

FAST-DOWNWARD

Daniel Granizo Sánchez.

En este documento voy a presentar el planner Fast-downward, se trata de un sistema de planning clásico basado en una búsqueda heurística. Se trata de un planner que realiza una búsqueda progresiva hacia delante. Es capaz de soportar gran parte de PDDL 2.2 incluso algunas características de PDDL 3.1 como action-costs. Sin embargo este planner no soporta durative-actions por lo que no será posible usarlas en nuestros dominios. Además cuenta con algunas limitaciones como no soportar todos los axiomas o las universal conditions. Todo ello lo podemos ver en el siguiente enlace: <http://www.fast-downward.org/PddlSupport> .

Este planner es relativamente fácil de instalar y utilizar. Para ello debemos antes que nada comprobar nuestra versión de python en el ordenador ya que será necesario que sea mayor o igual a la versión 3.6. (Toda esta explicación es aplicable a Linux, no explicaré para otros sistemas operativos).

Primero comprobamos la versión de python y si es necesario instalaremos nuevos paquetes:

```
sudo apt install cmake g++ git make python3
```

Después debemos dirigirnos a un directorio donde queramos que se cree la carpeta donde estarán los paquetes del planner y después clonar su repositorio de la siguiente forma:

```
git clone https://github.com/aibasel/downward.git DIRNAME
```

Nuestro siguiente paso será compilar el planner, para ello nos dirigimos a la carpeta del planner y ejecutamos el build del programa:

```
cd downward  
./build.py
```

Una vez terminado el fichero de compilado podremos comenzar a diseñar nuestros dominios y problemas en PDDL para que los resuelva el planner.

Pongamos que en este caso hemos definido un fichero **prueba_dominio.pddl** y un **problema1.pddl**. Para poder ejecutar el planner bastará con utilizar el comando:

```
./fast-downward.py prueba_dominio.pddl problema1.pddl
```

Este comando hará que se planifiquen las tareas según el dominio y el problema seleccionados. Hay que tener en cuenta que la solución aparecerá en un fichero creado en el mismo directorio donde estemos trabajando con nombre **output.sas** ahí podremos comprobar qué acciones ha llevado a cabo el planificador.

Otra forma de ejecutar el planificar si queremos utilizarlo con un determinado algoritmo de búsqueda como puede ser A* y una determinada heurística es realizarlo de la siguiente manera:

```
./fast-downward.py prueba_dominio.pddl problema1.pddl --search "astar(blind())"
```

Referencias:

- <https://planning.wiki/ref/planners/fd>
- <http://www.fast-downward.org/ObtainingAndRunningFastDownward>
- <http://www.fast-downward.org/PddlSupport>
- <http://www.fast-downward.org/PlannerUsage>
- <http://www.fast-downward.org/PddlSupport>
- <http://csci431.artifice.cc/notes/pddl.html>