

# Algoritmo de búsqueda Bidireccional

Stalin Muñoz Gutiérrez

Centro de Ciencias de la Complejidad  
Universidad Nacional Autónoma de México (UNAM)

Hoy veremos el algoritmo de búsqueda bidireccional.  
Este es una variación interesante del algoritmo BFS.  
Para poder aplicar el algoritmo vamos a requerir de condiciones adicionales.

Una condición es conocer el estado meta. Si no conocemos el estado meta o hay varios estados meta, no podemos aplicar el algoritmo directamente.

La otra condición es que para cualquier estado podamos conocer el conjunto de acciones y estados de los que es sucesor.

# Algoritmo de búsqueda bidireccional

Algoritmo basado en BFS

Dos búsquedas por turnos.

## Algoritmo de búsqueda Bidireccional

2018-09-25

└ Algoritmo de búsqueda bidireccional

Algoritmo de búsqueda bidireccional

Algoritmo basado en BFS  
Dos búsquedas por turnos.

El algoritmo de búsqueda bidireccional, es en realidad dos búsquedas por turnos.

Las búsquedas podrían hacerse de manera paralela, pero requeriría modificar ligeramente el algoritmo que vamos a presentar.

# Algoritmo de búsqueda bidireccional

## Algoritmo basado en BFS

Dos búsquedas por turnos.

- Una búsqueda progresiva o hacia adelante comenzando en el nodo inicial.

## Algoritmo de búsqueda Bidireccional

### └ Algoritmo de búsqueda bidireccional

2018-09-25

- Una búsqueda progresiva o hacia adelante comenzando en el nodo inicial.

La primera de las búsquedas es una búsqueda progresiva tipo BFS que comienza con el nodo inicial.

# Algoritmo de búsqueda bidireccional

## Algoritmo basado en BFS

Dos búsquedas por turnos.

- Una búsqueda progresiva o hacia adelante comenzando en el nodo inicial.
- Otra búsqueda regresiva o hacia atrás comenzando con el nodo meta.

## Algoritmo de búsqueda Bidireccional

2018-09-25

### Algoritmo de búsqueda bidireccional

Algoritmo de búsqueda bidireccional

Algoritmo basado en BFS

Dos búsquedas por turnos.

- Una búsqueda progresiva o hacia adelante comenzando en el nodo inicial.
- Otra búsqueda regresiva o hacia atrás comenzando con el nodo meta.

La segunda búsqueda es una búsqueda regresiva o hacia atrás tipo BFS empezando del nodo meta.

# Algoritmo de búsqueda bidireccional

## Algoritmo basado en BFS

Dos búsquedas por turnos.

- Una búsqueda progresiva o hacia adelante comenzando en el nodo inicial.
- Otra búsqueda regresiva o hacia atrás comenzando con el nodo meta.
- El algoritmo termina cuando las búsquedas se encuentran a la mitad del camino.

## Algoritmo de búsqueda Bidireccional

2018-09-25

## Algoritmo de búsqueda bidireccional

Algoritmo de búsqueda bidireccional

Algoritmo basado en BFS

Dos búsquedas por turnos.

- Una búsqueda progresiva o hacia adelante comenzando en el nodo inicial.
- Otra búsqueda regresiva o hacia atrás comenzando con el nodo meta.
- El algoritmo termina cuando las búsquedas se encuentran a la mitad del camino.

Cada dirección de exploración tiene una frontera de búsqueda.

El objetivo de cada exploración es descubrir un estado que es común a las dos fronteras.

Cuando dicha intersección se encuentra se puede conformar la ruta total a partir de las rutas parciales al estado común.

# Búsqueda bidireccional

estado inicial



meta



Búsqueda progresiva

Búsqueda regresiva

## Algoritmo de búsqueda Bidireccional

2018-09-25

### Búsqueda bidireccional

Búsqueda bidireccional

estado inicial



meta

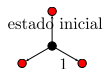


Búsqueda progresiva

Búsqueda regresiva

El estado inicial se agrega a la frontera de la búsqueda progresiva.  
El estado meta se agrega a la frontera de la búsqueda regresiva.

# Búsqueda bidireccional



Búsqueda progresiva



Búsqueda regresiva

## Búsqueda bidireccional

- La búsqueda progresiva expande el nodo inicial.
- La nueva frontera son todos los estados a profundidad 1.
- La búsqueda progresiva busca que su frontera se intersecte con la frontera de la búsqueda regresiva.
- Esto no ha pasado aún.

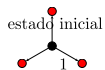


Búsqueda progresiva



Búsqueda regresiva

# Búsqueda bidireccional



Búsqueda progresiva



Búsqueda regresiva

## Algoritmo de búsqueda Bidireccional

### Búsqueda bidireccional

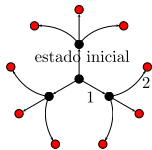
Ahora es el turno de la búsqueda regresiva.  
 Si hacemos las búsquedas por turnos garantizamos que al intersectarse las fronteras, tendremos la solución óptima.  
 La búsqueda regresiva expande el estado meta.





- ↳ Búsqueda bidireccional

2018-09-25

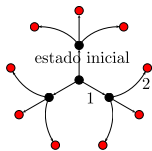


Búsqueda progresiva

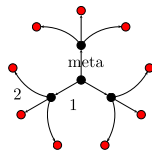


Búsqueda regresiva

# Búsqueda bidireccional



Búsqueda progresiva



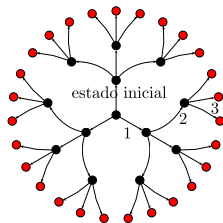
Búsqueda regresiva

## Búsqueda bidireccional

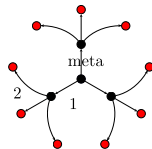


Ahora es el turno de la búsqueda regresiva.  
Agregamos todos los estados a profundidad 2.  
Aún no se intersectan las fronteras.

## Búsqueda bidireccional



Búsqueda progresiva



Búsqueda regresiva

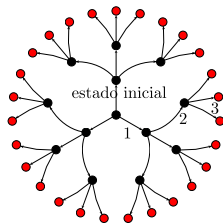
- ↳ Búsqueda bidireccional

Hacia adelante con profundidad 3.

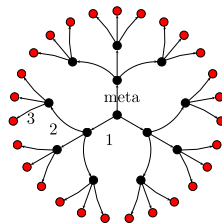
*Elisoperla pygmaea**Elisapetia regalis*

- ↳ Búsqueda bidireccional

Hacia atrás con profundidad 3.

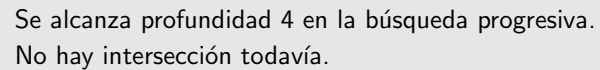


Búsqueda progresiva



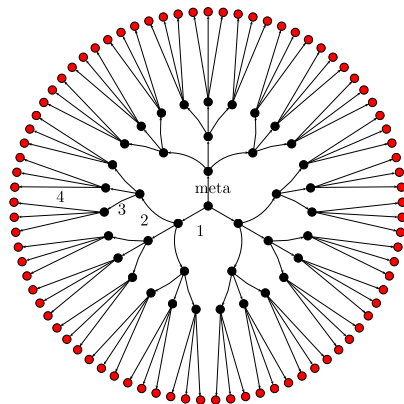
Búsqueda regresiva

- ↳ Búsqueda bidireccional



Búsqueda regresiva

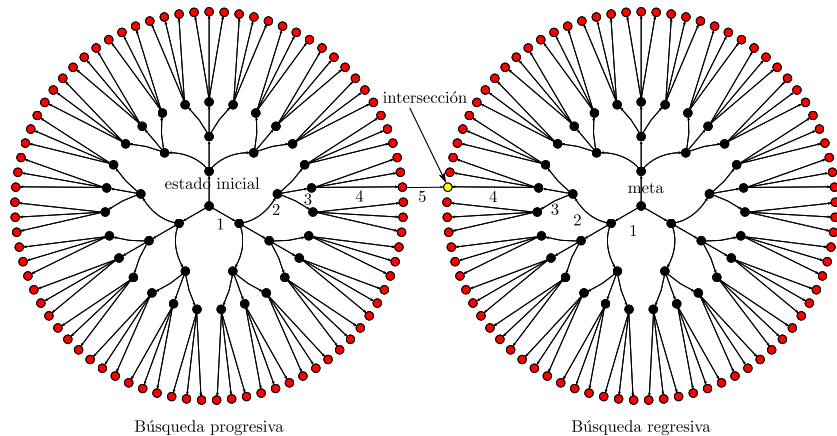
- ↳ Búsqueda bidireccional



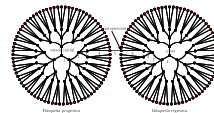
Búsqueda regresiva

La búsqueda regresiva también alcanza profundidad 4.  
Sin intersección.

## Búsqueda bidireccional

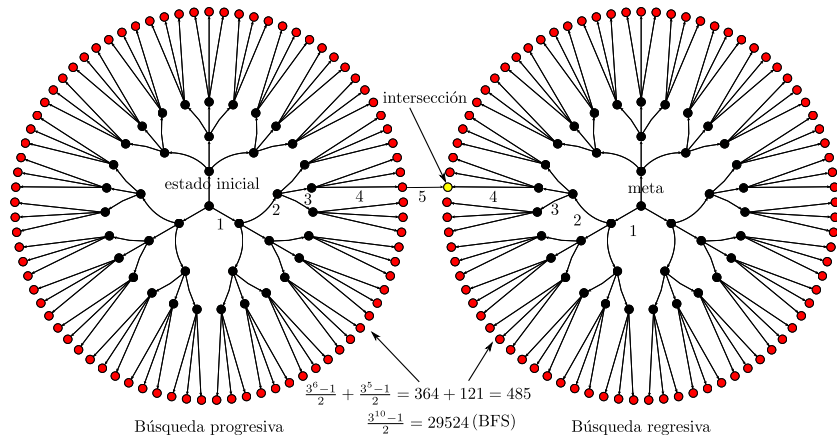


### Búsqueda bidireccional



Turno de la búsqueda hacia adelante.  
 No mostramos todos los nodos a profundidad 5.  
 Únicamente el nodo de intersección entre las dos fronteras.

# Búsqueda bidireccional

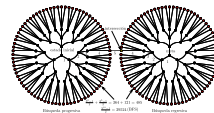


## Algoritmo de búsqueda Bidireccional

2018-09-25

### Búsqueda bidireccional

Búsqueda bidireccional



Considerando que la intersección se encuentra tras expandir todos los estados a profundidad 5 en la frontera de la búsqueda progresiva, y sumando el total de estados expandidos con el total de ambas fronteras,

tenemos que hemos expandido poco menos de 500 estados.

Dado que en este ejemplo la solución está a profundidad 9, un algoritmo BFS convencional habría expandido cerca de 30 mil estados para encontrar la solución.

Esto es un ahorro significativo en memoria.



# Algoritmo Bidireccional



## Algoritmo de búsqueda Bidireccional

2018-09-25

### └ Algoritmo Bidireccional

Algoritmo Bidireccional



Explicamos el algoritmo Bidireccional en el pizarrón.

## Análisis Asintótico del algoritmo Bidireccional

### Algoritmo Bidireccional

#### 1. Memoria.

- $O(b^{d/2})$

#### 2. Tiempo.

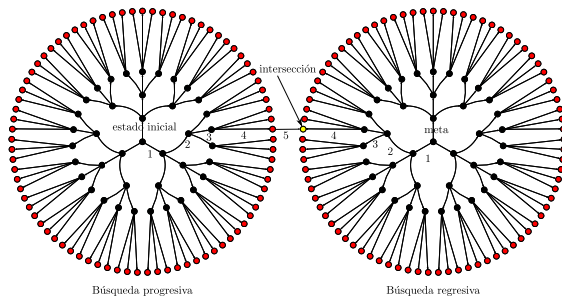
- $O(b^{d/2})$

#### 3. Calidad.

- Solución óptima.

#### 4. Completez.

- Completo.



El ahorro de memoria del algoritmo Bidireccional es significativo.

## Algoritmo de búsqueda Bidireccional

2018-09-25

### Análisis Asintótico del algoritmo Bidireccional

#### Análisis Asintótico del algoritmo Bidireccional

##### Algoritmo Bidireccional

##### 1. Memoria.

- $O(b^{d/2})$

##### 2. Tiempo.

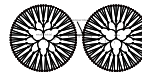
- $O(b^{d/2})$

##### 3. Calidad.

- Solución óptima.

##### 4. Completez.

- Completo.



El ahorro de memoria del algoritmo Bidireccional es significativo.

El desempeño del algoritmo bidireccional es muy bueno.

A pesar de utilizar memoria exponencial, al dividirse la búsqueda en dos exploraciones por turnos, el crecimiento de memoria es  $O(b^{d/2})$ .

Este ahorro es sustancial.

Respecto del tiempo, crece a la par con la memoria.

El algoritmo nos da la solución óptima, y es completo.

Para problemas grandes, aún este ahorro de memoria, puede no ser suficiente.

La desventaja más notable es que el algoritmo no es tan general como los otros.

Se requiere de conocer el estado meta y también de poder ir hacia atrás de los estados para recuperar los estados predecesores.