



## Práctica 2: *Satisfacción de Restricciones y Búsqueda Heurística* **Heurística y Optimización**

Grado de Ingeniería Informática, 2020 - 2021

### 1. Objetivo

El objetivo de esta práctica es que el alumno aprenda a modelar y resolver problemas tanto de satisfacción de restricciones (CSP) como de búsqueda heurística.

### 2. Enunciado del problema

Una importante agencia espacial se ha puesto en contacto con alumnos del curso de Heurística y Optimización. Tras varias entrevistas os han seleccionado a tu compañero y a ti para dar solución a varios de sus problemas.

#### 2.1. Parte 1: Asignación de antenas de transmisión a satélites

En los últimos años se ha visto un crecimiento considerable en el número de satélites que orbitan alrededor de la tierra y sus requisitos de comunicación, lo que ha aumentado también el número de solicitudes de asignación de franjas horarias a antenas de transmisión en tierra. En el problema en particular que debemos resolver, la agencia nos dice que tenemos 6 satélites, y 12 antenas de transmisión. Cada satélite dispone de una o varias franjas horarias en las que son “visibles” por una o varias antenas de transmisión. Estas relaciones entre satélites, franjas horarias y antenas se muestran en la Tabla 1. En la primera columna de esta tabla se muestra el identificador del satélite, y en la segunda la franja horaria en la que está visible para las antenas cuyos identificadores se muestran en la tercera columna.

La agencia nos pide determinar cual es la asignación de satélites y sus franjas horarias a antenas de transmisión considerando las siguientes restricciones:

- Todos los satélites deben tener asignada una antena de transmisión en tierra para cada una de sus franjas horarias visibles. Por ejemplo, en el caso de SAT3 que tiene dos franjas visibles (06:00 - 12:00 y 13:00 - 16:00), se debe asignar una antena de transmisión para la primera franja y otra para la segunda, encontrándose ambas en el conjunto de antenas para los cuales el satélite está visible en esas franjas.

Satélite	Franja	Antenas
SAT1	00:00 - 12:00	ANT1, ANT2, ANT3, ANT4
SAT2	00:00 - 12:00	ANT1, ANT2, ANT3
SAT3	06:00 - 12:00	ANT4, ANT6
SAT3	13:00 - 16:00	ANT7, ANT9, ANT10
SAT4	16:00 - 00:00	ANT8, ANT11, ANT12
SAT5	06:00 - 13:00	ANT1, ANT7, ANT12
SAT6	09:00 - 13:00	ANT7, ANT9
SAT6	13:00 - 19:00	ANT3, ANT4, ANT5

Tabla 1: Satélites, franjas horarias y antenas

- Puesto que los satélites SAT1 y SAT2 comparten trayectorias similares, se requiere que a ambos se les asigne la misma antena de transmisión.
- Los satélites SAT2, SAT4 y SAT5 deben tener asignados antenas de transmisión diferentes.
- Si SAT5 se comunica con ANT12, SAT4 no se puede comunicar con ANT11.
- Si en una solución se asignan las antenas ANT7 y ANT12, se deben asignar ambas a franjas horarias que comiencen antes de las 12:00 o a franjas horarias que comiencen después de las 12:00. No podría ocurrir, por ejemplo, que ANT7 estuviera asignada a la franja 06:00-12:00 de SAT3, y ANT12 a la franja 16:00-00:00 de SAT4.

Para esta parte se pide:

- Modelar el problema como un problema de satisfacción de restricciones CSP.
- Utilizando la librería *python-constraint*, desarrollar un programa que codifique el modelo anterior, y permita determinar para cada satélite y sus franjas horarias la antena que tiene asignada.

El programa deberá ejecutarse desde una consola o terminal con el siguiente comando:

```
python CSPScheduling.py
```

- Cada grupo debe generar sus propios casos de prueba, por ejemplo, modificando el caso propuesto añadiendo/quitando satélites/antenas, ampliando/reduciendo las franjas de horarias en las que los satélites son “visibles”, con el fin de analizar el comportamiento del programa.

## 2.2. Parte 2: Búsqueda Heurística

Para la segunda parte, la agencia nos pide resolver un problema diferente. En Figura 1 se muestran los satélites SAT1 y SAT2 que orbitan alrededor de la tierra utilizando órbitas diferentes, y donde cada uno tiene asociada una banda de observación.

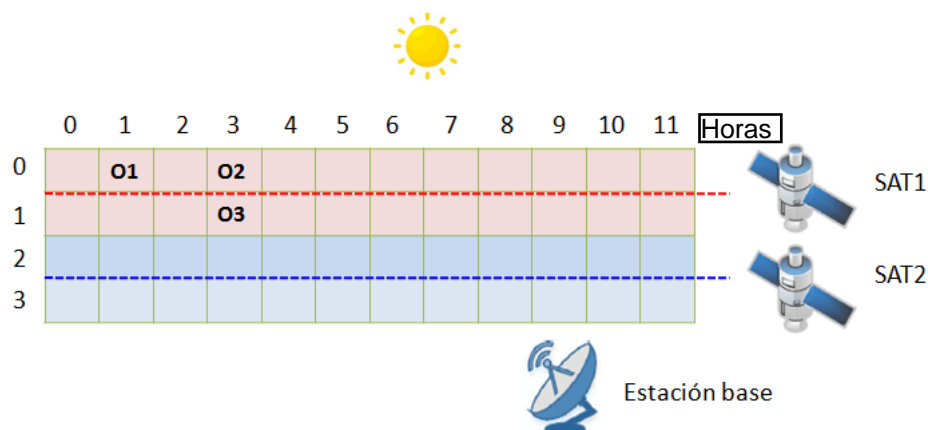


Figura 1: Ejemplo de dos satélites y tres puntos de observación.

Así por ejemplo, el satélite SAT1 tiene asociada la banda de observación correspondiente a las filas 0 y 1 de la Figura 1, mientras que el satélite SAT2 tiene asociadas la banda de observación correspondientes a las filas 2 y 3, de forma que solo pueden tomar observaciones de los objetos que se encuentren en su banda de observación en el momento que sobrevuelan sobre ellos. Además, los satélites solo están “visibles” por la estación base durante 12 horas al día, quedando el resto de horas fuera de su alcance. Los satélites pueden realizar distintas

operaciones durante esta franja de 12 horas, y cada una de 1 hora de duración: pueden no hacer nada, pueden tomar observaciones del terreno, pueden transmitir estas observaciones a la estación base, pueden girar para cambiar su banda de observación o pueden realizar operaciones de carga de energía utilizando la luz solar. Solo se permite que cada satélite haga una de estas operaciones a la vez. Con respecto a las operaciones de giro, se permite que el satélite SAT1 cambie su banda de observación de las filas (0, 1) a la (1, 2) y viceversa, y el SAT2 puede cambiar su banda de observación de las filas (2, 3) a la (1, 2) y viceversa. Además, cada satélite está equipado con una batería con un número de unidades de energía, y tiene asociado un consumo de unidades de energía por la toma de cada observación, por cada transmisión de datos que realiza a la estación base, y por cada operación de giro. Por último, cada operación de carga permite a cada satélite recargar un número de unidades determinado en su batería.

A modo de ejemplo, en la Figura 1 aparecen tres observaciones localizadas en puntos diferentes de la superficie: O1, O2 y O3. Podemos asumir, además, que tanto para SAT1 y SAT2 el consumo de energía por cada observación, transmisión de datos y por cada giro es de 1 unidad. Además, para ambos casos una operación de carga les permiten obtener 1 unidad de energía. Sin embargo SAT1 tiene una batería de 1 unidad, y SAT2 de 8 unidades. El plan resultante que permite realizar las tres observaciones programadas y transmitir las a la estación base es:

```
1. SAT1: IDLE, SAT2: IDLE
2. SAT1: Observa O1, SAT2: IDLE
3. SAT1: Carga, SAT2: Gira
4. SAT1: Observa O2, SAT2: Observa O3
5. SAT1: Carga, SAT2: Transmite O3
6. SAT1: Transmite O1, SAT2: IDLE
7. SAT1: Carga, SAT2: IDLE
8. SAT1: Transmite O2, SAT2: IDLE
```



Aquí explica el texto que sale

En cada línea de la secuencia de acciones anterior, se muestra por cada satélite su identificador y la acción que tiene que realizar. Así, por ejemplo, en la primera hora de la franja ambos satélites no hacen nada. En la segunda hora, SAT1 toma una observación del objetivo O1, y SAT2 no hace nada. De hecho, en este caso, solo SAT1 puede tomar la observación de O1 puesto que es inalcanzable para SAT2 incluso aunque éste último girara. Al realizar la observación O1, SAT1 consume 1 unidad de energía, por lo que necesita realizar una operación de carga en la siguiente hora de la franja para seguir operando, mientras que SAT2 gira para situar su banda de observación en (1, 2) y tener al alcance el objetivo O3. En la siguiente hora, SAT1 toma una observación de O2 y SAT2 de O3. Por último se transmiten las observaciones tomadas a la estación base. No es necesario en ningún caso tomar o transmitir las observaciones en orden. El plan se considera finalizado cuando se han transmitido todas las observaciones programadas a la estación base.

Hay que tener en cuenta que es posible que no se puedan realizar todas las observaciones en una sola pasada de los satélites por la franja visible, con lo cual será necesario que los satélites transiten por las 12 horas de sombra, antes de volver a reanudar con sus operaciones. En la franja de sombra la única operación disponible para los satélites es la de no hacer nada que no tiene consumo de energía asociado. Como ejemplo de esta situación, si en la figura anterior colocamos un objetivo O4 en la posición (0, 4), solo sería alcanzable por SAT1. Pero SAT1 no puede realizar la observación O2 y O4 en una sola pasada por la franja visible, puesto que después de tomar O2 debe realizar necesariamente una operación de carga para estar en disposición de tomar otra observación. Será en la siguiente pasada por la franja visible donde pueda tomar la observación de O4 y transmitirla a la estación base.

En esta parte se pide:

- Modelar el problema de planificación de toma de observaciones propuesto como un problema de búsqueda heurística.
- Implementar el algoritmo A\* (en el lenguaje de programación que se considere) que permite resolver el problema, e implementar dos funciones heurísticas informadas y admisibles que estimen el coste restante, de modo que sirvan de guía para el algoritmo.

La implementación desarrollada se deberá ejecutar desde una consola o terminal con el siguiente comando:

```
./Cosmos.sh <problema.prob> <heurística>
```

donde:

- `Cosmos.sh`: Nombre del script que permite invocar al programa desarrollado.
- `problema.prob`: Nombre del fichero que contiene el problema que queremos resolver. Un ejemplo de fichero de entrada que modela el problema de la figura 1 se muestra a continuación:

```
OBS: (0,1);(0,3);(1,3)
SAT1: 1;1;1;1;1
SAT2: 1;1;1;1;8
```

donde en la primera línea se muestran las coordenadas de cada una de las observaciones, y las dos siguientes muestran la configuración de cada satélite. En la configuración de cada satélite se muestran en orden separados por “;” el coste de unidades de energía de cada observación, transmisión, y giro, las unidades que se recargan en cada operación de carga, y las unidades de energía disponibles en las baterías.

- `heurística`: Nombre de la heurística. Los posibles valores para el parámetro heurística deben ser detallados en la memoria y en la ayuda de la implementación desarrollada.

El programa debe generar dos ficheros de salida. Ambos se deben generar en el mismo directorio donde se encuentre el problema de entrada y deben tener el mismo nombre del problema (extensión incluida) más una extensión adicional. Los ficheros son los siguientes:

- **Solución del problema**. Debe contener las operaciones que realizan cada uno de los satélites para obtener todas las observaciones y transmitirlas a la estación base. Se puede ver un ejemplo de formato de salida en el plan que se mostraba para resolver el problema de la figura 1. La extensión de este fichero debe ser `.output`.

Ejemplo: `problema.prob.output`

- **Fichero de estadísticas**. Este fichero debe contener información relativa al proceso de búsqueda, como el tiempo total, coste total, longitud del plan, nodos expandidos, etc. Por ejemplo,

```
Tiempo total: 145
Coste total: 54
Longitud del plan: 27
Nodos expandidos: 132
```

La extensión de este fichero debe ser `.statistics`. Ejemplo: `problema.prob.statistics`.

- **Proponer casos de prueba con diversas configuraciones de problemas** (distinto número de observaciones, configuraciones de satélites) y **resolverlos con la implementación desarrollada**. Estos casos se deben generar razonablemente dependiendo de la eficiencia alcanzada en la implementación.
- **Realizar un estudio comparativo utilizando las dos heurísticas implementadas** (número de nodos expandidos, tiempo de cómputo, etc.).

### 3. Directrices para la Memoria

La memoria debe entregarse en formato pdf y tener un máximo de 15 hojas en total, incluyendo la portada, el índice y la contraportada. Al menos, ha de contener:

1. Breve introducción explicando los contenidos del documento.
2. Descripción de los modelos, argumentando las decisiones tomadas.
3. Análisis de los resultados.
4. Conclusiones acerca de la práctica.

**Importante:** Las memorias en un formato diferente a pdf serán penalizadas con 1 punto.

La memoria **no debe incluir código fuente** en ningún caso.

### 4. Evaluación

La evaluación de la práctica se realizará sobre 10 puntos. Para que la práctica sea evaluada deberá realizarse al menos la primera parte de la práctica y la memoria.

La distribución de puntos es la siguiente:

1. Parte 1 (3 puntos)
  - Modelización del problema (0.75 punto)
  - Implementación del modelo (1.25 puntos)
  - Resolución y análisis de los casos de prueba (1 punto).
2. Parte 2 (7 puntos)
  - Modelización del problema (1 puntos)
  - Implementación del algoritmo (3 puntos)
  - Resolución de casos de prueba y análisis comparativo de las heurísticas implementadas (3 puntos)

En la evaluación de la modelización de los problemas, un modelo correcto supondrá la mitad de los puntos. Para obtenerse el resto de puntos, la modelización del problema deberá:

- Ser formalizada correctamente en la memoria.
- Ser, preferiblemente, sencilla y concisa.
- Estar bien explicada (ha de quedar clara cuál es la utilidad de cada variable/restricción).
- Justificarse en la memoria todas las decisiones de diseño tomadas.

En la evaluación de la implementación de los modelos, y cualquier aspecto relacionado con ella, sólo se corregirán implementaciones que compilen y que respondan a lo que se solicite en el enunciado. Además, una implementación correcta supondrá la mitad de los puntos. Para obtenerse el resto de puntos, la implementación del problema deberá:

- Corresponderse íntegramente con el modelo propuesto en la memoria.
- Entregar código fuente correctamente organizado y comentado a lo que se pide en el enunciado. Los nombres deben ser descriptivos.
- Contener casos de prueba que aporten diversidad para la validación de la implementación.

En la evaluación del análisis de resultados, se valorará positivamente la inclusión de conclusiones personales sobre la dificultad de la práctica y sobre lo que se ha aprendido durante el desarrollo de la misma.

## 5. Entrega

Se tiene de plazo para entregar la práctica hasta el 22 de Diciembre a las 23:55. El deadline es firme y no se extenderá en ningún caso.

Sólo un miembro de cada pareja de estudiantes debe subir:

- Un único fichero `.zip` a la sección de prácticas de Aula Global denominado *Entrega Práctica 2*.
- El fichero debe nombrarse `p2-NIA1-NIA2.zip`, donde NIA1 y NIA2 son los últimos 6 dígitos del NIA (rellenando con 0s por la izquierda si fuera preciso) de cada miembro de la pareja. Ejemplo: `p2-054000-671342.zip`.
- La memoria en formato pdf debe entregarse a través del enlace Turnitin denominado *Entrega Memoria Práctica 2*. La memoria debe entregarse en formato pdf y debe llamarse `NIA1-NIA2.pdf` — después de sustituir convenientemente el NIA de cada estudiante. Ejemplo: `054000-671342.pdf`

La descompresión del fichero entregado en primer lugar debe generar un directorio llamado `p2-NIA1-NIA2`, donde NIA1 y NIA2 son los últimos 6 dígitos del NIA (rellenando con 0s por la izquierda si fuera preciso) de cada miembro de la pareja. Este directorio debe contener: primero, la misma memoria en formato pdf que ha sido entregada a través de Turnitin, y debe llamarse `NIA1-NIA2.pdf` — después de sustituir convenientemente el NIA de cada estudiante; segundo, un fichero llamado *autores.txt* que identifique a cada autor de la practica en cada línea con el formato: NIA Apellidos, Nombre. Por ejemplo:

```
054000 Von Neumann, John
671342 Turing, Alan
```

La descompresión de este fichero debe producir al menos dos directorios llamados exactamente “`parte-1`” y “`parte-2`”, que contengan los archivos necesarios para ejecutar correctamente cada una de las partes.

**Importante:** no seguir las normas de entrega puede suponer una pérdida de hasta 1 punto en la calificación final de la práctica.

Se muestra a continuación una distribución posible de los ficheros que resultan de la descompresión:

