

Universidad Nacional de Córdoba

# Segmentación interactiva de imágenes

Trabajo final del curso de posgrado

## Teoría de grafos

*Mario Alejandro García*

*DNI 25141066*

*Universidad Tecnológica Nacional*

*Facultad Regional Córdoba*

`mgarcia@frc.utn.edu.ar`

`malejandrogarcia@gmail.com`

# Contenido

<b>1</b>	<b>Introducción</b>	<b>2</b>
1.1	Imágenes digitales . . . . .	2
1.2	Segmentación de imágenes . . . . .	2
<b>2</b>	<b>Representación de la imagen mediante un grafo</b>	<b>2</b>
2.1	Función de energía . . . . .	3
2.2	Representación de la imagen en un grafo . . . . .	3
<b>3</b>	<b>Segmentación y corte</b>	<b>4</b>
<b>4</b>	<b>Algunos conceptos necesarios de teoría de grafos</b>	<b>6</b>
4.1	Red . . . . .	6
4.2	Flujo . . . . .	6
4.3	Capacidad residual . . . . .	6
4.4	Red residual . . . . .	6
<b>5</b>	<b>Flujo máximo/corte mínimo</b>	<b>6</b>
<b>6</b>	<b>Método de segmentación</b>	<b>7</b>
6.1	Definiciones . . . . .	8
6.2	Etapas . . . . .	8
<b>7</b>	<b>Ejemplo de cálculo</b>	<b>9</b>

# 1 Introducción

En este trabajo se muestra un caso de aplicación de la teoría de grafos. El problema a resolver es la segmentación de imágenes digitales. Se presenta una técnica basada en la representación de la imagen como un grafo y la determinación de las fronteras entre los objetos de la imagen como la solución de un problema de flujo máximo/corte mínimo en el grafo asociado.

## 1.1 Imágenes digitales

Una imagen digital es una matriz formada por píxeles. Un pixel, acrónimo de *picture element*, es la menor unidad homogénea en color e intensidad de una imagen. En este trabajo se utilizan imágenes en escalas de grises, por lo tanto, existe solo un valor numérico por pixel. El valor del pixel indica la intensidad luminica en esa región de la imagen. Los valores válidos de intensidad son los enteros entre 0 y  $2^n - 1$ , donde  $n$  es la cantidad de bits elegidos para codificar la imagen. El 0 representa al color negro,  $2^n - 1$  al blanco y los valores en  $\{1, \dots, 2^n - 2\}$  a los grises intermedios.

## 1.2 Segmentación de imágenes

En procesamiento de imágenes se denomina segmentación al proceso de extraer (es decir, identificar y aislar) regiones de interés en una imagen. Al igual que Boykov y Jolly en [1], este trabajo se enfoca en la segmentación del tipo figura-fondo, donde se busca dividir la imagen en dos segmentos (segmentación binaria), el objeto de interés y el fondo. La salida del proceso de segmentación es una lista por cada región segmentada conteniendo los píxeles que la componen. Tal como en [1, 2, 3], la técnica presentada realiza una segmentación interactiva porque se espera que un usuario señale por lo menos un pixel del objeto y un pixel del fondo.

# 2 Representación de la imagen mediante un grafo

Los píxeles seleccionados y asignados a la figura o al fondo se llaman semillas. Las semillas imponen restricciones duras (*hard constraints*) a la segmentación. Intuitivamente, estas restricciones proveen pistas sobre lo que el usuario pretende segmentar.

El resto de la imagen se segmenta automáticamente mediante el cálculo de un óptimo global entre todas las segmentaciones que satisfacen las restricciones duras. La función de costo o energía se define en términos de dos propiedades de los segmentos de la imagen, límite y región. Estas propiedades pueden ser vistas como restricciones blandas para la segmentación.

## 2.1 Función de energía

Considerando un conjunto de píxeles (vértices)  $P$  y algún tipo de vecindad<sup>1</sup> representada por el conjunto  $N$  de todos los pares no ordenados<sup>2</sup>  $\{p, q\}$  para  $p, q \in P$  y  $p, q$  vecinos en la imagen, se define a  $A = (A_1, \dots, A_p, \dots, A_{|P|})$  como un vector binario cuyos componentes  $A_p$  determinan la asignación de cada uno de los píxeles  $p \in P$  a la figura o al fondo. Cada  $A_p$  puede valer "obj" o "bkg", abreviaciones de "object" (figura) y "background" (fondo). El vector  $A$  determina la segmentación. Entonces, las restricciones blandas asociadas a las propiedades región y borde se describen por la función de energía  $E(A)$ :

$$E(A) = \lambda R(A) + B(A) \quad (1)$$

donde

$$R(A) = \sum_{p \in P} R_p(A_p) \quad (2)$$

$$B(A) = \sum_{\{p, q\} \in N} B_{\{p, q\}} \delta(A_p, A_q) \quad (3)$$

y

$$\delta(A_p, A_q) = \begin{cases} 1 & \text{si } A_p \neq A_q \\ 0 & \text{si } A_p = A_q \end{cases}$$

El coeficiente  $\lambda \geq 0$  en (1) especifica la importancia relativa del término  $R(A)$  (región) versus el término  $B(A)$  (borde). El término  $R(A)$  asume que las penalidades individuales por asignarle a un píxel  $p$  el valor "obj" o "bkg" son dadas. Por ejemplo,  $R_p(\cdot)$  podría reflejar cómo la intensidad del píxel  $p$  se ajusta a una distribución de intensidad definida por los valores de las semillas.

El término  $B(A)$  aporta la información relativa a los bordes, es decir, la frontera entre el objeto y el fondo. El coeficiente  $B_{\{p, q\}} \geq 0$  se debe interpretar como una penalidad por la discontinuidad (corte) entre  $p$  y  $q$ . Generalmente  $B_{\{p, q\}}$  es grande cuando los píxeles  $p$  y  $q$  tienen intensidades similares y  $B_{\{p, q\}}$  es cercano a cero cuando las intensidades son muy diferentes. La penalidad  $B_{\{p, q\}}$  puede estar basada en el gradiente local de la intensidad.

## 2.2 Representación de la imagen en un grafo

Se crea un grafo pesado no dirigido  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , donde  $\mathcal{V}$  es el conjunto de vértices y  $\mathcal{E}$  el conjunto de aristas. Cada píxel de la imagen es un vértice de  $\mathcal{G}$ . Se añaden además los vértices terminales  $s$  y  $t$ , llamados "fuente" y "sumidero" respectivamente. Entonces,  $\mathcal{V} = \mathcal{P} \cup \{s, t\}$ , donde cada vértice en  $\mathcal{P}$  representa un píxel en  $P$ . El vértice  $s$  está asociado al objeto y el vértice  $t$  al fondo.

<sup>1</sup>La vecindad de un píxel se define como el conjunto de píxeles adyacentes a un píxel dado. Existen distintas convenciones para definir vecindad. Por ejemplo, para un píxel  $p$ , con coordenadas  $(x, y)$ , la 4-vecindad ( $N_4$ ) es el conjunto de los píxeles con coordenadas  $(x-1, y-1)$ ,  $(x+1, y+1)$ ,  $(x-1, y+1)$  y  $(x+1, y-1)$ .

<sup>2</sup>La elección de pares  $\{p, q\}$  ordenados implicaría la existencia de dos pares  $(p, q)$  y  $(q, p)$ . La elección de vecinos ordenados, como en [2] y [3], determina la necesidad de un grafo dirigido, mientras que para los vecinos no ordenados, como en [1], se usa un grafo no dirigido.

El conjunto  $\mathcal{E}$  está formado por dos tipos de aristas: *n-links* (aristas de vecindad) y *t-links* (aristas terminales). Cada arista  $e \in \mathcal{E}$  tiene asignado un peso no-negativo  $w_e$ .

Cada vértice en  $\mathcal{P}$  tiene dos aristas *t-link*  $\{s, p\}$  y  $\{p, t\}$  conecátandolo con cada uno de los vértices terminales. Por otro lado, cada par de vértices en  $\mathcal{N}$  (correspondientes a los pares de pixeles en  $N$ ) está conectado por un *n-link*.

Asumiendo que  $\mathcal{O}$  y  $\mathcal{B}$  son los subconjuntos de  $\mathcal{P}$  con pixeles semilla marcados como "obj" y "bkg", se cumple que  $\mathcal{O} \cap \mathcal{B} = \emptyset$ .

La tabla 1 muestra la asignación de pesos a las aristas.

Arista	Peso	Para
$\{p, q\}$	$B_{\{p, q\}}$	$\{p, q\} \in \mathcal{N}$
$\{s, p\}$	$\lambda R_p(\text{"bkg"})$	$p \in \mathcal{P}, p \notin \mathcal{O} \cup \mathcal{B}$
	$K$	$p \in \mathcal{O}$
	$0$	$p \in \mathcal{B}$
$\{p, t\}$	$\lambda R_p(\text{"obj"})$	$p \in \mathcal{P}, p \notin \mathcal{O} \cup \mathcal{B}$
	$0$	$p \in \mathcal{O}$
	$K$	$p \in \mathcal{B}$

Tabla 1: Pesos de las aristas del grafo  $\mathcal{G}$ .

donde

$$K = 1 + \max_{p \in \mathcal{P}} \sum_{q: \{p, q\} \in \mathcal{N}} B_{\{p, q\}}$$

El valor de  $K$  es mayor que la suma de los pesos de los *n-links* conectados a cualquier vértice. Esta es una forma de asegurar que el método (que se verá más adelante) no cortará las aristas de las restricciones duras ( $\{s, p\}$  o  $\{p, t\}$  para  $p \in \mathcal{O} \cup \mathcal{B}$ ).

### 3 Segmentación y corte

La segmentación que minimiza la función de energía coincide con el corte de costo mínimo  $\hat{\mathcal{C}}$  en el grafo  $\mathcal{G}$  tal que los vértices  $s$  y  $t$  queden en dos componentes conexas distintas.

El corte de costo mínimo  $\hat{\mathcal{C}}$  define la segmentación  $\hat{A}$ . A continuación se prueba que  $\hat{A}$  es óptima. Para probarlo es necesario un lema. Se asume que  $\mathcal{F}$  es el conjunto de todos los cortes posibles  $\mathcal{C}$  sobre el grafo  $\mathcal{G}$  tal que

- $\mathcal{C}$  corta exactamente un *t-link* por cada  $p$
- $\{p, q\} \in \mathcal{C}$  si  $p$  y  $q$  están conectados a terminales distintos
- si  $p \in \mathcal{O}$  entonces  $\{p, t\} \in \mathcal{C}$
- si  $p \in \mathcal{B}$  entonces  $\{s, p\} \in \mathcal{C}$

**Lema 1.** *Existe el corte de costo mínimo sobre  $\mathcal{G}$ , es decir  $\hat{\mathcal{C}} \in \mathcal{F}$ .*

*Demostración.*  $\hat{\mathcal{C}}$  es un corte que separa los terminales, por lo tanto corta al menos un  $t$ -link por cada pixel. Por otro lado,  $\hat{\mathcal{C}}$  no puede cortar los dos  $t$ -links de un pixel porque, en ese caso, no sería mínimo dado que uno de los  $t$ -link se podría eliminar del corte. De la misma forma, un corte mínimo debe cortar un  $n$ -link  $\{p, q\}$  si  $p$  y  $q$  están conectados a terminales distintos porque cualquier corte debe separar los terminales. Si  $p$  y  $q$  están conectados al mismo terminal,  $\hat{\mathcal{C}}$  no debería cortar innecesariamente el  $n$ -link  $\{p, q\}$  dada su minimalidad. Las últimas dos propiedades son ciertas porque la constante  $K$  es mayor que la suma de todos los  $n$ -links para cualquier pixel  $p$ . Por ejemplo, si  $p \in \mathcal{O}$  y  $\hat{\mathcal{C}}$  corta  $\{s, p\}$  (con costo  $K$ ), se podría construir un corte de costo menor restaurando  $\{s, p\}$  y cortando todos los  $n$ -links de  $p$  más el  $t$ -link  $\{p, t\}$  (este último con costo 0).  $\square$

Para cada corte  $\mathcal{C} \in \mathcal{F}$  se puede definir una única segmentación correspondiente  $A(\mathcal{C})$  tal que

$$A_p(\mathcal{C}) = \begin{cases} \text{"obj"} & \text{si } \{p, t\} \in \mathcal{C} \\ \text{"bkg"} & \text{si } \{s, p\} \in \mathcal{C} \end{cases} \quad (4)$$

La definición anterior es coherente porque todo corte elimina uno de los dos  $t$ -links para cada pixel  $p$ . El lema 1 muestra que un corte mínimo  $\hat{\mathcal{C}}$  es posible. Entonces, se puede definir una segmentación correspondiente  $\hat{A} = A(\hat{\mathcal{C}})$ .

**Teorema 1.** *La segmentación  $\hat{A} = A(\hat{\mathcal{C}})$  definida por el corte de costo mínimo  $\hat{\mathcal{C}}$  como en (4) minimiza (1) entre todas las segmentaciones posibles satisfaciendo las restricciones*

$$\forall p \in \mathcal{O}, \quad A_p = \text{"obj"} \quad (5)$$

$$\forall p \in \mathcal{B}, \quad A_p = \text{"bkg"} \quad (6)$$

*Demostración.* Usando la tabla 1, la definición de  $\mathcal{F}$  y la ecuación (4), se puede mostrar que el costo de cualquier corte  $\mathcal{C} \in \mathcal{F}$  es

$$\begin{aligned} |\mathcal{C}| &= \sum_{p \notin \mathcal{O} \cup \mathcal{B}} \lambda R_p(A_p(\mathcal{C})) + \sum_{\{p, q\} \in \mathcal{N}} B_{\{p, q\}} \delta(A_p(\mathcal{C}), A_q(\mathcal{C})) \\ &= E(A(\mathcal{C})) - \sum_{p \in \mathcal{O}} \lambda R_p(\text{"obj"}) - \sum_{p \in \mathcal{B}} \lambda R_p(\text{"bkg"}) \end{aligned}$$

Por lo tanto,  $|\mathcal{C}| = E(A(\mathcal{C})) - \text{const}$ . Notar que para cualquier  $\mathcal{C} \in \mathcal{F}$  la asignación  $A(\mathcal{C})$  satisface las restricciones (5, 6). La ecuación (4) da una correspondencia uno a uno entre el conjunto de todos los cortes posibles en  $\mathcal{F}$  y el conjunto  $\mathcal{H}$  de todas las asignaciones  $A$  que satisfacen las restricciones (5, 6). Entonces,

$$E(\hat{A}) = |\hat{\mathcal{C}}| + \text{const} = \min_{\mathcal{C} \in \mathcal{F}} |\mathcal{C}| + \text{const} = \min_{\mathcal{C} \in \mathcal{F}} E(A(\mathcal{C})) = \min_{\mathcal{C} \in \mathcal{H}} E(A)$$

y el teorema queda probado.  $\square$

## 4 Algunos conceptos necesarios de teoría de grafos

### 4.1 Red

Si  $v$  es un vértice en un grafo dirigido y  $v$  no tiene aristas entrantes, se dice que  $v$  es una fuente, mientras que si  $v$  no tiene aristas saliente, es un sumidero.

Una red es un grafo dirigido que tiene exactamente una fuente y un sumidero. Los vértices se llaman nodos y las aristas se llaman arcos.

Si los arcos de la red tienen capacidades asignadas, la red se puede definir en base a un grafo dirigido  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  que incluye los vértices terminales (fuente y sumidero) más la función de capacidades

$$c : \mathcal{E} \rightarrow \mathbb{R}_+$$

Entonces, la red  $\mathcal{M}$  queda definida como  $\mathcal{M} = \{\mathcal{G}, c\}$ .

### 4.2 Flujo

Se considera una red con  $n$  nodos, una fuente  $v_1$  y un sumidero  $v_n$ , a través de la cual fluye cierto elemento o sustancia desde  $v_1$  hasta  $v_n$ . Se asume que existe una capacidad máxima de flujo  $u_{ij}$  para cada arco  $\{v_i, v_j\}$ . Se asume también que hay una *conservación de flujo* en cada nodo distinto de la fuente y el sumidero. Es decir, que la suma de las cantidades que fluyen hacia el interior de un nodo intermedio  $v$  es igual a la cantidad que fluye hacia afuera de  $v$  para todo  $v$  excepto por  $v_1$  y  $v_n$ .

### 4.3 Capacidad residual

La capacidad  $c$  de un arco  $\{p, q\}$  determina la cantidad máxima de flujo  $f$  que puede ir de  $p$  a  $q$  por el arco  $\{p, q\}$ . La capacidad residual  $c_f$  en un arco  $\{p, q\}$  se define como el máximo flujo adicional que puede ir de  $p$  a  $q$  a través de los arcos  $\{p, q\}$  y  $\{q, p\}$ .

$$c_f(\{p, q\}) = \begin{cases} c(\{p, q\}) - f(\{p, q\}) & \text{si } f(\{p, q\}) \geq 0 \\ c(\{p, q\}) + f(\{q, p\}) & \text{si } f(\{q, p\}) > 0 \end{cases}$$

### 4.4 Red residual

Se define la red residual  $\mathcal{M}_f$  de la red  $\mathcal{M} = \{\mathcal{G}, c\}$  como  $\mathcal{M}_f = \{\mathcal{G}, c_f\}$ .

## 5 Flujo máximo/corte mínimo

El problema de la segmentación se reduce ahora a encontrar el corte de costo mínimo  $\hat{C}$  en  $\mathcal{G}$ .

Según el teorema *Max-Flow/Min-Cut*, en una red  $\mathcal{M}$ , el mayor flujo posible entre la fuente y el sumidero es igual al corte de costo mínimo en  $\mathcal{M}$ . La prueba del teorema es simple pero extensa. Se puede ver en [4].

Para aplicar el teorema al problema de segmentación es necesario crear una red a partir del grafo  $\mathcal{G}$ . Se define entonces la red  $\mathcal{M}$ , con el mismo conjunto de vértices  $\mathcal{V}$  de  $\mathcal{G}$  donde

- Los vértices  $s$  y  $t$  son la fuente y el sumidero de  $\mathcal{M}$  respectivamente.
- Por cada arista  $n-link$   $\{p, q\}$  con peso  $w$  en  $\mathcal{G}$ , existen dos aristas dirigidas  $\{p, q\}$  y  $\{q, p\}$  en  $\mathcal{M}$  con capacidad  $w$ . Es decir,  $c(\{p, q\}) = w_{pq}$ .
- Por cada arista  $t-link$   $\{s, p\}$  con peso  $w$  en  $\mathcal{G}$ , existe una arista dirigida  $\{s, p\}$  en  $\mathcal{M}$  con capacidad  $w$ .
- Por cada arista  $t-link$   $\{p, t\}$  con peso  $w$  en  $\mathcal{G}$ , existe una arista dirigida  $\{p, t\}$  en  $\mathcal{M}$  con capacidad  $w$ .

El corte de costo mínimo en  $\mathcal{M}$  es igual a  $\hat{\mathcal{C}}$  en  $\mathcal{G}$ . Esta afirmación es válida porque la red  $\mathcal{M}$  tiene la misma estructura subyacente que  $\mathcal{G}$  salvo por la aparente duplicación de las aristas  $n-link$ . El hecho de que existan dos aristas dirigidas  $\{p, q\}$  y  $\{q, p\}$ , ambas con capacidad  $c$  implica que el flujo máximo<sup>3</sup> en cualquier dirección entre  $p$  y  $q$  no puede superar a  $c$ , por lo tanto la capacidad de flujo entre  $p$  y  $q$  en  $\mathcal{M}$  es igual al peso de la arista  $\{p, q\}$  en  $\mathcal{G}$ .

En la figura 1 se muestra una red que representa una imagen y un corte o frontera entre segmentos de la imagen.

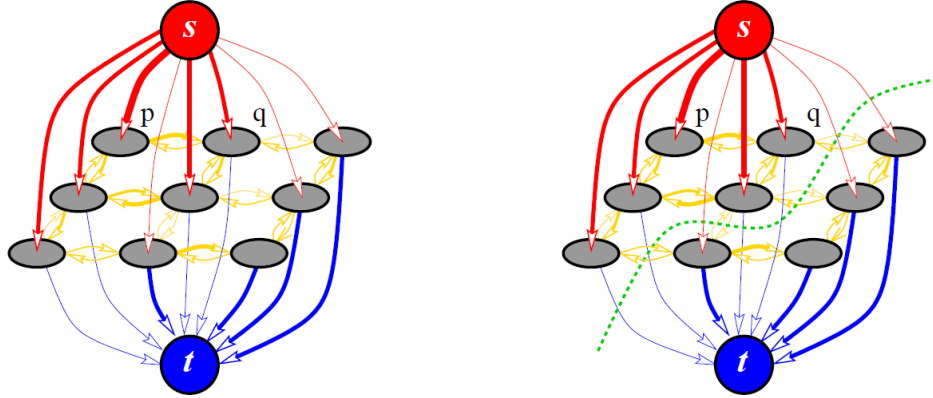


Figura 1: Ejemplo de un grafo pesado dirigido que incluye los vértices fuente ( $s$ ) y sumidero ( $t$ ). El costo de los arcos se refleja en sus espesores. En la figura de la derecha se puede ver un corte que segmenta la imagen.

## 6 Método de segmentación

A continuación se explica el método presentado en [3] para segmentar imágenes mediante la búsqueda del corte de costo mínimo en una red.

El método comienza con flujo nulo en la red. Iterativamente se buscan caminos entre los nodos  $s$  y  $t$  y se asigna a cada camino encontrado una cantidad de flujo igual a su capacidad máxima. Cuando todos los caminos posibles se encuentran saturados, se ha alcanzado el flujo máximo. El corte de costo mínimo queda formado por un subconjunto de los arcos saturados, es decir, con cantidad de flujo igual a su capacidad o capacidad residual nula.

<sup>3</sup> $f_{\{p,q\}} = |f_{p \rightarrow q} - f_{q \rightarrow p}|$ , donde  $f_{\{p,q\}}$  es el flujo total entre  $p$  y  $q$  y  $f_{p \rightarrow q}$  es el flujo a través de la arista dirigida que sale de  $p$  y entra a  $q$ .



## 6.1 Definiciones

**Árboles  $\mathcal{S}$  y  $\mathcal{T}$ .** Sobre la red  $\mathcal{M}$  se definen los árboles no superpuestos  $\mathcal{S}$  y  $\mathcal{T}$  con raíces en los nodos  $s$  y  $t$  respectivamente. Inicialmente los árboles estarán formado solo por sus raíces y evolucionarán durante la ejecución del método.

**Nodos libres.** Los nodos de la red  $\mathcal{M}$  que no pertenecen a los árboles  $\mathcal{S}$  y  $\mathcal{T}$  son llamados "nodos libres".

**Nodos activos y pasivos.** Los nodos que pertenecen a los árboles  $\mathcal{S}$  y  $\mathcal{T}$  pueden ser activos o pasivos. Los nodos marcados como pasivos son aquellos donde el método detectó que el árbol no se puede expandir. Por el contrario, los nodos activos son los que todavía pueden ser analizados con el objeto de expandir el árbol.

## 6.2 Etapas

Se utiliza una lista ordenada de nodos  $L$  para definir el orden en que se recorre la red. El método parte del siguiente estado inicial y ejecuta iterativamente las tres etapas que se explican a continuación.

- Los árboles  $\mathcal{S}$  y  $\mathcal{T}$  están formados solo por sus raíces  $s$  y  $t$ .
- Los nodos  $s$  y  $t$  están activos.

**Etapas de Crecimiento.** En esta etapa se realiza la expansión los árboles  $\mathcal{S}$  y  $\mathcal{T}$ . Se recorren en orden los nodos activos, primero del árbol  $\mathcal{S}$  y después los del árbol  $\mathcal{T}$ . Durante el recorrido se exploran en orden los nodos libres conectados a nodos activos. Dos nodos se consideran conectados cuando el arco es saliente del nodo activo, en el caso del árbol  $\mathcal{S}$ , y entrante al nodo activo para el caso del árbol  $\mathcal{T}$ . Los nodos explorados se añaden a los árboles como nodos activos. Cuando se han explorado todos los arcos de un nodo activo, este se convierte en pasivo.

La etapa termina cuando, durante la exploración, se encuentra un arco que conecta un nodo del árbol  $\mathcal{S}$  con un nodo del árbol  $\mathcal{T}$ . Se ha encontrado un camino entre los nodos terminales.

**Etapas de Camino.** Se determina la cantidad máxima de flujo  $f$  que puede pasar por el camino  $path$  encontrado en la etapa anterior.

$$f = \min_{e \in path} c_f(e)$$

Se reemplaza la red  $\mathcal{M}$  por la red residual  $\mathcal{M}_f$  resultante de la circulación del flujo  $f$  a través del camino  $path$  de  $\mathcal{M}$ .

Las aristas de menor capacidad en  $path$  quedarán saturadas, es decir, con  $c_f = 0$ . Una arco con capacidad nula se considera inexistente. Por lo tanto, algunos nodos podrían perder la conexión con el árbol al que pertenecen. Estos nodos se llaman huérfanos.

**Etapas de Adopción.** En esta etapa se intenta encontrar un nuevo padre para cada nodo huérfano  $h$ . La búsqueda se realiza entre los nodos del árbol al que pertenecía  $h$  según el orden de la lista  $L$ . Si no es posible encontrar un padre,  $h$  se declara libre. La etapa termina cuando no quedan nodos huérfanos.

**Condición de corte.** El método termina cuando los árboles no pueden seguir creciendo. En otras palabras, no se encuentran caminos no saturados entre  $s$  y  $t$ . Los árboles  $\mathcal{S}$  y  $\mathcal{T}$  quedan separados por arcos saturados<sup>4</sup>.

Si después del corte de las iteraciones existen nodos libres, es porque hay más de un corte de costo mínimo. En este caso, todos los nodos libres se asignan a alguno de los dos árboles.

Finalmente, la segmentación de la imagen se logra asignando los pixels correspondientes a los nodos del árbol  $\mathcal{S}$  a la figura y a los del árbol  $\mathcal{T}$  al fondo.

$$A_p = \begin{cases} \text{"obj"} & \text{si } p \in \mathcal{S} \\ \text{"bkg"} & \text{si } p \in \mathcal{T} \end{cases}$$

## 7 Ejemplo de cálculo

Se muestra un ejemplo de cálculo del flujo máximo y corte de costo mínimo sobre una red. Claramente la red utilizada en el ejemplo no tiene algunas de las características de las redes de segmentación de imágenes. Por ejemplo, no todos los nodos están conectados a la fuente y al sumidero. Tanto la topología de la red, como las capacidades de los arcos, se eligieron de forma que se presenten todas las situaciones importantes para comprender el método. Los ejemplos con redes de segmentación de imágenes son demasiado simples para cantidades pequeñas de pixeles.

En las figuras 2 - 25 se puede ver el estado del flujo en la red original (izquierda) y las redes residuales (derecha) durante cada etapa del proceso.

Se utiliza una lista para guardar la relación *padre/hijo* que también se muestra en las figuras. La lista  $L$  tiene el mismo orden que la lista de padres.

En las etiquetas de los arcos de las figuras del lado izquierdo se expresa flujo/capacidad. Durante las etapas de Camino el camino encontrado se marca con arcos de mayor espesor.

En la parte derecha de las figuras, los nodos asignados al árbol  $\mathcal{S}$  se colorean de amarillo y los del árbol  $\mathcal{T}$  de color verde. Los nodos sin color son nodos libres. Por otro lado, los nodos no-terminales con bordes discontinuos son nodos activos, mientras que los que tienen bordes continuos son pasivos. Durante la etapa de adopción, se indica con color azul (sobre el arco) la elección del nuevo padre.

---

<sup>4</sup>Los arcos que se saturan son los que tienen dirección  $\mathcal{S} \rightarrow \mathcal{T}$

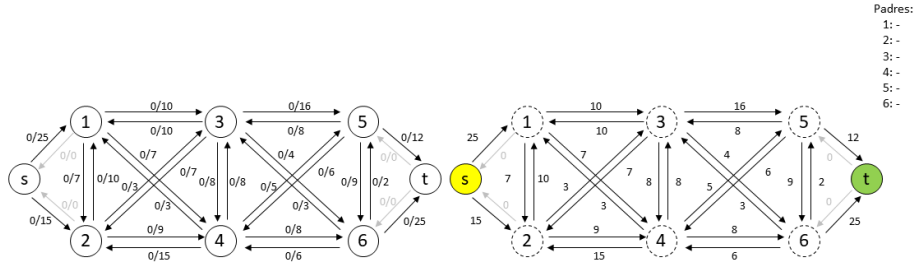


Figura 2: Situación inicial. No hay flujo en la red. Salvo las raíces, no hay nodos asignados a los árboles  $\mathcal{S}$  y  $\mathcal{T}$ .

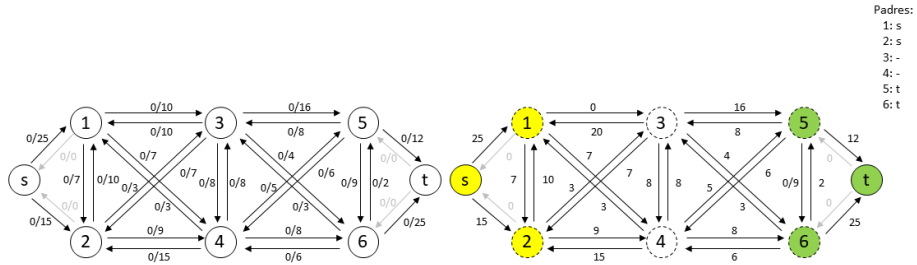


Figura 3: Etapa de crecimiento. Se asignan los nodos 1 y 2 a  $\mathcal{S}$  y los nodos 5 y 6 a  $\mathcal{T}$

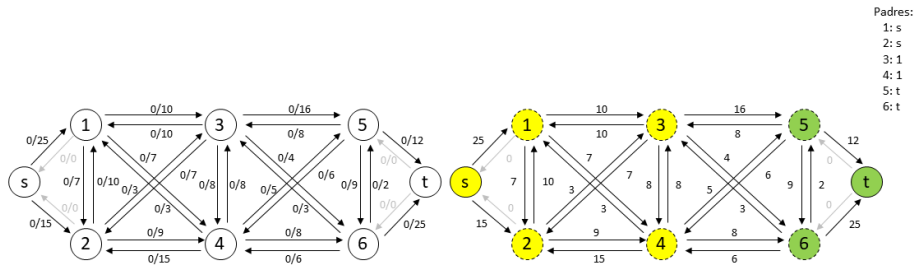


Figura 4: Sigue la etapa de crecimiento. Se asignan los nodos 3 y 4 a  $\mathcal{S}$ , ambos con padre: 1.

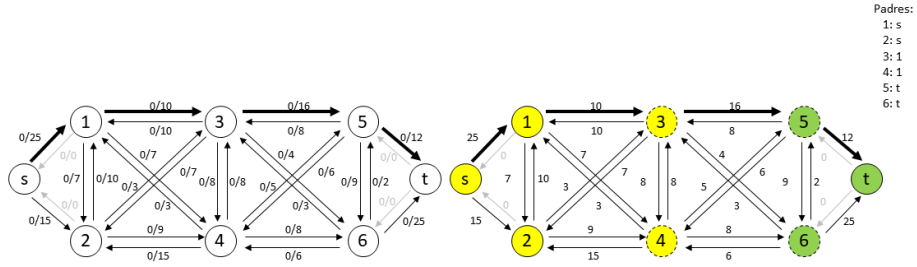


Figura 5: Sigue etapa de crecimiento Los nodos 1 y 2 se marcan como pasivos. Termina la etapa al encontrar el camino  $s-1-3-5-t$ .

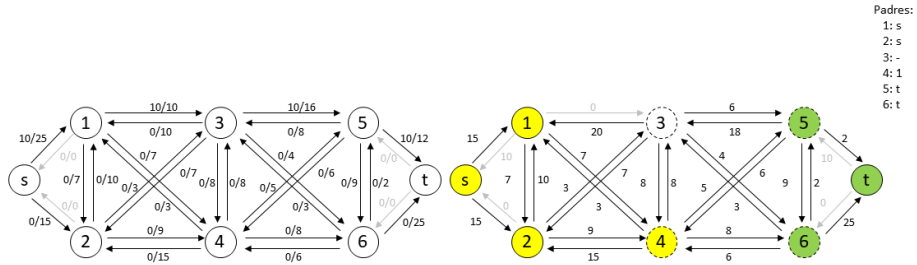


Figura 6: Etapa de camino. Se asigna el flujo máximo al camino según la capacidad mínima de sus arcos (izquierda). Se calcula la red residual para  $f = 10$  en el camino  $s-1-3-5-t$  (derecha). El nodo 3 queda huérfano porque la capacidad residual  $c_f(\{1, 3\}) = 0$ .

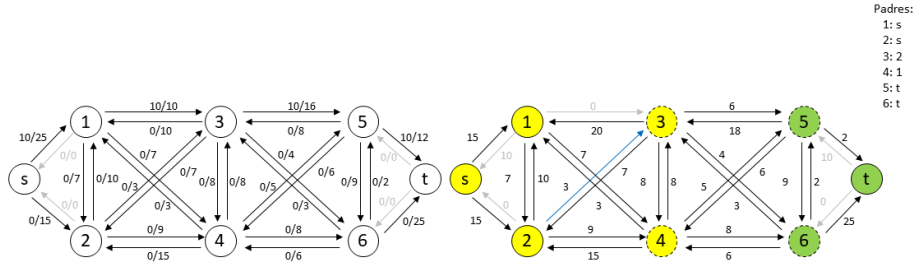


Figura 7: Etapa de adopción. El nodo 3 se asigna nuevamente al árbol  $\mathcal{S}$ , ahora con padre: 2.

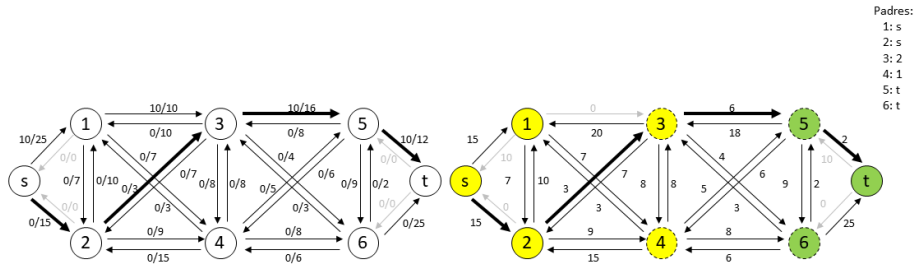


Figura 8: Etapa de crecimiento. Se encuentra el camino  $s-2-3-5-t$ .

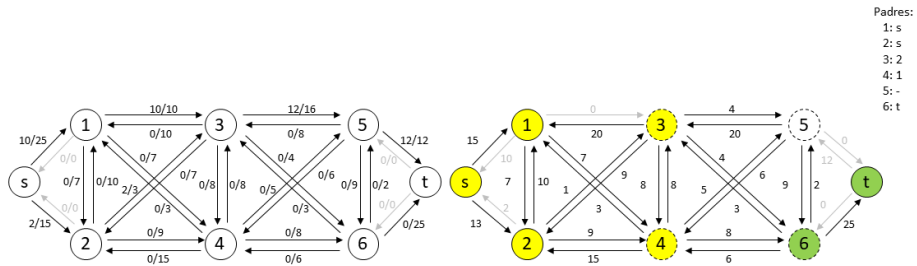


Figura 9: Etapa de camino. Se incrementa en 2 unidades el flujo del camino  $s-2-3-5-t$  y se calcula la red residual. El nodo 5 queda huérfano porque se satura el arco  $\{5, t\}$ .

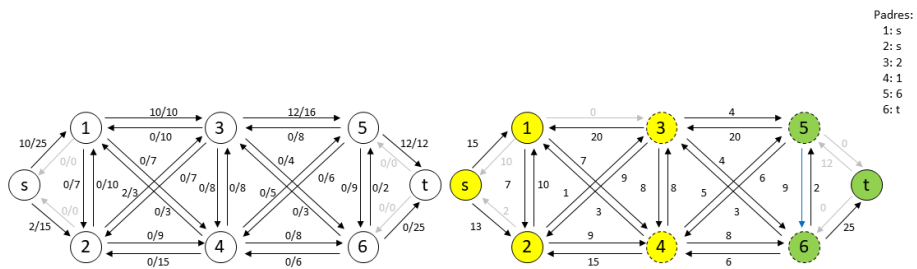


Figura 10: Etapa de adopción. El nodo 5 es nuevamente asignado al árbol  $\mathcal{T}$  con padre: 6.

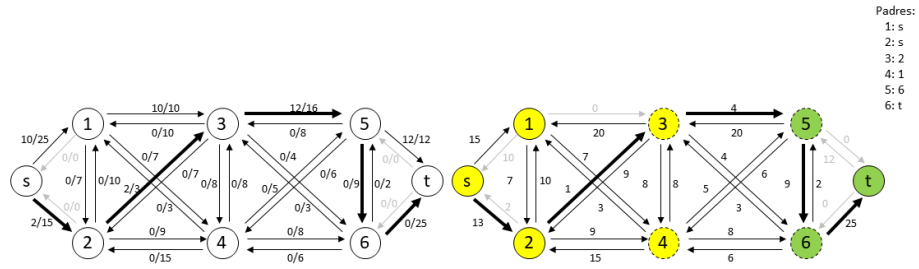


Figura 11: Etapa de crecimiento. Se encuentra el camino  $s-2-3-5-6-t$ .

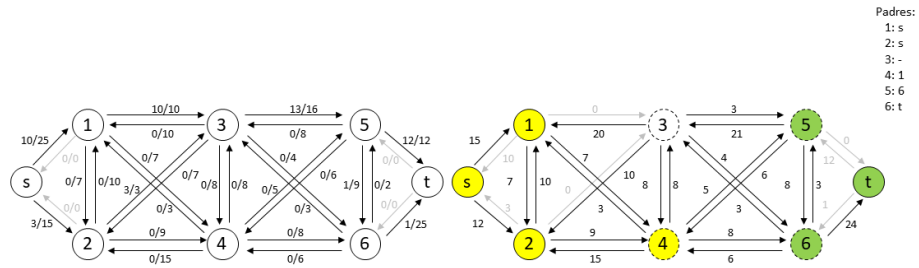


Figura 12: Etapa de camino. Se incrementa en 1 unidad el flujo del camino camino  $s-2-3-5-6-t$ . El nodo 3 queda huérfano.

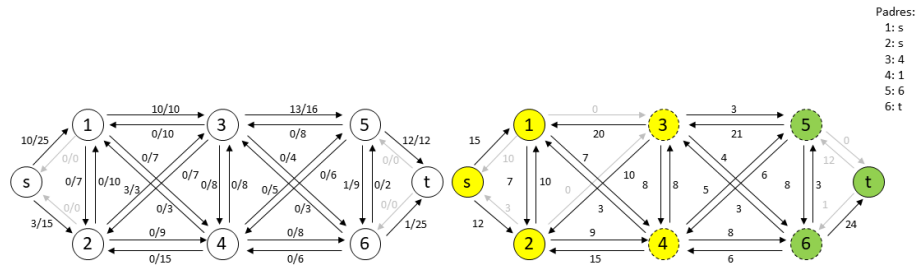


Figura 13: Etapa de adopción. Se asigna nuevamente el nodo 3 al árbol  $\mathcal{S}$  con padre: 4.

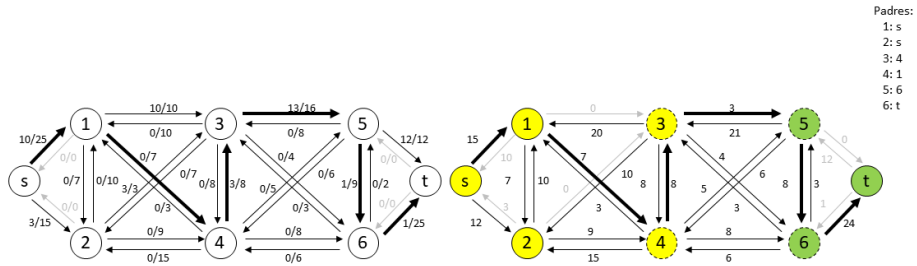


Figura 14: Etapa de crecimiento. Se encuentra el camino  $s-1-4-3-5-6-t$ .

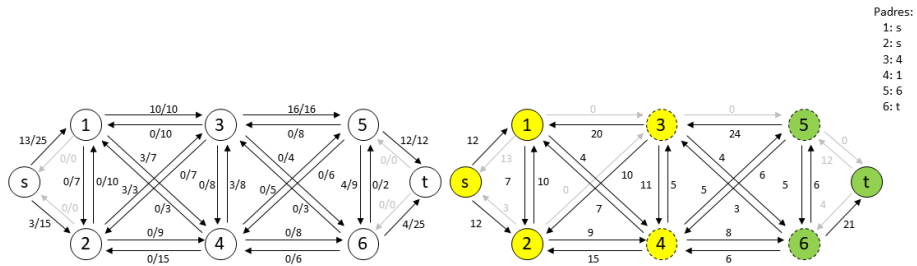


Figura 15: Etapa de camino. Se incrementa en 3 unidades el flujo del camino  $s-1-4-3-5-6-t$ . No hay nodos huérfanos.

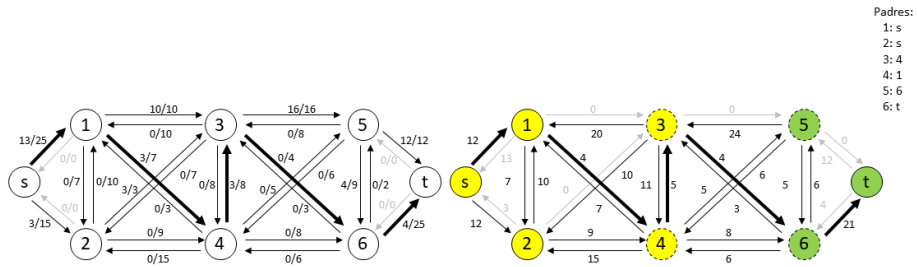


Figura 16: Etapa de crecimiento. Se encuentra el camino  $s-1-4-3-6-t$ .

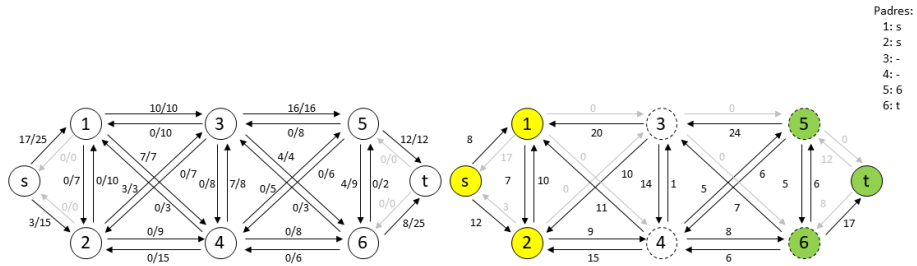


Figura 17: Etapa de camino. Se incrementa en 4 unidades el flujo del camino  $s-1-4-3-6-t$ . Los nodos 3 y 4 quedan huérfanos.

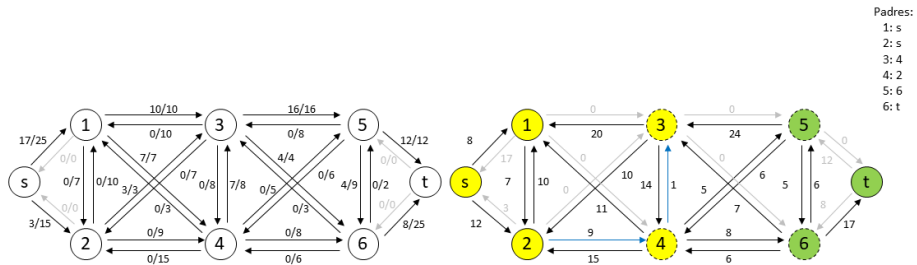


Figura 18: Etapa de adopción. Los nodos 3 y 4 se asignan nuevamente al árbol  $S$  con padre: 2 (nodo 4) y padre: 4 (nodo 3).

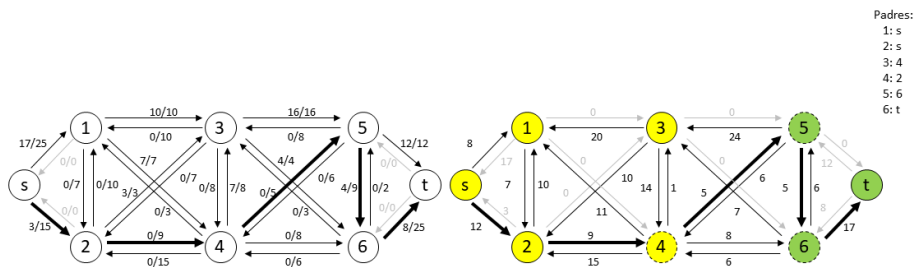


Figura 19: Etapa de crecimiento. El nodo 3 se marca como pasivo y se encuentra el camino  $s-2-4-5-6-t$ .



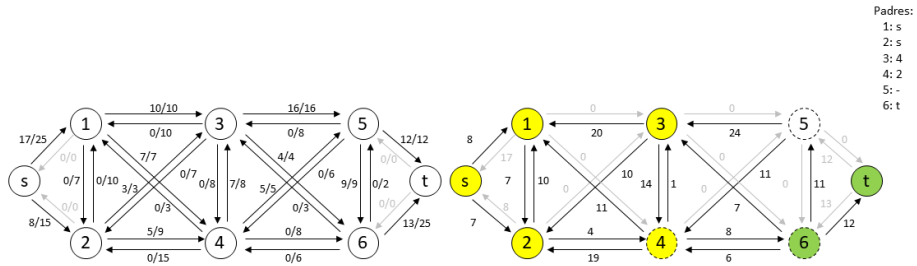


Figura 20: Etapa de camino. Se incrementa en 5 unidades el flujo del camino  $s$ -2-4-5-6- $t$ . El nodo 5 queda huérfano.

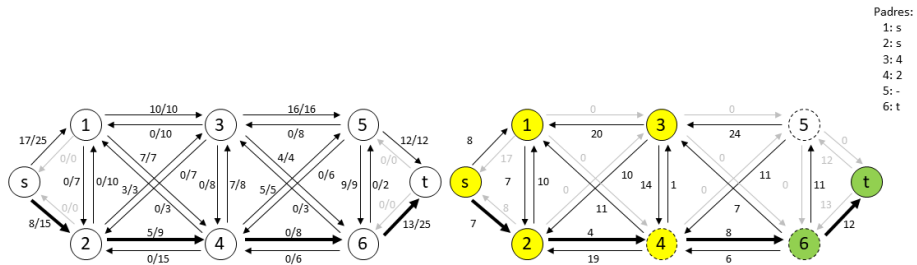


Figura 21: Etapa de adopción. No se puede asignar el nodo 5 al árbol  $\mathcal{T}$ . Etapa de crecimiento. Se encuentra el camino  $s$ -2-4-6- $t$ .

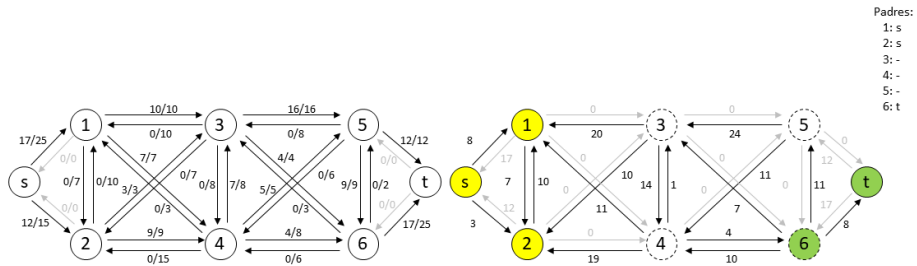


Figura 22: Etapa de camino. Se incrementa en 4 unidades el flujo del camino  $s$ -2-4-6- $t$ . Los nodos 3 y 4 quedan huérfanos.

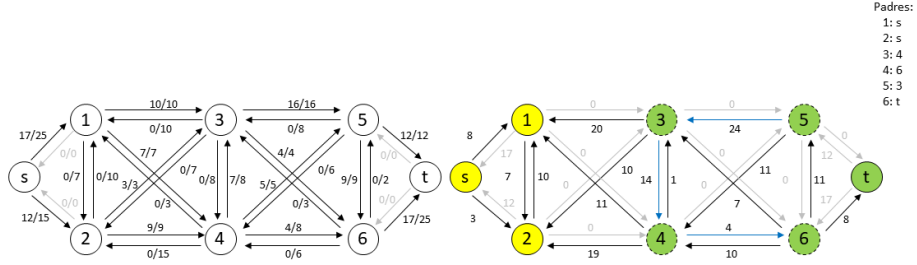


Figura 23: Etapa de adopción. Los nodos 3 y 4 no se pueden asignar al árbol  $\mathcal{S}$ . Etapa de crecimiento. Los nodos 4, 3 y 5 (en ese orden) se asignan al árbol  $\mathcal{T}$  con padre: 6, padre: 4 y padre: 3 respectivamente.

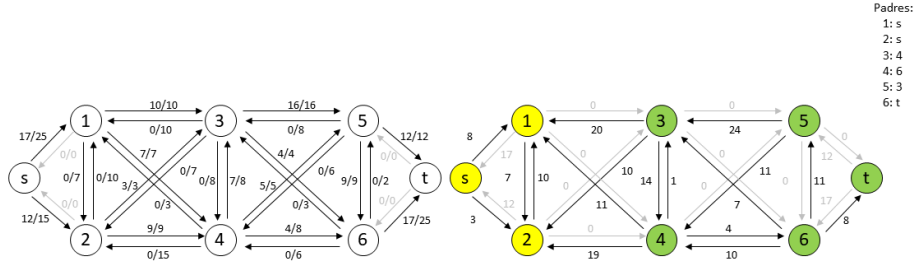


Figura 24: Sigue etapa de crecimiento. Los nodos 3, 4, 5 y 6 se marcan como pasivos y el método termina.

En la figura 25 se muestran los arcos saturados (en sentido  $\mathcal{S} \rightarrow \mathcal{T}$ ) marcados en rojo. Se puede ver que la suma de los flujos en los arcos saturados es 29, al igual que el flujo total que sale de  $s$  y entra en  $t$ .

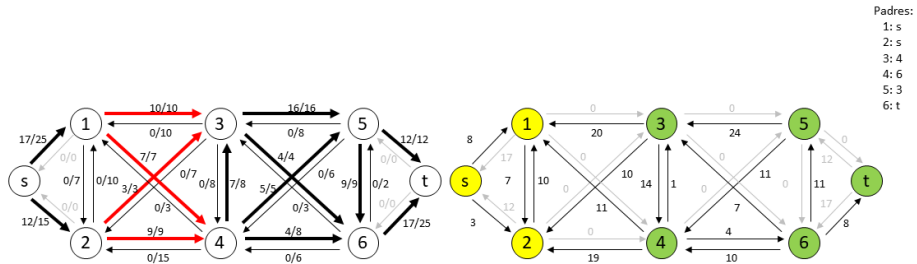


Figura 25: Arcos saturados al finalizar el proceso.

Finalmente, el corte de costo mínimo de la red es  $\hat{\mathcal{C}} = \{\{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}\}$  y el resultado de la segmentación  $A_1 = A_2 = \text{"obj"}$ ,  $A_3 = A_4 = A_5 = A_6 = \text{"bkg"}$ .

## Referencias

- [1] Yuri Y Boykov and M-P Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, volume 1, pages 105–112. IEEE, 2001.
- [2] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (9):1124–1137, 2004.
- [3] Susana M Rodríguez Lázaro. Segmentación de imágenes y teoría de grafos. Master’s thesis, Universidad Politécnica de Madrid, España, 2012.
- [4] Leslie R Foulds. *Graph theory applications*. Springer Science & Business Media, 1992.