xi)$ and $\sigma_l(\xi)$.

3. If $0 < \xi < p_{\max}$ and $g_l(\xi) = g_l(\xi - 1)$, then for each integer $\kappa = \xi + 1, \ldots, p_{\max}$, set $g_l(\xi) \to g_l(\kappa)$ and $\sigma_l(\xi) \to \sigma_l(\kappa)$ before going to Step 5. Otherwise, go to Step 4.

4. If $\xi < p_{\max}$, then set $\xi + 1 \to \xi$ and return to Step 2. Otherwise, go to Step 5.

5. If $l \leq m - 2$, then set $0 \to \xi$ and $l + 1 \to l$ and return to Step 2. Otherwise, if $l = m - 1$, then set $p_{\max} \to \xi$ and $l + 1 \to l$ and return to Step 2. Otherwise, go to Step 6.

6. For each $l = m, \ldots, 1$, compute

$$\xi_l = p_{\max} - \sum_{\kappa=l+1}^{m} p_\kappa^*$$

and

$$p_l^* = \sigma_l(\xi_l).$$

7. Stop: $p_1^*, \ldots, p_m^*$ is a solution of Problem (2).

In Step 2 of Algorithm 4.2, we call Algorithm 4.1 to compute $g_l(\xi)$ and $\sigma_l(\xi)$. This step is performed for every $l = 1, \ldots, m$ and potentially for every $\xi = 0, \ldots, p_{\max}$. Recall that Algorithm 4.1 performs 1 expected cost evaluation if $\xi = 0$, and $M + 2$ expected cost evaluations if $\xi > 0$, where M is defined by equation (29). Hence, an upper bound for the total number of expected cost evaluations in Algorithm 4.2 is

$$\bar{M} := m\left[1 + \sum_{\xi=1}^{p_{\max}}(M + 2)\right]$$

$$= m + 3mp_{\max} + m\sum_{\xi=1}^{p_{\max}} \left\lfloor -\frac{\ln \xi}{\ln r} \right\rfloor. \tag{32}$$

As we will see in the next section, Algorithm 4.2 is far more efficient than the brute force approach of evaluating the cost function at every feasible point.

5. Numerical Results and Discussion

In this section, we present some numerical results from using Algorithm 4.2 to solve various classes of random test problems. We first consider problems in which the vehicle requirements are deterministic; in this case, both Algorithm 4.2 and the algorithm developed in [7] are applicable. We compare the performance of these two algorithms on different classes of deterministic problems. We then consider more realistic problems in which the vehicle requirements follow a binomial distribution. Such problems cannot be solved using the algorithm in [7]. The results in

this section were produced on a MacBook Pro with 2.2GHz Intel i7 CPU.

5.1. Deterministic Vehicle Requirements

Suppose that q_{ij}, the number of type-i vehicles required during period j, is known. Then for each $i = 1, \dots, m$ and $j = 1, \dots, n$,

$$\theta_{ijk} = P(q_{ij} = k) = \begin{cases} 1, & \text{if } k = q_{ij}, \\ 0, & \text{if } k \neq q_{ij}. \end{cases}$$

In this case, Problem (2) is identical to the *deterministic fleet composition problem* considered in reference [7]. Thus, the algorithm developed in [7] is applicable.

We wrote a Fortran program that applies Algorithm 4.2 and the algorithm in [7] to the same set of test problems. The test problems are created by randomly choosing the costs and vehicle demands as follows:

$$\alpha_i \in [0, 100],$$
$$\beta_i \in [0, 100],$$
$$\gamma_i \in (\alpha_i + \beta_i, 2\alpha_i + 2\beta_i],$$
$$q_{ij} \in \{0, \dots, p_{\max}\}.$$

Note that the problem dimensions m, n, and $p_{\max}$ are fixed during each run of the program. Choosing γ_i to be greater than $\alpha_i + \beta_i$ ensures that the cost of operating an owned vehicle for one period is less than the cost of hiring the same vehicle for one period—if this doesn't hold, then the optimal solution of the fleet composition problem is $p_1^* = \cdots = p_m^* = 0$.

We ran the program with 10 different sets of problem dimensions. During each run, the program generated and solved 1000 random problems. In all cases, the optimal cost from Algorithm 4.2 matched the optimal cost from the old algorithm in [7]. The average number of expected cost evaluations (AE) used by each algorithm is shown in Table 3. Notice that, on average, Algorithm 4.2 uses about 75% fewer cost evaluations than the old algorithm. Thus, Algorithm 4.2 is a major improvement.

5.2. Stochastic Vehicle Requirements: Small Problems

We now consider problems in which the vehicle requirements are stochastic. For these problems, the algorithm

in [7] is not applicable. Thus, we will instead compare Algorithm 4.2 to the brute force approach of evaluating the cost function at every feasible point.

We suppose that the vehicle requirements q_{ij} follow a binomial distribution:

$$\theta_{ijk} = P(q_{ij} = k) = \binom{p_{\max}}{k} \rho_{ij}^k (1 - \rho_{ij})^{p_{\max}-k}, \quad (33)$$

where $k \in \{0, \dots, p_{\max}\}$ and $\rho_{ij} \in [0, 1]$ is a given parameter.

For numerical testing, we wrote a Fortran program that uses Algorithm 4.2 to solve a series of test problems with binomial vehicle requirements (in each problem, q_{ij} is distributed according to (33)). The program uses brute force to verify the solution obtained by Algorithm 4.2. The problem dimensions m, n, and $p_{\max}$ are fixed during each run of the program, but the other parameters $\alpha_i \in [0, 100]$, $\beta_i \in [0, 100]$, $\gamma_i \in (\alpha_i + \beta_i, 2\alpha_i + 2\beta_i]$, and $\rho_{ij} \in [0, 1]$ are chosen randomly to generate the different test problems. Since $p_{\max}$ is fixed, each random problem generated by the program has the same number of feasible points (FP). Hence, the total number of expected cost evaluations needed for brute force verification is

$$\text{AE (BF)} = \text{FP} \times m.$$

We ran our program with 10 different sets of "small" problem dimensions (brute force is only practical for small problems). During each run, the program generated and solved 1000 random problems. For each problem, the solution obtained by Algorithm 4.2 agreed with the solution obtained by brute force. Our numerical results are summarized in Table 4, where $\bar{M}$ is the upper bound defined by equation (32) and AE is rounded up to the nearest integer. Note that Algorithm 4.2 easily outperforms brute force, as expected.

5.3. Stochastic Vehicle Requirements: Large Problems

Problems of small dimension can be easily solved using brute force. However, the brute force approach quickly becomes impractical when the problem dimensions increase. To demonstrate Algorithm 4.2's performance on large-scale problems, we disabled the brute force verification procedure in our program, and then ran the modified program with 10 sets of "large" problem dimensions. Each run of the program generated and solved 100 test problems. As

Dimensions						
m	n	$p_{\max}$	AE (Alg. 4.2)	AE (ref. [7])	AE (Alg. 4.2)/AE (ref. [7])	
5	50	50	1,407	4,025	34.96%	
5	50	100	3,260	15,526	21.00%	
10	50	50	3,578	10,655	33.58%	
10	50	100	8,301	41,281	20.11%	
15	50	50	5,749	17,285	33.26%	
15	50	100	13,344	67,036	19.91%	
20	50	50	7,918	23,915	33.11%	
20	50	100	18,382	92,791	19.81%	
25	50	50	10,089	30,545	33.03%	
25	50	100	23,423	118,546	19.76%	

Table 3: Comparing Algorithm 4.2 with the old algorithm in [7]. Note that AE is rounded up to the nearest integer.

Dimensions							
m	n	$p_{\max}$	AE (Alg. 4.2)	$\bar{M}$	FP	AE (BF)	AE (Alg. 4.2)/AE (BF)
1	20	30	10	231	31	31	32.2581%
1	50	50	11	435	51	51	21.5686%
2	20	30	112	462	496	992	11.2903%
2	50	50	199	870	1,326	2,652	7.5038%
3	20	30	305	693	5,456	16,368	1.8634%
3	50	50	565	1,305	23,426	70,278	0.8040%
4	20	30	534	924	46,376	185,504	0.2879%
4	50	50	988	1,740	316,251	1,265,004	0.0781%
5	20	30	761	1,155	324,632	1,623,160	0.0469%
5	50	50	1,418	2,175	3,478,761	17,393,805	0.0082%

Table 4: Numerical results from using Algorithm 4.2 to solve 1000 small-scale fleet composition problems. Note that AE is rounded up to the nearest integer.

before, the test problems were created by randomly selecting the parameters α_i, β_i, γ_i, and ρ_{ij}. Table 5 gives the average computation time (ACT) for Algorithm 4.2. The algorithm's efficiency is clearly evident here: on average, it takes just 13 seconds to solve a fleet composition problem with $m = 200$. Solving the same problem using brute force would require days of computation.

References

[1] Bazaraa MS, Sherali HD, Shetty CM. Nonlinear Programming: Theory and Algorithms. 3rd ed. New Jersey: John Wiley; 2006.

[2] Bellman R. Dynamic Programming. Dover ed. New York: Dover Publications; 2003.

[3] Ghiani G, Laporte G, Musmanno R. Introduction to Logistics Systems Planning and Control. Chichester: John Wiley; 2004.

[4] Gould J. The size and composition of a road transport fleet. Operational Research Quarterly 1969; 20:81-92.

Dimensions					
m	n	$p_{\max}$	AE	$\bar{M}$	ACT (secs)
10	100	50	3,575	4,350	0.0325
10	200	100	8,292	10,090	0.2906
25	100	50	10,110	10,875	0.0897
25	200	100	23,491	25,225	0.8474
50	100	50	20,950	21,570	0.1962
50	200	100	48,615	50,450	1.9515
100	100	50	42,627	43,500	0.4412
100	200	100	99,027	100,900	3.9658
200	100	50	86,055	87,000	1.5289
200	200	100	199,886	201,800	13.7035

Table 5: Average computation time for Algorithm 4.2. Note that AE is rounded up to the nearest integer.

[5] Hoff A, Andersson H, Christiansen M, Hasle G, Løkketangen A. Industrial aspects and literature survey: Fleet composition and routing. Computers & Operations Research 2010; 37:2041-2061.

[6] Kirby D. Is your fleet the right size? Operational Research Quarterly 1959; 10:252.

[7] Loxton R, Lin Q. Optimal fleet composition via dynamic programming and golden section search. Journal of Industrial & Management Optimization 2011; 7:875-890.

[8] Luenberger DG, Ye Y. Linear and Nonlinear Programming. 3rd ed. New York: Springer; 2008.

[9] Royden HL. Real Analysis. 3rd ed. New Jersey: Prentice Hall; 1988.

[10] Salhi S, Rand GK. Incorporating vehicle routing into the fleet composition problem. European Journal of Operational Research 1993; 66:313-330.

[11] Simms BW, Lamarre BG, Jardine AKS, Boudreau A. Optimal buy, operate and sell policies for fleets of vehicles. European Journal of Operational Research 1984; 15:183-195.