# Emergencia en NÃŋger

Grupo 14: Bianca Dufour & William HedÃĺn

May 24, 2015

Se trata del problema de determinar el esquema oÌĄptimo de ayudar la emergencia en NÃŋger en un dia.

## 1 Conjuntos

$$\begin{aligned} i, j &: \text{Ciudad}, \quad i = 1, \ldots, 7 \\ k &: \text{Tipo de vehiculo}, \quad j = 1, 2 \end{aligned}$$

## 2 ParÃąmetros

$$\begin{aligned} av_i &: \text{Ayuda disponible en ciudad i} \\ d_i &: \text{Demanda de ayuda en ciudad j} \\ cota_k &: \text{Cota superior de k} \\ cap_k &: \text{Capacidad de ayuda de vehiculo k} \\ velv_k &: \text{Velocidad vehÃŋculo k} \\ cf_k &: \text{Un coste fijo para la conducciÃşn con el coche k por kilometro.} \\ cv &: \text{Un coste fijo para mover una unidad de carga por kilometro.} \\ velc_{i,j} &: \text{Velocidad mÃąximo en el camino entre ciudad i y j.} \\ vav_{k,i} &: \text{NÃžmero de vehiculos k disponibles en ciudad i.} \\ dist_{i,j} &: \text{Distancia entre la ciudad i y j en kilometros.} \\ budget &: \text{El coste total no puede superar el presupuesto.} \\ qglobal &: \text{Este dia solo podemos enviar una carga total de qglobal.} \end{aligned}$$

## 3 Variables

$X_{i,j,k}$ : NÃžmero de vehÃŋculos k que van entre ciudad i y j.

$carga_{i,j,k}$ : Cantidad de carga que van entre ciudad i y j con vehÃŋculo k.

$$Y_{i,j,k} = \begin{cases} 1 & \text{si vehÃŋculo tipo k va de ciudad i a j.} \\ 0 & \text{si no} \end{cases}$$

$load_i$ : Cantidad de carga que se queda en ciudad i.

$Time_i$ : Tiempo en llegar a ciudad i.

$Coste$ : Coste total del ayuda a NÃŋger.

$Equidad$ : La carga que se queda en ciudad i dividido por la demanda de ciudad i.

$Tiempo$ : El tiempo total del ayuda a NÃŋger.

## 4 Modelo

Queremos hacer tres cosas en este modelo:

- Minimizar el coste del ayuda total.
- Maximizar la equidad entre Agadez y Zinder.
- Minimizar el tiempo de hacer el operaciÃşn.

Vamos a tratarlas una a una, y despuÃľs resolvemos como un problema multi-objectivo por metas.

$$
\begin{aligned}
\min Coste &= \sum_{i,j,k \mid dist_{i,j} > 0} dist_{i,j} \cdot (2 \cdot X_{i,j,k} \cdot cf_k + cv \cdot carga_{i,j,k}) \\
\max Equidad &\leq \frac{load_i}{d_i}, \quad \forall i \mid d_i > 0 \\
\min Tiempo &\geq Time_i, \quad \forall i \mid d_i > 0
\end{aligned}
$$

restricciÃşnes:

$$
\begin{aligned}
\forall j, \quad & \sum_{i,k \mid dist_{i,j} > 0} carga_{i,j,k} + av_j = \sum_{i,k \mid dist_{j,i} > 0} carga_{j,i,k} + load_j \\
\forall j, k, \quad & \sum_{i \mid dist_{i,j} > 0} X_{i,j,k} + vav_{k,j} \geq \sum_{i \mid dist_{j,i} > 0} X_{j,i,k} \\
\forall j, \quad & load_j \leq d_j + av_j \\
\forall j \mid d_j > 0, \quad & \sum_j load_j = qglobal \\
\forall i, j, k \mid dist_{i,j} > 0, \quad & carga_{i,j,k} \leq cap_k \cdot X_{i,j,k} \\
& Coste \leq budget \\
\forall i, j, k \mid dist_{i,j} > 0, \quad & Time_j \geq Time_i + \frac{dist_{i,j}}{\min(velv_k, velc_{i,j})} - 10000 \cdot (1 - Y_{i,j,k}) \\
\forall i, j, k \mid dist_{i,j} > 0, \quad & X_{i,j,k} \leq cota_k \cdot Y_{i,j,k}
\end{aligned}
$$

# 5 Solucion

- Minimizar el coste del ayuda total:
  Coste = 65579.1667, Equidad = 0.2667, Tiempo = 127.2500
- Maximizar la equidad entre Agadez y Zinder:
  Coste = 80000.0000, Equidad = 0.4762, Tiempo = 127.2500
- Minimizar el tiempo de hacer la operaciÃşn:
  Coste = 78018.7500, Equidad = 0.2667, Tiempo = 93.5000

| | Coste | Equidad A | Equidad Z | Tiempo A (h) | Tiempo Z (h) |
|---|---|---|---|---|---|
| Min coste | 65579.1667 | 0.2667 | 1 | 127.25 | 116.25 |
| Max equidad | 80000.0 | 0.4762 | 0.4762 | 127.25 | 116.25 |
| Min tiempo | 78018.75 | 0.2667 | 1 | 94.25 | 83.25 |

Table 1: matriz de pagos

# 6 ProgramaciÃşn por metas

Ahora queremos encontrar una soluciÃşn multiobjetivo donde el coste debe ser menor o igual a 80 000 euros, el tiempo menor o igual que un tiempo T, y la equidad debe ser major o igual que 0. Elegimos el tiempo mas grande de la matriz de pago para el valor de T.

$$\begin{cases} Coste + n_1 - p_1 = 80000 \\ Time + n_2 - p_2 = T \\ Equidad + n_3 - p_3 = 0 \end{cases}$$

Donde

$$\begin{aligned} p_1 &= \text{cantidad arriba de 80 000} \\ p_2 &= \text{tiempo arriba de T} \\ n_3 &= \text{cantidad debajo de 0} \end{aligned}$$

La equidad es definida por ser siempre major o igual a 0, entonces queremos minimizar $p_1 + p_2$.

$$\begin{aligned} &\min p_1 + p_2 \\ \text{s.a.} & \\ & Coste + n_1 - p_1 = 80000 \\ & Time + n_2 - p_2 = 127.25 \\ & Equidad + n_3 - p_3 = 0 \end{aligned}$$

# 7 Codigo GAMS