European Journal of Operational Research 125 (2000) 381–397

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/dsw

Theory and Methodology

A comparative evaluation of modeling approaches to the labor shift scheduling problem

Turgut Aykin *

Engineering Research Center, Bell Laboratories, Princeton, NJ 08542, USA

Received 1 November 1996; accepted 1 August 1998

Abstract

The labor shift scheduling problem has traditionally been formulated using the set covering approach proposed by Dantzig, G.B. [1954. Operations Research 2 (3), 339–341]. The size of the resulting integer model with this approach, however, has been found to be very large to solve optimally in most practical applications. Recently, Bechtold, S.E., Jacobs, L.W. [1990. Management Science 36 (11), 1339–1351] proposed a new integer programming formulation requiring significantly smaller number of variables and nonzeros in the A-matrix than the equivalent set covering formulation. However, due to its assumptions, this approach has remained limited to the special cases of the shift scheduling problem. In Aykin, T. [1996. Management Science 42, 591–603], we presented another implicit modeling approach that is applicable to the general problem without any of the limitations of Bechtold and Jacobs (loc. cit.). This approach also requires substantially smaller number of variables and nonzeros in the A-matrix than the equivalent set covering formulation. In this paper, we relax the assumptions of Bechtold and Jacobs (loc. cit.) and present a generalized version of it. The extended formulation of Bechtold and Jacobs, although requiring smaller number of variables, has more constraints, more nonzeros in the A-matrix, and significantly higher density than the formulation of Aykin (loc. cit.). We compare these modeling approaches through solving (optimally) 220 problems involving as many as 32 928 shift variations. Our computational results show that the time needed to locate an optimal shift schedule and the percentage of the problems solved optimally with Aykin's formulation are substantially better than those with the formulation of Bechtold and Jacobs.

Keywords: Labor shift scheduling; Manpower management; Service operations management; Implicit modeling; Integer programming

1. Introduction

One of the critical tasks the manager of a service system faces daily is the effective scheduling of its manpower. In most service systems, demand for services changes during the course of an operating day. In order to develop a shift schedule, demand

* Correspondence address: 23 John st., Red Bank, NJ 07701, USA.

E-mail address: taykin@lucent.com (T. Aykin).

0377-2217/00/$ - see front matter
PII: S 0 3 7 7 - 2 2 1 7 (9 9) 0 0 4 1 3 - 0

needs to be forecasted and converted (usually with the help of queuing models) into labor requirements needed to maintain a desired service level during the course of an operating day. Employees are then assigned to various shifts specified by their start times, lengths, number and timing of relief and lunch breaks to meet these requirements. While understaffing will lower total labor cost, it will result in deterioration of the service quality and longer waiting times, and consequently, higher total cost. Overstaffing, on the other hand, will improve the service quality but will result in underutilization and excessive labor costs. Therefore, it is important for a service organization to schedule its manpower in an efficient manner to minimize labor costs while providing the desired service level.

The shift scheduling problem involves determining the number of employees to be assigned to various shifts and the timing of their breaks within the limits allowed by legal, union, and company requirements. To improve manpower utilization and lower labor costs, flexibility in shift types, start times, length, number and duration of breaks is provided. Flexibility in scheduling relief and lunch breaks for a given shift is provided with break windows specifying the time intervals within which employees assigned to that shift must start and complete their breaks. Since its introduction by Edie (1954), this problem has attracted considerable attention in the literature. Both heuristic and integer programming based approaches have been proposed. The integer programming based approaches have traditionally been based on the set covering formulation proposed by Dantzig (1954). For nearly four decades, this was the only modeling approach available for the general problem. Although the set covering formulation has been studied extensively, it is not a practical approach to solve this problem optimally. The set covering approach treats every possible shift type, shift start time, and break placement combination as a separate shift and requires a separate integer variable for it. As a result, the size of the integer model increases very rapidly with the number of shift types, shift start times, breaks, and break window sizes, making it impractical to analyze with the available integer programming tools.

Recently, two new modeling approaches have been developed by Bechtold and Jacobs (1990), and Aykin (1996). Both studies model break placements implicitly to reduce the size of the integer model. The objective of this study is to comparatively evaluate these two modeling approaches to the general shift scheduling problem in a variety of problem situations. For this purpose, we first relax a number of assumptions made in Bechtold and Jacobs (1990) and present a generalized version of their formulation that was used in the computational experiments reported. The modeling approaches are compared using a set of 220 test problems involving as many as 32 928 shift variations. The problem set considered in this study include, to the best of our knowledge, the largest shift scheduling problems solved optimally in the literature. Our results show that the formulation given in Aykin (1996) requires more variables than the formulation of Bechtold and Jacobs but substantially less than the set covering model. The number of constraints, number of nonzeros and the density of the A-matrix with the approach given in Aykin (1996) are, however, substantially smaller than those for the formulation given in Bechtold and Jacobs, and the set covering model. More importantly, our results show that the time needed to locate an optimal shift schedule and the percentage of the problems solved optimally with Aykin's formulation are substantially better than those with the formulation of Bechtold and Jacobs; while no optimal solution could be found in 32 (14.5%) problems with the formulation of Bechtold and Jacobs, only 2 problems remained unsolved with the formulation of Aykin. The average IP solution time for the problems solved by both modeling approaches was 331.36 seconds with the formulation of Bechtold and Jacobs, and 113.04 seconds (about 66% less) with the formulation of Aykin. The fact that these results were obtained with a problem set including the largest shift scheduling problems solved optimally in the literature using an off-the-shelf integer programming software (LINDO) is very encouraging in terms of the practical applications of implicit formulations in service systems.

The remainder of the paper is organized as follows. In Section 2, we introduce the set covering formulation, extend the formulation given in Bechtold and Jacobs into the scheduling environments involving 24-hour continuous operations, multiple relief/lunch breaks, and multiple disjoint break windows, and present the formulation given in Aykin (1996). Computational results obtained with 220 test problems are presented in Section 3. Finally, in Section 4, we summarize our results.

2. Modeling approaches to the shift scheduling problem

This section introduces the notation used and presents the three modelings available for the general shift scheduling problem.

2.1. Set covering model of Dantzig (1954)

The following set covering formulation was introduced by Dantzig (1954):

$$\text{Minimize} \quad \sum_{k \in K} c_k X_k \tag{1}$$

subject to

$$\sum_{k \in K} a_{kt} X_k \geqslant b_t \quad \text{for all } t \in T,$$

$$X_k \geqslant 0 \text{ and integer}, \quad k \in K, \tag{2}$$

where X_k is an integer variable defined as the number of employees assigned to shift k, T is the set of planning periods covered by the shift schedule, K is the set of all shifts, b_t is the number of employees needed in period t to achieve the desired service level, c_k is the cost of assigning an employee to shift k, and a_{kt} is equal to one if period t is a work period for shift k and zero otherwise.

With this formulation, all shift type, shift start time, and break placement combinations are enumerated and included, as separate shifts, in set K.

2.2. Implicit modeling approach of Bechtold and Jacobs (1990)

Recognizing the difficulties caused by the enumeration of all shift variations in the set covering model and making use of the fact that not all information concerning the break placements included in that model is necessary to develop an optimal shift schedule, Bechtold and Jacobs (1990) developed an implicit formulation of the problem that is substantially smaller than the equivalent set covering model. Although the implicit modeling idea itself is not new in the manpower scheduling literature (see, for example, Moondra, 1976; Gaballa and Pearce, 1979), their formulation uses a novel approach for modeling break placements implicitly. Bechtold and Jacobs (1990) compared their formulation with the set covering model experimentally and reported superior results with their approach.

Unfortunately, due to its assumptions, this formulation is limited to certain instances of the problem and cannot be applied to the general case in its present form. The following assumptions are listed in Bechtold and Jacobs (1990, p. 1341): (1) the system operates less than 24 hours daily, (2) planning periods are equal in length, (3) each shift is given a single break, (4) the break duration is identical for all shifts, (5) the break duration is one or more periods, (6) each shift has a single break window associated with its workspan, (7) breaks should start and end during the shifts, (8) no extraordinary break window overlap exists (this condition exists when the break window of one shift is a strict subset of the break window of another, and has different starting and ending periods than the larger window), and (9) no understaffing is allowed. Besides these, Bechtold and Jacobs (1990) also assume that every break window consists of contiguous planning periods (i.e. no split windows). They noted in their study that assumptions (2), (5), and (7) are consistent with the scheduling practice and are common in the literature, and (3) and (9) can be relaxed by modifying their formulation. With assumptions (4) and (6), however, their formulation would not be appropriate for a shift scheduling problem involving multiple relief and lunch breaks with unequal duration (e.g. two 15-minute rest breaks and one 30-minute lunch break arrangement found in most service systems) and multiple disjoint break windows associated with them. Furthermore, with assumption (1), their approach is limited to the problems involving shifts starting and ending during the same 24-hour period, and with assumption (8) to the cases not involving any extraordinary break window overlaps. Recognizing these limitations, after discussing the effects of extraordinary break window overlaps, they state in the summary and conclusions section of their study that

> "*Numerous research opportunities exist for the extension of the implicit modeling approach for use in other operating environments. For example, shift scheduling for telephone operators is typically done across 24 hour planning horizons. Moreover, shift scheduling environments exist in practice for which the organization desires to include optimal placement of rest breaks as well as meal breaks in shift assignments* (Hendersen and Berry, 1976)" (Bechtold and Jacobs, 1990, p. 1350).

The study cited by Bechtold and Jacobs (1990) in the above paragraph (Henderson and Berry, 1976), considers a shift scheduling problem involving full-time shifts with two relief breaks and one lunch break with unequal durations and multiple disjoint break windows. Since, this type of shifts are found in most practical applications, we assumed the same scheduling environment to compare the two modeling approaches. Moreover, since the case involving 24-hour continuous operation (cyclical environment) has received considerable attention in practice and in the literature (see, for example, Baker, 1976; Bartoldi, 1981; Bartoldi et al., 1980; Bedworth and Bailey, 1987; Vohra, 1988; Nanda and Browne, 1992, among others), we considered both cyclical and acyclical (less than 24-hour operation) scheduling environments. Solving shift scheduling problems in the scheduling environments considered with the formulation of Bechtold and Jacobs (1990), however, requires the relaxation of assumptions (1), (3), (4), and (6). Below, we present the modifications needed to relax these assumptions. The modifications needed to relax the assumptions concerning the extraordinary break window overlaps (assumption 8) and contiguous break windows, and extend this approach to the general tour scheduling problem (solving daily shift and days-off scheduling together) can be found in Aykin (1995a). Let T be the set of planning periods covered by the shift schedule, and K be the set of shifts including all feasible combinations of different shift types and start times. Define X_k as an integer variable representing the number of employees assigned to shift k, $k \in K$. To model the placement of breaks of a particular type (e.g. first relief break taken before the lunch break) in an environment involving 24-hour continuous operations (acyclical case is discussed later in this section), one needs to distinguish shifts with a break window starting and ending within the same day (for convenience, assume that a work day starts at 7:00 a.m. and ends 7:00 a.m. the following day) from those with a break window starting in one day but continuing into the following day. Consider the first relief break windows for the shifts in K. Let KU1 be the set of shifts with a first relief break window starting and ending within the same day, and KU2 be the set of shifts with a first relief break window starting in one day and continuing into the following day (note that $K \equiv \mathrm{KU1} \cup \mathrm{KU2}$). In the extended formulation, first relief breaks for the shifts in these two sets are scheduled separately by defining separate sets of first relief break variables and introducing two sets of break placement constraints. Let $U1_t$ be an integer variable representing the number of employees assigned to shifts in KU1 and taking their first relief breaks in period t, and $U2_t$ be an integer variable representing the number of employees assigned to a shift in KU2 and taking their first relief break in period t. In this case, first breaks for the shifts in KU1 can be scheduled within their break windows by using the forward pass and backward pass constraints given in Bechtold and Jacobs (1990). In the cyclical scheduling environment considered, a similar set of constraints is needed for the shifts in KU2 in order to schedule correct number of first relief breaks within their break windows. The number of constraints added for the shifts in KU2 is usually small and is determined by the length of the break window. When an optimal solution to the extended formulation is obtained, break times can be assigned to shifts by applying the simple break scheduling procedure given in Bechtold and Jacobs (1990) to each set separately. Thus, the following two sets of constraints are needed for scheduling the first relief breaks in the extended formulation.

$$\sum_{t\in \mathrm{SU1}_j} U1_t - \sum_{k\in \mathrm{TU11}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{NU1} - \{\mathrm{qu1}\}, \tag{3}$$

$$\sum_{t\in \mathrm{ZU1}_j} U1_t - \sum_{k\in \mathrm{TU21}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{MU1} - \{\mathrm{pu1}\}, \tag{4}$$

$$\sum_{k\in \mathrm{KU1}} X_k = \sum_{t\in \mathrm{TU1}} U1_t, \tag{5}$$

$$\sum_{t\in \mathrm{SU2}_j} U2_t - \sum_{k\in \mathrm{TU12}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{NU2} - \{\mathrm{qu2}\}, \tag{6}$$

$$\sum_{t\in \mathrm{ZU2}_j} U2_t - \sum_{k\in \mathrm{TU22}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{MU2} - \{\mathrm{pu2}\}, \tag{7}$$

$$\sum_{k\in \mathrm{KU2}} X_k = \sum_{t\in \mathrm{TU2}} U2_t, \tag{8}$$

where for the first relief breaks for the shifts in KU1, we define

- pu1 — earliest first relief break start time for the shifts in KU1
- qu1 — latest first relief break start time for the shifts in KU1
- TU1 — {pu1, ... ,qu1}, the set of all possible first relief break start times for the shifts in KU1
- $\mathrm{SU1}_j$ — {pu1,... , j}, the set of all first relief break start times between periods pu1 and j
- $\mathrm{ZU1}_j$ — {j, ..., qu1}, the set of all first relief break start times between periods j and qu1
- NU1 — set of latest first relief break start times (in ascending order) for the shifts in KU1
- MU1 — set of earliest first relief break start times (in ascending order) for the shifts in KU1,
- $\mathrm{TU11}_j$ — the set of shifts in KU1 with a first relief break window lying completely between periods pu1 and j, and
- $\mathrm{TU21}_j$ — the set of shifts in KU1 with a first relief break window lying completely between periods j and qu1

In a similar fashion, for the shifts in KU2, we define the parameters and the sets pu2, qu2, TU2, $\mathrm{SU2}_j$, $\mathrm{ZU2}_j$, NU2, MU2, $\mathrm{TU12}_j$, and $\mathrm{TU22}_j$.

To model lunch break placements correctly, define, similar to KU1 and KU2 defined for the first relief breaks above, the sets KW1 and KW2 ($K \equiv \mathrm{KW1} \cup \mathrm{KW2}$). Also define the integer variables $W1_t$ and $W2_t$ as the number of employees assigned, respectively, to shifts in sets KW1 and KW2 and starting their lunch breaks in period t. The following two sets of constraints are needed to schedule lunch breaks within the correct break windows for the shifts in KW1 and KW2.

$$\sum_{t\in \mathrm{SW1}_j} W1_t - \sum_{k\in \mathrm{TW11}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{NW1} - \{\mathrm{qw1}\}, \tag{9}$$

$$\sum_{t\in \mathrm{ZW1}_j} W1_t - \sum_{k\in \mathrm{TW21}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{MW1} - \{\mathrm{pw1}\}, \tag{10}$$

$$\sum_{k\in \mathrm{KW1}} X_k = \sum_{t\in \mathrm{TW1}} W1_t, \tag{11}$$

$$\sum_{t\in \mathrm{SW2}_j} W2_t - \sum_{k\in \mathrm{TW12}_j} X_k \geqslant 0 \quad \text{for all } j \in \mathrm{NW2} - \{\mathrm{qw2}\}, \tag{12}$$

$$\sum_{t\in \mathrm{ZW2}_j} W2_t - \sum_{k\in \mathrm{TW22}_j} X_k \geqslant 0$$
$$\text{for all } j \in \mathrm{MW2} - \{\mathrm{pw2}\}, \quad (13)$$

$$\sum_{k\in \mathrm{KW2}} X_k = \sum_{t\in \mathrm{TW2}} W2_t, \quad (14)$$

where for the lunch breaks for the shifts in KW1, we define

pw1	earliest lunch break start time for the shifts in KW1
qw1	latest lunch break start time for the shifts in KW1
TW1	{pw1,...,qw1}, the set of all possible lunch break start times for the shifts in KW1
$\mathrm{SW1}_j$	{pw1, ..., j}, the set of all lunch break start times between periods pw1 and j
$\mathrm{ZW1}_j$	{j,..., qw1}, the set of all lunch break start times between periods j and qw1
NW1	set of latest lunch break start times (in ascending order) for the shifts in KW1
MW1	the set of earliest lunch break start times (in ascending order) for the shifts in KW1
$\mathrm{TW11}_j$	set of shifts in KW1 with a lunch break window lying completely between periods pw1 and j, and
$\mathrm{TW21}_j$	the set of shifts in KW1 with a lunch break window lying completely between periods j and qw1

Similarly, define pw2, qw2, TW2, $\mathrm{SW2}_j$, $\mathrm{ZW2}_j$, NW2, MW2, $\mathrm{TW12}_j$, and $\mathrm{TW22}_j$ for the lunch breaks for the shifts in KW2.

Above, constraints (9)–(11) and (12)–(14) ensure that two sets of lunch break times are scheduled for the shifts in KW1 and KW2 separately.

To schedule the second relief breaks, define the sets KV1 and KV2 ($K \equiv \mathrm{KV1} \cup \mathrm{KV2}$). Also define the integer variables $V1_t$ and $V2_t$ as the number of employees assigned, respectively, to shifts in KV1 and KV2 and taking their second relief breaks in period t. The following two sets of constraints are needed to schedule second relief breaks within the correct break windows for the shifts in KV1 and KV2.

$$\sum_{t\in \mathrm{SV1}_j} V1_t - \sum_{k\in \mathrm{TV11}_j} X_k \geqslant 0$$
$$\text{for all } j \in \mathrm{NV1} - \{\mathrm{qv1}\}, \quad (15)$$

$$\sum_{t\in \mathrm{ZV1}_j} V1_t - \sum_{k\in \mathrm{TV21}_j} X_k \geqslant 0$$
$$\text{for all } j \in \mathrm{MV1} - \{\mathrm{pv1}\}, \quad (16)$$

$$\sum_{k\in \mathrm{KV1}} X_k = \sum_{t\in \mathrm{TV1}} V1_t, \quad (17)$$

$$\sum_{t\in \mathrm{SV2}_j} V2_t - \sum_{k\in \mathrm{TV12}_j} X_k \geqslant 0$$
$$\text{for all } j \in \mathrm{NV2} - \{\mathrm{qv2}\}, \quad (18)$$

$$\sum_{t\in \mathrm{ZV2}_j} V2_t - \sum_{k\in \mathrm{TV22}_j} X_k \geqslant 0$$
$$\text{for all } j \in \mathrm{MV2} - \{\mathrm{pv2}\}, \quad (19)$$

$$\sum_{k\in \mathrm{KV2}} X_k = \sum_{t\in \mathrm{TV2}} V2_t, \quad (20)$$

where for the second relief breaks for the shifts in KV1, we define

pv1	earliest second relief break start time for the shifts in KV1
qv1	latest second relief break start time for the shifts in KV1
TV1	{pv1,...,qv1}, the set of all possible second relief break start times for the shifts in KV1
$\mathrm{SV1}_j$	{pv1,...,j}, the set of all second relief break start times between periods pv1 and j
$\mathrm{ZV1}_j$	{j,...,qv1}, the set of all second relief break start times between periods j and qv1
NV1	set of latest second relief break start times (in ascending order) for the shifts in KV1

MV1 set of earliest second relief break start times (in ascending order) for the shifts in KV1

$\mathrm{TV11}_j$ set of shifts in KV1 with a second relief break window lying completely between periods pv1 and j, and

$\mathrm{TV21}_j$ set of shifts in KV1 with a second relief break window lying completely between periods j and qv1

Similarly, define pv2, qv2, TV2, $\mathrm{SV2}_j$, $\mathrm{ZV2}_j$, NV2, MV2, $\mathrm{TV12}_j$, and $\mathrm{TV22}_j$ for the second relief breaks for the shifts in KV2.

Constraints (15)–(17) and (18)–(20) above ensure that two sets of second relief are scheduled within the break windows specified for the shifts in KV1 and KV2 separately.

We now present the extended formulation of Bechtold and Jacobs for a cyclical case involving multiple rest and lunch breaks, and disjoint break windows. Let c_k be the cost of assigning an employee to shift k, $k \in K$, a_{kt} be equal to one if period t is within the workspan (a work or a break period) of shift k, and zero otherwise, and b_t be the number of employees needed in period t. Then, the extended formulation can be stated as follows:

$$\text{Minimize} \quad \sum_{k \in K} c_k X_k \tag{21}$$

subject to

$$\sum_{k \in K} a_{kt} X_k - \sum_{j \in \mathrm{TU1}} \upsilon 1_{jt} U1_j - \sum_{j \in \mathrm{TW1}} \omega 1_{jt} W1_j - \sum_{j \in \mathrm{TV1}} v1_{jt} V1_j - \sum_{j \in \mathrm{TU2}} \upsilon 2_{jt} U2_j - \sum_{j \in \mathrm{TW2}} \omega 2_{jt} W2_j - \sum_{j \in \mathrm{TV2}} v2_{jt} V2_j \geqslant b_t$$

for all $t \in T$,

$$\text{constraints (3)–(20),} \tag{22}$$

$X_k, U1_t, U2_t, W1_t, W2_t, V1_t, V2_t \geqslant 0$ and integer,

where

$\upsilon 1_{jt}$ 1 if period t is a first relief break period for the employees starting their breaks in period j, $j \in \mathrm{TU1}$, and zero otherwise,

$\omega 1_{jt}$ 1 if period t is a lunch break period for the employees starting their breaks in period j, $j \in \mathrm{TW1}$, and zero otherwise,

$v1_{jt}$ 1 if period t is a second relief break period for the employees starting their breaks in period j, $j \in \mathrm{TV1}$, and zero otherwise,

and $\upsilon 2_{jt}, \omega 2_{jt}$, and $v2_{jt}$ are defined similarly for the shifts in TU2, TW2, and TV2, respectively.

In the above formulation, constraint (22) ensures that the number of employees assigned to various shifts and available in period t minus the number of employees taking their first relief, lunch, and second relief breaks is sufficient to meet demand and provide the desired level of service in that period.

The equality constraints (5), (8), (11), (14), (17), and (20) may be substituted in the associated break constraints in a manner similar to Bechtold and Jacobs (1990) to reduce the number of nonzeros whenever appropriate. Our preliminary tests with this reduced form, however, did not show any computational advantage over the basic model.

Note that if the scheduling problem is an acyclical one involving multiple relief and lunch breaks and break windows, then $\mathrm{KU1} \equiv K$, $\mathrm{KW1} \equiv K$, and $\mathrm{KV1} \equiv K$, with $\mathrm{KU2} \equiv \mathrm{KW2} \equiv \mathrm{KV2} = \emptyset$. Hence, variables $U2_t$, $W2_t$, and $V2_t$, and constraints (6)–(8), (12)–(14), and (18)–(20) are not needed in the above formulation.

Recently, Thompson (1995) applied the formulation of Bechtold and Jacobs together with the implicit representation of shifts described in Moondra (1976) in a shift scheduling environment involving a high degree of shift length and start time flexibility. Given a set of shift start times and a set of shift lengths associated with every shift start time (e.g. shifts with lengths 6, 6.5, …, 9.5, and 10 hours are assumed to be available to start every 30 minutes throughout the day), Moondra's approach was used to model shifts with varying lengths and start times, and the modeling approach of Bechtold and Jacobs to model break placements implicitly to reduce the size of the formulation. The extension presented in Thompson (1995), however, has a number of limitations. First, it assumes that a high degree of flexibility in terms of both the shift start times and shift lengths is available. In practice, usually the company, union, and legal requirements are followed in determining shift start times together with shift lengths (see, for example, Segal, 1974; Henderson and Berry, 1976; Buffa, 1980; Bedworth and Bailey, 1987; Love and Hoey, 1990; Nanda and Browne, 1992; Spencer et al., 1992). Hence, except for such systems as fast food restaurants, the degree of combined shift length and shift start time flexibility assumed is not common in practice. If a high degree of shift length flexibility is not available with *most* of the shift start times, on the other hand, the formulation presented in Thompson (1995) would offer little advantage over the formulation of Bechtold and Jacobs. For instance, in a problem involving only full-time employees (e.g. Henderson and Berry, 1976), or in a problem with alternative shift lengths that are available only at certain shift start times (e.g. Segal, 1974), the formulation given in Thompson would offer little (if any) advantage over the formulation of Bechtold and Jacobs. Another limitation of the formulation given in Thompson (1995) is that the break windows for a shift type are defined with reference to the shift start and finish times by specifying the minimum and maximum pre- and post-break work periods. As a result, even when two shifts share similar attributes (e.g. same break duration, even the same start time), the length of the break window decreases with the shift length. In the illustrative example given in Thompson (1995, Section 2.2.), for example, while the break window is 3 hours long for a 7-hour shift, the length of the break window decreases to 2 hours for a 6-hour shift, and to 1 hour (that is, a "fixed" meal break time for a 1-hour meal break) for a 5-hour shift even when they all start at the same time. In practice, start time and length of a shift determine the number and type of breaks (e.g. rest, lunch, idle period for a split-shift) employees receive. Together with them, break windows are specified with their lengths and position with respect to some other shift features (e.g. start time, "ideal" break time, etc). As reported extensively in the literature (see, for example, Segal, 1974; Henderson and Berry, 1976; Bedworth and Bailey, 1987; Bechtold and Jacobs, 1990; Nanda and Browne, 1992; Jarrah et al., 1994), the length and positioning of a break window usually do not change with the shift start time or length. This way the degree of break scheduling flexibility provided by break windows remains the same for all shift types to better match demand, improve manpower utilization, and provide timely relief to employees. This type of scheduling environment cannot be modeled efficiently with the formulation described in Thompson (1995). Also, the minimum shift length for a particular shift type, and the minimum pre- and post-break work periods should be examined carefully to avoid infeasibility.

Another limitation of the formulation presented by Thompson is its inability to model a realistic cost structure accurately. The formulation presented assumes that the cost of assigning an employee to a shift is a linear function of only its duration and not whether it is a full-time or a part-time shift, its start time or the time periods it covers (e.g. day shift vs. night shift). As also stated in Thompson (1995), to model a realistic cost structure accurately, a formulation with explicit shift representation is needed. And, finally, like the formulation of Bechtold and Jacobs, the formulation given in Thompson is limited to the cases covering less than 24 hours (i.e. acyclical problem). In summary, the formulation presented in Thompson (1995) is applicable to special cases satisfying these conditions and cannot be applied to the general shift scheduling problem in its present form. Therefore, this extension of Bechtold and Jacobs (1990) is not considered in this study.

2.3. Implicit formulation of Aykin (1996)

In a recent study (Aykin, 1996), we presented another implicit formulation for the general shift scheduling problem. Like Bechtold and Jacobs (1990), Aykin (1996) is modeling the break placements implicitly. The approaches used in these two studies, however, are different in the way the breaks times are matched with the shifts. While the formulation given by Bechtold and Jacobs (1990) uses break variables associated with each planning period and matches breaks with the shifts implicitly, the formulation given in Aykin (1996) defines separate break variables for each shift. This approach extends the implicit break representation idea introduced in Gaballa and Pearce (1979) to the general case involving multiple rest and lunch breaks and multiple disjoint break windows. In the special case considered by Gaballa and Pearce, this approach required more variables and constraints than the equivalent set covering formulation. Therefore, their approach has not been generalized or considered any further in the literature. Aykin (1996) showed that although apparently not beneficial in the special case considered by Gaballa and Pearce (1979), this approach is surprisingly very successful in reducing the number of variables needed in the general shift scheduling problem. The number of nonzeros and the density of the A-matrix are also much lower than the set covering formulation. Besides these, the new formulation makes the same assumptions with the set covering formulation and is applicable to all cases that the set covering approach is applicable.

To formulate the problem using this approach, we define K as the set of all shifts including all feasible combinations of shift start times and lengths, and X_k as the number of employees assigned to shift k. Let U_{kt}, W_{kt}, and V_{kt} be the break variables representing the number of employees assigned to shift k and taking their first relief, lunch, and second relief breaks in period t, respectively. Let $B1_k$, BL_k, and $B2_k$ be the sets of planning periods forming the first relief, lunch, and second relief break windows of shift k, respectively. In this formulation, the first relief break variable U_{kt} is defined only for $t \in B1_k$, lunch break variable W_{kt} for $t \in BL_k$, and the second relief break variable V_{kt} for $t \in B2_k$. Further, define $T1_t$, TL_t, and $T2_t$ as the sets of shifts for which period t is a break start time within the break windows for the first relief, lunch, and second relief break, respectively. Also, let a_{kt} be one if period t is in the shift span (a work or a break period during the shift) of shift k and zero otherwise. The general shift scheduling problem can then be formulated as follows:

$$\text{Minimize} \quad \sum_{k \in K} c_k X_k \tag{23}$$

subject to

$$\sum_{k \in K} a_{kt} X_k - \sum_{k \in T1_t} U_{kt} - \sum_{k \in TL(t-1)} W_{k(t-1)} - \sum_{k \in TL_t} W_{kt} - \sum_{k \in T2_t} V_{kt} \geqslant b_t \quad \text{for all } t \in T, \tag{24}$$

$$X_t - \sum_{t \in B1_k} U_{kt} = 0 \quad \text{for all } k \in K, \tag{25}$$

$$X_t - \sum_{t \in BL_k} W_{kt} = 0 \quad \text{for all } k \in K, \tag{26}$$

$$X_t - \sum_{t \in B2_k} V_{kt} = 0 \quad \text{for all } k \in K, \tag{27}$$

$X_t, U_{kt}, W_{kt}, V_{kt} \geqslant 0$ and integer.

In the above formulation, break windows are specified by $B1_k$, BL_k, and $B2_k$. In the extended formulation of Bechtold and Jacobs presented before, break windows determine the earliest and latest break start times (i.e. pu1, qu1, etc.) and the break placement requirements in the break constraints. Except for the no-extraordinary break window overlaps and contiguous break window requirements of Bechtold and Jacobs (1990) (both are relaxed in Aykin, 1995a,b), the extended formulation of Bechtold and Jacobs given above and the implicit modeling approach given by Aykin are equivalent to the set covering formulation.

Also note that the number of variables and the number of constraints in the above formulation can be reduced by substituting one of the break constraints (25)–(27) associated with each shift in the other break constraints and the demand constraints (24). Our tests with this reduced form, however, did not show any computational advantage over the basic model in terms of model reliability and solution time.

3. Experimental evaluation

In this section, we describe the shift scheduling environment considered in the computational experiments, discuss the characteristics of the implicit formulations, and report our results with 220 problems.

3.1. Scheduling environment

We varied the scheduling environment considered in the test problems along four dimensions: relief and lunch break window sizes, shift start time pattern, cyclical-acyclical operations, and demand pattern. Employee demand is assumed to be determined for 96 quarter-hour ($= 4 \times 24$ hours) planning periods, $T = \{1, \ldots, 96\}$. Concerning the shift start times, we considered two schemes; (i) 9-hour shifts starting on the hour or half-hour, and (ii) 9-hour shifts starting every 15 minute. Each employee is assumed to be given one 30-minute lunch break and two 15-minute relief breaks (one before and one after the lunch break). We specify the break windows with reference to "ideal" break times as follows: the ideal first relief break time for a shift is specified as 2 hours after the start of the shift, the ideal lunch start time is 4 hours and 15 minutes (4 hours of work plus the first relief break) after the start of the shift, and the ideal second relief break time is 6 hours and 45 minutes (6 hours of work plus the first relief and lunch breaks) after the start of the shift. The break windows are specified by setting the earliest break start times half an hour earlier than the ideal break start times. To test the modeling approaches in situations with varying difficulty level, we considered 11 break window length combinations allowing between 4 and 7 break start times for the relief breaks (1 hour to 1 hour and 45 minutes windows) and between 3 and 7 lunch break start times (1–2 hour windows). Together with these, we considered five demand patterns (Fig. 1). The following three demand patterns were reported in the literature and were either obtained from or resembling real patterns from service systems: (i) the trimodal demand profile given in Segal (1974) for telephone operators with a minimum quarter hour demand of 2 and a maximum of 89, (ii) the bimodal demand profile with a minimum quarter hour demand of 2 and a maximum of 9 reported by Henderson and Berry (1976) for the General Telephone and Electronics Company of California, and (iii) the unimodal demand profile with a minimum quarter hour demand of 2 and a maximum of 27 given in Buffa et al. (1976) for the General Telephone Company of California. The two other demand patterns were synthetically generated; a bimodal and a trimodal sinusoidal demand pattern with a minimum quarter hour demand of 5 and a maximum of 50. We generated a total of 220 problems by considering both cyclical and acyclical versions of the combinations of these factors and their levels.

3.2. Model size characteristics

The number of variables, number of constraints, number of nonzeros and the density of the A-matrix for the implicit formulations of Bechtold and Jacobs (1990), Aykin (1996), and the set covering formulation are shown for some representative cases (from the smallest to the largest considered in this study) in Tables 1 and 2, respectively. In these tables, n_K is the number of shift start times in K, and $n1_k$, $n2_k$, and nL_k show the number of break start times in the first relief, second relief, and lunch break windows, respectively. As can be seen from Table 1, in the acyclical problems considered, the number of variables (that is, the number of shift variations) in the set covering model varies between 1984 and 20 923. In contrast, the extended implicit modeling approach of Bechtold and Jacobs requires between 223 and 262, and the implicit modeling approach of Aykin requires between 403 and 1342 variables. In every case, the extended formulation of Bechtold and Jacobs requires the smallest number of variables. Aykin's approach requires more than the extended formulation of Bechtold and Jacobs but significantly less than the set covering formulation. The formulation of Aykin, on the other hand, has a substantially smaller number of nonzeros in the A-matrix than the other two approaches; while the set covering formulation

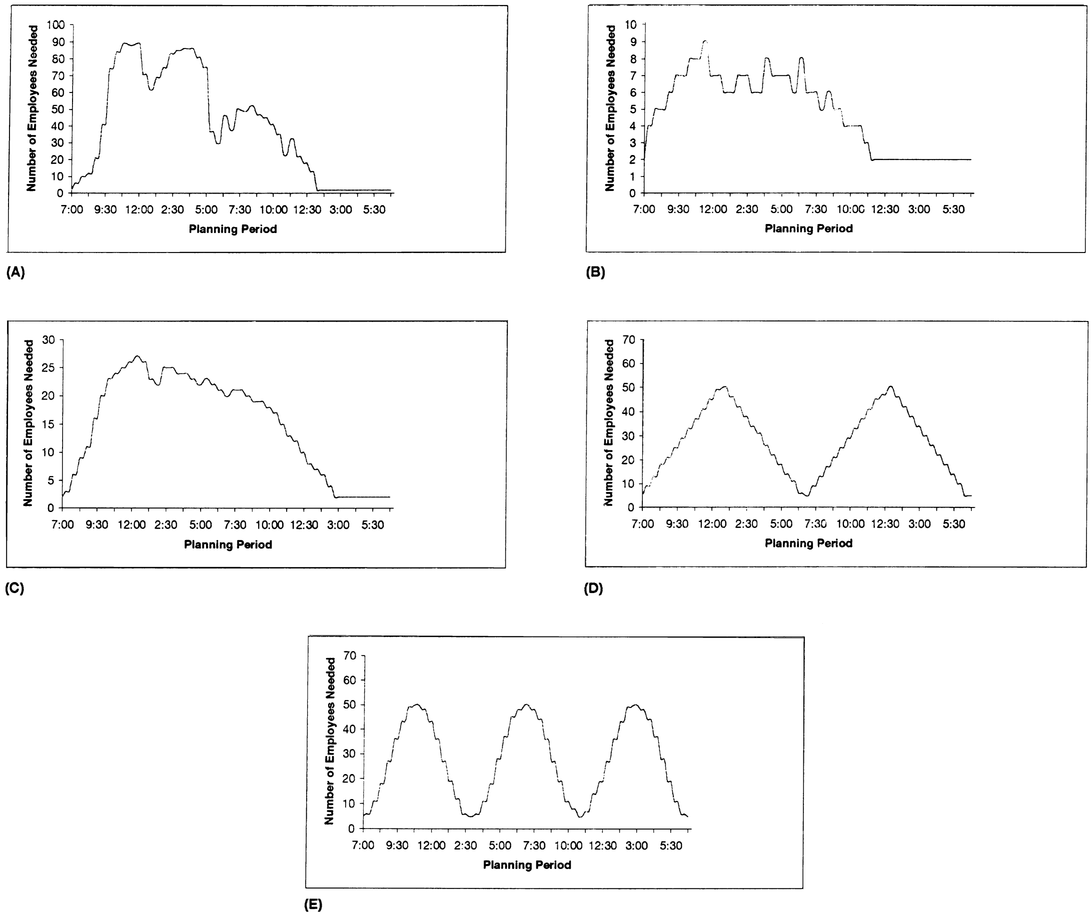


Fig. 1. Demand patterns: (A) Segal (1974), (B) Henderson and Berry (1976), (C) Buffa et al. (1976), (D) bimodal demand pattern, and (C) trimodal demand pattern.

has between 63 488 and 1 053 696 nonzeros and the formulation of Bechtold and Jacobs has between 10 387 and 56 124 nonzeros in the A-matrix, the formulation of Aykin (1995a,b) has between 2077 and 8448 nonzeros (i.e. about 80% less than the equivalent formulation of Bechtold and Jacobs, and 96.7–99.2% less than the equivalent set covering formulation). Furthermore, while the density of the A-matrix of Aykin's formulation varies between 1.5% and 2.9%, it is between 16.8% and 22.4% for the A-matrix of the extended formulation of Bechtold and Jacobs, and 33.3% for the set covering formulation. Note that since, in the set covering model, all shift variations are enumerated, only demand constraints are needed. As can be seen from Table 1, the implicit modeling approaches require more constraints in order to assure correct break placements. Similar conclusions can be reached from the results presented in Table 2 for the cyclical problems.

As discussed before, the implicit formulations of this problem are taking advantage of the fact that only some of the information about the break times generated by the set covering formulation is needed in the model to obtain an optimal schedule. This is true since, in general, the shift scheduling problem has a large number of alternative optimal solutions. In most cases, given an optimal solution, one can generate alternative optimal schedules by

Table 1
Problem characteristics: acyclical problems

n_K	$n_{1k} = n_{2k}$	n_{Lk}	Number of variables			Number of constraints			Number of nonzeros in A-matrix			% Nonzeros in A-matrix		
			Set covering	Bechtold and Jacobs	Aykin	Set covering	Bechtold and Jacobs	Aykin	Set covering	Bechtold and Jacobs	Aykin	Set covering	Bechtold and Jacobs	Aykin
31	4	4	1984	223	403	96	279	189	63 488	10 387	2077	33.3	16.8	2.9
	5	3	2325	224	434	96	279	189	74 400	10 448	2108	33.3	16.8	2.7
	6	5	5580	228	558	96	279	189	178 560	10 698	2418	33.3	16.9	2.4
	7	7	10 638	232	682	96	279	189	340 256	10 964	2728	33.3	17.0	2.2
61	4	4	3904	253	793	96	459	279	124 928	25 867	4087	33.3	22.3	1.9
	5	3	4575	254	854	96	459	279	146 400	25 988	4148	33.3	22.3	1.8
	6	5	10 980	258	1098	96	459	279	351 360	26 478	4758	33.3	22.4	1.6
	7	7	20 923	262	1342	96	459	279	669 536	26 968	5368	33.3	22.4	1.5

Table 2
Problem characteristics: cyclical problems

n_K	$n_{1k} = n_{2k}$	n_{Lk}	Number of variables			Number of constraints			Number of nonzeros in A-matrix			% Nonzeros in A-matrix		
			Set covering	Bechtold and Jacobs	Aykin	Set covering	Bechtold and Jacobs	Aykin	Set covering	Bechtold and Jacobs	Aykin	Set covering	Bechtold and Jacobs	Aykin
48	4	4	3072	348	624	96	378	240	98 304	22 042	3216	33.3	16.8	2.1
	5	3	3600	350	672	96	378	240	115 200	22 136	3264	33.3	16.7	2.0
	6	5	8640	358	864	96	378	240	276 480	21 996	3744	33.3	16.3	1.8
	7	7	16 464	378	1056	96	378	240	526 848	21 868	4224	33.3	15.8	1.7
96	4	4	6144	402	1248	96	666	384	196 608	57 522	6432	33.3	21.5	1.4
	5	3	7208	404	1344	96	666	384	230 400	57 364	6528	33.3	21.3	1.3
	6	5	17 280	412	1728	96	666	384	552 960	56 720	7488	33.3	20.7	1.2
	7	7	32 928	420	2112	96	666	384	1 053 696	56 124	8448	33.3	20.1	1.1

Table 3
Summary of computational experiments

	Bechtold and Jacobs	Aykin
Total number of problems considered	220	220
Number of problems remained unsolved		
After 1st pass	45	15
After 2nd pass	32	2
Number of problems remained unsolved with both approaches		
After 1st pass	11	11
After 2nd pass	1	1
Number of problems with same CPU time[a]	4	4
Number of problems with a better CPU time	12 (5.5%)	203 (92.3%)
Average[b] CPU time (s) per problem	331.36 (188)	113.04 (218)
Lower bound on CPU time (s) per problem	504.85	200.53

[a] The difference between the solution times $\leqslant 0.5$ s.
[b] After two passes.

switching the breaks of different shifts that have overlapping break windows, provided this does not violate any of the demand or break constraints. Consequently, the set covering, and the implicit modeling approaches are three possible strategies to solve this problem; as the set covering formulation requires, one may enumerate all rest and work patterns and determine only one of these optimal solutions that provides a complete schedule or, as in the extended formulation of Bechtold and Jacobs (1990), an optimal solution specifies only the active shifts and sufficient number of breaks,and a complete schedule is obtained from this solution by postprocessing. As a third strategy, the implicit formulation of Aykin (1996) is providing an approach that is taking advantage of the fact that one can formulate the problem to obtain slightly more information in the optimal solution than Bechtold and Jacobs (1990) (but still less than the set covering formulation) by scheduling shifts and breaks associated with each shift separately. Thus, the formulation of Bechtold and Jacobs (1990) is more compact in terms of the number of variables. The primary advantage of Aykin (1996), on the other hand, is that it requires significantly smaller number of constraints and nonzeros in the A-matrix than Bechtold and Jacobs (1990). As discussed in the computational results section, these approaches are significantly different in the amount of computational effort needed and reliability.

3.3. Experimental results

All 220 problems were formulated using the implicit modeling approaches of Bechtold and Jacobs (1990) and Aykin (1996), and were solved using LINDO (Version 5.3.) together with a FORTRAN interface that generates and inputs problem data. All runs were completed in batch mode on a Sun Sparc 512 computer. Since the superiority of both implicit modeling approaches to the set covering formulation has been demonstrated separately with computational experiments in Bechtold and Jacobs (1990), and Aykin (1998), the set covering formulation was not considered in our computation experiments. We also note that the memory requirements (e.g. number of variables) for the set covering formulation of most of the problems considered in our study would exceed the limits of the software/hardware combination used.

Both Bechtold and Jacobs (1991) and Aykin (1996) reported that the break variables in their formulations tend to be integer when the shift variables are integer. Therefore, a branching strategy giving higher priority to the shift variables X_k was implemented in LINDO. The CPU time limit was set to 1200 seconds. No attempt was made to determine the branching requirements and total solution time if LINDO could not locate an integer optimal solution to a problem within the CPU time limit. As seen in Table 3, this was the case in 45 (20.5%) out of 220 problems with the

Table 4
Computational results

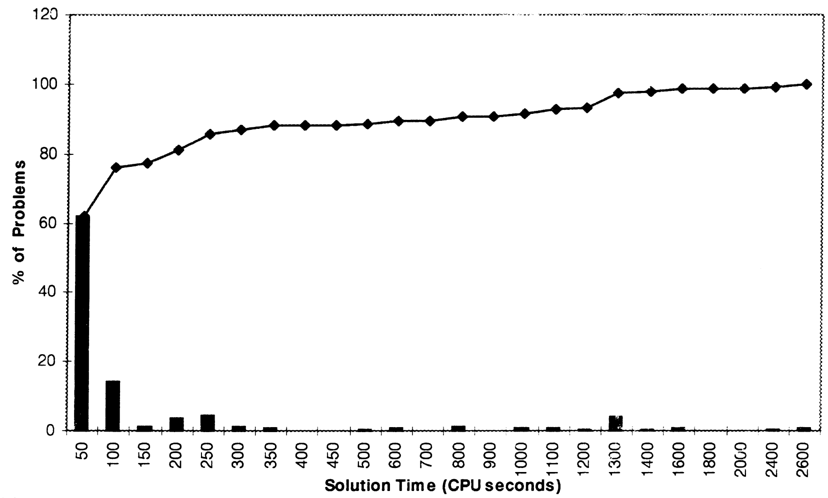
Number of shift start times (n_K)	Number of shift variations			Bechtold and Jacobs				Aykin			
				Solution time[a]			Number of prob. remain unsolved	Solution time[a]			Number of prob. remain unsolved
	Min.	Avg.	Max.	Min.	Avg.	Max.		Min.	Avg.	Max.	
31	1984	4856	10 638	3.77	171.01 (292.59)	1231.25	3	3.16	97.44 (147.4)	1226.30	0
48	3072	7519	16 464	48.88	274.76 (313.40)	1342.58	1	5.20	98.20 (96.86)	1224.73	0
61	3904	9555	20 923	19.46	244.17 (438.76)	1437.14	6	5.63	119.02 (231.84)	1414.82	1
96	6144	15 037	32 928	165.93	806.14 (974.66)	2376.66	22	16.32	138.06 (326.03)	1419.65	1

[a] CPU seconds.

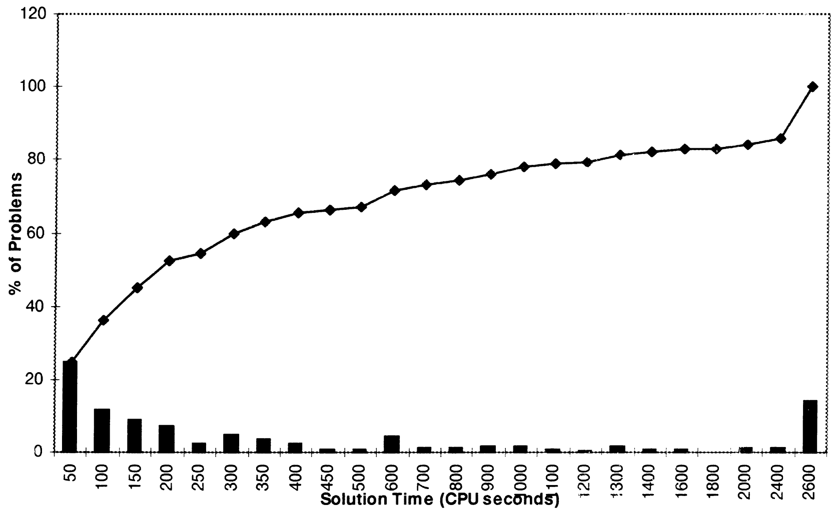
extended formulation of Bechtold and Jacobs, and in 15 (6.8%) out of 220 problems with the approach of Aykin. These cases include the 11 problems in which both approaches failed to locate an optimal integer solution within the allowed time. In order to gain more information, we made a second attempt using a branching strategy giving a higher priority to the variables associated with the shifts covering higher total demand during its shift span. In case of a tie, the variables associated with the shifts with an earlier start time were chosen for branching. If a second attempt was made, the total time for both passes is reported. Thus, the CPU time limit for two passes is 2400 seconds. Out of 45 problems which remained unsolved after the first pass with the approach of Bechtold and Jacobs, 13 more solved in the second pass. With the approach of Aykin, on the other hand, out of 15 not solved during the first pass, only 2 problems remained unsolved after the second pass. In summary, after two passes, 32 problems with the extended formulation of Bechtold and Jacobs, or about 14.5% of the problems, and 2 problems with the formulation of Aykin, or about 0.9%, remained unsolved within the allotted time.

As can be seen from Table 3, in 203 (92.3%) of the problems, the formulation given in Aykin (1996) required less CPU time to locate an optimal integer solution than the extended formulation of Bechtold and Jacobs (1990). In 12 problems, the extended formulation of Bechtold and Jacobs (1990) required less CPU time than that of Aykin (1996). And in 4 of the remaining 5 problems (out of 220), the difference between the solution times with the two formulations was less than 0.5 seconds. Hence, these cases were reported as requiring the same amount of CPU time. Only one case remained unsolved with both formulations.

Out of 220 problems, 187 of them were solved optimally by both formulations. The average CPU time for these problems with the extended formulation of Bechtold and Jacobs is 331.36 seconds. In contrast, for the same 187 problems, the average CPU time with the formulation of Aykin is 113.04, about 1/3 of the average time needed with the formulation of Bechtold and Jacobs. Table 4 provides more detailed information concerning the solution times. In this table, the minimum, average, and maximum number of shifts, and minimum, average, and maximum CPU times with both formulations for the problems in each group are also presented. As seen from these results, the average solution time increases with both formulations as the number of shifts increases. The rate of increase with the formulation of Bechtold and Jacobs is, however, significantly higher than with the formulation of Aykin. Also note that the average solution time needed with the extended formulation of Bechtold and Jacobs in these problem groups is between 1.76 and 5.84 times higher than that with the formulation of Aykin. Table 4 also shows the number of problems remained unsolved (after two passes) in each group. For each group, below the average solution time, we report (in parenthesis) the average CPU time spent per problem in the group to include the unsolved



(A)



(B)

Fig. 2. Distribution of the solution times: (A) with the formulation of Aykin (1996), (B) with the formulation of Bechtold and Jacobs (1990).

problems. These averages were obtained by including 2400 seconds spent for each case that is remaining unsolved after two passes together with the actual solution times for the problem that were solved optimally. These average times provide a lower bound on the average solution times per problem. As can be seen, the lower bounds for the formulation of Bechtold and Jacobs are also higher than the corresponding lower bounds for the formulation of Aykin.

Finally, Fig. 2 shows the distribution of the individual solution times. In this figure, the histogram shows the percentage of the problems with a solution time in the given interval, and the curve shows the cumulative percentage of the problems versus the solution time. The dashed lines mark the points where the scale of the solution time axis is changed due to the spread of the recorded solution times. As can be seen from this figure, in more than 60% of the problems, an optimal integer solution was obtained within the first 50 seconds with the formulation of Aykin and, in more than 75% of the problems, within 100 seconds. Also note that about 15% of the problems required between 100 and 500 seconds and only 10% required more than 500 seconds. In contrast, with the extended formulation of Bechtold and Jacobs, only about 25% of the problems were solved within the first 50 seconds, and about 37% within the first 100 seconds. About 30% of the problems required between 100 and 500 seconds, and 33% required more than 500 seconds. This approach requires more than 2400 seconds in 14.5% of the problems that remained unsolved.

As discussed in Section 3.2, in terms of the number of constraints, the extended formulation of Bechtold and Jacobs (1990) is a more compact formulation of the problem than Aykin (1996). However, the results presented in this section show that this is not providing any computational advantage over Aykin (1996). A comparison of the optimal LP objective values revealed that both formulations are producing the same LP objective values. However, in most cases (as reported in Table 3, 92.3%), solving (i.e. pivoting) the initial LP relaxation as well as the subsequent branch subproblems consume significantly more time with Bechtold and Jacobs (1990) than Aykin (1996). One reason for this may be the substantially higher density of the A-matrix of the Bechtold and Jacobs formulation. In summary, these results show that the formulation given in Aykin (1996) offers better modeling reliability, and requires less memory and, on average, one-third of the time needed with the formulation of Bechtold and Jacobs (1990).

4. Summary

In this paper, we experimentally compared the extended formulation of Bechtold and Jacobs (1990) and the implicit modeling approach of Aykin (1996) to the general shift scheduling problem through solving 220 problems. We considered two criteria; model reliability and solution time. In terms of model reliability, while 32 (14.5%) out of 220 problems remained unsolved after two passes with the extended formulation of Bechtold and Jacobs, only 2 (0.9%) of the problems remained unsolved with the formulation of Aykin. Further, the average solution time for the problems solved by both approaches was 331.36 seconds with the extended formulation of Bechtold and Jacobs, and 113.04 seconds with the formulation of Aykin. In summary, our results show that the formulation given in Aykin (1996), although requiring more variables than the equivalent formulation of Bechtold and Jacobs, requires smaller number of constraints, has lower density and smaller number of nonzeros in the A-matrix, offers better model reliability, and requires on average one-third of the time needed with the extended formulation of Bechtold and Jacobs. The fact that these results were obtained with a problem set including the largest shift scheduling problems solved optimally in the literature using an off-the-shelf integer programming software is very encouraging in terms of the practical applications of the implicit scheduling models in service systems.

References

Aykin, T., 1998. A composite branch and cut algorithm for optimal shift scheduling with multiple breaks and break windows. Journal of the Operations Research Society 48, 1–13.

Aykin, T., 1996. Optimal shift scheduling with multiple break windows. Management Science 42, 591–603.

Aykin, T., 1995a. A generalized implicit modeling approach for the shift and tour scheduling problems. GSM Working Paper #95-29, Graduate School of Management, Rutgers University, Newark, NJ.

Aykin, T., 1995b. Reformulating the tour scheduling problem to improve solvability. GSM Working Paper #95-35, Graduate School of Management, Rutgers University, Newark, NJ.

Baker, K., 1976. Workforce allocation in cyclical scheduling problems: A survey. Operational Research Q 27, 155–167.

Bartoldi, J.J., 1981. A guaranteed-accuracy round-off algorithm for cyclic scheduling and set covering. Operations Research 29, 501–510.

Bartoldi, J.J., Orlin, J.B., Ratliff, H.D., 1980. Cyclic scheduling via integer programs with circular ones. Operations Research 29, 1074–1085.

Bechtold, S.E., Jacobs, L.W., 1990. Implicit modeling of flexible break assignments in optimal shift scheduling. Management Science 36 (11), 1339–1351.

Bedworth, D.D., Bailey, J.E., 1987. Integrated Production Control Systems, second ed. Wiley, New York.

Buffa, E.S., 1980. Modern Production and Operations Management, sixth ed. Wiley, New York.

Dantzig, G.B., 1954. A comment on Edie's "Traffic delays at toll booths". Operations Research 2 (3), 339–341.

Edie, L.C., 1954. Traffic delays at toll booths. Operations Research 2 (2), 107–138.

Gaballa, A., Pearce, W., 1979. Telephone sales manpower planning at Qantas. Interfaces 9 (3), 1–9.

Henderson, W.B., Berry, W.L., 1976. Heuristic methods for telephone operator shift scheduling: An experimental analysis. Management Science 22 (12), 1372–1380.

Jarrah, A.I.Z., Bard, J.F., deSilva, A.H., 1994. Solving large-scale tour scheduling problems. Management Science 40, 1124–1144.

Love, R.R., Hoey, J., 1990. Management science improves fast-food operations. Interfaces 20, 21–29.

Moondra, S.L., 1976. An L.P model for work force scheduling for Banks. Journal of Bank Research 6, 299–301.

Nanda, R., Browne, J., 1992. Introduction to Employee Scheduling. Van Nostrand Reinhold, New York.

Segal, M., 1974. The operator-scheduling problem: A network flow approach. Operations Research 22, 808–823.

Spencer III, T., Brigandi, T., Barlow, G., Gomez, R., 1992. A decision support system for workforce management. Modelling and Simulation 23 (5).

Thompson, G.M., 1995. Improved implicit optimal modeling of the labor shift scheduling problem. Management Science 41, 595–607.

Vohra, R.V., 1988. A quick heuristic for some cyclic staffing problems with breaks. Journal of the Operational Research Society 39, 1057–1061.