

Determinar que paquete envía el nodo Source en su primer enlace.

La base de esta idea está en el hecho de que los nodos sumideros tienen un flujo de entrada de 2.

Un nodo sumidero debe al final recibir una de las siguientes 3 opciones:

- A y B
- A/B y A
- A/B y B

Esto puede extenderse a los nodos codificadores, pero para los nodos codificadores siempre deben recibir A y B para enviar  $A+B(xOR)$

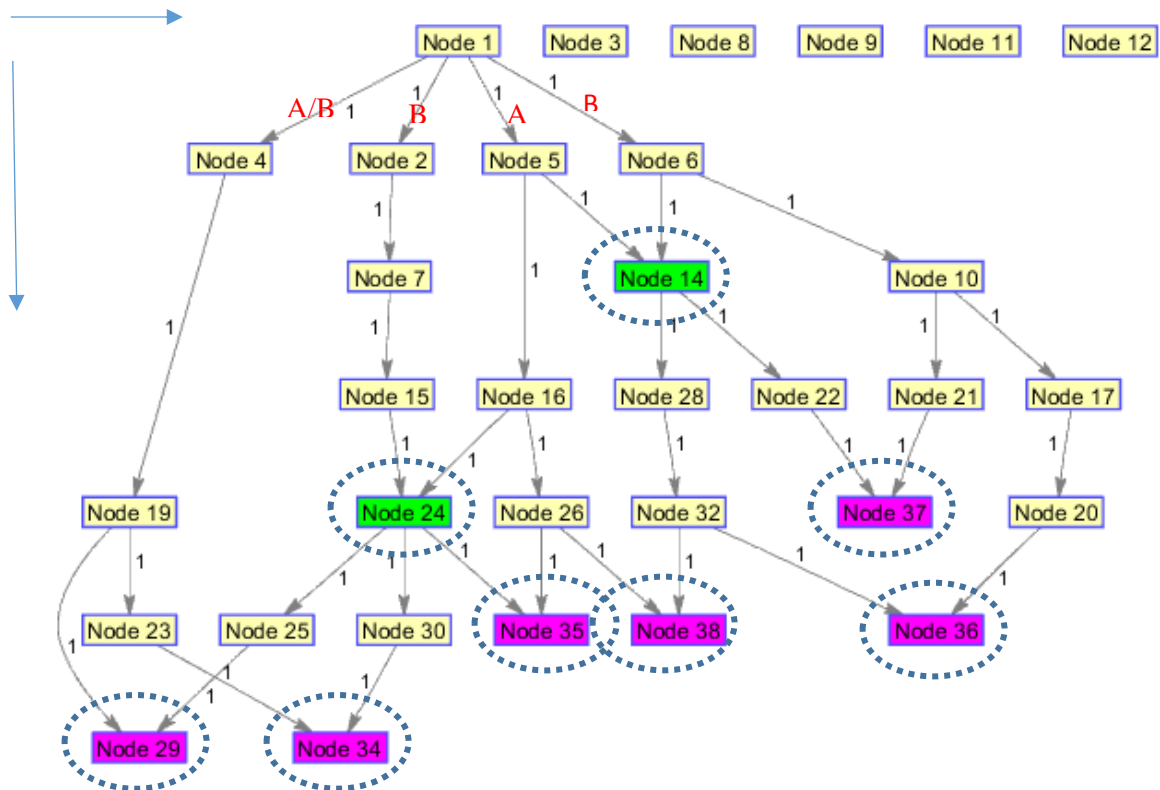
Por otra parte, luego de realizar Ford Fulkerson y combinar los grafos obtenidos se obtiene un grafo que se puede dividir por niveles.

La idea para determinar los paquetes enviados es básicamente:

- Explorar descendentemente por niveles el grafo .
- Cada nivel se examina nodo por nodo.
- Si un Nodo es Destino(Des) o Codificador(Cod), se marca y se explora hacia arriba hasta el nodo Source.
- Si el nodo es Destino:
  - El Nodo Des. Solo tiene 2 caminos hasta source.
  - Si en ninguno de los 2 caminos se pasa por un nodo codificador, entonces se marcan los enlaces  $S \rightarrow N^*$  con A y B respectivamente
  - Si por alguno de los 2 caminos hay un nodo codificador, se marca el camino sin nodo Codificador con A/B (no representa que el nodo Source envía A+B sino que puede enviar A o B no esta restringido todavía)
  - **NO** es posible que ambos caminos pasen por un nodo codificador.
- Si el nodo es Codificador
  - Es posible que el nodo Codificador tenga multiples caminos hacia Source, pero, en teoría, un nodo Codificador solo necesita recibir A y B a partir de dos caminos, si existen mas caminos estos pueden ser desechados.
  - Asegurar A y B con los  $S \rightarrow N$  enlazados al nodo codificador, tratándolo como un destino
  - No he visto que pase en ningún ejemplo pero, si hay un nodo codificador en cualquiera de los caminos hacia Source, sería un error? O simplemente funciona reenviando el A+B que recibe de este (\* no he encontrado que esto pase, no se si no es posible)
- Esto se hace hasta que, todos los  $S \rightarrow N$  estén marcados con A o B (A/B significa que no esta restringido por tanto puede seguir), o se pase por todos los nodos **Destino**.

\* $S \rightarrow N$ : Son los enlaces desde el nodo Source hacia un Nodo del primer Nivel.

## Ejemplo



Este es el grafo obtenido al final del ejemplo 38 nodos.

Empezamos explorando hacia abajo y a la derecha, el primer nodo que nos encontramos es el Cod. N14, desde el exploramos hacia arriba, no tiene inconvenientes, marcamos los S->5 y S->6

El siguiente es Cod N24, si exploramos por arriba vemos que tiene un camino libre y un camino ya fijo en A por lo que se fija el otro en B

Pasamos por Des N37, tiene en un camino al Cod N14 por lo que el otro camino debe ser A o B (ya está fijo en B, no pasa nada)

Los nodos Des N35, N38, N36. De igual manera tienen a un Codificador en uno de sus caminos pasa lo mismo que para N37.

Pasamos al ultimo nivel en Des N29, por un camino tiene al Cod N24, y el otro libre, se marca temporalmente S->4 como A/B.

N34 pasa por un Cod, por lo que su otro camino seria A/B (ya marcado)

Hemos pasado por todos los destinos. La respuesta de lo que se debe enviar es

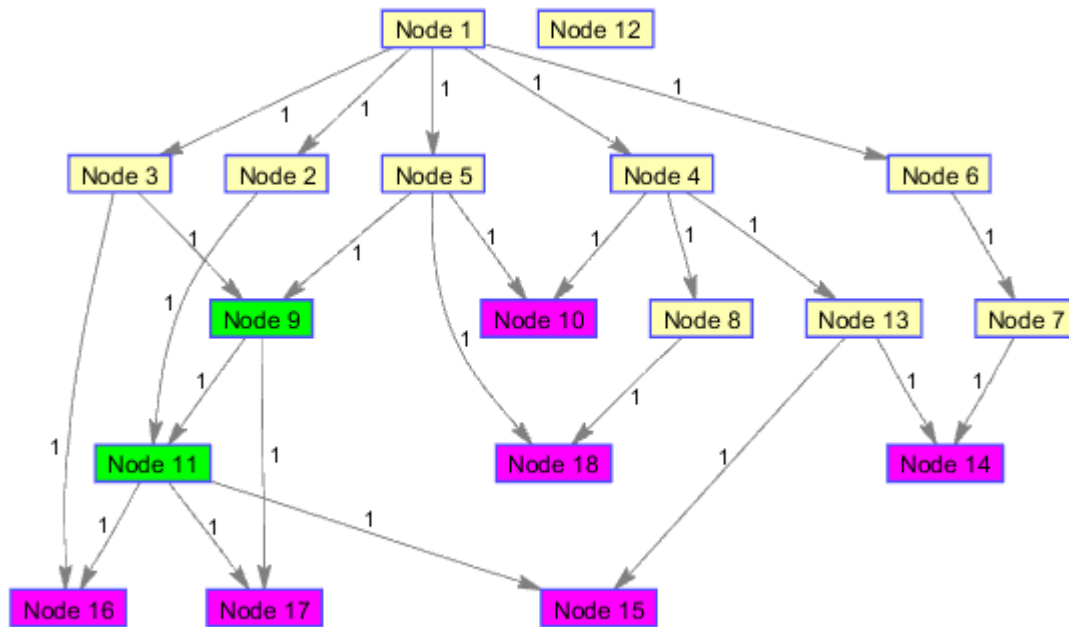
AoB(Cualquiera de los dos, no afecta el resultado) B A B

## Otros ejemplos

¡\* Probando el nuevo algoritmo con los otros casos, el ejemplo de 18 nodos

%Nodos: 18, Destinos: 10, 14, 15, 16, 17, 18

Presenta un error que había mencionado anteriormente.



Para el Nodo 17, ambos caminos tienen un nodo codificador. Esto es un error.

Según lo que analicé esto se dio porque los caminos hacia N16 y N17 se cruzan entre sí, tal vez sea posible arreglar esto expandiendo la idea anterior de contar el numero de repeticiones por camino, pero expandirlo para un cruce entre los grafos individuales no es practica y puede ser muy grande.

No tengo una solución exacta para esto pero tengo ideas al respecto.

En primer lugar, la solución tiene que ser deshacer 1 de los codificadores, para esto basta con quitar uno de los enlaces padres del nodo codificador.

En este caso los posibles candidatos son N3->N9, N5-> N9, N2->N11, N9->N11

En primer lugar pensé que se podría eliminar cualquiera pero, si por ejemplo eliminamos N2->N11, se deshace el Cod N11, pero queda un Loop con el A+B enviado de N9 hacia N11 y hacia N17

Eliminar N9-N11 no genera este error pero no se como se determina si genera error o no.

Pero por otro lado se me ocurre que este error no pasaría si se elimina el **Nodo Codificador del nivel más superior**, probando eliminando ambos enlaces N3->N9 o N5->N9 se deshace el Cod en N9 y probando con el algoritmo propuesto para determinar paquetes de envío el grafo está funcionando.

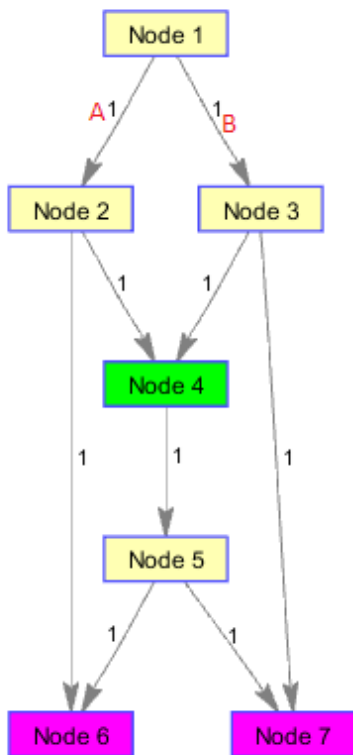
Pero no tengo certeza de que esta idea de eliminar el codificador más superior siempre funcione, hay que analizar más casos.

---

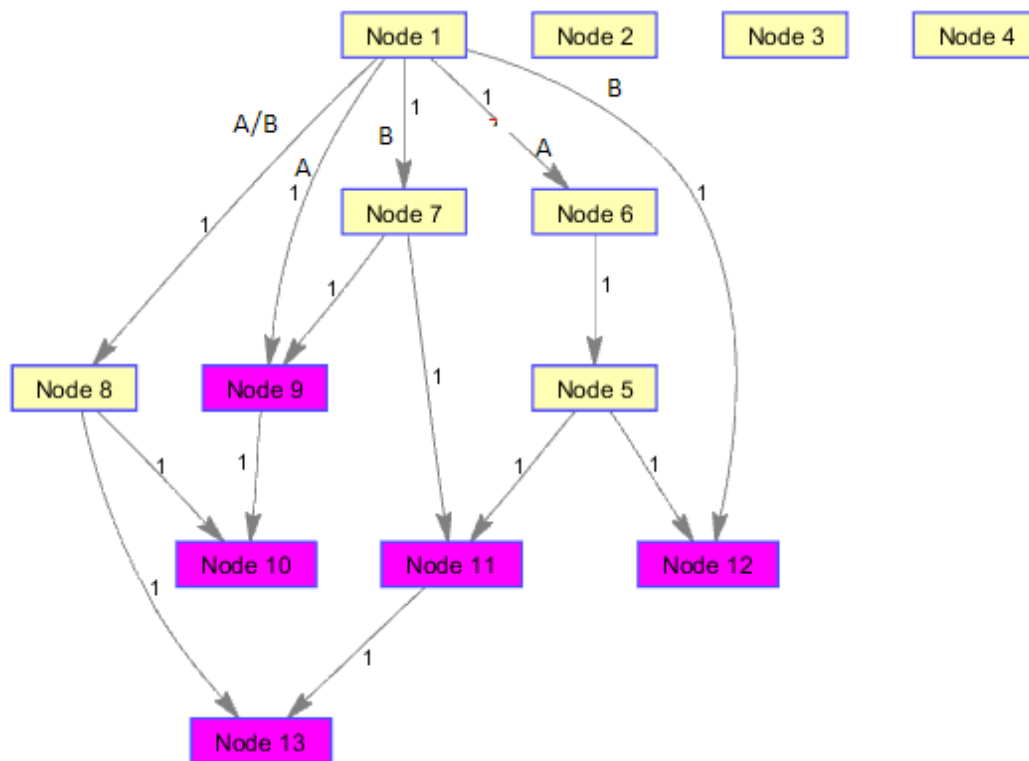
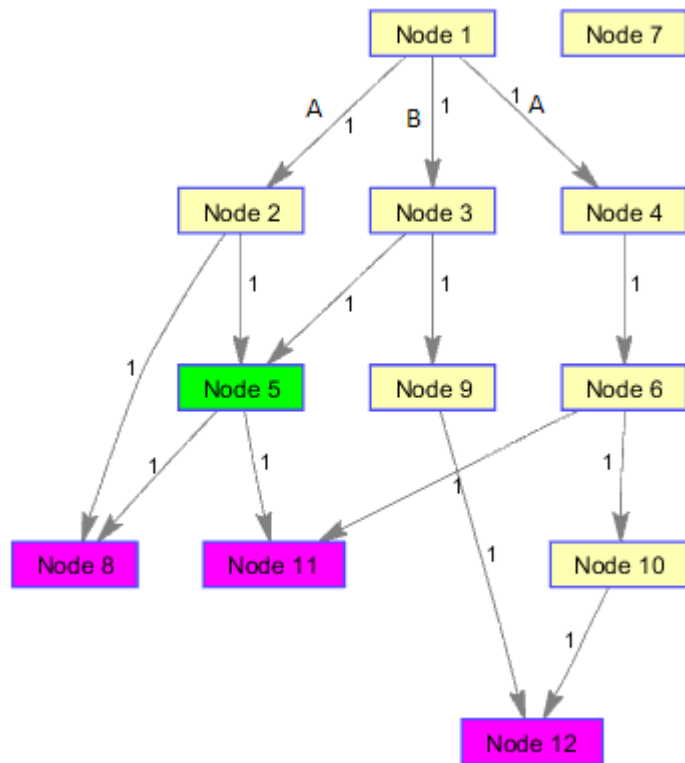
---

Ejemplos del algoritmo

---



Mariposa es trivial, pero el algoritmo funciona.





Tengo fé en que funciona o está en el camino correcto porque a mi parecer si no se cumple lo que digo entonces el grafo no tiene solución (por ejemplo, de los enunciados sobre que debe recibir un nodo Cod y Des).

También queda la pregunta de si el nodo Source solo envía A y/o B (que a mi parecer debería ser así, el nodo Source no es codificador como para enviar A+B)