CUỘC THI

TOÁN MÔ HÌNH 2024

VÒNG 1

02 - 28 tháng 07, 2024

Tên đội thi: Capybara
Ngày thực hiện: 11/07/2024

INDEX

1. PROBLEM DEFINITION

1.1. Given Data

- The data for the given problem is put into a tabular data frame of size 80x5, including 5 columns: ID, Name, Department, Role, Seniority.
- In which attributes will include:
  - ID: set of integers from 1 to 80.
  - 4 Departments: Internal Medicine, Surgery, Emergency, Clinical.
  - 3 Roles: Main Doctor (Bác sĩ cọc I), Assistant Doctor (Bác sĩ cọc II), Nurse (Điều dưỡng).
  - Seniority: set of integers running from 1 to 22.
- Number of Role in each Department:
  - 19 Main Doctor: 3 Internal Med, 6 Surgery, 6 Emergency, 4 Clinical
  - 17 Assistant Doctor: 2 Internal Med, 4 Surgery, 4 Emergency, 7 Clinical
  - 44 Nurse: 9 Internal Med, 11 Surgery, 13 Emergency, 11 Clinical
- Data about "shifts" will be added [1]:
  - Total number of working days: 61 days (31 days in August + 30 days in September).
  - 2 shifts per day: Day (8 AM - 8 PM), Night (8 PM - 8 AM).
  - Total number of shifts: 122 shifts.
- Data on "high seniority" will be added:
  - We will take the mean value of the "Seniority" column as the threshold for evaluating members with "high seniority".
  - Individuals with Seniority greater than 10.11 will be considered to have "high seniority".

1.2. Constraints and Objective

From the problem, we can come up with 3 constraints:

- Each shift must have 1 Main Doctor, 1 Assistant Doctor and 2 Nurses (Hard Constraint).
- Each shift should have enough members from 4 Departments (Soft Constraint).
- Each shift should have at least 1 senior member (Soft Constraint).

And ask for optimal solution for objective:

- The duty schedule needs to be reasonable in personnel rotation, no one has to be on duty too much or too little.

2. NOTATION

2.1. Sets

- ***E***: Set of employees $(e_1, e_2, e_3, ..., e_{|E|})$
- ***D***: Set of days $(d_1, d_2, d_3, ..., d_{|D|})$
- ***R***: Set of roles $(r_1, r_2, r_3, ..., r_{|R|})$
- ***P***: Set of departments $(p_1, p_2, p_3, ..., p_{|P|})$
- ***S***: Set of shifts $(s_1, s_2, s_3, ..., s_{|S|})$

2.2. Parameters

- p_e: Department of employee *e*
- r_e: Role of employee *e*
- n_r: Number of role *r* required per shift
- n_p: Number of department *p* required per shift
- sen_e: Seniority of employee *e*
- sen_{min}: Minimum required seniority for at least one employee per shift

2.3. Decision Variables

- $x_{e,d,s}$: Binary variable, equal to 1 if employee *e* is assigned to shift *s* on day *d*, 0 otherwise
- ns_e: Integer variable, number of shifts assigned to any employee
- ns_{min}: Integer variable, minimum number of shifts assigned to any employee
- ns_{max}: Integer variable, maximum number of shifts assigned to any employee

2.4. Constraints

- Role Coverage (Each shift must cover required number of roles):

$$\sum_{e \in E} x_{e,d,s} = n_{r_e} \text{ for all } r \in R,\ s \in S,\ d \in D$$

- Department Coverage (Each shift should cover required number of department):

$$\sum_{p \in P} \left(\sum_{e \in E \mid p_e = p} x_{e,d,s} \geq n_p \right) \text{ for all } s \in S,\ d \in D$$

- Senior Coverage (Each shift should cover required number of senior member):

$$\sum_{e \in E \mid sen_e \geq sen_{min}} x_{e,d,s} \geq 1 \text{ for all } s \in S,\ d \in D$$

3. MODELING

3.1. Add Constraints

To optimize the objective, we will add some “Hard Constraints”[2]:

- Each employee can be assigned to at most one shift per day:

$$\sum_{s \in S} x_{e,d,s} \leq 1 \text{ for all } e \in E,\ d \in D$$

- Each employee have to be assigned at least $ns_{min} = 4$ shifts:

$$ns_e \geq ns_{min} = 4$$

- No employee has a day shift right after a night shift (s_1 for 'day', s_2 for 'night'):

$$x_{e,d,s_2} + x_{e,d+1,s_1} \leq 1 \text{ for all } e \in E,\ d \in D\backslash\{|D|\}$$

- No employee has more than a 2 consecutive working days:

$$x_{e,d,s_1} + x_{e,d,s_2} + x_{e,d+1,s_1} + x_{e,d+1,s_2} + x_{e,d+2,s_1} + x_{e,d+2,s_2} \leq 2$$
$$\text{for all } e \in E,\ d \in D\backslash\{|D-1|, |D|\}$$

3.2. Objective Function

The problem request is to schedule the workforce so that no one has to be on duty too much or too little, which means that to minimize the difference in the number of shifts between employees.[4]

- Number of shifts per employee:

$$ns_{min} \leq ns_e \leq ns_{max} \text{ for all } e \in E$$

- Objective Function:

$$Min\ (ns_{max} - ns_{min})$$

Then because we have assigned $ns_{min} = 4$, our goal will be to minimize ns_{max}.

- Objective Function:

$$Min\ (ns_{max})$$

3.3. Evaluation Metrics

- Shift Variance:

$$\sigma^2 = \frac{1}{|E|} \sum_{e \in E} (ns_e - \mu)^2$$

$$\text{where } \mu = \frac{1}{|E|} \sum_{e \in E} ns_e$$

- Fairness Metric:

$$F = 1 - \frac{\sigma^2}{\mu^2}$$

- Role Coverage Correctness

- Department Diversity Average:

$$\overline{D} = \frac{1}{|D||S|} \sum_{d \in D,\ s \in S} |\{p \in P | \sum_{e \in E | p_e = p} x_{e,d,s} > 0\}|$$

- Senior Coverage:

$$\overline{S} = \frac{1}{|D||S|} \sum_{d \in D,\ s \in S} (\sum_{e \in E | sen_e > sen_{min}} x_{e,d,s} > 0)$$

3.4. Linear Programming:

We use PuLP's default solver to find an optimal or near-optimal solution satisfying all constraints.

PuLP library is an open source library in Python used to describe optimization problems as mathematical models. It is usually used for linear programming (LP) and mixed-integer linear programming (MILP).

We chose to use this library because of its convenience and simplicity. By using a math programming library in a programming language, we were able to take advantage of our own strengths in coding.

Based on the constraints set out in the previous section, we have converted those constraints into a programming language as follows:

- Define data objects.
- Initialize Model.
- Define Decision Variables: We construct the scheduling problem as a Minimize function.
- Define the Objective function: Minimize the max_shifts variable.
- Define the Constraints.
- Solve Model

4. RESULTS

4.1. Work Shift Schedule Demo

With programming power, we generate a schedule for 80 staffs in hospital in 61 days as a CSV file:

optimized_hospital_schedule.csv

1 to 50 of 488 entries

Day	Shift	Employee_ID	Name	Department	Role
2	night	1	Hường	Internal Medicine	Main Doctor
7	night	1	Hường	Internal Medicine	Main Doctor
30	day	1	Hường	Internal Medicine	Main Doctor
39	day	1	Hường	Internal Medicine	Main Doctor
51	night	1	Hường	Internal Medicine	Main Doctor
56	night	1	Hường	Internal Medicine	Main Doctor
3	day	2	Thư	Internal Medicine	Main Doctor
4	day	2	Thư	Internal Medicine	Main Doctor
13	day	2	Thư	Internal Medicine	Main Doctor
28	day	2	Thư	Internal Medicine	Main Doctor
35	day	2	Thư	Internal Medicine	Main Doctor
50	day	2	Thư	Internal Medicine	Main Doctor
54	day	2	Thư	Internal Medicine	Main Doctor
58	night	2	Thư	Internal Medicine	Main Doctor
2	day	3	Sơn	Internal Medicine	Main Doctor
13	night	3	Sơn	Internal Medicine	Main Doctor
25	night	3	Sơn	Internal Medicine	Main Doctor
27	day	3	Sơn	Internal Medicine	Main Doctor
32	day	3	Sơn	Internal Medicine	Main Doctor
40	night	3	Sơn	Internal Medicine	Main Doctor
50	night	3	Sơn	Internal Medicine	Main Doctor
59	day	3	Sơn	Internal Medicine	Main Doctor
10	night	4	Hưng	Internal Medicine	Assistant Doctor
15	night	4	Hưng	Internal Medicine	Assistant Doctor

4.2. Evaluation

Our evaluation is based on 9 different metrics:

1. Our priority - Correct Role Coverage: we need to ensure that each shift has enough 4 people: 1 main doctor, 1 assistant doctor, and 2 nurses. If this priority is not satisfied, our later metrics are meaningless
2. Shift coverage (%): we expect all shifts to have enough roles.
3. Max-Min shifts per Employee: this metric ensures that no employee is assigned too much or too few shifts.
4. Shift variance: Variance between a number of shifts between employees.
5. Fairness metrics
6. Average Department Diversity per Shift: Average number of departments per shift.
7. Senior Staff Coverage (%): Percentage of shifts that have at least 1 person with high seniority.
8. Number of Consecutive Night-Day shifts: as we mentioned above in section 3.5 when we set this as a hard constraint, we expect this number to be equal to 0.
9. Max Consecutive Working Days of any Employee: The maximum number of consecutive days an employee is assigned.

5. DISCUSSION

5.1 Efficiency

Based on the above metrics in section 4.2, we found that:

```
Schedule Evaluation:
Correct Role Coverage (PRIORITY): True
Total Shifts: 122.0
Expected Shifts: 122
Shift Coverage (%): 100.0
Max Shifts per Employee: 8
Min Shifts per Employee: 4
Average Shifts per Employee: 6.1
Shift Variance: 2.4962025316455696
Fairness Metric: 0.932915814790498
Average Department Diversity per Shift: 3.3278688524590163
Senior Staff Coverage (%): 98.36065573770492
Number of Consecutive Night-Day Shifts: 0
Max Consecutive Working Days of any Employee: 2
```

This result shows that our model is efficient in scheduling with 6 important parameters satisfied:

- Correct Role Coverage equals True, which ensures the prioritized hard constraint is satisfied.
- Shift Coverage equals 100% ensuring that all shifts in August and September are assigned.
- Average Department Diversity per Shift equals 3.32/4 showing that every shift will highly have members from at least 3 different departments.
- Senior Staff Coverage stays at 98.36%, which means that there are only around 2 over 122 shifts that do not have high seniority members.

- No employee has to work a morning shift consecutively with an evening shift from the previous day and no employee has to work over 2 days consecutively ensuring health care for employees and preventing burn-out.

Other optimization-based metrics also give quite good results as we reduce the difference between max shift and min shift to 4, Fairness metric gives another high result, up to 93% despite the fact that the distribution of personnel in each department and each role has a very large deviation.

5.2. Generalization
Based on our modeling formulation, our model can be proved to work with different datasets and different constraints' parameters.

5.3. Future Work
We can think of many potential areas for future work on this shift scheduling problem, such as:

- Multi-objective optimization: Consider additional objectives beyond just minimizing the difference between maximum and minimum shifts. For example, you could also try to minimize overtime costs or maximize employee preferences.
- Employee preferences: Include a system for employees to input their shift preferences, and incorporate these into the optimization model.
- Break scheduling: Extend the model to also schedule required breaks within shifts, ensuring labor law compliance.
- Rolling horizon planning: Implement a system that can update the schedule on an ongoing basis as new information becomes available, rather than creating a fixed schedule far in advance.
- Fairness metrics: Develop more sophisticated measures of fairness beyond just the difference between maximum and minimum shifts. This could include factors like distribution of weekend shifts or holiday work.

5.4. Conclusion
In conclusion, this shift scheduling optimization model can fairly distribute work shifts among employees while meeting operational requirements. By minimizing the difference between the maximum and minimum number of shifts assigned, and setting a minimum of 4 shifts per employee, the model promotes equitable workload distribution.
This approach not only ensures adequate staffing levels but also contributes to employee satisfaction by preventing overwork and underutilization. While the current model effectively addresses the core scheduling challenge, there remain opportunities for future enhancements. These could include incorporating employee preferences, skill-based assignments, and more complex fairness metrics. As workforce management continues to evolve, such optimization techniques will play an increasingly crucial role in balancing operational efficiency with employee well-being.

6. PREFERENCE

[1] R. Soto, B. Crawford, E. Monfroy, W. Palma, F. Paredes, Nurse and paramedic rostering with constraint programming: A case study, Romanian Journal of Information Science and Technology, 16(1), 52-64, 2013.
[2] Lizzy Augustine, Morgan Faer, Andreas Kavountzis, Reema Patel, A Brief Study of the Nurse Scheduling Problem (NSP), 2009.
[3] Operations Research II Group 4 Yijing Chen, Andrew Liu, Elizabeth Sciannella, Alice Zhang, A Comparison of Approaches to the Nurse Scheduling Problem, 2016.
[4] P. De Bruecker, J. Van den Bergh, J. Belien, E. Demeulemeester, Workforce planning incorporating skills: State of the art, European Journal of Operational Research, 243(1), 1-16, 2015.