Journal of Naval Sciences and Engineering
2024, Vol. 20, No. 1, pp. 67-90
Industrial Engineering/Endüstri Mühendisliği

DOI: 10.56850/jnse.1465463

RESEARCH ARTICLE

**An ethical committee approval and/or legal/special permission has not been required within the scope of this study.*

A WORKLOAD DEPENDENT RESOURCE CONSTRAINED SCHEDULING PROBLEM FOR NAVY HELICOPTER PILOTS

Fatih ÇELİK[1]*
Ertan GÜNER[2]

[1] *Gazi University, Graduate School of Natural and Applied Sciences, Ankara, Turkey, fatih.celik5@gazi.edu.tr*

[2] *Gazi University, Engineering and Architecture Faculty, Industrial Engineering Department, Ankara, Turkey, erguner@gazi.edu.tr*

Received: 05.04.2024 **Accepted: 29.05.2024**

ABSTRACT

In this paper, we consider a problem inspired by a real-life problem, which aims to schedule high multiplicity jobs on a single machine by taking into account the organization-specific constraints in a different schedule structure. The schedule is daily with daytime and nighttime periods. The operator is considered as an additional resource that varies in terms of consumption and scheduling depending on the period. There are specific rest periods before and after night-period jobs, and night-period jobs affect both the daily working time and number of the jobs in the daytime- period. In addition, the operator's daily workload is divided into two categories: normal and heavy. If the workload is heavy on consecutive days, specific rest periods must be scheduled. The integer programming model of the problem is presented. The feasible solutions obtained in a short time with greedy constructive heuristic algorithms are used in the exact solution approach as both upper bound and warm-start point. Finally, the effectiveness of the solution approaches is compared and evaluated through numerical experiments carried out for a variety of problem instances of different sizes.

Keywords: *Scheduling, Additional Resources, High-Multiplicity, Integer Programming, Greedy Constructive Heuristic.*

DENİZ HELİKOPTER PİLOTLARI İÇİN İŞ YÜKÜNE BAĞLI KAYNAK KISITLI BİR ÇİZELGELEME PROBLEMİ

ÖZ

Bu çalışmada; az çeşit yüksek sayıdaki (yüksek multiplisite) işlerin, organizasyona özgü çalışma düzeni ve kısıtlar eşliğinde farklı bir çizelge yapısı altında tek makinede çizelgelenmesini amaçlayan, gerçek yaşam probleminden kurgulanan bir problem ele alınmıştır. Gündüz ve gece olarak ikiye ayrılan çizelge yapısında, işlerin yapıldığı periyoda göre operatör ek kaynağının tüketilmesi ve çizelgelenmesi açısından farklı kısıtlar dikkate alınmaktadır. Gece periyodunda yapılan işler öncesi ve sonrasında operatöre yönelik özel dinlenme süreleri kısıtları bulunmakta, gece periyodunda yapılan işlerin hem süre hem de sayı olarak gündüz periyodundaki iş çizelgelemesine etkileri bulunmaktadır. Ayrıca, operatörün günlük iş yükü normal ve ağır olarak iki kategoriye ayrılmaktadır. Ardışık günlerde ağır kategori iş yükü oluştuğunda özel dinlenme sürelerinin çizelgelenmesi gerekmektedir. Problemin tam sayılı programlama modeli sunulmuştur. Açgözlü kurucu sezgisel algoritmalar ile kısa sürede elde edilen uygun çözümler hem üst sınır hem de sıcak başlangıç olarak tam çözüm yaklaşımında kullanılmıştır. Son olarak, çözüm yaklaşımlarının etkinliği farklı büyüklükteki örnek test problemleri kullanılarak karşılaştırılmış ve değerlendirilmiştir.

Anahtar Kelimeler: *Çizelgeleme, Ek Kaynaklar, Yüksek Multiplisite, Tam Sayılı Programlama, Açgözlü Kurucu Sezgisel.*

1. INTRODUCTION

Personnel scheduling has been studied extensively in the scheduling literature. The main reason for this is economic considerations, but another important reason is the inevitable changes in job characteristics and working rules over time. Organizations and companies must obey all the regulations on working time enforced by the authorities, as well as the direct or indirect costs of scheduling workforce. Therefore, all the restrictions enforced by government regulations, union agreements and company-specific rules must be taken into account in personnel scheduling.

There are different work regulations for different industries. The aviation industry probably has the most stringent policies and regulations regarding working hours due to the risks involved. Regulations on working hours and rest periods for pilots and flight crews are constantly monitored, particularly to reduce fatigue-related incidents. On the other hand, military aviation differs from civil aviation because of the different types of aircraft and the different purposes for which they are used. Thus, military pilots are subject to specific work and rest regulations. In this paper, we study the helicopter pilot scheduling problem with organization-specific work and rest regulations adapted from the Turkish Naval Air Force.

The study is organized as follows: Section 2 provides a brief literature review of the personnel scheduling problem, focusing on work and rest regulations. Section 3 presents the problem definition and integer programming model of the problem. Section 4 describes solution approaches including greedy heuristics and exact solution. Numerical experiments are performed in Section 5 to compare the solution approaches. Finally, Section 6 concludes the paper.

2. LITERATURE REVIEW

The personnel scheduling, or rostering, problem introduced by Dantzig in the 1950s has evolved over time (Dantzig, 1954; Bergh et al., 2013). The personnel scheduling problem is very diverse and can be classified according to different methods. Bergh et al. (2012) organized the personnel scheduling problem into 4 classification fields as follows: “personnel characteristics, decision delineation and shifts definitions”, “constraints, performance measures and flexibility”, “solution method and uncertainty incorporation” and “application area and applicability of research”. Ozder et al. (2020) categorize the personnel scheduling problem according to the characteristics: “Days-off scheduling problem”, “Shift scheduling problem”, “The cyclic staffing problem”, “Crew scheduling problem”, “Operator scheduling problem”. The constraints and solution methods of the operator scheduling problem are of primary interest in this paper.

The Nurse Scheduling Problem (also known as the Nurse Rostering Problem - NRP) is the problem of finding an optimal way to assign nurses to shifts takes the first place in the literature of personnel scheduling problem (Ozder et al., 2020). Burke et al. (2004) categorized NRP papers according to solution methods, constraints and performance measures. There are many different types of time-related constraints in the NRP. In addition to the time related constraints are enforced by government regulations and union agreements, there are also hospital-specific working rules. This gives some hospital administrators the flexibility to set and define the structure of the time related constraints.

The Driver Scheduling Problem (DSP) is another large area of research in the personnel scheduling. DSP consists of selecting a set of duties for the drivers or pilots of vehicles, (e.g., buses, trains, boats, or planes) for the transportation of passengers or goods (Portugal et al., 2009). The DSP can also be divided into sub-categories such as Bus Driver Scheduling Problem (BDSP), Truck Driver Scheduling Problem (TDSP).

Driver planning in road freight transportation is different from transportation in other areas -airlines, railways, mass transit and buses (Goel et al., 2012). All tasks to be performed by employees in regular shifts are determined from a given timetable (either flight, train, subway or bus) in which arrival times are fixed (Ernst et al., 2004), however, there is no regular shift in road freight transportation and arrival times are typically not fixed but can even be scheduled with some degree of freedom (Ernst et al., 2004). Even some of the studies combine Vehicle Routing Problem (VRP) and TDSP (Goel, 2009; Kok et al., 2010). Driving periods, breaks, and rest periods must be scheduled in TDSP according to the regulations. Regulations may vary country to country. The two most widely studied regulations in the literature are the US-TDSP for the United States of America (Goel & Kok, 2012) and the EU-TDSP for the European Union (Goel, 2009; Goel, 2010). For example, a driver cannot accumulate more than 11 hours of driving in the U.S. and 9 hours of driving in Europe between two consecutive daily rests. In addition, there may also be different company-specific rules that do not violate the rules of higher regulatory bodies in the same country.

The Crew Scheduling Problem (CSP) is another type of personnel scheduling problem which model is relatively different from the other personnel scheduling models. CSP and DSP are related problems. CSP appears in a number of transportation contexts such bus and rail transit, truck and rail freight transport, and freight and passenger air transportation similar to DSP. CSP particularly important in the transport sector in the airline industry and has received the most attention from both the industry and from the academic community (Ozder et al., 2020; Barnhart et al., 2003). The Airline Crew Scheduling Problem (ACSP) is one of the most comprehensive of crew scheduling applications in terms of economic size and impact. A large number of restrictive rules mandated by governing agencies (FAA in the US, EASA in the EU, DGCA in Türkiye), labor organizations and the airlines themselves make ACSP one of the hardest CSPs.

ACSP can be defined as the assignment of flight crew (cockpit and cabin) to scheduled flights, so as to ensure that the crew needed for all flights are covered. Due to the difficulty of solving the ACSP as one integrated problem, it is divided into two sub-problems: Crew Pairing Problem (CPP) and Crew Rostering Problem.

It is possible to give examples of personnel scheduling problem involving restrictions on working hours in other sectors. These constraints usually vary significantly between different organizations and these differences give rise to a wide variety of scheduling problems and models (Ernst et al., 2004). However, the impact of these constraints on the complexity has barely been studied (Bergh et al., 2013; Ozder et al., 2020). Brucker et al. (2011) underpin the theory of personnel scheduling, which unlike in traditional scheduling, needs theoretical studies on models and complexity.

On the other hand, in the vast majority of scheduling problems, only machines are considered as resources and limited additional resources, such as operators, tools, pallets and industrial robots are not taken into account (Pinedo, 1995; Ventura et al., 2003). An extensive amount of research has been done on pure personnel scheduling (independent of machine scheduling), but little research has been done on models that combine personnel scheduling with machine scheduling. Some more theoretical research has been done in other areas related to these types of problems, namely resource constrained scheduling (i.e., a limited number of personnel may be equivalent to a constraining resource) (Pinedo, 2022).

The Resource Constrained Scheduling Problem (RCSP) is a subclass of scheduling problems and is mostly related to the Project Scheduling domain. In other words, scheduling problems that deal with personnel or workforce constraints are referred to as Resource Constrained Project Scheduling Problems (RCPSP) (Pinedo, 2007; Artigues et al., 2008). Details of RCPSP are beyond the scope of this paper and the interested reader is referred to Brucker et al. (1999) and Hartmann and Briskorn (2010).

Considering operators as additional resources in machine or project scheduling problems is a variant of the personnel scheduling problem. The working hours of operators can be considered as *doubly constrained* additional resource (both renewable and non-renewable) according to the regulations. In EU-TDSP, the daily driving time shall not exceed 9 hours and the weekly driving time shall not exceed 56 hours (The harmonization of certain social legislation relating to road transport and amending Council Regulations (EEC) No 3821/85 and (EC) No 2135/98 and repealing Council Regulation (EEC) No 3820/85, Regulation 561/2006). Thus, the driver's working hours are renewable on a daily basis without violating break and rest period rules but not on a weekly basis. Similarly, in ACSP, the maximum daily flight time shall not exceed 6 hours and the maximum monthly flight time shall not exceed 110 hours for rotary wing aircrafts according to DGCA (SHT-6A.50, 2014). The flight planning department may prepare flight plans on a daily basis without exceeding the monthly flight limit considering rest periods.

Although similar in some aspects to the personnel scheduling problems mentioned above, the problem considered in this paper is related to helicopter pilot scheduling and has a new and different constraint structure from them. The working hours of pilots are considered as doubly constrained resource. The processing times of the jobs vary depending on the day period (daytime and nighttime), the fatigue coefficient is taken into account in the workload calculation and the workload is categorized as normal and heavy based on total daily working hours. Consecutive days of heavy category work and night work require special rest periods. We are not aware of any study that includes this constraint structure at the same time.

Scheduling problems tend to be NP-hard structure. There are many solution methods in the personnel scheduling literature. These are classified into mathematical programming categories such as integer programming, linear programming, dynamic programming and goal programming, or as constructive or improvement heuristics. Other categories are simulation, constraint programming and queuing (Bergh et al., 2013). The solution methods can also be combined to increase the efficiency of the approaches. The personnel scheduling problem can be modeled as a linear, integer or mixed integer programming model. Many of the studies are modeled as integer and mixed integer programming (Ozder et al., 2020). Unfortunately, linear integer programming often requires a large number of variables and it is difficult to find the optimal or feasible solution in a reasonable time.

Our problem is formulated as an integer programming model and we propose the exact solution approach using commercial solver (CPLEX) in this paper. To obtain faster solutions and improve the solution performance, greedy constructive algorithms are implemented which both set upper bounds and generate warm-start points for the exact solution.

3. PROBLEM DEFINITION

Our problem is a variant of the personnel scheduling problem with organization-specific constraints inspired by a real-life problem. The aim is to schedule the flights of helicopter pilots on a warship under specific work and rest regulations.

Navy planning is a comprehensive process and critical at every level -strategic, operational, and tactical. Navy planning can be applied whether conditions permit a lengthy, deliberate process or if the situation forces a compressed timeline (Navy Planning NWP 5-01, 2013). Navy planning staff has to consider several factors. These include the disposition and number of platforms such as ships, aircraft, weapons, and supplies. These platforms have different capabilities. While warships can operate at sea for long periods, helicopters (also known as rotary-wing aircraft) can operate for relatively short periods.

Maritime helicopters can embark on ships that have flight decks for shipboard helicopter operations such as patrol, surveillance, search and rescue (SAR), humanitarian support, transportation, anti-submarine warfare, anti-surface warfare. Warships can carry different numbers and types of helicopters depending on their size and capacity. Additionally, when flight crews using helicopters are taken into account, the problem arises in different configurations. For example; one helicopter one flight crew, one helicopter two flight crews, two helicopters three flight crews. Since the number of helicopters in fleets is limited, it is not an easy task to assign helicopters and flight crews to each warship. To make planning easier, it is assumed that each ship will have a helicopter and a crew where possible.

'*One helicopter one flight crew*' configuration is studied. For the sake of generalization, it is assumed that helicopters are machines, pilots are operators and missions are jobs. Since the helicopters can fly for about 2,5-5 hours due to their fuel capacity, the processing times of jobs are also limited. It is assumed that jobs are divided into a small number of sets and the processing time of all jobs in the same class is identical. In other words, jobs have a high multiplicity structure. The objective is to minimize the makespan. This problem can be denoted by $1/NR/C_{max}$ using the three field notation of Graham et al.(1979) where NR stands for "non-renewable resource". It is NP-hard in the strong sense (Gafarov et. al, 2011).

The problem has similarities to NRP, TDSP and ACSP but introduces new types of constraints. To the best of our knowledge, this is the first personnel scheduling problem that includes the following constraints at the same time.

- Categorization of total working hours per day
- Consecutive working and rest periods depending on the category of total working hours per day
- Fatigue coefficient for night-period work
- Effects of night-period work on daytime-period

3.1. Problem Formulation

Let n be the total number of jobs and let g be the number of job types. Each job type k has n_k jobs for $k = 1, 2, \dots, g$ with $\sum_{k=1}^{g} n_k = n$.

Indices and Sets:

$k \in K$: Set of job types
$j \in J$: Set of jobs
$J_k \subset J$: Subset of job type $k \in K$
$t, t' \in T$: Set of time periods
$d, d' \in \mathrm{D}$: Set of days
$T_d \subset \mathrm{T}$: Set of day periods $d \in D$
$T_d^{daytime} \subset \mathrm{T}$: Set of daytime-periods $d \in D$
$T_d^{night} \subset \mathrm{T}$: Set of nighttime periods $d \in D$

Parameters:

p_j : Processing time of job $\boldsymbol{j}$ $\left(p_j \in \mathbb{Z} \middle| 1 \le p_j \le 3\right)$
α : Fatigue coefficient for night-period work $(\alpha \in \mathbb{R} \mid \alpha \ge 1)$
L^{month} : Maximum total working hour per month
L^{day} : Maximum total working hour per day
L^{normal} : Maximum total working hour per day for normal category
L^{night} : Maximum total working hour per night
N^{night} : Maximum total number of jobs per night
$L^{daytime}$: Maximum total working hour per daytime if night job is done
N^{day} : Maximum total number of jobs per day
$N^{daytime}$: Maximum total number of jobs per daytime if night job is done
R^{night} : Uninterrupted rest period before night job
$R^{night'}$: Uninterrupted rest period after the last night job
R^{heavy} : Uninterrupted rest period after consecutive heavy category workload

β : Maximum number of consecutive days of heavy category workload

γ : Maximum number of repetition of consecutiveness of heavy category workload cycle

M : A large number

LB : Lower bound

UB : Upper bound

Decision Variables:

$$x_{jt} = \begin{cases} 1, & \text{if job } j \text{ starts at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

$$y_t = \begin{cases} 1, & \text{if job } j \text{ starts at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

$$c_d = \begin{cases} 1, & \text{if job } j \text{ starts at time } t, \\ 0, & \text{otherwise.} \end{cases}$$

$$q_d = \begin{cases} 1, & \text{if uninterrupted rest period starts on day } d \text{ after consecutive heavy workload,} \\ 0, & \text{otherwise.} \end{cases}$$

C_{max} = Makespan of the schedule (Completion time of the last job)

Integer Programming Model (IP):

$$Minimize \ C_{max} \tag{1}$$

$Subject\ to$

$$\sum_{t \in T} x_{jt} = 1 \qquad \forall j \in J \tag{2}$$

$$\sum_{t'=t}^{t+p_j-1} y_{t'} \geq p_j x_{jt} \qquad \forall j \in J, t \in \{1,2,\cdots,(|T| - p_j + 1)\} \tag{3}$$

$$\sum_{t \in T} y_t = \sum_{j \in J} p_j \tag{4}$$

$$\sum_{j \in J} \sum_{t'=t-p_j}^{t-1} x_{jt'} \leq 1 \qquad \forall j \in J, t \in \{1,2,\cdots,(|T| - p_j + 1)\} \tag{5}$$

$$\sum_{t \in T_d^{night}} y_t \leq L^{night} \qquad \forall d \in D \tag{6}$$

$$\sum_{j \in J} \sum_{t \in T_d^{night}} x_{jt} \leq N^{night} \qquad \forall d \in D \tag{7}$$

$$\sum_{t \in T_d^{daytime}} y_t \leq L^{daytime} + (1 - y_{t'})\left(L^{day} - L^{daytime}\right) \quad \forall d \in D, \; t' \in T_d^{night} \tag{8}$$

$$\sum_{j \in J} \sum_{t \in T_d^{daytime}} x_{jt} \leq N^{daytime} + \left(1 - \sum_{j \in J} x_{jt'}\right)\left(N^{day} - N^{daytime}\right) \quad t' \in T_d^{night}, \; \forall d \in D \tag{9}$$

$$R^{night}\left(1 - x_{jt}\right) \geq \sum_{t'=t-R^{night}}^{t-1} y_{t'} \qquad j \in J, t \in T_d^{night}, \forall d \in D \tag{10}$$

$$R^{night'} y_t \leq \sum_{t'=t+1}^{t+R^{night'}} (1 - y_{t'}) + \sum_{t'=t+1}^{d*24} y_{t'} \qquad \forall t \in T_d^{night}, \forall d \in D \tag{11}$$

$$\sum_{t \in T_d^{daytime}} y_t + \alpha \sum_{t \in T_d^{night}} y_t \leq L^{normal} + \left(L^{day} - L^{normal}\right) c_d \quad \forall d \in D \tag{12.a}$$

$$\sum_{t \in T_d^{daytime}} y_t + \alpha \sum_{t \in T_d^{night}} y_t \geq L^{normal} \; c_d + \varepsilon \qquad \forall d \in D \tag{12.b}$$

$$\sum_{d'=d-\beta+1}^{d} c_{d'} - \sum_{d'=d-\beta+1}^{d-1} q_{d'} \geq (\beta - 1) + q_d \quad \forall d \in \{\beta, \beta+1, \cdots, |D|\} \quad (13.a)$$

$$M(1-q_d) + M\left\{1 - \left(y_t - \sum_{t'=t+1}^{d*24} y_{t'}\right)\right\} \quad \geq \sum_{t'=t+1}^{t+R^{heavy}} y_{t'} \quad (13.b)$$

$$t_d \in \{24(d-1) + L^{normal} + 1, 24(d-1) + L^{normal} + 2, \cdots, 24d\}, \forall d \in D$$

$$q_d = 0 \quad \forall d \in \{1, \cdots, \beta - 1\} \quad (13.c)$$

$$q_d \leq c_{d'} \quad \forall d \in \{\beta, \beta+1, \cdots, |D|\}, d' \in \{d-\beta+1, d-\beta+2, \cdots, d\} \quad (13.d)$$

$$\sum_{d'=d}^{d+\beta-1} q_{d'} \leq 1 \quad \forall d \in \{1,2,3, \cdots, |D| - \beta + 1\} \quad (13.e)$$

$$\sum_{d \in D} q_d \leq \gamma \quad (14)$$

$$\sum_{t \in T_d^{daytime}} y_t + \alpha \sum_{t \in T_d^{night}} y_t \leq L^{month} \quad \forall d \in D \quad (15)$$

$$(p_j + t)\, x_{jt} \leq C_{max} \quad \forall j \in J, \forall t \in \{1,2, \cdots, |T| - p_j + 1\} \quad (16)$$

$$LB \leq C_{max} \leq UB \quad (17)$$

$$C_{max} \in \mathbb{Z}^+ \quad (18)$$

$$x_{jt} \in \{0,1\} \quad \forall j \in J, \forall t \in T \quad (19)$$

$$y_t \in \{0,1\} \quad \forall t \in T \quad (20)$$

$$c_d \in \{0,1\} \quad \forall d \in D \quad (21)$$

$$q_d \in \{0,1\} \quad \forall d \in D \quad (22)$$

As seen from the mathematical model our problem is formulated as an integer linear programming model. The objective function (1) minimizes the makespan, in other words, completion time of the last job. Constraint (2) requires that all jobs must be scheduled. Constraint (3) ensures that the operator is busy during the processing time. Constraint (4) imposes that the operator cannot be busy more than total processing time of jobs. Constraint (5) ensures that at most one job can be processed at any point in time. Constraint (6) limits total processing time of night jobs and Constraint (7) limits the total number of night jobs. Constraint (8) defines the maximum total working hour per daytime and Constraint (9) defines the maximum total number of jobs per daytime if night job is done. Constraint (10) enforces the minimum uninterrupted rest period before night job and Constraint (11) enforces the minimum uninterrupted rest period after the last night job. Constraints (12.a) and (12.b) determine the daily workload (normal or heavy) while defining the daily maximum total working hour. Constraint (13) enforces the minimum uninterrupted rest period after the consecutive heavy category workload. Constraint (13.a) determines the day that uninterrupted rest period starts after the consecutive heavy category workload while Constraint (13.b) determines the hour. Constraints (13.c), (13.d) and (13.e) are the technical constraints related to heavy category workload days and their consecutiveness. Constraint (14) limits the maximum number of repetition of consecutiveness of heavy category workload cycle. Constraint (15) defines the monthly maximum total working hour. Constraint (16) is used to compute the makespan within the lower bound and upper bound specified in Constraint (17). The calculation of the lower bound and upper bound values will be explained in detail in the next section. Constraints (18) - (22) declare decision variable domains. All of the decision variables except C_{max} are binary variables.

Assumptions:

The time unit is one hour and the scheduling horizon is up to one month. One month has 30 days and one day has 24 hours. Day is the period from sunrise to sunrise the next day. Daytime is the period between sunrise and sunset, night is the period between sunset and sunrise. Daytime and night equal 12 hours every day for simplicity. All of the parameter values except from fatigue coefficient α are positive integers. Due to operating restrictions of the machine, there are three job type according to deterministic processing times (1, 2 and 3 hour). The schedule is empty and all the jobs are available at time zero. There are no *machine non-availability* (MNA) and *operator non-availability* (ONA) intervals. The machine and the operator are available throughout the scheduling period without violating work and rest regulations. The machine can process only one job at a time. No preemption is allowed. A job, once taken up, is fully completed before the next is taken.

4. SOLUTION APPROACHES

Basically, we propose exact solution approach using commercial solver (CPLEX) to our integer programming problem. As a result of discretizing time, the model creates huge number of variables depending on size of the problem. So determining cardinality of time set $(|T|)$ is a critical step. Two greedy constructive heuristics that adapted from Offline Bin Packing Problem (BPP) algorithms have been used for this step. As it is known, computationally BBP is NP-hard and for this reason many approximation algorithms developed for getting faster solutions. Solutions from the heuristic approaches set both the upper bound and warm-start point for exact solution approach.

4.1. Greedy Constructive Heuristics

Days can be considered as bins and the capacities of the bins can be defined as working hours. *First-fit-decreasing (FFD)* and *First-fit-increasing (FFI)* algorithms are adapted for constructing feasible solution without violating work and rest regulations.

4.1.1. Greedy Constructive Heuristic (GR1)

In $GR1$ the capacity of bins is the maximum total working hour per day for normal category $\left(L^{normal}\right)$. If operator works for normal category each day, constraints related to heavy workload category become redundant, just as the constraints related to consecutiveness become redundant. Certainly, this increases the planning horizon and provides bad objective function value (C_{max}). The reason for overestimating the planning horizon is to investigate whether it has an impact on warm-start approach. The pseudocode of the $GR1$ is given in Algorithm 1.

List of jobs can be sorted according to the two different ordering criteria: descending and ascending. So, two upper bound values $(UB[GR1^{dec}], UB[GR1^{inc}])$ and two solution sets $(sol[GR1^{dec}], sol[GR1^{inc}])$ can be obtained. Minimum of the upper bounds and its associated solution is chosen C_{max} for $GR1$ using Equation (23).

$$GR1_{Cmax} = min(UB[GR1^{dec}], UB[GR1^{inc}]) \tag{23}$$

4.1.2. Greedy Constructive Heuristic (GR2)

In $GR2$ the capacity of bins is the maximum total working hour per day for heavy category (L^{heavy}). But for this time, constraints related to night jobs and heavy workload category step in. Algorithm 1 is modified to check solution feasibility as each job is scheduled. The modified algorithm also produces two upper bound values $(UB[GR2^{dec}], UB[GR2^{inc}])$ and two solution sets $(sol[GR2^{dec}], sol[GR2^{inc}])$ according to the ordering criteria. Minimum of the upper bounds and its associated solution is chosen C_{max} for $GR2$ using Equation (24).

$$GR2_{Cmax} = min(UB[GR2^{dec}], UB[GR2^{inc}]) \tag{24}$$

$GR2$ mostly has better objective function values than $GR1$ and provides tighter upper bounds. In exceptional problem instances, $GR2$ cannot find a feasible solution in a monthly planning horizon. This is one of the already known side effects of the greedy approach.

Algorithm 1: $FFD_{L^{normal}}$. Pseudocode of $GR1$ for determining upper bound and solution for warm-start.

Input: List of jobs sorted in *decreasing* order according p_j, L^{normal}
Output: $|T|, sol$

1 $sol \leftarrow \emptyset$
2 $d \leftarrow 1$
3 $workload(d) \leftarrow 0$
4 **for** *each job* $j \in J$ **do**
5   **for** *each* $d \in D$ **do**
6   **if** $workload(d) \leq L^{normal}$
7   **for** *each* $t \in T^d$ **do**
8   **if** $\sum_t^{t+p_j-1} y_t = 0$ **and** $t + p_j - 1 < d * 24$ **then**
9   **if** $p_j + workload(d) \leq L^{normal}$ **then**
10                         $x_{jt} = 1$
11                         $sol.$insert(x_{jt})
12                         $workload(d) \leftarrow p_j + workload(d)$
13   **break**
14   **end if**
15   **end if**
16   **end for**
17   **end if**
18   **if** $|sol| = j$ **then**
19   **break**
20   **end if**
21   **end for**
22   **if** $|sol| < j$ **then**
23         $|D| \leftarrow |D| + 1$ *//add new day*
24   **goto** line 5
25   **end if**
26 **end for**

4.2. Exact Solution Approach

The exact solution approach (IP) is applied in four configurations using the output of the greedy constructive heuristics $GR1$ and $GR2$ as shown in Table 1.

Table 1. *Exact Solution Configurations.*

Name	Description
$IP1$	$GR1_{Cmax}$ is used for *Upper Bound* value
$IP2$	$GR2_{Cmax}$ is used for *Upper Bound* value
$WS1$	$sol(GR1_{Cmax})$ is used as solution set for *Warm-Start* point
$WS2$	$sol(GR2_{Cmax})$ is used as solution set for *Warm-Start* point

It is observed that the solver cannot reach a feasible solution for large-size problem instances in reasonable computational times. It spends much time on presolving the model and solving the root node linear programming (LP) relaxation. To overcome this problem, lower bound (LB) and upper bound (UB) values are calculated and the warm-start technique is applied to the exact solution approach. As known, warm-start may sometimes improve the performance of the solver even though it is not guaranteed. The performance comparison of the exact solution configurations is presented in the computational experiments section.

Assuming that no rest period is allowed and operator can work heavy category every day, a safe lower bound (LB) has been formulated in Equation (25).

$$LB = \left\lfloor \left(\left\lceil \frac{\sum p_j}{L^{day}} \right\rceil - 1 \right) * 24 + \left\{ \sum p_j - \left[\left(\left\lceil \frac{\sum p_j}{L^{day}} \right\rceil - 1 \right) * L^{day} \right] \right\} \right\rfloor \tag{25}$$

5. COMPUTATIONAL EXPERIMENTS

We have performed computational experiments to compare the performance of the solution approaches. Since the problem is organization-specific and involves custom constraints, there are no available datasets in the literature for benchmarking purposes. Therefore, test instances are simply generated by randomly selecting a number of jobs for each job type. The naming convention for the test instances is shown in Figure 1.

$tP30\text{-}j16\text{-}(07\text{-}04\text{-}05)$

$|J^3|$: # of jobs with processing time 3 hour
$|J^2|$: # of jobs with processing time 2 hour
$|J^1|$: # of jobs with processing time 1 hour
$|J|$: Total # of jobs
$\sum p_j$: Total processing time of all jobs

Figure 1. *Test instance naming convention.*

Depending on the number of jobs in each job type, there may be test instances with the same total processing time but different total number of jobs, and test instances with the same total number of jobs but different total processing time. $tP30\text{-}j10\text{-}(00\text{-}00\text{-}10)$ have the same total processing time but different total number of jobs with $tP30\text{-}j16\text{-}(07\text{-}04\text{-}05)$. $tP40\text{-}j16\text{-}(02\text{-}04\text{-}10)$ have different total processing time but the same total number of jobs with $tP30\text{-}j16\text{-}(07\text{-}04\text{-}05)$. The values of the parameters used in the experiments are given in Table 2.

Table 2. *Parameter values for experiments.*

α	=	1.5	N^{day}	=	8	R^{heavy}	=	24
L^{month}	=	60	$N^{daytime}$	=	3	β	=	2
L^{day}	=	8	N^{night}	=	2	γ	=	2
L^{normal}	=	5	R^{night}	=	2			
L^{night}	=	3	$R^{night'}$	=	8			

Greedy constructive heuristic algorithms are coded using the C# programming language in the Visual Studio 2022 platform. All of the IP models are coded and solved using IBM ILOG CPLEX 22.1 with default optimality gap settings of (0.01%) and a CPU time limit of one hour. Each test instance was solved in 4 configurations; $IP1$ and $IP2$ with the same lower bound but different upper bounds, $WS1$ and $WS2$ with different warm-start points. Both the greedy heuristic algorithms and CPLEX are run on an Intel i7 2.2 GHz 8 GB RAM computer.

Computational results are shown in Table 3. The table is divided into eight main columns. The first main column is the name of the test instance. The second main column is the LB value. The third and fourth main columns show the solutions (UB) and the computation time of the greedy constructive algorithms $GR1$ and $GR2$. The remaining four columns show the solution (C_{max}), computation time (t) and gap (g) values for the exact solution configurations $IP1$, $IP2$, $WS1$ and $WS2$, respectively. The solution values are in hours, the computation time values are in seconds and the gap values are in percent. The star symbol near C_{max} values indicates optimal solutions. The dagger symbol in the computation time columns indicates that the solver was interrupted and no optimal solution was found within the time limit. Lastly, the double dagger symbol in C_{max} columns means that no solution was found within the time limit.

The computation times of $GR1$ and $GR2$ are less than one second. For small-size problems, all exact solution approaches show almost similar performance in finding the optimal solution in a relatively short time. For long total processing time problems consisting of long processing time jobs, although the total number of jobs is relatively small, the optimal solution is not found within the time limit. As the total processing time of the jobs increases the solver fails to find an optimal solution. As expected, in large-size problems, the constraints related to consecutive heavy category workload and rest periods start to activate.

$IP2$ shows relatively better performance than $IP1$. Tight upper bounds obtained by $GR2$ seem to help improve the solution. However, sometimes, as in problem instance $tP48$-$j16(00$-00-$16)$, $IP2$ cannot find a solution while $IP1$ finds a solution with a looser upper bound. When tight upper bounds are set for problem instances consisting of all or most of the jobs with the longest processing time, the solver has difficulty finding a feasible solution.

$WS1$ and $WS2$ show similar performance. So, the warm-start technique does not seem to provide a very significant improvement in computational efficiency. However, it at least provides a feasible solution where no solution can be found in a reasonable time.

Table 3. *Computational results.*

Instance	LB	$GR1$		$GR2$		$IP1$			$IP2$			$WS1$			$WS2$		
		UB	t	UB	t	c_{max}	t	g	c_{max}	t	g	c_{max}	t	g	c_{max}	t	g
tP40-j40(40-00-00)	104	173	0.003	**132**	0.016	**126***	542	0.00	**126***	653	0.00	**126***	**444**	0.00	**126***	1188	0.00
tP40-j20(00-20-00)	104	220	0.003	**124**	0.013	**124***	1418	0.00	**124***	**525**	0.00	**124***	1175	0.00	**124***	2400	0.00
tP40-j14(01-00-13)	104	291	0.003	**150**	0.014	171	†	26.90	‡	†	-	**150**	†	18.00	**150**	†	14.00
tP40-j20(05-10-05)	104	173	0.003	**125**	0.015	**124**	†	0.81	125	†	0.80	**124**	†	0.81	**124**	†	0.81
tP40-j22(10-06-06)	104	173	0.003	**145**	0.017	125	†	1.61	125	†	5.67	**124**	†	2.42	**124**	†	2.42
tP48-j48(48-00-00)	128	219	0.003	**152**	0.018	**151**	†	0.66	**151***	2149	0.00	**151***	**649**	0.00	**151***	1592	0.00
tP48-j24(00-24-00)	128	268	0.003	**152**	0.014	**152**	†	1.97	**152***	1900	0.00	**152**	†	1.97	**152**	†	1.97
tP48-j16(00-00-16)	128	363	0.003	**198**	0.014	**198**	†	22.73	‡	†	-	**198**	†	22.22	**198**	†	23.74
tP48-j24(08-08-08)	128	219	0.003	**169**	0.017	**151**	†	2.65	**151**	†	1.99	**151**	†	2.65	**151**	†	2.65
tP60-j60(60-00-00)	172	269	0.003	**199**	0.021	**198**	†	1.01	**198***	**1724**	0.00	**198**	†	0.51	**198***	2870	0.00
tP60-j20(00-00-20)	172	459	0.003	**723**	0.017	**267**	†	7.12	723	†	76.21	**267***	**520**	32.99	723	†	76.21
tP60-j30(00-30-00)	172	**340**	0.003	722	0.016	220	†	20.45	243	†	29.22	**200**	†	2.00	218	†	21.10
tP60-j30(10-10-10)	172	269	0.003	**218**	0.022	**198**	†	2.02	**198**	†	2.02	200	†	3.00	200	†	3.00

* Optimal. † Run terminated after 1 hour. ‡ No solution found in 1 hour.

6. CONCLUSION

In this paper, we consider workload dependent resource constrained scheduling problem with organization-specific work and rest regulations. The problem has custom constraints different from other personnel scheduling problems.

Exact solution approach using the commercial solver (CPLEX) is proposed to solve the problem. Due to its NP-hardness of the problem, the solver could not yield an optimal solution within a reasonable solution time, especially for large-size problem instances. In order to obtain faster solutions, we implemented modified versions of the greedy constructive heuristic algorithms for the BPP. The solutions obtained from the heuristics are used as upper bounds as well as warm-start points for the exact solution approach. Heuristic algorithms are able to find feasible solutions in a very short time. The warm-start technique does not significantly improve the performance, but may provide a feasible solution for some of the problem instances where the solver cannot.

The performance of the exact solution is affected by the distribution of high multiplicity. Although problem instances have the same total processing time, the solver cannot find a feasible solution for some of them. This is also the case for the problem instances that have the same number of jobs with different total processing times. When real data is available and the high multiplicity distribution is known, the effectiveness of the solution approaches can be evaluated more realistically by running the problem instances with real data.

Planners can use this study to determine how the schedule will be affected by changing parameter values, such as increasing workload category limits or reducing rest periods. Further studies can be addressed to investigate other solution approaches (metaheuristics, constraint programming, etc.) for this problem and to consider other machine-operator configurations such as 'one machine n operator', 'm machine n operator'.

ACKNOWLEDGEMENT

The authors declare no conflict of interest.

REFERENCES

Artigues, C., Demassey, S. & Neron, E. (2008). *Resource Constrained Project Scheduling: Models, Algorithms, Extensions and Applications*. France, John Wiley & Sons.

Barnhart, C., Cohn, A.M., Johnson, E.L., Klabjan, D., Nemhauser, G.L. & Vance, P.H. (2003). *Airline Crew Scheduling. In: Hall, R.W. (eds) Handbook of Transportation Science. International Series in Operations Research & Management Science, Vol 56*. Springer, Boston, MA. doi:10.1007/0-306-48058-1_14.

Bergh, J.V., Beliën, J., Bruecker P., Demeulemeester, E. & Boeck, L. (2013). "Personnel scheduling: A literature review". *European Journal of Operational Research*, Vol. 226, pp. 367–385. doi:10.1016/j.ejor.2012.11.029.

Brucker, P., Drexl, A., Möhring, R., Neumann, K. & Pesch, E. (1999). "Resource-constrained project scheduling: Notation, classification, models, and methods". *European Journal of Operational Research*, Vol. 112, pp. 3-41. doi:10.1016/S0377-2217(98)00204-5.

Brucker, P., Qu, R., & Burke, E. (2011). "Personnel scheduling: Models and complexity". *European Journal of Operational Research*, Vol. 210, pp. 467-473. doi:10.1016/j.ejor.2010.11.017.

Burke, E. K., De Causmaecker, P., Berghe, G. V. & Van Landeghem, H. (2004). "The state of the art of nurse rostering". *Journal of Scheduling*, Vol. 7, pp. 441-499. doi: 10.1023/B:JOSH.0000046076.75950.0b.

Dantzig, G.B. (1954). "Letter to the Editor---A Comment on Edie's 'Traffic Delays at Toll Booths'". *Journal of the Operations Research Society of America*, Vol. 2, No. 3, pp. 339-341. doi:10.1287/opre.2.3.339.

Ernst, A.T., Jiang, H., Krishnamoorthy, M. & Sier, D. (2004). "Staff scheduling and rostering: a review of applications, methods and models". *European Journal of Operational Research,* Vol. 153, pp. 3-27.

Gafarov, E. R., Lazarev, A. A., & Werner, F. (2011). "Single machine scheduling problems with financial resource constraints: Some complexity results and properties". *Mathematical Social Sciences*, Vol. 62(1), pp. 7-13. doi:10.1016/ j.mathsocsci.2011.04.004.

Goel, A. (2009). “Vehicle Scheduling and Routing with Drivers’ Working Hours”. *Transportation Science,* Vol. 43(1), pp. 17-26. doi: 10.1287/trsc.1070.0226.

Goel, A. (2010). “Truck driver scheduling in the European Union”. *Transportation Science,* Vol. 44(4), pp. 429-441. doi:10.1287/trsc.1100.0330.

Goel, A., Archetti, C. & Savelsbergh, M. (2012). “Truck driver scheduling in Australia”. *Computers & Operations Research,* Vol. 39, pp. 1122-1132. doi:10.1016/j.cor.2011.05.021.

Goel, A. & Kok, L. (2012). “Truck Driver Scheduling in the United States”. *Transportation Science*, Vol. 46 (3), pp. 317-326. doi:/10.1287/trsc.1110.0382.

Graham, R. L., Lawler, E. E., Lenstra, J. K., & Kan, A. H. G. R. (1979). “Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey”. *In Annals of Discrete Mathematics,* Vol. 5, pp. 287–326. doi:10.1016/s0167-5060(08)70356-x.

Hartmann, S. & Briskorn, D. (2010). “A survey of variants and extensions of the resource-constrained project scheduling problem”. *European Journal of Operational Research*, Vol. 207 (1), pp. 1-14. doi:10.1016/j.ejor.2009.11.005.

Kok, A.L., Meyer, C.M., Kopfer, H.,Schutten, J.M.J. (2010). “A Dynamic Programming Heuristic for the Vehicle Routing Problem with Time Windows and the European Community Social Legislation”. *Transportation Science*, Vol. 44, Issue 4, pp. 442–454. doi:10.1287/trsc.1100.0331.

Özder, E., Özcan, E., & Eren, T. (2020). “A Systematic Literature Review for Personnel Scheduling Problems”. *International Journal of Information Technology and Decision Making,* Vol. 19, pp. 1695-1735. doi:10.1142/S0219622020300050.

Pinedo, M.L. (1995). *Scheduling: Theory, Algorithms and Systems*. Prentice-Hall, Englewood Cliffs.

Pinedo, M.L. (2007). *Planning and Scheduling in Manufacturing and Services (Springer Series in Operations Research and Financial Engineering)*. New York, Springer.

Pinedo, M.L. (2022). *Scheduling: Theory, Algorithms and Systems (6th ed.)*. Springer International Publishing. doi:10.1007/978-3-031-05921-6.

Portugal, R., Lourenço, H.R. & Paixão, J.P. (2009). "Driver scheduling problem modelling". *Journal of Public Transport,* Vol. 1, pp. 103–120. doi:10.1007/s12469-008-0007-0.

Regulation 561/2006. *The harmonisation of certain social legislation relating to road transport and amending Council Regulations (EEC) 3821/85 and (EC) 2135/98 and repealing Council Regulation (EEC) 3820/85*. European Parliament, Council of the European Union. http://data.europa.eu/eli/reg/2006/561/oj.

SHT-6A.50. (2014). *Instruction on the Flight, Working and Rest Time of Flight Crew and its Implementation Rules*. Directorate General of Civil Aviation. http://web.shgm.gov.tr/doc5/sht6arev06.pdf.

US Department of the Navy. (2013). *Navy Planning, NWP 5-01*. Washington, DC: Headquarters US Navy.

Ventura, J.A. & Kim, D. (2003). "Parallel machine scheduling with earliness-tardiness penalties and additional resource constraints". *Computers & Operations Research*, Vol. 30, pp. 1945-1958. doi:10.1016/S0305-0548(02)00118-1.