

UNIVERSIDAD DE GRANADA

INGENIERÍA INFORMÁTICA

Computación y Sistemas Inteligentes

Práctica 3: Planificación HTN

Autor: JOSÉ ANTONIO RUIZ MILLÁN

email: jantonioruiz@correo.ugr.es

Asignatura: Técnicas de los Sistemas Inteligentes

3 de junio de 2018



Índice

1. Ejercicio 1	2
2. Ejercicio 2:	2
3. Ejercicio 3	2
4. Ejercicio 4	3
4.1. Solución:	4
4.2. Experimentos:	5

1. Ejercicio 1

El siguiente problema a resolver (problema-zeno-V01.pddl) consiste en transportar 3 personas (inicialmente en las ciudades C1, C2 y C3) a la ciudad C5, considerando que el avión está en la ciudad C4. Se asume al igual que en el problema ejemplo que no hay restricciones de fuel.

Este problema no era capaz de solucionarse con el dominio que nos entregaban porque no contemplaba la opción de que un avión estuviese desde un inicio en una ciudad distinta que la persona que tiene que transportar, por ello, lo único que tuve que realizar fué añadir un nuevo method dentro del task *transport-person* de la siguiente forma:

- **Precondiciones:** Que la persona no esté en la misma ciudad que el avión.
- **Tareas:** Mover el avión de la ciudad en la que se encuentra a la ciudad de la persona, embarcar a esa persona, llevarla a su destino y desembarcarla.

2. Ejercicio 2:

El problema 2 (fichero problema-zeno-V02.pddl) consiste en asumir que hay restricciones de fuel. El fuel inicial del avión es de 200 y la capacidad total de 300. Deben contemplarse ahora acciones de repostaje. La situación de partida de personas y avión es la misma que en el problema anterior.

Ahora el problema que nos plantean no se resuelve ya que antes asumíamos que siempre se podía viajar independientemente de la gasolina que tuviese el avión. Para solucionar esto, modifiqué el *derivate* que nos comprueba si tenemos fuel, de tal forma que ahora la condicion es que si la gasolina que tenemos, sumada a la gasolina que vamos a gastar para hacer el viaje, es suficiente. En caso contrario no tenemos gasolina y tenemos que repostar.

Una vez realizado ese cambio, tuve que añadir un method nuevo al task *mover-avion* ya que sólo teníamos que podíamos moverlo si teníamos fuel suficiente, por lo que como he comentado añadí este method que hace lo siguiente:

- **Precondiciones:** Que no haya fuel suficiente para volar.
- **Tareas:** Repostar fuel en la ciudad en la que se encuentre y una vez haya llenado el tanque, volar (se asume que con el tanque lleno puede volar a cualquier lado).

3. Ejercicio 3

Este problema (fichero problema-zeno-V03.pddl) consiste en considerar acciones de vuelo lento y rápido para tratar de transportar las personas lo más rápido posible con un límite de fuel. En el dominio se tienen que codificar los métodos y tareas de manera que se priorice el uso de acciones de velocidad rápida. El límite de fuel se define con la función (fuel-limit) y es asignado en el estado inicial a 1500. La suma

total de fuel gastado en todos los transportes no puede superar 1500 unidades, pero el avión debe viajar siempre lo más rápido posible.

Ahora tenemos que tener en cuenta los dos tipos de vuelos, el rápido y el lento, así que yo he añadido tres nuevos predicados, uno para comprobar si tenemos fuel pero comparando ahora con lo terminos del vuelo rápido, y otros dos para comprobar si se cumple el límite de fuel establecido tanto para vuelo lento como para rápido. Por otra parte he añadido la funcion *fuel-limit* para especificarlo en el problema.

Por lo que referido a comprobaciones ya está listo. Lo que tuve que añadir para el correcto funcionamiento son dos nuevos method en el task *mover-avion* ya que ahora tenemos que tener en cuenta los vuelos rápidos tambien. Como nos dicen que prioricemos estos vuelos rápidos, lo que he echo es poner los method de la siguiente forma:

- Volar si tenemos fuel para volar rápido.
- Si no tenemos fuel para volar rápido pero podemos volar rápido, repostamos y volamos rápido.
- Volar si tenemos fuel para volar lento.
- Si no tenemos fuel para volar lento pero podemos volar lento, repostamos y volamos lento.

Los nuevos method son iguales que los de volar lento explicados en apartados anteriores, cambiando en los tasks que hace vuelo rápido en vez de lento. Con esto siempre volará rapido antes que lento, siempre que pueda.

4. Ejercicio 4

Representar un dominio, extendiendo el obtenido en el último problema, con las siguientes características:

1. Modificar la representación de acciones *board* y *debark* para poder representar que cada avión tiene una capacidad máxima de pasajeros, que cada vez que se embarca/desembarca el número de pasajeros de un avión se incrementa/decrementa en 1.
2. Añadir/modificar las tareas compuestas necesarias para que puedan embarcarse varios pasajeros en un avión, una vez que el avión esté en la ciudad adecuada, o desembarcar varios pasajeros de un avión, una vez el avión haya llegado al destino del pasajero.
3. Añadir/modificar las tareas primitivas, compuestas, predicados y/o funciones para poder representar que hay una duración limitada para los viajes de cualquier avión.

4.1. Solución:

1. Para este apartado, lo único que tuve que realizar fué añadir dos nuevas funciones para controlar el número de pasajeros que lleva cada avión y el máximo de pasajeros que puede llevar un avión, por lo que creé dos funciones para controlar cada uno de estos aspectos. Una vez hecho esto en el dominio, tuve que modificar las primitivas (por ello las adjunto a la practica) para que cuando cargamos una persona, se suma 1 a las personas que lleva ese avión y cuando se baja uno del avión se decremente en 1 ese valor.

Claramente en cada una de los method y task dónde vayamos a cargar o desembarcar a una persona, se comprueba que puede hacerlo gracias a estas funciones.

2. En el segundo apartado tuve que modificar el task *transport-person* ya que éste transporta a una persona y nada más. Mi objetivo era hacerlo recursivo por lo que tuve que eliminar los parámetros de la tarea y pasé a llamarla *transport-persons*. Explicaré como he definido esta tarea ya que es totalmente nueva a la de los ejercicios anteriores. Antes de nada, comentar que he creado dos task más, uno llamado *embarcar* y otro llamado *desembarcar* que a su vez son recursivos y lo que realizan es, estando en una ciudad una persona y una avión donde esa persona va a ser montada, embarca a todas las personas que vayan al mismo destino que ésta. Idem para desembarcar.

Una vez definido esto. voy a especificar el orden de mis method dentro del task *transport-person*

- Lo primero que comprueba es si el avión está en una ciudad y lleva pasajeros y los pasajeros quieren ir a esa ciudad. Si eso ocurre, los desembarca y vuelve a llamar a *transport-person*
- Comprueba las restricciones de vuelo, es decir, si puede seguir volando por la gasolina o por la duración del vuelo (que veremos en el apartado 3). Si ese avión no va a poder seguir volando y lleva personas dentro, las suelta para que otro avión las pueda llevar.
- Compruebo si el avión y la persona están en la misma ciudad teniendo en cuenta que esa persona tiene un destino distinto y también que el avión esté vacío. Si esto se cumple, monta a esa persona y a todas las que vayan a su mismo destino utilizando la tarea *embarcar* comentada anteriormente y llama recursivamente.
- Compruebo ahora si la persona y el avión están en distintas ciudad y el avión está vacío. Si esto se cumple, el avión se transporta hacia esa ciudad y se llama recursivamente, ya que los method anteriores se encargarán de cargarlo.
- Ahora compruebo que el avión y la persona estén en la misma ciudad pero el avión tiene pasajeros que van al mismo destino que ésta persona. Lo que hace es montarla directamente y llamar recursivamente.
- Ahora compruebo que el avión y la persona están en la misma ciudad como en el apartado anterior pero ahora el avión tiene personas de distinto destino al suyo. Comprueba a través de las distancias entre las ciudades, si sale rentable recoger a esta persona o no. Si sale rentable, la recoge y se llama recursivamente.
- Ahora se realiza el mismo proceso pero con avion y persona en distintas ciudades, es decir, tenemos avión y persona en distinta ciudad y el avión lleva pasajeros del mismo

destino que esa persona. Si esto ocurre, la recoge, ya que siempre es preferible para el avión recoger a todas las personas que van al mismo lado en vez de ir 1 por 1. Después se llama recursivamente.

- Ahora toca el caso de que el avión y la persona están en distinta ciudad pero los pasajeros son de distinto destino a ésta. Como antes, comprueba con las distancias entre ciudades, si nos conviene pasarnos a por él y si nos conviene vamos y los recogemos.
- Ya tenemos contemplados los casos de recoger personas por lo que ahora empieza a dejar viajeros en sus destinos.

Con este orden de preferencia he realizado esta tarea obteniendo unos resultados aceptables como veremos en los experimentos.

3. Para este apartado he añadido dos nuevos predicados, uno para indicar la duración que lleva cada uno de los aviones, y otro para indicar las horas máximas que puede realizar cada avión. También he creado dos nuevas funciones, una que me indica que un avión puede volar otra que me indica que no, comprobando sus horas de vuelo. Esto lo he hecho para cuando un avión no puede volar más, pero lleva viajeros, que los suelte para que otro avión que si le queden horas de vuelo, pueda llevarlos.

4.2. Experimentos:

1. **Representar problemas con varios aviones y varios límites de fuel, de manera que cada avión tiene una cantidad de fuel limitado a 4 viajes entre las 2 ciudades más distanciadas. Hacer experimentos Intentando obtener planes que gasten el menor fuel posible.**

En mi caso, los datos para mi problema son:

- **Personas:** 5 personas localizadas en: Gibraltar, Jaen, Madrid, Jaen, Granada, con destino: Barcelona, Madrid, Bilbao, Madrid, Madrid.
- **Aviones:** 3 aviones localizados en: Gibraltar, Córdoba, Bilbao.
- **Fuel-limit:** 5136.
- **Fuel:** 400, 1500, 600 respectivamente.
- **Capacidad:** 1500 ambos.
- **Pasajeros:** 5 ambos.
- **Max-duración:** 100000 ambos. Esto lo hago para que todos ellos puedan volar sin tener cuenta el límite de duración que lo veremos en otro apartado.

El **resultado** dado por mi implementación para este problema es:

<pre>:action (board p1 a1 gibraltar) start: 05/06/2007 08:00:00 end: 05/06/2007 09:00:00 :action (zoom a2 cordoba jaen) start: 05/06/2007 09:00:00 end: 05/06/2007 14:00:00 :action (board p2 a2 jaen) start: 05/06/2007 14:00:00 end: 05/06/2007 15:00:00 :action (board p4 a2 jaen) start: 05/06/2007 15:00:00 end: 05/06/2007 16:00:00 :action (refuel a3 bilbao) start: 05/06/2007 16:00:00 end: 13/07/2007 04:00:00 :action (zoom a3 bilbao madrid) start: 13/07/2007 04:00:00 end: 14/07/2007 00:00:00</pre>
--

```

:action (board p3 a3 madrid) start: 14/07/2007 00:00:00 end: 14/07/2007 01:00:00
:action (zoom a2 jaen granada) start: 14/07/2007 01:00:00 end: 14/07/2007 06:00:00
:action (board p5 a2 granada) start: 14/07/2007 06:00:00 end: 14/07/2007 07:00:00
:action (refuel a1 gibraltar) start: 14/07/2007 07:00:00 end: 29/08/2007 03:00:00
:action (fly a1 gibraltar barcelona) start: 29/08/2007 03:00:00 end: 02/09/2007 19:00:00
:action (debark p1 a1 barcelona) start: 02/09/2007 19:00:00 end: 02/09/2007 20:00:00
:action (zoom a2 granada madrid) start: 02/09/2007 20:00:00 end: 03/09/2007 17:00:00
:action (debark p2 a2 madrid) start: 03/09/2007 17:00:00 end: 03/09/2007 18:00:00
:action (debark p4 a2 madrid) start: 03/09/2007 18:00:00 end: 03/09/2007 19:00:00
:action (debark p5 a2 madrid) start: 03/09/2007 19:00:00 end: 03/09/2007 20:00:00
:action (refuel a3 madrid) start: 03/09/2007 20:00:00 end: 06/10/2007 18:00:00
:action (zoom a3 madrid bilbao) start: 06/10/2007 18:00:00 end: 07/10/2007 14:00:00
:action (debark p3 a3 bilbao) start: 07/10/2007 14:00:00 end: 07/10/2007 15:00:00

```

2. Representar problemas en los que haya límites de duración para cada uno de los aviones, Hacer experimentos intentando obtener planes lo más cortos posibles en duración.

En mi caso, los datos para mi problema son:

- **Personas:** 9 personas localizadas en: (ver el en problema adjuntado), con destino: (ver en el problema adjuntado)).
- **Aviones:** 3 aviones localizados en: Barcelona, Madrid, Sevilla.
- **Fuel-limit:** 5136.
- **Fuel:** 400, 800, 600 respectivamente.
- **Capacidad:** 1400, 1400, 1500 respectivamente.
- **Pasajeros:** 5 ambos.
- **Max-duración:** 200, 1000, 30 respectivamente. Con esto podemos comprobar cómo el tercer avión realizará menos vuelos debido a la duración.

El **resultado** dado por mi implementación para este problema es:

```

:action (board p1 a2 madrid) start: 05/06/2007 08:00:00 end: 05/06/2007 09:00:00
:action (board p2 a2 madrid) start: 05/06/2007 09:00:00 end: 05/06/2007 10:00:00
:action (zoom a3 sevilla gibraltar) start: 05/06/2007 10:00:00 end: 05/06/2007 20:00:00
:action (board p9 a3 gibraltar) start: 05/06/2007 20:00:00 end: 05/06/2007 21:00:00
:action (refuel a1 barcelona) start: 05/06/2007 21:00:00 end: 17/07/2007 13:00:00
:action (zoom a1 barcelona bilbao) start: 17/07/2007 13:00:00 end: 18/07/2007 20:00:00
:action (board p8 a1 bilbao) start: 18/07/2007 20:00:00 end: 18/07/2007 21:00:00
:action (board p7 a1 bilbao) start: 18/07/2007 21:00:00 end: 18/07/2007 22:00:00
:action (board p3 a2 madrid) start: 18/07/2007 22:00:00 end: 18/07/2007 23:00:00
:action (refuel a1 bilbao) start: 18/07/2007 23:00:00 end: 08/09/2007 15:00:00
:action (fly a1 bilbao cadiz) start: 08/09/2007 15:00:00 end: 13/09/2007 01:00:00
:action (board p6 a1 cadiz) start: 13/09/2007 01:00:00 end: 13/09/2007 02:00:00
:action (refuel a1 cadiz) start: 13/09/2007 02:00:00 end: 27/10/2007 04:00:00
:action (zoom a1 cadiz granada) start: 27/10/2007 04:00:00 end: 27/10/2007 19:00:00
:action (board p5 a1 granada) start: 27/10/2007 19:00:00 end: 27/10/2007 20:00:00
:action (zoom a1 granada jaen) start: 27/10/2007 20:00:00 end: 28/10/2007 01:00:00
:action (board p4 a1 jaen) start: 28/10/2007 01:00:00 end: 28/10/2007 02:00:00
:action (refuel a2 madrid) start: 28/10/2007 02:00:00 end: 22/11/2007 01:00:00

```

```

:action (zoom a2 madrid sevilla) start: 22/11/2007 01:00:00 end: 23/11/2007 04:00:00
:action (debark p2 a2 sevilla ) start: 23/11/2007 04:00:00 end: 23/11/2007 05:00:00
:action (debark p1 a2 sevilla ) start: 23/11/2007 05:00:00 end: 23/11/2007 06:00:00
:action (zoom a1 jaen malaga) start: 23/11/2007 06:00:00 end: 23/11/2007 16:00:00
:action (debark p4 a1 malaga) start: 23/11/2007 16:00:00 end: 23/11/2007 17:00:00
:action (refuel a1 malaga) start: 23/11/2007 17:00:00 end: 12/01/2008 01:00:00
:action (fly a1 malaga bilbao) start: 12/01/2008 01:00:00 end: 15/01/2008 23:00:00
:action (debark p5 a1 bilbao) start: 15/01/2008 23:00:00 end: 16/01/2008 00:00:00
:action (debark p6 a1 bilbao) start: 16/01/2008 00:00:00 end: 16/01/2008 01:00:00
:action (debark p7 a1 bilbao) start: 16/01/2008 01:00:00 end: 16/01/2008 02:00:00
:action (debark p8 a1 bilbao) start: 16/01/2008 02:00:00 end: 16/01/2008 03:00:00
:action (refuel a2 sevilla ) start: 16/01/2008 03:00:00 end: 29/02/2008 15:00:00
:action (fly a2 sevilla bilbao) start: 29/02/2008 15:00:00 end: 04/03/2008 12:00:00
:action (board p8 a2 bilbao) start: 04/03/2008 12:00:00 end: 04/03/2008 13:00:00
:action (board p7 a2 bilbao) start: 04/03/2008 13:00:00 end: 04/03/2008 14:00:00
:action (board p6 a2 bilbao) start: 04/03/2008 14:00:00 end: 04/03/2008 15:00:00
:action (refuel a2 bilbao) start: 04/03/2008 15:00:00 end: 12/04/2008 13:00:00
:action (zoom a2 bilbao madrid) start: 12/04/2008 13:00:00 end: 13/04/2008 09:00:00
:action (debark p6 a2 madrid) start: 13/04/2008 09:00:00 end: 13/04/2008 10:00:00
:action (debark p7 a2 madrid) start: 13/04/2008 10:00:00 end: 13/04/2008 11:00:00
:action (debark p8 a2 madrid) start: 13/04/2008 11:00:00 end: 13/04/2008 12:00:00
:action (refuel a2 madrid) start: 13/04/2008 12:00:00 end: 16/05/2008 10:00:00
:action (zoom a2 madrid barcelona) start: 16/05/2008 10:00:00 end: 17/05/2008 17:00:00
:action (debark p3 a2 barcelona) start: 17/05/2008 17:00:00 end: 17/05/2008 18:00:00
:action (refuel a3 gibraltar ) start: 17/05/2008 18:00:00 end: 11/07/2008 00:00:00
:action (zoom a3 gibraltar jaen) start: 11/07/2008 00:00:00 end: 11/07/2008 17:00:00
:action (debark p9 a3 jaen) start: 11/07/2008 17:00:00 end: 11/07/2008 18:00:00

```

3. Representar problemas en los que varíe el número de acciones totales para transportar un número considerable de personas (más de 20, por ejemplo), intentar obtener planes que tengan el menor número posible de acciones.

En mi caso, los datos para mi problema son:

- **Personas:** 25 personas localizadas en: (ver el en problema adjuntado), con destino: (ver en el problema adjuntado)).
- **Aviones:** 6 aviones localizados en: (ver en el problema adjuntado).
- **Fuel-limit:** 5136.
- **Fuel:** (ver en el problema adjuntado).
- **Capacidad:** 1500 todos.
- **Pasajeros:** 5 todos.
- **Max-duración:** Todos 700 menos el avion 2 que tiene 200.

El **resultado** dado por mi implementación para este problema es:

```

:action (board p3 a3 madrid) start: 05/06/2007 08:00:00 end: 05/06/2007 09:00:00
:action (board p4 a1 jaen) start: 05/06/2007 09:00:00 end: 05/06/2007 10:00:00
:action (board p5 a1 jaen) start: 05/06/2007 10:00:00 end: 05/06/2007 11:00:00
:action (board p6 a1 jaen) start: 05/06/2007 11:00:00 end: 05/06/2007 12:00:00
:action (board p8 a6 bilbao) start: 05/06/2007 12:00:00 end: 05/06/2007 13:00:00

```


:action (board p10 a5 cadiz) start: 05/06/2007 13:00:00 end: 05/06/2007 14:00:00
 :action (board p11 a2 barcelona) start: 05/06/2007 14:00:00 end: 05/06/2007 15:00:00
 :action (board p1 a4 madrid) start: 05/06/2007 15:00:00 end: 05/06/2007 16:00:00
 :action (board p2 a4 madrid) start: 05/06/2007 16:00:00 end: 05/06/2007 17:00:00
 :action (board p9 a6 bilbao) start: 05/06/2007 17:00:00 end: 05/06/2007 18:00:00
 :action (board p12 a2 barcelona) start: 05/06/2007 18:00:00 end: 05/06/2007 19:00:00
 :action (board p15 a5 cadiz) start: 05/06/2007 19:00:00 end: 05/06/2007 20:00:00
 :action (refuel a3 madrid) start: 05/06/2007 20:00:00 end: 13/07/2007 08:00:00
 :action (zoom a3 madrid jaen) start: 13/07/2007 08:00:00 end: 14/07/2007 01:00:00
 :action (board p7 a3 jaen) start: 14/07/2007 01:00:00 end: 14/07/2007 02:00:00
 :action (refuel a6 bilbao) start: 14/07/2007 02:00:00 end: 20/08/2007 14:00:00
 :action (fly a6 bilbao granada) start: 20/08/2007 14:00:00 end: 24/08/2007 01:00:00
 :action (board p13 a6 granada) start: 24/08/2007 01:00:00 end: 24/08/2007 02:00:00
 :action (board p14 a6 granada) start: 24/08/2007 02:00:00 end: 24/08/2007 03:00:00
 :action (zoom a5 cadiz sevilla) start: 24/08/2007 03:00:00 end: 24/08/2007 09:00:00
 :action (board p16 a5 sevilla) start: 24/08/2007 09:00:00 end: 24/08/2007 10:00:00
 :action (board p17 a5 sevilla) start: 24/08/2007 10:00:00 end: 24/08/2007 11:00:00
 :action (zoom a6 granada gibraltar) start: 24/08/2007 11:00:00 end: 25/08/2007 00:00:00
 :action (board p18 a6 gibraltar) start: 25/08/2007 00:00:00 end: 25/08/2007 01:00:00
 :action (refuel a2 barcelona) start: 25/08/2007 01:00:00 end: 19/09/2007 01:00:00
 :action (fly a2 barcelona huelva) start: 19/09/2007 01:00:00 end: 23/09/2007 19:00:00
 :action (debark p11 a2 huelva) start: 23/09/2007 19:00:00 end: 23/09/2007 20:00:00
 :action (board p22 a2 huelva) start: 23/09/2007 20:00:00 end: 23/09/2007 21:00:00
 :action (board p23 a2 huelva) start: 23/09/2007 21:00:00 end: 23/09/2007 22:00:00
 :action (refuel a5 sevilla) start: 23/09/2007 22:00:00 end: 10/11/2007 21:00:00
 :action (zoom a5 sevilla almeria) start: 10/11/2007 21:00:00 end: 11/11/2007 17:00:00
 :action (board p24 a5 almeria) start: 11/11/2007 17:00:00 end: 11/11/2007 18:00:00
 :action (zoom a3 jaen almeria) start: 11/11/2007 18:00:00 end: 12/11/2007 05:00:00
 :action (board p25 a3 almeria) start: 12/11/2007 05:00:00 end: 12/11/2007 06:00:00
 :action (refuel a3 almeria) start: 12/11/2007 06:00:00 end: 28/12/2007 12:00:00
 :action (zoom a3 almeria gibraltar) start: 28/12/2007 12:00:00 end: 29/12/2007 05:00:00
 :action (board p19 a3 gibraltar) start: 29/12/2007 05:00:00 end: 29/12/2007 06:00:00
 :action (zoom a3 gibraltar malaga) start: 29/12/2007 06:00:00 end: 29/12/2007 13:00:00
 :action (board p20 a3 malaga) start: 29/12/2007 13:00:00 end: 29/12/2007 14:00:00
 :action (refuel a2 huelva) start: 29/12/2007 14:00:00 end: 15/02/2008 02:00:00
 :action (zoom a2 huelva malaga) start: 15/02/2008 02:00:00 end: 15/02/2008 17:00:00
 :action (board p21 a2 malaga) start: 15/02/2008 17:00:00 end: 15/02/2008 18:00:00
 :action (refuel a3 malaga) start: 15/02/2008 18:00:00 end: 26/03/2008 04:00:00
 :action (fly a3 malaga barcelona) start: 26/03/2008 04:00:00 end: 30/03/2008 09:00:00
 :action (debark p3 a3 barcelona) start: 30/03/2008 09:00:00 end: 30/03/2008 10:00:00
 :action (debark p25 a3 barcelona) start: 30/03/2008 10:00:00 end: 30/03/2008 11:00:00
 :action (debark p7 a3 barcelona) start: 30/03/2008 11:00:00 end: 30/03/2008 12:00:00
 :action (zoom a2 malaga jaen) start: 30/03/2008 12:00:00 end: 30/03/2008 22:00:00
 :action (debark p21 a2 jaen) start: 30/03/2008 22:00:00 end: 30/03/2008 23:00:00
 :action (refuel a3 barcelona) start: 30/03/2008 23:00:00 end: 11/05/2008 12:00:00
 :action (fly a3 barcelona granada) start: 11/05/2008 12:00:00 end: 15/05/2008 03:00:00
 :action (debark p20 a3 granada) start: 15/05/2008 03:00:00 end: 15/05/2008 04:00:00
 :action (zoom a3 granada almeria) start: 15/05/2008 04:00:00 end: 15/05/2008 12:00:00
 :action (debark p19 a3 almeria) start: 15/05/2008 12:00:00 end: 15/05/2008 13:00:00
 :action (refuel a5 almeria) start: 15/05/2008 13:00:00 end: 18/06/2008 17:00:00
 :action (zoom a5 almeria huelva) start: 18/06/2008 17:00:00 end: 19/06/2008 18:00:00
 :action (debark p24 a5 huelva) start: 19/06/2008 18:00:00 end: 19/06/2008 19:00:00
 :action (debark p10 a5 huelva) start: 19/06/2008 19:00:00 end: 19/06/2008 20:00:00
 :action (refuel a2 jaen) start: 19/06/2008 20:00:00 end: 31/07/2008 20:00:00
 :action (fly a2 jaen bilbao) start: 31/07/2008 20:00:00 end: 03/08/2008 21:00:00
 :action (debark p23 a2 bilbao) start: 03/08/2008 21:00:00 end: 03/08/2008 22:00:00

```

:action (debark p22 a2 bilbao) start: 03/08/2008 22:00:00 end: 03/08/2008 23:00:00
:action (debark p12 a2 bilbao) start: 03/08/2008 23:00:00 end: 04/08/2008 00:00:00
:action (refuel a1 jaen) start: 04/08/2008 00:00:00 end: 14/09/2008 16:00:00
:action (fly a1 jaen bilbao) start: 14/09/2008 16:00:00 end: 17/09/2008 17:00:00
:action (board p12 a1 bilbao) start: 17/09/2008 17:00:00 end: 17/09/2008 18:00:00
:action (board p22 a1 bilbao) start: 17/09/2008 18:00:00 end: 17/09/2008 19:00:00
:action (refuel a1 bilbao) start: 17/09/2008 19:00:00 end: 18/10/2008 05:00:00
:action (zoom a1 bilbao madrid) start: 18/10/2008 05:00:00 end: 19/10/2008 01:00:00
:action (debark p22 a1 madrid) start: 19/10/2008 01:00:00 end: 19/10/2008 02:00:00
:action (debark p12 a1 madrid) start: 19/10/2008 02:00:00 end: 19/10/2008 03:00:00
:action (refuel a6 gibraltar) start: 19/10/2008 03:00:00 end: 13/12/2008 21:00:00
:action (fly a6 gibraltar barcelona) start: 13/12/2008 21:00:00 end: 18/12/2008 13:00:00
:action (debark p18 a6 barcelona) start: 18/12/2008 13:00:00 end: 18/12/2008 14:00:00
:action (debark p8 a6 barcelona) start: 18/12/2008 14:00:00 end: 18/12/2008 15:00:00
:action (refuel a5 huelva) start: 18/12/2008 15:00:00 end: 29/01/2009 17:00:00
:action (fly a5 huelva bilbao) start: 29/01/2009 17:00:00 end: 02/02/2009 15:00:00
:action (debark p17 a5 bilbao) start: 02/02/2009 15:00:00 end: 02/02/2009 16:00:00
:action (refuel a5 bilbao) start: 02/02/2009 16:00:00 end: 13/03/2009 19:00:00
:action (fly a5 bilbao gibraltar) start: 13/03/2009 19:00:00 end: 18/03/2009 10:00:00
:action (debark p16 a5 gibraltar) start: 18/03/2009 10:00:00 end: 18/03/2009 11:00:00
:action (debark p15 a5 gibraltar) start: 18/03/2009 11:00:00 end: 18/03/2009 12:00:00
:action (refuel a6 barcelona) start: 18/03/2009 12:00:00 end: 04/05/2009 09:00:00
:action (zoom a6 barcelona madrid) start: 04/05/2009 09:00:00 end: 05/05/2009 16:00:00
:action (debark p14 a6 madrid) start: 05/05/2009 16:00:00 end: 05/05/2009 17:00:00
:action (refuel a6 madrid) start: 05/05/2009 17:00:00 end: 26/06/2009 11:00:00
:action (zoom a6 madrid jaen) start: 26/06/2009 11:00:00 end: 27/06/2009 04:00:00
:action (debark p13 a6 jaen) start: 27/06/2009 04:00:00 end: 27/06/2009 05:00:00
:action (debark p9 a6 jaen) start: 27/06/2009 05:00:00 end: 27/06/2009 06:00:00
:action (refuel a4 madrid) start: 27/06/2009 06:00:00 end: 03/08/2009 18:00:00
:action (zoom a4 madrid sevilla) start: 03/08/2009 18:00:00 end: 04/08/2009 21:00:00
:action (debark p2 a4 sevilla) start: 04/08/2009 21:00:00 end: 04/08/2009 22:00:00
:action (debark p1 a4 sevilla) start: 04/08/2009 22:00:00 end: 04/08/2009 23:00:00
:action (refuel a1 madrid) start: 04/08/2009 23:00:00 end: 06/09/2009 21:00:00
:action (zoom a1 madrid cadiz) start: 06/09/2009 21:00:00 end: 08/09/2009 06:00:00
:action (debark p6 a1 cadiz) start: 08/09/2009 06:00:00 end: 08/09/2009 07:00:00
:action (debark p5 a1 cadiz) start: 08/09/2009 07:00:00 end: 08/09/2009 08:00:00
:action (debark p4 a1 cadiz) start: 08/09/2009 08:00:00 end: 08/09/2009 09:00:00

```