Brief Description of the Assignment Problem and its MILP Model

1 Introduction

The objective of this work is to solve, using integer linear programming, a problem in which crews of workers must be assigned to different tasks in order to maximize the obtained benefit, under various constraints detailed below.

2 Problem Description

This is a typical example of a task assignment problem using MILP. It involves a weekly planning of work crews to fulfill service orders (installations, inspections, etc.). Each order has an associated benefit if completed and requires a fixed number of workers. Orders must be scheduled in a single shift, and each worker can be assigned to multiple orders with several restrictions:

- Workers are paid according to the number of tasks performed, with marginal costs increasing in bands.
- Each day has 5 shifts, and the planning horizon is one week (6 days).
- Workers cannot work more than 5 days or multiple shifts on the same day.
- Orders may have precedence constraints (some must be done after others).
- Some workers may be in conflict and ideally should not be assigned to the same task.
- Some tasks are considered repetitive and ideally should not be assigned to the same worker.

The objective function balances the trade-off between maximizing the benefit of completed tasks and minimizing the cost of worker assignments, while optionally considering penalties for soft constraints.

3 Variables used

W_{xj} # hours worked by worker j in wage band x.

$$Y_{xj} \begin{cases} 1 & \text{if worker } j \text{ works in wage band } x \\ 0 & \text{otherwise} \end{cases}$$

$$O_{ik} \begin{cases} 1 & \text{if order } i \text{ is carried out in shift } k \\ 0 & \text{otherwise} \end{cases}$$

A_{ij} $\begin{cases} 1 & \text{if worker } j \text{ performs task } i \\ 0 & \text{otherwise} \end{cases}$

X_{kj} $\begin{cases} 1 & \text{if worker } j \text{ works in shift } k \\ 0 & \text{otherwise} \end{cases}$

V_{dj} $\begin{cases} 1 & \text{if worker } j \text{ works on day } d \\ 0 & \text{otherwise} \end{cases}$

δ_{ikj} $\begin{cases} 1 & \text{if worker } j \text{ performs task } i \text{ in shift } k \\ 0 & \text{otherwise} \end{cases}$

max_work number of shifts worked by the worker with the *most* assigned shifts.

min_work number of shifts worked by the worker with the *fewest* assigned shifts.

H_i $\begin{cases} 1 & \text{if task } i \text{ is performed} \\ 0 & \text{otherwise} \end{cases}$

$\lambda_{ijj'}$ $\begin{cases} 1 & \text{if the workers in bad relationship } (j, j') \text{ carry out task } i \\ 0 & \text{otherwise} \end{cases}$

$\theta_{ii'j}$ $\begin{cases} 1 & \text{if the repetitive tasks } (i, i') \text{ are performed by worker } j \\ 0 & \text{otherwise} \end{cases}$

$$\sum_{i=0}^{O} \sum_{k=1}^{30} \text{benefit}[i] \cdot H_i - \sum_{j=0}^{T} (1000W_{1j} + 1200W_{2j} + 1400W_{3j} + 1500W_{4j}) - \sum_{i,j,j'} \lambda_{ijj'} \cdot c_\lambda - \sum_{i,i',j} \theta_{ii'j} \cdot c_\theta$$

Here, c_λ and c_θ represent the penalty weights for assigning conflicting workers and repetitive tasks, respectively.

4 Constraints

In order to satisfy the conditions in the statement and to relate the variables used in solving the problem, the following constraints were defined. For readability, the constraints modelling the desirable conditions are also included.

Henceforth we make the following abuse of notation: when we refer to index i, we mean the *id* of task i.[1]

1. The sum over the orders performed by each worker must equal the sum of the tasks performed in each wage band:
$$\sum_{i=0}^{O} A_{ij} = \sum_{x=1}^{4} W_{xj} \quad \forall\, 0 \leq j \leq T. \tag{1}$$

[1] The value M corresponds to 10^9.

2. Workers' pay is modelled as a piecewise-linear function of the number of shifts worked in each band. Four bands are considered: $[1, 5]$, $[6, 10]$, $[11, 15]$, and $[16, +\infty)$. Each Y_{xj} caps its band once the maximum number of tasks for the corresponding W_{xj} is reached (hours worked by worker j in band x). Notice that once band4 is reached a worker may work any number of hours.

$$5Y_{1j} \leq W_{1j} \leq 5 \quad \forall\, j \tag{2}$$

$$5Y_{2j} \leq W_{2j} \leq 5Y_{1j} \quad \forall\, j \tag{3}$$

$$5Y_{3j} \leq W_{3j} \leq 5Y_{2j} \quad \forall\, j \tag{4}$$

$$5Y_{4j} \leq W_{4j} \leq MY_{3j} \quad \forall\, j \tag{5}$$

3. A worker cannot perform different tasks in the same shift:

$$X_{kj} \leq 1 \quad \forall\, 1 \leq k \leq 30,\ 0 \leq j \leq T. \tag{6}$$

4. No worker may work all 6 days of the schedule:

$$\sum_{d=1}^{5} V_{dj} \leq 5 \quad \forall\, j. \tag{7}$$

5. No worker may work all 5 shifts of a day:

$$\sum_{k=d5+1}^{d5+5} X_{kj} \leq 4 \quad \forall\, 0 \leq d \leq 5,\ 0 \leq j \leq T. \tag{8}$$

6. Linking variables V and X:

$$V_{dj} \leq \sum_{k=d5+1}^{d5+5} X_{kj} \leq V_{dj} M \quad \forall\, d, j. \tag{9}$$

7. The difference between the worker with the most hours and the least may not exceed 8:

$$\text{max\_work} \geq \sum_{i=1}^{O} A_{ij} \quad \forall\, j \tag{10}$$

$$\text{min\_work} \leq \sum_{i=1}^{O} A_{ij} \quad \forall\, j \tag{11}$$

$$\text{max\_work} - \text{min\_work} \leq 8. \tag{12}$$

8. Orders are completed in a single shift:

$$\sum_{k=1}^{30} O_{ik} \leq 1 \quad \forall\, 0 \leq i \leq O. \tag{13}$$

9. At most one task is solved per shift:

$$\sum_{i=1}^{O} O_{ik} \leq 1 \quad \forall\, 1 \leq k \leq 30. \tag{14}$$

10. Some pairs of conflicting orders (i, i') cannot be solved consecutively by any worker:

$$\delta_{ikj} \leq O_{ik} \qquad \forall\, i, k, j \qquad (15)$$
$$\delta_{ikj} \leq A_{ij} \qquad \forall\, i, k, j \qquad (16)$$
$$\delta_{ikj} \leq X_{kj} \qquad \forall\, i, k, j \qquad (17)$$
$$\delta_{ikj} \geq O_{ik} + A_{ij} + X_{kj} - 2 \qquad \forall\, i, k, j \qquad (18)$$
$$\delta_{ikj} \geq O_{ik} + A_{ij} + X_{kj} - 2 \qquad \forall\, i, k, j \qquad (19)$$
$$\delta_{ikj} + O_{i'k} + A_{i'j} + X_{k+1\,j} \leq 3 \qquad \forall\, i, k, j. \qquad (20)$$

11. If a task i precedes task i', then i' must be carried out in the next shift of the same day. It is necessary to forbid performing i' in the first shift of a day:

$$O_{i'k} \leq O_{ik} \qquad \forall\, i, i', k \mid (i, i') \in \text{precedence} \qquad (21)$$
$$O_{i'k} = 0 \qquad \forall\, i,\ k = c5 + 1,\ 0 \leq c \leq 5. \qquad (22)$$

12. δ_{ikj} links X_{kj}, O_{ik} and A_{ij}:

$$3\delta_{ikj} \leq X_{kj} + O_{ik} + A_{ij} \qquad \forall\, i, k, j \qquad (23)$$
$$X_{kj} + O_{ik} + A_{ij} \leq \delta_{ikj} + 2 \qquad \forall\, i, k, j. \qquad (24)$$

13. Some workers conflict, so it is desirable that they are not assigned to the same order:

$$\lambda_{ijj'} \leq A_{ij'},\ \lambda_{ijj'} \leq A_{ij} \qquad \forall\, i, j, j' | \,(j, j') \in \text{worker conflicts} \qquad (25)$$
$$\lambda_{ijj'} \geq A_{ij'} + A_{ij} - 1 \qquad \forall\, i, j, j' | \,(j, j') \in \text{worker conflicts}. \qquad (26)$$

14. Some pairs of tasks are repetitive, so it is desirable that the same worker is not assigned to both:

$$\theta_{ii'j} \leq A_{ij},\ \theta_{ii'j} \leq A_{i'j} \quad \forall\, j, i, i' | \,(i, i') \in \text{repetitive}. \qquad (27)$$

15. Integer variables:

$$W_{xj},\ \text{max\_work},\ \text{min\_work} \in \mathbb{Z}_{\geq 0}. \qquad (28)$$

16. Binary variables:

$$Y_{xj}, O_{ik}, A_{ij}, X_{kj}, V_{dj}, \delta_{ikj}, \lambda_{ijj'}, \theta_{ii'j}, H_i \in \{0, 1\}. \qquad (29)$$

17. Relationship between tasks performed in a certain shift and tasks performed overall:

$$H_i = \sum_{k=1}^{30} O_{ik}. \qquad (30)$$

18. If an order is carried out, exactly the required number of workers must be assigned to it:

$$H_i = \sum_{j=1}^{T} A_{ij}. \qquad (31)$$