Winter 2018

Shift Scheduling Optimization for PSU Library

Aayushi Gupta
Portland State University

Anju Babu
Portland State University

Lipishree Vrushabhendra
Portland State University

Shivani Purwar
Portland State University

Shravankumar Doosa
Portland State University

TITLE: SHIFT SCHEDULING OPTIMIZATION FOR PSU LIBRARY

Course Title: **Operations Research**

Course Number: **ETM 540**

Instructor: **Dr. Timothy Anderson**

Term: **Winter**

Year: **2018**

Authors: **Aayushi Gupta,**

Anju Babu

Lipishree Vrushabhendra

Shivani Purwar

Shravankumar Doosasreenath

Date: **03/23/2018**

ETM OFFICE USE ONLY

Report No.:

Type: Student Project

Note:

TABLE OF CONTENTS

ABSTRACT

Scheduling is important in any business as it creates an order and flow ensuring that all the tasks are covered at appropriate times. According to experts, scheduling determines the economics of a job, the quality of the team, and the skill-building and motivation of professionals doing the work. Therefore, it is essential to have optimized staff schedules to meet the requirements of staff availability, tasks coverage, shift equity and staff preferences. Though staff scheduling is of such prime importance, it is mostly implemented in traditional ways of manually creating spreadsheets and web calendars proving to be laborious and often leaving room for errors.

Additionally, staff preferences are arbitrarily handled through this format which results in overstaffing /understaffing of resources. Our project is aimed at developing an optimization model of staff scheduling for the PSU library using linear programming and create a tool with open solver that reduces the surplus working hours of the staff in the library while maximizing the staff preferences. We expect our model to achieve better efficiency and flexibility than the traditional format of scheduling implemented by the library. Also, our model could have broader capabilities of implementation in different departments of the Portland State University.

EXECUTIVE SUMMARY

A library is an integral part of any university proving to be a source of innumerable resources and providing a platform for those who seek such resources. The Miller library at the Portland State University is no different and is aligned with the university's values and commitment to advance the intellectual vitality, knowledge and creativity. It is a reservoir of articles, books, student and course guides spanning over wide range of specialties. The library staff play a vital role in enabling these resources to reach the students/faculty at any point of time. Additionally, the staff provides various student services of resolving issues, providing guidance and information related to the availability of resources and dealing with enquiries. Therefore, taking into account the various responsibilities of the library staff, scheduling plays a key role by being the interface resulting in the library staff being utilized to the fullest.

As described earlier, the library staff is engaged with various tasks and responsibilities. These tasks are performed through two important desks. The first one is the reference desk and the second is the circulation desk. The circulation desk comprises of the student staff who cater to the request volumes. These requests differ variably, depending on the time of the day, day of the week, and time of the year. The other responsibilities of the circulation desk staff include such stacking, drops/hold, pick up, giving information and meeting faculty requests.

Our project carried out an in-depth analysis of the circulation desk and the current application used for scheduling the staff. We learnt from our findings that the staff scheduling is performed using spreadsheets wherein the where the admin uses google sheets to assign the schedules to the staff on preferred days and times. The other method of scheduling is via emails with the schedule information for the respective staff. It is observed that this traditional format of assigning schedule is proving to be labor intensive and prone to errors. Also, this method fails to adequately meet the preferences of the library staff.

Therefore, our project intends to build a model that provides an optimized solution for staff scheduling taking into account both the demand and the preferences of the staff. By defining two objectives, our final model consists of three sub models which are designed using the techniques of integer linear programming and goal programming. Additionally, we implement these models using the open solver as we have huge data sets and variables. The results obtained after implementation show that the model developed is scalable and optimized for the resource utilization and scheduling. We also believe that the final model has achieved an improved efficiency that would help the library to reduce the operating costs of staffing of the circulation desk.

PROBLEM DEFINITION

The staff scheduling is performed using spreadsheets where the admin uses either google sheets or emails to assign schedules to the staff according to the demand requirements at the library. The total number of student staff is 24. While assigning a schedule to each one of them, their availability and preferences should be considered as most or all of them are full-time students who have to attend classes and finish other academic obligations. The student employment in campus aims to provide employment to students while not compromising on their academic performance. Hence, the availability and preference criteria become important.

The scheduling is dependent on other constraints as well; the top priority being customer service. The line at the circulation desk at any given time has to be maintained less than 5 students. Thus, at least 3 employees, where 2 would be student employees, have to be there at the circulation desk during peak hours. Another factor that should be considered is the maximum number of hours a student can work. The Federal Work-Study law mandates full-time students not to work for more than 20 hours per week. Also, if a student works for more than 5 hours continuously during a day, he/she has to be given a mandatory break which will add towards the unproductive time and unnecessary expenditure on the library's part.

Taking all these requirements and constraints into account and developing a schedule manually is a herculean task. Also, a manual schedule entry could be prone to human errors that result in overstaffing or understaffing.

Therefore, our problem statement is:

"The current application being used requires manual entry of the student's time schedule and reporting via emails, which eventually is creating an overlap on student's staffing time schedules".

OBJECTIVE

Our objective is to create a comprehensive and scalable tool for assigning schedules that will also take care of staff preferences. Since there is a surplus of hours to take care of and adhere to maximize the preferences of the student staff. Our project intends to support the library by developing an optimization model which will allow scheduling of student staff within the constraints of the library's operations.

Therefore, we define the following two objectives,

- Objective 1-Minimizing the surplus hours on each day of the week
- Objective 2-Maximizing the preferences of the student staff

LITERATURE REVIEW

1.Operations Research & Linear Programming

Operation research is the application of scientific & mathematical methods to the study & analysis of problems involving complex systems. The field of operations research began in the 1940s as mathematicians developed techniques for practical problem solving. Today, Operations Research is the application of advanced analytical methods to help make better decisions. In today's competitive business environment, it is increasingly important to make sure that a company's limited resources are used in the most efficient manner possible [2]. Typically, this involves determining how to allocate the resources in such a way as to maximize profits or minimize costs. Mathematical programming (MP) in Operations Research finds an efficient, way of using limited resources to achieve the objectives of an individual or a business. Mathematical programming [MP]is therefore referred to as optimization [2]. A diverse number of MP problems have been encountered, which has led to the development of many techniques to solve such problems. Linear programming (LP) is one of those techniques, which involves creating and solving optimization problems with linear objective functions and linear constraints [2]. An integer programming model is a linear program with the requirement that some or all of the decision variables must be integers [1]. The ability to treat variables as integer valued, and, in particular, the ability to designate certain variables as binary, opens up a wide variety of optimization models that can be addressed with Solver [1]. It is regarded as one of the powerful tools that can be applied in many business situations.

2. Open Solver

The Solver is an inbuilt program of Microsoft Excel for Windows that could be used for what-if analysis and find an optimal solution for problems that are subjected to constraints or limits and the values of other formula cells on a worksheet [3]. It works with a group of cells, called decision variables or variable cells that are used in computing the formulas in the objective and constraint cells. Solver adjusts the values in the decision variable cells to satisfy the limits on constraint cells and produce the result or the optimal solution for the objective cell. Though this in-built solver is an ideal tool for delivering optimal results to the end users, it poses limitations on the maximum size of the data or models it can hold and provide the optimal solution [5]. Open Solver which has been developed as a freely available open source excel add-in for Microsoft Windows overcomes this limitation of the excel solver and can handle large sets of data variables of linear programming or integer linear programming models [4]. The open solver is also compatible with the existing solver models and allows models to be solved without any change to the spreadsheets [5]. Additionally, the open solver is faster than the excel solver and provides novel model construction with better on-sheet visualization capabilities, by highlighting the model’s decision variables, objective and constraints directly on the spreadsheets [5].

3. Scheduling

Workforce scheduling has been used extensively in all the industries ranging from healthcare and manufacturing. The objective is to minimize the amount of labor used while providing enough service to satisfy demand [6]. Appropriate scheduling has a direct impact on cost and employee satisfaction. Scheduling problems can be modeled by binary and integer linear programming methodologies. The workflow of the organization/division and the availability of workers are taken into consideration. There could be capacity and schedule conflict constraints that need to be incorporated in the linear model. Depending on the size of the LP, feasibility and quality tests are done using smaller samples from the available data. Once found feasible and tested for the quality requirements, the model can be implemented for the whole set of data.

4. Goal Programming

Multiple objectives and multiple criteria demand the use of non-traditional linear programming techniques. Recently, goal programming (GP) has received the most attention among optimization techniques, as it attempts to optimize a number of objectives simultaneously. These objectives include: maximizing utilization of full-time staff, minimizing under-scheduling and overstaffing costs, minimizing payroll costs, as well as minimizing deviations from desired staffing requirements, staff preferences, and staff special requests [7]. Even if there is no more than one goal to be optimized in a model, the concept introduces the deviation constraints which can be used to minimize the difference within a dataset or within different datasets.

OPTIMIZATION MODEL

3 models were used to attain the aforementioned objectives. While Model 1 attempts to achieve Objective 1, Model 2 & 3 aim to achieve Objective 2.

Model 1

The first objective of the project which is to minimize the surplus hours is achieved by Model 1. The constraint of this model would be to meet the demand of required student staff while trying to reduce the surplus.

Data & Decision Variables

We first define the indices that will be used in all three models. The indices are described below:

s: Shift
l: length of shift
t: time slot of 1 hour
w: day of the week

The decision variables are,

s_{ltw}: 1, if shift of length l covers time t on day w; 0, otherwise
r_{tw}: required number of students at time t on day w
n_{ltw}: number of students working at time t on day w with shift length l

Objective Function:

Min $\sum_t [r_{tw} - (n_{ltw} \times s_{ltw})]$

The above function represents the objective of minimizing the surplus students.

Constraints:

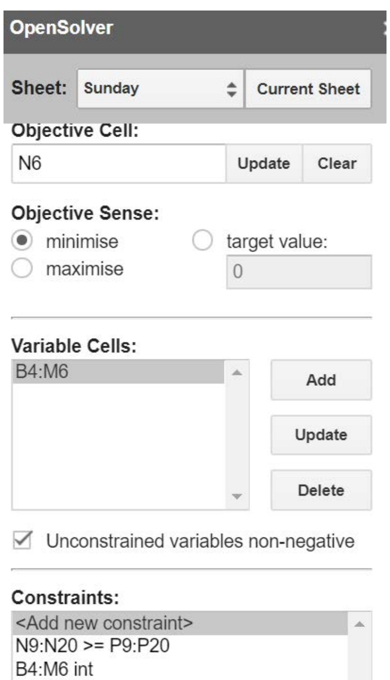
1. This is the demand Constraint where n is the number of students working over shift length l at time t, on day w and should meet the required demand r
2. $\sum_l (n_{ltw} \times s_{ltw}) \geq r_{tw} \quad \forall\ l, t, w$ (1.1)
3. Integrality Constraint: n_{ltw} is integer (1.2)

Implementation and Results

The above model is implemented by building the data of the 3-hour, 4-hour and 5-hour shifts for every hour and each day of the week to calculate the minimum number of students required to cover those shifts. The screenshot below, shows data for a particular Sunday where the constraint is set to meet the demand of the number of students required for each 3 hr., 4 hr. and 5 hr. shift while meeting the objective of reducing the surplus.

Results indicate that with the implementation of this model, the number of total hours for all the students which was previously 406 is now reduced to a number of 388. In comparison with the existing application used in the library, this model achieves a 4.5% of reduction in surplus.

Surplus Hour Minimization: Sunday																
Time	12	13	14	15	16	17	18	19	20	21	22	23				
Decision Variables																
5-hr	1	1	0	0	1	0	0	0	0	0	0	0				
4-hr	1	0	0	0	1	0	0	0	0	0	0	0	8		Total no: of Students	
3-hr	0	0	0	0	0	0	1	0	0	2	0	0	0		Surplus	
Data															**Constraints**	
12	1	0	0	0	0	0	0	0	0	0	0	0	2	>=	2	
13	1	1	0	0	0	0	0	0	0	0	0	0	3	>=	3	
14	1	1	1	0	0	0	0	0	0	0	0	0	3	>=	3	
15	1	1	1	1	0	0	0	0	0	0	0	0	3	>=	3	
16	1	1	1	1	1	0	0	0	0	0	0	0	4	>=	4	
17	0	1	1	1	1	1	0	0	0	0	0	0	3	>=	3	
18	0	0	1	1	1	1	1	0	0	0	0	0	3	>=	3	
19	0	0	0	1	1	1	1	1	0	0	0	0	3	>=	3	
20	0	0	0	0	1	1	1	1	0	0	0	0	2	>=	2	
21	0	0	0	0	0	1	1	1	0	0	0	0	2	>=	2	
22	0	0	0	0	0	0	1	1	0	0	0	0	2	>=	2	
23	0	0	0	0	0	0	0	1	0	0	0	0	2	>=	2	
12	1	0	0	0	0	0	0	0	0	0	0	0	32			
13	1	1	0	0	0	0	0	0	0	0	0	0				
14	1	1	1	0	0	0	0	0	0	0	0	0				
15	1	1	1	1	0	0	0	0	0	0	0	0				
16	0	1	1	1	1	0	0	0	0	0	0	0				
17	0	0	1	1	1	1	0	0	0	0	0	0				
18	0	0	0	1	1	1	1	0	0	0	0	0				
19	0	0	0	0	1	1	1	1	0	0	0	0				
20	0	0	0	0	0	1	1	1	1	0	0	0				
21	0	0	0	0	0	0	1	1	1	0	0	0				
22	0	0	0	0	0	0	0	1	1	0	0	0				
23	0	0	0	0	0	0	0	0	1	0	0	0				
12	1	0	0	0	0	0	0	0	0	0	0	0				
13	1	1	0	0	0	0	0	0	0	0	0	0				
14	1	1	1	0	0	0	0	0	0	0	0	0				
15	0	1	1	1	0	0	0	0	0	0	0	0				
16	0	0	1	1	1	0	0	0	0	0	0	0				
17	0	0	0	1	1	1	0	0	0	0	0	0				
18	0	0	0	0	1	1	1	0	0	0	0	0				
19	0	0	0	0	0	1	1	1	0	0	0	0				
20	0	0	0	0	0	0	1	1	1	0	0	0				
21	0	0	0	0	0	0	0	1	1	1	0	0				
22	0	0	0	0	0	0	0	0	1	1	0	0				
23	0	0	0	0	0	0	0	0	0	1	0	0				



Screenshots of Model 1 Implementation and Results

Model 2

The second objective of the project is achieved by model 2 and model 3. Model 2 maximizes the preferences of the student staff.

Decision variables:

p_{itw}: Preferences of student i to work or not on time t on the day w

a_{iw}: Number of hours student i is available on day w

$$a_{iw} = \sum_{t} x_{itw} \quad \forall i, w$$

where x_{itw} is the generated schedule for student i on day w.

If student i is unavailable on day w

then, a_{iw}=0

x_{itw}= 1, student i works at time t of the day w;0, otherwise

N=The last shift of any day w

Objective function:

$maximize \sum_i \sum_t \sum_w (p_{itw} \times x_{itw})$

The above function represents the objective of maximizing the preferences of the student staff.

Constraints:

1. This constraint is a schedule demand constraint

$$\sum_i x_{itw} = \sum_l n_{ltw} \quad \forall t, w \qquad (2.1)$$

2. This constraint tells the maximum Hours a student can work per week

$$\sum_t \sum_w x_{itw} \leq 20 \quad \forall i \qquad (2.2)$$

3. Integrality constraint: This constraint assigns x_{itw} as binary

$$x_{itw} = \text{binary} \qquad (2.3)$$

4. Time constraint: The maximum number of hours any student can work on a day should not exceed 5.

$$\sum_t x_{itw} \leq 5 \quad \forall i, w \qquad (2.4)$$

5. This constraint ensures that any student who works the last shift of a day is not assigned the first shift of the very next day

$$x_{iN(w-1)} + x_{i1w} \leq 1 \qquad (2.5)$$

where x_{i1w} is the 1st shift of any day and $x_{iN(w-1)}$ is the last shift of the previous day

6. The constraint shows that the total number of hours when student is unavailable. So,

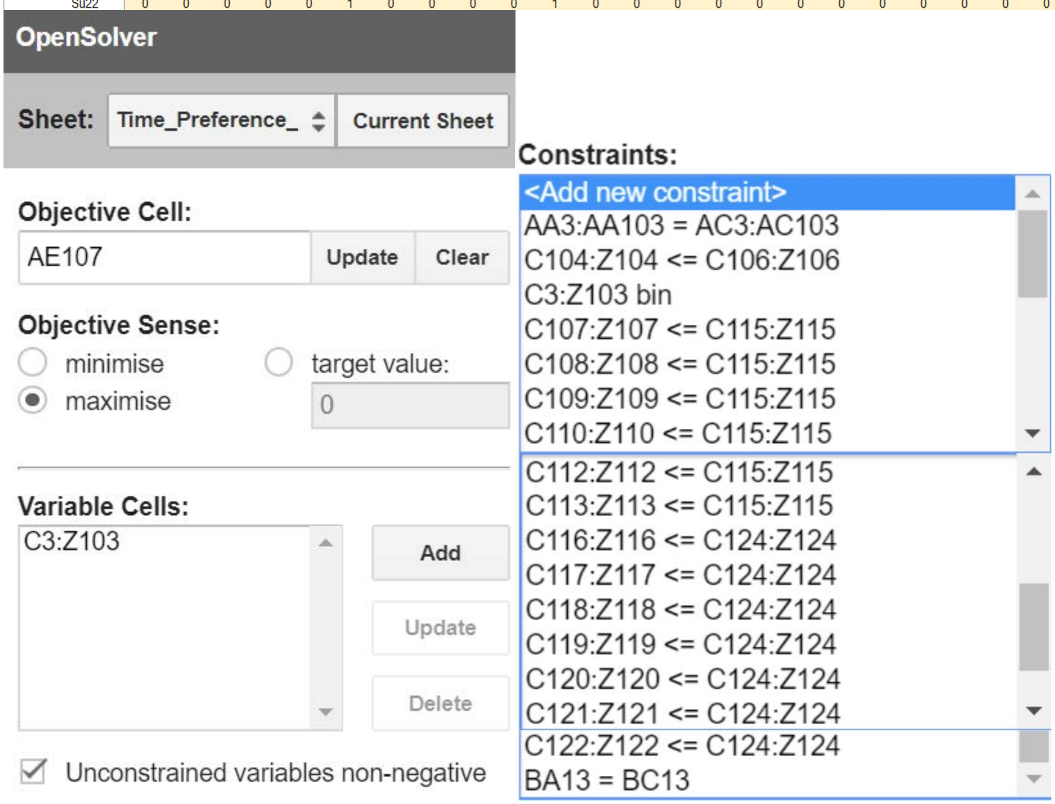
$$\sum_t x_{itw} = 0 \quad \forall i, w, a_{iw} = 0 \qquad (2.6)$$

Implementation & Results:

The above model is implemented by building the data based on the preferences of the student staff and meeting the demands of the library schedule. The first constraint that we put is that every student can work only 20 hours per week. The next constraint is that the maximum number of hours that a student can work per day cannot be more than 5 hours. And the third constraint states that the student who works the last shift of any day should not be assigned the first shift of the very next day.

The screenshot below shows that we have managed to maximize the preferences of the student staff by optimizing them efficiently. Hence the model resulted in total matching schedule and preference hours to be 335 hours and a match percentage of the preferred hours and generated hours of 86.34%

Decision Variables	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	x18	x19	x20	x21	x22	x23	x24			
Sa14	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	4	=	4
Sa15	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	3	=	3
Sa16	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	3	=	3
Sa17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	2	=	2
Sa18	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	2	=	2
	18	17	17	20	19	18	16	19	13	14	16	16	12	18	15	12	20	12	17	18	15	20	15	11			388
Constraints	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=			
	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20			
Sunday	0	0	0	3	3	5	3	0	0	0	5	0	0	0	0	0	0	4	0	4	0	5	0	0			
Monday	0	4	3	5	4	4	0	4	0	3	3	5	0	5	3	3	0	5	0	5	5	5	0	4			
Tuesday	5	0	5	5	4	4	0	5	5	4	5	3	5	5	0	0	5	3	0	0	0	5	0	0			
Wednesday	0	4	0	0	0	5	5	5	5	0	0	4	3	3	4	0	5	0	4	5	5	5	5	3			
Thursday	4	4	4	4	3	0	4	5	3	4	3	4	4	5	0	4	5	0	3	0	0	0	5	0			
Friday	4	5	5	3	5	0	0	0	0	0	0	0	0	0	5	5	0	0	5	4	5	0	5	4			
Saturday	5	0	0	0	0	0	4	0	0	3	0	0	0	0	3	0	5	0	5	0	0	0	0	0	Objective Function		
	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=		335	
	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	Efficiency	86.34	
Sun-Mon	0	1	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0			
Mon-Tue	0	0	0	1	0	1	0	1	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0			
Tue-Wed	0	1	0	0	1	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0			
Wed-Thu	0	0	0	0	0	1	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0			
Thu-Fri	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0			
Fri-Sat	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1			
Sat-Sun	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0			
	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=	<=			
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
Data																											
	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16	p17	p18	p19	p20	p21	p22	p23	p24			
Su12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0			
Su13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0			
Su14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0			
Su15	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0			
Su16	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0			
Su17	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
Su18	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
Su19	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0			
Su20	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0			
Su21	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0			
Su22	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0			



Student Schedule

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Sun	0	0	0	3	3	5	3	0	0	0	5	0	0	0	0	0	0	4	0	4	0	5	0	0
Mon	0	4	3	5	4	4	0	4	0	3	3	5	0	5	3	3	0	5	0	5	5	5	0	4
Tue	5	0	5	5	4	4	0	5	5	4	5	3	5	5	0	0	5	3	0	0	0	5	0	0
Wed	0	4	0	0	0	5	5	5	5	0	0	4	3	3	4	0	5	0	4	5	5	5	5	3
Thu	4	4	4	4	3	0	4	5	3	4	3	4	4	5	0	4	5	0	3	0	0	0	5	0
Fri	4	5	5	3	5	0	0	0	0	0	0	0	0	0	5	5	0	0	5	4	5	0	5	4
Sat	5	0	0	0	0	0	4	0	0	3	0	0	0	0	3	0	5	0	5	0	0	0	0	0

Screenshots of Model 2 Implementation and Results

(Student names are represented as A-X in the spreadsheet)

Model 3

The Model 3 is a different approach in achieving Objective 2 - maximizing the student preferences. While the second model explicitly attempted to maximize the total number of overlap between the preferred and generated schedule hours, this model aims to minimize the difference between the preferred hours and scheduled hours.

The model is more in alignment with the current practices in the library, of focusing on the total number of hours in a day rather than the exact hour. Thus, the timing constraint is removed; however, the unavailability and total hour preference per day is maintained.

Data:

1. The preference of the student, $\sum_t p_{itw} \quad \forall\, i, w$
2. The number of students required each hour on each day, calculated from model 1, n_{ltw}
3. The availability of the students, a_{iw}

Decision variables:

The sum of the number of hours each student works on each day of the week,

$\sum_t x_{itw}\ \forall i, w;\ d_{iw}^- \ ;\ \ d_{iw}^+$

d_{iw}^- and d_{iw}^+ are deviation constraints used in the model.

Objective function:

Minimize the sum of the deviation variables

Minimize z = $\sum_w \sum_t (d_{iw}^- + \ d_{iw}^+)$

Constraints:

1. The prepared schedule hours should be able to meet the demand for each hour, each day, as calculated from the previous models

   $$\sum_i \sum_t x_{itw} \geq \sum_t \sum_l n_{ltw}\ \forall w \qquad (3.1)$$

2. The maximum number of hours any student can work on a day should not exceed 5.

   $$\sum_t x_{itw} \ \leq 5 \quad \forall i, w \qquad (3.2)$$

3. If a student is unavailable on a particular day, which could be 2 days per week, the total number of hours on that day is nil.

$\sum_t x_{itw} \;\; =0 \qquad \forall a_{iw} = 0, i, w \qquad\qquad (3.3)$

4. No student should work for more than 20 hours in a week.

   $\sum_w \sum_t x_{itw} \leq \; 20 \qquad \forall i \qquad\qquad (3.4)$

5. As per the goal programming methodology, the sum of the differences between the Δ (Δ is the difference between preferred schedule and generated schedule) and the negative and positive deviation constraints should be equated to 0.

   $\sum_t \Delta - d_{iw}^{-} + d_{iw}^{+}] \quad \forall i, w \qquad\qquad (3.5)$

6. The number of students is an integer.

   $\sum_t x_{itw}$ is an integer $\qquad\qquad (3.6)$

Implementation and Results:

The total preferred hours for each week is calculated and set up in a table. The difference between that and the calculated hours are captured in the Δ table.

$$\Delta = \sum_t x_{itw} - \sum_t p_{itw} \;\; \forall\, i, w$$

Further, the deviation variable tables are created. Another table that equates the sum of the differences between the Δ and the negative and positive deviation constraints is created. The model had a total of 504 variables, including the deviation variables. Opensolver was used for solving as Microsoft Excel has a limitation of 200 variables. A screenshot of the model is given below and the complete model is available in the Appendix.

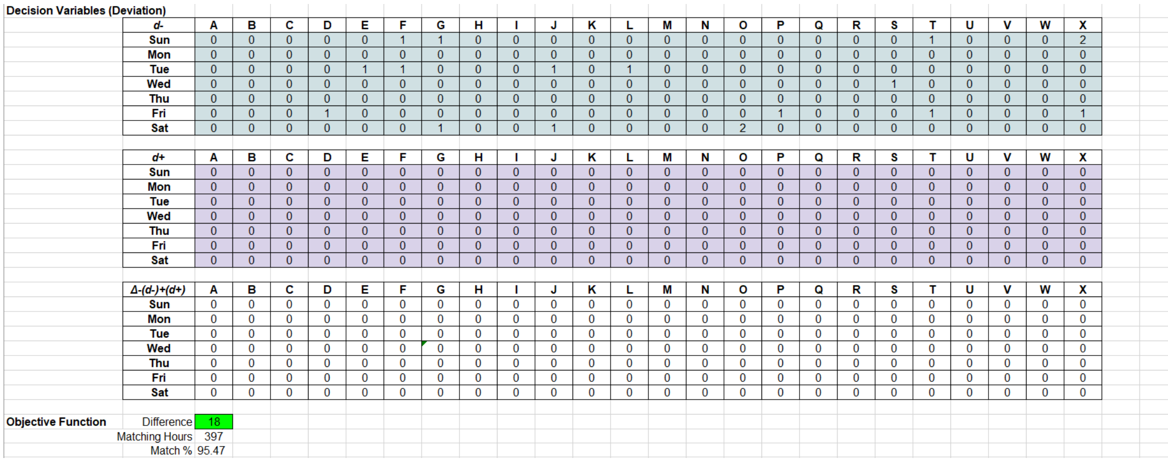
Minimize Schedule-Preference Difference

Decision Variables		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X			Constraints
	Sun	0	0	0	0	0	5	5	0	0	0	5	0	0	0	0	0	0	5	0	5	0	5	0	2	32	>=	32
	Mon	5	5	5	3	4	5	0	0	0	3	3	5	0	5	3	3	0	5	0	5	5	5	0	3	72	>=	70
	Tue	5	0	5	3	5	5	0	5	5	5	5	5	5	5	0	0	5	0	0	0	0	5	0	0	68	>=	68
	Wed	0	5	0	0	0	5	5	5	5	0	0	4	0	3	5	0	5	0	5	5	5	5	5	3	70	>=	70
	Thu	5	0	5	5	4	0	4	5	3	4	5	4	4	5	0	4	5	5	3	0	0	0	5	0	75	>=	68
	Fri	5	5	5	5	5	0	0	0	0	0	0	0	0	0	0	5	0	0	5	5	5	0	5	5	55	>=	55
	Sat	0	0	0	0	0	0	5	0	0	5	0	0	0	0	5	0	5	0	5	0	0	0	0	0	25	>=	25
																									5			
																							0	=	0			
Total Hours per Week		20	15	20	16	18	20	19	15	13	17	18	18	9	18	13	12	20	15	18	20	15	20	15	13		<=	20

Data

Preferred Hours		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
	Sun	0	0	0	0	0	4	4	0	0	0	5	0	0	0	0	0	0	5	0	4	0	5	0	0
	Mon	5	5	5	3	4	5	0	0	0	3	3	5	0	5	3	3	0	5	0	5	5	5	0	3
	Tue	5	0	5	3	4	4	0	5	5	4	5	4	5	5	0	0	5	0	0	0	0	5	0	0
	Wed	0	5	0	0	0	5	5	5	5	0	0	4	0	3	5	0	5	0	4	5	5	5	5	3
	Thu	5	0	5	5	4	0	4	5	3	4	5	4	4	5	0	4	5	5	3	0	0	0	5	0
	Fri	5	5	5	4	5	0	0	0	0	0	0	0	0	0	0	4	0	0	5	4	5	0	5	4
	Sat	0	0	0	0	0	0	4	0	0	4	0	0	0	0	3	0	5	0	5	0	0	0	0	0

Difference (calculated)	Δ	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
	Sun	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2
	Mon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Tue	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Wed	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Thu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Fri	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1
	Sat	0	0	0	0	0	0	1	0	0	1	0	0	0	0	2	0	0	0	0	0	0	0	0	0

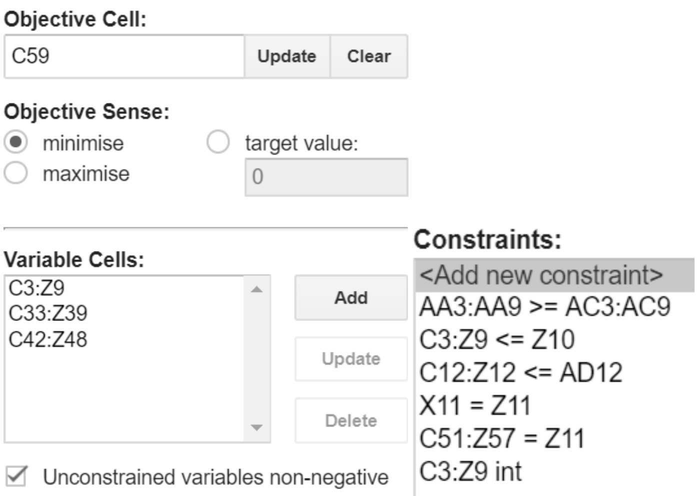

Decision Variables (Deviation)

d-	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Sun	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2
Mon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tue	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Wed	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Thu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fri	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1
Sat	0	0	0	0	0	0	1	0	0	1	0	0	0	0	2	0	0	0	0	0	0	0	0	0

d+	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Sun	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tue	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wed	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Thu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fri	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Sat	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Δ-(d-)+(d+)	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Sun	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Mon	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tue	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Wed	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Thu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fri	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Sat	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Objective Function	Difference	18
	Matching Hours	397
	Match %	95.47



Screenshots of Model 3 Implementation and Results
(Student names are represented as A-X in the spreadsheet)

The results gave the minimum difference between the preferred and generated schedule hours on all days as 18. This means 95.5% match between the generated and preferred schedule hours.

RESULT SUMMARY

The summary of the results are given below:

- The number of total hours for all the students previously were 406. It was reduced to 388 with Model 1 implementation. This is a reduction of 4.5%
- The match between the generated schedule time and preferred time of students for each day was maximized to 86.34% using Model 2. This means, 86.34% of the time, the student would be able to exactly work on the same day and same time of his/her preference. This condition is met while keeping the total number of hours at a minimum.

- The match between the generated schedule hours and preferred hours of students for each day of the week was maximized to 95.5%. For 95.5% of times, the demand request of days and number of hours per day of the student is met.
- At all times, the minimum requirement of students for duty is met.
- No student was assigned duty on any of his/her unavailable days.
- The model takes into consideration of all 24 students currently working in the PSU library circulation desk, their current schedule, availability, and preferences.
- The maximum limit of 20 hours per week for each student was never exceeded.
- The maximum hour per student per day was limited to 5 hours, thereby improving the current system that requires the mandatory break after every 5 hours.

CONCLUSION

In this project, we used integer linear programming and goal programming concepts to schedule the student staff in the PSU library. The aim to minimize the surplus hours worked by the students and maximize the student preferences were achieved. Open Solver made it possible to build and test the developed model and thus, the variable number limitation of Microsoft Excel was overcome.

Shift scheduling is important in any organization, as it has a direct effect on productivity, quality of service, and employee morale. With the implementation of linear programming, the project was able to achieve significant improvements over the current system used in the library.

The aim to develop a scalable model was nearly achieved as far as the current scheduling system in PSU library and available data are concerned. PSU library could use the model for the coming terms for minimizing the number of student hours without compromising much on the student preferences.

FUTURE WORK & RECOMMENDATIONS

1. The model implementation has been entirely done on the google spreadsheets. Developing a good front-end user interface would enable easy access for the library manager/staff to easily input data and generate schedule automatically.
2. The model can be extended to other departments such as OIT, PSU pantry, etc. as well for their student staff scheduling.
3. With changed data and constraints, the model can be used for scheduling non-student staffs at PSU library reference desk and other departments as well.
4. The model can be implemented using R or other tools that support linear programming.

REFERENCES

1. K.R. Baker., "Optimization Modeling With Spreadsheets"., 3rd edition.
2. C. T. Ragsdale., "Spreadsheet Modeling and Decision Analysis"., 5th edition.
3. https://support.office.com/en-us/article/define-and-solve-a-problem-by-using-solver-5d1a388f-079d-43ac-a7eb-f63e45925040
4. https://opensolver.org
5. A. J. Mason; "OpenSolver - An Open Source Add-in to Solve Linear and Integer Programmes in Excel"; Operations Research Proceedings, 2011; pp 401- 406
6. Oded Berman, Richard C. Larson, Edieal Pinker, "Scheduling Workforce and Workflow in a High Volume Factory"
7. M.N. Azaieza, S.S. Al Sharif, "A 0-1 goal programming model for nurse scheduling"

APPENDIX

The link to the Google spreadsheet where the models were implemented: https://docs.google.com/spreadsheets/d/1EGETSIttRFPV4nC9XJU8ICsXwJF2Wx70RctLDLp-JtE/edit?usp=sharing