

Taller de Economatemática

Práctica 2: Emergencia en Níger

Grupo 4

Sergio Escudero Medina

José López Gea

Silvia Mª Moral Muñoz

Índice

[**Introducción al problema**](#_lqz09828r19) **2**

[Descripción del problema](#_fet455az1osq) 2

[Modelización del problema](#_r2rp4j2dmq37) 2

[**Matriz de pagos**](#_z0sanpcm1hvg) **5**

[**Frontera de Pareto**](#_mjqpa0mo0n39) **6**

[**Programación por metas**](#_jraz51nty70h) **7**

[**Solución Compromiso**](#_wz5oaa5pb9jw) **8**

[Norma 1](#_wg1783dle1o5) 8

[Norma 2](#_lghefrae3wl4) 8

[Norma Infinito](#_xmi89zluv48c) 8

# Introducción al problema

## Descripción del problema

Se nos plantea una situación en la que Níger sufre una situación de hambruna y gran vulnerabilidad antes algunas enfermedades, ante lo cual las distintas ONG: FARMAMUNDI y CADEV han querido aportar ayuda en dos fases: distribución de apoyo alimentario esencial y distribución de distintos tipos de medicamentos.

Se nos pide realizar un modelo que nos permite enviar todos los suministros con ciertas restricciones, como lo son el presupuesto máximo, la disponibilidad de vehículos y el estado de las carreteras.

Planteamos el problema no como si fuese de flujo, sino permitiendo que los camiones puedan usar la misma carretera para ir y volver en el caso de que uno de los viajes haya sido sin mercancía, por ejemplo, que puedan ir de una ciudad A a B vacíos, y puedan volver de B a A cargados con mercancía.

## Modelización del problema

Para realizar el modelo matemático que nos permita encontrar una solución factible, hemos utilizado lo siguiente:

**Conjuntos:**

* Ciudades: Niamey, Gaya, Dosso, Tahoua, Maradi, Agadez y Zinder. Los subíndices {i, j} se mueven en este conjunto. Además, emplearemos alias(i,j,jj).
* Tipo de camiones: camiones de tipo 1 y camiones de tipo 2. El subíndice *k* se mueve dentro de este conjunto.

**Parámetros:**

* Demanda de suministros, en toneladas, de la ciudad *i*, denotada por *demi*
* Disponibilidad de suministros, en toneladas, de la ciudad *i*, denotada por *dispi*
* Capacidad de cada camión en toneladas, denotada por *capk*
* Coste fijo de mover cada camión [€/km], denotado por *c\_camk*
* Velocidad máxima de cada camión [km/h], denotada por *v\_camk*
* Distancia entre ciudades [km], denotada por *dij*
* Fiabilidad de la carretera de la ciudad i a j, denotada por *fiabij*
* Velocidad máxima a la que se puede circular por las carreteras entre ciudades [km/h], denotada por *v\_maxij*
* Cantidad de camiones de tipo k disponibles en la ciudad i, denotada por disp\_cik

**Escalares:**

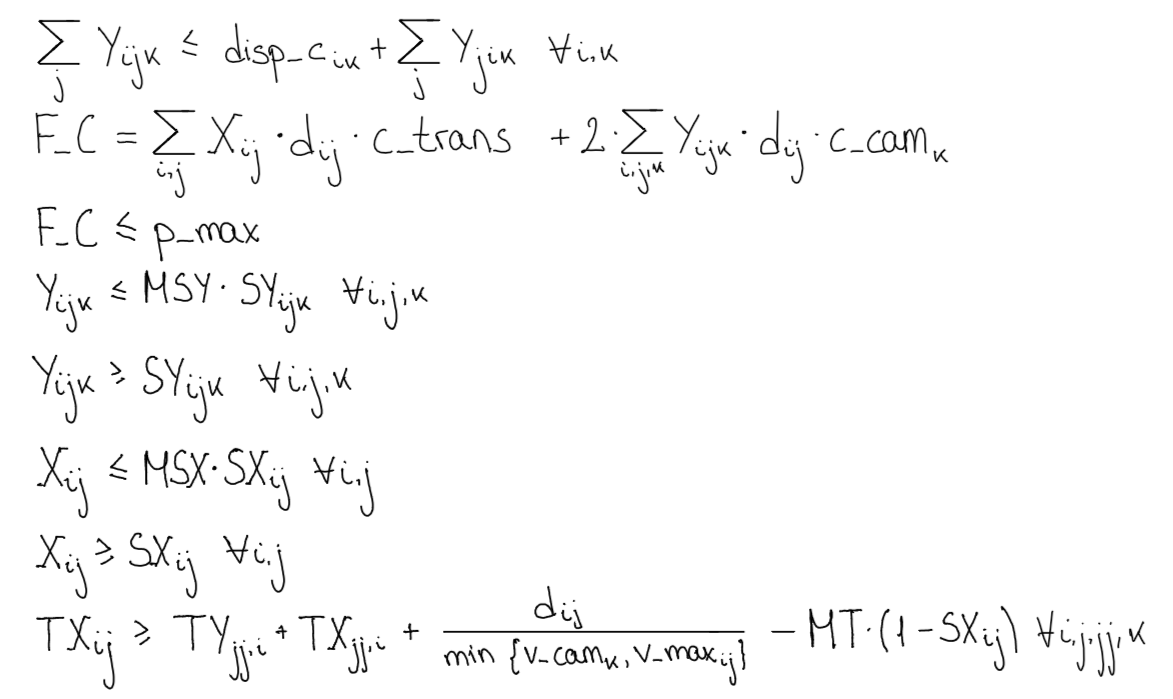
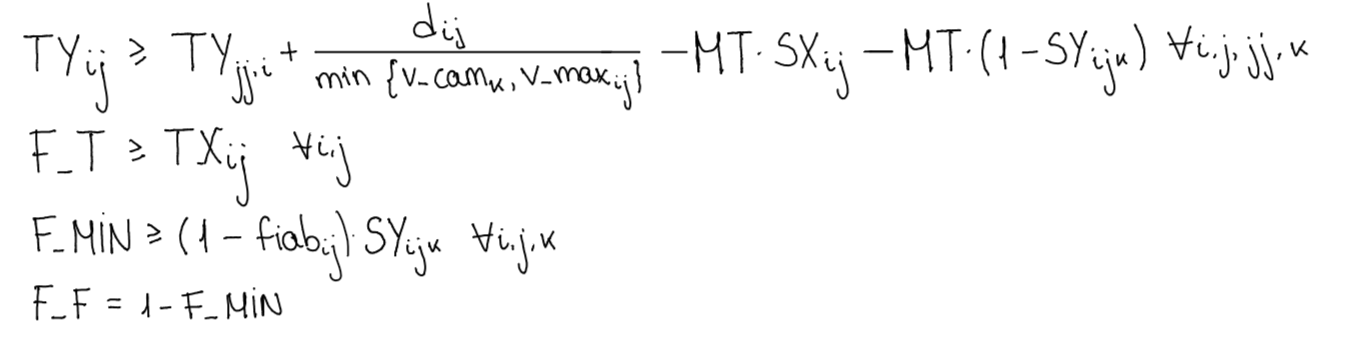
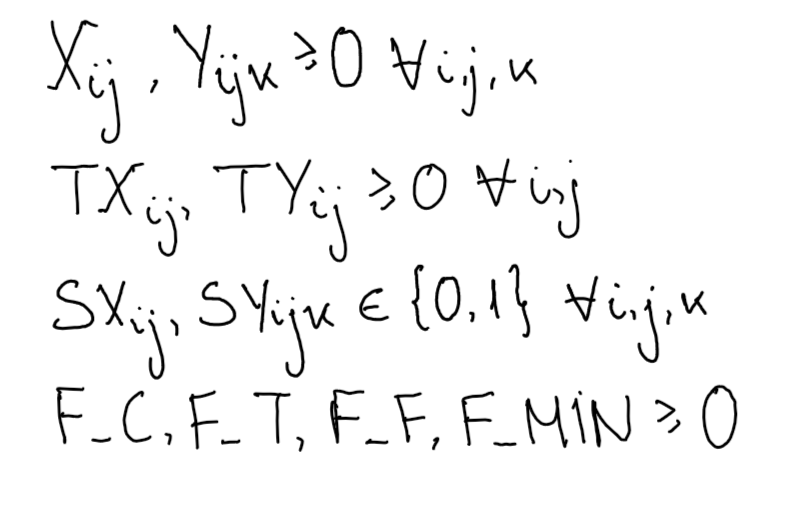
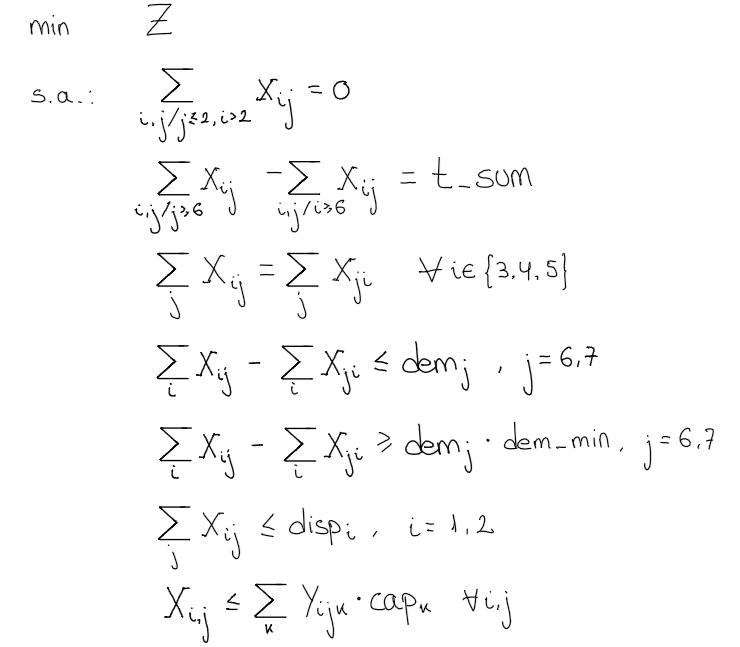
* Toneladas de ayuda que se suministran en el sistema: *t\_sum*
* Coste variable de transporte por cada tonelada [€/km]: *c\_trans*
* Presupuesto máximo: *p\_max*
* Porcentaje de mínima demanda a satisfacer: *dem\_min*
* Cotas:
  + - *MSY:* Cota para calcular SY
    - *MSX:* Cota para calcular SX
    - *MT:* Cota para calcular T

**Variables:**

* *Xij*: Toneladas que se transportan de la ciudad i a la ciudad j
* *Yijk*: Número de camiones de tipo k que van de la ciudad i a la ciudad j
* *TXij*: Instante de llegada de los camiones al ir de la ciudad i a la ciudad j **con** mercancía.
* *TYij*: Instante de llegada de los camiones al ir de la ciudad i a la ciudad j **sin** mercancía.
* *F\_MIN*: 1 - fiabilidad mínima
* *SYijk*: Variable binaria que indica si van camiones de tipo k de la ciudad i a la ciudad j, con o sin mercancía
* *SXij*: Si se va de la ciudad i a la ciudad j
* *F\_C*: Coste total del problema
* *F\_T*: Instante en el que toda la ayuda ya ha sido suministrada
* *F\_C*: Peor estado de las carreteras por las que se distribuye la ayuda

En todas las restricciones del modelo tomamos solamente valores de distancia mayores que cero para ahorrar variables en GAMS.

De esta manera, hemos implementado el modelo matemático mostrado a continuación, y para obtener los valores buscados, realizamos el problema tres veces: en cada una, igualamos Z a cada una de las funciones objetivo (F\_C, F\_T, F\_F).



# Matriz de pagos

**Parámetros:**

* Valor óptimo de F\_OBJw del conjunto de soluciones, denotado por *few*
* Peor valor de F\_OBJw del conjunto de soluciones eficientes, denotado por *faw*

**Variables:**

* *F\_OBJW*: Conjunto de funciones objetivo (*F\_C, F\_T, F\_F*)

**Matriz de pagos inicial:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Valor coste** | **Valor tiempo** | **Valor fiabilidad** |
| **Minimizar coste** | 47657.50 | 61 | 0.40 |
| **Minimizar tiempo** | 73687.50 | 50 | 0.40 |
| **Minimizar fiabilidad** | 64437.50 | 61 | 0.60 |

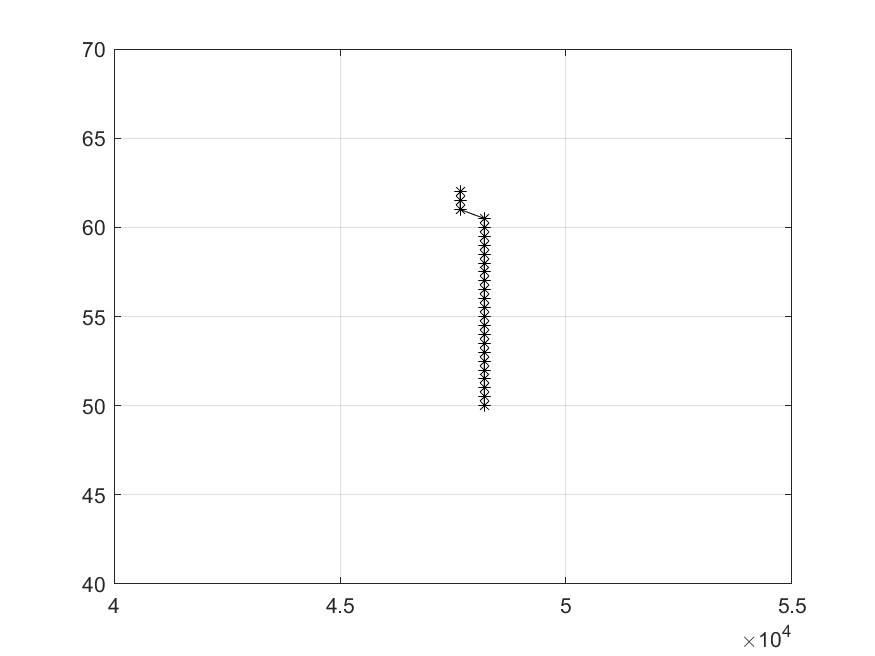
**Matriz de pagos final:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Valor coste** | **Valor tiempo** | **Valor fiabilidad** |
| **Minimizar coste** | 47657.50 | 61 | 0.40 |
| **Minimizar tiempo** | 48191.88 | 50 | 0.50 |
| **Minimizar fiabilidad** | 63287.50 | 61 | 0.60 |

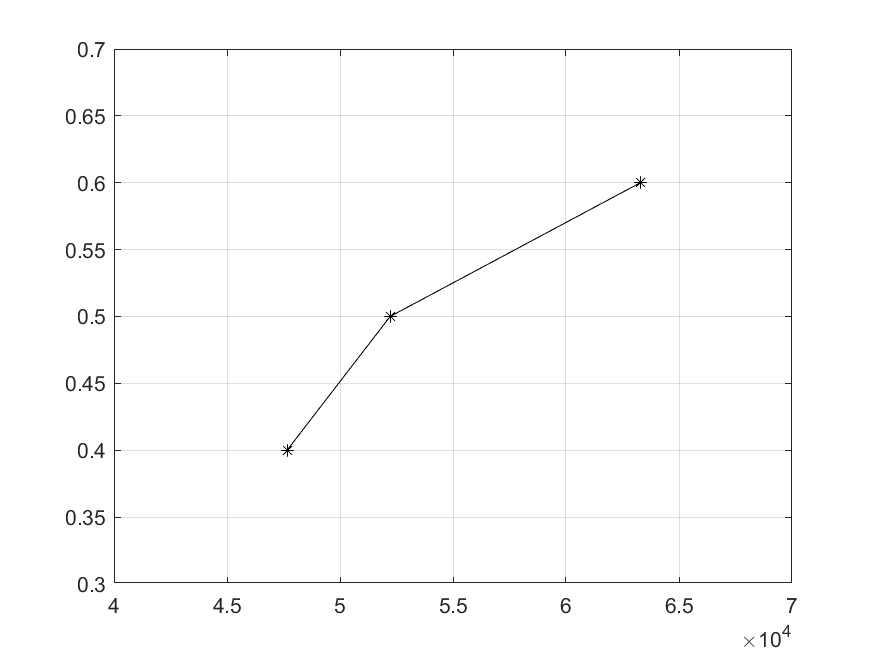
# Frontera de Pareto

Para calcular la frontera de Pareto utilizaremos el método de las ε-restricciones. El método consiste en optimizar uno de los objetivos, incorporando el resto como restricciones.

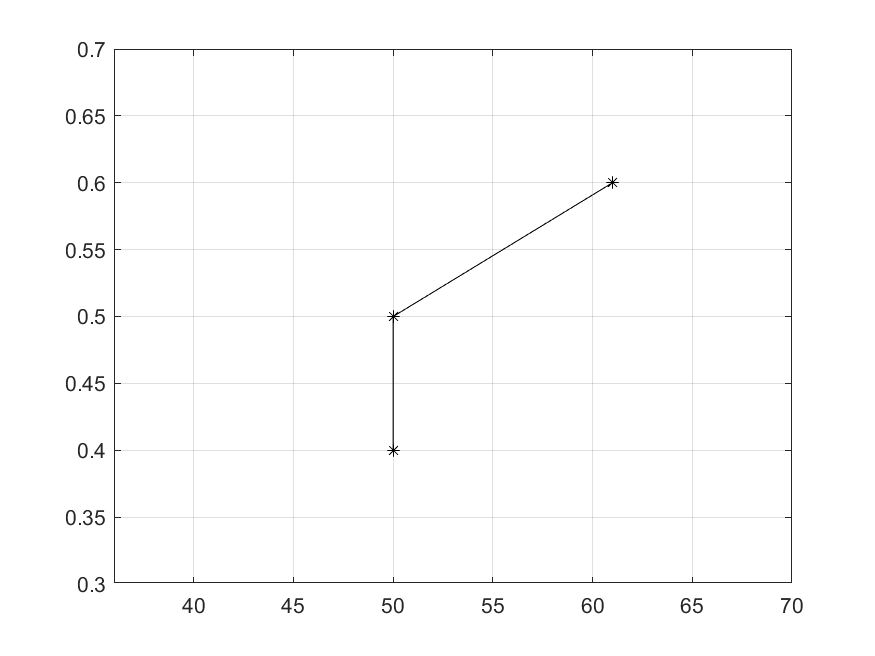
**Coste-Tiempo:**



**Coste-Fiabilidad:**



**Tiempo-Fiabilidad:**



# Programación por metas

Para cada objetivo se fijan unos niveles de aspiración que se desean alcanzar *fmw*.

**Parámetros:**

* Metas para la función a optimizar, denotada por *fmw*

**Variables:**

* *Nw*: Defecto de solución obtenida respecto a la solución deseada
* *Pw*: Exceso de solución obtenida respecto a solución deseada

Probamos con 3 ternas de niveles de aspiración:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Coste** | **Tiempo** | **Fiabilidad** |
| **Caso 1** | 60000 | 55 | 0.6 |
| **Caso 2** | 60000 | 50 | 0.6 |
| **Caso 3** | 60000 | 50 | 0.5 |

En el primer caso, observamos que, haciendo programación por metas, el coste total es de 63287.50€, el tiempo total de 61h y una fiabilidad de 0.6, por lo que tenemos un exceso de solución en el coste de 3287.50€ y en el tiempo de 6h.

En el segundo caso, probamos a reducir el tiempo a 50h. Nos queda un coste total de 60000€, un tiempo total de 50h y una fiabilidad de 0.5, por tanto, ahora no tenemos un exceso de solución, sino un defecto de solución de 0.1 en la fiabilidad.

En el tercer caso, reducimos la fiabilidad a 0.5 y observamos que no tenemos ni exceso ni defecto de solución respecto a la solución deseada, como era de esperar, por lo que estos niveles de aspiración serían una solución factible del problema.

# Solución Compromiso

En este caso, al igual que en el anterior, tomamos pesos iguales, por lo que ⍵.

**Variables:**

* F\_OBJnW: Conjunto de funciones objetivo normalizadas
* MAXn: Máximo de las funciones objetivo normalizadas

## Norma 1

La solución que obtenemos es coste total de 48191.88€, tiempo total de 50h y una fiabilidad de 0.4.

## Norma 2

La solución que obtenemos es coste total de 59062.50€, tiempo total de 50h y una fiabilidad de 0.5.

## Norma Infinito

La solución que obtenemos es coste total de 59062.50€, tiempo total de 58.03h y una fiabilidad de 0.5.