Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL
Pacific Journal of Optimization
Vol. 10, No. 4, pp. 809–829, October 2014

AN EFFICIENT HEURISTIC FOR TRANSPORT FLEET MAINTENANCE SCHEDULING IN A MAINTENANCE SYSTEM WITH LIMITED CAPACITY

Ming-Jong Yao and Jia-Yen Huang*

Abstract: In the real world, the managers of a transportation fleet need to carefully determine the maintenance cycles of vehicles so as to minimize the average total costs (including the operating costs and the maintenance costs). Furthermore, since the maintenance system is usually equipped with limited manpower and facilities, it has limited capacity to provide maintenance service for the transportation fleet. Therefore, we are motivated to determine the optimal maintenance cycle of vehicles and to generate a feasible maintenance schedule of a transport fleet in this study. To solve this problem, we first obtain candidate solutions by solving an unconstrained model using a search algorithm. Then, we test feasibility for each one of these candidate solutions. Finally, among these candidate solutions, we pick the 'best' solution that secures a feasible maintenance schedule with the minimum average costs. Based on our random experiments, we demonstrate that the proposed search algorithm could effectively solve the Transportation Fleet Maintenance Scheduling Problem in a maintenance system with limited capacity.

Key words: *transport fleet, maintenance, capacity-constrained, scheduling*

Mathematics Subject Classification: *90B06, 90B35*

1 Introduction

In the past decade, logistics service providers have generally experienced low profit margins due to the intensive competition that exists in the industry and prices skyrocketing of the crude oil. It is necessary to have a fleet maintenance scheduled economically. A well planned maintenance schedule can not only bring down the cost but also raise the utilization rate of a transportation fleet.

The mathematical model for the Transport Fleet Maintenance Scheduling Problem (without considering maintenance capacity) was proposed by Goyal et al. [4], we name this problem as the *TFMSP*. The problem of determining the economic maintenance of a machine has been dealt with extensively in management science, operations research, and industrial engineering (see [1–3, 5, 7, 8, 12, 14]). But researchers pay limited attention to the problem of determining the operating and maintenance schedules for a transport fleet. Some of the researches regarding maintenance scheduling problem of aircraft fleet have been studied for many years, but the constraints are completely different from the *TFMSP* due to its special characteristics such as heterogeneous fleet of aircraft, the routine inspection regulated by Federal Aviation Administration, the consideration of flight hours and number of take-off

*Corresponding author

and landing cycles (Sriram et al. [11]). Moreover, the aircraft maintenance scheduling problem concerns more about assignment (Moudani et al. [9]), maintenance routing and crew scheduling (Papadakos [10]), which is different from the *TFMSP*.

Goyal et al. [4] algorithm is based on two equations that are derived by setting the first derivative of the objective function with respect to the decision variables. Later, Egmond et al. [13] indicate that the function of the *TFMSP* is not convex as [4] assumed. They also indicate that Goyal and Gunasekaran's search procedure often stops after its first iteration without obtaining an optimal solution. In fact, it is often stuck in a local optimal solution. However, they only suggest trying different starting values to find an optimal solution, but without proposing a new solution approach. Yao and Huang [18] show that the optimal objective function value of the *TFMSP* is piece-wise convex with respect to T (some decision maker's planning basic period). They also propose a search algorithm that solves the optimal solution for the *TFMSP*. An extended version of the *TFMSP* is the Transport Fleet Maintenance Scheduling Problem for a Logistic Service Provider with many sub-companies, which was studied in [6]. They show that a Logistic Service Provider can enjoy significant cost savings from coordinating the maintenance policy of the transport fleets among sub-companies.

The above studies all assume that the maintenance system has *unlimited* capacity as providing maintenance service for the transportation fleet. However, it is a common practice that the capacity of a maintenance system is usually constrained due to limited manpower and facilities. Therefore, we were motivated to investigate the optimal maintenance schedule for a transport fleet in a capacity-constrained maintenance system so as to minimize the average total costs in this paper. In order to distinguish this problem from the previous studies, we name this problem the "Capacity Constrained Transport Fleet Maintenance Scheduling Problem", which is abbreviated as the *CC-TFMSP*.

This paper is organized as follows. We will present the mathematical model for the *CC-TFMSP* in Section 2. Section 3 presents theoretical analysis on the optimal cost curve of the *unconstrained* problem. Based on our theoretical background, we propose an effective heuristic for obtaining candidate solutions and a procedure to generate a feasible maintenance schedule in Section 4. Next, Section 5 presents a numerical example and random experiments for illustrating the implementation and verifying the effectiveness of the proposed search algorithm, respectively. Finally, we addressed our concluding remarks.

2 The Mathematical Model

We first introduce the assumptions made and the notation used later. There are m groups of vehicles, and the number of vehicles in group i is denoted as n_i. In the *CC-TFMSP*, the decision maker plans the maintenance schedules of the vehicle groups in some *basic period*, denoted by T, (*e.g.,* in days, weeks, or bi-weeks, *etc.*). The maintenance work on a group of vehicles is carried out at a fixed, equal-time interval that is called the "maintenance cycle" for that group of vehicles. The vehicles in the i^{th} group are sent for maintenance once in k_i basic periods where k_i is a positive integer. Therefore, the maintenance cycle for the vehicles in the i^{th} group is k_iT. It should be noted that the model for the *CC-TFMSP* is for planned maintenance, and the model does not consider unplanned fleet vehicle failure in the scheduling of fleets. (The proposed model may accommodate unplanned vehicle failures by reserving some portion of maintenance capacity. We will have further discussion later.)

We consider two categories of costs in the *CC-TFMSP*, namely, the operating cost and the maintenance cost. The operating cost of a vehicle depends on the length of the maintenance cycle, and it is assumed to increase linearly with respect to time since the last maintenance on the vehicle. Specifically, the operating cost per unit of time at time t after the last maintenance for a vehicle in group i is given by $f_i(t) = a_i + b_i t$ where a_i is the fixed cost and b_i indicates the increase in the operating cost per unit of time. Also, for each vehicle in group i, it takes X_i units of time for its maintenance work. And the utilization factor of a vehicle in the i^{th} group on the road is Y_i, which is a known constant. Further discussions on the utilization factor of a vehicle can be referred to [15]. Therefore, the actual time during which a vehicle can operate is equal to $Y_i(k_i T - X_i)$, and the total operating cost for a vehicle in group i is given by

$$\begin{aligned} &\int_0^{Y_i(k_iT-X_i)} f_i(t)dt = \int_0^{Y_i(k_iT-X_i)} (a_i + b_i t)dt \\ &= Y_i(a_i - b_i X_i Y_i)k_i T + 0.5 b_i Y_i^2 k_i^2 T^2 - X_i Y_i (a_i - 0.5 b_i X_i Y_i) \end{aligned} \tag{2.1}$$

When a vehicle of the i^{th} group is sent to maintain, it takes τ_i and π_i constant time units for its setup and maintenance, respectively. Therefore, it holds that

$$X_i = \tau_i + \pi_i \tag{2.2}$$

In this study, we assume that there is only one maintenance facility, and the maintenance facility could maintain only one group of vehicle at one time. (We did not specify the number of servers working in the maintenance facility in this study. Obviously, changing the number of servers in the maintenance facility may affect the setup time and the maintenance time required for each vehicle group. The decision maker should adjust the values of τ_i and π_i carefully accordingly as adding or reducing the number of servers.)

We define P_t as the number of vehicle groups maintained during the t^{th} basic period. Also, we use $\sigma_{t(i)}$ to indicate the i^{th} group of vehicle at basic period t. Here we take a small example in which a maintenance schedule has a cycle of 8 basic periods. We assume the vector of time multipliers $(k_1, k_2, k_3, k_4) = (2, 2, 4, 8)$, the number of vehicle groups assigned to the 8 basic periods (i.e., corresponding to P_1 to P_8) is $(2, 2, 1, 1, 2, 1, 1, 1)$. Figure 1 shows a maintenance schedule $\langle\{1, 3\}, \{2, 4\}, \{1\}, \{2\}, \{1, 3\}, \{2\}, \{1\}, \{2\}\rangle$. We denote $\tau_{\sigma_t(i)}$ and $\pi_{\sigma_t(i)}$ as the setup time and the maintenance time of i^{th} group in P_t, respectively. After its set up, the first group of vehicle starts its maintenance. As soon as the first group is done, the second group shall start its setup and maintenance tasks.

One may reserve maintenance capacity for taking care of unplanned vehicle failures. An easy way is to include the maintenance of the "virtual" vehicle group by setting $k_i = 1$ to reserve the required maintenance capacity (for unplanned vehicle failures) in each basic period accordingly. Also, the decision maker may refer to the historical data to determine the reserved maintenance capacity in each basic period.

The fixed cost of starting the maintenance for a vehicle group i (fixed cost term as the sum of setup cost of vehicle group i over all stages) is given by s_i. On the other hand, as maintenance work is carried out at intervals of T, a fixed cost, denoted by S, will be incurred for all vehicle groups scheduled for maintenance in each basic period.

The objective function of the *CC-TFMSP* is to minimize the average total costs incurred per unit of time. Therefore, we divide the cost terms of vehicle group by its cycle time respectively to obtain their corresponding terms in the objective function. By the derivation above, the mathematical model for the *CC-TFMSP* can be expressed as problem (P).

$$(P)\text{Min} Z\left((k_1, k_2, \cdots, k_m), T\right) = \frac{S}{T} + \sum_{i=1}^{m} \frac{n_i z_i(k_i, T)}{k_i T} = \frac{S}{T} + \sum_{i=1}^{m} \Phi_i(k_i, T) + u \tag{2.3}$$

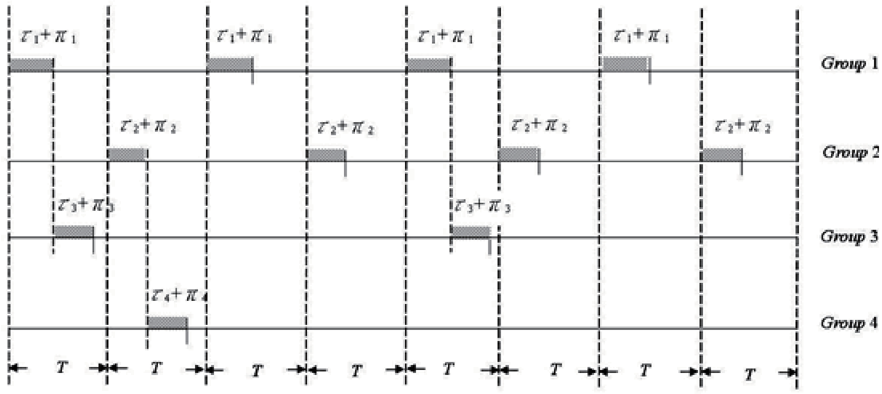

Figure 1: An example of a maintenance schedule with 4 vehicle groups

Subject to

$$\sum_{i=1}^{m}(\tau_i+\pi_i)w_{i\varphi(i,t)} \leq T, for\ t=1,...,K \tag{2.4}$$

$$\sum_{t=1}^{k_i} w_{it}=1, for\ i=1,...,m \tag{2.5}$$

$$\varphi(i,t)=\begin{cases} t \bmod k_i, & \text{if } t\neq\gamma k_i, \gamma\in\mathbb{N}^+ \\ k_i, & \text{if } t=\gamma k_i, \gamma\in\mathbb{N}^+ \end{cases} \quad \text{for } i=1,\cdots,m, t=1,\cdots,K \tag{2.6}$$

where

$$\begin{aligned} &z_i(k_i,T)=Y_i(a_i-b_iX_iY_i)k_iT+0.5b_iY_i^2k_i^2T^2-X_iY_i(a_i-0.5b_iX_iY_i),\\ &X_i=\tau_i+\pi_i,\\ &\Phi_i(k_i,T)=\frac{n_iC_{1i}}{k_iT}+n_iC_{2i}k_iT,\\ &C_{1i}=s_i-X_iY_i(a_i-0.5b_iX_iY_i)\\ &C_{2i}=0.5b_iY_i^2, u=\sum_{i=1}^{m}n_iY_i(a_i-b_iX_iY_i),\\ &K=lcm(k_1,...,k_m)\\ &w_{it}=\begin{cases} 1, & \text{if group } i \text{ is maintenanced in } i^{th} \text{ basic period} \\ 0, & \text{Otherwise.} \end{cases} \end{aligned}$$

Recall that whenever group i appears in a basic period, it takes $\tau_i+\pi_i$ for its maintenance run. Inequalities in (2.4) mandate the time needed for the vehicle groups assigned to the same basic period must not exceed T for each basic period t in the maintenance schedule. Eq. (2.5) meet the assumption that the maintenance runs for group i must perform exactly once during the horizon k_i, and the first maintenance run is no later than the $(k_i)^{th}$ basic period. Eq. (2.6) assure the maintenance of group i repeats after k_iT.

Note that the decision variables of the proposed model are $(k_1,k_2,\ldots,k_m)$ and T. Each k_i is a positive integer. The number of constraints, i.e., $K=lcm\,(k_1,k_2,\ldots,k_m)$, in eq. (2.4) and the number of endogenous variables w_{it} are unknown *a priori* since both of them depend on the values of k_i, that are unknown to the decision maker. To the best of our knowledge, any commercial software is NOT able to solve the proposed mathematical model.

3 Theoretical Results on the Unconstrained Models

In the *CC-TFMSP*, the decision maker needs to determine T (*i.e.,* the basic period) and $k_1, k_2, \ldots, k_m$ (*i.e.,* the frequency of maintenance for vehicles in each group) so as to minimize the total costs incurred per unit of time.

In this study, we would solve the *CC-TFMSP* using two policies, namely, *General Integer (GI)* policy and *Power-of-two (PoT)* policy. *GI* policy requires that all the k_i's must be positive integers. On the other hand, *PoT* policy restricts all the k_i's to be the powers-of-two integers (i.e., $k_i = 1, 2, 4, ..., 2^p; p \in \mathbb{N}$). Note that *PoT* policy enjoys an interesting and important property, *viz.,* $K = lcm(k_1, ..., k_m) = \max(k_1, ..., k_m)$, which usually significantly reduces the number of constraints in (2.4), especially, comparing to those cases in which several k_i's are prime numbers. Therefore, *PoT* policy simplifies the complexity and computational loading for generating a feasible maintenance schedule. We will compare the effectiveness of both policies in Section 5.

We will present our discussions on the (unconstrained) *TFMSP* under *GI* and *PoT* policies in this section first. Theoretical properties on the (unconstrained) *TFMSP* not only provide insights into the optimal cost function, but also facilitate the derivation of the proposed search algorithm for solving the *CC-TFMSP* in Section 4.

3.1 The unconstrained TFMSP model under GI policy

We investigate an unconstrained version of the problem (P) by ignoring the constraints in (2.4)-(2.6). Also, since u is a constant, we omit it in our theoretical analysis here. The unconstrained *TFMSP* model under *GI* policy may be expressed as (P') as follows.

$$(P') \underset{k_i \in \mathbb{N}^+, \forall i, T \in \mathbb{R}^+}{\text{Minimize}} \; Z^{'}\left((k_1, k_2, \ldots, k_m), T\right) = S/T + \sum_{i=1}^{m} \Phi_i(k_i, T) \tag{3.1}$$

We note that the authors had examined the theoretical properties of (P') in their previous study, *viz.*, Yao and Huang [18]. We review some key results in the following presentation. Therefore, we are motivated to study the properties of $\Phi_i(k_i, T)$ since they shall establish foundation for our further investigation on the function $Z^{'}\left((k_1, k_2, \ldots, k_m), T\right)$.

Recall that $\Phi_i(k_i, T) = n_i C_{1i}/k_i T + n_i C_{2i} k_i T$. By observing the right-side of (3.1), the terms are separable. Obviously, for any given $k_i \in \mathbb{N}^+$, the function $\Phi_i(k_i, T)$ is strictly convex. (See also Yao and Huang [18], Proposition 3.1, pp. 36)

Define a new function $g_i(T)$ by taking the optimal value of k_i at any value $T' > 0$ for the function $\Phi_i(k_i, T)$ as follows.

$$g_i(T) := \inf_{k_i \in N^+} \{\Phi_i(k_i, T') | T = T' \in \mathbb{R}^+\} \tag{3.2}$$

Note that the curve of the $g_i(T)$ function is actually the lower envelope of the $\Phi_i(k_i, T)$ functions.

Next, we define a "junction point" for $g_i(T)$ as a particular value of T where two consecutive convex curves $\Phi_i(k_i, T)$ and $\Phi_i(k_i+1, T)$ concatenate. The junction point corresponding to $k_i = k$ locates at

$$\delta_i(k) = \sqrt{\frac{C_{1i}}{C_{2i}(k+1)k}} = \sqrt{\frac{2(s_i - X_i Y_i(a_i - 0.5 b_i X_i Y_i))}{b_i Y_i^2 (k+1)k}} \tag{3.3}$$

Importantly, two interesting observations on the $g_i(T)$ function at a junction point w are as follows: (See also Yao and Huang [18], pp. 37)

(a) The function $g_i(T)$ is *piece-wise convex* with respect to T.

(b) Suppose that $k^*(w^-)$ and $k^*(w^+)$, respectively, are the optimal multipliers of the left-side and right-side convex curves with regard to a junction point w of the $g_i(T)$ function. Then, $k^*(w^-) = k^*(w^+) + 1$, where $w^- = w - \varepsilon, w^+ = w + \varepsilon$ and $\varepsilon \to 0^+$.

Following (3.3), it obviously holds that

$$\delta_i(v_i) < \cdots < \delta_i(k+1) < \delta_i(k) < \cdots < \delta_i(2) < \delta_i(1) \tag{3.4}$$

where v_i is an (unknown) upper bound on the value of k_i. Theorem 3.1 is an immediate result from (3.3) and (3.4).

Theorem 3.1 (Yao and Huang [18], Theorem 3.1, pp. 37)**.** *Suppose that $k^*(w^-)$ and $k^*(w^+)$ are the optimal multipliers of the left-side and right-side convex curves with regard to a junction point w of the $g_i(T)$ function, then $k^*(w^-) = k^*(w^+) + 1$.*

The following corollary provides an easy way to obtain the optimal multiplier for the $g_i(T)$ function for any given $T > 0$.

Corollary 3.2 (Yao and Huang [18], Corollary 3.1, pp. 37)**.** *For any given $T > 0$, the optimal multiplier $k_i^*(T) \in \mathbb{N}^+$ for the $g_i(T)$ function is given by*

$$k_i^*(T) = \left\lceil -\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{4C_{1i}}{C_{2i}T^2}} \right\rceil \tag{3.5}$$

with $\lceil . \rceil$ denoting the upper-entier function.

Denote the function $TC_{GI}^U(T)$ as the best unconstrained objective function value of the *CC-TFMSP* under *GI* policy. Then, $TC_{GI}^U(T)$ is actually the sum of the minimum cost function of the m groups of vehicles, which can be re-written as

$$(U - GI)\ TC_{GI}^U(T) = \inf_{T>0} \{S/T + \sum_{i=1}^{m} g_i(T)\} \tag{3.6}$$

Following the theoretical analyses on the $\Phi_i(k_i, T)$ and $g_i(T)$ functions, one can gain more insights into the $TC_{GI}^U(T)$ as follows.

Proposition 3.3 (Yao and Huang [18], Proposition 3.2, pp. 38)**.** *The $TC_{GI}^U(T)$ function is piece-wise convex with respect to T.*

Proposition 3.4 (Yao and Huang [18], Proposition 3.3, pp. 38)**.** *All the junction points of $g_i(T)$ function of each group i will be inherited by the $TC_{GI}^U(T)$ function. In other words, if w is a junction point for a group i, w must also show as a junction point on the piece-wise convex curve of the $TC_{GI}^U(T)$ function.*

To make our notation simpler, we define $\mathbf{k}_{GI}^U(T)$ as the vector of the *unconstrained* optimal multipliers at T. Theorem 3.5 is an immediate result of Theorem 3.1 and Proposition 3.4.

Theorem 3.5 (Yao and Huang [18], Theorem 3.2, pp. 38)**.** *Suppose that $\mathbf{k}_{GI}^U(w^-)$ and $\mathbf{k}_{GI}^U(w^+)$, respectively, are the set of optimal multipliers for the left-side and right-side convex curves with regard to a junction point w in the plot of the $TC_{GI}^U(T)$ function. Then, $\mathbf{k}_{GI}^U(w^-)$ can be secured from $\mathbf{k}_{GI}^U(w^+)$ by changing at least one of k_i from $k_i^*(w^-)$ to $k_i^*(w^+) = k_i^*(w^-) + 1$.*

3.2 The unconstrained TFMSP model under PoT policy

Next, we investigate the unconstrained *TFMSP* model under *PoT* policy. Its mathematical model may be expressed as (P") as follows.

$$(P'') \underset{k_i=2^{p_i}, p_i \in \mathbb{N}, \forall i, T \in \mathbb{R}^+}{\text{Minimize}} \ Z^{''}\left((k_1, k_2, \ldots, k_m), T\right) = S/T + \sum_{i=1}^{m} \Phi_i(k_i, T) \tag{3.7}$$

To the best of the authors' knowledge, the following theoretical results on the unconstrained *TFMSP* model under *PoT* policy were not investigated in the literature before this study. We will employ different notation to distinguish.

It can be easily shown that for any power-of-two integer k_i, the function $\Phi_i(k_i, T)$ is strictly convex. Similar to the function $g_i(T)$, we define another function $h_i(T)$ by taking the optimal power-of-two k_i at any $T' > 0$ as follows.

$$h_i(T) := \inf_{k_i=2^{p_i}, p_i \in \mathbb{N}, \forall i} \{\Phi_i(k_i, T') | T = T' \in \mathbb{R}^+\} \tag{3.8}$$

We may also find a "junction point" for $h_i(T)$ where two consecutive convex curves $\Phi_i(k_i, T)$ and $\Phi_i(2k_i, T)$ concatenate. These junction points determine at "what value of T" where one should change the value of k_i so as to obtain the optimal value for the $h_i(T)$ function. To derive a closed-form for the location of the junction points, we define the difference function $\Delta_i(k, T)$ by

$$\begin{aligned} \Delta_i(k, T) &= \Phi_i(2k, T) - \Phi_i(k, T) \\ &= \frac{n_i C_{1i}}{(2k)T} + n_i C_{2i}(2k)T - \frac{n_i C_{1i}}{kT} - n_i C_{2i} kT = -\frac{n_i C_{1i}}{2kT} + n_i C_{2i} kT \end{aligned}$$

where $k_i = 2^j$; $j \in \aleph$. By letting $\Delta_i(k, T) = 0$, the junction point corresponding to $k_i = 2^j$ locates at

$$\delta_i(k) = \frac{1}{k}\sqrt{\frac{C_{1i}}{2C_{2i}}} = \frac{1}{k}\sqrt{\frac{s_i - X_i Y_i (a_i - 0.5 b_i X_i Y_i)}{b_i Y_i^2}} \tag{3.9}$$

Similarly, the $h_i(T)$ function also has two interesting observations at any junction point w.

(a) The function $h_i(T)$ is piece-wise convex with respect to T.

(b) Suppose that $k^*(w^-)$ and $k^*(w^+)$ are the optimal power-of-two multipliers of $h_i(T)$ at the left-side and right-side convex curves with regard to a junction point w, respectively. Then, $k^*(w^-) = 2k^*(w^+)$, where $w^- = w - \varepsilon, w^+ = w + \varepsilon$ and $\varepsilon \to 0^+$.

Following (3.9), it obviously holds that

$$\delta_i(2^{v_i}) < \cdots < \delta_i(2k) < \delta_i(k) < \cdots < \delta_i(2) < \delta_i(1) \tag{3.10}$$

where 2^{v_i} is an (unknown) upper bound on the value of k_i. We have the following theorem from (3.9) and (3.10).

Theorem 3.6. *Suppose that $k^*(w^-)$ and $k^*(w^+)$ are the optimal power-of-two multipliers of $h_i(T)$ at the left-side and right-side convex curves with regard to a junction point w of the $h_i(T)$ function, then $k^*(w^-) = 2k^*(w^+)$.*

The junction point $\delta_i(2^m)$ provides us the information that one should choose $k_i = 2^m$ for $T > \delta_i(2^m)$ and choose $k_i = 2^{m+1}$, *vice versa*, to secure a lower value for the $h_i(T)$ function. In other words, provided that 2^m is the optimal multiplier for $T \geq \delta_i(2^m)$, one should replace $k_i = 2^m$ with $k_i = 2^{m+1}$ as the optimal multiplier for group i at the junction point $\delta_i(2^m)$ if one searches from higher values to lower values of T. The following corollary provides an easy way to obtain the optimal multiplier for the $h_i(T)$ function for any given $T > 0$.

Corollary 3.7. *For any given $T > 0$, the optimal power-of-two multiplier $k_i^*(T)$ for the $h_i(T)$ function is given by*

$$k_i^*(T) = \begin{cases} 1, & T \in [\delta_i(1), \infty) \\ 2^{m+1}, & T \in \left[\delta_i(2^{m+1}), \delta_i(2^m)\right), m = 0, 1, ..., v_i. \end{cases} \quad (3.11)$$

Denote the function $TC^U_{PoT}(T)$ as the best unconstrained objective function value of the *CC-TFMSP* under PoT policy. Then, $TC^U_{PoT}(T)$ can be written as

$$(U - PoT) TC^U_{PoT}(T) = \inf_{T>0} \{S/T + \sum_{i=1}^{m} h_i(T)\} \quad (3.12)$$

Similar to $TC^U_{GI}(T)$, we have the following theoretical results for the $TC^U_{PoT}(T)$ function.

Proposition 3.8. *The $TC^U_{PoT}(T)$ function is piece-wise convex with respect to T.*

Proposition 3.9. *All the junction points of $h_i(T)$ function of each group i will be inherited by the $TC^U_{PoT}(T)$ function. In other words, if w is a junction point for a group i, w must also show as a junction point on the piece-wise convex curve of the $TC^U_{PoT}(T)$ function.*

We define $\mathbf{k}^U_{PoT}(T)$ as the vector of the *unconstrained* optimal power-of-two multipliers at T. Theorem 3.10 is an immediate result of Theorem 3.6 and Proposition 3.9.

Theorem 3.10. *Suppose that $\mathbf{k}^U_{PoT}(w^-)$ and $\mathbf{k}^U_{PoT}(w^+)$ are the set of optimal power-of-two multipliers for the left-side and right-side convex curves with regard to a junction point w in the plot of the $TC^U_{PoT}(T)$ function, respectively. Then, $\mathbf{k}^U_{PoT}(w^-)$ can be secured from $\mathbf{k}^U_{PoT}(w^+)$ by changing at least one of k_i from $k_i^*(w^-)$ to $k_i^*(w^+) = 2k_i^*(w^-)$.*

4 The Proposed Heuristics

In this section, we propose two efficient heuristics - one for the *CC-TFMSP* under *GI* policy (or, the *CC-TFMSP(GI)* for abbreviation) and the other for the *CC-TFMSP* under PoT policy (or, the *CC-TFMSP(PoT)*), respectively. Note that both proposed heuristics share a common framework in setting the search range, following the junction points of the unconstrained model as the "road map", and testing the feasibility of the candidate solutions. To simplify our discussion, we focus mainly on the proposed heuristic for the *CC-TFMSP(GI)*, but only highlight the difference in the other for the *CC-TFMSP(PoT)* in the following presentation. Before discussing the proposed heuristics, we give an overview of the discussions in this section as follows. Section 4.1 sets the search range for both proposed heuristics. Then, we present our search within the search range (following the junction points of the unconstrained model) in Section 4.2. As proceeding with our search to the next junction point, we locate the local optimum between two consecutive junction points and test if we are able to generate a feasible maintenance schedule for our candidate solution. Section 4.3 presents a procedure for generating a feasible maintenance schedule. Finally, we summarize the proposed search algorithms in Section 4.4.

4.1 Set the search range

In order to set the search range, we need to set the initial point and the termination point for the proposed heuristics. The first part of this subsection presents a procedure for finding an initial point. Then, we locate the termination point in the second part.

4.1.1 Find an initial point

Note that for the proposed heuristics we would like to find an initial point (which is an upper bound of the search range) in which we are able to generate a feasible maintenance schedule. We denote T_{ub} as the initial point for our proposed heuristics. To obtain T_{ub}, we start with a candidate obtained from the Common Cycle (CC) approach in which it requires that $k_i = 1$ for all i, *i.e.,* all the vehicle groups share a common maintenance cycle. If we are not able to generate a feasible maintenance schedule at the candidate, we will keep finding an unconstrained junction point with a larger value of T so that we may generate a feasible maintenance schedule there. We denote T_{CC} as the local minimum obtained from the CC approach where

$$T_{CC} = \max\left\{\sqrt{(S+\sum_i n_i C_{1i})/\sum_i n_i C_{2i}}, \sum_{i=1}^{m}(\tau_i+\pi_i)\right\} \tag{4.1}$$

When solving both the *CC-TFMSP(GI)* and the *CC-TFMSP(PoT)*, we take T_{CC} as the first candidate for the initial point. Lemma 4.1 supports such rationale.

Lemma 4.1. *There exist no local minima for both the CC-TFMSP(GI) and CC-TFMSP(PoT) functions.*

Proof. One may prove this lemma easily from the first derivative of the objective functions of the *CC-TFMSP(GI)* and *CC-TFMSP(PoT)*. □

We are ready to present an Initialization Procedure (*Proc IP*) for locating T_0 as follows.

Proc IP

1. Obtain $\{w_j\}$, i.e., the sorted sequence of all the junction points of $TC^U_{GI}(T)$ (or $TC^U_{PoT}(T)$ for the *CC-TFMSP(PoT)*) using the following steps.
   (a) Compute all the junction points $\delta_i(T)$ by eq.(3.3) (or, by eq.(3.9) for $TC^U_{PoT}(T)$) for each group of vehicle.
   (b) Generate a sequence $\{w_j\}$ by sorting all $\delta_i(T)$'s in descending order.
2. Compute T_{CC} by eq. (4.1) and check:
   (a) If $T_{CC} > \max_i\{\delta_i(1)\}$, set $T_{ub} = T_{CC}$. Let $\mathbf{k}^U_{GI}(T_{CC}) = \{k_i = 1\,|\forall i\}$(or $\mathbf{k}^U_{PoT}(T_{CC}) = \{k_i = 1\,|\forall i\}$, for the *CC-TFMSP(PoT)*) and stop *Proc IP*.
   (b) If $T_{CC} \le \max_i\{\delta_i(1)\}$, go to Step 3.
3. Obtain $\mathbf{k}^U_{GI}(T_{CC})$ by eq.(3.5) (or, $\mathbf{k}^U_{PoT}(T_{CC})$ by eq.(3.11) for the *CC-TFMSP(PoT)*) and set $T_s = T_{CC}$.
4. If $(\mathbf{k}^U_{GI}(T_s), T_s)$ (or, $(\mathbf{k}^U_{PoT}(T_s), T_s)$) is able to obtain a feasible maintenance schedule (using the Feasibility Testing Procedure in Section 4.3), then $T_{ub} = T_s$ and stop the *Proc IP*. Otherwise go to Step 5.

5. Find the smallest junction point of $TC^U_{GI}(T)$ larger than T_s, namely, $w_{s_0} = \min\{\delta_i(T) > T_s|\forall i\}$ and set $T_s = w_{s_0}$. Go to Step 4.

Note that, in the worst case, Step 5 ends up with the point $T_{ub} = \max\{\delta_i|\forall i\}$ with $\mathbf{k}^U_{GI}(T_0) = \{k_i = 1\,|\forall i\}$ (or, $\mathbf{k}^U_{PoT}(T_0) = \{k_i = 1\,|\forall i\}$ for the *CC-TFMSP(PoT)*).

4.1.2 Locate a termination point

We denote T_{lb} as the termination point (which is also a lower bound of the search range) for our proposed heuristics. Intuitively, we are able to locate a termination point T_{lb} if the global optimum sits in the range $[T_{lb}, T_{ub}]$. However, it is extremely hard to precisely find T_{lb}, especially, because the complexity from the capacity constraints in (2.4) - (2.6) in the *CC-TFMSP(GI)* (or, the *CC-TFMSP(PoT)*).

From the experience of our numerical experiments, we observe that it is generally more difficult to generate a feasible maintenance schedule for those lower-value T (with $\mathbf{k}^U_{GI}(T)$ or $\mathbf{k}^U_{PoT}(T)$). Note that the lower the value of T, the larger the value of $\mathbf{k}^U_{GI}(T)$ or $\mathbf{k}^U_{PoT}(T)$. Therefore, usually, it is not easy to generate a feasible maintenance schedule for a low-value T using the set of unconstrained optimal multipliers.

Following the above discussion, one has less chance to obtain the global optimum at low values of T. For a low-value T, the best optimal objective function value of the *CC-TFMSP(GI)* (or, the *CC-TFMSP(PoT)*) could be much more than its unconstrained version, i.e., $TC^U_{GI}(T)$ (or, $TC^U_{PoT}(T)$). Also, the gap between the constrained version and the unconstrained one tends to grow larger following our observations from the numerical experiments (though we have no rigorous proof for this observation). Therefore, we employ the lower bound for obtaining the global optimum for the $TC^U_{GI}(T)$ (or, $TC^U_{PoT}(T)$) function as a heuristic rule for locating a lower bound of the search range for the *CC-TFMSP(GI)* (or, the *CC-TFMSP(PoT)*).

Lemma 4.2 provides a lower bound for the search of the $TC^U_{GI}(T)$ function.

Lemma 4.2 (Yao and Huang [18], Lemma 4.2, pp. 39)**.** *Denote the optimal objective function value and the optimal value of the basic period for the $TC^U_{GI}(T)$ function as Ψ^* and T^*, respectively. Then, the value of β serves as a lower bound for T^* where*

$$\beta = 2S/\Psi^U \tag{4.2}$$

and Ψ^U is an upper bound on the optimal objective function value of $TC^U_{GI}(T)$.

One may easily obtain an estimate of Ψ^U by $Z'\left(\mathbf{k}^U_{GI}(T_{CC}), T_{CC}\right)$ where $Z'(\mathbf{k}, T)$ is expressed in (3.1) at the beginning of the search.

Suppose that during the search, we locate a new feasible local minimum at some $\breve{T}$ and its corresponding $\mathbf{k}(\breve{T})$ with the up-to-date, lowest objective function value $Z\left(\mathbf{k}(\breve{T}), \breve{T}\right)$. Then, we may update the lower bound β by setting $\Psi^U = Z\left(\mathbf{k}(\breve{T}), \breve{T}\right)$ since $Z\left(\mathbf{k}(\breve{T}), \breve{T}\right)$ serves as a better and tighter upper bound.

On the other hand, we shall locate a termination point by locating a lower bound for the search of the $TC^U_{PoT}(T)$ function with the following theorem.

Theorem 4.3. *The global optimum for the $TC^U_{PoT}(T)$ function must exist in the range of $\left(\breve{T}_1/2\,, T_{CC}\right]$ where is the largest local minimum of the $TC^U_{PoT}(T)$ function.*

The proof is presented in Appendix A.1.

Note that we may employ $\breve{T}_1/2$ as a lower bound of the search range since the $TC^U_{PoT}(T)$ function finds no better objective function value for the (unconstrained) *CC-TFMSP(PoT)*, but the set of capacity constraints leads to only a possible larger objective function value for $T < \breve{T}_1/2$.

4.2 Proceed with the search in the search range

The proposed heuristics search from the initial point T_{ub} to lower values of T until passing the termination point T_{lb}.

Since any local minimum serves as a candidate of the optimal solution, one may obtain its local minimum $\breve{T}(\mathbf{k})$ by first taking the derivative of the objective function in (3.1) with respect to T and then, equating it to zero for any given set of $\mathbf{k}$ obtained during the search process. Therefore, $\breve{T}(\mathbf{k})$ is given by eq. (4.3) as follows.

$$\breve{T}(\mathbf{k}) = \sqrt{(S + \sum_{i=1}^{m} \frac{n_i C_{1i}}{k_i})/\sum_{i=1}^{m} n_i C_{2i} k_i} \tag{4.3}$$

As starting from initial point T_{ub}, the search first finds $\mathbf{k}^U_{GI}(T_{ub})$ (or $\mathbf{k}^U_{PoT}(T_{ub})$) and its corresponding local minimum $\breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right)$ (or, $\breve{T}\left(\mathbf{k}^U_{PoT}(T_{ub})\right)$). If the candidate solution $\left(\mathbf{k}^U_{GI}(T_{ub}), \breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right)\right)$ (or, $\left(\mathbf{k}^U_{PoT}(T_{ub}), \breve{T}\left(\mathbf{k}^U_{PoT}(T_{ub})\right)\right)$) obtains a feasible maintenance schedule, we save it as the best-on-hand solution and move to the next junction point (which is the largest one less than T_{ub}). If it obtains no feasible maintenance schedule, we use a binary search heuristic to search for a value of T, denoted by $\tilde{T}$, so that enables $\left(\mathbf{k}^U_{GI}(T_{ub}), \tilde{T}\right)$ (or, $\left(\mathbf{k}^U_{PoT}(T_{ub}), \tilde{T}\right)$) to secure a feasible maintenance schedule with the minimal cost. When employing a binary search, we should search for larger values of T to attempt for a feasible maintenance schedule. Due to the fact that $k_i \geq 1, \forall i$, there should exist no local optimum for $T > T_{CC}$ following Lemma 4.1. Therefore, we may use T_{CC} in (4.1) as the upper bound of the search range. We locate the first candidate at $T = \left(\breve{T}(\mathbf{k}) + T_{CC}\right)/2$, and test its feasibility by *Proc FT*. We repeat such a binary search scheme to iteratively look for the minimum basic period $\tilde{T}$ that could generate a feasible maintenance schedule and, this iterative step stops when the difference between two successive T is less than a specified tolerance; for example, 1% in this study.

Recall that the proposed heuristics take the sequence of sorted junction points $\{w_j\}$ as the "roadmap" to proceed with the search and move from the upper bound of the search range to smaller values of T (where $w_{j+1} < w_j$, for all j). We denote the currently-visited junction point as w_p and the corresponding set of the on-hand multipliers as $\mathbf{k}(w_p)$, respectively. For the (unconstrained) *CC-TFMSP(GI)*, if we ignore the capacity constraints, we should replace k_ξ with $k_\xi + 1$ to seek for optimality at the next junction point following Theorem 3.5 where $\xi = \arg\max_i\{\delta_i(k_i) < w_p\}$. In other words, we should try to take $\mathbf{k}(w_{p+1})$ in (4.4) to obtain the lowest objective function value where

$$\mathbf{k}(w_{p+1}) \triangleq (\mathbf{k}(w_p)\backslash\{k_\xi\} \bigcup \{k_\xi + 1\}) \tag{4.4}$$

Similarly, the (unconstrained) *CC-TFMSP(PoT)*, we should use $\mathbf{k}(w_{p+1})$ in (4.5) where

$$\mathbf{k}(w_{p+1}) \triangleq (\mathbf{k}(w_p)\backslash\{k_\xi\} \bigcup \{2k_\xi\}) \tag{4.5}$$

Beside of pursuing the lowest objective function value, we should take into account the issue of feasibility as the search algorithm proceeds. We divide it into three possible situations as the proposed heuristic proceeds with the searching. Suppose that the current junction point is w_p and the corresponding set of multipliers is $\mathbf{k}(w_p)$. Also, we should replace k_ξ with $(k_\xi + 1)$ for the *CC-TFMSP(GI)* (or, with $2k_\xi$ for the *CC-TFMSP(PoT)*) at the next junction point is w_{p+1}.

There are three possible cases as we take into accounts the capacity constraints in (2.4)-(2.6) in the search process.

Case 1: As $(\mathbf{k}(w_{p+1}), w_{p+1})$ is able to obtain a feasible maintenance schedule, we move to w_{p+1} and proceed with the search by $p = p + 1$.

Case 2: If $(\mathbf{k}(w_p), w_{p+1})$ is able to obtain a feasible maintenance schedule, but $(\mathbf{k}(w_{p+1}), w_{p+1})$ is *not*, we proceed with the search by $p = p+1$ and $\mathbf{k}(w_{p+1}) = \mathbf{k}(w_p)$.

Case 3: As both $(\mathbf{k}(w_{p+1}), w_{p+1})$ and $(\mathbf{k}(w_p), w_{p+1})$ are *not* able to obtain a feasible maintenance schedule, we stop the search at w_{p+1} since the search algorithm is not able to obtain a feasible maintenance schedule for $T < w_{p+1}$.

4.3 The feasibility testing procedure

This section presents a procedure that assists us in generating a feasible maintenance schedule for the candidate solutions obtained during the search.

Note that this procedure was originally developed for generating a feasible production schedule for the Economic Lot Scheduling Problem. (One may refer to Yao [16], and Yao et al. [17] for the details.) Let W denote a candidate assignment, and let $L(W)$ be the maintenance load in W. Assume we are given a set of multipliers $\mathbf{k}$ and T. Use Initial Schedule Procedure (*Proc IS*) (presented in Appendix A.2) to obtain an initial schedule of maintenance W, and calculate $L(W)$. Determine L^* as the minimal maximum load secured to date, and W^* its corresponding maintenance schedule. (If W is the first maintenance schedule then set $L^* = L(W)$ and $W^* = W$). Obviously, when $L^* \leq T$, *i.e.,* the peak maintenance load among all the $K = lcm(k_1, ..., k_m)$ basic periods is no larger than the length of the *basic period*, we have a feasible maintenance schedule. We define an indicator ϕ in *Proc FT*. If a feasible maintenance schedule is obtained in *Proc FT*, ϕ is equal to 1; otherwise, $\phi = 0$. After *Proc IS*, if one obtains no feasible maintenance schedule, *i.e.,* $\phi = 0$. Use *Proc SS* (presented in Appendix A.2) to improve L^* until $\phi = 1$ or L^* can no longer be improved. If L^* has not been improved for Υ consecutive iterations, Stop.

Before presenting *Proc FT*, we need to define some new parameters. Let the number of iterations of the re-optimization be indexed by γ. The value of χ denotes the number of consecutive times that the heuristic is not able to improve $L^*(W^*)$. Let W^m be the minimal peak load schedule in a particular run of a local search (or, re-optimization), and $L(W^m)$ be the minimal peak load using the maintenance schedule W^m. We are ready to summarize *Proc FT* as follows.

Summary of Proc FT

(a) Initialization: let $\gamma = 0$ and $\chi = 0$.

(b) Start a local search with an initial schedule W obtained by *Proc IS* and *Proc GS* (presented in Appendix A.2). Let $W^m = W$ and $L(W^m) = L(W)$. If $\gamma = 0$, then let $W^* = W$ and $L^*(W^*) = L(W)$.

(c) Employ *Proc SS* to improve the peak load $L(W^m)$ in W^m.

(d) Check the improvement in $L^*(W^*)$: If $L(W^m) < L^*(W^*)$, then let $L^*(W^*) = L(W^m)$ and $W^* = W^m$. Let $\chi = 0$. Go to Step 5. If $L(W^m) \geq L^*(W^*)$, $\chi = \chi + 1$, go to Step 5.

(e) Check the termination condition: If $\chi \leq \gamma$, go to Step 2 for re-optimiation. If $\chi > \gamma$, stop the heuristic and output the $L^*(W^*)$ and W^*.

4.4 Summary of the proposed search algorithm

In this section, we summarize the proposed search algorithm in a step-by-step fashion as follows.

Step 1. Employ *Proc IP* to locate the initial point T_{ub} and $\{w_j\}$, i.e., the sorted sequence of all the junction points of $TC^U_{GI}(T)$ (or $TC^U_{PoT}(T)$ for the *CC-TFMSP(PoT)*). Do the followings:

(a) Find $\mathbf{k}^* = \mathbf{k}^U_{GI}(T_{ub})$ (or, $\mathbf{k}^* = \mathbf{k}^U_{PoT}(T_{ub})$ for the *CC-TFMSP(PoT)*), and its corresponding local minimum $\breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right)$ (or, $\breve{T}\left(\mathbf{k}^U_{PoT}(T_{ub})\right)$) by eq. (4.3).

(b) If $\left(\mathbf{k}^U_{GI}(T_{ub}), \breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right)\right)$ (or, $\left(\mathbf{k}^U_{PoT}(T_{ub}), \breve{T}\left(\mathbf{k}^U_{PoT}(T_{ub})\right)\right)$) obtains a feasible maintenance schedule, set $T^* = \breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right)$(or, $T^* = \breve{T}\left(\mathbf{k}^U_{PoT}(T_{ub})\right)$); otherwise, use a binary search heuristic to search for a value of T, denoted by $\tilde{T}$, so that enables $\left(\mathbf{k}^U_{GI}(T_{ub}), \tilde{T}\right)$ (or, $\left(\mathbf{k}^U_{PoT}(T_{ub}), \tilde{T}\right)$) to secure a feasible maintenance schedule with the minimal cost, and set $T^* = \tilde{T}$.

(c) Set $TC^* = Z\left(\mathbf{k}^*, T^*\right)$, $\Psi^U = TC^*$, and obtain $T_{lb} = \beta$ by eq. (4.2). Let $p = 0$ and $r = 0$.

Step 2. If $p = 0$, $w_1 = \max\{w_j | w_j < T_{ub}\}$; otherwise, directly locate w_{p+1}. If $w_{p+1} < T_{lb}$, then go to Step 5; otherwise, obtain $\mathbf{k}(w_{p+1}) \triangleq (\mathbf{k}(w_p)\backslash\{k_\xi\} \bigcup \{k_\xi + 1\})$ using eq. (4.4) (or, $\mathbf{k}(w_{p+1}) \triangleq (\mathbf{k}(w_p)\backslash\{k_\xi\} \bigcup \{2k_\xi\})$ using eq. (4.5) for the *CC-TFMSP(PoT)*) where $\varepsilon = \arg\max_i\{\delta_i(k_i) < w_p\}$, and test the feasibility of $(\mathbf{k}(w_{p+1}), w_{p+1})$ using *Proc FT*. If $(\mathbf{k}(w_{p+1}), w_{p+1})$ is able to obtain a feasible maintenance schedule, go to Step 3; otherwise, go to Step 4.

Step 3. Obtain $\breve{T}(\mathbf{k}(w_p))$ by (4.3) and compute $Z(\mathbf{k}(w_p), \breve{T}(\mathbf{k}(w_p)))$. If $Z(\mathbf{k}(w_p), \breve{T}(\mathbf{k}(w_p)))$ $< TC^*$, set $\mathbf{k}^* = \mathbf{k}(w_p)$, $T^* = \breve{T}(\mathbf{k}(w_p))$, and $TC^* = Z\left(\mathbf{k}^*, T^*\right)$. Obtain $\Psi^U = Z\left(\mathbf{k}^*, T^*\right)$ and update $T_{lb} = \beta$ if $\beta = 2S/\Psi^U > T_{lb}$. (For the *CC-TFMSP(PoT)*, we do the following step additionally: if $r = 0$, we set $\breve{T}_1 = \breve{T}(\mathbf{k}(w_p))$ and $r = 1$ if $\breve{T}(\mathbf{k}(w_p)) \in [w_{p+1}, w_p]$. Also, update $T_{lb} = \breve{T}_1/2$ if $\breve{T}_1/2 > T_{lb}$.) Proceed with the search by $p = p + 1$(i.e., moving to w_{p+1}) and go to Step 2.

Step 4. If $(\mathbf{k}(w_p), w_{p+1})$ is not able to obtain a feasible maintenance schedule, go to Step 5; otherwise, set $\mathbf{k}(w_{p+1}) = \mathbf{k}(w_p)$. (For the *CC-TFMSP(PoT)*, we do the following step additionally: if $r = 0$, we set $\breve{T}_1 = \breve{T}(\mathbf{k}(w_p))$ and $r = 1$ if $\breve{T}(\mathbf{k}(w_p)) \in [w_{p+1}, w_p]$. Also, update $T_{lb} = \breve{T}_1/2$ if $\breve{T}_1/2 > T_{lb}$.) Proceed with the search by $p = p + 1$ (i.e., moving to w_{p+1}) and go to Step 2.

Table 1: The data set of the five-group example

Group i	n_i	Y_i	a_i (\$/day)	b_i (\$/day)	s_i (\$)	setup time τ_i (in days)	maintenance time π_i (in days)
1	10	0.90	23	35	88	0.09	0.81
2	34	0.95	8	18	192	0.08	1.62
3	30	0.85	21	5	193	0.03	0.87
4	36	0.95	69	60	205	0.09	1.43
5	12	0.94	13	4	204	0.04	0.76

Step 5. Output the optimal solution (k^*, T^*) with its corresponding objective function value TC^*_{GI} for the *CC-TFMSP(GI)* (or TC^*_{PoT} for the *CC-TFMSP(PoT)*).

5 Numerical Experiments

The first part of this section demonstrates the implementation of the proposed search algorithm via an example. Then, using randomly generated instances, we compare the solution obtained from the proposed search algorithm with a trivial solution from the common cycle approach in the second part of this section.

5.1 A demonstrative example

In this section, we present a five-group example with a fixed cost $S = 50$. The data set of this example is shown in Table 1.

Under *GI* policy, we present the implementation details of the proposed search algorithm as follows.

We start with the procedure *Proc IP*. Then, we obtain all the junction points $\delta_i(k_i)$ by eq. (3.3) (or, eq.(3.9)) for each vehicle group and generate a sequence $\{w_j\}$ by sorting all the junction points of the $TC^U_{GI}(T)$ function with unconstraint capacity in descending order. In this example, we locate $T_{CC} = 5.80$ by eq. (4.1) between the 2^{nd} and the 3^{rd} junction points. Since T_{CC} is less than $\max_i\{\delta_i(1)\} = 10.51$, since $((1, ..., 1), T_{CC})$ is able to obtain a feasible maintenance schedule, we have $T_{ub} = T_{CC}$ and stop the *Proc IP*. We obtain $\mathbf{k}^U_{GI}(T_{ub})$ = (1,1,2,1,2) by eq. (3.5) and $\breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right) = 3.379$ by eq. (4.3). Since $\left(\mathbf{k}^U_{GI}(T_{ub}), \breve{T}\left(\mathbf{k}^U_{GI}(T_{ub})\right)\right)$ obtains a feasible maintenance schedule, set $\mathbf{k}^*$ = (1,1,2,1,2) and $T^* = 3.379$. We also obtain $TC^* = Z(\mathbf{k}^*, T^*) = \$9,689.4$, set $\Psi^U = TC^*$, and obtain $T_{lb} = \beta = 0.01032$ by eq. (4.2). Now, we have $p = 0$ and $r = 0$.

Since $p = 0$, we locate the next junction point at $w_1 = \max\{w_j | w_j < T_{ub}\} = 5.257$. We next obtain $\varepsilon = \arg\max_i\{\delta_i(k_i) < T_{CC}\} = 5$ with $\mathbf{k}(w_1) = (1, 1, 2, 1, \underline{3})$, and compute $\breve{T}(\mathbf{k}(w_1)) = 3.319$. Since $\left(\mathbf{k}(w_1), \breve{T}(\mathbf{k}(w_1))\right)$ is able to secure a feasible maintenance schedule, we compute $Z\left(\mathbf{k}(w_1), \breve{T}(\mathbf{k}(w_1))\right) = \$9,643.8$ ($< TC^*$). Therefore, we update $\Psi^U = TC^* = \$9,643.8$, $T^* = 3.319$, $\mathbf{k}^*$ = (1,1,2,1,3) and $T_{lb} = \beta = 0.01036$.

The search algorithm proceeds with its search until it meets the lower bound $\beta(= 0.01036)$. The optimal solution is obtained $T^* = 3.319$ and $\mathbf{k}^*$ = (1,1,2,1,3) with $TC^* = \$9,643.8$.

On the other hand, we summarize the details of search process under *PoT* policy as follows.

Before obtaining w_1, all the steps are the same as those details under *GI* policy. We also obtain $\varepsilon = \arg\max_i\{\delta_i(k_i) < T_{CC}\} = 5$, but with $\mathbf{k}(w_1) = (1, 1, 2, 1, \underline{4})$ since we update $\mathbf{k}(w_1) \triangleq (\mathbf{k}(T_{ub})\backslash\{k_\xi\} \bigcup\{2k_\xi\})$ and obtain $\breve{T}(\mathbf{k}(w_1)) = 3.28$. Since $\left(\mathbf{k}(w_1), \breve{T}(\mathbf{k}(w_1))\right)$ is able to secure a feasible maintenance schedule, we compute $Z\left(\mathbf{k}(w_1), \breve{T}(\mathbf{k}(w_1))\right) = \$9,654.5$ ($< TC^* = \$9,689.4$). Therefore, we update $\Psi^U = TC^* = \$9,654.5$, $T^* = 3.28$, $\mathbf{k}^* = (1, 1, 2, 1, 4)$ and $T_{lb} = \beta = 0.010358$. In fact, it is also the optimal solution under *PoT* policy, which is 0.1% higher than the solution obtained under *GI* policy.

One may be interested in the impact of the capacity constraints on the optimal average total cost of the whole transport fleet. In this example, the optimal solution for the unconstrained version is $T^* = 2.49$, $\mathbf{k}^* = (1, 2, 4, 1, 4)$, with its $TC^* = \$9,087.27$. Note that this solution is 6.0% less than the optimal solution of the constrained problem (under *GI* policy). However, one may easily versify that this solution is actually infeasible. (The sum of X_1 and X_4 is 2.42 where $X_i = \tau_i + \pi_i$. Obviously, it leads to infeasibility in the maintenance schedule as adding the maintenance duration of any other group.)

On the other hand, we would like to compare the obtained solution with an easy solution from the Common Cycle (*CC*) approach in which it requires that $k_i = 1$ for all i, *i.e.,* all the vehicle groups share a common maintenance cycle, T_{CC}, which is expressed in eq. (4.1). Note that the solution from the *CC* approach is always a feasible solution so that it may serve as an upper bound on the optimal objective function value of the *CC-TFMSP*. In this example, we have $Z((1, ..., 1), T_{CC}) = \$10,487$, which is 8.7% larger than the optimal solution obtained from our proposed algorithm under *GI* policy. Evidently, thought the *CC* approach is easy for implementation, the managers has to sacrifice the quality of the maintenance scheduling comparing with the solutions obtained from the proposed search algorithms in this study.

5.2 Random experiments

To verify the effectiveness of the proposed search algorithms, we would present a summary of our random experiments. In our experiments, we test for five numbers of vehicle groups ($m = 3, 5, 7, 10, 20$), and five values of the fixed cost ($S = 25, 50, 100, 200, 250$). This yielded 25 combinations of parameter settings. For each combination, we randomly generate 1,000 instances by picking the (random) values of $Y_i, a_i, b_i, \tau_i, \pi_i$ and s_i using uniform distribution functions with their ranges indicated in Table 2. Therefore, we have a total of 25,000 instances in our experiments.

We define *CR*, namely, the percentage of cost reduction, as a performance measure of the obtained solution from our proposed search algorithms as (5.1).

$$CR = \left[\left(Z((1, ..., 1), T_{CC}) - TC^*\right) / Z((1, ..., 1), T_{CC})\right] \cdot 100\% \tag{5.1}$$

We take the optimal solution from the *CC* approach as the benchmark in our experiments. For each instance, we solve it using the *CC* approach, our proposed search algorithm under *GI* policy, and that under *PoT* policy. Table 3 summarizes our experimental results of those 25 settings. For each instance, the value of *CR* under *GI* policy is greater than (or equal to) that under *PoT* policy, thought the difference is insignificant. One may also observe that *PoT* policy is more efficient than *GI* policy by comparing the run time of both proposed search algorithm.

One may have some more interesting observations on Table 3. For those instances with a small number of vehicle groups, the maximum value of *CR* is more significant. Obviously, the managers have to pay much more attention to these cases since they should take into

Table 2: The parameter settings of our random experiments.

m	3, 5, 7, 10, 20
S	25, 50, 100, 200, 250
n_i	U[10-35]
τ_i	U[0.03-0.09]
π_i	U[0.8-1.0]
Y_i	U[0.9-0.95]
a_i	U[5-70]
b_i	U[5-70]
s_i	U[80-210]

accounts the capacity constraints for generating a feasible maintenance schedule. As the number of vehicle groups increases, it becomes more difficult to secure a feasible maintenance schedule. It often happened that only the *CC* approach is able to obtain a feasible solution, and the proposed search algorithm resulted in no *CR* for the generated instance in such cases.

On the other hand, we would like to test if the optimal solution obtained from the unconstrained model in Yao and Huang [18] is able to obtain a feasible maintenance schedule. (It surely solves an optimal solution if it obtains one.) After obtaining the optimal solution from Yao and Huang [18], we test its feasibility by the equations in (2.4) to (2.6) and record those being able to obtain a feasible maintenance schedule. Table 4 summarizes the results of our numerical experiments.

The unconstrained model in Yao and Huang [18] solves an optimal solution with a feasible maintenance schedule for those small-size (e.g., 3 and 5) and large-setup cost (e.g., larger than 100) instances. We may observe that the capacity constraints have minor impacts on these instances. However, as the problem-size (m) increases, it is more difficult to generate a feasible maintenance schedule. Therefore, for most of the larger-size instances, the optimal solutions from Yao and Huang [18] are not able to obtain a feasible maintenance schedule.

6 Concluding Remark

This paper deals with the problem of determining maintenance frequency for the vehicle groups in a transport fleet with limited maintenance capacity. We formulate the mathematical model for the interested problem. Also, we conduct theoretical analysis on this problem under both *GI* and *PoT* policies. By utilizing the theoretical results on the junction points, we proposed two search algorithms for solving the problem. Our numerical experiments demonstrate that the proposed search algorithms are efficient and may serve as a useful decision-support tool for planning the maintenance scheduling for the vehicle groups in a transport fleet.

In this study, we assume that all the maintenance activities are done in a single facility. The authors are currently working on an extension in which the vehicles have to go through a series of facilities for conducting the required maintenance activities. Obviously, the capacity scheduling in this problem will be even more complicated since the managers have to deal with the challenge from the constraint that the activity for a vehicle group in one maintenance facility must be finished before the vehicle group enters the posterior maintenance facility.

Table 3: Computational results for the random experiments.

m	S	GI policy			PoT policy		
		Avg CR(%)	Max CR(%)	Run-time (sec)	Avg CR(%)	Max CR(%)	Run-time (sec)
3	25	2.405	20.330	0.741	2.399	19.140	0.231
	50	2.533	16.620	0.661	2.533	16.620	0.170
	100	2.741	19.410	0.661	2.735	19.150	0.150
	200	2.497	21.530	0.661	2.497	21.530	0.160
	250	2.252	16.910	0.670	2.252	16.710	0.161
5	25	2.509	15.780	1.563	2.506	14.320	0.200
	50	2.453	15.500	1.572	2.452	14.420	0.210
	100	2.447	17.060	1.582	2.439	17.890	0.200
	200	2.143	18.070	1.593	2.141	17.730	0.211
	250	2.275	16.790	1.562	2.275	16.160	0.220
7	25	1.576	11.090	2.924	1.567	10.570	0.280
	50	1.533	11.540	2.914	1.530	11.540	0.271
	100	1.487	10.470	2.914	1.481	9.180	0.280
	200	1.476	13.130	2.915	1.469	13.130	0.291
	250	1.613	14.460	2.914	1.613	15.070	0.270
10	25	0.656	7.660	5.758	0.656	7.660	0.391
	50	0.713	7.410	5.748	0.713	7.700	0.400
	100	0.609	5.940	5.739	0.609	6.060	0.391
	200	0.714	8.770	5.758	0.711	6.560	0.390
	250	0.608	5.640	5.758	0.606	5.640	0.401
20	25	0.007	2.050	23.203	0.007	2.050	0.941
	50	0.009	2.160	22.563	0.009	2.160	1.002
	100	0.002	1.150	22.372	0.002	1.150	0.951
	200	0.005	1.280	23.374	0.005	1.280	0.971
	250	0.007	2.330	22.332	0.007	2.330	0.942

Table 4: The portion of the instances where the unconstrained model obtains a feasible maintenance schedule.

		S				
		25	50	100	200	250
m	3	3.6%	4.0%	6.2%	6.3%	7.2%
	5	1.0%	0.4%	1.4%	2.2%	2.6%
	7	0.0%	0.0%	0.2%	0.4%	0.6%
	10	0.0%	0.0%	0.0%	0.0%	0.0%
	20	0.0%	0.0%	0.0%	0.0%	0.0%

A.1 Proof of Theorem 4.3

Recall that $\breve{T}_1$ is the largest local minimum for the $TC^U_{PoT}(T)$ function. Let T^* and $\mathbf{k}^*$ be the optimal value of T and the set of optimal multipliers secured in the range of $\left(\breve{T}_1/2\ ,\breve{T}_1\right]$, respectively, and $TC^U_{PoT}(T^*)$ be the optimal objective function value.

We prove Theorem 4.3 by showing that the $TC^U_{PoT}(T)$ function secures no local minimum below $\breve{T}_1/2$ such that its objective function value is lower than $TC^U_{PoT}(T^*)$. We define a function $\Gamma^U_{PoT}(T)$ by

$$\Gamma^U_{PoT}(T) = \sum_{i=1}^{m} h_i(T) = \sum_{i=1}^{m} \left\{ \inf_{k_i = 2^{p_i}, p_i \in \mathbb{N}, \forall i} \Phi_i(k_i, T) \right\} \tag{A.1}$$

Let $\tilde{T}_1$ be the largest local minimum for the $\Gamma^U_{PoT}(T)$ function. Suppose that $\tilde{T}^*$ and $\tilde{\mathbf{k}}^*$ obtains the minimum value for the $\Gamma^U_{PoT}(T)$ function in $\left(\tilde{T}_1/2\ ,\tilde{T}_1\right]$. Then, one must secure another optimal solution at $\tilde{T}^*/2$ for the $\Gamma^U_{PoT}(T)$ function with the set of optimal multipliers being $2\tilde{\mathbf{k}}^*$ (otherwise, it contradicts with the assumption that $2\tilde{\mathbf{k}}^*$ and $\tilde{\mathbf{k}}^*$ are optimal in $\left(\tilde{T}_1/2\ ,\tilde{T}_1\right]$. Also, if we would like to locate the global optimum of the $\Gamma^U_{PoT}(T)$ function, we may skip the range of $\left(0, \tilde{T}_1/2\ \right]$ since the $\Gamma^U_{PoT}(T)$ function repeats the shape of its curve in $\left(0, \tilde{T}_1/2\ \right]$ (so that there will be no better solution in $\left(0, \tilde{T}_1/2\ \right]$).

Clearly, $TC^U_{PoT}(T) = S/T\ + \Gamma^U_{PoT}(T)$. Since S/T is a monotone function of T, the term S/T increases as the values of T decrease. We may assert that no better solution than $(\mathbf{k}^*, T^*)$ in $\left(0, \tilde{T}_1/2\ \right]$ owing to the characteristics of the $\Gamma^U_{PoT}(T)$ function. It is easy to show that $\breve{T}_1 < \tilde{T}_1$. Therefore, T^* must be obtained in the range of $\left(\breve{T}_1/2\ ,\breve{T}_1\right]$. On the other hand, Lemma 4.1 asserts that there exists no local minimum $T > T_{CC}$. Therefore, the global optimum for the $TC^U_{PoT}(T)$ function resides in the range of $\left(\breve{T}_1/2\ , T_{CC}\right]$. *Q.E.D.*

A.2 The procedures in the feasibility testing procedure

We present the three major procedures in the Feasibility Testing Procedure (*Proc FT*).

A.2.1 The Initial Schedule Procedure (Proc IS)

The detail of rocedure *Proc IS* is as follows. Denote by N as the set of all m groups. Let ϑ be a subset of the groups of vehicle, $|\vartheta| \leq m$, and let $W(\vartheta)$ be a partial maintenance schedule containing only the subset ϑ of groups. When *Proc IS* is applied for the first time, one starts with an empty set of group $\vartheta = \phi$ and the empty maintenance schedule, $W(\vartheta) = \phi$. Then, one assigns the maintenance durations of groups to $W(\vartheta)$ following *Proc GS* (described next), and update ϑ accordingly, until all m groups are assigned, securing an initial schedule $W = W(N)$.

Next, consider the case when *Proc IS* has been used at least once. Suppose that W_0 is the maintenance schedule selected for re-improvement from the last iteration. To start a new iteration of improvement, one randomly selects a subset of groups which frequencies and durations are *fixed*. Let it be denoted by F. The maintenance schedule in the new W' for the groups in the set of F are fixed at their previous values. Let $\bar{F} = (N - F)$ be the set of groups which are yet-to-be-scheduled in W'. *Proc IS* is now used to generate an initial schedule for the next run of re-improvement in the following manner: Let $\vartheta = F$, and $W'(\vartheta) = W_0(F)$, and subtract the maintenance durations of the groups in the set F from N. $W'(F)$ is a new partial schedule. For the groups in the set $\bar{F}$, *Proc IS* calls upon *Proc GS* which iteratively assign group i with the *longest maintenance duration* (including the setup time and the maintenance time) to the least loaded basic period from among the unassigned groups in $\bar{F}$.

The rationale behind this heuristic rule is as follow. If a group i with a long maintenance duration is scheduled after most of the groups have been assigned, it may create a relative large peak load since one may be forced to assign that long maintenance duration to an already heavily loaded basic period.

A.2.2 The Group Scheduling Procedure (Proc GS)

The Group Scheduling Procedure (*Proc GS*) constructs the schedule W incrementally. Let ϑ be subset of the groups, $|\vartheta| \leq m$, and let $W(\vartheta)$ be a partial maintenance schedule containing only the subset ϑ of groups. Determine the least common multiplier for all the k_i's in the set ϑ, $\kappa(\vartheta) = lcm\{k_i | i \in \vartheta\}$.

Suppose that group $\hat{i}$ is the next group to be added to the set ϑ. To assign the occupancy time X_i to the maintenance schedule $W(\vartheta)$, one first updates $\kappa(\vartheta)$ by $\kappa\left(\vartheta \cup \{\hat{i}\}\right) = lcm\{\kappa(\vartheta), k_{\hat{i}}\}$. If $\kappa\left(\vartheta \cup \{\hat{i}\}\right) > \kappa(\vartheta)$, one should make $\kappa\left(\vartheta \cup \{\hat{i}\}\right)/\kappa(\vartheta)$ copies of $W(\vartheta)$ in the entire planning horizon of $\kappa\left(\vartheta \cup \{\hat{i}\}\right)$ basic periods to construct a 'layout' for $W\left(\vartheta \cup \{\hat{i}\}\right)$. If $\kappa\left(\vartheta \cup \{\hat{i}\}\right) = \kappa(\vartheta)$, then one employs $W(\vartheta)$ directly as a layout for $W\left(\vartheta \cup \{\hat{i}\}\right)$. Then, one obtains the minimal peak load by choosing among the $k_{\hat{i}}$ ways of assigning X_i to the layout of $W\left(\vartheta \cup \{\hat{i}\}\right)$.

A.2.3 The Schedule Smoothing Procedure (Proc SS)

The Schedule Smoothing Procedure (*Proc SS*) is used after an initial schedule W has been obtained by *Proc IS*. Its aim, as the name suggests, is to 'smooth' the load, *i.e.,* minimize the maximal load, on the basic periods in the planning horizon. Such 'smoothing' is accomplished *via* three subroutines which we label the *Removal Routine*, the *Pair-Exchange Routine* and the *Two-to-One Exchange Routine*. The *Removal Routine* attempts to accomplish the objective by removing some X_i from the maximally loaded basic period and assigning it to some other basic period. The *Pair-Exchange Routine* tries to achieve the same result by exchange a group maintained in the maximally loaded basic period with another group which has a shorter maintenance duration and is *not* produced in the maximally loaded basic period. Finally, the *Two-to-One Exchange Routine* exchanges the occupancy times of two groups with another group which has a longer maintenance duration, and which is maintained in the maximally occupied basic period (It is evident that the *Two-to-One* exchange procedure can be expanded (or even optimized) for more combinations of groups exchanged. One may refer to Yao *et al.* (2003) for the pseudo-codes of the *Removal Routine*, the *Pair-Exchange Routine* and the *Two-to-One Exchange Routine.*

References

[1] B. N. Amotz, R. Bhatia, J. Naor and B. Schieber, Minimizing service and operation costs of periodic scheduling, *Math. Oper. Res.* 27 (2002) 518–544.

[2] S. Anily, C.A. Glass and R. Hassin, Scheduling of maintenance services to three machines, *Ann. Oper. Res.* 86 (1999) 375–391.

[3] A.H. Christer and T. Doherty, Scheduling overhauls of soaking pits, *J.Oper. Res. Soc.* 28 (1997) 915–926.

[4] S.K. Goyal and A. Gunasekaran, Determining economic maintenance frequency of a transport fleet, *Internat. J. Systems Sci.* 23 (1992) 655–659.

[5] S.K. Goyal and M.I. Kusy, Determining economic maintenance frequency for a family of machines, *J. Oper. Res. Soc.* 36 (1985) 1125–1128.

[6] J.Y. Huang and M.J. Yao, A genetic algorithm for solving economic lot scheduling problem in flow shops, *Internat. J. Production Res.* 46 (2008) 3737–3761.

[7] H. Luss, Maintenance policies when deterioration can be observed by inspections, *Oper. Res.* 24 (1976) 359–366.

[8] H. Luss and Z. Kander, Preparedness model dealing with N systems operating simultaneously, *Oper. Res.* 22 (1974) 117–128.

[9] W.E. Moudani and F. Mora-Camino, A dynamic approach for aircraft assignment and maintenance scheduling by airlines, *J. Air Transp. Manag.* 6 (2000) 233–237.

[10] N. Papadakos, Integrated airline scheduling, *Comput. Oper. Res.* 36 (2009) 176–195.

[11] C. Sriram and A. Haghani, An optimization model for aircraft maintenance scheduling and re-assignment, *Transp. Res. Part A: Policy and Pract.* 37 (2003) 29–48.

[12] D.R. Sule and B. Harmon, Determination of coordinated maintenance scheduling frequencies for a group of machines, *AIIE Trans.* 11 (1979) 48–53.

[13] R. van Egmond, R. Dekker and R.E. Wildeman, Correspondence: determining economic maintenance of a transport fleet, *Internat. J. Systems Sci.* 26 (1995) 1755–1757.

[14] R.E. Wildeman and R. Dekker, Dynamic influence in multi-component maintenance, *Qual. Reliab. Eng. Internat.* 13 (1997) 199–207.

[15] S. Yanagi, Iteration method for reliability evaluation for a fleet system, *J. Oper. Res. Soc.* 43 (1992) 885–896.

[16] M.J. Yao, The peak load minimization problem in cyclic production, *Comput. Oper. Res.* 28 (2001) 1441–1460.

[17] M.J. Yao, S.E. Elmaghraby and I.-C. Chen, On the feasibility testing of the economic lot scheduling problem using the extended basic period approach, *J. Chin. Inst. Ind. Engineers* 20 (2003) 435–444.

[18] M.J. Yao and J.Y. Huang, A new optimal search algorithm for the transportation fleet maintenance scheduling problem, *J. Oper. Res. Soc. Japan* 49 (2006) 33–48.

Manuscript received
revised
accepted for publication

Ming-Jong Yao
Department of Transportation & Logistics Management
National Chiao Tung University
E-mail address: `myaoie@gmail.com`

Jia-Yen Huang
Department of Information Management
National Chin-Yi University of Technology
No.57, Sec. 2, Zhongshan Rd., Taiping Dist.
Taichung City 41170, Taiwan, R.O.C.
E-mail address: `jygiant@ncut.edu.tw`