

1. Explicación del Algoritmo

Para iniciar, debemos definir un grafo. Este grafo estará representado por una matriz de adyacencia. Cada uno de los vértices será una célula y será representado sin colación por su identificador (En concreto en la matriz de adyacencia es su identificador -1). Además, existe un arco entre dos vértices si comparten péptidos y si la distancia entre ambos es menor a la distancia máxima en la que puede ocurrir una conexión. Además, cada arco tiene un peso correspondiente al número de péptidos que comparten. Esto nos da un grafo no dirigido con pesos.

Lo que nos interesa es saber el número máximo de mensajes que se pueden enviar por este grafo, esto esencialmente es un problema de flujo. Lo que nos interesa ver es cuál de las células Calculadoras tiene un mayor efecto en el flujo total.

Para esto se hizo uso del algoritmo de Ford-Fulkerson. Este algoritmo es ejecutado sobre el grafo original y luego se aplica de manera subsecuente a el grafo en donde fue retirada la célula calculadora concreta.

En un principio dudaba de aplicar Ford-Fulkerson en cada paso. Sin embargo, es necesario hacerlo pues es posible que la ruta de flujo cambie radicalmente y que el flujo que pasa por la neurona que estamos mirando pueda ser tomado por otra en su mayoría afectando mínimamente el flujo máximo total.

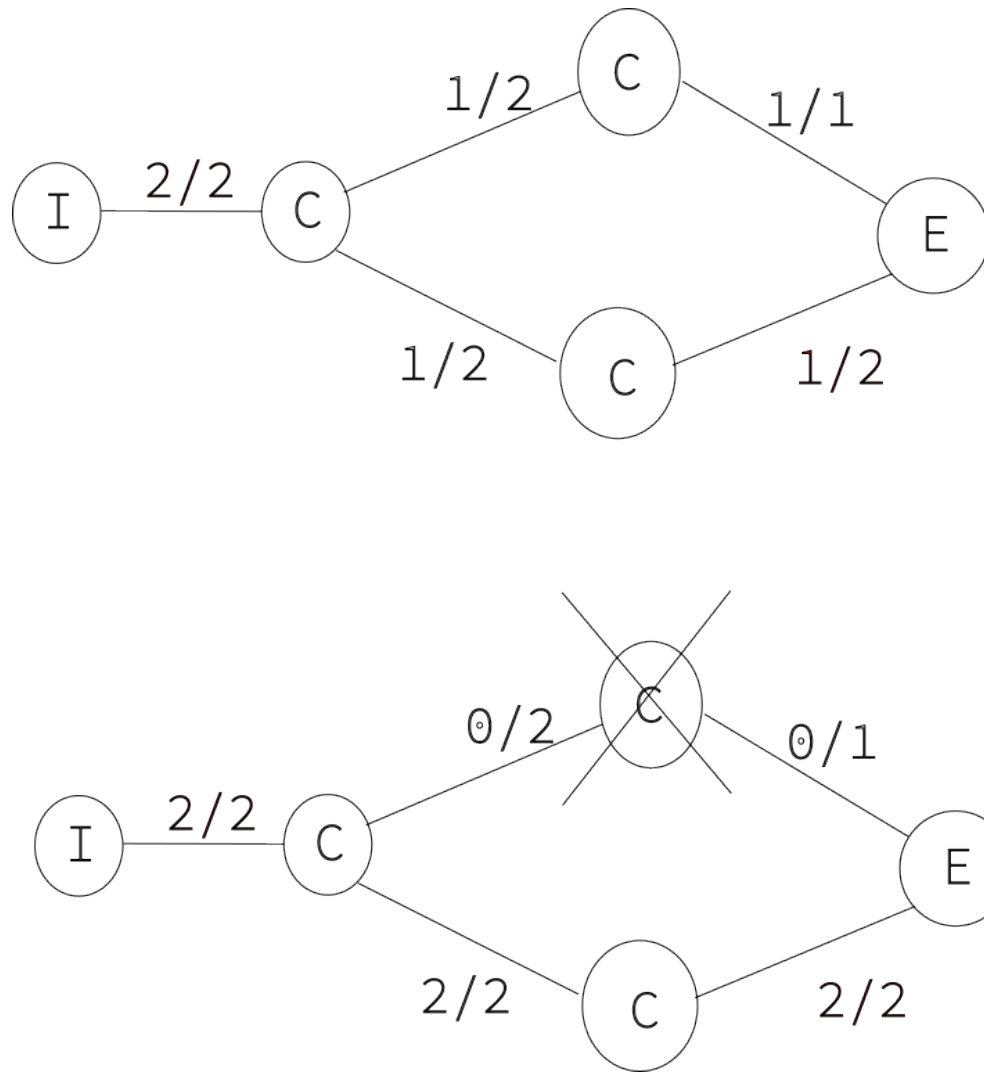


Figura 1: Ejemplo del algoritmo probando para una de las células calculadoras.

2. Complejidad

Para la complejidad temporal la tomaremos solamente de la solución como tal. La construcción del grafo apartar de la entrada tiene una complejidad polinomial pero podría variar ampliamente. En concreto, trabajaremos con la función `get_max_flow` que esta aproximadamente en la línea 210.

2.1. Temporal

Lo primero que hace esta función es crear una nueva copia. Esto tiene una complejidad $O(V^2)$. Luego de esto remueve la célula que se le indica lo cual es una complejidad de $O(V)$ y por ultimo calcula el flujo maximo con FORD-FULKERSON lo cual tiene una complejidad de $O(|V| \cdot E^2)$.

Por lo tanto esta función tiene una complejidad de

$$O(V^2) + O(V) + O(|V| \cdot E^2) = O(|V| \cdot E^2).$$

Ahora bien, esta función es ejecutada C veces donde C es el numero de células Calculadoras. En el peor de los casos $C = V$ por lo tanto para el peor de los casos la complejidad temporal es:

$$O(V \cdot |V| \cdot E^2).$$

2.2. Espacial

Para la complejidad espacial debemos tomar en cuenta que esta función crea una copia de la matriz lo que tiene un costo de $O(V^2)$ y además FORD-FULKERSON tiene un costo de $O(V)$. Por lo tanto la complejidad total sería

$$O(V^2) + O(V) = O(V^2).$$

3. Preguntas Comprensión

3.1. Primera Pregunta

En ese caso, se sabe que el número mínimo de mensajes que podría tener el flujo es el de la suma de los mensajes que pasan de las células iniciadoras a las células ejecutoras. Pues incluso si se apagarán todas las células calculadoras seguirían estando estos caminos.

3.2. Segunda Pregunta

En ese caso el número de mensajes máximo sería el del número de células calculadoras pues serían como un cuello de botella que solo permitiría que el flujo máximo fuera de uno en ellas.