Electronic Preprint for Journal of Information Processing Vol.26

Regular Paper

A Support System for Nursery Staff Shift Scheduling —A Case Study at a Nursery School

Mari Ito[1,a)] Masatake Hirao[2,b)] Hiroki Hamahara[†1,c)]

Received: June 6, 2017, Accepted: December 8, 2017

Abstract: This paper introduces a 0-1 integer programming model for scheduling nursery staff shifts. A support system is designed to manage challenges such as satisfying nursery school requests and reducing the scheduling workload. The scheduling goals are to reduce the number of times per month nursery staff are given the same shift as well as the numbers of consecutive early, late, and identical shifts. To achieve this, we place weighted penalties on the constraints requested by the nursery school and use an objective function that minimized these penalties. The schedule obtained by the support system improved the management of nursery staff shift scheduling, and comparisons with schedules drawn up by hand confirmed the superiority of those produced by the system.

Keywords: nursery staff shift scheduling, 0-1 integer programming model, support system

1. Introduction

In recent years, Japan has been facing some problems due to a declining birth rate and a rapidly aging population. Childcare support is becoming an important policy issue. Due to the trend toward dual-income families, the number of children waiting to get into publicly certified nursery schools reached a record 23,167 as of April 2016 [1]. Kakiuchi [2] surveyed 1,000 nursery staff using a questionnaire, asking about their working environment. Kakiuchi emphasized the risks due to a heavy workload and long working hours and reported that the stressful working environment in nursery schools could not be overlooked. Kato and Ando [3] reported that the working environment needed to be improved by sharing a sense of purpose among nursery staff, developing interpersonal skills, and finding ways to cope with stress. Managers of nursery schools should improve the working environment by creating shift schedules that put less of a burden on nursery staff and meet the standards required by the Labor Standards Act.

The shift scheduling problem has been studied for a long time for airline, railway, and bus crews; medical workers; and store employees, among others. For example, the airline crews' scheduling problem is to assign pilots to flights considering the pilots' status, age, and need for rest days. Arabeyre et al. [4] surveyed approaches to airline crew scheduling, reporting that they usually involved 0-1 integer programming, the matrix of coefficients having a very special form. The nurse scheduling problem is assigning nurses to shifts considering the required levels of medical services and nursing staff. Miller et al. [5] formulated the nurse scheduling problem in terms of minimizing an objective function that balances the trade-off between staff coverage and the schedule preferences of individual nurses. It is difficult to apply these methods to nursery staff shift scheduling owing to the different working patterns in different industries. Moreover, no papers have yet proposed scheduling methods for nursery staff shift scheduling. Ernst et al. [6] reviewed papers on staff scheduling and rostering, observing that there is still significant room for improvement in this area. They did, however, foresee the wider applicability of rostering in the future because of the flexibility that more sophisticated rostering software tools will be able to provide.

Most nursery schools draw up schedules by hand or by, for example, assigning shifts randomly. Creating schedules manually while satisfying all constraints is a time-consuming and labor-intensive affair. On the contrary, when creating schedules by methods such as assigning shifts randomly, correcting the schedule takes time since the initial schedule does not satisfy the constraints. The target nursery school creates schedules manually; however, this is a time-consuming process and, even so, the schedules created are not always approved by the management. In fact, the schedule imposes a burden on some nursery staff. For example, some staff members have to work the same shift many times or work several consecutive early, late, or identical shifts per month. One reason for this is that there are multiple shifts, which are not easy to allocate properly. Our motivation for this research is to improve this situation in nursery schools.

We implement a support system for nursery staff shift scheduling (SS-NSSS) that resolves these problems, automatically creating schedules using simple processes. The system solves the

1 Department of Industrial Administration Faculty of Science and Technology, Tokyo University of Science, Noda, Chiba 278–8510, Japan
2 School of Information and Science Technology, Aichi Prefectural University, Nagakute, Aichi 480–1198, Japan
†1 Presently with Plainnovation Co., Ltd., Koriyama, Fukushima 963–8041, Japan
a) mariito@rs.tus.ac.jp
b) hirao@ist.aichi-pu.ac.jp
c) hiroki.hamahara@plainnovation.com

scheduling problem, which is formulated as a 0-1 integer programming problem, with weighted penalty terms for the constraints requested by the nursery school. We then verify the effectiveness of the SS-NSSS by applying it to an actual nursery staff schedule.

The remainder of this paper is organized as follows. Section 2 introduces the SS-NSSS implementation in detail. Section 3 formulates the nursery staff shift scheduling as a 0-1 integer programming problem. Section 4 shows the results of using this method at the target nursery school. Section 5 concludes the paper with suggestions for future research.

2. Support System for Nursery Staff Shift Scheduling

We implemented an SS-NSSS that can automatically create schedules using simple operations. The purpose of the system is to support nursery school managers in making nursery staff shift scheduling decisions.

Figure 1 shows the SS-NSSS's user interface. Here, the user inputs the information necessary for scheduling. For example, there are seven input items: "Number of staff," "Number of staff in charge of classes," "Year," "Month," "Number of early shifts," "Number of late shifts," and "Desired leave days." Here, for example, the "Number of staff in charge of classes" item shows that there are four classes such as one class requires 3 staff and the others all require 2 staff. The "Number of early shifts" item means that there are as many early shifts as there are brackets (i.e., here there are three early shifts). There are upper and lower bounds on the number of staff working early shifts per day, as well as upper and lower bounds on the number of early shifts worked by a staff member per month (given in parentheses on the left). The "Number of late shifts" item means that there are as many late shifts as there are parentheses (i.e., here there are three late shifts). There are upper and lower bounds on the number of staff working late shifts per day, as well as upper and lower bounds on the number of late shifts worked by a staff member per month (given in parentheses on the left). The "Desired leave days" item shows there have been six leave requests, and the names of the staff members and the days they have requested are shown in parentheses on the left.

Clicking the "Scheduling" button creates a monthly schedule using the PuLP Python package to solve the nursery staff shift scheduling problem formulated in Section 3. The schedules obtained are displayed on a single page (see Fig. 2). The SS-NSSS created a monthly schedule in 10 s, including setup time.

3. Model

We now introduce a model for nursery staff shift scheduling that allocates nursery staff to the most suitable shifts while satisfying as many of the constraints as possible. The target days are from Monday to Friday on regular working days. The constraints can be divided into the following three categories.

Category 1: Constraints due to the Labor Standards Act
Category 2: Constraints due to the nursery school's regulations
Category 3: Constraints requested by the nursery school

All Category 1 and 2 constraints must be followed, and as many Category 3 constraints should be satisfied as possible. The objective function minimizes the number of Category 3 constraint violations. We assigned multiple types of shift each day. The nursery staff shift scheduling problem is formulated as a 0-1 integer programming problem as follows.

<u>**Notation**</u>

Index sets

$\boldsymbol{I}$: the set of nursery staff
$\boldsymbol{C}$: the set of classes
$\boldsymbol{I_c}$ $(\boldsymbol{c \in C})$: the set of nursery staff in charge of class c, $I_c \subset I$
$\boldsymbol{J}$: the set of shifts
$\boldsymbol{J_e}$: the set of early shifts, $J_e \subset J$
$\boldsymbol{J_n}$: the set of late shifts, $J_n \subset J$, $J_e \cap J_n = \emptyset$
$\boldsymbol{T}$: the set of days
$\boldsymbol{T_0}$: the subset of T that excludes Fridays and the days just before a holiday, $T_0 \subset T$

Parameters

$\boldsymbol{w_{it}}$ $(\boldsymbol{i \in I, t \in T_0})$: 1 if staff member i desires to be off-duty on day t, and 0 otherwise
$\boldsymbol{\check{D}_i, \hat{D}_i}$ $(\boldsymbol{i \in I})$: lower and upper bounds on staff member i per day
$\boldsymbol{\check{O}_j, \hat{O}_j}$ $(\boldsymbol{j \in J})$: lower and upper bounds on shift j per month
$\boldsymbol{\check{E}_i, \hat{E}_i}$ $(\boldsymbol{i \in I})$: lower and upper bounds on the number of days worked by staff member i per month
$\boldsymbol{A_c}$ $(\boldsymbol{c \in C})$: lower bound on the number of staff in charge of class c (early shifts)
$\boldsymbol{B_c}$ $(\boldsymbol{c \in C})$: lower bound on the number of staff in charge of class c (late shifts)
$\boldsymbol{\bar{x}_{ij0}}$ $(\boldsymbol{i \in I, j \in J})$: 1 if staff member i worked shift j on the last day ($t = 0$) of the previous month, and 0 otherwise
$\boldsymbol{l}$: the last shift among the late shifts, $l \in J_n$
$\boldsymbol{\iota, \kappa, \nu, \upsilon, \zeta}$ **and** $\boldsymbol{\epsilon}$: weighting factors

Variables

$\boldsymbol{x_{ijt}}$ $(\boldsymbol{i \in I, j \in J, t \in T})$: 1 if staff member i is working shift j on day t, and 0 otherwise
$\boldsymbol{p_i, \check{s}_{ij}, \hat{s}_{ij}, \check{g}_i, \hat{g}_i, q_{ij}, k_i, f_i, h_i}$ **and** $\boldsymbol{r_i}$ $(\boldsymbol{i \in I, j \in J})$: penalties for constraint Eqs. (8), (9), (10), (11), (12), (13), (14) and (15)

<u>**Formulation**</u>

Objective function

$$\textbf{minimize} \quad \iota \sum_{i \in I} p_i + \kappa \sum_{i \in I} \sum_{j \in J} (\check{s}_{ij} + \hat{s}_{ij}) + \nu \sum_{i \in I} (\check{s}_i + \hat{s}_i) \tag{1}$$
$$+ \upsilon \sum_{i \in I} \sum_{j \in J} q_{ij} + \zeta \sum_{i \in I} (k_i + h_i) + \epsilon \sum_{i \in I} (f_i + r_i),$$

subject to

$$\sum_{j \in J} x_{ijt} \leq 1 \quad \forall i \in I, \forall t \in T, \tag{2}$$

$$\check{D}_i \leq \sum_{i \in I} x_{ijt} \leq \hat{D}_i \quad \forall t \in T, \forall j \in J, \tag{3}$$

$$\sum_{i \in I_c} \sum_{j \in J_e} x_{ijt} \geq A_c \quad \forall c \in C, \forall t \in T, \tag{4}$$

$$\sum_{i \in I_c} \sum_{j \in J_n} x_{ijt} \geq B_c \quad \forall c \in C, \forall t \in T, \tag{5}$$

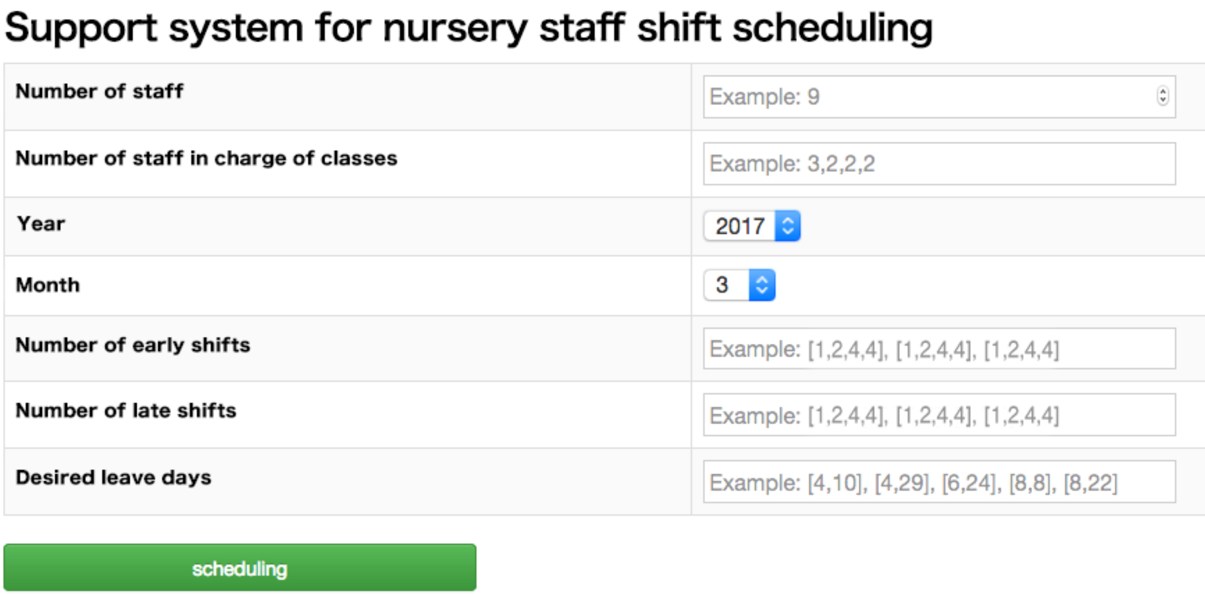

Fig. 1 SS-NSSS's user interface.

$$\bar{x}_{il0} + x_{i11} \le 1 \quad \forall i \in I, \tag{6}$$

$$\sum_{j \in J_n} x_{ijt} + x_{i1(t+1)} \le 1 \quad \forall i \in I, \forall t \in T, \tag{7}$$

$$\sum_{t \in T} \sum_{j \in J} w_{it} x_{ijt} \le p_i \quad \forall i \in I, \tag{8}$$

$$\check{O}_j - \check{s}_{ij} \le \sum_{t \in T} x_{ijt} \le \hat{O}_j + \hat{s}_{ij} \quad \forall i \in I, \forall j \in J, \tag{9}$$

$$\check{E}_i - \check{g}_i \le \sum_{t \in T} \sum_{j \in J} x_{ijt} \le \hat{E}_i + \hat{g}_i \quad \forall i \in I, \tag{10}$$

$$x_{ijt} + x_{ij(t+1)} \le 1 + q_{ij} \quad \forall i \in I, \forall j \in J, t \in T_0, \tag{11}$$

$$\sum_{j \in J_e} \bar{x}_{ij0} + \sum_{j \in J_e} x_{ij1} \le 1 + k_i \quad \forall i \in I, \tag{12}$$

$$\sum_{j \in J_n} \bar{x}_{ij0} + \sum_{j \in J_l} x_{ij1} \le 1 + f_i \quad \forall i \in I, \tag{13}$$

$$\sum_{j \in J_e} x_{ijt} + \sum_{j \in J_e} x_{ij(t+1)} \le 1 + h_i \quad \forall i \in I, \tag{14}$$

$$\sum_{j \in J_n} x_{ijt} + \sum_{j \in J_l} x_{ij(t+1)} \le 1 + r_i \quad \forall i \in I, \tag{15}$$

$$x_{ijt} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall t \in T, \tag{16}$$

$$p_i, \check{s}_{ij}, \hat{s}_{ij}, \check{g}_i, \hat{g}_i, q_{ij}, k_i, f_i, h_i, r_i \in \mathbb{Z}_+, \forall i \in I, \forall j \in J. \tag{17}$$

In the formulation above, the objective function Eq. (1) minimizes the sum of the weighted penalty terms for the constraints. Equation (2) means that staff member i is assigned to one shift on day t. Equation (3) means that the number of staff assigned to shift j on day t is between the lower bound $\check{D}_i$ and the upper bound $\check{D}_i$. Equation (4) means that a sufficient number of the staff in charge of class c are assigned to shift j for the early part of day t. Equation (5) means that a sufficient number of the staff in charge of class c are assigned to shift j for the last part of day t. Equation (6) does not assign staff member i to the earliest shift ($j = 1$) on the first day ($t = 1$) of this month if they were assigned to the last shift l on the last day ($t = 0$) of last month. Equation (7) does not assign staff member i the earliest shift ($j = 1$) on day $t + 1$ if they are assigned the last shifts on day t. Equation (8) denotes that we take account of the penalty for ignoring the staff members' leave requests. Equation (9) means that the number of shifts j per month is equal for all staff. Equation (10) means that the number of day t per month is equal for all staff. Equation (11) means that staff member i cannot be assigned the same shift j for both day t and day $t + 1$. Equations (12) and (14) denote that staff member i cannot be assigned early shifts for two consecutive days. Equations (13) and (15) denote that staff member i cannot be assigned late shifts for two consecutive days. Equation (16) is a binary constraint. Equation (17) requires that some of the variables should be non-negative integers.

Equation (2) is in Category 1, as it indirectly limits working hours by assigning only one shift per day (each shift takes a maximum of 8 hours). Equations (3)–(7) are in Category 2, while Equations (8)–(15) are in Category 3.

4. Nursery Staff Shift Scheduling at the Target Nursery School

The target nursery school is a private nursery school in Japan.

4.1 Data

At present, there are 9 full-time and 16 part-time nursery staff at the nursery school. Number of infants enrolled in this school is 43. There are 6 shifts for full-time staff and 18 shifts for part-time staff. The part-time are allocated workdays and shifts on a monthly basis. The shifts of part-time nursery staff are not subject to scheduling because the shifts of part-time nursery staff are given as fixed shifts. We assign 6 shifts to the 9 full-time staff (i.e., $|I| = 9$ and $|J| = 6$). There are 4 classes (i.e., $|C| = 4$). One class requires 3 staff and the others all require 2 staff (i.e., $|I_1| = 3; |I_2| = 2; |I_3| = 2$ and $|I_4| = 2$). **Table 1** shows the working hours for each shift, including breaks.

There are three early shifts ($J_e = 1; 2; 3$) and three late shifts

	Mar-17																														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri
staff member 1	F	C	A			D	A	E	B	E			C	F	C		A				C	F	D	A			B	D	F	D	E
staff member 2	D	E	A			F	C	A	E	C			D	D	F	C	E				F	D	A	E			F	B		B	C
staff member 3	B	D	F			C	E	D		A			E	B	A	F	B				A	C	E	C			D	F	D	F	B
staff member 4	E	C	E			C	F	B	F	B			D	F	D	E	C				B	E	C	F			E	A	E	A	F
staff member 5	A	F	C			E	B	F	D	F			E	A	E	A	E				F	C	E	B			C	E	C	F	D
staff member 6	E	A	F			E	D	F	C	E			F	C	F	B	F				E	A	F	C			D	E	A	E	B
staff member 7	C	F	D			A	F	C	F	D			B	E	B	F	D				A	F	B	E			E	C	F	B	E
staff member 8	B	E	B			B	E	C	E	D			F	D	E	D	F				D	E	C	F			A	F	B	E	A
staff member 9	F	B	E			F	D	E	A	F			A	E	A	E	A				E	B	F	D			F	C	E	C	F

Fig. 2 Schedule obtained by the SS-NSSS, A: 7:00–16:00, B: 8:00–17:00, C: 8:30–17:30, D: 9:00–18:00, E: 9:15–18:15, F: 9:30–20:15.

Table 1 Working hours for each shift, including breaks.

	Shift	Index (j)	Working hours
Early shift	A	1	7:00–16:00
Early shift	B	2	8:00–17:00
Early shift	C	3	8:30–17:30
Late shift	D	4	9:00–18:00
Late shift	E	5	9:15–18:15
Late shift	F	6	9:30–20:15

Table 2 Parameters of w_{it}.

Nursery staff	Off-duty desired by nursery staff in March 2017
Staff member 1	None
Staff member 2	None
Staff member 3	None
Staff member 4	10, 29
Staff member 5	None
Staff member 6	24
Staff member 7	None
Staff member 8	8, 22
Staff member 9	None

($J_n = 4; 5; 6$). There were 22 working days (days other than holidays, Saturdays, and Sundays) in March 2017 (i.e., $|T| = 22$). T_0 excludes Fridays and the days just before a holiday (i.e., $|T_0| = 18$). The off-duty requests w_{it} are shown in **Table 2**. On the last day of the previous month, the staff members were working as follows (i.e., $\bar{x}_{ij0}$): staff member 1 was on shift A, staff member 2 was on shift E, staff member 3 was off-duty, staff member 4 was on shift B, staff member 5 was on shift D, staff member 6 was on shift C, staff member 7 was on shift F, staff member 8 was on shift A, and staff member 9 was on shift D. The lower and upper bounds on the number of shifts per staff member per month were both 21 (i.e., $\check{E}_i = 21, \hat{E}_i = 21$), while the bounds on the number of shift j per month were both 4 (i.e., $\check{O}_j = 4, \hat{O}_j = 4$). The lower bounds on staff member i per day was 1 (i.e., $\check{D}_i = 1$), on the other hand the upper bounds on staff member i per day was 2 (i.e., $\hat{D}_i = 2$). The lower bounds on the number of staff in charge of class c for the early and late shifts were 1 (i.e., $A_c = 1; B_c = 1$). The weighting factors ($\iota; \kappa; \nu; \upsilon; \zeta$ and ϵ) were all 1.

The computer used to generate the schedule was equipped with an Intel Core i7 3.5-GHz CPU and 32 GB of 1600-MHz DDR3 of RAM. The CPU time required to solve the formulated 0-1 integer programming problem was 1 s using PuLP 1.6.5 on Python 3.6.0.

4.2 Nursery Staff Shift Scheduling Results

The problem involved 1152 variables and 1,920 constraints. **Figure 2** shows the schedule obtained by the SS-NSSS. We compared three schedules: a manually constructed schedule, a random schedule, and the schedule obtained by the SS-NSSS. We now consider these three schedules from the following standpoints.

Table 3 Number of times per month nursery staff members work the same shift: manual schedule.

Nursery staff	Shifts					
	A	B	C	D	E	F
Staff member 1	4	0	9	1	4	4
Staff member 2	4	2	6	4	2	4
Staff member 3	4	2	4	4	3	5
Staff member 4	5	2	3	4	2	4
Staff member 5	5	3	6	1	2	5
Staff member 6	4	2	8	2	1	4
Staff member 7	5	3	6	2	2	4
Staff member 8	4	3	6	1	2	4
Staff member 9	3	4	6	2	3	4

Table 4 Number of times per month nursery staff members work the same shift: random schedule.

Nursery staff	Shifts					
	A	B	C	D	E	F
Staff member 1	6	2	1	3	6	4
Staff member 2	3	4	3	0	8	4
Staff member 3	3	4	3	2	4	6
Staff member 4	3	2	3	1	4	7
Staff member 5	1	2	6	3	6	4
Staff member 6	2	1	4	3	6	5
Staff member 7	3	4	2	3	4	6
Staff member 8	2	4	0	2	7	5
Staff member 9	2	1	3	6	5	5

Table 5 Number of times per month nursery staff members work the same shift: SS-NSSS schedule.

Nursery staff	Shifts					
	A	B	C	D	E	F
Staff member 1	4	2	4	4	3	4
Staff member 2	3	2	4	4	4	4
Staff member 3	3	4	3	4	3	4
Staff member 4	2	3	4	2	6	5
Staff member 5	3	2	4	2	6	5
Staff member 6	3	2	3	2	6	6
Staff member 7	2	4	3	3	4	6
Staff member 8	2	4	2	4	6	4
Staff member 9	4	2	2	2	6	6

(a) : Number of times per month nursery staff members work the same shift

Tables 3–5 list the number of times each staff member would have been on duty on a given shift for each of the three schedules. Note that the manual and random schedules put some staff on the same shifts many times over the month, whereas the SS-NSSS schedule reduces such cases. From

	Mar-17																														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri
staff member 1	F	D	F			C	F	B	E	B			F	B	A	E	C				A	C	E	C			A		B	E	D
staff member 2	B	F	B			A	C		D	F			D	D	E	A	A				A	E	D	E			F	C	F	C	E
staff member 3	A		C			E	A	E	A	D			A	F	C	D	F				F	D	F	C			B	E	C	E	B
staff member 4	D	E	A			E	B	F	D	X			E	C	F	C	E				B	B	F	B			F	D	X	D	A
staff member 5	E	A	E			D	E	A	F	C			D	F	B	F	B				C	E	B	A			D	F	B		C
staff member 6	E	C	E			A	A	E	C	B			C	B	E	D	F				B	F	C	X			D	F	D	F	B
staff member 7	C	E	A			B	E	D	F	C			B	E	A	F	D				D	A	E	D				B	F	B	F
staff member 8	F	B	D			F	D	X	B	E			C	E	D	E	C				D	X	A	F			E	A	E	A	F
staff member 9	D	F	C			B	F	C	E	A			E	A	F	B	D				E	B		B			C	E	A	F	D

Fig. 3 Schedule that satisfies all the staff members' leave requests, obtained using SS-NSSS, A: 7:00–16:00, B: 8:00–17:00, C: 8:30–17:30, D: 9:00–18:00, E: 9:15–18:15, F: 9:30–20:15, X: requested leave days.

Table 6 Number of consecutive times nursery staff members work an early shift.

Nursery staff	Manual	Random	SS-NSSS
Staff member 1	2	1	1
Staff member 2	6	0	2
Staff member 3	1	0	2
Staff member 4	4	0	0
Staff member 5	8	1	0
Staff member 6	7	0	0
Staff member 7	6	0	0
Staff member 8	5	0	0
Staff member 9	6	0	0

Table 7 Number of consecutive times nursery staff members work an late shift.

Nursery staff	Manual	Random	SS-NSSS
Staff member 1	4	6	4
Staff member 2	3	1	3
Staff member 3	6	4	5
Staff member 4	4	3	3
Staff member 5	3	6	3
Staff member 6	1	5	3
Staff member 7	3	4	3
Staff member 8	1	5	6
Staff member 9	2	8	3

Table 3, staff member 1 is on shift C nine times but never on shift B. From Table 4, staff member 4 is on shift F seven times, but staff member 2 is never on shift D and staff member 8 is never on shift C. By contrast, from Table 5, staff members 4, 5, 6, 8, and 9 are each on shift E six times, while staff members 6 and 9 are on shift F six times.

(b) : Number of consecutive times nursery staff members work an early or late shift

Tables 6 and **7** list the number of consecutive early or late shifts for each staff member. Note that whereas the manual and random schedules make some staff work several consecutive early or late shifts, the SS-NSSS schedule reduces such cases. From Table 6, the manually constructed schedule gave staff member 5 eight consecutive early shifts, but only gave staff member 3 one. In the random schedule, only staff members 1 and 5 even had one consecutive early shift. In the SS-NSSS schedule, staff members 2 and 3 were given two consecutive early shifts and staff member 1 was given one. From Table 7, the manually constructed schedule gave staff member 3 six consecutive late shifts, while staff members 6 and 8 only received one. In the random schedule, staff members 1 and 5 were given six consecutive late shifts. In the SS-NSSS schedule, only staff member 8 was given six consecutive late shifts.

Table 8 Number of consecutive times nursery staff members work the same shift.

Nursery staff	Manual	Random	SS-NSSS
Staff member 1	2	0	0
Staff member 2	1	0	1
Staff member 3	1	0	0
Staff member 4	0	2	0
Staff member 5	0	0	0
Staff member 6	1	1	0
Staff member 7	1	0	0
Staff member 8	0	3	0
Staff member 9	0	0	0

(c) : Number of consecutive times nursery staff members work the same shift

Table 8 lists the number of times each staff member was given the same shift consecutive times. Note that whereas the manual and random schedules make some nursery staff work the same shift several times consecutively, the SS-NSSS schedule reduces such cases. In the manually constructed schedule, five staff have to work the same shift consecutively. In particular, staff member 1 has to work the same shift twice consecutively. In the random schedule, three staff have to work the same shift consecutively. In the SS-NSSS schedule, however, only one staff member has to work the same shift consecutively.

From all three standpoints, the SS-NSSS schedule is superior to the manually constructed and random schedules.

Figure 2 shows the optimal solution when all the weighting factors were set to one. If we want all the leave requests by the nursery staff to be satisfied, the weighting factor ι should be large. **Figure 3** shows a schedule that satisfies all the staff members' leave requests, obtained using SS-NSSS. The requested leave days are marked with an "X." In creating this schedule, we eased the Category 2 Eqs. (4) and (5). The nursery school's schedule also eases these requirements in order to satisfy all the staff members' leave requests. **Table 9** shows number of times per month nursery staff members work the same shift in Fig. 3. **Table 10** shows the number of times each staff member was given each shift by the schedule in Fig. 3, and Table 10 shows the corresponding numbers of consecutive early, late, and identical shifts. Table 9 and Table 10 show that the schedule in Fig. 3 is superior to the manually constructed and random schedules from all three of the standpoints above. From Table 9, most of the staff re-

Table 9 Number of times per month nursery staff members work the same shift in Fig. 3.

Nursery staff	Shifts					
	A	B	C	D	E	F
Staff member 1	3	4	4	2	4	4
Staff member 2	4	2	3	4	4	4
Staff member 3	4	2	4	3	4	4
Staff member 4	2	4	2	4	4	4
Staff member 5	3	4	3	3	4	4
Staff member 6	2	4	4	3	4	4
Staff member 7	3	4	2	4	4	4
Staff member 8	3	2	2	4	4	4
Staff member 9	3	4	3	3	4	4

Table 10 Number of consecutive times nursery staff members work early, late and the same shift in Fig. 3.

Nursery staff	Number of early	Number of late	Number of the same
Staff member 1	2	3	0
Staff member 2	1	4	2
Staff member 3	0	3	0
Staff member 4	0	3	1
Staff member 5	1	3	0
Staff member 6	2	5	1
Staff member 7	0	4	0
Staff member 8	0	3	0
Staff member 9	0	2	0

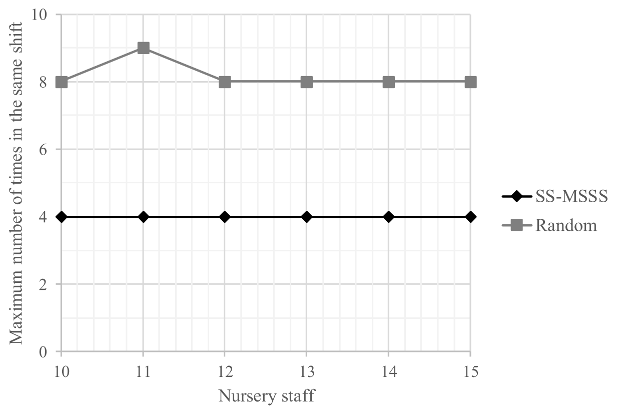

Fig. 4 Effect of number of nursery staff on maximum number of times in the same shift.

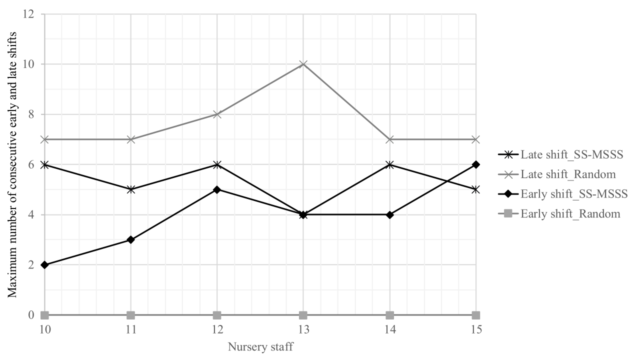

Fig. 5 Effect of number of nursery staff on maximum number of consecutive early and late shifts.

ceived each shift four times. From Table 10, staff members 1 and 6 were given two consecutive early shifts, while staff members 2 and 5 received one. Staff member 6 was given five consecutive late shifts. Three staff worked the same shift consecutively at least once. We verify whether equality of nursery staff's shift is maintained even when the parameter affecting the shift changes. The effect of number of nursery staff $|I|$ on maximum number of times in the same shift is analyzed and shown in **Fig. 4** for the given $|I|$ from 10 to 15. It is identified that SS-NSSS schedule suppresses assignment to the same shift more than the random one. **Figure 5** shows the effect of number of nursery staff $|I|$ on maximum number of consecutive early and late shifts. Whereas random schedules have a lot of consecutive late shifts, the SS-NSSS schedules reduce such cases. For example, a nursery works consecutive late shifts at 10 times when $|I| = 13$. Figure 5 has the same tendency as Section 4.2, (b). Even if the parameter affecting the shift changes, the SS-NSSS schedules are superior to random schedules from an equality point of view. These results come from the formulation of Section 3.

5. Conclusion

In this study, we developed a support system, the SS-NSSS, to address shift scheduling problems at nursery schools. We verified the effectiveness of the SS-NSSS by applying it to an actual nursery schedule. The resulting system reduced the number of times staff members had to work the same shift, as well as the numbers of consecutive early, late, or identical shifts they were given. Therefore, the SS-NSSS can provide schedules that ease the burdens on nursery staff and save time. There is a high possibility that SS-NSSS-generated schedules will improve nursery school management. In future work, we will focus on other nursery school cases to enhance the system's versatility.

Acknowledgments This work was supported by JSPS KAKENHI Grant Number JP16H07226.

References

[1] Ministry of Health, Labour and Welfare, http://www.mhlw.go.jp/stf/houdou/0000078441.html (accessed 2017-06-02).
[2] Kakiuchi, K.: *Nurseries'present: Expertise and working environment*, Minerva Shobo (2007) (in Japanese).
[3] Kato, Y. and Ando, M.: Research Trends in Difficulties Faced by New Preschool Teachers, *Bulletin of Graduate School of Education, Okayama University*, Vol.151, pp.23–32 (2012) (in Japanese).
[4] Arabeyre, J.P., Fearnley, J., Steiger, F.C. and Teather, W.: The airline crew scheduling problem: A survey, *Transportation Science*, Vol.3, No.2, pp.140–163 (1969).
[5] Miller, H.E., Pierskalla, W.P. and Rath, G.J.: Nurse scheduling using mathematical programming, *Operations Research*, Vol.24, No.5, pp.857–870 (1976).
[6] Ernst, A.T., Jiang, H., Krishnamoorthy, M. and Sier, D.: Staff scheduling and rostering: A review of applications, methods and models, *European Journal of Operational Research*, Vol.153, No.1, pp.3–27 (2004).

Mari Ito was born in 1989. She received Master and Doctor of Mathematical Science and Information Technology from Nanzan University in 2013 and 2016, respectively. She is currently an assistant professor at Tokyo University of Science. Her research interests are operations research, scheduling in health services, economic analysis of policies.

Masatake Hirao was born in 1980. He received Master and Doctor of Information Science from Nagoya University in 2006 and 2010, respectively. He is currently an associate professor at Aichi Prefectural University. His research interests are mathematical statistics, probability theory, combinatorics and numerical analysis.

Hiroki Hamahara was born in 1990. He received Bachelor of the School of Business Administration, Meiji University. He currently belongs to Plainnovation Co., Ltd.