



# Entrega 1 Optimización

«*Proyecto Semestral*»

Bruno Morici: 202373555-8

Martín Ignacio Aranda Hernández: 202373021-1

Cristóbal Alonso Martínez Pereira: 202373555-8

Juan Pablo Fuenzalida Fredes: 202373102-1

Segundo Semestre 2025

## Índice

|                             |   |
|-----------------------------|---|
| 1. Modelo Matemático        | 2 |
| 2. Generador de Instancias  | 4 |
| 3. Análisis de Factibilidad | 6 |

# 1. Modelo Matemático

## 1.1. Conjuntos

Para resolver este problema de optimización, es necesario definir los conjuntos numéricos que faciliten su comprensión, modelación y resolución. A continuación, se presentan los conjuntos identificados:

- $I = \{1, 2, \dots, N\}$ : Conjunto entero de trabajadores, siendo  $N$  el número total de trabajadores y con  $i \in I$ .
- $D = \{1, 2, \dots, H\}$ : Conjunto entero de días, siendo  $H$  el número total de días y con  $d \in D$ .
- $S = \{1, 2, 3\}$ : Conjunto entero de turnos a trabajar, siendo 1 el turno de la mañana, 2 el de la tarde y 3 el de la noche, y con  $s \in S$ .
- $T = \{1, 2, \dots, \lceil H/7 \rceil\}$ : Conjunto entero de semanas, con  $t \in T$ .

## 1.2. Variables

Una vez definidos los conjuntos necesarios, procede la definición de las variables que operarán sobre ellos, las cuales son fundamentales para la modelación y resolución del problema. Estas variables son las siguientes:

- $X_{i,d,s} = \begin{cases} 1, & \text{Se asigna al trabajador } i \text{ en el día } d \text{ en la turno } s. \\ 0, & \text{etoc.} \end{cases}$
- $Y_{i,t} = \begin{cases} 1, & \text{El trabajador } i \text{ trabajó el fin de semana } t. \\ 0, & \text{etoc.} \end{cases}$

## 1.3. Parámetros

Para la construcción de la función objetivo y de las restricciones planteadas en el problema, es necesario identificar y definir los parámetros proporcionados en el enunciado, los cuales son fundamentales para su correcta resolución. Estos son:

- $G_{i,d,s} = \{0, 1, \dots, 10\}$ : La disposición del trabajador  $i$  para el día  $d$  en el turno  $s$ .
- $R_{d,s} = \{R \in \mathbb{N}\}$ : La demanda de trabajadores el día  $d$  en el turno  $s$ .

## 1.4. Función Objetivo

Una vez identificados los conjuntos, variables y parámetros, corresponde definir la función objetivo que operará sobre nuestro modelo matemático. Esta es:

$$\text{máx } z = \sum_{d \in D} \sum_{i \in I} \sum_{s \in S} G_{i,d,s} \cdot X_{i,d,s}$$

De forma resumida, la función objetivo busca maximizar la asignación de los trabajadores a sus turnos, utilizando la variable binaria  $X_{i,d,s}$  y el parámetro  $G_{i,d,s}$ . De esta manera, se busca optimizar su desempeño y la calidad de atención brindada a los clientes.

## 1.5. Restricciones

Finalmente, nuestro modelo matemático requiere de un conjunto de restricciones que delimiten la zona factible de la función objetivo. Para ello, se identifican las siguientes restricciones a partir del enunciado, necesarias para obtener una solución óptima del problema:

1. No más de 2 turnos al día por trabajador:

$$\sum_{s=1}^3 X_{i,d,s} \leq 2 : \forall i \in I, \forall d \in D$$

2. Se prohíbe turno de noche luego turno de mañana del día siguiente por trabajador:

$$X_{i,d,3} + X_{i,d+1,1} \leq 1 : \forall i \in I, \forall d \in \{1, 2, \dots, H-1\}$$

3. Se debe cumplir con la demanda de trabajadores cada día:

$$\sum_{i \in I} X_{i,d,s} = R_{d,s} : \forall s \in S, \forall d \in D$$

4. No se puede trabajar 3 fin de semanas seguidos por trabajador:

$$X_{i,6,t,s} + X_{i,7,t,s} \geq Y_{i,t} : \forall i \in I, \forall s \in S, \forall t \in T$$

$$Y_{i,t} + Y_{i,t+1} + Y_{i,t+2} \leq 2 : \forall i \in I, \forall t \in \{1, 2, \dots, \lceil H/7 \rceil - 2\}$$

## 2. Generador de Instancias

Para nuestro generador de instancias, en el cual probaremos el modelo matemático de optimización, se decidió utilizar el lenguaje Python. En primer lugar, se definió la función `generar_instancia`, que utiliza la librería `random` para generar de forma aleatoria los arreglos de disposición y demanda, según el tamaño de la instancia. Posteriormente, la función escribe el archivo `.dzn` correspondiente a cada instancia, dándole el formato necesario para su correcto funcionamiento:

```
1 import random
2 def generar_instancia(nombre_archivo, N, H, T, min_demand, max_demand):
3     # Generar arreglo de disposicion aleatoria
4     avail = [[[random.randint(0, 10) for s in range(T)]
5                for d in range(H)]
6                for i in range(N)]
7     # Generar arreglo de demanda aleatoria
8     r = [[random.randint(min_demand, max_demand) for s in range(T)]
9           for d in range(H)]
10    with open(nombre_archivo, "w") as f:
11        # N y H
12        f.write(f"int: N = {N};\n")
13        f.write(f"int: H = {H};\n")
14        f.write(f"int: T = {T};\n\n")
15        # Escribimos las disposiciones en el archivo
16        f.write(f"array[1..N, 1..H, 1..T] of int: avail = array3d(1..N, 1..H, 1..T
17    , [")
18        valores_avail = []
19        for i in range(N):
20            for d in range(H):
21                for s in range(T):
22                    valores_avail.append(str(avail[i][d][s]))
23        f.write(", ".join(valores_avail))
24        f.write("]);\n\n")
25        # Escribimos las demandas en el archivo
26        f.write(f"array[1..H, 1..T] of int: r = array2d(1..H, 1..T, [")
27        valores_r = []
28        for d in range(H):
29            for s in range(T):
30                valores_r.append(str(r[d][s]))
31        f.write(", ".join(valores_r))
32        f.write("]);\n")
33    print(f"Archivo {nombre_archivo} generado con N={N}, H={H}, T={T}")
```

Además, se pre-generaron las configuraciones de las instancias indicadas en el enunciado, creando un diccionario de diccionarios que almacena las configuraciones solicitadas para los distintos tamaños, pequeñas, medianas y grandes:

```
1 # --- Generamos diversas instancias de la mas simple a la mas compleja ---
2 ### Agregar mas instancias si se desea
3 instancias = {
4     # Pequeñas
5     "instancia_1.dzn": {
6         "N": 3,
7         "H": 1,
8         "T": 3,
```

```
9         "min_demand": 1,  
10        "max_demand": 1  
11    },  
12    # Medianas  
13    "instancia_2.dzn": {  
14        "N": 10,  
15        "H": 30,  
16        "T": 3,  
17        "min_demand": 1,  
18        "max_demand": 3  
19    },  
20    # Grandes  
21    "instancia_3.dzn": {  
22        "N": 90,  
23        "H": 30,  
24        "T": 3,  
25        "min_demand": 1,  
26        "max_demand": 3  
27    },  
28 }
```

Finalmente, se recorre el diccionario instancias, aplicando la función previamente definida generar\_instancia y entregando como argumentos los parámetros contenidos en cada configuración:

```
1 # --- Generar las instancias ---  
2 for nombre_archivo, params in instancias.items():  
3     generar_instancia(  
4         "instancias/"+nombre_archivo,  
5         params["N"],  
6         params["H"],  
7         params["T"],  
8         params["min_demand"],  
9         params["max_demand"]  
10    )
```



---

### 3. Análisis de Factibilidad