George K. Chacko, ed., HEALTH HANDBOOK

33

NURSE UTILIZATION ALGORITHM*

Holmes E. MILLER, Ph.D., Frank A. PIERCE, William P. PIERSKALLA, Ph.D., and Gustave J. RATH, Ph.D.

Dr. Holmes E. Miller is Project Director, the Operations Research Group of the Union Carbide Corporation, Bound Brook, New Jersey; and Mr. Frank A. Pierce is Manager for The Medicus Systems Corporation, 1725 West Harrison Street, Chicago, Illinois.

Dr. Gustave J. Rath is Professor, Industrial Engineering/Management Sciences, Educational Organizational Development, Northwestern University, Evanston, Illinois. He has been engaged in organizational development, evaluation, man-machine systems design, the application of systems analysis and program planning and budgeting to education, police, transportation, health, churches, and industry, and research on research. He is a Fellow of American Psychological Association and is the author of *Punch Card Data Processing*.

Dr. William P. Pierskalla is Executive Director of the Leonard Davis Institute of Health Economics and Professor of Decision Sciences of the Wharton School, University of Pennsylvania, Philadelphia, PA. He is Editor of *Operations Research*, and a member of the Medical Advisory Committee of the Chicago Regional Blood Program.

OVERVIEW

One of the areas of successful application of Operations Research in Health Care Delivery is the scheduling of nurses. The nurse scheduling process may be viewed as one of generating a configuration of nurse schedules that specify the number and identities of the nurses working each day of the scheduling period. The authors discuss the cyclic coordinate descent algorithm of computer-based nurse scheduling which has been successfully implemented in a number of hospitals in the United States and Canada. The mathematical programming model underlying the algorithm finds first a feasible solution and then an optimum solution. The special feature of the model is that there are two types of constraints, constraints which may not be violated except under rare circumstances and constraints involving nurse preferences which may be violated occasionally but only if the cost of violation is greater than the staffing level cost. The model assesses penalties for violations of nurse preferences and of staffing levels. The two penalties can be weighted differentially; trial-and-error procedure has been developed to make the costs approximately equal. In the solution procedure, an initial configuration of the nurse schedules is

---

*Adapted with kind permission from articles published in *Operations Research* and Proceedings of *Nurse Staffing*.

made, one for each nurse. Fixing the schedules of all nurses but one, say nurse i, *it searches* π_i, *the set of feasible schedule patterns for nurse* i *until no lower cost schedule can be found. If we view the feasibility region as the cartesian product of the feasibility regions* $\pi_1, \pi_2, \ldots, \pi_I$, *the algorithm is a simple cyclic coordinate descent algorithm along the coordinate directions* π_i. *The empirical results of application of the algorithm have been most favorable, as shown by the detailed discussion of one implementation, i.e., a major Medical Center, which is a 40-unit, 800-bed hospital with approximately 900 full-time and part-time nursing personnel. The algorithm chose the lowest cost schedule nearly 44% of the time, and chose a schedule that was in the 90th percentile or better, nearly 88% of the time.*

1. Successful Multi-Site Implementation

Computer-based nurse scheduling systems have been successfully implemented in a number of hospitals in the United States and Canada. The theoretical basis is mathematical programming; and the computer basis is the *cyclic coordinate descent algorithm.* Among the locations where the algorithm has been implemented are:

(1) Mount Zion Medical Center, San Francisco, California
(2) Pacific Medical Center, San Francisco, California
(3) Stanford University Medical Center, Stanford, California
(4) Kingston General Hospital, Kingston, Ontario
(5) Rush Presbyterian St. Luke Medical Center, Chicago, Illinois

2. Alternative Objectives in Nurse Scheduling

The nurse scheduling process may be viewed as one of generating a configuration of nurse schedules that specify the number and identities of the nurses working each day of the scheduling period. By specifying nurse identities, a pattern of scheduled days off and on is created for the individual nurses. These patterns along with the hospital staffing requirements define the nurse scheduling problem: How to generate a configuration of nurse schedules that satisfy the hospital staffing requirements while simultaneously satisfying the individual nurse's preferences for various schedule pattern characteristics.

2.1. Cost, Effectiveness, Decision Levels

A number of mathematical programming applications to nurse staffing have appeared in the literature beginning with Wolfe and Young[1],[2] who constructed mathematical models which *minimized the cost* of assigning nurses to various classes to do various tasks. Liebman[3],[4] also proceeded from a task orientation by assigning nursing tasks in a manner which *maximized the effectiveness* of nurses performing tasks on various patients. Warner and Prawda[5],[6] sought to minimize a "*shortage cost*" of nursing care services for a period of three to four days subject to total personnel capacity, integral assignment and other relevant constraints. Abernathy, Baloff, Hershey and Wandel[7],[8] considered three different *decision levels* impinging on the nurse staffing problem, and formulated an interactive model where the outputs of one level (e.g. staffing policies) are the inputs of another.

2.2. *Cyclical and Noncyclical Scheduling*

Much of the work relating to nurse scheduling has concerned cyclical scheduling (Morrish and O'Conner,[9] Price,[10] Howell,[11] and Maier-Rothe and Wolfe,[12] e.g., where each nurse works a cycle of n weeks, where n is the length of the scheduling period. Cyclical schedules are easily generated but are characterized by excessive rigidity vis-à-vis variations in the supply of and demand for nursing services. Two *noncyclical scheduling* papers of note have been by Rothstein[13] and Warner[14]. Rothstein's application was to hospital housekeeping operations. He sought to maximize the number of day off pairs (e.g., Monday—Tuesday) subject to constraints requiring two days off each week and integral assignments. Warner presented a two phase algorithm to solve the nurse scheduling problem. Phase I is involved with finding a feasible solution to various staffing constraints while Phase II seeks to improve the Phase I solution by maximizing individual preferences for various schedule patterns while maintaining the Phase I solution.

3. The Mathematical Programming Model

The mathematical programming model developed here schedules days on and days off for all nurses on a given unit or ward for a given shift for a two, four, six, or eight week scheduling horizon subject to certain hospital policy and employee constraints. Because of the large number of constraints, it is possible that no feasible solutions to the nurse scheduling problem would exist if all the *constraints were binding.* For this reason we divide the constraints into two classes: Feasibility set constraints, which define the sets of feasible nurse schedules, and *nonbinding constraints,* whose violation incurs a penalty cost which appears in the objective function. *Each hospital has the discretion to define which constraints go into each class.*

3.1. *Constraints: The Feasibility Set*

Because of the possibility of special requests by nurses, no constraints are binding in the sense that they hold under all circumstances except those constraints emanating from the special requests. We do, however, distinguish between constraints we would *like to hold* in the absence of special requests, and those which we shall always allow to be violated while incurring a penalty cost.

The former constraints define what we call the *feasibility set* π_i, i.e., $\pi_i =$ the set of feasible schedule patterns for nurse i.

In the *absence* of special requests, this set might include all schedules satisfying:

(1) A nurse works ten days every pay period (i.e., 14 day scheduling period),

(2) No work stretches (i.e., stretches of consecutive days on) are allowed in excess of σ days (e.g., $\sigma = 7$)*, and

(3) No work stretches of τ or fewer days are allowed (e.g., $\tau = 1$)[1]

[1] These are calculated within a scheduling period and also at the interface of a scheduling period with past and future periods.

Hence one schedule in a π_i satisfying these might be (with $\sigma = 7$, $\tau = 1$)

1 1 1 1 1 1 1 0 0 1 1 1 0 0

Now suppose a nurse has special requests. For example, suppose the nurse requests the schedule: 1 1 1 1 1 1 1 1 0 1 0 0 0 B, where the B indicates a birthday. In this case *all* of the above constraints would be violated and π_i would consist of only the schedule just given. Thus in the general case π_i is the set of schedules which:

(1) Satisfies a nurse's special requests, and

(2) Satisfies as many of the constraints we *would like* to see binding as possible, given the nurse's special requests.

The constraints we would like to hold are a function of the hospital in which the model is applied. Thus, for example, we could as easily specify five out of seven days as ten out of fourteen or specify additional constraints we would like to see satisfied such as one weekend off each pay period.

3.2. *Constraints: Nonbinding*

Each schedule pattern $x^i \varepsilon \pi_i$ may violate a number of *nonbinding schedule pattern constraints* while incurring a penalty cost. Define

N_i = the index set of the nonbinding schedule pattern constraints for nurse i.

For example, if the hospital in which the model was being implemented deemed them as nonbinding, the following constraints might define N_i:

No work stretches longer than S_i days (where $S_i \leqq \sigma$);*
No work stretches shorter than T_i days (where $T_i \geqq \tau$);*
No day on, day off, day on patterns (1 0 1 pattern);*
No more than κ consecutive 1 0 1 patterns;*
Q_i weekends off every scheduling period (4 or 6 weeks);*
No more than W_i consecutive weekends working each scheduling period;*
No patterns containing four consecutive days off;*
No patterns containing split weekends on (i.e., a Saturday on—Sunday off—pattern, or vice versa).

In addition to nonbinding schedule pattern constraints, we also have *nonbinding staffing level constraints.* Define: d_k = the desired staffing level for day k; and m_k = the minimum staffing level for day k. Then we have:

(a) the number of nurses scheduled to work on day k is greater than or equal to m_k and

(b) the number of nurses scheduled to work on day k is equal to d_k.

3.3. *Objective Function*

As was mentioned, the objective function is composed of the sum of two classes of

* These are calculated within a scheduling period and also at the interface of a scheduling period with past and future scheduling periods.

penalty costs; *penalty costs* due to violation of *nonbinding staffing* level constraints and *penalty costs* due to violation of *nonbinding schedule* pattern constraints.

3.4. *Staffing Level Costs*

Define the *group* to be scheduled as the set of all the nurses in the unit who are to be scheduled by one application of the solution algorithm. Further define a *subgroup* as a subset of the group specified by the hospital. For example, the group to be scheduled may be all those nurses assigned to a nursing unit and the subgroups may be Registered Nurses (RNs) and Licensed Practical Nurses (LPNs) and Nursing Aides. Alternatively, the group may be defined as all RNs and a subgroup might be those capable of performing as head nurses.

Then, for each day $k = 1, \ldots, 14$ (where there are I nurses), the group staffing level costs are given by:

$$f_k\left(\sum_{i=1}^{I} x_k^i\right), \quad \text{where } x^i = (x_1^i, \ldots, x_{14}^i).$$

For example, this function might appear as in Fig. 1. Now define

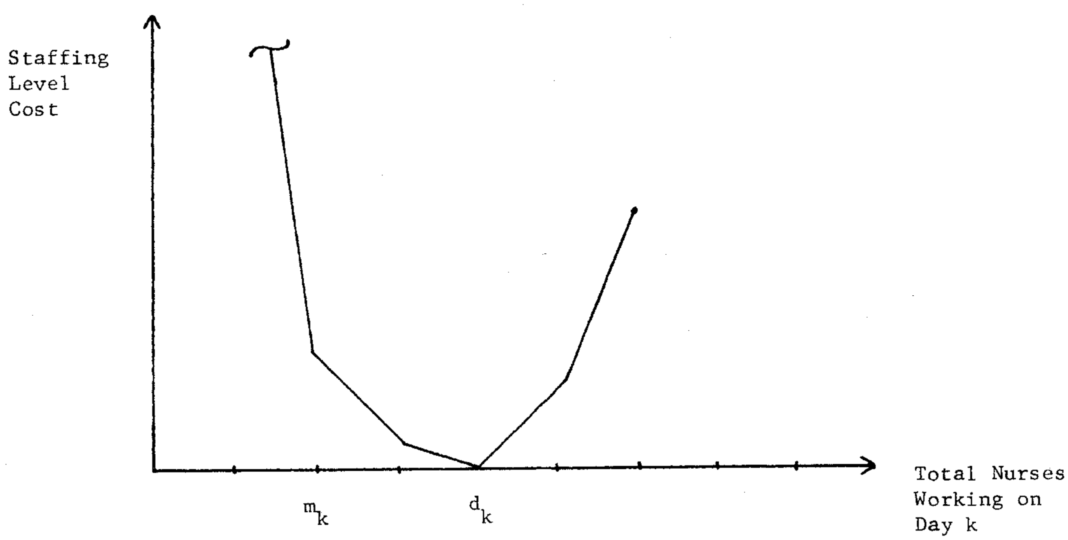

Fig. 1 Group Staffing Level Costs

B_j = the index set of nursing subgroups j.

J = the index set of all subgroups.

If m_k^j and d_k^j are the minimum and desired number of nurses required on day k for subgroup j, we define the staffing cost for violating those constraints on day k for subgroup j as:

$$h_{jk}\left(\sum_{i \in B_j} x_k^i\right)$$

where $h_{jk}(.)$ is defined similarly as $f_k(.)$.

Then the total staffing level costs for all 14 days of the pay period are:

$$\sum_{k=1}^{14} f_k\left(\sum_{i=1}^{I} x_k^i\right) + \sum_{k=1, j\varepsilon J}^{14} \sum h_{jk}\left(\sum_{i\varepsilon Bj} x_k^i\right).$$

3.5. Schedule Pattern Costs

For each nurse $i = 1, \ldots, I$ the schedule pattern costs for a particular pattern x^i measure:

(1) The costs inherent in that pattern in relation to which constraints in N_i are violated;

(2) How nurse i perceives these costs in the light of nurse i's schedule preferences;

(3) How this cost is weighed in the light of nurse i's schedule history.

For example, for (1) the pattern

1 1 1 1 1 0 0 1 1 1 0 0 1 1

may incur a cost for a nurse whose minimum desired work stretch is 4 days. This is a cost inherent in the pattern. Considering (2), we next ask how nurse i perceives violations of the minimum desired stretch constraint, i.e., how severely are violations of this nonbinding constraint viewed vis-à-vis others in N_i. Finally (3) gives us some indication of how we should weigh this revised schedule pattern cost in the light of the schedules nurse i has received in the past. Intuitively, if nurse i has been receiving poor schedules, we would want the cost of a given schedule to be relatively higher than the costs for schedules of other nurses in order to cause a good schedule to be accepted when the solution algorithm is applied and vice versa. Thus, we define:

$g_{in}(x^i)$ = the cost of violating nonbinding constraint $n \varepsilon N_i$ of schedule x^i.

α_{in} = the "weight" nurse i gives a violation of nonbinding constraint $n \varepsilon N_i$, which we shall call the *aversion coefficient.*

A_i = the *aversion index* of nurse i; i.e., a measure of how good or bad nurse i's schedules have been historically vis-à-vis nurse i's preferences.

Then the total schedule pattern cost to nurse i for a schedule pattern x^i is:

$$A_i \sum_{n\varepsilon N_i} \alpha_{in} g_{in}(x^i),$$

and the sum of these costs for all nurses $i = 1, \ldots, I$ is the total schedule pattern cost.

3.6. Problem Formulation

Let $\lambda \varepsilon (0, 1)$ be a parameter that weighs staffing level and schedule pattern costs. It is chosen such that the weighted staffing and schedule pattern costs are of approximately equal magnitude. Experience has shown a trial-and-error procedure to be effective in arriving at satisfactory values of λ.

Given λ, the problem is to find $x^1, x^2, \ldots, x^I$ which minimize:

$$\lambda\left[\sum_{k=1}^{14} f_k\left(\sum_{i=1}^{I} x_k^i\right)+\sum_{k=1}^{14}\sum_{j\varepsilon J} h_{jk}\left(\sum_{i\varepsilon B_j} x_k^i\right)\right]$$

$$+(1-\lambda)\sum_{i=1}^{I} A_i \sum_{n\varepsilon N_i} \alpha_{in} g_{in}(x^i)$$

subject to $x^i \varepsilon \pi_i$, $i = 1, \ldots, I$.

4. The Solution Procedure: Cyclic Descent Algorithm

The solution procedure used is a near-optimal algorithm. It starts with an initial configuration of nurse schedules, one for each nurse. Fixing the schedules of all nurses but one, say nurse i, it searches π_i. The lowest present cost and best schedule configuration are updated if, when searching π_i, a schedule is found which results in a lower schedule configuration cost than the lowest cost to date. When all the schedules in π_i have been tested, either 1) a lower cost configuration has been found, or 2) no lower cost configuration has been found. The process cycles among the I nurses and terminates when no lower cost configuration has been found in I consecutive tests.

Each set π_i will always contain at least one feasible schedule, due to the manner in which the feasibility sets are constructed. To arrive at an initial solution one may select one schedule from each π_i in an appropriate manner (e.g., select the schedule with the lowest dissatisfaction cost).

If we view the feasibility region as the cartesian product of the feasibility regions π_1, $\pi_2, \ldots, \pi_I$, the algorithm is simply a *cyclic coordinate descent algorithm* along the coordinate directions π_i. Each π_i contains all feasible schedules for nurse i. When 4 days are given off every 14 day pay period, π_i contains at most $\binom{14}{4} = 1001$ schedules. This number is reduced considerably when previous schedules, special requests, and other feasibility set constraints are considered. The convergence of the algorithm is assured since the cartesian product contains a finite number of points, namely, $\prod_{i=1}^{I} ||\pi_i||$, where $||\pi_i||$ is the number of schedules in the set π_i.

The following steps describe the algorithm in detail:

1. Determine the set of feasible schedules for each employee's, π_i. Let $||\pi_i||$ denote the number of schedules in π_i.

2. Calculate the schedule pattern costs for each schedule $x^i \varepsilon \pi_i$, for $i = 1, \ldots, I$.

3. Choose an initial schedule mix (i.e., a schedule for each nurse $i = 1, \ldots, I$) and let BEST = its cost (e.g., choose the lowest cost schedule from each π_i).

4. Let $i = 1$, $K = ||\pi_i||$, $k = 1$ and CYCLE = 0.

5. Try the kth candidate schedule, x^{ik}, in the schedule mix by temporarily removing the present schedule for nurse i from the current schedule mix and inserting schedule x^{ik}. Let TEST = the cost of this new schedule mix.

6. If TEST < BEST go to Step 8.
7. Let $k = k+1$. If $k = K+1$ go to Step 9. Otherwise go to Step 5.
8. Let CYCLE = 0 and BEST = TEST. Insert x^{ik} in place of the current schedule for nurse i in the schedule mix. The schedule mix now contains the "best schedules found so far." Go to Step 7.
9. If CYCLE = I stop. Otherwise let $i = i+1$ (if $i > I$, let $i = 1$) and let $K = ||\pi_i||$, $k = 1$, and CYCLE = CYCLE + 1. Go to Step 5.

5. A Major Medical Center Nurse Schedules

The Medical Center is a 40 unit, 800 bed hospital with approximately 900 full-and part-time nursing personnel. The hospital had collected historical data regarding nurse schedule preferences and minimum and desired staffing levels. Since the same data base were used to develop both the hospital schedule and the computer schedule, it is possible to compare them directly.

5.1. Computer Generated Schedules

Fig .2 presents some schedules generated by an early version of the algorithm for four weeks of the six month trial period: October 22 to November 18. It should be noted that in 14 of the 28 days the actual staffing levels were identical with the desired staffing levels. The unit is understaffed by one nurse on two days and overstaffed by one nurse on twelve days.

Table 1 presents data relating to the per cent of the schedules in π_i having employee dissatisfaction costs greater than or equal to that of the chosen schedule.

Note that in all cases except one, the nurses were given a schedule better than 90 per cent or more of those in the feasible pattern set. Moreover, we see that in most instances the number of schedules in the set of feasible patterns, i.e., $||\pi_i||$, was well over 100; so there were many schedules to choose from.

We also note how the algorithm schedules equitably over time. In all cases except one, a nurse received the lowest cost schedule pattern in the feasible set during one of the two pay periods, and in that one instance the nurse received a schedule in the 99th percentile in each of the two periods. The effect of the *aversion index* is evident when we note the general pattern of nurses who receive their best schedules during the first two weeks receiving a slightly worse schedule in the second two weeks and vice versa.

More extensive results will now be given for the *entire six month* scheduling test. Fig. 3 presents a histogram of deviations from desired staffing levels.

On 90 per cent of the days, the deviation from the desired staffing level was either 0 or $\pm$ 1. Moreover, we do not include measures of under or overstaffing of the units in question. Hence, if a unit was understaffed for a pay period, we would expect a number of negative deviations. Similar results would hold for overstaffed units. In the light of this we see how well the algorithm works in meeting staffing criteria.

In Fig. 4 a histogram presents data taken over the six months relating to the percentile of a nurse's feasible schedule pattern set in which the schedule pattern chosen

Group 1 RN's	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S
1A	V	V	R	1	1	1	1	R	M	1	1	1	0	0	0	1	M	1	1	1	1	0	M	1	1	1	0	0
1B	1	1	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	0	1	1	1	1	M	1	1	0	0	0
1C	1	1	1	1	0	0	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	0
1D	1	0	0	1	1	0	0	1	1	1	1	1	1	1	0	1	1	1	1	0	0	1	1	1	0	1	1	1
1E	1	1	1	1	1	1	1	0	0	1	1	1	0	0	1	1	1	1	0	B	0	1	1	1	0	0	1	1
1F	1	1	1	0	1	1	1	1	1	1	1	0	0	0	1	1	1	1	0	0	0	1	1	0	1	1	1	1
1G	V	V	1	1	1	0	0	1	1	0	0	1	1	1	1	0	1	1	1	0	0	1	1	0	1	1	1	1
Group 2 LPN																												
2A	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	1	1	1	1	R	R	R	1	1	1	R	1	1
2B	1	1	0	B	1	1	1	1	0	1	1	1	0	0	1	1	1	0	1	1	1	1	1	1	1	0	0	0
2C	1	1	1	C	0	0	0	1	1	0	C	1	1	1	1	1	0	C	1	1	1	1	0	1	C	1	0	0
2D	0	1	1	1	1	1	1	V	V	1	1	0	0	0	1	1	1	1	0	0	0	1	1	1	0	1	1	1
2E	0	1	1	1	1	0	0	1	1	1	0	1	1	1	0	0	1	1	1	0	0	1	1	1	1	1	1	1
Total Desired	9	9	9	9	9	5	6	9	9	9	8	8	6	5	9	9	9	9	9	5	6	9	9	9	8	8	7	6
Total Actual	8	9	9	9	9	6	6	9	9	10	9	8	6	6	9	10	10	10	9	5	5	10	10	10	9	8	7	7

Legend:
1 = Day Scheduled On
0 = Day Scheduled Off
M = Day On For Meeting
V = Vacation Day off
R = Requested Day Off
B = Birthday Off
C = Day On For Class

Fig. 2 A Four Week Set of Nurse Schedules Generated by the Solution Algorithm

Table 1
Ranking of schedules chosen by computer algorithm as judged by employee dissatisfaction cost criteria

Nurse	$\|\|\pi_i\|\|$	*Pay Period* 1 *Percentile*	$\|\|\pi_i\|\|$	*Pay Period* 2 *Percentile*
1A	15	100	167	92
1B	331	100	233	93
1C	331	100	370	94
1D	302	99	128	99
1E	331	93	166	100
1F	390	94	331	100
1G	156	100	349	100
2A	235	80	1	100
2B	202	100	331	95
2C	163	100	182	92
2D	52	98	331	100
2E	390	100	390	100

Legend: $||\pi_i||$ = Number of Schedules in Feasible Schedule Set of Nurse i.
Percentile = Per cent of Schedules in Feasible Schedule Set with Employee Dissatisfaction Cost Greater than or Equal to Schedule Selected by the Solution Algorithm.

fell (where percentile is defined as in Table 1 and where $||\pi_i|| \geqq 10$ were the only sets considered).

Note that the algorithm chose the *lowest cost schedule* from a nurse's feasible schedule pattern set almost 44% *of the time* and the algorithm chose a schedule that was in the *90th percentile or better* of the feasible pattern set almost *88 per cent* of the time.

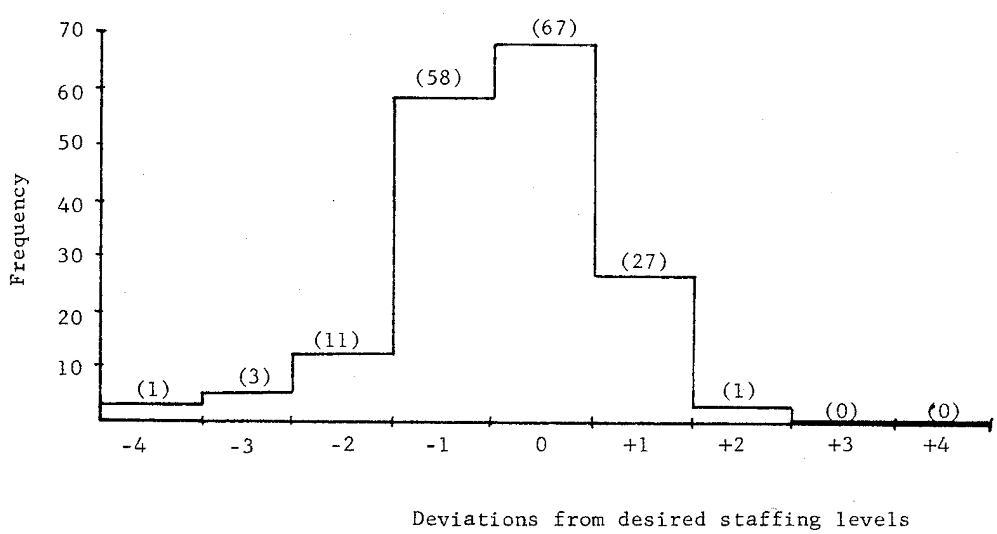

Fig. 3 Distribution of Deviations from Desired Staffing Levels by Schedules Chosen by the Algorithm During the Six Month Test Period

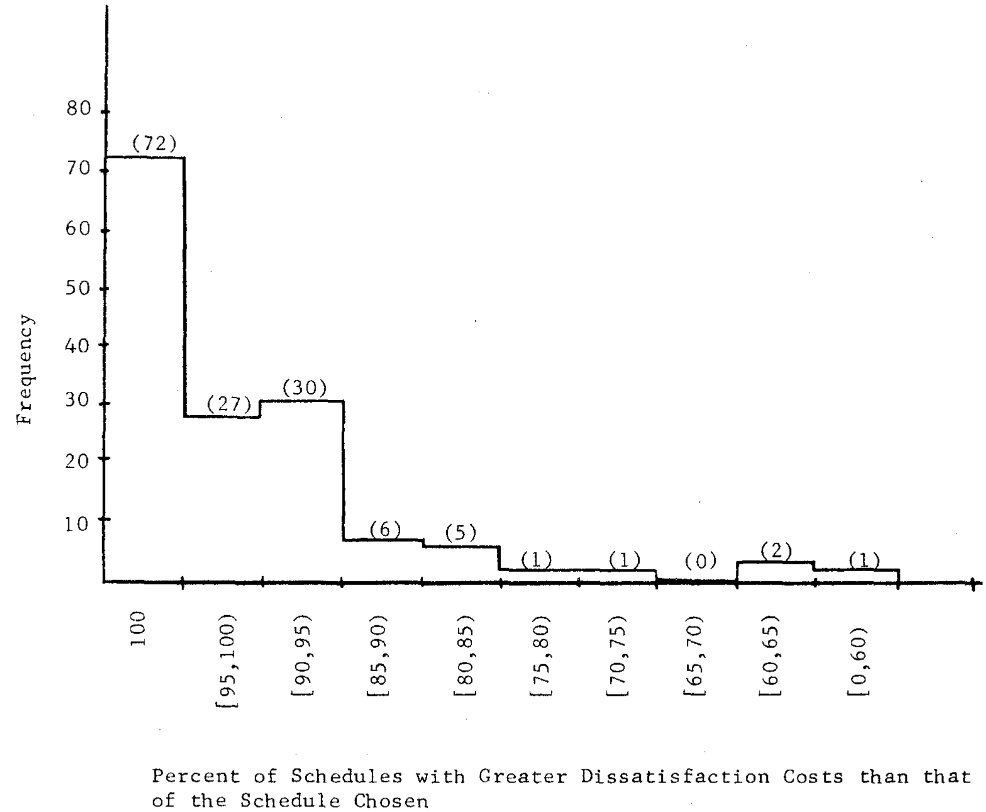

Fig. 4 Distribution of the Number of Schedules in the Feasibility Sets with Employee Dissatisfaction Costs Greater than that of the Schedules Chosen by the Algorithm (Over a Six Month Period)

5.2. Computer Running Time

The Central Processing Unit (CPU) time for the solution algorithm ranges from about $2\frac{1}{2}$ seconds to 8 seconds (on a CDC 6400), depending on the number of nurses and the number of schedules in their feasibility sets. In one run on a 4 nurse, 20 schedule problem, the cost of the algorithm generated *schedule* was *12.3* while the optimal cost was 7.55 (the *initial cost of the algorithm solution was 239.45*).The average solution time was around 5 seconds. In most instances the groups consisted of from five to seven nurses with an average of about 200 schedules in their feasible schedule sets. The current algorithm is coded for the IBM 360 and 370.

As was mentioned, the actual schedules used by the hospital in which the test was made were on record. Table 2 presents data relating to various schedule characteristics.

Four-or-more-days-off weekends were nearly the same in both cases. This is not surprising since most of these were due to special requests by the individual nurses. The algorithm, however, generated *13 more three- and two-day weekends* than the hospital did. This is a favorable feature since most nurses desire as many weekends off as possible. Moreover the algorithm generated *far fewer split weekends*. Again this is favorable since the hospital often does not desire to have such patterns. Both the algorithm and the hospital performed equally well in generating stretches under the

Table 2
Comparison of schedule pattern characteristics between hospital and the algorithm generated schedules

	Weekend Days Off				*Working Stretches*		*Consecutive Split Days Off (101 pattern)*		
	≥4 Day	3 Day	2 Day	Split	Under min	Over Desired	1	2	3
Hospital Schedules	11	17	43	25	19	57	26	13	5
Algorithm Schedules	9	21	52	10	21	44	27	21	6

individual nurse's minima but the hospital generated far more stretches over the nurse's maxima. In considering consecutive split days, the algorithm generated more in all instances although the only significant difference occurred in the generation of two consecutive split days.

We now define X_k as the number of personnel in a group scheduled for day k, d_k as the desired number of personnel needed on day k, and $D = |X_k - d_k|$ as the absolute deviation of actual from desired. This gives us some measure of the deviation of the actual staffing levels from the desired staffing levels. Table 3 gives more data relating to these deviations over the schedule periods in question.

Table 3
Comparison of some staffing level statistics from hospital and the NSS generated schedules

		Average D		Variance of D		$\sum_{k=1}^{14} (X_k - d_k)^2$	
Schedule Period	Pay Period	ALG	HOS	ALG	HOS	ALG	HOS
9/24	1	.786	.929	.169	.352	11	17
	2	1.071	1.071	.209	.495	19	23
10/22	1	.357	.786	.229	.454	5	15
	2	.643	.786	.230	.454	9	15
11/19	1	.357	.786	.229	.312	5	13
	2	.429	.571	.245	.387	6	10
12/17	1	1.000	1.286	.714	1.061	24	38
	2	1.143	1.429	1.694	1.673	42	52
1/14	1	.214	.786	.168	.454	3	15
	2	1.000	1.143	.429	.551	20	26
2/11	1	1.071	1.214	.495	.455	23	27
	2	.643	.786	.230	.454	9	15

Legend: $D = |X_k - d_k|$
HOS = Hospital Generated Schedule
ALG = Algorithm Generated Schedule by NSS

In every pay period of every schedule period, the average deviation D of the actual staffing level from the desired staffing level was as small as or smaller for the schedules generated by the algorithm than those generated for the hospital. Moreover in all cases but two, the variance of D was smaller for the algorithm generated schedules. One of these two occasions occurred in the pay period containing New Year's Day and on the second occasion, the variances differed by only .04.

Another measure of the variability of the actual vs. desired staffing levels is given in the last two columns of Table 3. This is the sum of the squares of the deviations. *In all cases* the sum of the squared deviations arising from the *algorithm* generated schedules *is less* than those from the hospital generated schedules.

6. The Implementation Process

Once the hospital's top *administrators have decided* to install the computerized algorithm, the implementation process begins. It proceeds through a series of steps over several months. The initial step is to *meet* with the *Director of Nursing* to explain the system, gather data on hospital scheduling policies such as the number of weekends off-on, maximum and minimum stretches, beginning day and length of pay periods, schedule horizon (usually four or six weeks), rotation, use of part-time and/or float personnel, etc.

Next there is a *group meeting* with the head nurses to explain the operations of the system, what it can and cannot do for them, the types of reports and schedules they will receive, the time savings to them, the problems which it eliminates for them, and the need for timely data on special requests. Emphasis is placed on the importance of cooperation on both sides and that *the computerized schedules do not take away any of the authority of the head nurse* in approving special requests of changing the schedules to meet unanticipated needs. The computerized algorithm is a tool which removes some onerous tasks so that they may have more time for more important tasks related to health care delivery.

Following the group meeting, *individual meetings* are scheduled with each head nurse. The purposes of the individual meetings are to answer any system questions and, more importantly, to gather data needed by the algorithm. The data comprise such items as who are the charge nurses, what groups and subgroups must be scheduled together, minimum and desired group and subgroup staffing levels, what specific scheduling problems are on the unit such as parallel people, part-time restrictions, rotation restrictions, fixed patterns, team vs. primary care groups, etc. Another important purpose of this meeting is to explain the limitations of the scheduler. For example, any group on a unit may specify which two days, Friday–Saturday or Saturday–Sunday, constitute a weekend; however, all of the nurses in *that* group must use the same definition for their weekend.

The next step is an *orientation meeting* with groups of nurses on the units. The main purpose here is to remove the fear of impersonalization by the computer and emphasize that the head nurse still controls the schedules and that the computer gives them fairer and *more individualized* schedules that meet their particular requests and preferences.

After the general orientation meetings each *individual is interviewed* (with more time devoted to nurses' aides, orderlies, and medical technicians) to obtain rankings of her (or his) preferences for weekends, stretches, split days off, etc. and to explain to her the interactions of such preferences on her schedules. They are also informed that all schedule changes or requests must be approved by the head nurse just as in the past.

Following each of the interviews, *data are prepared* and stored in the *Master File* of the algorithm. After all of the above data have been stored, trial schedules are run to adjust the various hospital and individual parameters. These trial schedules are reviewed by the respective head nurses to catch any items missed in prior interviews.

The nurse scheduling is now in operating condition and *periodic schedules* are produced. Even in this production phase, however, the schedules are reviewed every time and adjustments are made for new hires, terminations, changes in workload requirements and/or nurse preferences, etc. On a continuing basis it usually requires the *full-time work of one trained high school graduate*, who works well with people and enjoys the challenge of producing the best schedule for each head nurse, to operate the algorithm for a 40-unit hospital with 900 full and part-time nursing personnel. If the hospital is one-half this size, then only one-half the work is needed since the effort in running and maintaining the algorithm is essentially linear with respect to the number of units and people being scheduled.

The *savings* in head nurse time alone has been pointed out in other studies but at a minimum it is *one day per month per head nurse* and usually two to four days per month. Of course, head nurse's time is not the only advantage of the algorithm. Other advantages are: *fairer schedules* which meet nurse preferences, more even staffing of units and a *flexibility* which allows the nursing administration to examine the effects of changes in policies *prior to* the implementation of such changes.

7. The Model Extensions

The model may be extended to include shift rotation and part-time employees by redefining the feasibility sets π_i in an appropriate manner. For example, if we consider *shift rotation*, we:

(1) Schedule night and evening shifts first.

(2) If the staffing level patterns require shift rotation to reduce staffing costs, and if the day shift has nurses available to be rotated, select nurses from those available to rotate and have them rotate to the night and evening shifts. The exact rotation patterns selected must conform to various rotation constraints and must result in reduction of staffing costs on the shifts rotated to.

(3) Schedule the day-shift treating these rotation patterns as fixed conditions.

The problem of *part-time employees* is handled in a way analogous to full-time employees. Feasibility sets π_i are constructed for part-time employees depending on appropriately defined constraints (e.g., a nurse must work four days out of every fourteen). Then the schedules are listed according to how they meet a set of appropriately defined nonbinding constraints. Then we proceed in the same manner as with full-time employees, choosing schedules from the sets π_i where now some of these sets contain part-time nurses' schedules and some contain full-time nurses' schedules.

8. Summary and Conclusions

We have presented a mathematical programming model formulating the nurse scheduling problem as one where we find a set of feasible nurse schedules which seek to minimize the sum of schedule pattern *aversion costs* incurred by the nurses and staffing level pattern costs incurred by the hospital. The *model is realistic* in the sense that it can handle any number of any type of schedule pattern and staffing level constraints, and is general enough to be able to be applied in a number of different hospital settings with differing operating policies.

The solution algorithm given is a *cyclic coordinate descent method* which generates a solution specifying which days are on and off for each individual nurse being scheduled. The algorithm runs relatively quickly on the computer and generates solutions in a number of ways superior to those presently used in the hospital with which it was compared. Moreover, the algorithm generated solutions compared favorably with the optimal solution for a test problem.

A nurse scheduling algorithm based on the ideas presented in this paper has been implemented in a number of hospitals in the United States and Canada. The results from these implementation sites have been most favorable.

Notes

[1] H. Wolfe and J. P. Young, "Staffing the Nursing Unit: Part I," *Nursing Research*, Vol. 14, No. 3, 1965, pp. 236–243.

[2] H. Wolfe and J. P. Young, "Staffing the Nursing Unit: Part II," *Nursing Research*, Vol. 14, No. 4, 1965, pp. 299–303.

[3] J. S. Liebman, *The Development and Application of a Mathematical Programming Model of Personnel Allocation in an Extended Care Nursing Unit*, Ph.D. Thesis, Department of Operations Research and Industrial Engineering, Johns Hopkins University, Baltimore, Maryland, 1971.

[4] J. S. Liebman, J. P. Young, and M. Bellmore, "Allocation of Nursing Personnel in an Extended Care Facility," *Health Service Research Journal*, Vol. 7, No. 4, Fall, 1972, pp. 209–220.

[5] D. M. Warner, *A Two-Phase Model for Scheduling Nursing Personnel in a Hospital*, Doctoral Dissertation, Tulane University, New Orleans, 1971.

[6] D. M. Warner and J. Prawda, "A Mathematical Programming Model for Scheduling Nursing Personnel in a Hospital," *Management Science*, Vol. 19, No. 4, December 1972, pp. 411–422.

[7] W. Abernathy, N. Baloff, and J. Hershey, "A Variable Nurse Staffing Model," *Decision Sciences*, Vol. 5, No. 1, January, 1974, pp. 58–72.

[8] W. Abernathy, N. Baloff, J. Hershey, and S. Wandel, "A Three Stage Manpower Planning and Scheduling Model–A Service Sector Example," *Operations Research*, Vol. 21, No. 3, May–June, 1973, pp. 693–711.

[9] A. Morrish and A. O'Conner, "Cyclic Scheduling," *Hospitals J.A.H.A.*, Vol. 14, February, 1970, pp. 66–71.

[10] E. Price, "Techniques to Improve Staffing," *American Journ. Nursing*, Vol. 70, October, 1970, pp. 2112–15.

[11] J. P. Howell, "Cyclical Scheduling of Nursing Personnel," *Hospital J.A.H.A.*, Vol. 40, June, 1966, pp. 77–85.

[12] C. Maier-Rothe and H. Wolfe, "Cycle Scheduling and Allocation of Nursing Staff," Arthur D. Little & Co., Inc., Cambridge, Massachusetts, 1972, pp. 34.

[13] M. Rothstein, "Hospital Manpower Shift Scheduling by Mathematical Programming", *Health Service Research Journal,* Vol. 8, No. 1, Spring, 1973, pp. 60–66.

[14] D. M. Warner, "Scheduling Nursing Personnel According to Nursing Preference: A Mathematical Programming Approach," Paper presented at 45th ORSA Meeting, Boston, Massachusetts, April, 1974.