**Task 1:**

**1.1** Maximizar a dimensão do grupo e indicar um qualquer encaminhamento.

Problema: Caminhos de capacidade máxima (adaptação do Algoritmo de Dijkstra)

**Formalização**

Constantes:

Variáveis:

*(variável associada à formulação do problema de caminho mais curto associado ao Dijkstra)*

*(variável associada à formulação do problema do fluxo máximo)*

*(arestas do caminho da solução: y igual)*

*(resto das aresta: y = 0)*

Função objetivo:

*(maximizar a soma de y de todas as arestas a começar no nó origem --> só um y terá valor !=0)*

Restrições:

*(restrição do nó origem: só há uma aresta percorrida a começar no nó origem )*

*(restrição do nó destino: só há uma aresta percorrida a acabar no último nó )*

*(restrição dos nós intermédios: se há uma aresta a entrar num nó, tem de haver uma a sair )*

*(restrição dos nós intermédios: pessoas a entrar num nó = pessoas a sair)*

1.2 Problema multi-objetivo (2 funções objetivo) e restrições anteriores.

Funções objetivo:

*(maximizar nº de pessoas transportado)*

*(minimizar o número de arestas do caminho -1 = nº de transbordos)*

Estratégia usada para cálculo das soluções ótimas de Pareto extremas deste problema.

1º -> Cálculo de todas as soluções ótimas alternativas que maximizam f1: Conjunto C1

2º -> Escolher as soluções com o valor mínimo de f2 dentro do C1, isto é, eliminar as soluções dominadas de C1

*Exemplo*

|  |  |  |
| --- | --- | --- |
|  | *max f1 (nº de pessoas)* | *min f2(nºtransbordos)* |
| *Caminho 1* | *10* | *3* |
| *Caminho 2* | *10* | *4* |
| *Caminho 3* | *10* | *5* |

3º -> Cálculo de todas as soluções ótimas alternativas que minimizam f2: Conjunto C2

4º -> Escolher as soluções com o valor máximo de f1 dentro do C2, isto é, eliminar as soluções dominadas de C2

*Exemplo*

|  |  |  |
| --- | --- | --- |
|  | *max f1 (nº de pessoas)* | *min f2(nºtransbordos)* |
| *Caminho 4* | *7* | *2* |
| *Caminho 5* | *7* | *2* |
| *Caminho 6* | *5* | *2* |