

1. *Árbol de Intervalos*

1.1. Ideas para la elección de la estructura del *Árbol de Intervalos*

1.1.1. Dos Árboles

La primera idea que se nos ocurrió para extender el árbol de intervalos a dos dimensiones, fue simplemente tener dos árboles (uno para cada coordenada).

La búsqueda entonces consistía en buscar las imágenes correspondientes a la coordenada x y luego las imágenes correspondientes a la coordenada y . Por último, debíamos encontrar las imágenes que aparecieran en los dos conjuntos.

1.1.2. Árbol con ambas coordenadas

Otra idea fue, hacer un solo árbol en el que los “Intervalos Elementales” se construyeran con todos los puntos de inicio y finalización de las coordenadas x e y . En cada nodo de este árbol se encontraban un arreglo (o alguna otra estructura) con las imágenes que le correspondían en x y otro arreglo con las imágenes que le correspondían en y .

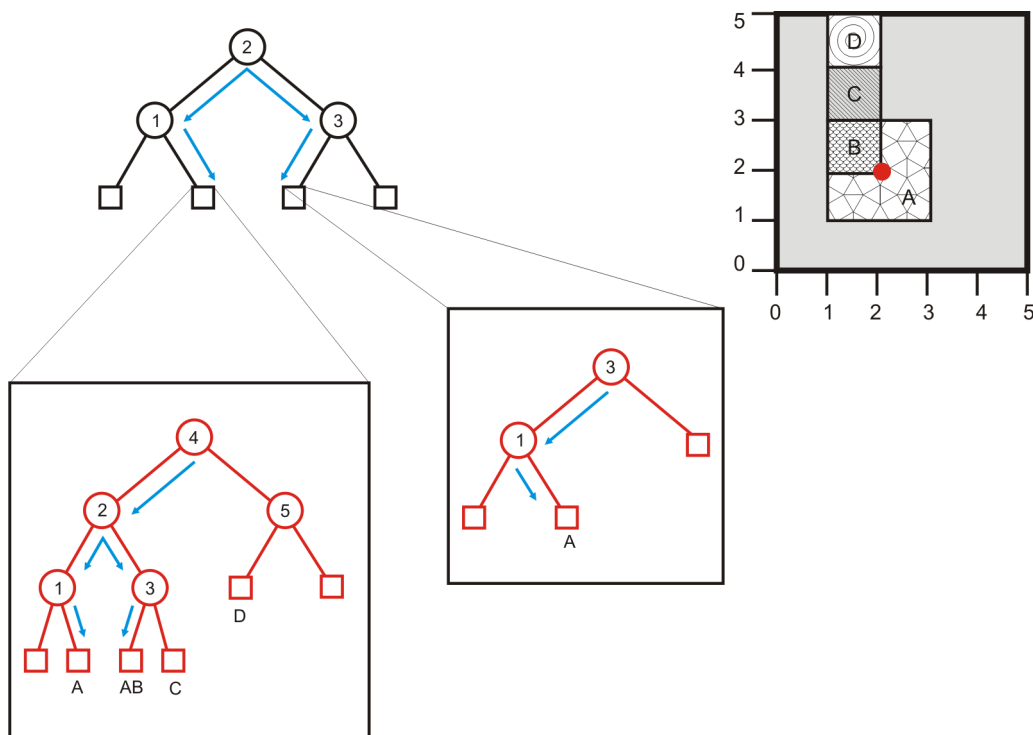
De esta manera, si había coincidencias entre los puntos de x y los puntos de y , reducíamos espacialmente la cantidad de nodos. La consulta consistía en buscar cada coordenada independientemente de la otra, como en un árbol convencional, tomando como arreglo asociado al nodo el correspondiente a la coordenada deseada en cada caso.

1.1.3. Árbol de árboles

Otra forma de extender el árbol de intervalos a 2 dimensiones era la siguiente:

Tener un árbol de intervalos en una coordenada y en cada nodo tener un árbol de intervalos con las imágenes correspondientes a ese nodo ordenadas por la otra coordenada.

Aquí hay un pequeño ejemplo para facilitar la comprensión:



1.1.4. Árbol único

La última idea es mucho más simple que todas las anteriores. Se basa en tener un sólo árbol con una de las coordenadas.

Haremos primero la búsqueda en este árbol. Luego, de los resultados obtenidos, verificamos (consultando la lista de todas las imágenes) cuáles coinciden también en la otra coordenada.

1.2. Elección de la estructura del *Árbol de Intervalos*

Nos decidimos por implementar la idea del árbol único, por los siguientes motivos:

Sabíamos que la búsqueda de un punto en un árbol de una sola coordenada era de $O(\log(n) + k)$

1.2.1. “Dos Árboles”

En “Dos Árboles” la costo espacial y temporal de la construcción era cercano al doble del “Árbol único”, (pues tiene un árbol más).

En la búsqueda, el “Árbol único” también resolvía el problema en aproximadamente la mitad del tiempo:

En el “Árbol único” la búsqueda se realiza en $\log(n) + k'$ (donde k' son las imágenes que coinciden en x) más k' para verificar si coincide la coordenada y .

En cambio en el “Dos Árboles” la búsqueda sería $\log(n) + k' + \log(n) + k''$ (donde k'' son las imágenes que coinciden en y) más $\min(k', k'')$ para verificar coincidencias.

NOTA: Para poder hacer estas verificaciones en tiempo lineal, en la estructura del *Árbol de Intervalos* tenemos un arreglo de bool.

1.2.2. “Árbol con ambas coordenadas”

En “Árbol con ambas coordenadas” el costo espacial y temporal de la construcción, podía disminuir solamente si existía gran cantidad de coincidencias entre los puntos de x y de y , pero de todos modos aunque la coincidencia sea absoluta no tendremos menos nodos que en el “Árbol único”. En caso de que no existan muchas coincidencias la cantidad de los nodos será cercana el doble que en el “Árbol único”.
NOTA: Además cada nodo en el “Árbol con ambas coordenadas” tiene un arreglo más.

Vimos que en el “Árbol único” la búsqueda se realiza en $\log(n) + k' + k'$, en el “Árbol con ambas coordenadas”, en peor caso, se realizaría en $2\log(2n) + k' + k'' + \min(k', k'')$, que es trivialmente mayor.

1.2.3. “Árbol de árboles”

En cuanto al costo espacial de la construcción, tanto en “Árbol único” como en “Árbol de árboles”, tenemos un árbol de n nodos. La diferencia entre estos 2 árboles es que en “Árbol único” cada nodo cuenta con un arreglo con índices de imágenes, mientras que en cada nodo del “Árbol de árboles” hay un árbol de intervalos con esta información, el cual es más costoso espacialmente que un arreglo. En cuanto al costo temporal de la construcción, para hacer el “Árbol de árboles” primero necesitamos construir un “Árbol único”, para recién entonces construir cada subárbol.

En el “Árbol de árboles” el costo temporal de la búsqueda es $\log(n)$ (para recorrer el camino correspondiente en la coordenada x) más $\sum k_i$ (donde k_i es la cantidad de imágenes que se encuentran en el nodo i) más k (que son las imágenes que devuelve el algoritmo).