Asia-Pacific Journal of Operational Research 17 (2000) 137-148

DUAL-BASED OPTIMIZATION OF CYCLIC THREE-DAY WORKWEEK SCHEDULING

Hesham K. ALFARES
System Engineering Department
King Fahd University of Petroleum & Minerals
Dhahran 31261, Saudi Arabia
email: hesham@ccse.kfupm.edu.sa

An optimization method is presented for the cyclic labour scheduling problem, in which workers are given three consecutive workdays per week. The dual solution is utilized to calculate the minimum workforce size and days-off assignments that minimize labour cost. Using simple manual calculations, the need for integer programming is eliminated.

Keywords: Workforce scheduling, optimization, integer programming, labour planning, staffing.

1. Introduction and Background

Demand for an organization's goods or services usually varies according to season, day of week, or even time of the day. Organizations must allocate resources, including labour, to satisfy these varying customer demands effectively and efficiently. Labour is usually the most important and expensive resource of any organization. Effective labour scheduling allows businesses to satisfy customers with the minimum labour cost.

Days-off scheduling is an important and practical problem that applies to organizations operating seven days a week, such as airlines, restaurants and hospitals. Since workers must be given weekly breaks, they must be assigned to specific days-off work patterns. The most common type of days-off work patterns includes five workdays and two consecutive off days per week, thus it is usually referred to as the (5, 7) problem. Recently, there has been a lot of interest in alternative work schedules, including 3-day and 4-day workweeks. This is illustrated by Browne and Nanda (1987), Burns *et al.* (1998), Hung (1993, 1994), Hung and Emmons (1993), and Steward and Larsen (1971).

This paper applies a methodology similar to that used by Alfares (1998) for the 5-day workweek (5, 7) problem, to the 3-day workweek or the (3, 7) problem. It assumes that each employee is given three consecutive workdays and four consecutive off days per week. A simple, yet optimum solution

Received October 1999, revised version received June 2000.

method is presented. Using this method, the solution can be easily obtained manually. The method can be used to obtain either the minimum number or the minimum cost of the workforce. To find the optimum primal solution, the method utilizes the dual solution and primal-dual relations.

In solving the (3, 7) problem, the new method offers two advantages over integer programming (IP): simplicity and computational efficiency. While IP is a specialized technique which requires training and availability of certain software packages, the new method can be implemented manually. If programmed, the new algorithm should be much faster than integer programming since it requires only substitutions in simple formulae, and no iterations. Computational efficiency is a significant benefit in practical applications that require the solution of a large number of days-off problems, such as the simultaneous scheduling of operations and labour for projects (Alfares and Bailey, 1997) and airlines (Vance *et al.*, 1997).

This paper is organized as follows. First, a review of relevant literature is given. Then, the integer programming models of the problem and its dual are presented. Subsequently, the procedures for determining the minimum workforce size and assigning workers to days-off patterns are described. Finally, a numerical example is solved and conclusions are given.

2. Survey of Literature

Baker (1976) classifies labour scheduling problems into three types: (1) shift, or time-of-day, scheduling, (2) days-off, or days-of-week scheduling, and (3) tour scheduling, which combines the first two types. Baker (1976), Tien and Kamiyama (1982), and Nanda and Browne (1992) provide comprehensive surveys of literature on all these types. The scope of this review is limited to the days-off scheduling problem, with emphasis on non-traditional work schedules and compressed workweek scheduling.

Several approaches have been developed for the (5, 7) problem in which only consecutive pairs of off days are allowed. Tiberwala *et al.* (1972) develop a procedure in which the number of iterations equals the number of workers required. Browne and Tiberwala (1975) simplify the three steps involved, but do not reduce the number of iterations. Baker (1974) develops a two-phase algorithm which starts by calculating the lower bound on workforce size, then uses trial and error to determine days-off assignments. Morris and Showalter (1983) describe an iterative, three-step cutting plane procedure to optimally solve the (5, 7) days-off problem. Another iterative, manual procedure utilizing three simple rules was developed by Bechtold and Showalter (1985). The objective of all these procedures is to minimize the workforce size.

Various approaches have been applied to different other versions of the days-off scheduling problem. Bechtold (1988) develops heuristics for assigning full- and part-time employees in multiple-objective, multiple-location environments. Emmons (1985) and Koop (1986) schedule a single type of workers, with 2 days off per week and A out of B weekends off. Emmons and Burns (1991), and Narasimhan (1996) consider the same problem with m types of workers. Emmons and Burns (1991) assume a constant labour demand for all days of the week, while Emmons (1985) and Narasimhan (1996) assume two different demand levels: D for regular weekdays and E for weekends.

Browne and Nanda (1987) analyze the efficiency of 4-day workweek scheduling. Hung (1993, 1994) develops multiple-shift models for both 3-day and 4-day workweeks. Hung and Emmons (1993), and Steward and Larsen (1971) model the 3-4 workweek in which each worker works three days in one week and four days in the other during a 2-week cycle. Hung assumes D workers are required during weekdays and E workers during weekends, while Hung and Emmons (1993) assume D workers are required every day, but all minimize workforce size. In comparison, the model presented below does not assume labour demands to be constant for weekdays or weekend days, and it does not assume days-off work pattern costs to be equal, thus it can minimize workforce size or cost.

3. Integer Programming Models

The (3, 7) labour scheduling problem can be represented as an integer programming model as follows:

$$\text{Minimize } W = \sum_{j=1}^{7} x_j \tag{1}$$

subject to

$$\sum_{j=1}^{3} x_{i+j} \geq r_i \,, \quad i = 1, 2, ..., 7 \tag{2}$$

$$x_j \geq 0, \quad j = 1, 2, ..., 7 \tag{3}$$

where

W = workforce size, i.e. total number of workers assigned to days-off patterns.

x_j = number of workers assigned to weekly days-off pattern j, i.e. number of workers off on days j, $j+1$, $j+2$, and $j+3$.

r_i = number of workers required on day i, $i = 1, 2, ..., 7$.

Note:

Since the problem has a one-week cycle, all the subscripts are modular 7.

Since $\sum_{j=1}^{7} x_j$ is equal to W, (2) can be written as

$$\sum_{j=1}^{4} x_{i-j+1} \leq b_i \ , \quad i = 1, 2, ..., 7 \tag{4}$$

where

$$b_i = W - r_i \ , \qquad i = 1, 2, ..., 7 \tag{5}$$
$$= \text{number of workers off on day } i$$

The dual of the (3, 7) days-off scheduling model with dual variables y_i, $i = 1, 2, ..., 7$, is given by

$$\text{Maximize } \ W = \sum_{i=1}^{7} r_i y_i \tag{6}$$

subject to

$$\sum_{j=1}^{3} y_{i-j} \leq 1, \quad i = 1, 2, ..., 7 \tag{7}$$

$$y_i \geq 0, \qquad j = 1, 2, ..., 7 \tag{8}$$

4. Determining the Minimum Workforce Size

Given seven daily labour demands $r_1, ..., r_7$, the minimum workforce size W can be easily obtained using the dual formulation shown above, without integer programming. An optimum solution to the above dual problem is a feasible solution to the primal (original) days-off scheduling problem. Moreover, the value of the *optimal* objective function W is the same for the two problems. To solve the dual problem we allocate the unit resource (right hand size of (7) equal to 1) among the dual variables in order to maximize the dual objective W which is a linear combination of labour demands. There are two possible dual solutions, depending on the given labour demands $r_1, ..., r_7$, which are discussed next.

1. Since each constraint (7) contains only three dual variables, it is possible to divide the unit right-hand side of each constraint among those three variables. Thus, it is feasible to assign a value of 1/3 to all *seven* dual variables. Since $y_i = 1/3$ for $i = 1, ..., 7$, the workforce size $W = \sum ry$ is equal to $\sum r/3$.

2. A similar situation pertains to seven sets $s_i = \{y_i,\ y_{i+3}\},\ i = 1, ..., 7$. Considering modularity, at least one of any four *adjacent* variables must belong to a set s_i. Therefore, at least one of the four *adjacent* variables absent from constraints (7), representing four successive off days, must belong to a set $\{y_i,\ y_{i+3}\}$. Since a maximum of one variable from each set is present in any constraint, it is possible to assign a value of 1 to *both* variables in a given set. Defining $S_i = r_i + r_{i+3}$, we would choose the set k which has the maximum total demand $S_{\max} = S_k$. Then, $y_k = y_{k+3} = 1$, all other $y_i = 0$, and $W = r_k + r_{k+3}$.

To determine the workforce size, we choose the maximum value of W obtained from the two above cases, and must also round up W in case it is not an integer. Therefore, we obtain the following expression for the minimum W:

$$W = \max\ \{\lceil \frac{1}{3} \sum_{i=1}^{7} r_i \rceil\ ,\ S_{\max}\} \tag{9}$$

where

$S_{\max} = \max\ \{S_1, S_2, ..., S_7\}$

$\lceil a \rceil$ = smallest integer $\geq a$, i.e. a rounded up to the nearest integer

and

$$S_i = r_i + r_{i+3}\ ,\quad i = 1, 2, ..., 7 \tag{10}$$

5. Assigning Workers to Days-off patterns

5.1. Minimizing total labour cost

This section will illustrate how minimum-cost days-off assignments x_1, ..., x_7 are found for the (3, 7) problem. Having determined the minimum workforce size W by (9), the objective now is to assign workers to days-off work patterns in order to minimize total cost. The costs of different days-off work patterns are related to the number of overtime-paid weekend workdays. Obviously, patterns 4, 5 and 6 are the cheapest with no weekend workdays, followed by patterns 3 and 7 with one weekend workday each, while patterns 1 and 2 are the most expensive with two weekend workdays each. Let us assume each worker costs A per regular-week workday and $A(1 + B)$ per weekend workday $(A, B \geq 0)$. The weekly costs of the seven days-off patterns are given by

$$c_1, ..., c_7 = (3 + 2B,\ 3 + 2B,\ 3 + B,\ 3,\ 3,\ 3,\ 3 + B)A \tag{11}$$

where

c_j = weekly cost per work of days-off pattern j

If these varying costs of different days-off patterns were taken into consideration, the objective would be to minimize the total *cost* of workers, or

$$\text{Minimize } \; Z = \sum_{j=1}^{7} c_j x_j \tag{12}$$

To incorporate differential costs, the right-hand side of the dual constraints (7) changes to the transposed cost vector $(c_1, ..., c_7)^T$. Naturally, the dual solutions corresponding to cases when S_i is maximum do slightly change, however, the minimum workforce W obtained by (9) does not change. This means that for the cost structure defined by (11), the minimum labour cost Z is always obtained with the minimum number of workers W.

To attain objective (12), we must assign as many workers as possible, out of W determined by (9), to the cheapest days-off patterns: x_4, x_5 and x_6 then to x_3 and x_7. This value of W is based on the optimum dual solution. Therefore, complementary slackness primal-dual relationships will be used to obtain the solution of the primal (original) days-off scheduling problem. The solution will depend on which argument of the right-hand side of (9) is maximum, thus there are two possible cases.

5.2. $\lceil \frac{1}{3} \sum_{i=1}^{7} r_i \rceil$ is maximum

In this case, $W = \lceil \sum r/3 \rceil$, all dual variables are basic ($y_i = 1/3$, $i = 1, 2, ..., 7$), and all dual constraints are equations. Therefore, all primal variables are also basic and all primal constraints are equations. Constraint system (4) is transformed into the following set of equations:

$$\sum_{j=1}^{4} x_{i-j+1} = b_i \;, \quad i = 1, 2, ..., 7 \tag{13}$$

The solution of the 7×7 linear system of equations is given by

$$x_i = W - S_i \;, \qquad i = 1, 2, ..., 7 \tag{14}$$

5.3. $S_{\max}$ is maximum

Let $S_i = S_{\max}$. In this case $W = S_i$, dual variables $\{y_i, \; y_{i+3}\} \cup \{y_6, \; y_7\}$ are basic, and dual constraint i is an inequality. Thus, the primal problem has three or four equations in the ith, $(i+3)$rd, 6th and 7th constraints and the variable x_i is equal to zero. Since weekend work is more expensive, constraints 6 and 7 are always satisfied as equations indicating that the number of workers assigned on weekends equals but never exceeds the number required. The optimum solution for each $i (i = 1, ..., 7)$ is found by assigning as many workers as possible to the cheapest days-off patterns. For example, if S_1 is maximum, then $x_1 = 0$ and constraints 1, 4, 6 and 7 are equations. Including this information in system (4) and simplifying, we obtain

$$\begin{array}{lllllllllllllr}
 & & & & & & & & x_5 & + & x_6 & + & x_7 & = b_1 \quad (15.1)\\
x_2 & & & & & & & & & + & x_6 & + & x_7 & \le b_2 \quad (15.2)\\
x_2 & + & x_3 & & & & & & & & & + & x_7 & \le b_3 \quad (15.3)\\
x_2 & + & x_3 & + & x_4 & & & & & & & & & = b_4 \quad (15.4)\\
 & & & & & & & & x_5 & & & & & \le b_5 - b_4 \quad (15.5)\\
 & & x_3 & + & x_4 & + & & & x_5 & + & x_6 & & & = b_6 \quad (15.6)\\
 & & & & x_4 & & & & & & & & & = b_7 - b_1 \quad (15.7)
\end{array}$$

Note that introducing differential costs does not change the workforce size W obtained by (9). Since $x_1 = 0$, W is given by the sum of (15.1) and (15.4) as

$$W = b_1 + b_4 = (W - r_1) + (W - r_4) = 2W - (r_1 + r_4) = 2W - S_1$$

thus $W = S_1$

Subtracting (15.6) from the sum of (15.1) and (15.4), we obtain

$$x_2 + x_7 = b_1 + b_4 - b_6 = W - b_6$$

but from (15.2),

$$x_6 \le b_2 - x_2 - x_7$$

Therefore,

$$x_6 \le b_6 + b_2 - W = (W - r_6) + (W - r_2) - W = W - (r_6 + r_2)$$

or

$$x_6 \le W - S_6$$

By maximizing x_6 and x_5, the solution is given by

$$\begin{aligned}
x_4 &= b_7 - b_1\\
x_6 &= \min\{b_1,\ W - S_6,\ b_6 - x_4\}\\
x_5 &= \min\{b_1 - x_6,\ b_5 - b_4,\ b_6 - x_4 - x_6\}\\
x_7 &= b_1 - x_5 - x_6\\
x_3 &= b_6 - x_4 - x_5 - x_6\\
x_2 &= b_4 - x_3 - x_4
\end{aligned}$$

Solutions for the remaining cases, S_i is maximum, $i = 2, ..., 7$, can be obtained similarly. Days-off assignments corresponding to each case are shown in Table 1.

Table 1. Values of days-off assignments, $x_1, \ldots, x_7$, for all possible values of W

W	x_1	x_2	x_3	x_4	x_5	x_6	x_7
$\lceil \Sigma r/3 \rceil$	$W - r_1 - r_4$	$W - r_2 - r_5$	$W - r_3 - r_6$	$W - r_4 - r_7$	$W - r_1 - r_5$	$W - r_2 - r_6$	$W - r_3 - r_7$
$S_1 = r_1 + r_4$	0	$b_4 - x_3 - x_4$	$b_6 - x_4 - x_5 - x_6$	$b_7 - b_1$	$\min\{b_1 - x_6, b_5 - b_4, b_6 - x_4 - x_6\}$	$\min\{b_6 - x_4, b_1, W - S_6\}$	$b_1 - x_5 - x_6$
$S_2 = r_2 + r_5$	$b_2 - x_6 - x_7$	0	$b_5 - x_4 - x_5$	$\min\{b_7 - x_6, b_5, W - S_4\}$	$\min\{b_1 - b_2, b_5 - x_4, b_7 - x_4 - x_6\}$	$b_6 - b_5$	$b_7 - x_4 - x_5 - x_6$
$S_3 = r_3 + r_5$	$\min\{b_3 - x_7, b_4 - x_4, b_1 - x_5 - x_6 - x_7\}$	$b_3 - x_1 - x_7$	0	$\min\{b_5, b_6, W - S_4\}$	$\min\{b_1 - x_7, b_5 - x_4, b_6 - x_4\}$	$b_6 - x_4 - x_5$	$b_7 - b_6$
$S_4 = r_4 + r_7$	$\min\{b_4 - x_3, b_1 - b_7, b_2 - x_6 - x_7\}$	$b_4 - x_1 - x_3$	$b_6 - x_5 - x_6$	0	$\min\{b_5, b_6, b_7\}$	$\min\{b_6 - x_5, b_7 - x_5, W - S_6\}$	$b_7 - x_5 - x_6$
$S_5 = r_1 + r_5$	$b_1 - x_6 - x_7$	$b_5 - x_3 - x_4$	$b_6 - x_4 - x_6$	$\min\{b_5, b_6, W - S_4\}$	0	$\min\{b_1, b_6 - x_4, b_7 - x_4\}$	$b_7 - x_4 - x_6$
$S_6 = r_2 + r_6$	$b_2 - x_2 - x_7$	$\min\{b_2 - x_7, b_5 - b_6, b_4 - x_3 - x_4\}$	$b_6 - x_4 - x_5$	$\min\{b_6, b_7, W - S_4\}$	$\min\{b_6 - x_4, b_7 - x_4, W - S_5\}$	0	$b_7 - x_4 - x_5$
$S_7 = r_3 + r_7$	$b_3 - x_2 - x_3$	$\min\{b_2 - x_6, b_3 - x_3, b_5 - x_3 - x_4 - x_5\}$	$b_6 - b_7$	$b_7 - x_5 - x_6$	$\min\{b_5, b_1 - x_6, b_7 - x_6\}$	$\min\{b_1, b_7, W - S_6\}$	0

6. Steps of the Algorithm

1. Determine the minimum workforce size W using equation (9).
2. If $\max\{\sum r/3,\ S_{\max}\} = \sum r/3$ then,
   (a) If $\sum r/3$ is not integer, increment daily demands by $(3W - \sum r)$ to make $\sum r$ a multiple of 3, choosing the lowest demands but avoiding weekend demands $(r_6,\ r_7)$.
   (b) Calculate $b_1, ..., b_7$ using equation (5), then compute $\sum r/3$ row in Table 1.
3. If $\max\{\sum r/3,\ S_{\max}\} = S_i$,
   Calculate $b_1, ..., b_7$ using equation (5), then compute S_i row in Table 1.
   In the case of ties, apply any one of steps 2 or 3 arbitrarily.

7. A Solved Example

The following example is used to illustrate the simple calculations required for implementing the algorithm. However, using any spreadsheet software, Table 1 itself can be expressed as a spreadsheet which can be used to conveniently implement the algorithm. Given the following daily labour demands for a work week:

$$r_1, r_2, ..., r_7 = 8, 3, 6, 2, 5, 4, 8$$

Calculate

$$\sum r = 36$$
$$S_1, S_2, ..., S_7 = 10, 8, 10, 10, 13, 7, 14$$

Thus,

$$\sum r/3 = 36/3 = 12$$
$$S_{\max} = S_7 \quad = 14$$

Using (9), the workforce size is

$$W = S_7 = 14$$

Using equation (5), $b_i = 14 - r_i$, we obtain

$$b_1, b_2, ..., b_7 = 6, 11, 8, 12, 9, 10, 6$$

Using S_7 row in Table 1, we obtain the following days-off assignments:

$$
\begin{aligned}
x_7 &= 0 \\
x_6 &= \min\{b_1,\ b_7,\ W - S_6\} = \min\{6,\ 6,\ 14-7\} = 6 \\
x_5 &= \min\{b_1 - x_6,\ b_5,\ b_7 - x_6\} = \min\{6-6,\ 9,\ 6-6\} = 0 \\
x_4 &= b_7 - x_5 - x_6 = 6 - 0 - 6 = 0 \\
x_3 &= b_6 - b_7 = 10 - 6 = 4 \\
x_2 &= \min\{b_2 - x_6,\ b_3 - x_3,\ b_5 - x_3 - x_4 - x_5\} \\
&= \min\{11-6,\ 8-4,\ 9-4-0-0\} = 4 \\
x_1 &= b_3 - x_2 - x_3 = 8 - 4 - 4 = 0
\end{aligned}
$$

8. Conclusions

An efficient optimization algorithm for the cyclic (3, 7) labour days-off scheduling problem has been presented. The algorithm is based on the solution of the dual linear programming model but does not involve linear or integer programming. Since the costs of different days-off work patterns are not assumed to be equal, the algorithm can be used to minimize either the total number or the total cost of workers assigned. The simplicity of the algorithm makes it easy to implement manually, removing the need for specialized training and software. A similar approach could be used to solve other cyclic scheduling problems in which the dual solution is easily obtained.

Acknowledgements

The author would like to thank King Fahd University of Petroleum and Minerals for supporting this research project.

References

Alfares, H. K. (1998), An efficient two-phase algorithm for manpower days-off scheduling, *Computers and Operations Research* 25, 913-923.

Alfares, H. K. and J. E. Bailey (1997), Integrated project task and manpower scheduling, *IIE Transactions* 29, 711-718.

Baker, K. R. (1974), Scheduling a full-time workforce to meet cyclic staffing requirements, *Management Science* 20, 1561-1568.

Baker, K. R. (1976), Workforce allocation in cyclical scheduling problems: A survey, *Operational Research Quarterly* 27, 155-167.

Bechtold, S. E. (1988), Implicit optimal and heuristic labor staffing in a multiobjective, multilocation environment, *Decision Science* 19, 353-372.

Bechtold, S. E. and M. J. Showalter (1985), Simple manpower scheduling methods for managers, *Production and Inventory Management* 26, 116-133.

Browne, J. E. and R. K. Tiberwala (1975), Manpower scheduling, *AIIE Transactions* 7, 22-23.

Browne, J. and R. Nanda (1987), Scheduling efficiency of the four-day workweek at transportation facilities, *Proceedings of the Institute of Transportation Engineers 57th Annual Meeting*, New York, pp. 58-62.

Burns, R. N., N. Narasimhan and L. D. Smith (1998), A set-processing algorithm for scheduling staff on 4-day or 3-day workweeks, *Naval Research Logistics* 45, 839-853.

Emmons, H. (1985), Workforce scheduling with cyclic requirements and constraints on days-off, weekends off and work stretch, *IIE Transactions* 17, 8-16.

Emmons, H. and R. N. Burns (1991), Off-day scheduling with hierarchical worker categories, *Operations Research* 39, 484-495.

Hung, R. (1993), A three-day workweek multiple-shift scheduling model, *Journal of the Operational Research Society* 44, 141-146.

Hung, R. (1994), A multiple-shift workforce scheduling model under the 4-day workweek with weekday and weekend labour demands, *Journal of the Operational Research Society* 45, 1088-1092.

Hung, R. and H. Emmons (1993), Multiple-shift workforce scheduling under the 3-4 compressed workweek with a hierarchical workforce, *IIE Transactions* 25, 82-89.

Koop, G. J. (1986), Cyclic scheduling of offweekends, *Operations Research Letters* 4, 259-263.

Morris, G. M. and M. J. Showalter (1983), Simple approaches to shift, days-off and tour scheduling problems, *Management Science* 29, 942-950.

Nanda, R. and J. Browne (1992), *Introduction to Employee Scheduling*, Van Nostrand Reinhold, New York, NY.

Narasimhan, N. (1996), An algorithm for single shift scheduling of hierarchical workforce, *European Journal of Operational Research* 96, 113-121.

Steward, G. V. and J. M. Larsen (1971), A four-day three-day application to a continuous production operations, *Management of Personnel Quarterly* 10, 13-20.

Tiberwala, R., D. Philippe and J. Browne (1972), Optimal scheduling of two consecutive idle periods, *Management Science* 19, 71-75.

Tien, J. M. and A. Kamiyama (1982), On manpower scheduling algorithms, *SIAM Review* 24, 275-287.

Vance, P. H., C. Barnhart, E. L. Johnson and G. L. Nemhauser (1997), Airline crew scheduling, *Operations Research* 45, 188-200.

Hesham ALFARES is Associate Professor at the Systems Engineering Department of King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia. He obtained his B.S. in Electrical and Computer Engineering from the University of California at Santa Barbara, and M.S. and Ph.D. in Industrial Engineering from the University of Pittsburgh and Arizona State University, respectively. His current research interests include personnel and project scheduling, modelling and simulation of maintenance systems, and modelling and optimization of petrochemical systems. He has published in several journals, including *IIE Transactions, Computers and Operations Research, Computers and Industrial Engineering, Journal of the American Society for Information Science, Production Planning and Control*, and *Journal of Quality in Maintenance Engineering.* He has refereed several papers and funded research projects for a number of international journals and institutions.