Akış Ağları
(Flow Networks)

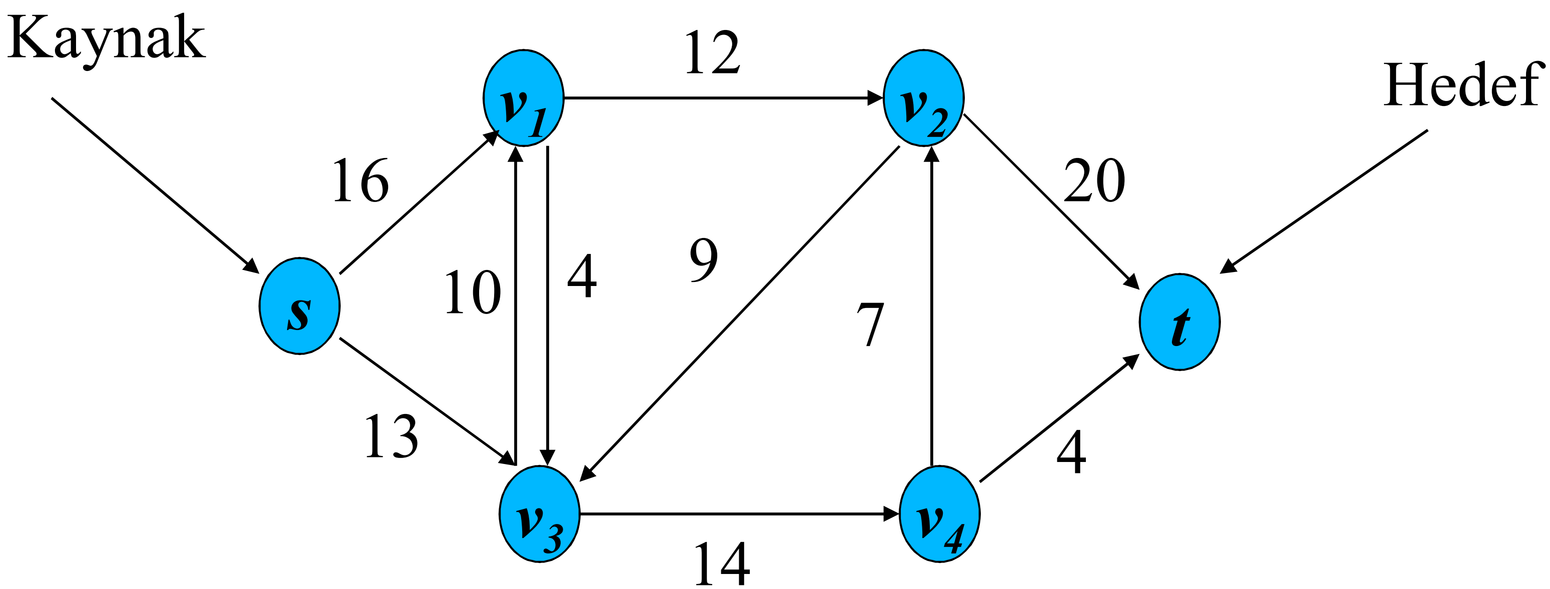
Akış Ağları

- Aşağıda örnekleri verilen problemleri modellemek için kullanılır:
  - Borulardan geçen sıvı/gazlar
  - Birleşme hatlarındaki parçalar
  - Elektrik ağlarındaki akımlar
  - İletişim ağlarındaki veriler
  - vs.

Akış Ağları & Maksimum Akış Problemi:

- Kenarlar: materyalin (sıvı/gaz/elektrik vs.) geçebileceği maksimum kapasiteler (ağırlık) ile belirtilir. (Örn. dakikada 50 litre)
- Düğümler: kenarların birleşmelerini gösterir. Kaynak ve hedef düğümü hariç hiçbir düğüm akışı eksiltmez veya arttırmaz.

Akış Ağları & Maksimum Akış Problemi



Maksimum Akış Problemi: Kaynaktan düğüme hiçbir kapasite sınırını ihlal etmeden gidebilecek maksimum akış miktarını bulma.

Akış Ağları

- Bir akış ağı $G = (V, E)$ yönlü bir çizgedir öyleki her bir kenar $(u,v) \in E$ negatif olmayan kapasitesi vardır, $c(u,v) \geq 0$
  - Eğer $(u,v) \notin E$, $c(u,v) = 0$ olduğunu kabul ederiz.
- Akış ağının iki tane özel düğümü vardır: kaynak (s ile gösterilir) ve hedef (t ile gösterilir).
  - Her düğüm kaynak ile hedef arasındaki bir yolda bulunur.
  - Çizge bağlıdır ve $|E| \geq |V| -1$.

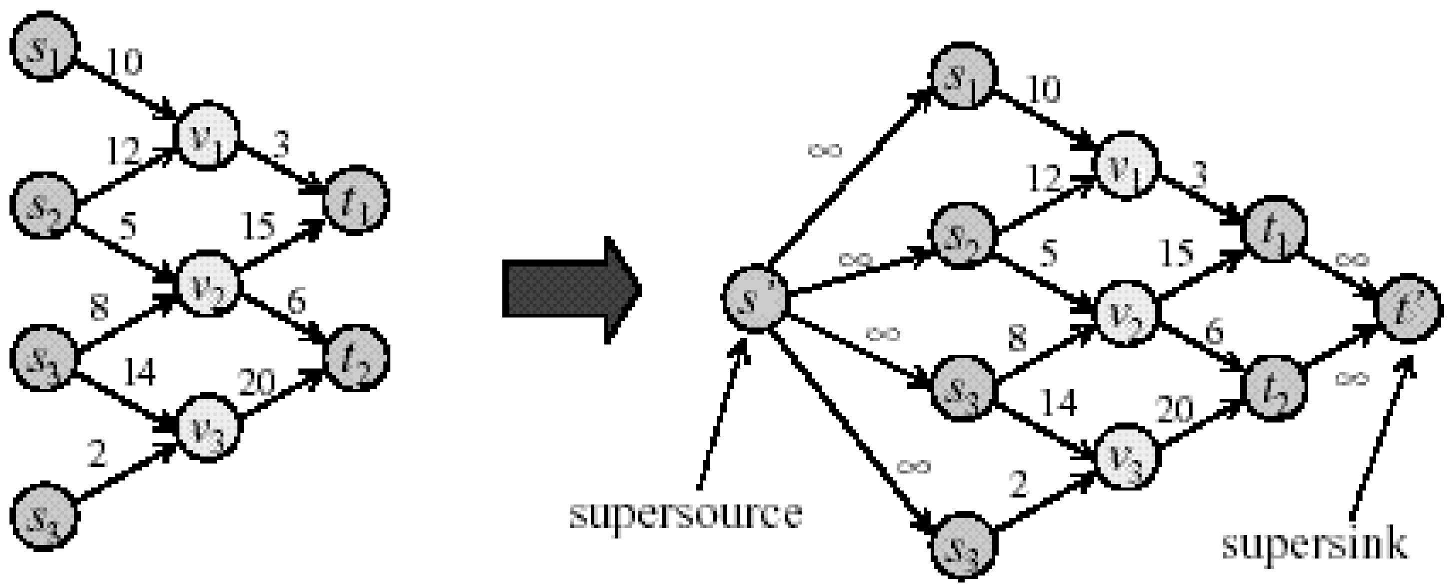
Kısıtlar

- $G = (V, E)$ akış ağı olsun ve c'de kapasite fonksiyonunu belirtsin. s ve t kaynak ve hedef düğümler olsun. G ‘deki akış reel değerli bir fonksiyondur $f: V \times V \rightarrow \mathbf{R}$. Bu fonksiyon aşağıdaki özellikleri sağlar:
- ***Kapasite Kısıtları***: Tüm $u,v \in V$ için, $f(u,v) \leq c(u,v)$
- ***Simetri***: Tüm $u,v \in V$ için, $f(u,v) = - f(v,u)$
- ***Akış Koruması***: Tüm $u \in V - \{s,t\}$'ler için,

$\sum_{v \in V} f(u,v) = 0$; bir düğümden çıkan toplam net akış 0'dır.

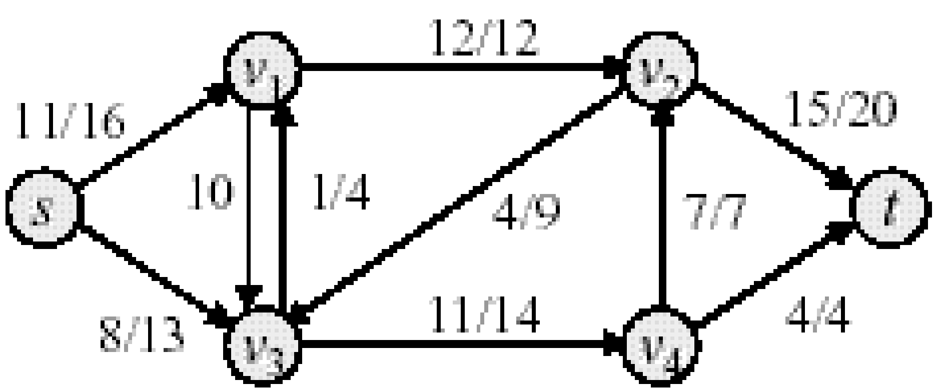
Birden fazla kaynak ve hedefi olan ağlar

- 1 kaynak ve 1 hedefi olan düğümlere kolayca çevrilebilir:



Net Akış

- net akışkapasite
  - $f(u,v)$: u düğümünden v düğümüne olan akış pozitif veya negatif olabilir.
- ***Simetri*:** Tüm $u,v \in V$ için, $f(u,v) = -f(v,u)$. Simetri özelliğine göre, akış koruma şöylede yazılabilir:
  - Bir düğümün toplam net akışı (kaynak ve hedef hariç) 0'dır.



Total net flow out of v_2:

$$\sum_{v \in V} f(v_2, v) = 15-7+4-12 = 0$$

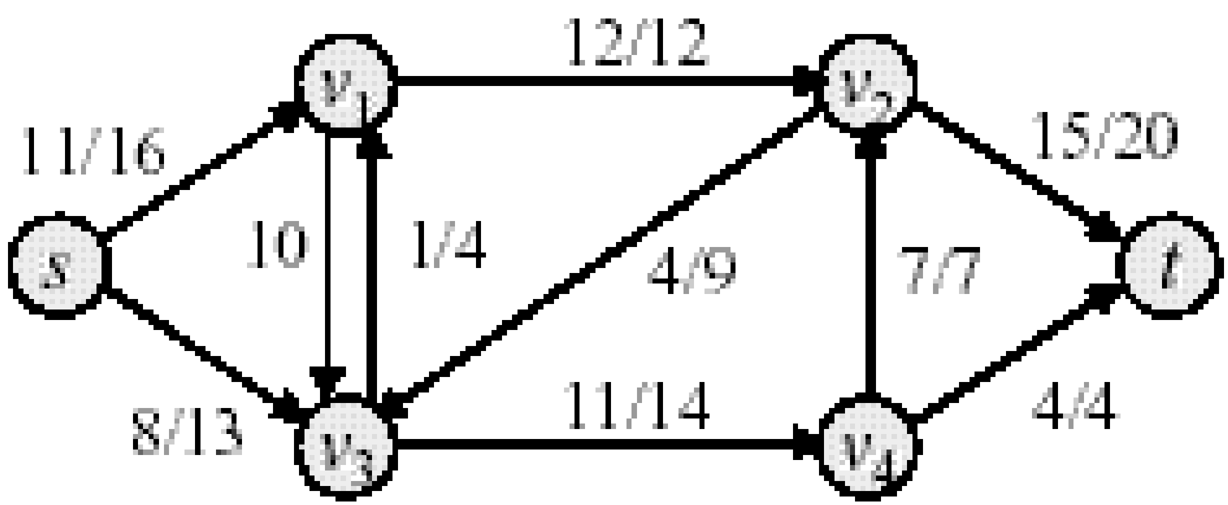
Total net flow into of v_2:

$$\sum_{v \in V} f(v, v_2) = -15+7-4+12 = 0$$

Net Akış

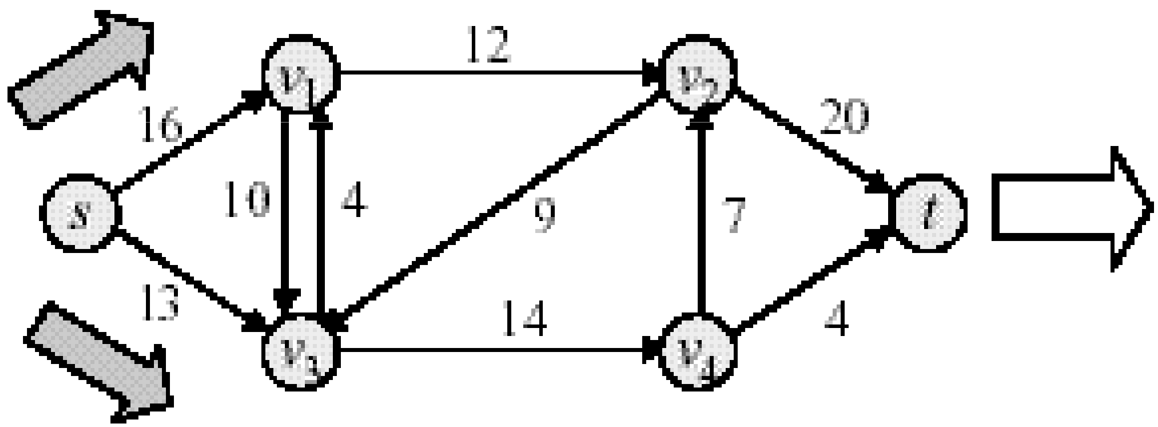
- net akış/capacity
  - Akış *f* 'nin miktarı, kaynak düğümünün toplam net akışı ile ölçülür.

$$|f| = \sum_{v \in V} f(s, v)$$



Maksimum Akış Problemi

- Kaynak s ve hedef t'si verilmiş akış ağı *G* için, *s*'den *t*'ye gönderilebilecek maksimum akış değerini bulma..



maximize $\Rightarrow |f| = \sum_{v \in V} f(s, v)$

Bazı noktalar:

- u ile v arasında bir kenar yoksa net akış sıfırdır.
  - Eğer $(u,v) \notin E$ ve $(v, u) \notin E$ ise $\boldsymbol{c(u,v) = c(v,u) = 0}$
- Kapasite sınırına göre:
  - $f(u,v) \leq \mathrm{c}(u,v), f(u,v) \leq 0$
  - $f(v,u) \leq \mathrm{c}(v,u), f(v,u) \leq 0$
- Simetri özelliğine göre:
  - $f(u,v) = - f(v,u) \Rightarrow f(u,v) = - f(v,u) = 0$
- Yani, sıfır olmayan v'den u'ya net akış şu manaya gelir:

  $(u, v) \in E$ veya $(v, u) \in E$ (veya her ikisi)

Positif net akış

- V düğümüne gelen pozitif net akış:

$$\sum_{\substack{u \in V \\ f(u,v) > 0}} f(u, v)$$

- Bir düğümden çıkan pozitif net akışda benzer olarak tanımlanır.
- Bir düğüme gelen pozitif net akış, o düğümden çıkan pozitif net akışa eşittir.

Maksimum Akış Problemi Çözümü

- **Ford-Fulkerson** iteratif metodu
  - $f(u,v) = 0$ ile başlar tüm $u,v \in V$ için.
- Her bir iterasyonda, akış değerini artan yol ile arttırır:
  - *Artan yol*: s ile t arasında daha fazla akış alabilecek bir yol
  - Bu yol üzerinden akış gönderilmektedir
- Her hangi bir artan yol bulunamayana kadar tekrarla.
- ***max-flow min-cut teoremi algoritma sonlandığında bu metodun maksimum akışı bulduğunu göstermektedir.***

Ford-Fulkerson Metodu

Ford-Fulkerson-Method(*G*,*s*,*t*)

Initialize flow f to 0

While there exists an augmenting path p

do augment flow along p

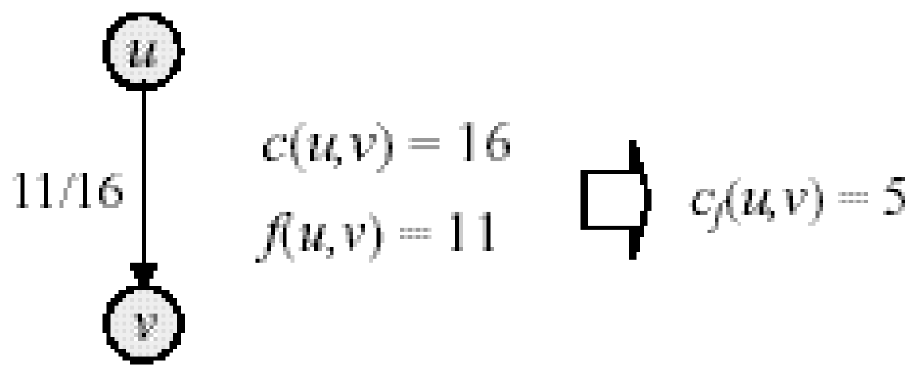
return f

3 tane önemli kavrama dayanır:

- Artan Ağlar
- Artan Yollar
- Kesimler

Artan Ağlar

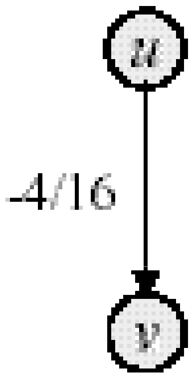
- Bir akış ağı ve akış için, **Artan Ağ** daha fazla akış alabilecek kenarları içerir.
- G=(V, E) akış ağı olsun, kaynak s ve hedef t düğümü ile. f, G'deki akış olsun. Bir düğüm çiftini göz önüne alın: $u,v \in V$
- Kapasite $c(u,v)$'yi aşmadan u'dan v'ye gönderilebilecek ek net akış miktari, (u,v)'nin artan kapasitesidir. $c_f(u,v) = c(u,v) - f(u,v)$



Artan Kapasite

- Net akış $f(u,v)$ negatif olduğunda, artan kapasite

$c_f(u,v)$, $c(u,v)$'den büyüktür.



$c(u,v) = 16$

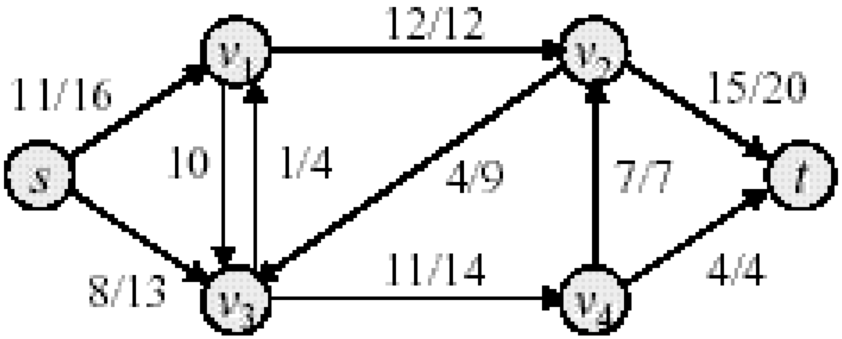
$f(u,v) = -4$ $\Rightarrow$ $c_f(u,v) = c(u,v) - f(u,v) = 16 + 4 = 20$

Artan Ağlar

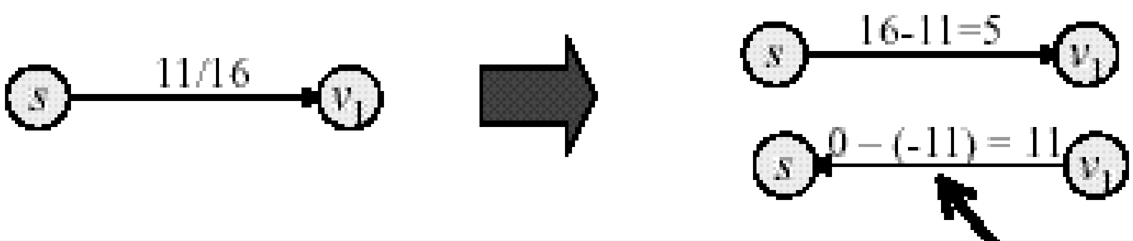
- Verilen bir akış ağı $G = (V, E)$ ve akış f için, G'nin artan ağı $G_f = (V, E_f)$, burada

$$E_f = \{(u,v) \in V \times V : c_f(u,v) > 0\}$$

Artan Ağ: Örnek

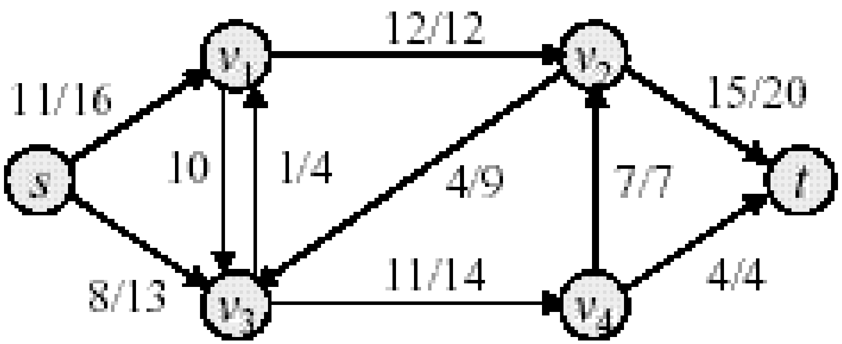


s ile v_1 arasındaki artan kenarlar:

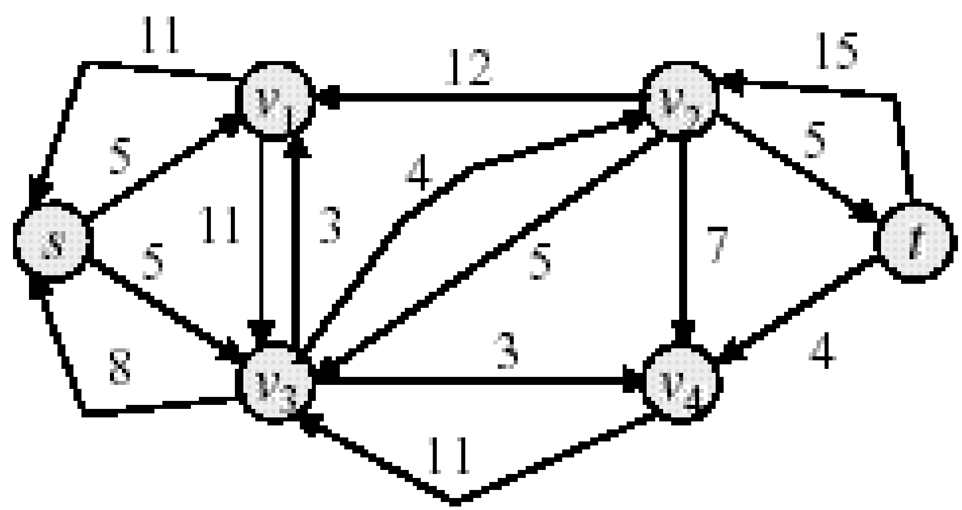


(u,v) Ef ‘de bir kenar olabilir ve bu kenar E’de olmayabilir
(kapasite bu örnekte sıfırdır)

Artan Ağ: Örnek

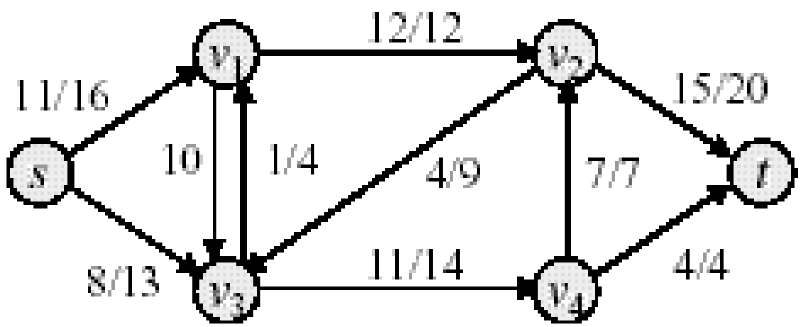


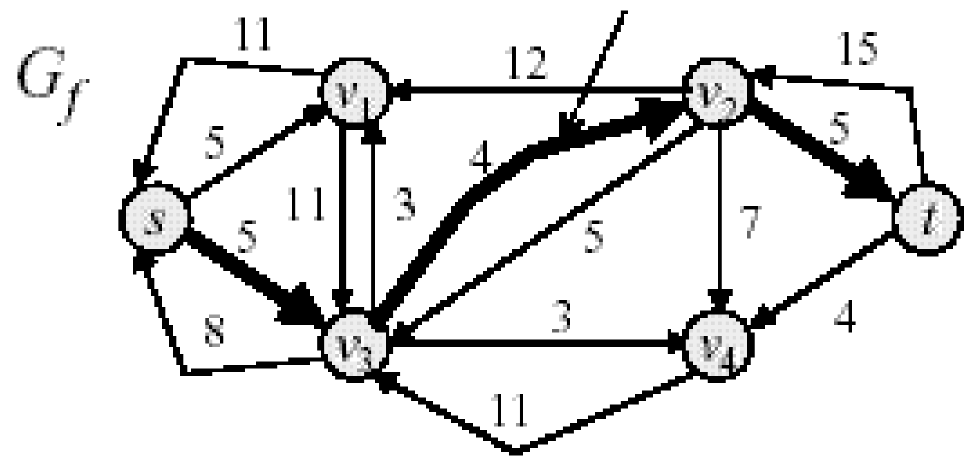
Artan Ağ G_f



Artan Yollar

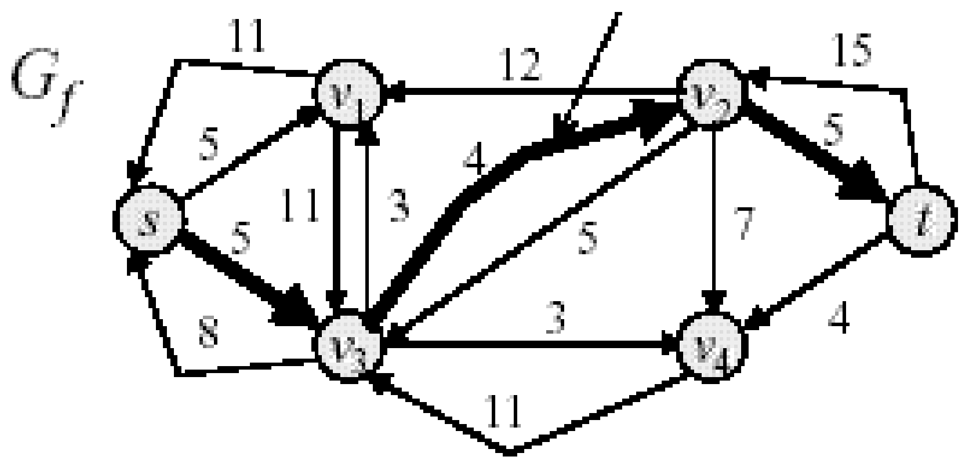
- Verilen bir akış ağı $G = (V, E)$ ve akış f için, artan yol p, artan ağ G_f içinde s'den t'ye giden basit bir yoldur.
- Artan Ağ tanıma göre, artan yol üzerindeki her bir kenar (u,v) u'dan v'ye kapasite sınırını aşmadan ek pozitif net akış alabilir.





Artan yolun artan kapasitesi

$$c_f(p) = \min\{c_f(u,v) : (u,v) \text{ kenarı } p \text{ üzerindedir}\}$$



Bu yolun artan kapasitesi

$c_f(v, v2) = 4.$

Ford-Fulkerson Metodu

- **Ford-Fulkerson-Method(*G,s,t*)**

  Initialize flow *f* to 0

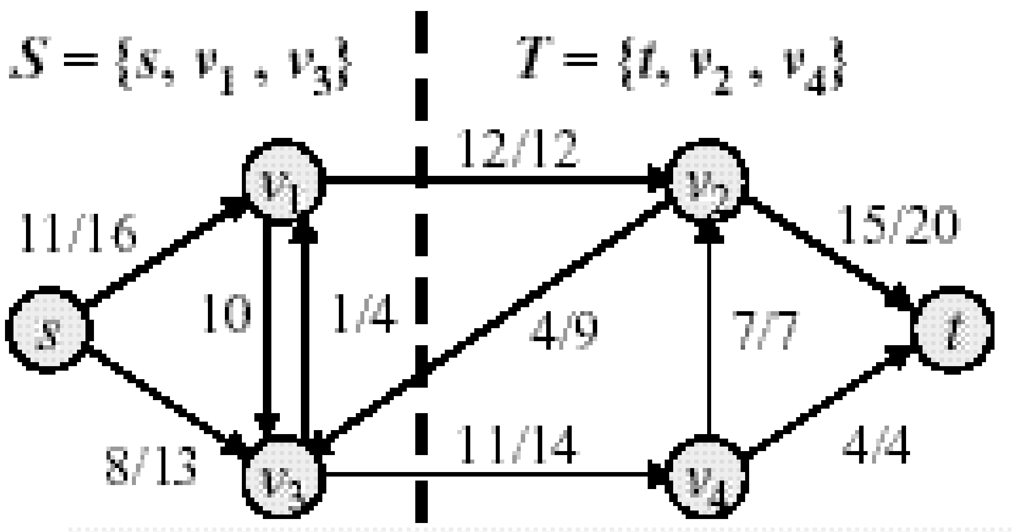
  while there exists an augmenting path *p*

  do augment flow along *p*

  return *f*

Akış Ağları Kesimleri

- Bir akış ağının $G = (V,E)$ kesimi (S,T) V ‘yi iki tane kümeye böler, S ve $T=V-S$ öyleki $s \in S$ ve $t \in T$.
- Eğer f akış ise, kesim (S,T)’den net akış $f(S,T)$ ile tanımlanır. Kesim (S,T) ‘nin kapasitesi $c(S,T)$’dir.



Kesimin net akışı:
$f(S,T) = f(v1,v2) + f(v3,v2) + f(v3,v4)$
$= 12 + (-4) + 11 = 19$

Kesimin kapasitesi:
$c(S,T) = c(v1,v2) + c(v3,v2) + c(v3,v4)$
$= 12 + 0 + 14 = 26$

Max-Flow Min-Cut Teoremi

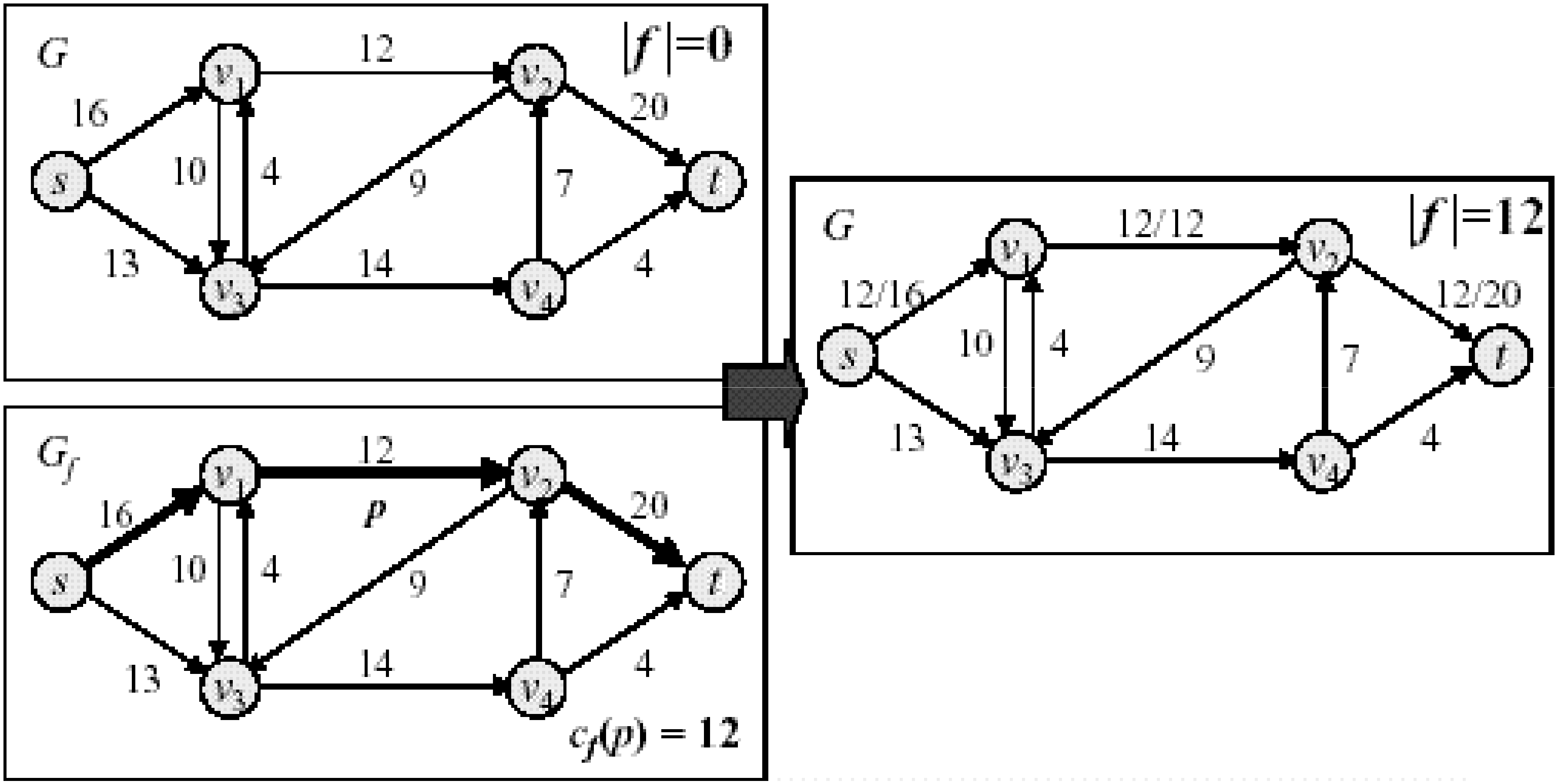
- f akış ağı $G = (V,E)$ içinde bir akış ise (kaynak s ve hedef t), aşağıdaki durumlar birbirlerine denktir:

(1) $\boldsymbol{f}$, G'deki maksimum akıştır.

(2) Artan Ağ G_f artan yol içermez.

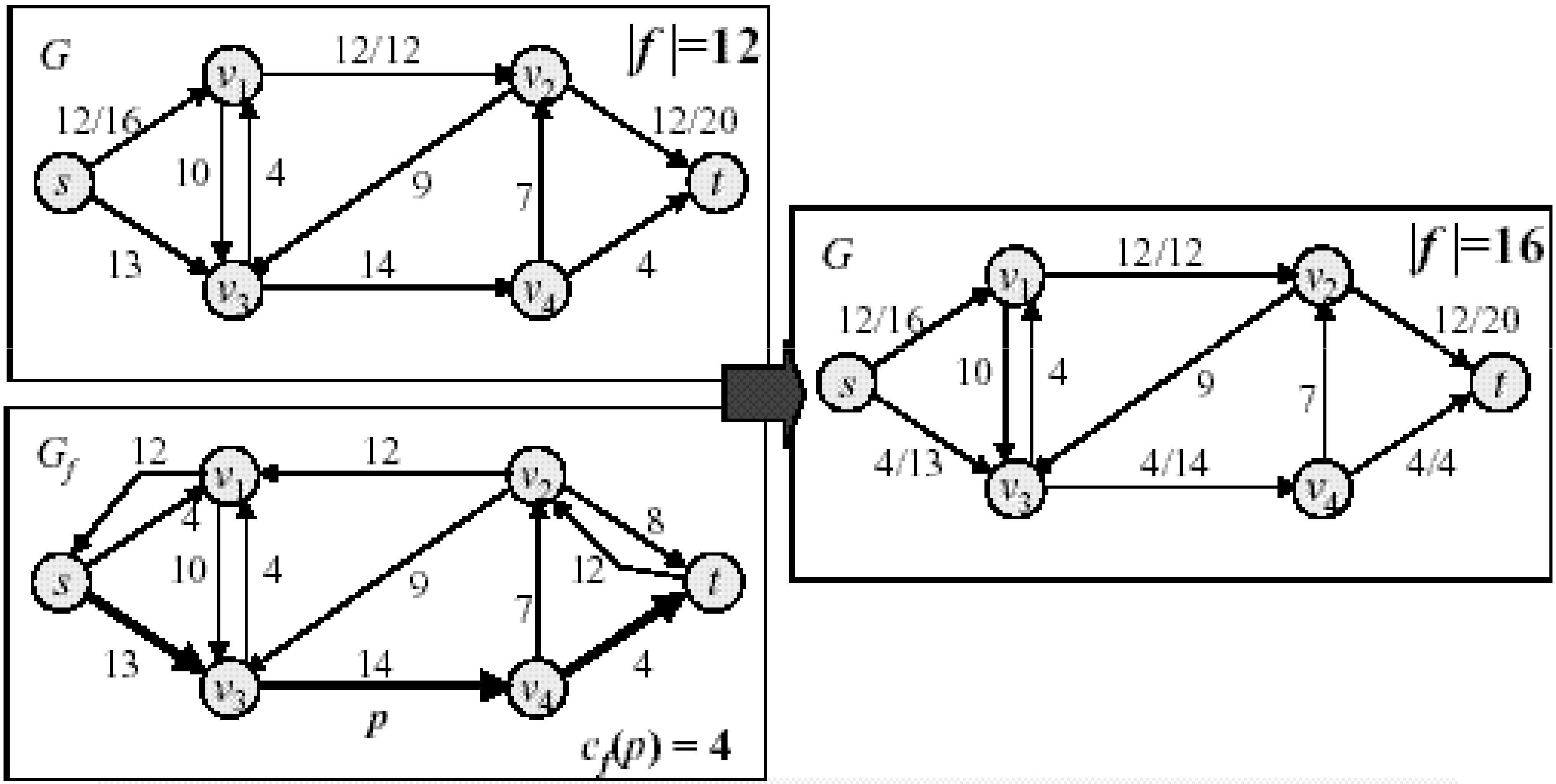
(3) $|f| = c(S,T)$ bir kesim (S,T) için. (Kesim kapasitesinin net akış miktarına eşit olduğu bir kesim vardır.)

Örnek



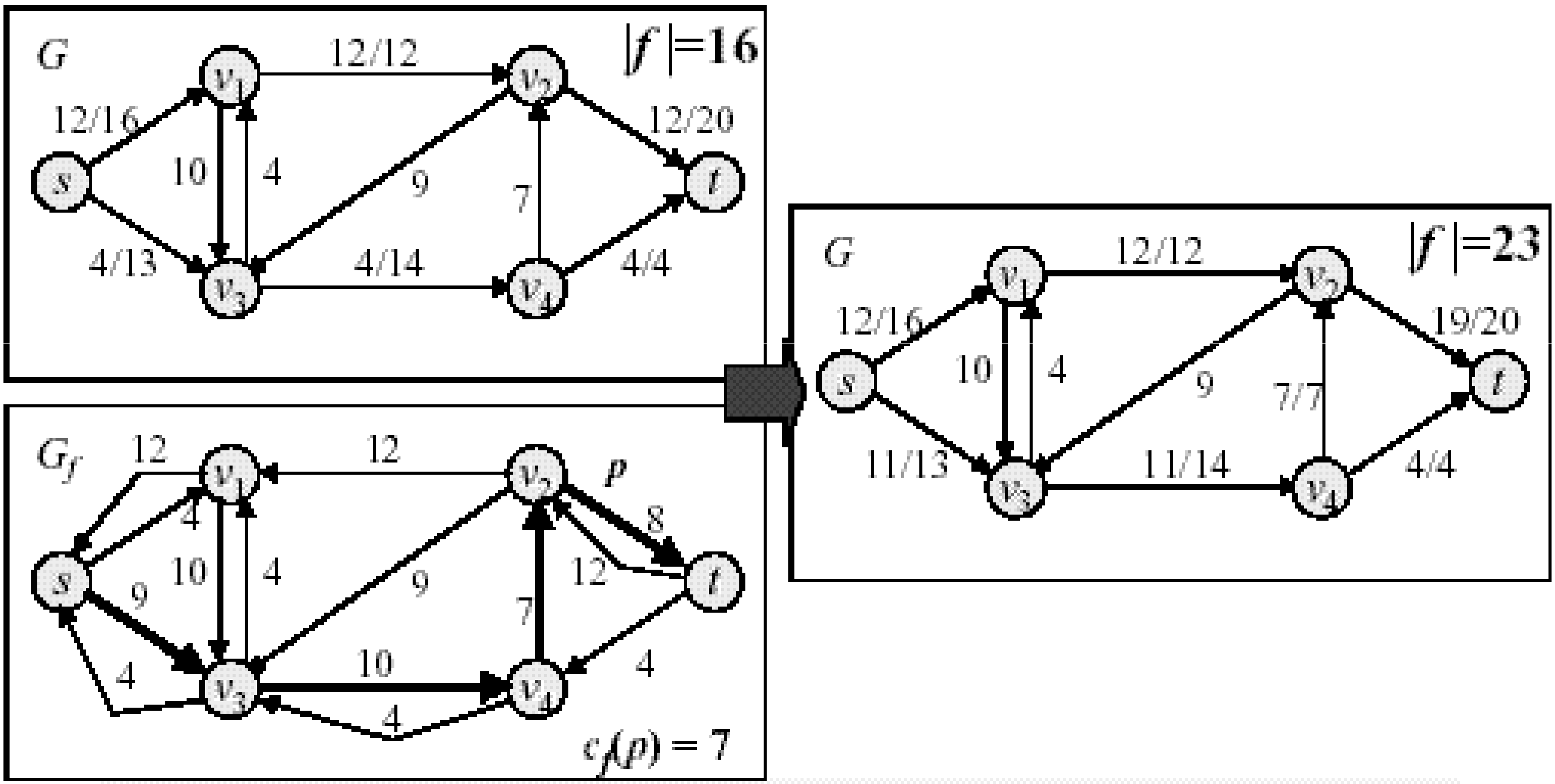
G
|f|=0
s
v1
v2
v3
v4
t
16
12
20
10
4
9
7
13
14
4
Gf
p
c_f(p) = 12
G
|f|=12
12/16
12/12
12/20

Örnek

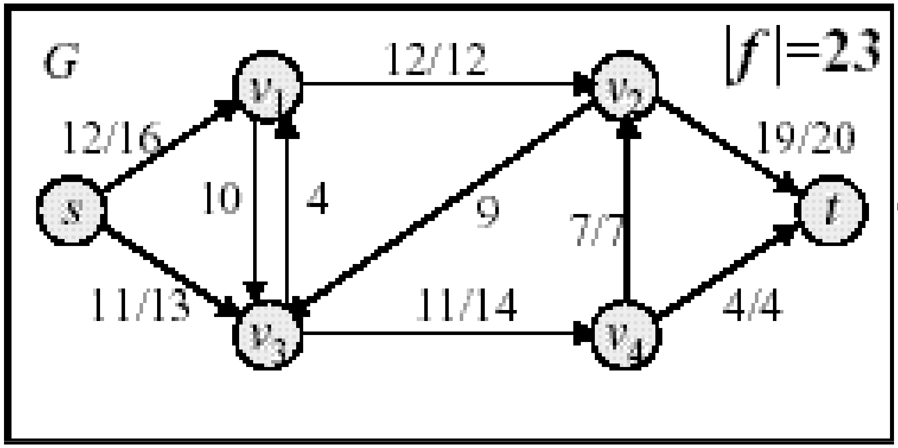


G
|f|=12
12/12
12/16
12/20
10
4
9
7
13
14
4
G
|f|=16
12/12
12/16
12/20
10
4
9
7
4/13
4/14
4/4
G_f
12
12
4
10
4
9
8
12
7
13
14
4
p
$c_f(p) = 4$

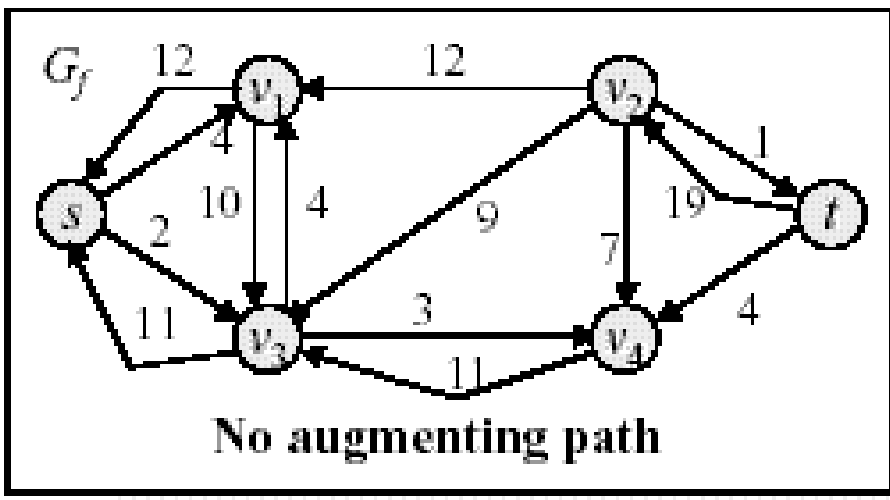
Örnek

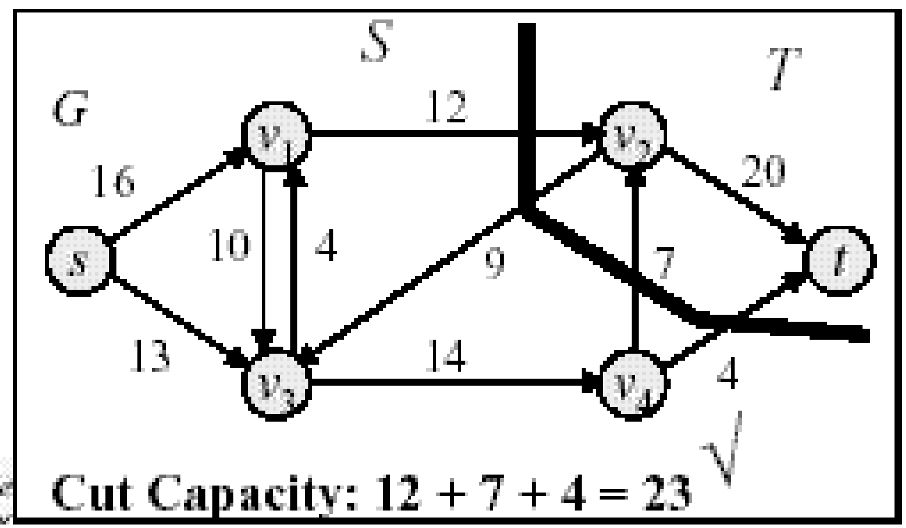

G
|f|=16
12/12
12/16
12/20
10
4
9
7
4/13
4/14
4/4
G_f
12
12
p
8
4
10
4
9
12
9
7
4
10
4
4
$c_f(p) = 7$
G
|f|=23
12/12
12/16
19/20
10
4
9
7/7
11/13
11/14
4/4

Örnek



max flow was found: |f|=23

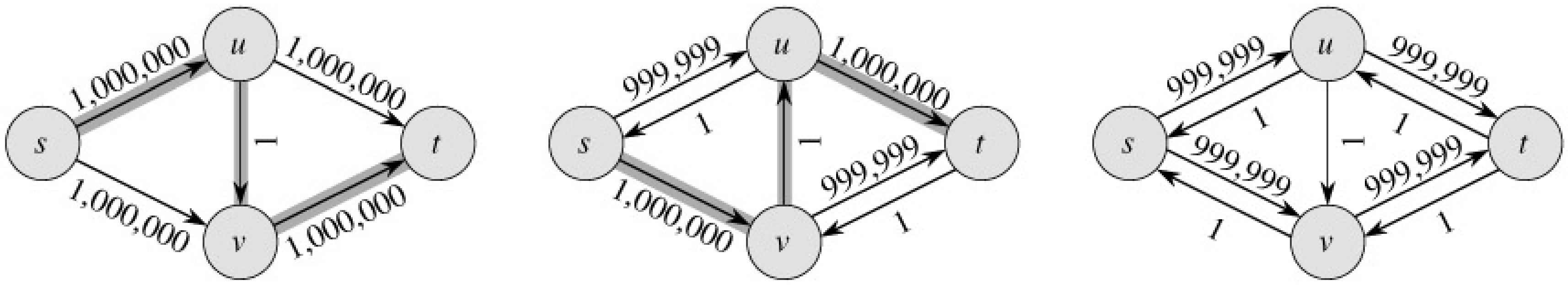




Ford-Fulkerson Algoritması: Çalışma Zamanı

- Ford-Fulkerson algoritmasının çalışma zamanı, artan yolun nasıl seçildiğine bağlıdır. En kötü durumda $O(E|f^*|)$ zaman alır, burada f^* maksimum akışı gösterir.
- Eğer artan yol genişlik öncelikli arama ile seçilirse (Edmonds-Karp), algoritmanın yapacağı iterasyon sayısı en fazla $O(VE)$'dir.
- Her bir iterasyon $O(E)$ zaman alır.
- Toplam çalışma zamanı: $O(VE^2)$

En kötü seneryo



(a) (b) (c)