2: RECORRIDOS DE GRAFOS

2.1 EULER: recorrido por aristas/arcos.

- El problema de los puentes de Königsberg.
- Recorridos eulerianos.
- Concepto de grafo euleriano.
- Algoritmo de Fleury.
- Ejercicios resueltos.

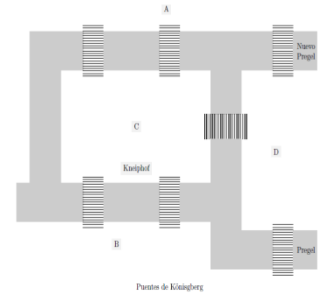
¿Podemos movernos por las aristas de un grafo comenzando en un vértice y volviendo a él después de pasar por cada arista exactamente una vez? ¿Podemos desplazarnos por las aristas de un grafo comenzando en un vértice y volviendo a él después de haber visitado todos los vértices sólo una vez? Estas cuestiones se refieren a los problemas de ***recorridos*** *de grafos que son aquellos que buscan la ruta óptima que recorre o bien todas o parte de las aristas/arcos de un grafo o bien que pasa por todos o algunos vértices. Son problemas propios de la rama matemática de Optimización Combinatoria que consiste en encontrar la solución óptima entre un número finito o infinito numerable de soluciones lo cual es aplicable a problemas reales como el reparto de mercancía, transporte, producción de circuitos electrónicos etc.*

El estudio de las rutas óptimas en un grafo teniendo en cuenta las aristas/arcos se debe al matemático suizo ***EULER*** (XVIII). Entre sus aplicaciones están la del *problema del Cartero Chino (CPP) o de la ruta de inspección y el problema del Cartero Rural (RPP).*

Por otro lado, los estudios de cómo encontrar el camino óptimo entre vértices pasando por todos o parte de ellos se debe al matemático ***HAMILTON.*** *Entre sus aplicaciones están la del problema del Viajante del comercio (TSP) y el problemas de las rutas de vehículos (VRP).*

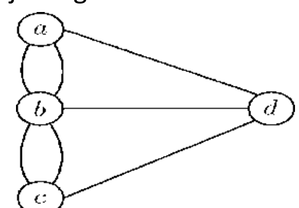
Problemas de recorridos por aristas/arcos

La primera referencia que tenemos de este tipo de problemas se debe al matemático suizo Leonard **Euler** (XVIII) al estudiar el problema de los puentes de la ciudad de Königsberg, que se enunciaba como:



> *"La ciudad de Königsberg, atravesada por el río Pregel, se dividía en cuatro regiones distintas, unidas a través de 7 puentes. Se plantea el problema de si es posible dar un paseo comenzando desde cualquiera de estas regiones, pasando por todos los puentes sólo una vez, y regresar al mismo punto de partida".*

En 1736 Euler publica *"Solutio problematis ad geometriam situs pertonentis"* en donde presenta la solución del problema mostrando que es imposible regresar al punto de partida. La abstracción del problema dio pie a la primera noción de grafo siendo, en particular, un multigrafo no dirigido sin bucles el que representaba la abstracción del mapa de Königsberg. La publicación de Euler es la primera que hace alusión a una geometría en que sólo interesan las propiedades estructurales de los objetos y no sus medidas. El matemático llama a esta nueva manera de ver los objetos geométricos «*geometriam situs*», término que hoy se traduce como topología.

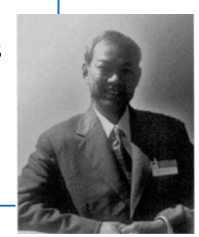


Grafo de Königsberg

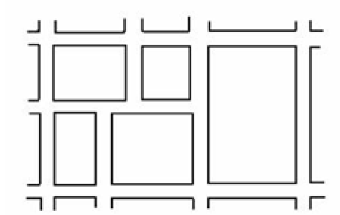
"La ruta del cartero chino" Planteado por el matemático **Kwan Mei-Ko** (1960):

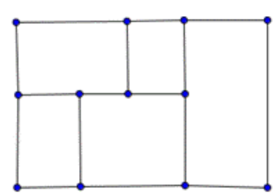
> *"Un cartero debe repartir la correspondencia recorriendo la menor distancia posible siendo la oficina su punto de partida y llegada".*

En teoría de grafos este problema consiste en encontrar un tour (no tiene que ser único) de coste mínimo que atraviese cada arista solo una vez . Que las aristas del grafo tengan costes (distancia) es irrelevante ya que veremos que al recorrer todas las aristas una sola vez se encuentra, de forma directa, un tour de mínimo coste. En el caso en que se tuvieran costes estos tendrían que ser no negativos ya que de no ser así el tour recorrería infinitas veces las aristas para minimizar el coste.



Mapa ejemplo
de la ruta del cartero





Formalización
mediante un grafo

>> Un **tour** es una cadena cerrada que atraviesa cada arista/arco al menos una vez.

>> Un **tour euleriano (TE)** es una cadena cerrada que recorre cada arista/arco exactamente una vez.

>> Un **grafo es eulerino** si tiene un tour euleriano.

Th_Euler-1. Sea G un grafo conexo y no dirigido, G es euleriano i.e., tiene un TE si, y sólo si el grado de todos sus vértices es par.

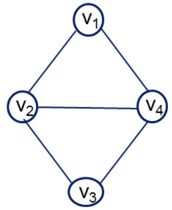
Dem/necesaria: Sup. que G tiene TE, vemos que todos sus vértices tienen grado par. Sup. que empezamos a recorrer el TE por un vértice *u* y que continua por una arista incidente con u, p.ej. {u,v}. Cada vez que el TE pase por un vértice contribuye con 2 al grado del vértice ya que tiene que entrar y salir por aristas diferentes. El TE termina en el vértice donde comienza, luego este vértice también tiene grado par. Evidente que los vértices intermedios por donde pasa el TE tiene grado par ya que el TE contribuye con 2 a su grado .

Dem/suficiente: Sup. que si G es conexo y todos sus vértices tienen grado par, entonces G tiene TE.
Si $|V| = 1$ o 2 es evidente que existe un TE ya que cuando $|V| = 1$ existe un bucle y cuando $|V| = 2$, p.ej. *u* y *v*, si $|E| = 2$ existe un TE, por ejemplo, uvu.
Si $|V| > 2$. Sean $u, v \in V$ vértices adyacentes. Como todos los vértices tienen grado par, existe un tour T1 que contiene, entre otras, a la arista *uv.*
Sea G' = (V,E'), el subgrafo de G tal que E' = E – T1. G' tiene todos sus vértices de grado par (o cero) ya que cada vértice aporta en el tour dos aristas.
Si E' = 0 entonces ya hemos conseguido TE → stop.
Si E' ≠ 0 seguimos. Elegimos un vértice cualquiera *w* que esté en T1.

- Si d(w) es par. Construimos un tour, T2.
- Si d(w) = 0, elegimos otro vértice *v* / d(v) sea par.

La primera arista del camino que une a ambos vértices en G y que no esté en T1 comenzará en un vértice de T1 no aislado en G'. Bastaría tomar este vértice y uno de sus adyacentes para hallar el tour T2. Unimos T1 con T2 de la siguiente forma: recorremos T1 hasta llegar al vértice elegido para construir T2 seguimos con este tour hasta terminarlo y luego se continua por T1. Se construye un tour T3 y se reitera el proceso hasta que no quedan aristas.

Ejemplo



El grafo no es GE porque tiene dos vértices de grado impar. Es imposible conseguir un TE.
¿Tiene un camino euleriano?

\>> Un **camino euleriano (CE)** es una cadena simple que atraviesa cada arista/arco exactamente una vez.

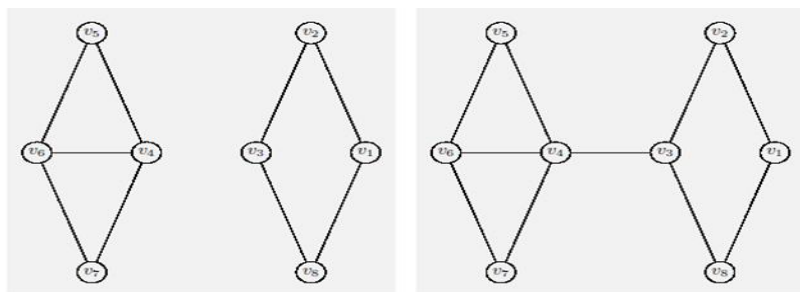
Th_Euler-2. Un grafo G conexo y no dirigido contiene un CE si, y sólo si, tiene exactamente dos vértices de grado impar.

Dem/necesaria**.** Suponemos que existe un CE en G, que llamaremos *P*, entre los vértices *u* y *v*. Sea G' el grafo que resulta de añadir la arista e = {u,v} a *G.* Entonces P U {e} es un TE en *G'* y entonces todos los vértices de *G'* tienen grado par.

Luego en *G* todos los vértices tiene grado par, menos los´vértices *u* y *v*.

Dem/suficiente. Sup. que *u* y *v* son los únicos vértices de grado impar en G. Sea e={u,v} entonces el grafo G' = G (V', E U {e}) tiene todos sus vértices de grado par y por lo tanto existe un TE, *P* en *G'*. Claramente P $\sim$ e es un CE en *G*.

Ejemplo



a) b)

El grafo a) no tiene CE porque no es conexo aunque las dos CC tengan exactamente 2 vértices de grado impar.

El grafo b) tiene, al menos, un CE porque además de ser conexo tiene exactamente 2 vértices de grado impar.

>> Para que un grafo sea euleriano es necesario que sea conexo.

Th_Euler-3. Sea G un grafo dirigido y débilmente conexo

a) G es euleriano si, y sólo si, de(v) = ds(v).

b) G tiene un CE si y sólo si, $d_e(v) = d_s(v), \forall v \neq p, q$ **y**

$$d_e(p) = d_s(p) - 1, \quad d_e(q) = d_s(q) + 1.$$

Siendo p y q los vértices inicial y final, respectivamente.

El grafo a) no es GE porque no es conexo aunque las dos CC tengan todos sus vértices con grado par.

El grafo b) es GE porque además de ser conexo todos sus vértices tienen grado par.

ALGORITMO de FLEURY para determinar CE o TE. **Coste O|A|**

Para que el problema tenga solución el grafo debe ser conexo en el caso GND y débilmente conexo en GD y si tiene costes en las aristas/arcos estos deben ser no negativos.

>> Sea G no dirigido:

Si el grafo tiene TE se elige un vértice cualquiera como comienzo del tour.

Si el grafo tiene CE se elige uno de los vértices de grado impar y se tiene que llegar al otro vértice.

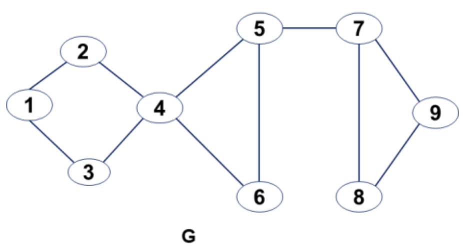
Al pasar por un vértice se elige una arista que formará parte del CE o TE y se elimina para no repetirla en el camino.

Dicha arista debe ser tal que al eliminarla no desconecte el grafo. Una vez eliminada, el vértice al que llegaba se marca como local y desde este vértice se vuelve a elegir otra arista que se añade al camino y se elimina.

Este proceso se repite hasta eliminar todas las aristas del grafo.

Arista de corte: arista de E que hace que el grafo **H = (V', E ~ {e}) no sea conexo.**

Arista de corte: {5,7}



Algoritmo de Fleury

Paso 1. $i \leftarrow 1$; Gi = G. Iniciar cadena (CE o TE), C = Φ.
Para CE: **elegir v/** d(v) es impar.
Para TE cualquier $v \in V$.
Etiquetar v: local.

Paso 2. **No hay arista incidente** con **v** → **parar**
C es CE o TE.

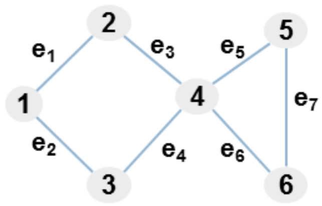
{5, 7}

Paso 3. **Sólo hay 1 arista** incidente con v / **e = {v,u}**
→ **Eliminar e** $E_{i+1} = E_i - \{e\}$
→ **Eliminar v** $V_{i+1} = V_i - \{v\}$

Paso 4. **Hay más de 1 arista** incidente con v → elegir e = {v,u} **no de corte**
→ **Eliminar e** $E_{i+1} = E_i - \{e\}$

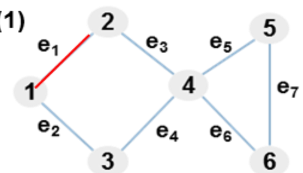
Paso 5. **Añadir** e = {v,u} a la cadena C = C U {e}.
Etiquetar u : local.
$i \leftarrow i + 1$,
Volver paso 2

EJEMPLO. ALGORITMO FLEURY, GND



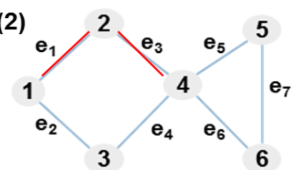
Como todos los vértices tienen grado par, **existe TE**.
Se busca.

(1)



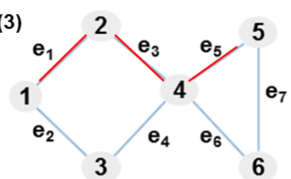
(1) i $\leftarrow$1,
TE = Φ . vértice inicial: 1
P4: aristas incidentes a 1: e_1,e_2. Eliminar:e_1
P5: añadir e_1 al tour → TE = e_1;
Vértice local: 2
i $\leftarrow$ 2

(2)



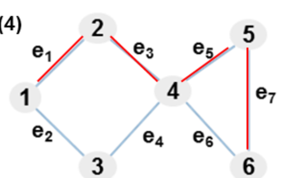
(2) i $\leftarrow$ 2
P3: arista incidente a 2: e_3. Eliminar: e_3
P5: añadir e_3 al tour → TE = e_1 e_3
Vértice local: 4
i $\leftarrow$ 3

(3)



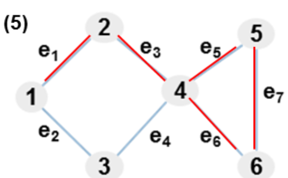
(3) i $\leftarrow$ 3
P4: aristas incidentes a 4: e_4 e_5 e_6. Eliminar:e_5
Ojo: arista de corte e_4 no elegirla
P5: añadir e_5 al tour → TE = e_1 e_3 e_5
Vértice local: 5
i $\leftarrow$ 4

(4)



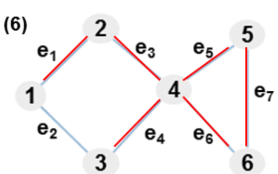
(4) i $\leftarrow$ 4
P3: arista incidente a 5: e_7. Eliminar:e_7
P5: añadir e_7 al tour → TE = e_1 e_3 e_5 e_7
Vértice local: 6
i $\leftarrow$ 5

(5)



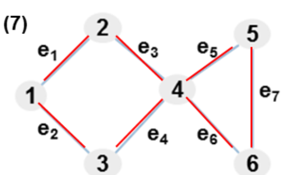
(5) i $\leftarrow$ 5
P3: arista incidente a 6: e_6. Eliminar:e_6
P5: añadir e_6 al tour → TE = e_1 e_3 e_5 e_7 e_6
Vértice local: 4
i $\leftarrow$ 6

(6)



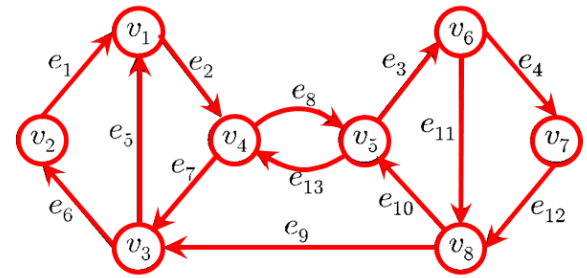
(6) i $\leftarrow$ 6
P3: arista incidente a 4: e_4. Eliminar:e_4
P5: añadir e_6 al tour → TE = e_1 e_3 e_5 e_7 e_6 e_4
Vértice local: 3
i $\leftarrow$ 7

(7)

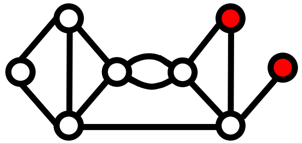


(7) i $\leftarrow$ 7
P3: arista incidente a 3: e_2. Eliminar:e_2
P5: añadir e_2 al tour → TE = e_1 e_3 e_5 e_7 e_6 e_4 e_2

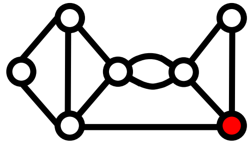
EJEMPLO. ALGORITMO FLEURY, GD



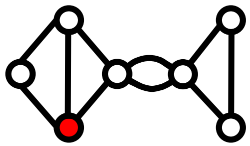
Iteration 1: T= e_4



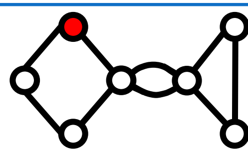
Iteration 2: T= e_4e_{12}



Iteration 3: T= $e_4e_{12}e_9$

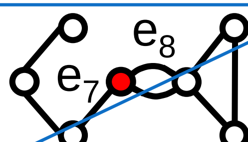


Iteration 4: T= $e_4e_{12}e_9e_5$

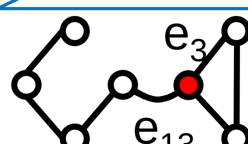


At this point, if we choose the arc ***e_7***, this arc is a cut edge of the undirected associated graph. Then, we have to choose ***e_8***.

Iteration 5: T= $e_4e_{12}e_9e_5e_2$

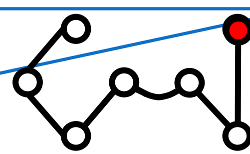


Iteration 6:
T= $e_4e_{12}e_9e_5e_2e_8$

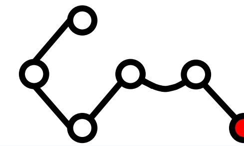


At this point, if we choose the arc ***e_{13}***, this arc is a cut edge of the undirected associated graph. Then, we have to choose ***e_3***.

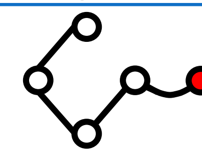
Iteration 7:
T= $e_4e_{12}e_9e_5e_2e_8e_3$



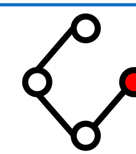
Iteration 8:
T= $e_4e_{12}e_9e_5e_2e_8e_3e_{11}$



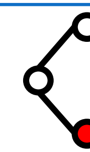
Iteration 9:
T= $e_4e_{12}e_9e_5e_2e_8e_3e_{11}e_{10}$



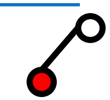
Iteration 10:
T= $e_4e_{12}e_9e_5e_2e_8e_3e_{11}e_{10}e_{13}$



Iteration 11:
T= $e_4e_{12}e_9e_5e_2e_8e_3e_{11}e_{10}e_{13}e_7$



Iteration 12: T= $e_4e_{12}e_9e_5e_2e_8e_3e_{11}e_{10}e_{13}e_7e_6$



Euler trail

Iteration 13: T= $e_4e_{12}e_9e_5e_2e_8e_3e_{11}e_{10}e_{13}e_7e_6e_1$

Bibliografía

>Migallón V., Penadés J., Matemática discreta, Puntero y Chip, Alicante.

>Grassman, W.K. and Tremblay, J.P., Matemática discreta y Lógica, Prentice Hall.

>Rosen K.H., Matemática discreta y aplicaciones, Editorial McGraw-Hill.

>Johnsonbaugh, R., Matemáticas discretas, Prentice Hall.

>Grimaldi, R.P., Matemáticas discretas y combinatoria, Prentice Hall