

Complejidad del algoritmo de Demoucron , Malgrange y Pertuiset

Definiciones y notaciones

Dado un grafo conexo $G = (V(G), E(G))$, una representación planar R de un subgrafo H de G es un conjunto de regiones. Donde cada región r_i está conformado por un conjunto de nodos $V(r_i)$ y otro de aristas fronteras $E(r_i)$. Claramente, $V(H) = \bigcup_{r_i \in R} V(r_i) = V(R)$ y $E(H) = \bigcup_{r_i \in R} E(r_i) = E(R)$. El algoritmo de Demoucron , Malgrange y Pertuiset parte de una representación planar R de un ciclo de G que es el subgrafo inicial, luego en cada iteración modifica R para que sea la representación planar de un subgrafo mayor, hasta que llega a abarcar todo el grafo G o concluir que G es no planar. En cada iteración, agrupa los elementos de G no representado en R en “partes”. Existen dos tipos de partes:

- (i) arista (colgante) $e \in E(G) \cap (V(R) \times V(R)) \setminus E(R)$
- (ii) una componente conexa c de $G \setminus V(R)$ más aristas (también se llaman colgantes) que tienen un extremo en la componente y otro extremo en $V(R)$. Llamamos $V(p) = V(c)$ al subconjunto de nodos de la parte p que son exactamente los vértices de la componente conexa c .

Llamamos P_R^1 el conjunto de partes de tipo (i) y P_R^2 el de las de tipo (ii). A continuación, daremos más definiciones que son necesarias para el análisis de complejidad:

- $contact_R(v) = \{w \in V(R) / (v, w) \in E(G)\}$, los nodos de contacto de un vértice $v \in V(G) \setminus V(R)$.
- $contact_R(p) = \{x, y\}$, los nodos de contacto de una parte $p \in P_R^1$, siendo p la arista (x, y) .
- $contact_R(p) = \bigcup_{v \in V(p)} contact_R(v)$, los nodos de contacto de una parte $p \in P_R^2$.
- $F_R(w) = \{r_i \in R / w \in V(r_i)\}$, las regiones que contienen al nodo $w \in V(R)$.
- $F_R(p) = \bigcap_{w \in contact_R(p)} F_R(w)$, las regiones donde p es potencialmente dibujable.
- $F_R(v) = \bigcap_{w \in contact_R(v)} F_R(w)$, las regiones donde $v \in V(G) \setminus V(R)$ es potencialmente dibujable. Se puede verificar fácilmente que para cualquier parte $p \in P_R^2$, $F_R(p) = \bigcap_{v \in V(p)} F_R(v)$.

Algoritmo de Demoucron , Malgrange y Pertuiset

Sea $R :=$ una representación planar de cualquier ciclo de un grafo conexo G

Mientras $E(R) \neq E(G)$ **hacer**

Calcular $F_R(p)$, $\forall p \in P_R^1 \cup P_R^2$

Si para algun $p \in P_R^1 \cup P_R^2$, $F_R(p) = \emptyset$

entonces Retornar FALSO

Si para algun $p \in P_R^1 \cup P_R^2$, $F_R(p) = \{f\}$

Entonces elegir p y f

Sino elegir cualquier p y $f \in F_R(p)$

Si p tiene dos aristas colgantes distintos e y e'

Entonces buscar un camino/circuito q que comienza con e y termina con e'

Sino q es el camino formado por la unica arista colgante

$R := R \cup q$

Retornar VERDADERO y R representación planar de G

Análisis de Complejidad - $O(n^2)$

Claramente, la cantidad de iteraciones es $O(n)$, pues en cada iteración se incorpora a R al menos una arista y la cantidad de arista de un grafo planar es a lo sumo $3n - 6$. Por otro lado, tanto P_R^1 como P_R^2 se pueden calcular fácilmente en tiempo $O(n)$. El problema radica en como calcular $F_R(p), \forall p \in P_R^1 \cup P_R^2$ en tiempo $O(n)$ para que la complejidad total no sea mayor que $O(n^2)$. Para la primera iteración es fácil de lograrlo porque R tiene exactamente dos regiones en este caso. Podemos llamar a estas regiones como r_1 y r_2 y es fácil de ver que $F_R(w) = F_R(v) = F_R(p) = \{r_1, r_2\}$. Vamos a analizar el resto de las iteraciones. Por ahora para poder calcular $F_R(p)$, necesitamos previamente calcular explícitamente $F_R(w)$ y $F_R(v)$, más adelante vamos a encontrar una manera de evitarlo. Sea R la representación de la iteración anterior donde sean p la parte elegida y r_j la región elegida para generar la representación planar $R' = R \cup q$ de esta iteración. Claramente, $R \setminus R' = \{r_j\}$ y $R' \setminus R = \{r_{j_1}, r_{j_2}\}$ o $R = \{r_{j_1}\}$, ocurre este último caso cuando $p \in P_R^2$ y tiene una única arista colgante (vamos a suponer que ocurre el primer caso que es el más complicado y se pueden adaptar fácilmente las consideraciones para el segundo caso). En esta iteración, vamos a calcular $F_{R'}(p), \forall p \in P_{R'}^1 \cup P_{R'}^2$. Para ello, vamos a considerar en primer lugar los valores de $F_{R'}(w), \forall w \in V(R')$. Se puede observar que $V(R') = V(R) \cup V(q)$ y es posible calcular $F_{R'}(w)$ de acuerdo a los siguientes caso:

- (a) si $w \in V(R') \setminus V(R)$ entonces $F_{R'}(w) = R' \setminus R = \{r_{j_1}, r_{j_2}\} = \{r_{j_1}\} \cup \{r_{j_2}\}$
- (b) si $w \in V(R) \setminus V(r_j)$ entonces $F_{R'}(w) = F_R(w) = F_R(w) \setminus \{r_j\}$
- (c) si $w \in V(r_j) \cap V(r_{j_1})$ y $w \notin V(r_{j_2})$ entonces $F_{R'}(w) = F_R(w) \setminus \{r_j\} \cup \{r_{j_1}\}$
- (d) si $w \in V(r_j) \cap V(r_{j_2})$ y $w \notin V(r_{j_1})$ entonces $F_{R'}(w) = F_R(w) \setminus \{r_j\} \cup \{r_{j_2}\}$
- (e) si $w \in V(r_j) \cap V(r_{j_1}) \cap V(r_{j_2})$ entonces $F_{R'}(w) = F_R(w) \setminus \{r_j\} \cup \{r_{j_1}, r_{j_2}\} = F_R(w) \setminus \{r_j\} \cup \{r_{j_1}\} \cup \{r_{j_2}\}$

En realidad, no vamos a computar explícitamente $F_{R'}(w), \forall w \in V(R')$ sino un número de 3 bits $f_{R'}(w), \forall w \in V(R')$, donde bit 1 representa a $F_R(w) \setminus \{r_j\}$, bit 2 a $\{r_{j_1}\}$ y bit 3 a $\{r_{j_2}\}$. Entonces para los distintos casos que consideramos anteriormente tenemos:

- (a) si $w \in V(R') \setminus V(R)$ entonces $f_{R'}(w) = 011$
- (b) si $w \in V(R) \setminus V(r_j)$ entonces $f_{R'}(w) = 100$
- (c) si $w \in V(r_j) \cap V(r_{j_1})$ y $w \notin V(r_{j_2})$ entonces $f_{R'}(w) = 110$
- (d) si $w \in V(r_j) \cap V(r_{j_2})$ y $w \notin V(r_{j_1})$ entonces $f_{R'}(w) = 101$
- (e) si $w \in V(r_j) \cap V(r_{j_1}) \cap V(r_{j_2})$ entonces $f_{R'}(w) = 111$

Claramente, en tiempo $O(n)$ se puede computar $f_{R'}(w), \forall w \in V(R')$. Luego podemos computar $f_{R'}(v) = \bigwedge_{w \in \text{contact}(v)} f_{R'}(w), \forall v \in V(G) \setminus V(R')$ también en el mismo tiempo porque para cada $f_{R'}(v)$ cuesta tiempo $O(d(v))$ y el tiempo total sería $O(m) = O(n)$. Por último, calcular $f_{R'}(p) = f_{R'}(w_1) \wedge f_{R'}(w_2), \forall p \in P_{R'}^1$ donde p es la arista (w_1, w_2) y $f_{R'}(p) = \bigwedge_{w \in V(p)} f_{R'}(w), \forall p \in P_{R'}^2$, nuevamente esto se hace en tiempo $O(n)$. Una vez calculado $f_{R'}(p)$, es fácil ver que se puede generar $F_{R'}(p)$ para todas las partes en tiempo $O(n)$ (salvo las partes nuevas que son las partes $p \in (P_{R'}^2 \setminus P_R^2) \cup (P_{R'}^1 \setminus P_R^1)$ que veremos más adelante) habiendo implementado $F_R(p)$ y $F_R^{-1}(r_i)$ como listas doblemente encadenadas de punteros a regiones y partes, respectivamente. Por último, se puede observar que si la parte elegida p^* es de tipo (i) entonces

$P_{R'}^1 = P_R^1 \setminus \{p^*\}$ y $P_{R'}^2 = P_R^2$ (lo cual implica que $(P_{R'}^2 \setminus P_R^2) \cup (P_{R'}^1 \setminus P_R^1) = \emptyset$) y si p^* es de tipo (ii) entonces $V(q) \subseteq V(p^*)$, $V(q) \setminus V(R) \neq \emptyset$, $P_R^2 \setminus P_{R'}^2 = \{p^*\}$, $P_R^1 \setminus P_{R'}^1 = \emptyset$ y las partes que están en $(P_{R'}^2 \setminus P_R^2) \cup (P_{R'}^1 \setminus P_R^1)$ son subpartes de p^* . Ahora bien, si una parte $p \in (P_{R'}^2 \setminus P_R^2) \cup (P_{R'}^1 \setminus P_R^1)$ entonces p^* es de tipo (ii) y podemos concluir que p está contenida estrictamente en p^* y tiene al menos una arista colgante con un extremo w en $V(q) \setminus V(R)$. Esto significa que $f_{R'}(w) = 011$ y $f_{R'}(p) \in \{000, 001, 010, 011\}$ y eso implica que $F_{R'}(p) \subseteq \{r_{j_1}, r_{j_2}\}$. Consecuentemente, las partes nuevas no afectan la generación de los $F_{R'}(p)$ en tiempo lineal. Es más, en cada iteración el algoritmo solamente elimina la parte elegida p^* e incorpora las partes nuevas.