Grafos sin espinas

(5) Grafos Hamiltonianos & Eulerianos

V 0.1 2023_04_05

Aprende sin espinas
con @carlos_cactus

A la inspiración del bucle_infinito,
al Cibergrupo y al tHash_A, por su amistad,
y sobre todo, a quienes dicen "pero quiero"
cuando sienten "no puedo".

¡Encuéntrame en Telegram o en mi canal de YT Aprende sin Espinas!

¡Un saludo sin espinas!
@carlos_cactus :D

Índice

1. Grafos eulerianos

1.1. Origen

s XVIII Königsberg 4 zonas / 7 puentes → Reto: volver al punto de salida cruzando todos los puentes exactamente 1 vez.
Euler demostró que era imposible y formalizó el concepto de grafo euleriano.

1.2. Recordatorio de tipos de recorrido

Cabe recordar:

- Una cadena es una secuencia de aristas/vértices vecinos.
- Una cadena puede repetir tanto vértices como aristas.
- Un recorrido solo puede repetir vértices, pero no aristas: es SIMPLE.
- Un recorrido simple abierto es un itinerario y si es cerrado es un circuito.
- Si un recorrido simple no repite vértices, es ELEMENTAL.
- Un recorrido ELEMENTAL abierto es un camino y si es cerrado es un ciclo.

1.3. Recorrido euleriano y grafo semieuleriano

Un recorrido euleriano, para un grafo conexo G, es una cadena que contiene TODAS las aristas del grafo G sin repetir ninguna. Nótese que sí puede repetir vértices.

Si un grafo contiene un recorrido euleriano se dice que es semieluleriano.

1.4. Condición de necesidad y suficiencia para la existencia de un recorrido euleriano

Si un grafo G conexo presenta exactamente 2 vértices de grado impar, contiene algún recorrido euleriano.

1.5. Circuito euleriano

Un circuito euleriano, para un grafo conexo G, es un recorrido cerrado que contiene TODAS las aristas de G sin repetición. Nótese que sí puede repetir vértices.

1.6. Grafo euleriano

Un grafo G conexo es euleriano si en él existe al menos un circuito euleriano. Entonces, un grafo puede contener recorridos eulerianos sin ser euleriano en sí mismo.

Para ser euleriano, todos sus vértices han de ser, necesariamente, de grado par.

1.7. Condición de suficiencia y necesidad de grafos eulerianos

Un grafo G conexo es euleriano si y sólo si todos sus vértices son de grado par.

Nótese que, si G es euleriano, no presentará recorridos eulerianos (que requieren de 2 exactamente 2 vértices de grado impar).

Es decir, la presencia de circuitos y recorridos eulerianos es mutuamente excluyente: un grafo euleriano no puede alojar recorridos eulerianos y un grafo con recorridos eulerianos no puede ser euleriano.

1.8. Consecuencia de la partición d aristas en grafos eulerianos

Todo grafo euleriano G = (V,A) conexo es compatible con una partición disjunta de sus aristas. Es decir:

$$A = A_1 \cup A_2 \cup A_3 \dots A_k \text{ donde } A_i \cap A_j = \emptyset, \forall\, i \neq j.$$

1.9. Propiedades de los grafos eulerianos:

Si G es euleriano:
- Es conexo.
- Todos sus vértices son de grado par.
- Contienen algún circuito euleriano.
- Admite alguna partición disjunta de sus aristas.

1.10. Construcción de un circuito euleriano: Algoritmo de Hierholzer

- Atribuido a Hierholzer (1873).
- Se basa en la propiedad de la partición disjunta de aristas de los grafos eulerianos.
- Consiste en la sucesiva concatenación de circuitos sin aristas en común.
- Tiene como entrada un grafo euleriano G = (V, A) de orden n con m aristas.
- Tiene como salida un circuito euleriano en G escrito como lista de vértices.
- Se inicia con la elección arbitraria en G de un vértice inicial s.
- Cursa con la construcción de cualquier circuito C dentro de G a partir del vértice s.
- A continuación, se construye otro circuito C' a partir del primer vértice de C con grado positivo (ha de tener aristas salientes).
- Se concatenan C y C' sustituyendo LA PRIMERA INSTANCIA de v en la lista de C por toda la lista de vértices del circuito recién formado C'.
- A medida que se forman circuitos C, se suprime de G las aristas incorporadas en ellos.
- Se finaliza cuando no quedan en G aristas pendientes de ser incluidas en algún circuito.

***EJEMPLO

1.11. Análisis del algoritmo de Hierholzer

- La elección de un vértice inicial de grado positivo en la lista C tiene como complejidad O(n) para un grafo de orden n.
- La complejidad de la construcción del circuito C' a partir del vértice v de grado positivo depende de las aristas que ya hayan sido incluidas en otros circuitos C previos, de modo que, en el peor de los casos, será O(m) para un grafo con m aristas.
- Concatenar los circuitos resultantes *** pág 15 3) y 4)
- La supresión de aristas, en el peor de los casos, terminará eliminándolas todas, es decir, tiene complejidad O(m).

Es decir, la complejidad global es:

$$\max\{O(n), O(m)\} = O(m)$$

Se escoge O(m) y se rechaza O(n) porque el algoritmo tiene como entrada un grafo G euleriano, o sea, G es conexo. Por tanto, en el peor de los casos, tendrá siempre más aristas que vértices, pero nunca podrá ser al revés.

2. Grafos hamiltonianos

2.1. Recorrido hamiltoniano

Un recorrido es hamiltoniano si pasa exactamente 1 vez por todos los vértices del grafo.

Nótese que no hay necesidad de recorrer todas las aristas.

2.2. Ciclo hamiltoniano

Un ciclo es hamiltoniano si pasa exactamente 1 vez por todos los vértices del grafo.

Nótese que no hay necesidad de recorrer todas las aristas.

2.3. Grafo hamiltoniano

Un grafo conexo G es hamiltoniano si aloja algún ciclo hamiltoniano, o sea, un recorrido cerrado que pasa por TODOS LOS VÉRTICES EXACTAMENTE UNA SOLA VEZ.

EJEMPLOS H VS. E 18***

2.4. 2-conectividad

Un grafo G es 2-conexo si para TODA pareja de vértices u y v de G existe, como mínimo, 2 caminos disjuntos (que no comparten arista alguna) que los conectan. O sea, los caminos disjuntos solo tienen en común los vértices extremos u y v.

2.5. Propiedades NECESARIAS de grafos hamiltonianos

No hay condición necesaria y suficiente para caracterizar un grafo hamiltoniano.

Hay algunas condiciones NECESARIAS que todo grafo G hamiltoniano cumple.

Por tanto, la insatisfacción de alguna de estas propiedades necesarias basta para descartar un grafo G como hamiltoniano.

Si G es hamiltoniano, entonces:
- G es conexo.
- G es 2-conexo.
- Todos sus vértices tienen grado mayor o igual a 2: $g(v_i) \geq 2$
- Si G es bipartito (V_1,V_2), sus particiones tienen la misma cardinalidad: $|V_1| = |V_2|$
  Esto responde a la distribución alternativa de los vértices consecutivos en el ciclo hamiltoniano entre las particiones V_1 y V_2 respectivamente.
- El número de componentes conexas c que resulta de la supresión de un subconjunto de vértices S del grafo G NUNCA puede exceder el número de vértices suprimidos |S|. O sea:
  $$c(G - S) \leq |S|$$
  Es decir que, si se eliminan r vértices, como mucho, se deben producir r componentes conexas y nunca más.

Quedan excluidos por tanto los grafos trayecto y los árboles.

2.6. Complejidad de la determinación de un ciclo hamiltoniano

La búsqueda de un ciclo hamiltoniano es un problema intratable.

2.7. Problema TSP

El problema del viajante de comercio (travelling salesman problem) es un problema intratable que consiste en la determinación de un ciclo hamiltoniano de peso mínimo en un grafo conexo y ponderado.

Ante la ausencia de un algoritmo eficiente que resuelva el problema, se recurre a una cierta simplificación. Cuando se cumple la desigualdad triangular en un grafo, sí que se puede usar un algoritmo eficiente para encontrar una solución APROXIMADA a la óptima, el error de la cual se puede acotar.

2.8. Desigualdad triangular

Un grafo ponderado (G,w) satisface la desigualdad triangular si para TODA terna de vértices u, v, x se cumple que el coste de ir directamente de un vértice a otro vecino NUNCA EXCEDE el coste que tiene hacer un relevo a través del tercero para ir de uno al otro. O sea:

$$w(u,v) \leq w(u,x) + w(x,v)$$

La satisfacción de esta propiedad conlleva, necesariamente, que el grafo sea COMPLETO.

2.9. Aproximación del ciclo hamiltoniano de peso mínimo: algoritmo TSP-aproximado

El problema TSP se puede reformular ahora como:

Dado un grafo conexo y ponderado (G,w) que satisface la desigualdad triangular (por tanto, es COMPLETO), se dese encontrar un ciclo hamiltoniano de peso mínimo.

Este problema sigue siendo INTRATABLE, pero considerando la desigualdad triangular, se dispone de un algoritmo EFICIENTE que ofrece una solución APROXIMADA a la búsqueda de un ciclo hamiltoniano de peso mínimo, o sea, al problema TSP con desigualdad triangular.

Para encontrar, en un grafo (g,w) un ciclo hamiltoniano cuyo peso sea aproxime al óptimo, basta con:

1. Elegir un vértice r al cual se asigna el estatus de raíz.
2. Aplicar el algoritmo de Prim para encontrar el árbol generador minimal desde la raíz r.
3. Ejecutar la exploración en PREORDEN del árbol obtenido.
4. Cerrar el recorrido en PREORDEN añadiendo al final el vértice r para cerrar el ciclo.

Esta estrategia se basa en construir un ciclo utilizando el mayor número de aristas del árbol generador minimal de G.

Ejemplo***

2.10. Cota de error asociada a TSP-aproximado

El ciclo hamiltoniano H que se desprende de la ejecución del algoritmo TSP-aproximado tiene como peso w(H).

Se denota por H^* el ciclo óptimo, asociado a un peso w(H).

Se cumple: $$w(H) \leq 2 \cdot w(H^*)$$

El peso del ciclo H encontrado nunca alcanzará el doble del peso del ciclo óptimo H^*.

Se puede demostrar así:

Se denota por T el árbol generador minimal obtenido vía Prim sobre G, cuyo peso es w(T).

En primer lugar, se observa, necesariamente:

$$w(T) \leq w(H^*)$$

Ya que T resulta de unir todos los vértices de G con el menor peso posible (por Prim).

En segundo lugar, como el recorrido del árbol generador minimal T se realiza en PREORDEN respecto la raíz r a la hora de cerrar el ciclo hamiltoniano H, cada vértice ingresa en la secuencia la primera vez que se visita y solo se retrocede cuando se ha visitado una hoja.

Esto conlleva que se recorra 2 veces cada arista, de modo que w(H) SERÍA 2·w(T).

Pero como una vez añadido el último vértice del árbol, se retrocede hasta r a través de una arista DIRECTA y, a la vez, se satisface la DESIGUALDAD TRIANGULAR, se cumplirá:

$$w(H) \leq 2 \cdot w(T)$$

De modo que NUNCA se recorrerán 2 veces todas las aristas.

Por tanto, de las 2 desigualdades, se puede deducir que:

$$w(H) \leq 2 \cdot w(T) \leq 2 \cdot w(H^*)$$

Es decir, el error cometido nunca supondrá más de la mitad del peso del ciclo H encontrado:

$$w(H) \leq 2 \cdot w(H^*)$$

O sea, el peso del ciclo H encontrado nunca alcanzará el doble del peso del ciclo óptimo H^*.

2.11. Análisis del algoritmo TSP-Aproximado

Aunque da una solución APROXIMADA a un problema INTRATABLE (TSP con desigualdad triangular), se puede considerar computacionalmente EFICIENTE.

Las 2 operaciones del algoritmo son:

- Obtención del árbol generador minimal vía algoritmo de Prim, con complejidad $O(n^2)$.
- Exploración en PREORDEN: se visita cada vértice 1 sola vez. Luego tiene complejidad O(n).

La complejidad global del algoritmo TSP-aproximado es:

$$\max\{O(n^2), O(n)\} = O(n^2)$$