

Optimización y Metaheurísticas I

Terminales

Rafael Andrade Ruíz Capetillo¹
José Miguel de la Mora Álvarez²
José Luis Lobera del Castillo³
Samantha Licea Domínguez⁴

Universidad Panamericana

December 1, 2021

Índice

1 Introducción al problema de Terminales

- Problema
- Aplicación

2 Solución

- Instancias
- GRASP

3 Resultado

Problema

Se tienen n estaciones de trabajo y m terminales. El costo de asignar una estación de trabajo i a una terminal j es $c_{i,j}$. Cada estación de trabajo consume o demanda w_i unidades de la capacidad de una terminal. La capacidad de una terminal j es u_j .

Objetivo:

Encontrar la asignación del mínimo costo para formar una red de conexiones entre estaciones de trabajo y terminales.

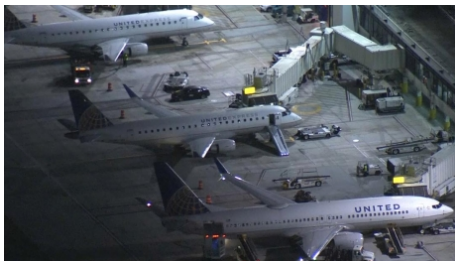
Observaciones:

Cada estación de trabajo debe ser asignada exactamente a una terminal. El costo $c_{i,j}$ se calcula usando las coordenadas de las terminales (x_j, y_j) y las estaciones de trabajo (x_i, y_i) de la siguiente forma:

$$c_{i,j} = \text{round} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Aplicación

Se tiene un cierto número de aviones que tienen p número de pasajeros que necesitan bajar del avión hacia las puertas de un aeropuerto. Dichas puertas soportan un determinado número de pasajeros que acaban de llegar (capacidad), por lo que los aviones tienen que llegar en una determinada posición (x,y) para cumplir con lo anterior. La tarea es asignar cuál avión va a qué puerta de forma que no se exceda el número de pasajeros por puerta y que se minimice la distancia que recorre cada avión para llegar a la puerta.



Instancias

Código en R utilizado para la generación de las 5 instancias.

```
1 a<-sample(1:6,100,replace=TRUE)
2 x<-sample(0:100,100,replace=TRUE)
3 y<-sample(0:100,100,replace=TRUE)
4
5 b<-sample(10:15,32,replace=TRUE)
6 x1<-sample(0:100,32,replace=TRUE)
7 y1<-sample(0:100,32,replace=TRUE)
8
9 for (i in 1:100){
10
11   cat(i,a[i],x[i],y[i],"\n")
12 }
13 cat("\n")
14 for (i in 1:32){
15   cat(i,b[i],x1[i],y1[i],"\n")
16 }
```

El algoritmo se compone de tres partes principales:

- **Pre-procesamiento:** Generar el diccionario que contiene las distancias de cada terminal con todas las estaciones.
- **Fase constructiva:** Generar un orden aleatorio de las terminales las cuales irán ocupando un lugar dentro de las estaciones de forma Greedy.
- **Fase de Mejora:** Cada terminal se asignará con la mejor opción en la que quepa todavía.

Resultado

Los resultados para cada una de las instancias fueron los siguientes:

INSTANCIA 1	
Media	1870.07
Mejor	1697
Peor	2086
Desviación	81.54756373746031
INSTANCIA 2	
Media	1567.64
Mejor	1420
Peor	1742
Desviación	70.43307877594579
INSTANCIA 3	
Media	1621.09
Mejor	1460
Peor	1821
Desviación	76.55355226007715
INSTANCIA 4	
Media	1189.61
Mejor	1099
Peor	1311
Desviación	46.942882766445464
INSTANCIA 5	
Media	1421.24
Mejor	1271
Peor	1582
Desviación	62.12111337760075

¡GRACIAS!
:)