13. Recordar que el *vecindario* de un vértice v es el conjunto $N(v)$ que contiene a todos los vértices adyacentes a v. El *vecindario cerrado* es $N[v] = N(v) \cup \{v\}$. Dos vértices u y v son *gemelos* cuando $N(u) = N(v)$, mientras que son *mellizos* cuando $N[u] = N[v]$ (Figura 2).

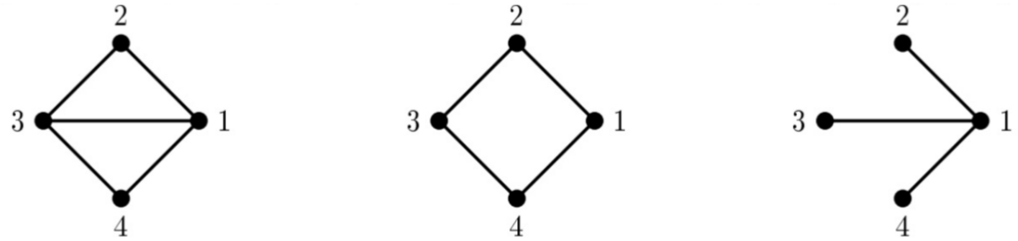


FIGURA 2. *El grafo diamante (izquierda), el grafo C_4 (centro) y el grafo garra (derecha). Los vértices 1 y 3 son mellizos en el diamante porque $N[1] = N[3] = \{1, 2, 3, 4\}$ mientras que 2 y 4 son gemelos porque $N(2) = N(4) = \{1, 3\}$. En la garra, $N(2) = N(3) = N(4) = \{1\}$ y, por lo tanto, 2, 3 y 4 son gemelos, mientras que en C_4 la partición en gemelos es $\{2, 4\}$ y $\{1, 3\}$. El diamante tiene 2 triángulos $\{1, 2, 3\}$ y $\{1, 3, 4\}$ mientras que C_4 y la garra no tienen triángulos. Notar que el diamante es threshold porque $N(2) = N(4)$ y $N[2] \subseteq N[1] = N[3]$; ciertamente $(2, 4, 1, 3)$ es una descomposición threshold del diamante. En cambio, C_4 no es threshold porque tanto $N(1)$ y $N(2)$ como $N[1]$ y $N[2]$ son incomparables por inclusión. Finalmente, la garra es threshold (por qué?).*

a) Observar que las relaciones de gemelos y mellizos son relaciones de equivalencia (i.e., son reflexivas, transitivas y simétricas).

b) Probar que el siguiente algoritmo encuentra la partición de $V(G)$ en vértices mellizos. **Ayuda:** demostrar por invariante que, luego del paso i, u y w pertenecen al mismo conjunto de $\mathcal{P}_i$ si y sólo si $N[u] \cap \{v_1, \ldots, v_i\} = N[w] \cap \{v_1, \ldots, v_i\}$.

1. Sea $\mathcal{P}_0 = \{V(G)\}$ ($\mathcal{P}$ es un conjunto de conjuntos)
2. Sea $v_1, \ldots, v_n$ un ordenamiento cualquiera de $V(G)$.
3. Para i desde 1 hasta n:
4. Poner $\mathcal{P}_i := \{W \cap N[v_i] \mid W \in \mathcal{P}_{i-1}\} \cup \{W \setminus N[v_i] \mid W \in \mathcal{P}_{i-1}\}$.
5. $\mathcal{P}_n$ es la partición buscada.

c) Describir la implementación del algoritmo, especificando las estructuras de datos utilizadas. La mejor implementación que conocemos tiene complejidad temporal $O(n + m)$.

d) ¿Qué debería modificarse para que el algoritmo encuentre la partición en vértices gemelos?

(a) QVQ ser gemelos (mellizos) es una relación de equivalencia

Reflexiva $x \sim x$: sea v un vértice de G, v es gemelo (mellizo) con v ya que $N(v) = N(v)$ / $N[v] = N[v]$

Simetría $x \sim y$, $y \sim x$ → si v y w son gemelos (mellizos) → w y v también ya que $N(v) = N(w)$ / $N[v] = N[w]$

→ $N(w) = N(v)$ / $N[w] = N[v]$

transitividad $x \sim y$ y $y \sim z$ → $x \sim z$: sean v, w, u vértice de G / v es gemelo (mellizo) con w

y w es gemelo (mellizo) con u

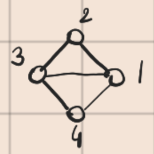
$N[v]=N[w]$ $\quad N[w]=N[u]$ $\quad N[w]=N[v]$

$\longrightarrow N(v)=N(w)$ y $N(w)=N(u)$ $\longrightarrow N(w)=N(v)$ pero

$N(w)=N(u)$ $\quad N[w]=N[u]$

$\longrightarrow N(v)=N(u)$

$N[v]=N[u]$

(b) Entendamos el algoritmo para 

Pre: tiene mellizos

1. $P_0=\{\{1,2,3,4\}\}$
2. ord $=[3,2,1,4]$
3. Para $i=1$ $\rightarrow P_1=\{\overset{W_1}{\{1,2,3,4\}}\cup\overset{W_2}{\{\emptyset\}}\}$

$\rightarrow P_2=\{\overset{W_3}{\{2,3,1\}}\cup\overset{W_4}{\{4\}}\}$

$P_3=\{\{1,2,3\}\cup\{\emptyset\}\cup\{4\}\cup\{\emptyset\}\}$

$P_4=\{\{1,3\}\cup\{2\}\cup\{4\}\}$

El algoritmo es correcto ya que por como está construido la única forma de que 2 vértices pertenezcan al mismo conjunto/partición es si $W\cap N[w]=W\cap N[v]$, es decir, tengan los mismos vecinos

ⓒ Implementación

SonMellizos? (lista de aristas G)

Pasamos de lista de aristas a adyacencias $O(n+m)$

Vector marcas / marcas[v] = v $O(n)$

$O(n+m)$ {

for (i=1 ; i < n+1 ; i++){ $O(n)$

for (j=1 ; j < |vecindario| ; j++){ $O(d(v))$

if obtenerVecindarioCerrado(G[j]) == obtenerVecindarioCerrado(G[i]) $O(1)$

marcas[i] = i ;

marcas[j] = j

endif

endfor

endfor

}

Recorro marcas y devuelvo agrupados los vértices con las mismas marcas

ⓓ Debería comparar vecindarios abiertos