ISE321 Notes on a Rotation Model

September 6, 2022

1 Basic Mathematical Notation

We define some mathematical notation that are commonly used in a mathematical optimization model.

- Set: a collection of elements – e.g., $I = \{1, 3, 4, 6, 7\}$
- "'$\in$" means "in" – e.g., if $I = \{1, 3, 4, 6, 7\}$, we have $3 \in I$ but $2 \notin I$
- "$\forall$" means "for all" – e.g., "$\forall i \in I$" means for all elements i in the set I
- "$\sum_i$" means "sum over i" – e.g., $\sum_{i=1}^{5} a_i = a_1 + a_2 + \cdots + a_5$
- "$I \times J$" for sets I and J means "cross over" – e.g., if $I = \{1, 4\}$ and $J = \{2, 6\}$, then $I \times J = \{(1, 2), (1, 6), (4, 2), (4, 6)\}$

2 Rotation Model

A full set of notation we used is given in Table 1. In the rotation problem, we consider the assignment of a set of residents to different rotations and blocks. We use P to denote the set of residents, R to denote the set of rotations, and B to denote the set of blocks. In the rotation model, we decide the assignment of the residents to rotations and blocks. Thus, we define the *decision variables* of the model as

$$x_{p,r,b} = \begin{cases} 1 & \text{if resident } p \text{ is assigned to rotation } r \text{ and block } b, \\ 0 & \text{otherwise,} \end{cases}$$

for all $p \in P$, $r \in R$, and $b \in B$. Note that the (mathematical) decision variables $x_{p,r,b}$ for all combinations $(p, r, b) \in P \times R \times B$ correspond to a schedule in the real life.

To ensure that a schedule is *implementable*, we need to impose some *constraints*. We can derive mathematical constraints that correspond to the physical constraints in the real world.

Example: Constraints Set I

In practice, we may require that for **each** resident $p \in P$ and **each** block $b \in B$, the number of rotations that one can perform at a time is *exact one*. We can formulate the following constraint:

$$\sum_{r\in R} x_{p,r,b} = 1, \quad \forall p \in P,\, b \in B. \tag{2.1}$$

First, the "**each**" components are represented using "$\forall p \in P$, $b \in B$", or equivalently, "$\forall (p,b) \in P \times B$". Now, **for a fixed** $p \in P$ **and** $b \in B$, the constraint reads $\sum_{r\in R} x_{p,r,b} = 1$. Recall that $x_{p,r,b} \in \{0,1\}$ (i.e., taking value 0 or 1 only), and it takes value 1 only when resident p is assigned to rotation r and block b. Therefore, we can interpret the summation $\sum_{r\in R} x_{p,r,b}$ as the **number of rotations** that resident p at block b performs. Therefore, the mathematical constraints (2.1) correspond to our requirements in the physical world.

Example: Constraints Set II

Next, suppose that for **each** resident $p \in P$ and **each** rotation $r \in R$, the number of blocks assigned to him/her should be at least $c^{\min}_{p,r}$ but at most $c^{\max}_{p,r}$. We can formulate the following constraint:

$$c^{\min}_{p,r} \le \sum_{b\in B} x_{p,r,b} \le c^{\max}_{p,r}, \quad \forall p \in P,\, r \in R. \tag{2.2}$$

Again, the "**each**" components are represented using "$\forall p \in P$, $r \in R$", or equivalently, "$\forall (p,r) \in P \times R$". Now, **for a fixed** $p \in P$ **and** $r \in R$, the constraint reads $c^{\min}_{p,r} \le \sum_{b\in B} x_{p,r,b} \le c^{\max}_{p,r}$. Using a similar logic in the previous example, we can interpret the summation $\sum_{b\in B} x_{p,r,b}$ as the **number of blocks** that resident p performs rotation r. Therefore, the mathematical constraints (2.2) correspond to our requirements in the physical world.

Exercise I

Consider the following physical constraint. For **each** rotation $r \in R$ and **each** block $b \in B$, the number of residents assigned should be at least $p^{\min}_r$ but at most $p^{\max}_r$. (Note that, in our case, the minimum and maximum number of residents depend on the rotation only). **Formulate the corresponding mathematical constraints; see model** (2.4) **for the solution.**

Exercise II

Consider the following physical constraint (e.g., due to some educational requirement). Let A_{pri} be the set of combinations $(p,r,b) \in P \times R \times B$ that we call priority assignments. These assignments **must be satisfied** in the output schedule. **Formulate the corresponding mathematical constraints.**

Despite the fact that we have formulated different constraints, we still need something to *optimize*. That is, we want to construct the *objective function* of our rotation model. In the current formulation, we can consider the following objective. Let A_{pref} be the set of combinations $(p, r, b) \in P \times R \times B$ that we call preference assignments. It is good that the output schedule can fulfill these assignments (or part of them), but they are *not necessary*. Thus, we may want to *maximize* the total number of preference assignments while keeping all other constraints. This gives rise to the following objective of the rotation model:

$$\text{maximize} \quad \sum_{(p,r,b)\in A_{\text{pref}}} x_{p,r,b}. \tag{2.3}$$

Note that the summation $\sum_{(p,r,b)\in A_{\text{pref}}} x_{p,r,b}$ counts **the number of preference assignments** in the schedule. Therefore, in the rotation model, we maximize this summation, which is exactly our goal.

Finally, we can combine our objective function (2.3) with constraints (2.1)–(2.2) and several other additional constraints. Define the following sets.

- A_{pri}: set of priority assignments $(p, r, b) \in P \times R \times B$ that must be satisfied
- A_{imp}: set of impossible assignments $(p, r, b) \in P \times R \times B$ that must be excluded
- B_{gone}: set of combinations $(p, b) \in B_{\text{gone}}$ that resident p has conflict with block b, i.e., no assignments are possible with such a combination of (p, b)
- C^1_{imp}: set of combinations $(p, r) \in C^1_{\text{imp}}$ that resident p cannot perform rotation r
- C^2_{imp}: set of combinations $(r, b) \in C^2_{\text{imp}}$ that rotation r cannot be performed in block b
- P_{all}: set of all-year residents
- R_{must}: set of must-do rotations (i.e., at least once) for all-year residents

With this notation, we can formulate our first rotation model as:

$$\begin{aligned}
\underset{x}{\text{maximize}} \quad & \sum_{(p,r,b)\in A_{\text{pref}}} x_{p,r,b} && \text{(2.4a)}\\
\text{subject to} \quad & \sum_{r\in R} x_{p,r,b} = 1, \quad \forall p \in P,\, b \in B, && \text{(2.4b)}\\
& c^{\min}_{p,r} \le \sum_{b\in B} x_{p,r,b} \le c^{\max}_{p,r}, \quad \forall p \in P,\, r \in R, && \text{(2.4c)}\\
& p^{\min}_{r} \le \sum_{p\in P} x_{p,r,b} \le p^{\max}_{r}, \quad \forall r \in R,\, b \in B, \quad \text{(Exercise I)} && \text{(2.4d)}\\
& x_{p,r,b} = 1, \quad \forall (p,r,b) \in A_{\text{pri}}, \quad \text{(Exercise II)} && \text{(2.4e)}\\
& x_{p,r,b} = 0, \quad \forall (p,r,b) \in A_{\text{imp}}, && \text{(2.4f)}\\
& x_{p,r,b} = 0, \quad \forall r \in R,\, (p,b) \in B_{\text{gone}}, && \text{(2.4g)}\\
& x_{p,r,b} = 0, \quad \forall b \in B,\, (p,r) \in C^1_{\text{imp}}, && \text{(2.4h)}\\
& x_{p,r,b} = 0, \quad \forall p \in P,\, (r,b) \in C^2_{\text{imp}}, && \text{(2.4i)}
\end{aligned}$$

$$\sum_{b \in B} x_{p,r,b} \geq 1, \quad \forall p \in P_{\text{all}},\, r \in R_{\text{must}}, \tag{2.4j}$$

$$x_{p,r,b} \in \{0, 1\}, \quad \forall p \in P,\, r \in R,\, b \in B. \tag{2.4k}$$

Exercise III

Give the interpretations/physical meanings of constraints (2.4f)–(2.4j).

3 Food for Thought

a. **Busy rotations.** Let R_{busy} be the set of busy rotations (e.g., heavy workloads). Consider the following physical constraints.

   i. B_{vac} is a set of combinations $(p, b) \in B_{\text{vac}}$ that resident p is on vacations on block b. Hospital cannot assign residents to busy rotations if residents are on vacation. How can we add constraint(s) to satisfy this requirement?

   ii. It is *not preferable* that a resident is assigned to two busy rotations in consecutive blocks. How can we modify (2.4) to include such a consideration?

b. **Interface with solvers.** Suppose we can solve the optimization by some back-end solver. What kinds of interface we can use for effective communication between users and the optimization model (both inputs and outpus)?

c. **Other practical constraints.** Can you think of any other reasonable/desirable constraints for the rotation model? If yes, how can you incorporate them into model (2.4)?

A Notation Summary

Table 1: Notation

Indices	
p	index of residents, $p \in P$
r	index of rotations, $r \in R$
b	index of blocks, $b \in B$
Parameters and sets	
P	set of residents
R	set of rotations
B	set of blocks
A_{pri}	set of priority assignments
A_{pref}	set of preference assignments
A_{imp}	set of impossible assignments
B_{gone}	set of (p, b) that a resident p is in conflict with block b
C^1_{imp}	set of (p, r) that resident p cannot perform rotation r
C^2_{imp}	set of (r, b) that rotation r cannot be performed in block b
P_{all}	set of all-year residents
R_{must}	set of rotations that all-year residents must do
$p^{\min}_r$	minimum number of residents in rotation r
$p^{\max}_r$	maximum number of residents in rotation r
$c^{\min}_{p,r}$	minimum number of blocks that resident p performs rotation r
$c^{\max}_{p,r}$	maximum number of blocks that resident p performs rotation r
Variables	
$x_{p,r,b}$	1 if resident p is assigned to rotation r and block b, and 0 otherwise