

# Inteligencia Artificial

*Laboratorio 1: Práctica de Búsqueda Local*

**Bachelor of Technology**  
in  
**COMPUTER SCIENCE AND ENGINEERING**

**Christian Donovan ( christian.donovan )**  
**Agnès Felip Díaz ( agnes.felip )**  
**Ignasi Fibla Figuerola ( ignasi.fibla )**  
**Marta Granero i Martí ( marta.granero.i )**



**DEPARTMENT OF COMPUTER SCIENCE**  
**Universitat Politècnica de Catalunya**  
**Abril, 2022**

**Contents**

**List of Figures**

**List of Tables**

## **Abstract**

A brief summary of the project goes here.

## 1 Introducción

## 2 Descripción del problema

Disponemos de un tablero de 50 x 50, o en otras palabras de un tablero de 2500 casillas. En el terreno representado por el tablero ha ocurrido un desastre natural. Existen  $C$  centros de rescate, los cuales se encuentran en los límites del tablero. De estos centros saldrán los helicópteros, con el objetivo de rescatar a grupos de personas.

Todos los grupos, los cuales son indivisibles, están formados por un mínimo de una persona y un máximo de doce. Además, dependiendo de si existe algún herido en este grupo, se le asignará una entero, el cual denota la prioridad. En el caso que haya uno, o más, heridos, el grupo tendrá prioridad uno y el tiempo que tardara el helicóptero en recogerlo será proporcional al número de personas, más concretamente dos minutos por cada integrante. Sin embargo, si no hay ningún herido en el grupo de personas, se le asignará prioridad dos, y el tiempo que tardara el helicóptero en recogerlo será de un minuto por cada persona.

Los helicópteros tienen una capacidad máxima de 15 personas, pero pueden no llegar a esta capacidad en uno de sus viajes. La velocidad máxima a la que viajaran es de  $100 \frac{km}{h}$ . Cuando regresan al centro de control, el helicóptero tendrá que esperar 10 minutos en poder volver a hacer un rescate.

El cálculo de la distancia recorrida entre dos grupos se realiza mediante la distancia euclídea entre sus coordenadas.

$$d(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

Figure 1: Distancia euclídea

### 3 Función heurística

En el mundo de la informatica, entendemos por función heuristica aquella función capaz de classificar las alternativas en algoritmos de busqueda en cada paso de ramificación, en función de la información disponible para decidir el camino a seguir.

En esta practica estamos tratando de resolver un problema de busqueda, más concretamente una busqueda local. ( Explicar por encima la necesidad de usar una heuristica ).

Como se ha mencionado anteriormente, se hace uso de una matriz para guardar el estado, más concretamente se usa un ArrayList de ArrayList de Nodos. Por lo tanto, de una forma sencilla podemos obtener la distancia que recorre un helicoptero y los viajes que hacen a los diferentes centros de control.

De primeras se penso en unicamente evaluar, mediante el heuristico, los viajes que hacia todos los helicopteros hacia un centro de control, o en otras palabras, solo se tenian en cuenta el siguiente tipo de eventos  $Gx -i Cy -i Gz$ , donde G representa grupo, C centro y x,y e z representa el indice en el conjunto de grupos y centros, respectivamente. Esta primera aproximación del heuristico se podia calcular mediante la siguiente formula:

$$h(n) = \sum_{h=0}^{|H|} V_{org:CC} + V_{dst:CC}$$

Figure 2: Función heurística. Primera aproximación

Rapidamente te das cuenta que esta función heuristica no era correcta, ( !TODO! ESCRIBIR MOTIVOS ). Por lo tanto se opto por una idea un poco mas compleja, pero que partia de esta primera formalización.

En este caso, el heuristico sigue teniendo en cuenta los viajes que tienen como origen u destino un centro de control, pero no nos retornan el valor completo de este. Estos viajes tienen un factor de ponderación, cuyo valor no va a permitir hacer diferentes experimentos. Ademas, le añadimos los viajes que hace entre grupos, con un factor de ponderación para tener mas posibilidades. Esto nos lleva a la siguiente formula:

$$h(n) = \sum_{h=0}^{|H|} \alpha \cdot V_{org:CC} + \beta \cdot V_{dst:CC} + \gamma \cdot V_{org:!CCdst:!CC}$$

Figure 3: Función heurística. Segunda aproximación