

Improved Implicit Optimal Modeling of the Labor Shift Scheduling Problem

Gary M. Thompson

Cornell University

Published in *Management Science* (1995), 41(4), 595-607

Author Note:

Gary M. Thompson is professor of Services Operations Management in the School of Hotel Administration at Cornell University.

The author thanks Professor Thomas and the anonymous referees for their helpful suggestions. This work was supported by a summer research grant while the author was on the faculty of the Eccles School of Business at the University of Utah.

Correspondence concerning this article should be addressed to Gary Thompson, School of Hotel Administration, Cornell University, Ithaca, NY 14853.

### Abstract

This paper presents an integer programming model for developing optimal shift schedules while allowing extensive flexibility in terms of alternate shift starting times, shift lengths, and break placement. The model combines the work of Moondra (1976) and Bechtold and Jacobs (1990) by implicitly matching meal breaks to implicitly represented shifts. Moreover, the new model extends the work of these authors to enable the scheduling of overtime and the scheduling of rest breaks. We compare the new model to Bechtold and Jacobs' model over a diverse set of 588 test problems. The new model generates optimal solutions more rapidly, solves problems with more shift alternatives, and does not generate schedules violating the operative restrictions on break timing.

Keywords: Labor Scheduling; Integer Programming; Implicit Modeling

## 1. Introduction

Labor scheduling—the process of matching, over the operating day, the number of employees working to the number of employees needed to provide the desired level of customer service—is frequently a large determinant of service organization efficiency. Dantzig (1954) was the first to model labor scheduling in mathematical programming form. Because of the explicit representation of shifts in Dantzig's model, the number of alternate shifts grows rapidly as flexibility (in terms of shift starting times, shift lengths, and break timing) increases.

Table 1 summarizes solution methodologies, numbers of alternate shifts, and planning interval durations used in the shift scheduling literature. Since real organizations commonly exhibit a high level of scheduling flexibility, it is not surprising that most published shift scheduling procedures are heuristics. The largest problem optimally solved contained 970 alternate shifts (Bechtold and Jacobs 1990). Most commonly, planning intervals have been 30 minutes or longer, a result of the difficulty in optimally solving models using shorter periods. Despite this, there are good reasons to use short-duration (10- or 15-minute) periods. First, short-duration periods are superior with higher variation of customer demand (Thompson 1991a). Second, short duration periods enable the scheduling of relief (rest) breaks. As Keith (1979) implies and as Thompson (1988) notes, it is desirable to represent at least some subset of the breaks to be scheduled. Not surprising, therefore, one sees short-duration periods used in research based on actual scheduling situations, such as Glover et al. (1984), Henderson and Berry (1976, 1977), Keith (1979), Luce (1973), and Thompson (1988).

Two insightful approaches to shift scheduling offered improvements over Dantzig's (1954) model. Moondra (1976) presented a method of implicitly representing shifts without meal breaks. He defined variable sets for the number of shifts starting and finishing in each planning

period, and used constraints to impose limits on the shifts' allowable durations. His approach facilitates modeling a high degree of shift length and starting time flexibility. Bechtold and Jacobs (1990, 1991) implicitly matched meal breaks to explicitly-represented shifts. Their model, called BJSSM here, was superior to Dantzig's model (1954) in terms of execution time, computer memory requirements, and its ability to optimally solve problems having greater meal-break-timing flexibility.

---

Insert Table 1 Here

---

(Implicit modeling has also been used by Thompson (1990) for scheduling employees having limited availability and by Thompson (1992) for scheduling work in services over which management has some temporal control.)

This paper presents a model which we call DISSM. By integrating the implicit shift modeling of Moondra (1976) with the implicit meal-break modeling of Bechtold and Jacobs (1990), DISSM is particularly effective in representing a high degree of scheduling flexibility. (In fact, we report the generation of optimal schedules on an 486DX33-based personal computer for an environment having 15,885 alternate shifts.) The paper also compares the relative performance of DISSM and BJSSM over a diverse set of 588 test problems.

The structure of the remainder of this paper is as follows: §2 presents the doubly implicit integer programming (IP) shift scheduling model, §3 describes the test problems used in evaluating the performance of DISSM and BJSSM, and §4 provides results. A discussion in §5 concludes the paper.

## **2. A Doubly-Implicit IP Shift Scheduling Model (DISSM)**

In this section, we define relevant terms and variables, present an example of DISSM, present a formal, mathematical definition of the integrative model, identify model-related assumptions, and address issues relevant to the application of the new model.

## 2.1. Terms and Variables

Shifts are uniquely specified by: the time at which they commence; the number of periods of work that they contain; and, if they receive a meal break, the length of the break and the time at which the break starts. Define:

*Overall shift length*—the total duration of a shift, in planning periods (including working and meal break periods);

*Work stretch*—the number of planning period(s) of work within a shift that are uninterrupted by a meal break (typically there are two work stretches in a shift, separated by a meal break); and

*Shift type*—a set of shifts having the same cost, per working period; the same meal break duration; identical restrictions on minimum and maximum overall shift duration; and, if applicable, identical restrictions on the minimum and maximum pre- and post-meal-break work stretch durations.

Define also the following variables:

$f_{tp}$  = number of type- $t$  shifts finishing at the end of period  $p$ ;

$m_{tp}$  = number of meal breaks for type- $t$  shifts commencing at the beginning of period  $p$ ;

and

$s_{tp}$  = number of type- $t$  shifts starting at the beginning of period  $p$ .

The break variables originate in BJSSM, while the others originate in Moondra's (1976) model. However, all three variable sets are newly defined with the subscript representing shift types.

## 2.2. An Example of DISSM for a Simple Scheduling Scenario

In this section we present an example of DISSM for a simple scheduling scenario. This example serves to clarify the general mathematical form of the model, which we present in §2.3.

The scheduling scenario assumes: a nine-period operating day; all shifts have the same cost, per working period; shifts are five to seven periods in overall duration; shifts receive a period-long meal break that must be preceded and followed by at least two but no more than four periods of work; and the employee requirements for periods one through nine are 1, 3, 4, 6, 5, 6, 4, 3, and 1 employee(s), respectively.

There is only a single shift type for this example. Since all shifts have the same cost, per working period, minimizing the cost of the schedule is equivalent to minimizing the number of periods worked. The number of periods worked is found by summing the difference between shift finishing and starting times, and subtracting the number of periods employees are on meal breaks. Thus, the objective is to:

$$\begin{aligned} \text{Min } Z = & -s_{1.2} - 2s_{1.3} - 3s_{1.4} - 4s_{1.5} + 5f_{1.5} + 6f_{1.6} + 7f_{1.7} + 8f_{1.8} + 9f_{1.9} - m_{1.3} - m_{1.4} \\ & - m_{1.5} - m_{1.6} - m_{1.7}. \end{aligned}$$

The first set of restrictions ensure that sufficient numbers of employees are present in each planning interval. One can determine the number of staff working in a period by summing the number of employees who have started work in the period or earlier, and subtracting out both



eight or nine, thus violating the maximum acceptable overall length). Also, more shifts cannot start in periods one and two than finish in periods five through eight. We thus obtain:

$$\begin{aligned} s_{1,1} &\leq f_{1,5} + f_{1,6} + f_{1,7} \\ s_{1,1} + s_{1,2} &\leq f_{1,5} + f_{1,6} + f_{1,7} + f_{1,8}. \end{aligned}$$

The fifth type of restriction equates the number of shift starts to the number of meal breaks scheduled:

$$\begin{aligned} s_{1,1} + s_{1,2} + s_{1,3} + s_{1,4} + s_{1,5} \\ = m_{1,3} + m_{1,4} + m_{1,5} + m_{1,6} + m_{1,7} \end{aligned}$$

Minimum pre-meal-break work stretches are imposed in the sixth set of restrictions. First, the number of shifts starting in period one must equal or exceed the number of meal breaks commencing in period three. If this were not true, then some meal breaks commencing in period three would be for shifts starting in periods two or later, thus violating the two-period minimum for the pre-meal-break work stretches. Next, the number of shifts starting in periods one and two must equal or exceed the number of meal breaks commencing in periods three and four, etc., so we obtain:

$$\begin{aligned} s_{1,1} &\geq m_{1,3} \\ s_{1,1} + s_{1,2} &\geq m_{1,3} + m_{1,4} \\ s_{1,1} + s_{1,2} + s_{1,3} &\geq m_{1,3} + m_{1,4} + m_{1,5} \\ s_{1,1} + s_{1,2} + s_{1,3} + s_{1,4} &\geq m_{1,3} + m_{1,4} + m_{1,5} + m_{1,6}. \end{aligned}$$

The seventh group of restrictions imposes the minimum post-meal-break work stretches. First, the number of shifts finishing in period nine must equal or exceed the number of meal breaks commencing in period seven. If this were not true, some meal breaks commencing in

period seven would be for shifts finishing in periods eight, or earlier, thus violating the two-period minimum for the post-meal-break work stretch. Next, the number of shifts finishing in periods eight and nine must equal or exceed the number of meal breaks commencing in periods six and seven, etc., giving:

$$\begin{array}{rcl}
 f_{1,9} & \geq & m_{1,7} \\
 f_{1,8} + f_{1,9} & \geq & m_{1,6} + m_{1,7} \\
 f_{1,7} + f_{1,8} + f_{1,9} & \geq & m_{1,5} + m_{1,6} + m_{1,7} \\
 f_{1,6} + f_{1,7} + f_{1,8} + f_{1,9} & \geq & m_{1,4} + m_{1,5} + m_{1,6} + m_{1,7}.
 \end{array}$$

Restrictions imposing the four-period maximum pre- and post-meal-break work stretches are unnecessary in this example. The reason for this is that the longest pre-meal-break (post-meal-break) work stretch is implicitly set at four periods: the longest overall shift length is seven periods, the meal break is one period, and the shortest post-meal-break (pre-meal-break) work stretch is two periods.

In the optimal solution to the example: employees perform 36 periods of work (compared to a lower bound of 33 employee-periods); the shift start variables  $s_{1,1}, \dots, s_{1,5}$  have values of 1, 2, 3, 2, and 0, respectively; the shift finish variables  $f_{1,5}, \dots, f_{1,9}$  have values of 0, 3, 1, 3, and 1, respectively; and the shift break variables  $m_{1,3}, \dots, m_{1,7}$  have values of 1, 2, 1, 3, and 1, respectively. Table 2 shows how to match DISSM's shift starts to shift finishes (and shift starts to meal breaks) on a first-in-first-out basis, yielding explicit shifts from the implicit model. For example, the first reconstructed shift starts in period one, finishes in period six, and has a meal break commencing in period three, while the second reconstructed shift starts in period two, finishes in period six, and has a meal break starting in period four.

---

Insert Table 2 Here

---

### 2.3. A Formal, Mathematical Definition of the New, Integrative Model

Integrating the models of Moondra (1976) and Bechtold and Jacobs (1990), DISSM is:

$$\begin{aligned} \text{Min } Z = & \sum_{t \in T} c_t \left[ \sum_{p=ef_t}^P p \cdot f_{tp} - \sum_{p=1}^{ls_t} (p-1) \cdot s_{tp} \right] \\ & - \sum_{t \in T_m} (c_t \cdot mbl_t) \cdot \left[ \sum_{p=emb_t}^{lmb_t} m_{tp} \right] \end{aligned}$$

subject to

$$\begin{aligned} & \sum_{t \in T} \left[ \sum_{j=1}^p s_{tj} - \sum_{j=ef_t}^{p-1} f_{tj} \right] \\ & - \sum_{t \in T_m} \sum_{j=p-mbl_t+1}^p m_{tj} \geq d_p \quad \text{for } p = 1, \dots, P, \end{aligned}$$

$$\sum_{p=1}^{ls_t} s_{tp} - \sum_{p=ef_t}^P f_{tp} = 0 \quad \text{for } t \in T,$$

$$\sum_{p=1}^i s_{tp} - \sum_{p=ef_t}^{i+nl_t-1} f_{tp} \geq 0$$

$$\text{for } t \in T \quad \text{and } i = 1, \dots, ls_t - 1,$$

$$\sum_{p=1}^i s_{tp} - \sum_{p=ef_t}^{i+xl_t-1} f_{tp} \leq 0$$

$$\text{for } t \in T \quad \text{and } i = 1, \dots, P - xl_t,$$

$$\sum_{p=1}^{ls_t} s_{tp} - \sum_{p=emb_t}^{lmb_t} m_{tp} = 0 \quad \text{for } t \in T_m,$$

$$\sum_{p=1}^i s_{tp} - \sum_{p=emb_t}^{i+nmb_t} m_{tp} \geq 0$$

$$\text{for } t \in T_m \quad \text{and } i = 1, \dots, ls_t - 1,$$

$$\sum_{p=1}^i s_{tp} - \sum_{p=emb_t}^{i+xmb_t} m_{tp} \leq 0$$

$$\text{for } t \in XBMB \quad \text{and } i = 1, \dots, ls_t - 1,$$

$$\text{for } t \in T_m \quad \text{and } i = ef_t + 1, \dots, P,$$

$$\sum_{p=i}^P f_{tp} - \sum_{p=i-xamb_t-mbl_t+1}^{lmb_t} m_{tp} \leq 0$$

$$\text{for } t \in XAMB \quad \text{and } i = ef_t + 1, \dots, P,$$

$$s_{tp}, f_{tp}, m_{tp} \geq 0 \text{ and integer}$$

$$\text{for } t \in T \quad \text{and } p = 1, \dots, P,$$

where managerially-specified constants and sets are defined as:

$c_t$	= the cost per working period of type- $t$ shifts;
$d_p$	= the number of employees needed in period $p$ to provide the desired level of customer service;
$mlb_t$	= number of periods in a meal break for type- $t$ shifts;
$namb_t$	= minimum number of post-meal-break working periods in type- $t$ shifts;
$nbmb_t$	= minimum number of pre-meal-break working periods in type- $t$ shifts;
$nl_t$	= minimum overall length of a type- $t$ shift, in periods;
$P$	= the number of planning periods in the day;
$T$	= the set of shift types;
$xamb_t$	= maximum number of post-meal-break working periods in type- $t$ shifts;
$xbmb_t$	= maximum number of pre-meal-break working periods in type- $t$ shifts; and
$xl_t$	= maximum overall length of a type- $t$ shift, in periods.

Other constants and sets are defined as:

$ef_t$	= earliest possible finishing time for any type- $t$ shift, = $nl_t$ ;
$emb_t$	= earliest possible meal break starting time for type- $t$ shifts, = $1 + nbmb_t$ ;
$lmb_t$	= latest possible meal break starting time for type- $t$ shifts, = $P + 1 - namb_t - mbl_t$ ;
$ls_t$	= latest possible starting time for any shift of type $t$ , = $P + 1 - nl_t$ ;
$T_m$	= shift types requiring a meal break, = $\{t \in T   mbl_t > 0\}$ ;
$XAMB$	= the shift types requiring maximum post-meal-break work stretch restrictions, = $\{t \in T   xamb_t < xl_t - nbmb_t - mbl_t\}$ ; and
$XBMB$	= the shift types requiring maximum pre-meal-break work stretch restrictions, = $\{t \in T   xbmb_t < xl_t - namb_t - mbl_t\}$ .

DISSM has as its objective (1) the minimization of total paid labor. Total paid labor is, for each shift type, given by the number of employee-periods worked for shifts of the type multiplied by the relative cost of shifts of the particular type. Constraint set (2) ensures that each planning period receives enough staff to provide the desired level of service. Constraint set (3) equates the number of starts and finishes for shifts of each type. Constraint sets (4) and (5) limit the minimum and maximum overall length of each type of shift, respectively. Constraint set (6) schedules the correct number of meal breaks, while constraint sets (7) through (10) fix the

minimum and maximum pre- and post-meal- break work stretch durations, respectively.

Constraint sets (6)-(10) are present only for those shift types that receive a meal break.

The objective (1) and constraint sets (2), (3), (4), and (5) originate in Moondra's (1976) model. Because Moondra's implicitly-defined shifts did not have breaks, the break component did not appear in the employee requirement constraints nor in the objective of his model. Constraint sets (6), (7), and (9), and the meal break component of constraint set (2) originate in BJSSM. Since BJSSM uses explicit shifts, its constraints differ somewhat from the form of constraint sets (6), (7), and (9). With implicitly defined shifts, two constraint sets not present either in the model of Moondra (1976) or in BJSSM, sets (8) and (10), are necessary sometimes to ensure that meal breaks fall only in the allowed periods.

#### **2.4. Model-related Assumptions**

The assumptions inherent in DISSM are: (1) employees have identical skills; (2) employees are continuously available for work; (3) shifts receive at most a single break; (4) shifts and breaks may start at the beginning and finish at the end of any period; (5) the restrictions defining minimum and maximum pre- and post-meal- break work stretch durations and minimum and maximum overall shift durations are consistent within each shift type; and (6) for each shift type, the cost of shifts is a linear function of the number of working periods in the shift (there are no shift premiums and meal breaks are unpaid).

Assumptions 1-3 are standard; they exist in the majority of the published shift scheduling literature. Nonetheless, assumption 3 is easily relaxed. First, one must define integer variable sets for the number of pre- and post-meal-break reliefs started in each planning interval. Next, two constraints (similar to constraint set (6)) equate the number of shift starts to the number of pre-

and post-meal-break reliefs. Eight more constraint sets (similar to constraint sets (7)-(10)) define the minimum and maximum work stretches that occur prior to and after the pre- and post-meal-break reliefs. Thompson (1991b) shows a mathematical representation of the expanded model.<sup>1</sup>

Assumption 4 serves to simplify the presentation of DISSM and may be relaxed without affecting the model. Assumption 5 reflects the fact that DISSM implicitly defines shifts and implicitly matches meal breaks to shifts, a matching that occurs by restricting the execution of meal breaks based on the times at which shifts start (in constraint sets (7) and (8)) and based on the times at which shifts finish (in constraint sets (9) and (10)). The implicit nature of DISSM also necessitates assumption 6. Using any form of implicit modeling can result in an inability to represent a cost structure accurately. For example, if the cost of a shift is dependent on the time at which it starts, its duration, and the length and the start time of its breaks, then a model with explicit shifts is appropriate. An implicit model, in contrast, can only approximate the true cost of individual shifts. Assumption 6 may be relaxed somewhat by incorporating differential shift premiums (by associating different fixed costs with shifts of different types), or by imposing penalties on shifts starting or finishing at undesirable times. One can also account for paid meal breaks by removing the meal break term from the objective function and for overtime, as shown in Thompson (1991b).

## 2.5. Model Application

Solution procedures that store only non-zero A-matrix elements may benefit from a lower density of nonzeros in DISSM. This density can be reduced following a process similar to that reported by Bechtold and Jacobs (1990): substituting, whenever beneficial, constraints from set (3) into those from sets (2), (4), and (5); substituting, when worthwhile, constraints from set (6)

---

<sup>1</sup> This working paper, which is available from the author, contains details not present in the current paper.

into those from sets (7) and (8); and, when useful, by making a double substitution of constraints from sets (3) and (6) into constraints from sets (9) and (10).

Since solutions to DISSM implicitly represent shifts, one must convert them into usable form. This is done by reconstructing explicit shifts from the information on shift starting and finishing times, and then matching both meal and relief breaks to the reconstructed shifts. Both shift reconstruction and the matching of all breaks to shifts are most simply carried out on a first-in-first-out basis, as illustrated in Table 2, since such matchings are inherent in DISSM.

The FIFO matchings of shift starts to shift finishes and of breaks to reconstructed shifts ensure that DISSM only generates schedules that satisfy the operative restrictions on break timing, even when extraordinary break window overlap (EBWO) exists (as Thompson 1991b proves). EBWO, which was defined by Bechtold and Jacobs (1990) as a break window for one shift completely enveloping the break window for another, typically occurs with extensive shift length and break timing flexibility. As Bechtold and Jacobs (1990) note, EBWO can lead to BJSSM generating "optimal" solutions that fail to satisfy operative restrictions on break timing. We call such solutions "optimal-but-unacceptable."

### **3. Experimental Evaluation of Model Performance**

This section describes the two sets of problems selected to compare the models and reports on the procedures used in solving the models.

#### **3.1. Test Problem Sets**

The first problem set, TPS1, addresses scheduling cashiers in a grocery store reported by Thompson (1988). Sixty 15-minute planning intervals comprised the 15-hour operating day in

this environment. TPS1 contains twenty-one problems, differing on two dimensions. Figure 1 illustrates the seven employee requirements patterns, the first problem characteristic. The second problem characteristic is the number of allowable shift types. There are three shift type categories (STCs), each having the attributes described in Table 3. The number of alternate shifts in TPS1 ranges from 2543 in STC 1 to 3877 in STC 3.

Our objective in developing the second test problem set, TPS2, was to select a broad range of problems differing on multiple dimensions. A more diverse set of problems than that found in TPS1 is desirable in that it is less likely that a broadly-effective model will be outperformed by one that is superior only for problems that are unusual in some respect. TPS2 problems vary along four dimensions: the employee requirements pattern, the length of the planning intervals, the length of the operating day, and the restrictions defining acceptable shifts (shift-type categories, or STCs). Respectively, these factors have 9, 2, 3, and 11 levels, resulting in a total of 567 problems in TPS2 (there is only one planning interval length for one of the STCs).

The nine employee requirements patterns all had mean requirements of 20 employees per period. Four patterns—unimodal, bimodal, trimodal, and random—had two levels of variability. Variability, measured as a coefficient of variation in the requirements, had values of 0.2 and 0.4. The fifth pattern, uniform, had identical requirements in all planning periods. Collectively, these patterns are representative of those occurring in many service organizations. Figure 2 shows examples of all save the uniform pattern.

Since we believed that DISSM would perform better than BJSSM with the greater diversity of shifts that exist under shorter-duration planning intervals, we used both 30- and 15-minute planning intervals. We also selected operating days of 12, 16, and 20 hours, since we

hypothesized that DISSM would perform better than BJSSM with longer operating days (and consequently more alternative shifts).

Table 3 lists the 11 STCs of TPS2, all of which exhibit EBWO because of the extensive shift-length and break timing flexibility present in these problems. STCs 1 through 6 contain only a single shift type. STCs 7 and 8 each contain two types of shifts. Shifts of one type have an hourly cost of 85% of the other, a situation similar to those occurring in service organizations where the relevant costs of part-time and full-time employees differ. To ensure that shifts of both types are scheduled, the less expensive shifts are restricted to providing no more than about half the needed labor-hours. STCs 9 and 10 allow the scheduling of overtime, while STC 11 schedules reliefs in addition to meal breaks. Since the reliefs are 15 minutes long, STC 11 only uses 15-minute planning intervals. Overall, the number of alternate shifts in TPS2 ranges from 125 in STC 1 to 15,885 in STC 6.

---

Insert Figure 1 Here

---



---

Insert Table 3 Here

---



---

Insert Figure 2 Here

---

### **3.2. Model Solution Procedure**

We attempted to generate optimal schedules for both DISSM and BJSSM for each test problem. Schedule development involves three distinct steps for each model. First, a FORTRAN

program created the model. Next, the branch and bound IP procedure of SAS-OR (SAS Institute 1988) optimally solved the model. Finally, another FORTRAN program constructed explicit shifts from the implicit solutions. All problems were solved on an 486DX33-based personal computer. This computer system had eight megabytes of expanded memory that was available to SAS-OR and approximately 25 megabytes of available hard disk storage.

In initial experimentation, we observed that BJSSM generated “optimal-but-unacceptable” schedules very frequently in STCs 7 through 10 of TPS2. We also observed that the difficulty seemed to be related to the restrictions defining acceptable break placements for the shift types in these STCs. This difficulty was largely overcome by defining separate break variables for each of the two shift types and by restricting the timing of these breaks separately. We note, however, that this action increased the A-matrix size of BJSSM by 36 to 64 percent.

#### **4. Results**

By all measures of model size—number of variables, number of A-matrix elements, and number of non-zero A-matrix elements—DISSM was smaller than BJSSM across the 3 STCs in TPS1. DISSM was also smaller than BJSSM, by nearly every measure, in all problem categories of TPS2. The only exception to this general pattern was A-matrix size in STCs 9 and 10 of TPS2 with 30-minute planning intervals.

Table 4 summarizes model performance (detailed results are given in Thompson 1991b). This table first identifies the number and type of problems that could not be solved by each model. Some problems were simply too large to solve on our computer system, either because the model would not fit into memory, or because the available hard disk storage was filled before the optimal solution was verified. Other entries in Table 4 indicate the problems for which the

models could not identify any integer solutions within the specified time limits (3 hours for DISSM and 6 hours for BJSSM), one or more integer solutions were identified but the best of these solutions was not verified as being optimal within the specified time limits, and BJSSM generated an "optimal but unacceptable" solution. Overall, DISSM and BJSSM respectively failed to solve 16 and 109 of the 588 test problems. More problematic is that of the 458 TPS2 problems that BJSSM "solved," 33 schedules failed to satisfy the break-timing restrictions. Thus, while DISSM yielded acceptable solutions in over 97% of the 567 TPS2 test problems, BJSSM did so in fewer than 75% of these problems.

Table 4 also reports DISSM's and BJSSM's schedule generation times. For the 471 problems solved by both models, BJSSM averaged 12.55 and DISSM 4.58 minutes per schedule. DISSM averaged 57.13 minutes per schedule for the 100 problems that it alone solved. BJSSM took 131.48 minutes to solve the single problem that it alone solved. On a straight comparison, DISSM's average schedule generation time is 36.5% of that required by BJSSM, for those problems where both models yielded solutions (including BJSSM's "optimal" but "unacceptable" solutions).

We performed a paired comparison t-test on the difference in schedule generation times between DISSM and BJSSM for those problems where both models yielded integer solutions. These tests indicated that the difference in schedule generation times was not significant at the 0.1 level for TPS1, but that the difference in schedule generation times was significant at the 0.0001 level for TPS2 (the sample size was 14 problems in TPS1 and 457 problems in TPS2).

Thompson (1991b) provides a comparison of the time to generate, solve, and interpret DISSM and BJSSM by problem category of TPS1 and TPS2. DISSM's mean schedule generation times were lower than those of BJSSM in 60 of TPS2's 63 problem categories and in

two of TPS1's three categories. In general, DISSM's schedule generation time advantage increased with longer operating days and with shorter planning intervals.

---

Insert Table 4 Here

---

## 5. Discussion

On the diverse set of test problems exhibiting extensive shift length and break timing flexibility (and, consequently, EBWO) described in §3, DISSM had three advantages over BJSSM: it solved faster, on average; it solved larger problems because it more compactly represents greater scheduling flexibility; and it did not generate the "optimal-but-unacceptable" schedules that BJSSM regularly did. Consider each advantage.

First, for those problems where both models yielded schedules, DISSM generated schedules in 36.5% of the time required by BJSSM, on average, a difference significant at the 0.0001 level. In absolute terms, DISSM's mean schedule generation time for these problems was under five minutes on a 486DX33-based personal computer.

Considering that the largest of these problems had 10,237 unique shifts, and that the largest problems previously solved optimally had 970 shifts (Bechtold and Jacobs 1990), DISSM's solution times must be viewed very favorably.

Second, consider the largest problems solved. The largest problems solved by BJSSM had 10,237 alternate shifts, while DISSM solved problems with 15,885 alternate shifts (55% more). Also, while BJSSM only solved 24 of the 99 TPS2 test problems having a 15-minute planning interval and a 20-hour operating day, DISSM solved all but 12. These results lead one

to expect that DISSM will enable the solution of larger, more flexible problems than BJSSM, whatever the capabilities of a computer system.

Finally, consider the impact of the 33 "optimal-but- unacceptable" schedules generated by BJSSM. A dilemma faces a manager whenever BJSSM generates such a schedule. Restricting the extent of shift length or break-timing flexibility BJSSM uses ensures that it develops only acceptable schedules, but reducing flexibility will most likely result in higher cost schedules (see, for example, Bechtold and Jacobs 1990, Mabert and Showalter 1990). On the other hand, if a manager modifies the initial schedule heuristically to make it acceptable, it is likely that the resultant schedule will also no longer be optimal. Importantly, DISSM does not generate unacceptable schedules, even when EBWO exists (see Thompson 1991b for the proof of this). We have presented an integer programming model for shift scheduling that extends the earlier work of Moondra (1976) and Bechtold and Jacobs (1990) by implicitly representing shifts and by implicitly matching breaks to the implicitly represented shifts. This model, DISSM, solved all but 17 problems from a diverse set of 588 test problems, the largest of which had 15,885 alternate shifts. Compared to the model of Bechtold and Jacobs (1990), DISSM had significantly lower solution times, solved problems with 55% more alternative shifts, and only generated optimal solutions that satisfy operative break-timing restrictions. We hope that DISSM's superior performance, and the general superiority of implicit models (Bechtold and Jacobs 1990, 1991; Thompson 1990, 1992), serve to inspire researchers to develop implicit models for other scheduling environments.

Table 1. Information on the Relevant Shift Scheduling Literature.

Reference	Planning Period Duration (minutes)	Number of Alternate Shifts	Solution Procedure <sup>a</sup>
Bailey and Field (1985)	30	72	LP
Bechtold and Jacobs (1990)	30	970	IP
Bechtold and Jacobs (1991)	30	444	IP
Buffa et. al. (1976)	30	500	H
Henderson and Berry (1976)	15	100	LP
Henderson and Berry (1977)	15	100	LP <sup>b</sup>
Keith (1979)	15	NA	LP
McGinnis et al. (1978)	60	504	H
Moondra (1976)	60	26	LP
Morris and Showalter (1983) <sup>d</sup>	60	24	LP
Segal (1974)	30	300	LP
Thompson (1990)	15	6,588	LP
Thompson (1992)	60	12	IP

**NOTES:**

<sup>a</sup> LP = linear-programming based heuristic, IP = integer programming (yields an optimal solution), H = heuristic (but not LP-based).

<sup>b</sup> Henderson and Berry (1977) developed an LP-based optimal branch and bound procedure.

<sup>c</sup> Not given.

<sup>d</sup> Although Morris and Showalter's (1983) prime focus was on the tour scheduling problem, the reported data is for the shift scheduling portion of their investigation.

Table 2. Constructing Explicit Shifts from DISSM's Implicit Solution to the Sample Problem Using a First-In-First-Out Matching.

Shift Number	Start Variable (Starting Period)	Finish Variable (Finishing Period)	Meal-Break Variable (Break Starting Period)
1	$s_{1,1}$ (1)	$f_{1,5}$ (6)	$m_{1,3}$ (3)
2	$s_{1,2}$ (2)	$f_{1,5}$ (6)	$m_{1,4}$ (4)
3	$s_{1,2}$ (2)	$f_{1,6}$ (6)	$m_{1,4}$ (4)
4	$s_{1,3}$ (3)	$f_{1,7}$ (7)	$m_{1,5}$ (5)
5	$s_{1,3}$ (3)	$f_{1,8}$ (8)	$m_{1,6}$ (6)
6	$s_{1,3}$ (3)	$f_{1,8}$ (8)	$m_{1,6}$ (6)
7	$s_{1,4}$ (4)	$f_{1,8}$ (8)	$m_{1,6}$ (6)
8	$s_{1,4}$ (4)	$f_{1,9}$ (9)	$m_{1,7}$ (7)

Table 3. Enumeration of Alternate Shifts in the Scheduling Environments of the test Problem Sets\*

Problem Set	STC	SWH	MBL	MBTR	RBTR	RSC	PIL: 30			15			
							ODL: 12	16	20	12	16	20	
1 <sup>a</sup>	1	6-8	1.0	2-4.75	NA	1.00	NA	NA	NA	NA	2543	NA	
	2	6-8	1.0	2-4.75	NA	1.00	NA	NA	NA	NA	2543	NA	
		5-5.75	0.5	2-3.75	NA	1.00	NA	NA	NA	NA	970	NA	
							Total				3513		
	3	6-8	1.0	2-4.75	NA	1.00	NA	NA	NA	NA	2543	NA	
		5-5.75	0.5	2-3.75	NA	1.00	NA	NA	NA	NA	970	NA	
3-4.75		0	NA	NA	1.00	NA	NA	NA	NA	364	NA		
						Total				3877			
2	1	6-8	1.0	3-5	NA	1.00	125	245	365	705	1425	2145	
	2	6-8	1.0	2-6	NA	1.00	305	585	865	1929	3801	5673	
	3	6-8	1.0	1-7	NA	1.00	485	925	1365	3153	6177	9201	
	4	6-10	1.0	3-5	NA	1.00	175	375	575	1053	2349	3645	
	5	6-10	1.0	2-6	NA	1.00	427	915	1403	2821	6293	9765	
	6	6-10	1.0	1-7	NA	1.00	679	1455	2231	4589	10237	15885	
	7	6-8	1.0	3-5	NA	1.00	125	245	365	705	1425	2145	
		4-8	1.0	2-6	NA	0.85 <sup>b</sup>	435	795	1155	2805	5253	7701	
							Total	560	1040	1520	3510	6678	9846
	8	6-8	1.0	2-6	NA	1.00	305	585	865	1929	3801	5673	
		4-8	1.0	2-6	NA	0.85 <sup>b</sup>	435	795	1155	2805	5253	7701	
						Total	740	1380	2020	4734	9054	13374	
9	6-8	1.0	3-5	NA	1.00	125	245	365	705	1425	2145		
	8-10	1.0	3-5	NA	1.50 <sup>c</sup>	50	130	210	348	924	1500		
						Total	175	375	575	1053	2349	3645	
10	6-8	1.0	2-6	NA	1.00	305	585	865	1929	3801	5673		
	8-10	1.0	2-6	NA	1.50 <sup>c</sup>	122	330	538	892	2492	4092		
						Total	427	915	1403	2821	6293	9765	
11	6.5-8	1.0	3.25-5	1.5-3.25	1.00	NA	NA	NA	2982	6342	9702		

\* STC – shift-type category; SWH – the range of working hours in acceptable shifts; MBL – duration of the meal break, in hours; MBTR – meal-break timing restrictions—the minimum and maximum number of work hours (ignoring reliefs) before or after the meal break; RBTR – relief-break timing restrictions—the minimum and maximum number of work hours before or after a 15-minute relief break (only STC 11 of TPS2 schedules relief breaks); RSC – the relative shift cost, per working period; PIL = planning interval duration, in minutes; and ODL = number of operating hours.

<sup>a</sup> Test problem set #1 uses only a 15-hour operating day and 15-minute planning periods.

<sup>b</sup> The less expensive shifts are limited to providing no more than about half of the total employee requirements.

<sup>c</sup> Overtime premium paid only on the portion greater than 8 hours.

Table 4. Schedule Generation Summary

Problem Category	Test Problem Set #1		Test Problem Set #2	
	BJSSM	DISSM	BJSSM	DISSM
Number of problems too large to solve with the available software and hardware	7	0	90	0
Number of problems that did not yield an integer solution within the specified time limits	0	1	19	16
Number of problems for which an integer solution was obtained that was not verified as being optimal within the specified time limits	0	0	0	18
Number of problems for which the "optimal" solution did not satisfy the break-timing restrictions	0	0	33	0
Number of shifts in the largest problems optimally solved	3513	3877	10237	15885
Mean schedule generation time for the test problems solved by both models <sup>a,b</sup>	27.12 minutes <sup>c</sup>	7.08 minutes <sup>c</sup>	12.11 minutes <sup>d</sup>	4.51 minutes <sup>d</sup>
Mean schedule generation time for the test problems solved only by BJSSM <sup>a,e</sup>	NA	NA	131.48 minutes	NA
Mean schedule generation time for the test problems solved only by DISSM <sup>a,f</sup>	NA	17.80 minutes	NA	59.64 minutes

<sup>a</sup> Times are in minutes on an 486DX33-based personal computer, and include the times required to develop the model, optimally solve it, and interpret the solution.

<sup>b</sup> Both models generated schedules for 14 problems in TPS1 and for 457 problems in TPS2 (including 32 TPS2 problems for which the schedules generated by BJSSM did not satisfy the operative break-timing restrictions).

<sup>c</sup> Differences are not significant at the 0.1 level.

<sup>d</sup> Differences are significant at the 0.0001 level.

<sup>e</sup> BJSSM was the only model to solve one TPS2 problem. However, its "optimal" solution to this problem did not satisfy the operative break-timing restrictions.

<sup>f</sup> DISSM was the only model to solve 6 TPS1 and 94 TPS2 problems. Neither model yielded solutions for 1 TPS1 problem and 15 TPS2 problems.

Figure 1 . The Seven Employee Requirements Patterns in Test Problem Set #1.

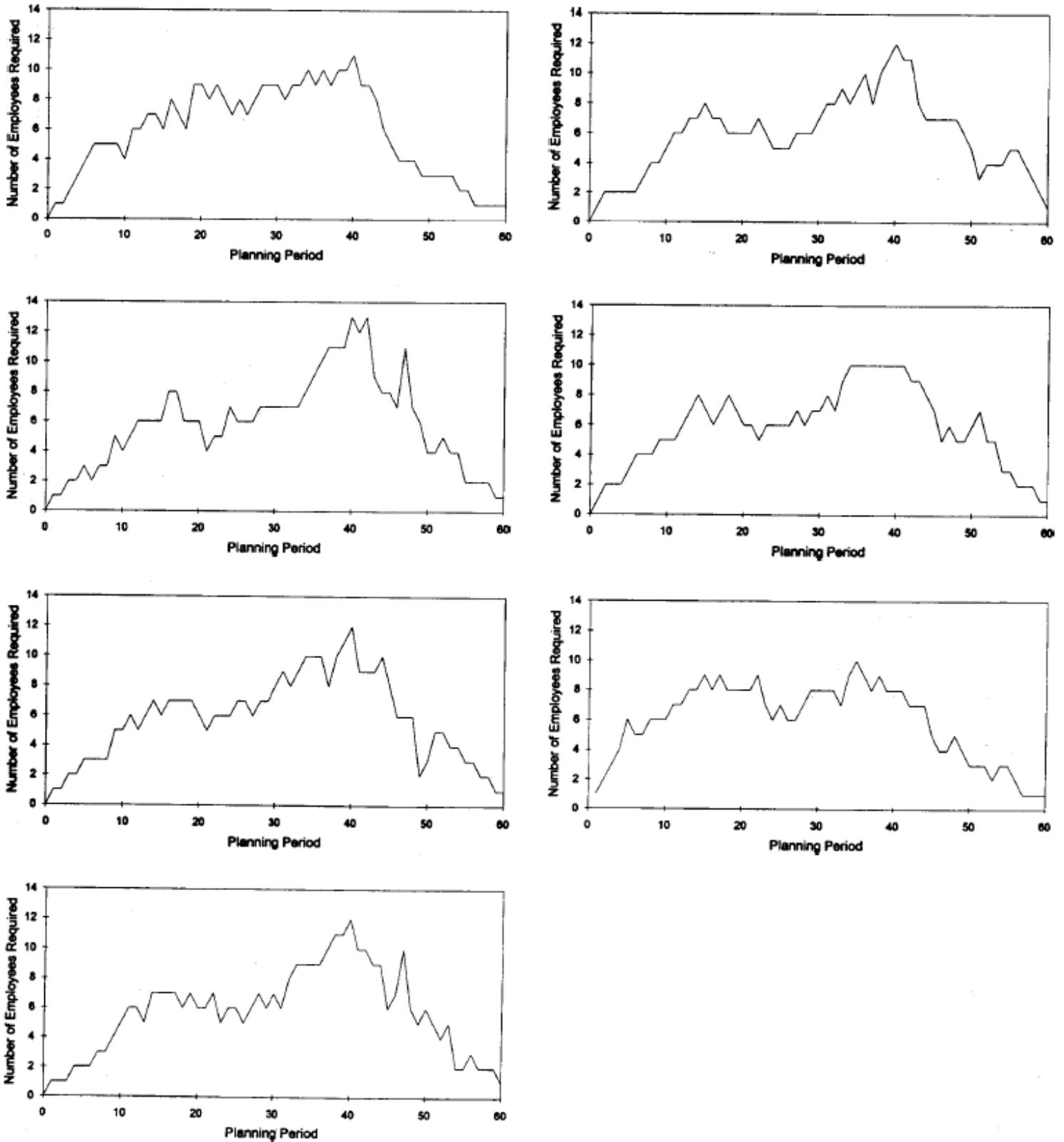
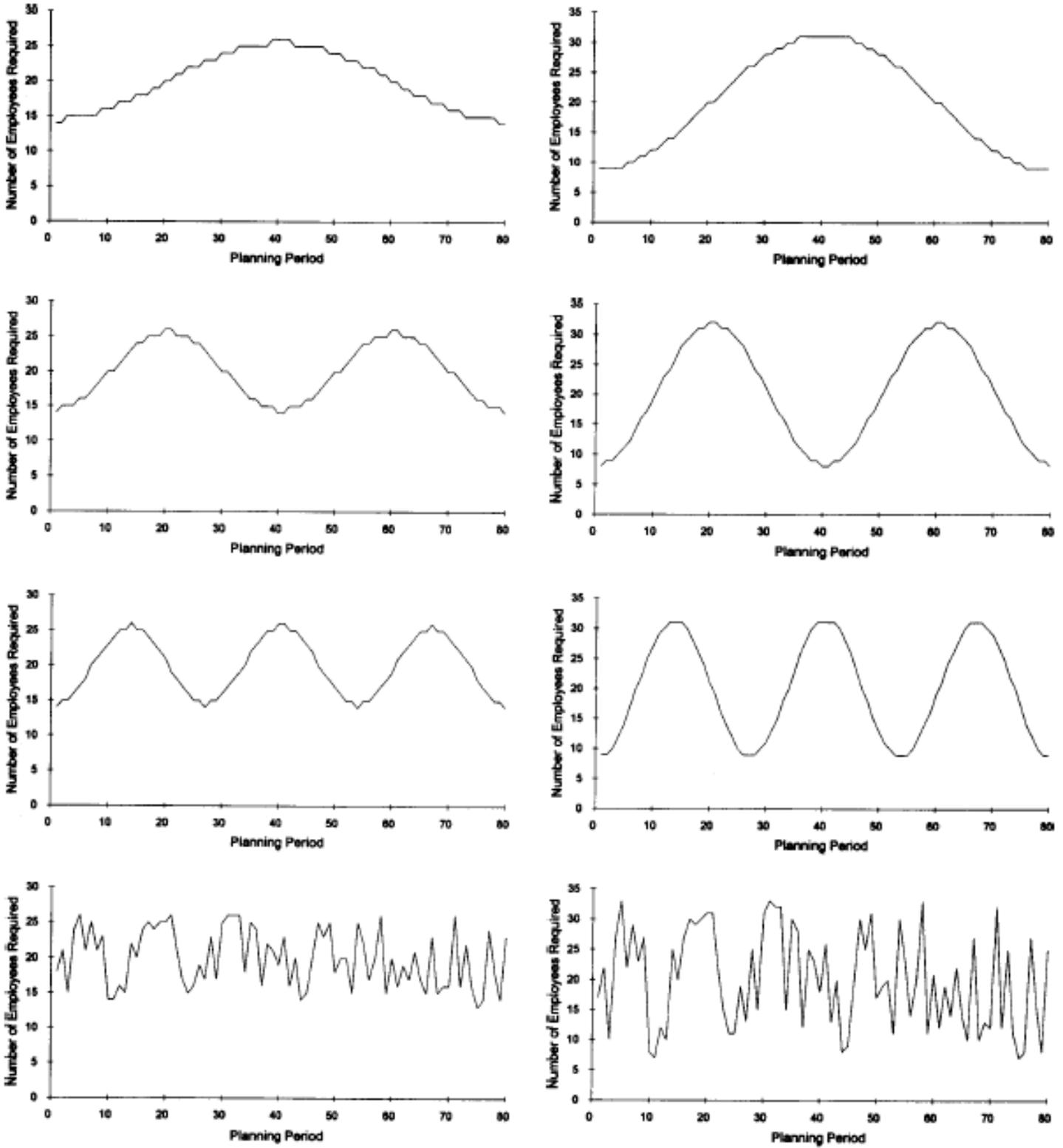


Figure 2. Eight of the Nine Employee Requirements Patterns with a 20-Hour Operating Day and 15-Minute Planning Periods in Test Problem Set #2



## References

- Bailey, J. and J. Field, "Personnel scheduling with flexshift models," *J. Oper. Management*, 5, 3 (1985), 327-338.
- Bechtold, S. E. and L. W. Jacobs, "Implicit optimal modeling of flexible break assignments in labor staffing decisions for service operations," *Management Sci.*, 36, 11 (1990), 1339-1351.
- , "Improvement of labor utilization in shift scheduling for services with implicit optimal modeling," *International J. of Oper. and Production Management*, 11, 2 (1991), 54-69.
- Buffa, E. S., M. J. Cosgrove, and B. J. Luce, "An integrated work shift scheduling system," *Decision Sciences*, 7 (1976), 620-630.
- Dantzig, G. B., "A comment on Edie's 'Traffic delays at toll booths'," *Oper. Res.*, 2, 3 (1954), 339-341.
- Glover, F., C. McMillan, and R. Glover, "A heuristic programming approach to the employee scheduling problem and some thoughts on 'managerial robots'," *J. of Oper. Management*, 4, 2 (1984), 113-128.
- Henderson, W. B. and W. L. Berry, "Heuristic methods for telephone operator shift scheduling: an experimental analysis," *Management Sci.*, 22, 12 (1976), 1372-1380.
- , "Determining optimal shift schedules for telephone traffic exchange operators," *Decision Sciences*, 8, 2 (1977), 239-255.
- Keith, E. G., "Operator scheduling," *AIIE Transactions*, 11, 1 (1979), 37-41.
- Luce, B. J., Jr, "A shift scheduling algorithm," presented at the 44th National Meeting of ORSA, November 1973.
- McGinnis, L. F., W. D. Culver, and R. H. Deane, "One- and two- phase heuristics for workforce scheduling," *Computers and Industrial Engineering*, 2, 1 (1978), 7-15.

- Mabert, V. A. and M. J. Showalter, "Measuring the impact of part-time workers in service organization," *J. of Oper. Management*, 9, 2 (1990), 209-229.
- Moondra, S. L., "An L. P. model for work force scheduling for banks," *J. Bank Res.*, 7, 4 (1976), 299-301.
- Morris, J. G. and M. J. Showalter, "Simple approaches to shift, days- off and tour scheduling problems," *Management Sci.*, 29, 8 (1983), 942-950.
- SAS Institute, *SAS Users Guide: OR, Version 6.03*. SAS Institute, Inc., Cary, NC, 1988.
- Segal, M., "The operator-scheduling problem: a network-flow approach," *Oper. Res.*, 22, 4 (1974), 808-823.
- Thompson, G. M., "A comparison of techniques for scheduling nonhomogeneous employees in a service environment subject to noncyclical demand," Unpublished Ph.D. Dissertation, The Florida State University, 1988.
- , "Shift scheduling when employees have limited availability: an L. P. approach," *J. of Oper. Management*, 9, 3 (1990), 352-370.
- , "The duration of planning intervals, customer service, and labor shift scheduling," Working Paper 91-1a, David Eccles School of Business, University of Utah, Salt Lake City, 1991a.
- , "Implicit modeling of shifts and breaks in labor scheduling," Working Paper 91-1b, David Eccles School of Business, University of Utah, Salt Lake City, 1991b.
- , "Improving the utilization of front-line service delivery system personnel," *Decision Sciences*, 23, 5 (1992), 1072-1098.