

Problem Set 1

Ángel Daniel Alonso Guevara

Clique Máximo

Definición del problema: Dado un grafo G con n vértices y un entero k , queremos determinar si existe un subgrafo completo (clique) de tamaño k .

Solución:

Demostración de que es NP-Hard:

Consideremos una entrada para 3-SAT, que es una fórmula F definida sobre n variables con m cláusulas.

Construimos el grafo G de la fórmula F de la siguiente manera:

1. Para cada literal en la fórmula generamos un vértice y etiquetamos el vértice con el literal al que corresponde. Cada cláusula corresponde a tres de estos vértices.
2. Conectamos dos vértices en el grafo si:
 - (a) están en diferentes cláusulas, y
 - (b) no son la negación uno del otro.

Relación entre ambos problemas:

Vamos a probar que F es satisfacible si y solo si existe un clique de tamaño m en G :

\Rightarrow Si F es satisfacible, entonces existe un clique de tamaño m en G .

Sean x_1, \dots, x_n las variables que aparecen en F , y sean $v_1, \dots, v_n \in \{0, 1\}$ la asignación que satisface F . La fórmula F se cumple si asignamos $x_i = v_i$, para $i = 1, \dots, n$.

Para cada cláusula C en F debe haber al menos un literal que evalúe a True.

Elige un vértice que corresponda a dicho valor True de cada cláusula. Sea W el conjunto resultante de vértices. Claramente, W forma un clique en G . El conjunto W tiene tamaño m , ya que hay m cláusulas y cada una contribuye con un vértice al clique.

\Leftarrow Si existe un clique de tamaño m en G , entonces F es satisfacible.

Sea U el conjunto de m vértices que forman un clique en G . Necesitamos traducir el clique G_U a una asignación satisfactoria de F .

- (i) Asignar $x_i \leftarrow TRUE$ si hay un vértice en U etiquetado con x_i .
- (ii) Asignar $x_i \leftarrow FALSE$ si hay un vértice en U etiquetado con $\overline{x_i}$.

Vamos a probar que la asignación es válida utilizando reducción al absurdo:

Supongamos que existe x_i tal que hay dos vértices u, v en U etiquetados con x_i y $\overline{x_i}$; es decir, asignamos valores contradictorios a x_i . Entonces, u y v , por construcción, no estarán conectados en G , y como tal G_U no es un clique. Esto produce una contradicción, por lo que lo supuesto es False, y la asignación es válida.

Para demostrar que es una asignación satisfactoria, notemos que hay al menos un vértice de U en cada cláusula, por lo que hay un literal evaluado como True en cada cláusula. Por tanto, F es True.

Por tanto, dado un algoritmo para clique máximo en tiempo polinomial, podemos resolver 3-SAT en tiempo polinomial, lo cual implica que el problema de encontrar el clique máximo de un grafo es NP-Hard.

Demostración de que es NP:

Dado un grafo G con n vértices, un parámetro k , y un conjunto W de k vértices, verificar que cada par de vértices en W forma una arista en G toma $O(u + k^2)$, donde u es el tamaño de la entrada. Por tanto, verificar una respuesta positiva a una instancia de Clique se puede hacer en tiempo polinómico, lo que implica que pertenece a NP.

Problema Retroalimentación de Vértices

Definición del problema: Dado un grafo $G = (V, E)$, un conjunto de retroalimentación de vértices es un subconjunto de vértices

$F \subseteq V$ tal que al eliminar todos los vértices de F (y sus aristas incidentes), el grafo resultante no contiene ciclos (es un grafo acíclico o un bosque, si es no dirigido). El objetivo del problema es encontrar el conjunto de retroalimentación de vértices de tamaño mínimo.

Solución: Vamos a demostrar que el problema es NP-Hard. Para ello, vamos a demostrar primero que el problema de retroalimentación de vértices de tamaño k es NP-Completo:

Demostración de que es NP: Dada una instancia de retroalimentación de vértices y un conjunto solución D , bastaría verificar que el tamaño de D es menor o igual que k y que al eliminar esos vértices del grafo este es acíclico, lo cual se puede hacer haciendo un recorrido por el grafo.

Demostración de que es NP-Hard: Para esta parte intentaremos reducir el problema de Vertex Cover (el cual sabemos que es NP-Completo) a Retroalimentación de Vértices. La reducción se creará de la siguiente forma: dada una instancia de Vertex Cover con $G = (V, E)$, crearemos un grafo dirigido $G' = (V', E')$ con $V' = V$ y por cada arista $(a, b) \in E$ crearemos las aristas (a, b) y (b, a) en E' , generando un ciclo en G' .

Relación entre ambos problemas:

Para Vertex Cover necesitamos un conjunto mínimo tal que cada arista sea cubierta. Para Retroalimentación de Vértices necesitamos un conjunto de vértices mínimo tal que si lo eliminamos del grafo, resulta en un grafo sin ciclos.

Dado que cada arista en G de Vertex Cover se transforma en dos aristas dirigidas en G' formando un ciclo, el conjunto seleccionado para satisfacer Retroalimentación de Vértices, digamos F , debe incluir al menos uno de los dos vértices para romper ese ciclo.

Si tenemos entonces F en G' , entonces F debe cubrir cada arista en G . Eso significa que para cada ciclo dirigido correspondiente en G' , F debe contener al menos uno de los vértices de ese ciclo para romperlo. Al eliminar F en G' , todos los ciclos dirigidos se eliminan lo que hace que G' sea acíclico. Por tanto, si en Retroalimentación de Vértices hay una solución de tamaño k , entonces también la hay en Vertex Cover.

Habiendo demostrado que este problema es NP-Completo podemos reducir el mismo al de hallar el mínimo conjunto que satisfaga Retroalimentación de Vértices. Pues si el mínimo es menor que k entonces hay una solución de longitud k y si no lo es pues no la hay.

Por tanto, el problema de hallar el mínimo conjunto que satisfaga Retroalimentación de Vértices es NP-Hard.

Problema Retroalimentación de Arcos

Definición del problema: Dado un grafo $G = (V, E)$, un conjunto de retroalimentación de arcos es un subconjunto de arcos $F \subseteq E$ tal que al eliminar todos los arcos en F , el grafo resultante no contiene ciclos. El objetivo del problema es encontrar el conjunto de retroalimentación de arcos de tamaño mínimo.

Solución: Al igual que en el problema anterior, demostraremos primero que el problema de encontrar un conjunto de arcos de longitud k tal que al quitarlos no queden ciclos en G es NP-Completo. Para ello intentaremos reducir el problema de Retroalimentación de Vértices a este en tiempo polinomial.

Dada una instancia (G, k) de Retroalimentación de Vértices con $G = (V, E)$, crearemos una instancia (G', k') de Retroalimentación de arcos de la siguiente forma: Por cada nodo $v \in V$ creamos dos nodos v_1 y v_2 en G' y el arco (v_1, v_2) , a estos arcos les llamaremos arcos internos. Luego por cada arco $(u, v) \in E$ crearemos en E' el arco (u_2, v_1) , estos serán los arcos externos.

Cada vértice en G se duplica en G' , creando dos copias de vértices y las aristas del grafo original se dividen en aristas internas (entre las copias de un mismo vértice) y aristas externas (entre copias de vértices diferentes).

Al resolver RA (Retroalimentación de arcos) en G' , buscamos un conjunto de aristas que rompa todos los ciclos en G' . La clave está en que todos los ciclos que involucran aristas externas también pasan por aristas internas, por lo que eliminar solo aristas internas (que corresponden a vértices en G) es suficiente para romper los ciclos.

Por lo tanto, cualquier conjunto de aristas internas que forme una solución para RA en G' corresponde a un conjunto de vértices que forman una solución para RV (Retroalimentación de Vértices) en G . De esta forma, resolver RA en G' es equivalente a resolver RV en G , ya que los ciclos que rompes en G' equivalen a los ciclos que rompes en G .

Y, análogamente, como en el problema anterior, demostramos que hallar el mínimo conjunto es NP-Hard. Pues en este problema

también sirve para demostrar que hallar el mínimo conjunto de aristas es NP-Hard.

Número Cromático

Definición del problema: Dado un grafo $G = (V, E)$, el número cromático de un grafo es el mínimo número de colores necesarios para colorear los vértices de tal forma que dos vértices adyacentes no tengan el mismo color. El problema de número cromático consiste en determinar el número cromático de un grafo.

Solución:

3-coloreo es NP porque dado G y k , un certificado de que la respuesta es sí es simplemente un k -coloreo. Se puede verificar en tiempo polinómico que se usan a lo sumo k colores y que ningún par de nodos unidos por una arista recibe el mismo color.

Demostración de NP-Compleitud

Dada una instancia de 3-SAT, con variables x_1, \dots, x_n y cláusulas C_1, \dots, C_k , lo resolveremos usando una caja negra para 3-Coloreo.

Definimos nodos v_i y \bar{v}_i correspondientes a cada variable x_i y su negación \bar{x}_i . También definimos tres "nodos especiales" T , F y B , que denominamos True, False y Base. Para comenzar, unimos cada par de nodos v_i , \bar{v}_i entre sí con una arista, y unimos ambos nodos a Base. (Esto forma un triángulo en v_i , \bar{v}_i y Base, para cada i). Unimos True, False y Base en un triángulo.

El grafo G resultante tiene las siguientes propiedades:

1. En cualquier 3-coloreo de G , los nodos v_i y \bar{v}_i deben obtener colores diferentes, y ambos deben ser diferentes de Base.
2. En cualquier 3-coloreo de G , los nodos True, False y Base deben recibir los tres colores en alguna permutación. Por lo tanto, podemos referirnos a los tres colores como el color True, el color False y el color Base, según cuál de estos tres nodos obtenga qué color. En particular, esto significa que para cada i , uno de v_i o \bar{v}_i obtiene el color True, y el otro obtiene el color False. Para el resto de la construcción, consideraremos que la variable

x_i está establecida en 1 en la instancia dada de 3-SAT si y solo si el nodo v_i recibe el color True.

Relación entre ambos problemas: Como en otras reducciones de 3-SAT, consideremos una cláusula como $x_1 \vee x_2 \vee x_3$, que representa que al menos uno de los nodos v_1 , v_2 o v_3 debería obtener el color True. Así que lo que necesitamos es un pequeño subgrafo que podamos enchufar a G , de modo que cualquier 3-coloreo que se extienda a este subgrafo debe tener la propiedad de asignar el color True a al menos uno entre v_1 , v_2 o v_3 .

Este subgrafo de seis nodos se adjunta al resto de G en cinco nodos existentes: True, False y aquellos correspondientes a los tres términos en la cláusula que estamos tratando de representar (en este caso, v_1 , v_2 y v_3). Ahora supongamos que en algún 3-coloreo de G todos los tres v_1 , v_2 y v_3 reciben el color False. Entonces los dos nodos sombreados inferiores en el subgrafo deben recibir el color Base, los tres nodos sombreados por encima de ellos deben recibir, respectivamente, los colores False, Base y True, y por lo tanto no hay color que se pueda asignar al nodo sombreado superior.

Algunos chequeos manuales de casos muestran que mientras al menos uno de v_1 , v_2 o v_3 reciba el color True, el subgrafo completo puede ser coloreado con 3 colores.

Entonces, a partir de esto, podemos completar la construcción: comenzamos con el grafo G definido arriba, y para cada cláusula en la instancia de 3-SAT, adjuntamos un subgrafo de seis nodos como se muestra en la Figura 2. Llamemos G' al grafo resultante.

Ahora afirmamos que la instancia dada de 3-SAT es satisfactoria si y solo si G' tiene un 3-coloreo. Primero, supongamos que hay una asignación satisfactoria para la instancia de 3-SAT. Definimos una coloración de G' coloreando primero Base, True y False arbitrariamente con los tres colores; luego, para cada i , asignando a v_i el color True si $x_i = 1$ y el color False si $x_i = 0$. Luego asignamos a \bar{v}_i el único color disponible. Finalmente, como se argumentó anteriormente, ahora es posible extender esta coloración de 3 colores en cada subgrafo de cláusula de seis nodos, resultando en una coloración de 3 colores de todo G' .

En sentido contrario, supongamos que G' tiene una coloración de 3 colores. En esta coloración, cada nodo v_i recibe ya sea el color True o el color False; establecemos la variable x_i en consecuencia. Ahora afirmamos que en cada cláusula de la instancia de 3-SAT, al

menos uno de los términos en la cláusula tiene el valor de verdad 1. Porque si no, entonces los tres nodos correspondientes tienen el color False en la coloración de 3 colores de G' y, como hemos visto anteriormente, no hay una coloración de 3 colores del subgrafo correspondiente consistente con esto, por lo que hemos llegado a una contradicción.