PROYECTO TERCERA NOTA

ANALISIS DE ALGORITMOS

CARLOS TORRES

1151123

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

CUCUTA, COLOMBIA

INFORME

AGORITMO DE ENCONTRAR PUNTOS DEBILES EN UN GRAFO CONEXO

El proceso para encontrar los puntos débiles (WEAK LINKS) en un grafo conexo consiste en comparar cada nodo con sus respectivos nodos adyacentes y buscar si se puede llegar a él por medio de otro enlace que no sea el que se está comparando.

Ejemplo:

Sea un nodo A comparando su enlace con un nodo B, si se puede llegar desde A a B por medio de otro enlace que no sea [A, B] se dice que este no es un punto débil, pero, si no se puede llegar si no es por [A, B] entonces ese enlace se considera enlace débil dentro del grafo.

Para realizar este proceso, se debe comenzar con recorrer la lista de nodos que hay en el grafo, donde cada nodo tiene además una lista con sus respectivos nodos adyacentes (hay que tener en cuenta que el grafo no está dirigido luego los enlaces son bidireccionales); cada nodo de la lista de nodos del grafo se pasa a comparar sus enlaces(con el objetivo de verificar si es un punto débil) con cada uno de sus nodos adyacentes, buscando si hay otra manera de conectarse con él, sin pasar por el enlace comparado, cada vez que se encuentra que no hay manera de conectarse sin pasar por el nodo comparado se toma como un punto débil dentro del grafo.

Hay que tener en cuenta que cuando un nodo que ha sido pasado a comparar con los nodos de su lista adyacente con el fin de buscar puntos débiles, después de que este termina su comparación con su lista adyacente, este se toma como visitado en el grafo, ya que el grafo no es dirigido luego puede llegar el momento en el que el algoritmo lo puede volver a tomar en comparación para buscar punto débil.

cuando se toma un enlace para el análisis si es un punto débil, se busca en los nodos de la lista adyacente del nodo que ha sido pasado a comparar si con ellos se puede conectar a el nodo que se está analizando si es un punto débil; debido a esto puede llegar el caso en el compare nodos que ya he comparado y se produzca un bucle infinito para eliminar este problema se utiliza un número que es único para cada conexión de análisis si es un punto débil, el nodo que tenga este número indica que ya ha sido visitado dentro de la comparación si es un punto débil.

EJEMPLO

Sea un nodo A que ha sido pasado a comparación, este se debe comparar con cada uno de sus nodos adyacentes para encontrar si alguno de estos enlaces es un punto débil.

A {B, C}.

A tiene como adyacente a B y C, si se compara A con B, se busca si con C puedo conectarme a B, para esto el nodo C debe buscar en su lista Adyacente si se conecta con B

Debido a que el grafo no es dirigido C tiene en su lista de nodos adyacentes a A, para controlar este proceso se utiliza un numero único para cada conexión, entonces cada vez que un nodo ha sido visitado durante cada conexión se le va colocando un numero único, que da como significado que ese nodo ya ha sido visitado en el proceso de comparación (con esto se controla no volver a comparar nodos que ya han sido comparados),

Luego una vez que A termina su comparación con su lista adyacente este se toma como visitado totalmente en el grafo es decir donde el aparezca en las listas adyacentes de cualquier otro nodo este no puede ser pasado a comparar si es punto débil en algún enlace.

En el caso del ejemplo cuando termina A con su lista adyacente, se procede a buscar el otro nodo que está en la lista de nodos en el grafo, continua B.

B {A, C}

B tiene como lista adyacente a A y C el no compara con A si es punto débil debido a que este ya ha sido tomado en cuenta como visitado dentro del grafo, luego se procede a comparar con C, entonces se busca si con A puedo llegar a C, cuando termina B con su lista adyacente este se toma como visitado luego se procede con el nodo C

C {A, B}

Para este caso el nodo C no compara con sus nodos adyacentes si es un punto débil debido a que sus nodos adyacentes ya han sido visitados en el grafo.