

# ***Duomenų struktūros ir algoritmai***

11 paskaita

2024-04-23

# ***Svarbi informacija***

## TOLESNI PLANAI:

- Balandžio 30 d. paskaitos skirtos kursui užbaigti (papildomos temos, kurios neįeina į egzaminą, tačiau liečia praktinius programavimo uždavinius).
- Gegužės 7 d. paskaitos skirtos plačiau susipažinti kompiuteriniu kreivių ir paviršių modeliavimu.
- Gegužės 14 d. paskaitos dedikuotos Jūsų pasirinktų temų pristatymams.
- Gegužės 21 d. paskaita skirta sukurtų 3D modelių pristatymams ir gražiausių 3D modelių išrinkimui. Likęs laikas skiriamas atsiskaitymams.
- Gegužės 28 d. paskaita – konsultacija prieš egzaminą.
- Atsiskaityti praktines užduotis galima iki gegužės galo.
- **Egzaminas birželio 20 d. 12 val. ??? (Nuotoliniu būdu).**
- Egzamino trukmė – 2 val.

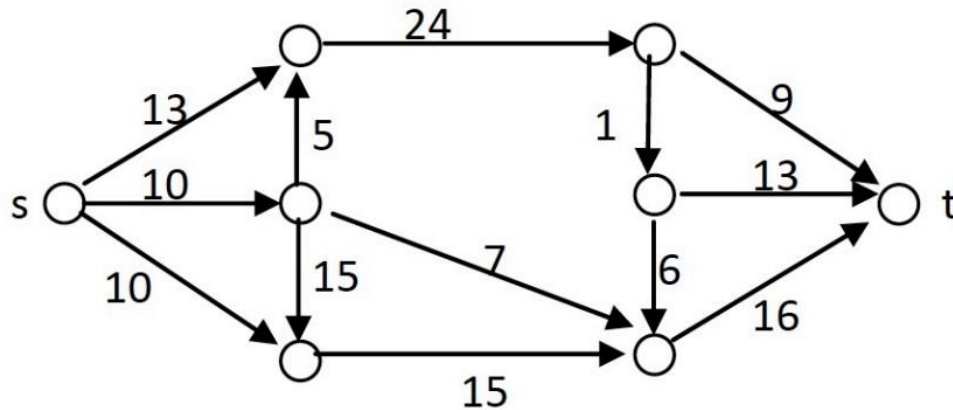
# ***11 paskaitos tikslas***

Susipažinti su maksimalaus srauto paieškos algoritmais:

- Fordo ir Fulkersono metodas,
- Edmondso–Karpio algoritmas,
- Priešsraučio stūmimo algoritmas,
- Maksimalaus srauto paieškos taikymai:
  - Maksimalus dvidalis suporavimas,
  - Ištrūkimo uždavinys (*angl. escape problem*).

# Srautai tinkluose

- Kas yra **srautas** ir kas yra **tinklas**?
- Tarkime, turime jungųjį digrafą  $G = (V, E)$ , kuriame išskirtos šaltinio  $s \in V$  ir tikslo viršūnės  $t \in V$ :



- Jei tokiame digrafe apibrėžta talpos funkcija  $c : E \rightarrow \mathbb{R}^+$  arba jos apibrėžimo sritis išplėsta  $V \times V$ , kai  $c(uv) = 0$ , jei  $uv \notin E$ , tai toks digrafas dar vadinamas *tinklu*.

# Srautai tinkluose

- **Apibrėžimas.** Srautu tinkle  $G = (V, E)$  vadinama funkcija  $f : V \times V \rightarrow \mathbb{R}$ , jei tenkinamos tokios sąlygos:

1)  $f(uv) \leq c(uv)$ ,

2)  $f(uv) = -f(vu)$ ,

- 3) kiekvienai  $u \in V \setminus \{s, t\}$  teisinga sąlyga

$$\sum_{v \in V} f(uv) = 0.$$

- 3 sąlyga dar vadinama *Kirchhofo* aksioma.
- Srauto didumu vadinamas dydis

$$|f| = \sum_{v \in V} f(sv) = \sum_{v \in V} f(vt).$$

# Maksimalaus srauto uždavinys

- Tinkle  $G = (V, E)$ , kuriame išskirtos šaltinio  $s$  ir tikslo viršūnės  $t$ , rasti srautą  $f$ , kurio dydis būtų maksimalus.

## Apibendrintas maksimalaus srauto uždavinys

- Tinkle  $G = (V, E)$ , kuriame išskirtos šaltinio  $s_1, \dots, s_k$  ir tikslo viršūnės  $t_1, \dots, t_r$ , rasti srautą  $f$ , kurio dydis būtų maksimalus.
- Tokiu atveju, įvedus papildomą šaltinį  $s$  ir tikslo viršūnę  $t$  ir briaunas  $ss_i$ ,  $t_jt$  bei apibrėžus  $c(ss_i) = \infty$  ir  $c(t_jt) = \infty$ , užtenka spęsti pirmąjį uždavinį.

# Fordo ir Fulkersono metodas (1956)

- Šio metodo idėja – nuolat ieškoti galimo srauto nuo šaltinio iki tikslo viršūnės, kiek galima didinti srautą.
- Pradedant  $f = 0$  ir radus taką  $p : s \Rightarrow t$ , tiesioginėse tako briaunose srautas didinamas, o priešingų krypčių (netiesioginėse) briaunose srautas mažinamas.
- Tokį srautą take  $p$  galima padidinti *taip*  $c(p)$ :  
$$c(p) := \min\{c(uv) : uv \in p\}.$$
- Kai  $c(p) > 0$ , takas  $p$  vadinamas auginančiu srautą.

Diskusijai:

- Kuo skiriasi algoritmas nuo metodo?

# Fordo ir Fulkersono metodas (1956)

- $G$  – jungusis svorinis digrafas (tinklas),  $s$  – šaltinio viršūnė,  $t$  – tikslo viršūnė.
- Algoritme daug kartų iškviečiama procedūra  $F-F(G, s, t)$ .  
 **$F-F(G, s, t)$ :**
  1.  $f \leftarrow 0$
  2. **while** egzistuoja srautą auginantis takas  $p$
  3.     **do** „augink  $f$  take  $p$ “
  4. **return**  $f$
- Realizuojant šią idėją paranku naudotis likutiniais tinklais  $G_f = (V, E_f)$ ,  $E_f = \{uv \in V \times V : c_f(uv) > 0\}$ , kur  $c_f(uv) = c(uv) - f(uv)$  yra likutinė talpa.



# Maksimalaus srauto ir minimalaus pjūvio savybės

- Jei  $f$  yra srautas tinkle  $G = (V, E)$  su šaltiniu  $s$  ir tikslo viršūne  $t$ , tai šie 3 teiginiai yra ekvivalentūs:
  - 1)  $f$  yra maksimalus srautas,
  - 2) likutinis tinklas  $G_f$  neturi srautą auginančių takų,
  - 3) kažkokiam pjūviui  $(S, T)$  turime  $|f| = c(S, T)$ ,čia  $V = S \cup T$  su savybe  $s \in S$  ir  $t \in T$ ,  $c(S, T)$  – pjūvio talpa, o  $f(S, T)$  – srautas, tekantis iš  $S$  į  $T$ .
- Taip pat teisinga savybė, kad bet kokiam pjūviui  $V = S \cup T$  turime  $|f| = f(S, T)$ .

# Edmondso–Karpo algoritmas

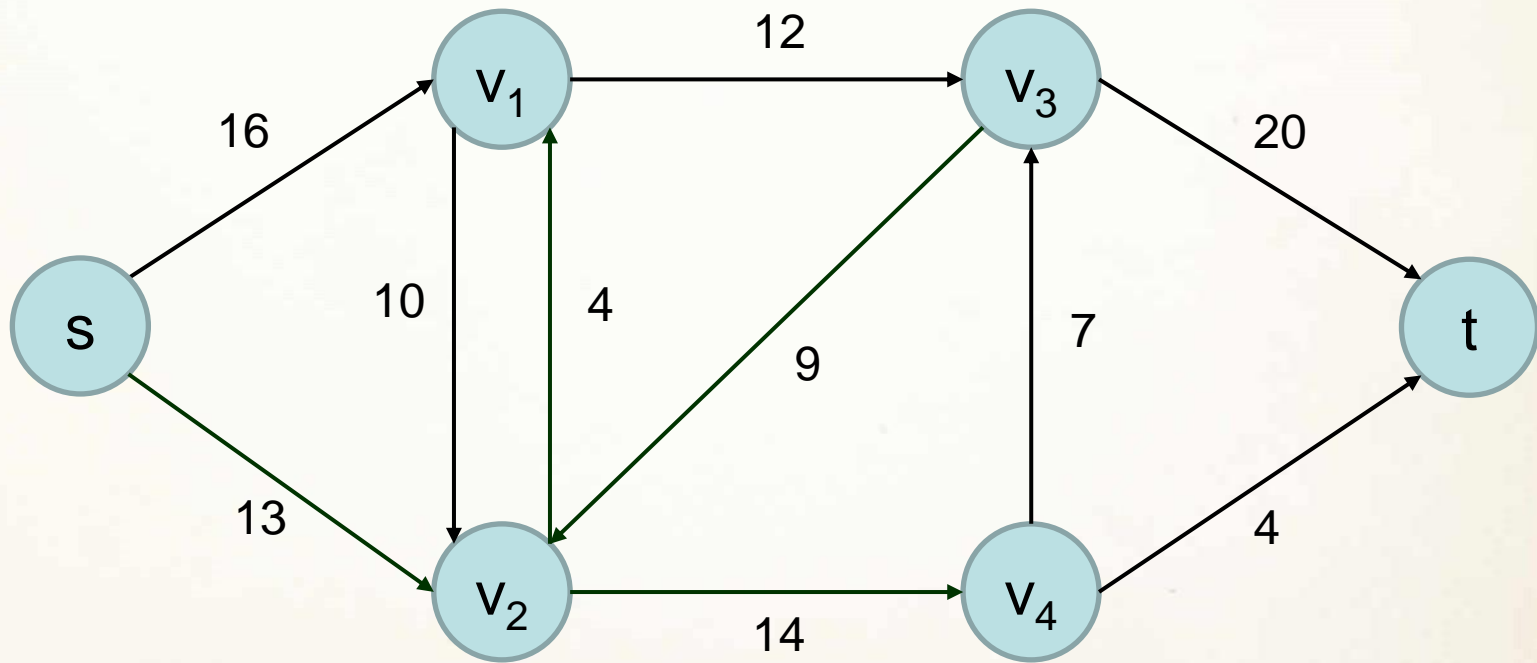
- Šis algoritmas realizuoja Fordo ir Fulkersono maksimalaus srauto paieškos idėją (metodą).

**F-F-met** ( $G, s, t$ ):

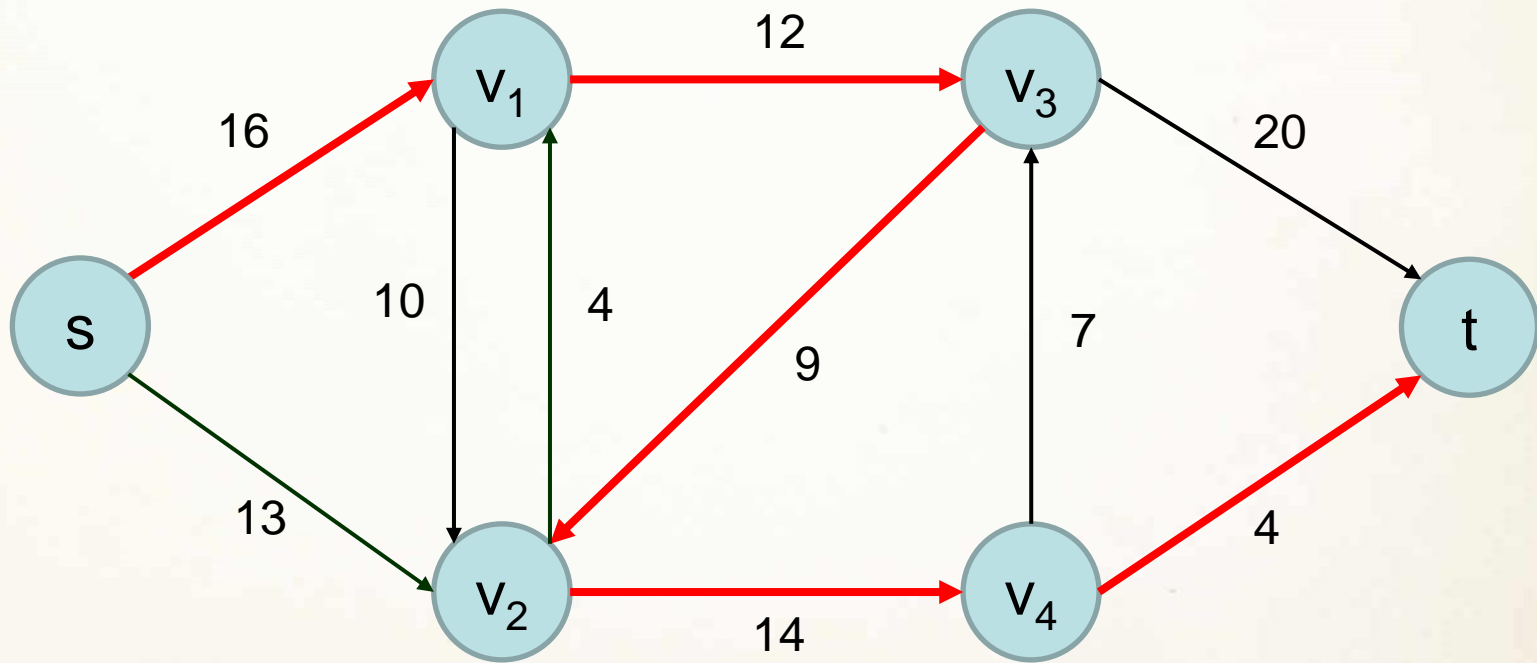
```
1. for  $\forall uv \in E$ 
2.   do  $f[uv] \leftarrow 0$ 
3.      $f[vu] \leftarrow 0$ 
4. while egzistuoja auginantis takas  $p$ 
5.   do  $c_f(p) \leftarrow \min\{c_f(uv) : uv \in p\}$ 
6.     for  $\forall uv \in p$ 
7.       do  $f[uv] \leftarrow f[uv] + c_f(p)$ 
8.          $f[vu] \leftarrow -f[uv]$ 
END
```

- Algoritmo sudėtingumas  $O(|f^*||E|)$ , kur  $f^*$  – maksimalus srautas.

# Edmondso–Karpo algoritmo pavyzdys (1)

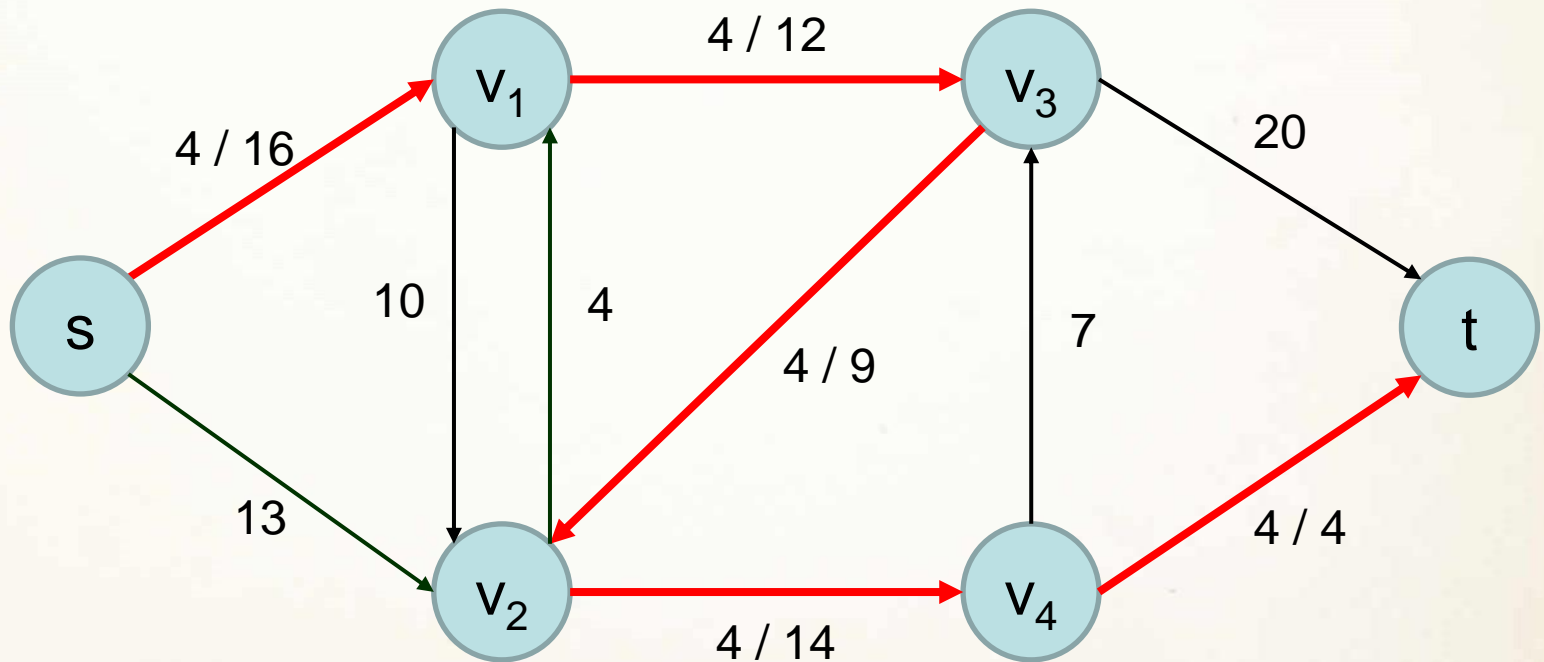


# Edmondso–Karpo algoritmo pavyzdys (2)



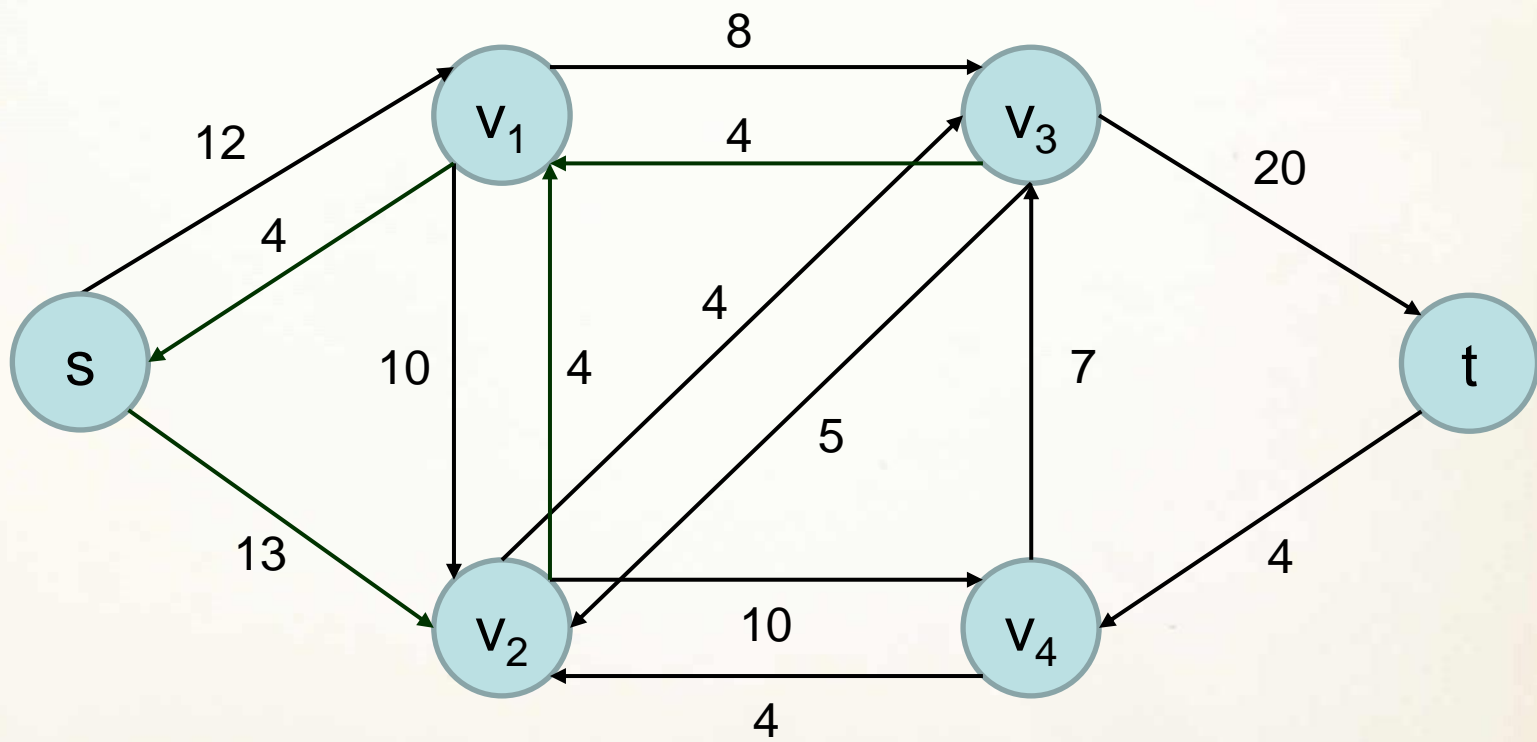
# Edmondso–Karpo algoritmo pavyzdys (3)

Ženklas „/“ atskiria srautą ir talpą, t. y.  $f(u, v) / c(u, v)$ .



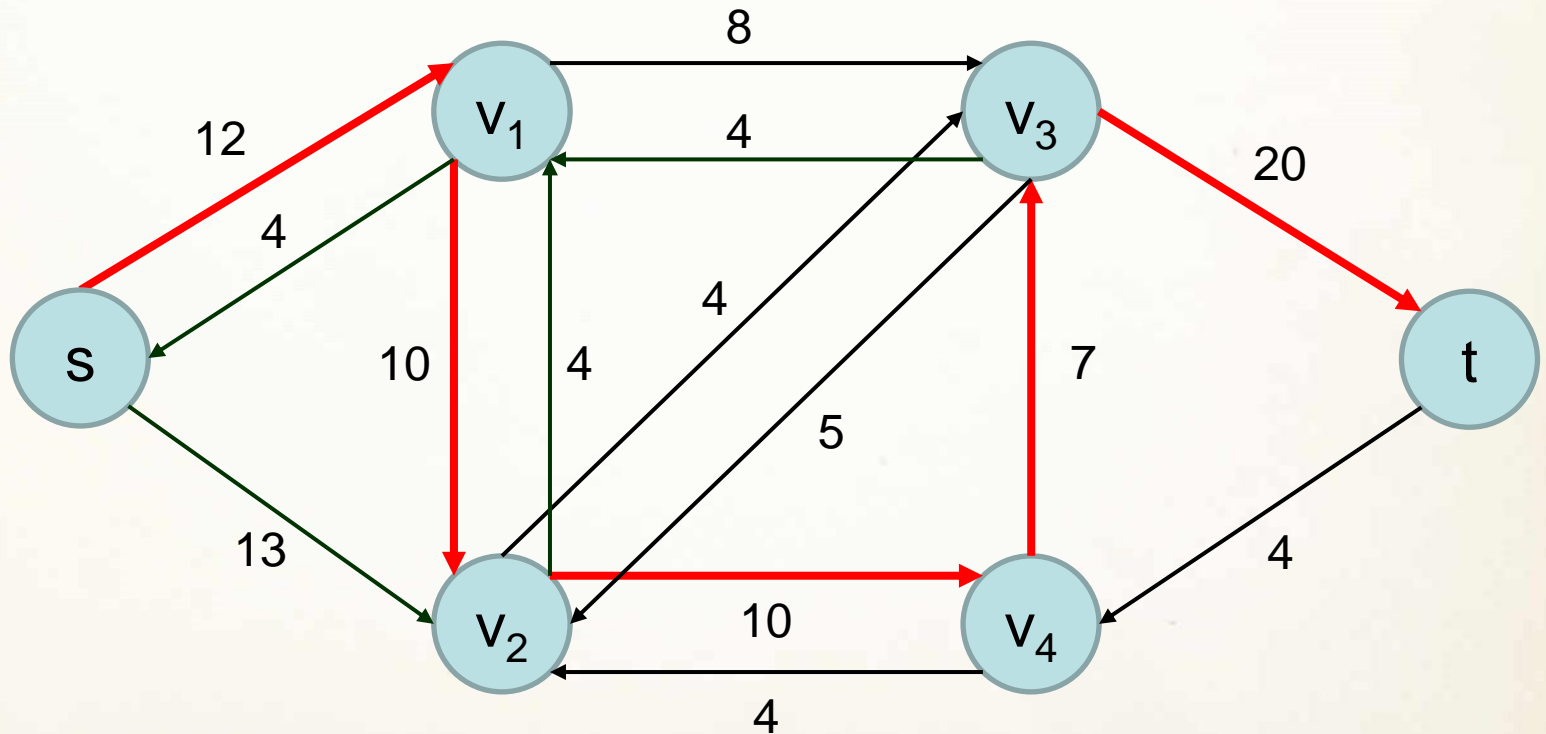
$$|f| = 4 +$$

# Edmondso–Karpo algoritmo pavyzdys (4)



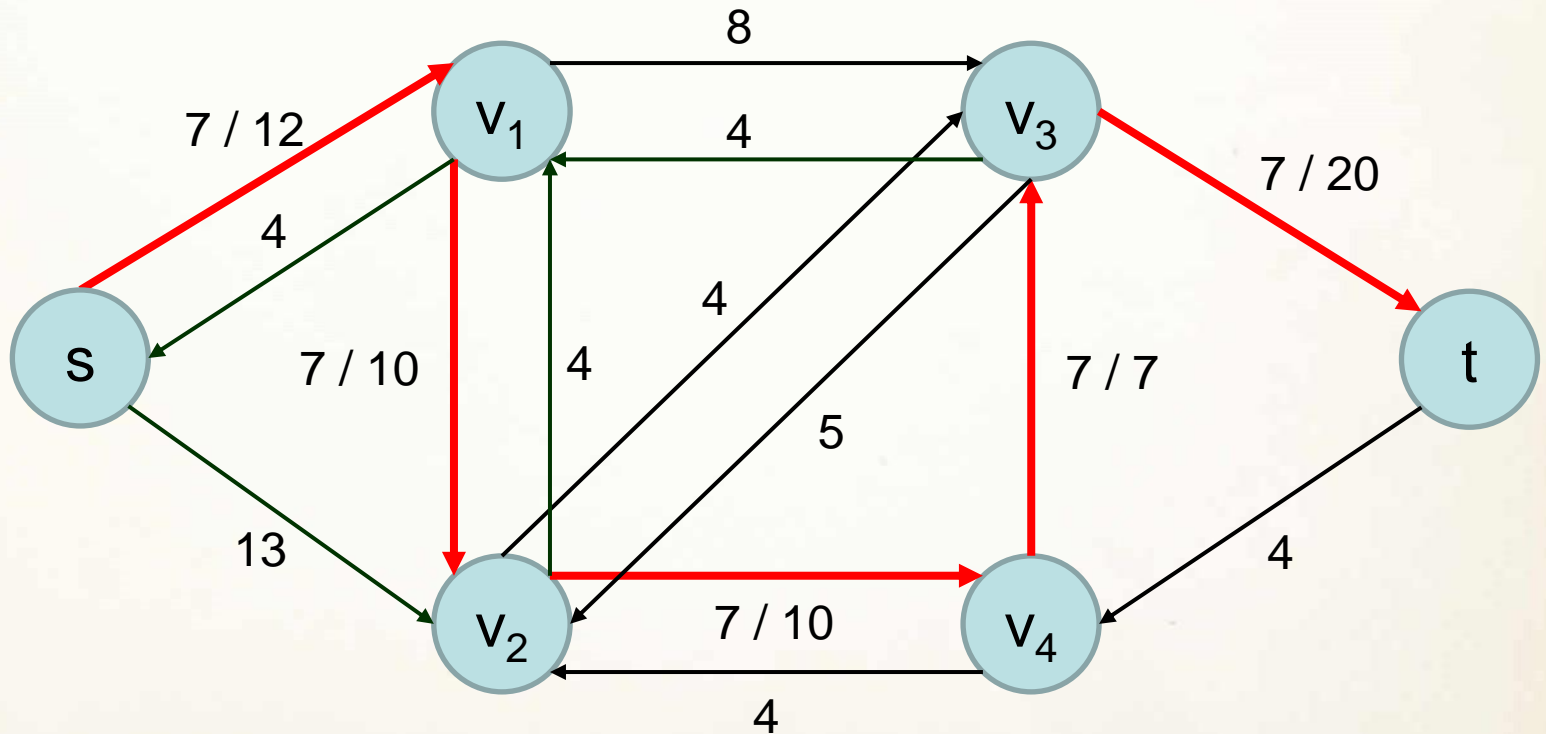
$$|f| = 4 +$$

# Edmondso–Karpo algoritmo pavyzdys (5)



$$|f| = 4 +$$

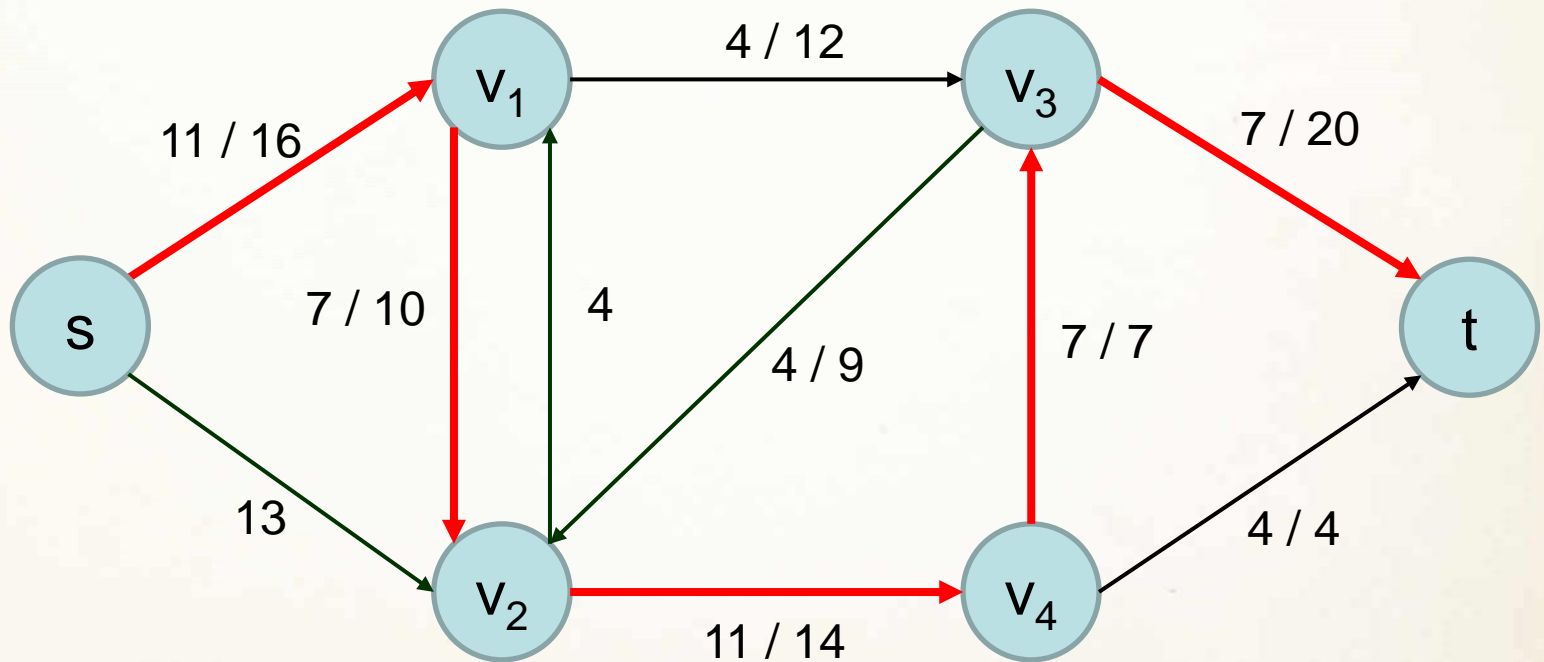
# Edmondso–Karpo algoritmo pavyzdys (6)



$$|f| = 4 + 7 +$$

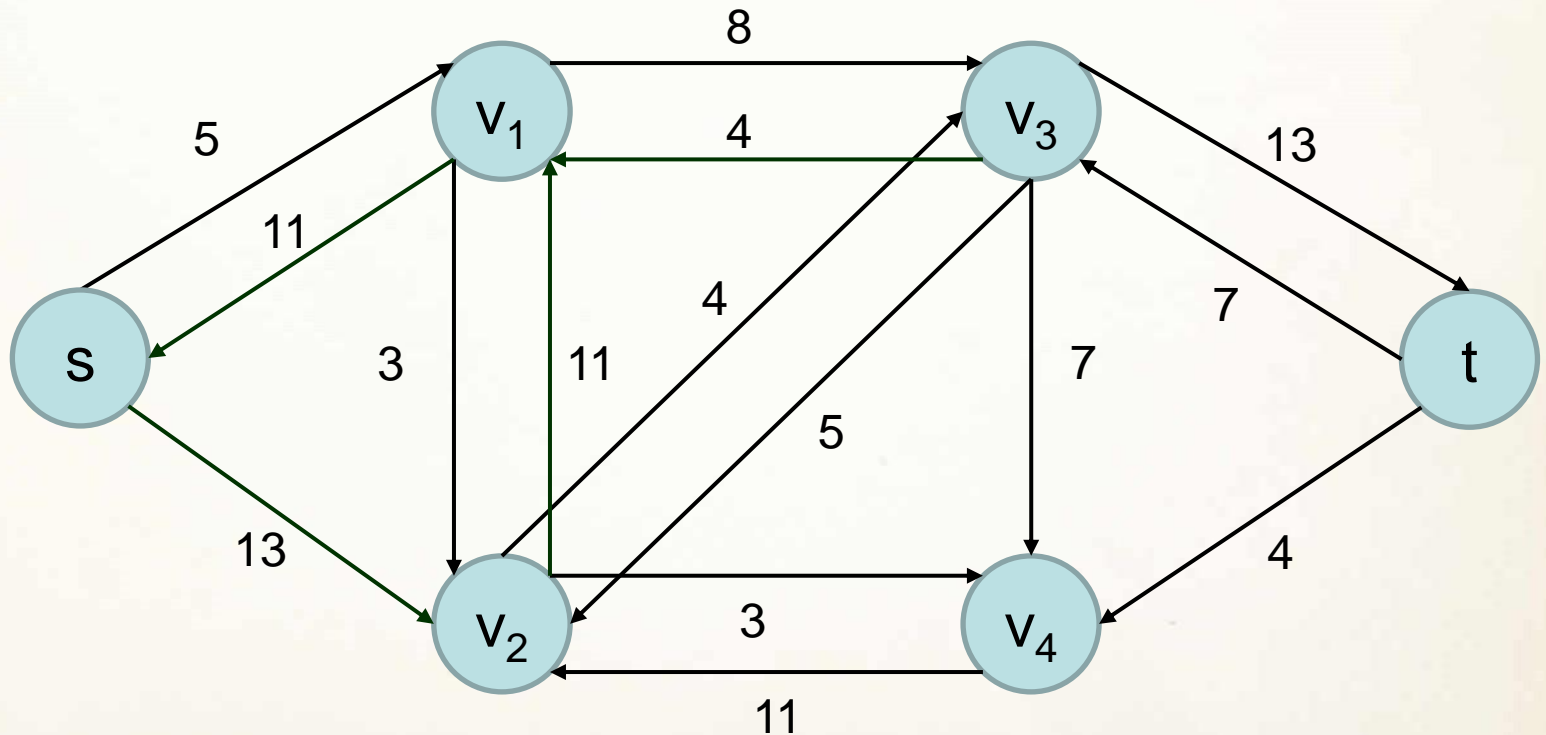


# Edmondso–Karpo algoritmo pavyzdys (7)



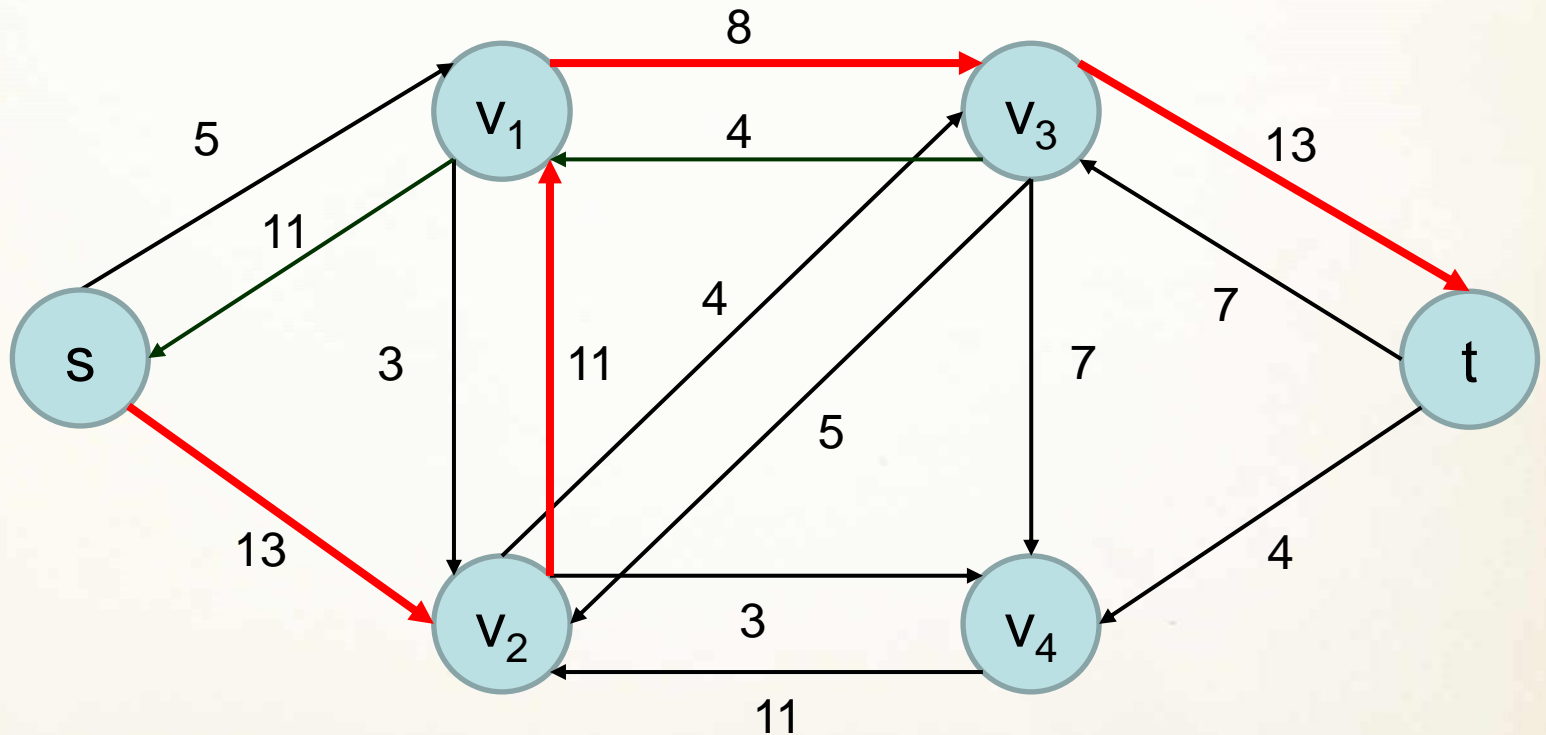
$$|f| = 4 + 7 +$$

# Edmondso–Karpo algoritmo pavyzdys (8)



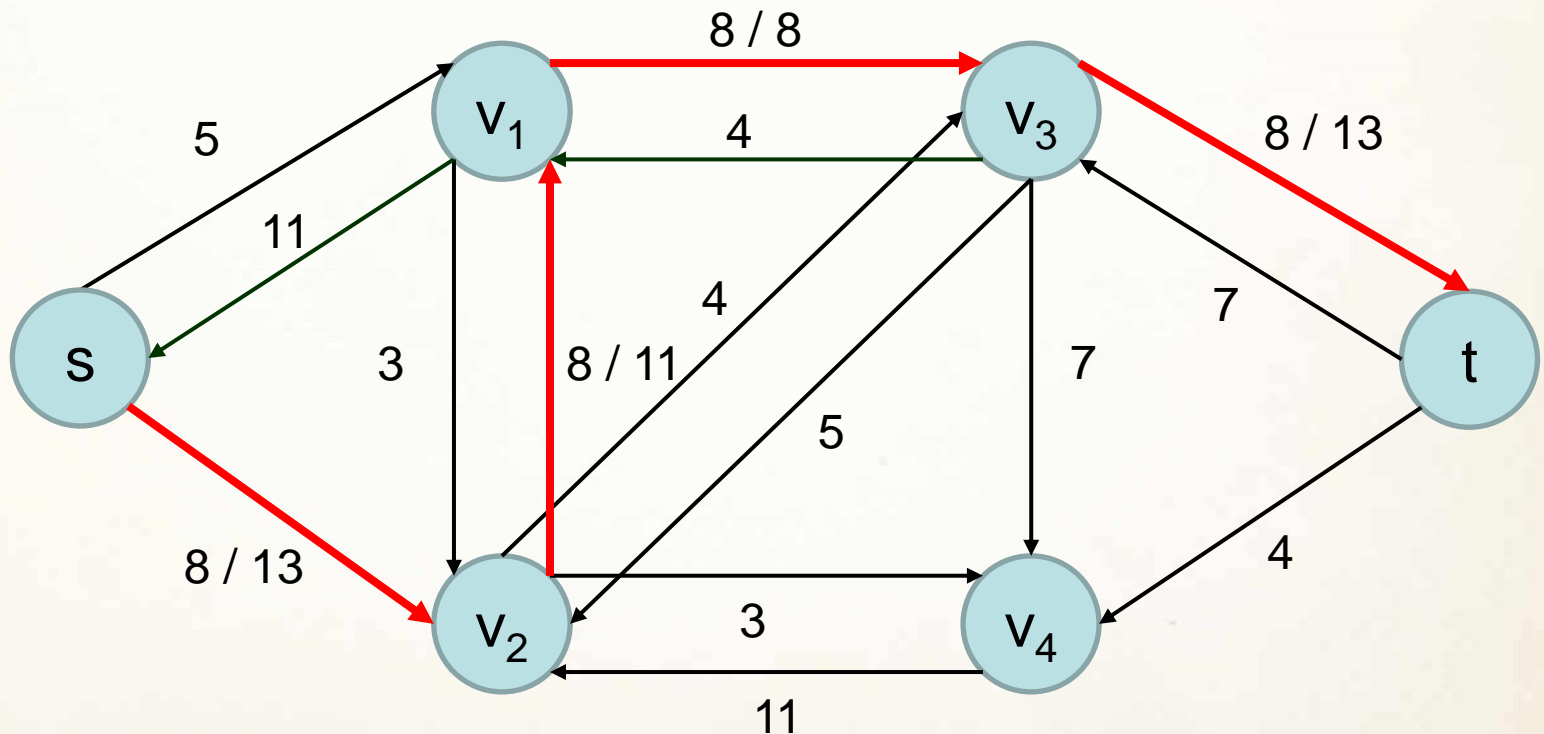
$$|f| = 4 + 7 +$$

# Edmondso–Karpo algoritmo pavyzdys (9)



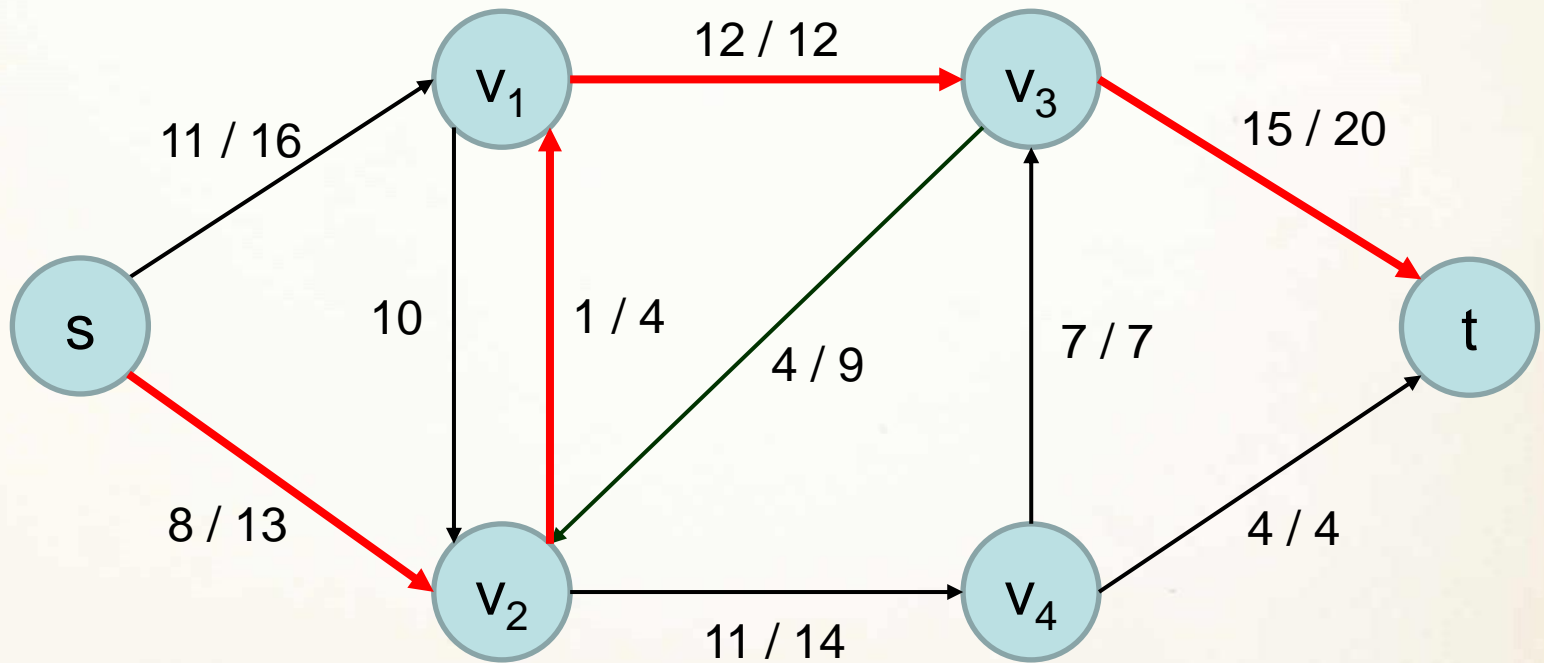
$$|f| = 4 + 7 +$$

# Edmondso–Karp algoritmo pavyzdys (10)



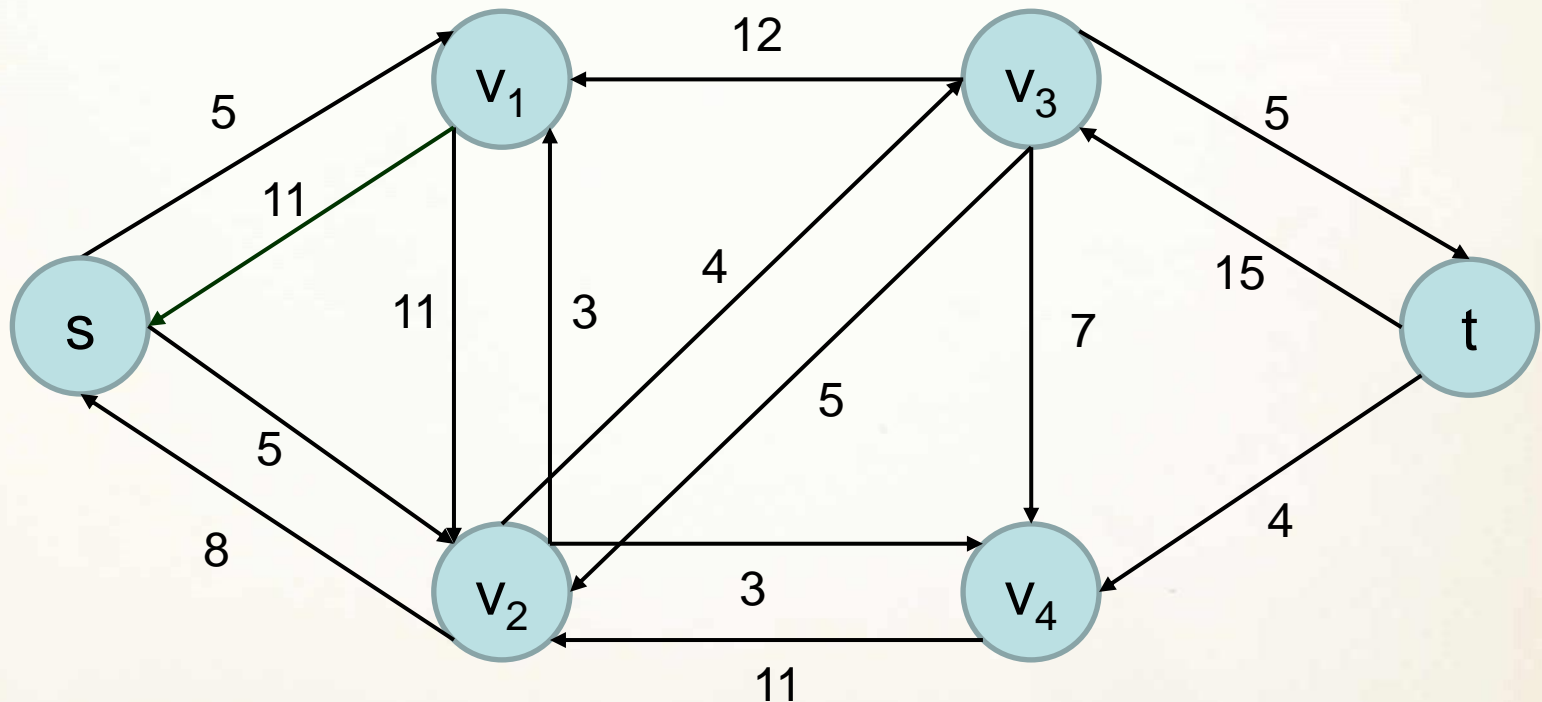
$$|f| = 4 + 7 + 8 +$$

# Edmondso–Karpo algoritmo pavyzdys (11)



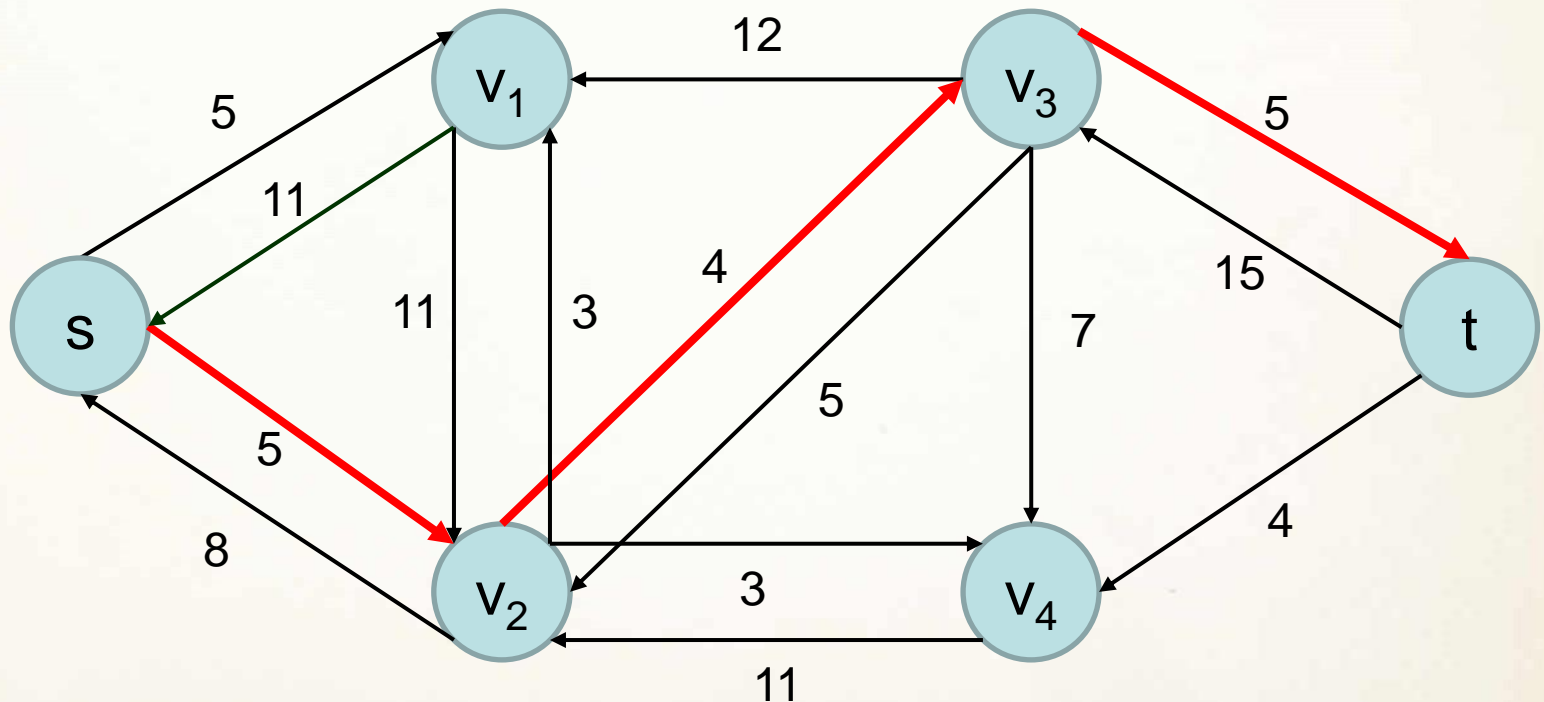
$$|f| = 4 + 7 + 8 +$$

# Edmondso–Karp algorithmo pavyzdys (12)



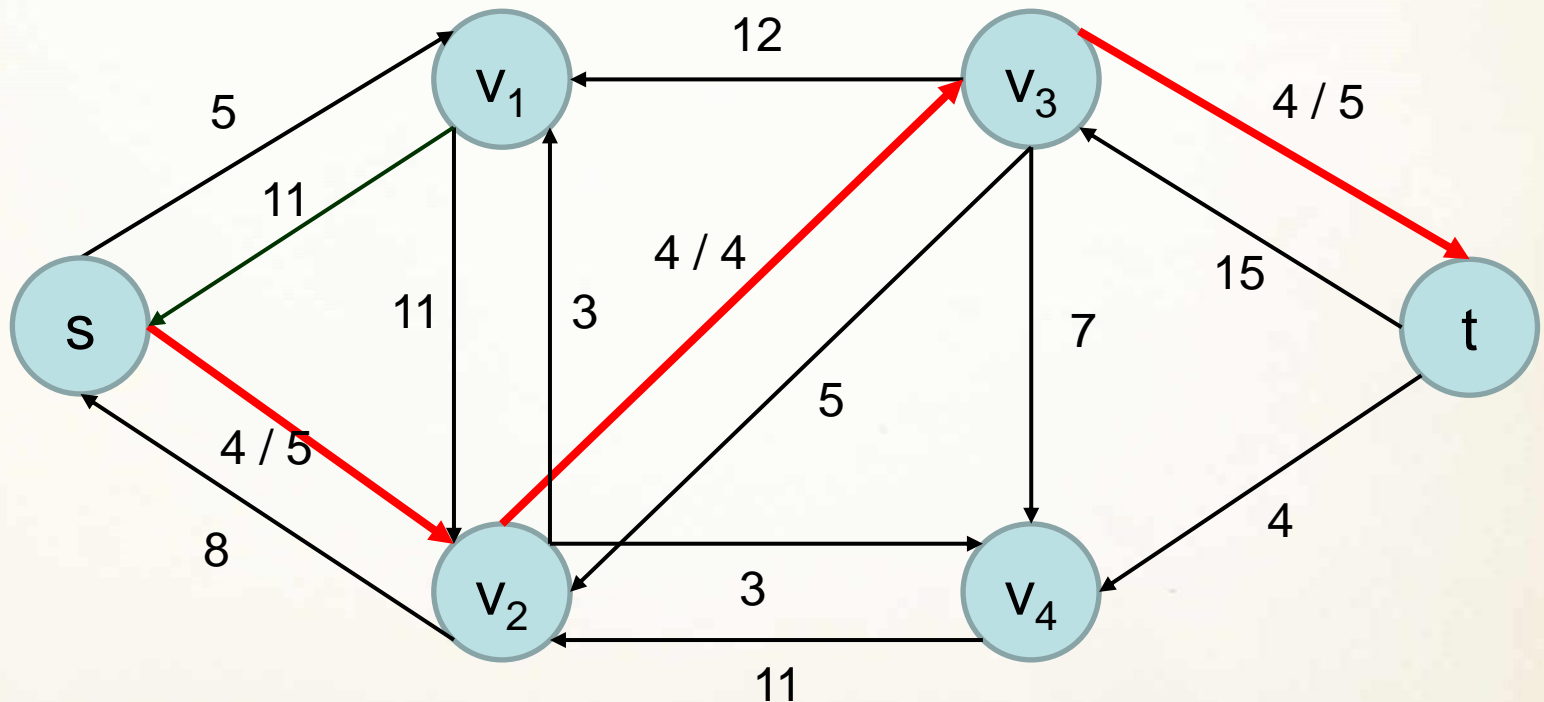
$$|f| = 4 + 7 + 8 +$$

# Edmondso–Karp algorithmo pavyzdys (13)



$$|f| = 4 + 7 + 8 +$$

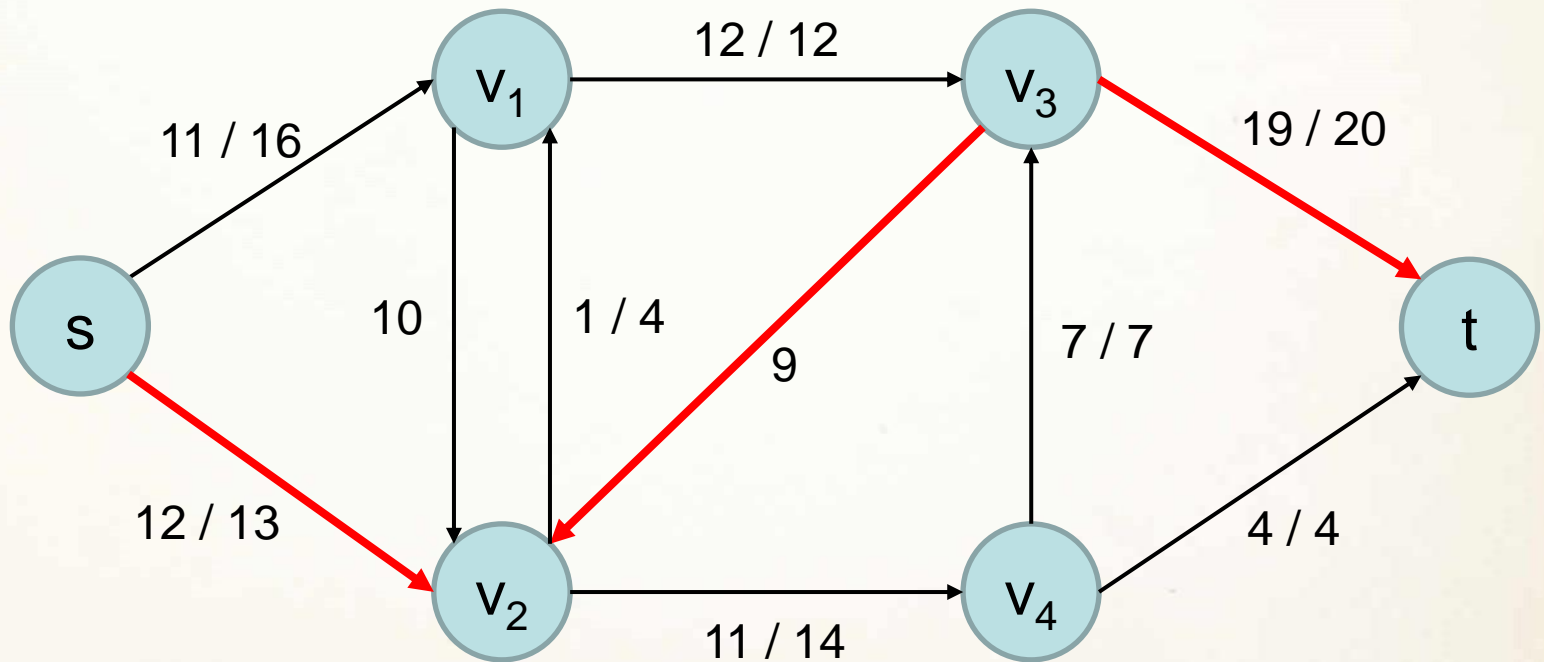
# Edmondso–Karp algorithmo pavyzdys (14)



$$|f| = 4 + 7 + 8 + 4 = 23$$

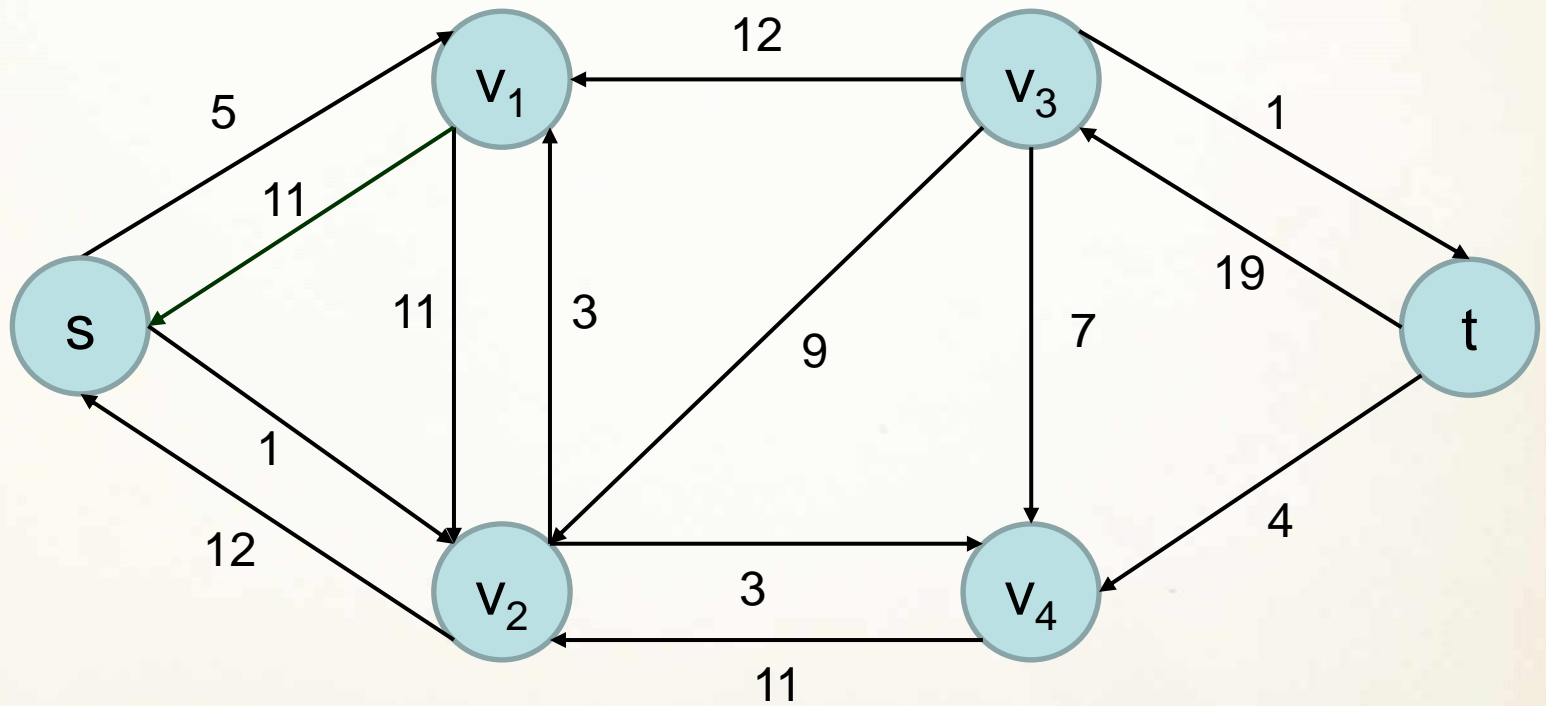


# Edmondso–Karp algoritmo pavyzdys (15)



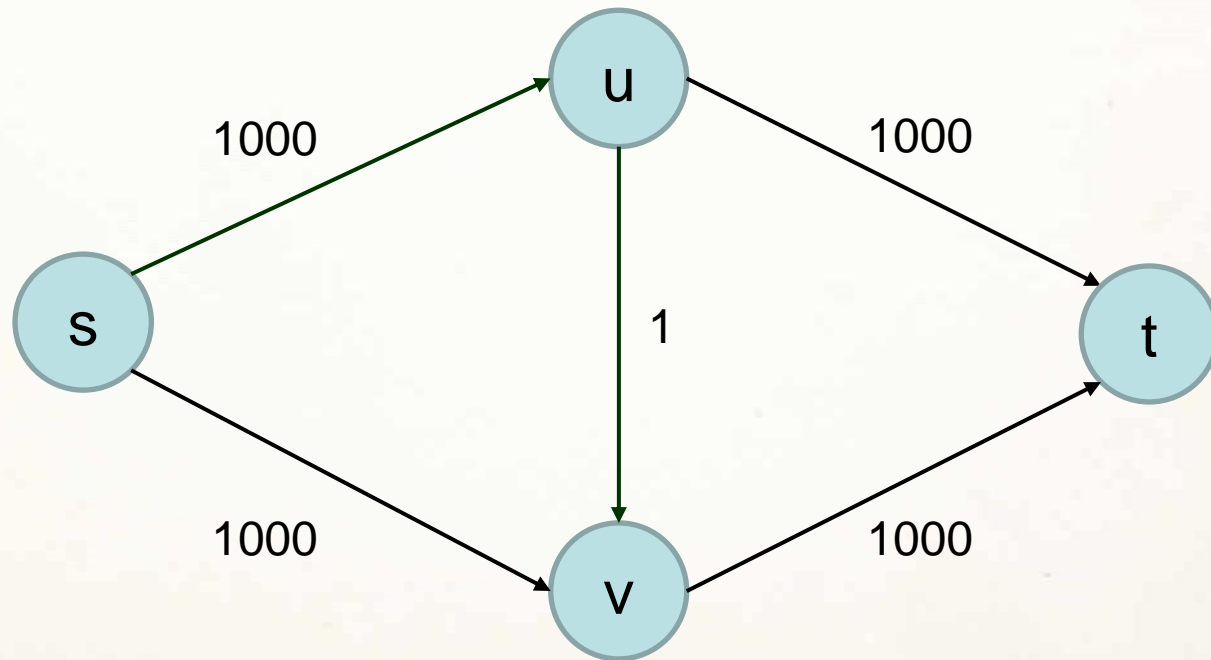
$$|f| = 23$$

# Edmondso–Karp algorithmo pavyzdys (16)

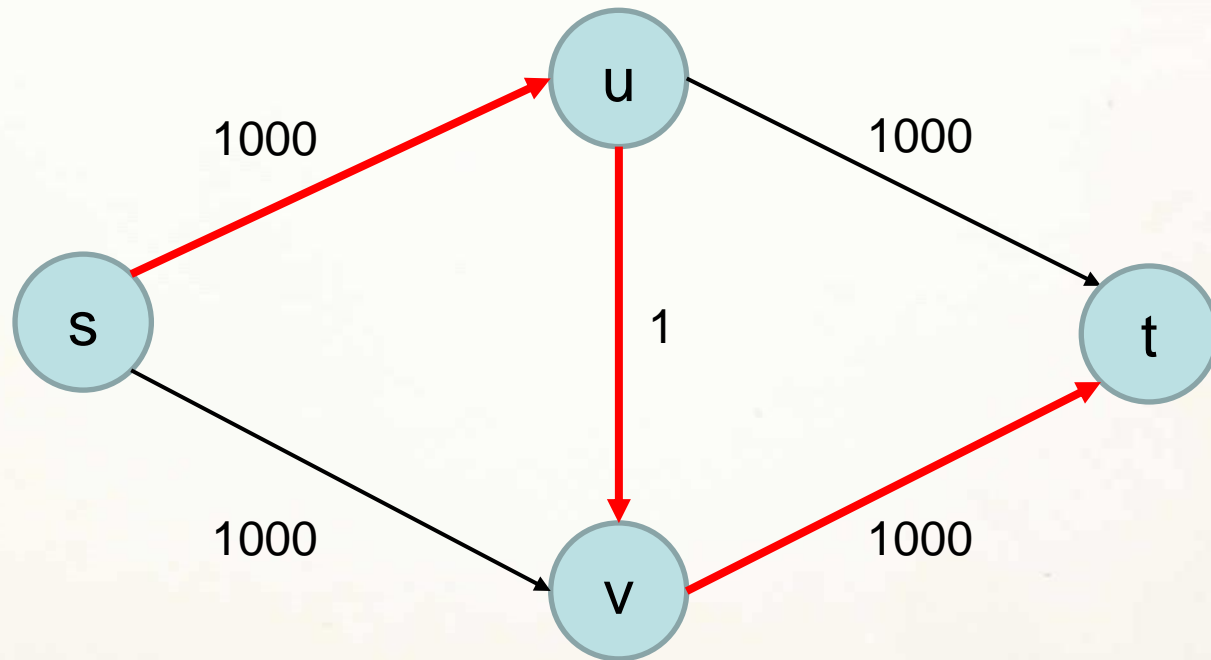


$$|f| = 23$$

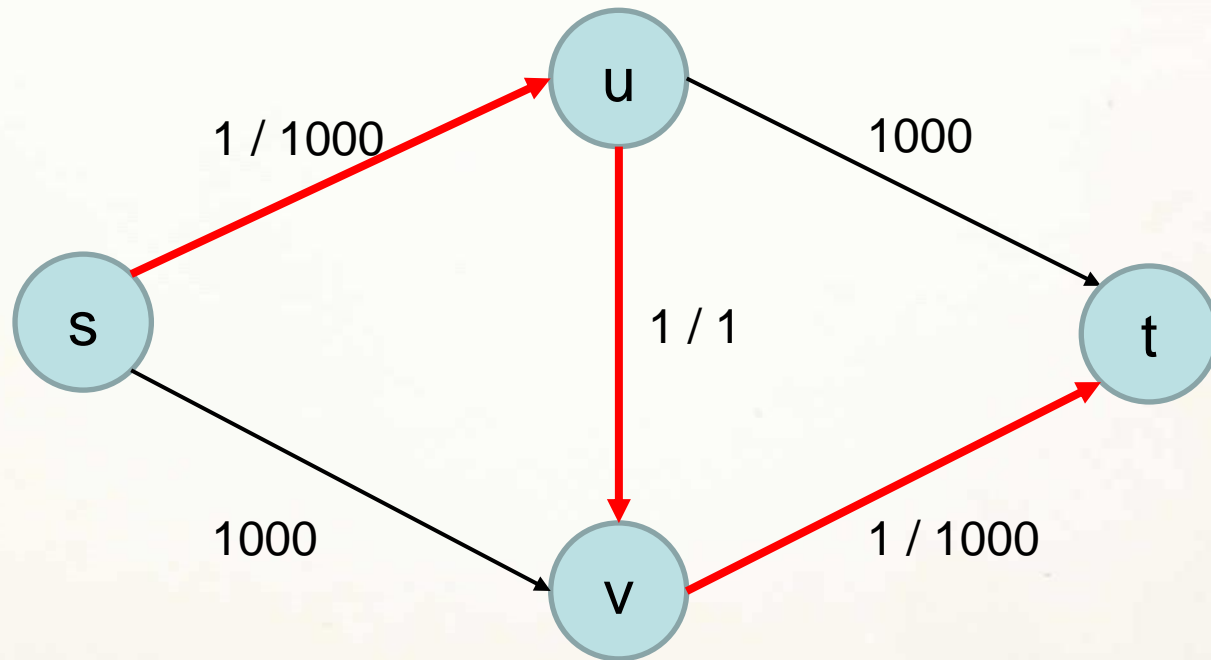
# Kitas pavyzdys (nepalankus atvejis)



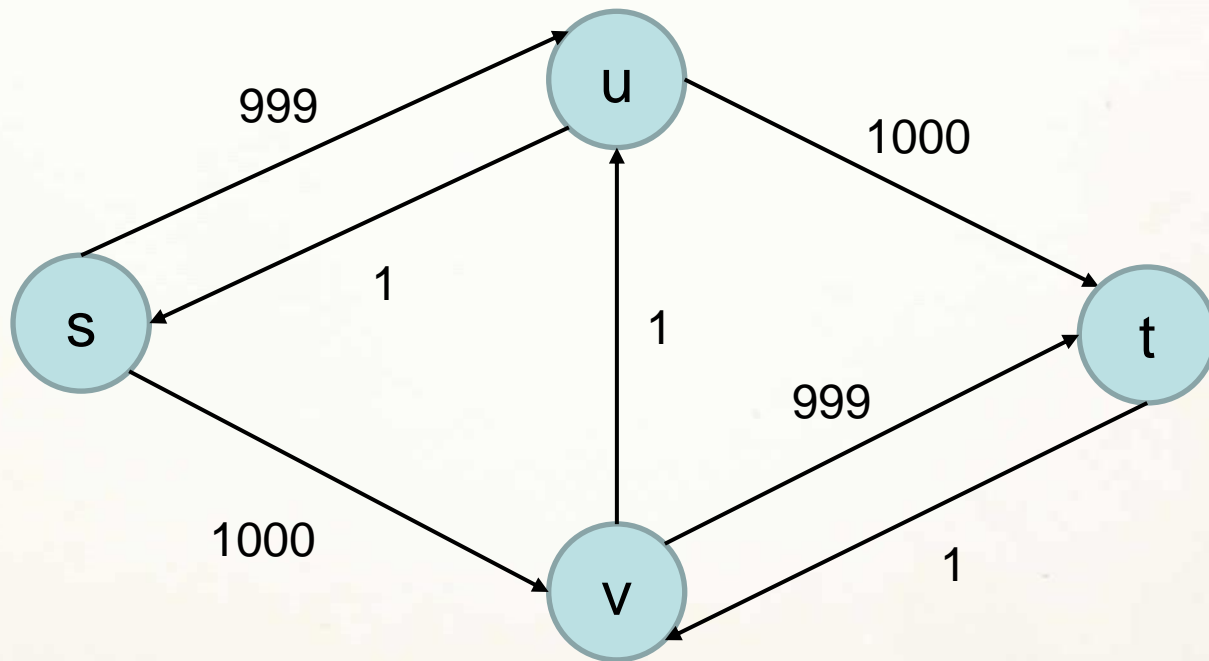
# Kitas pavyzdys (nepalankus atvejis)



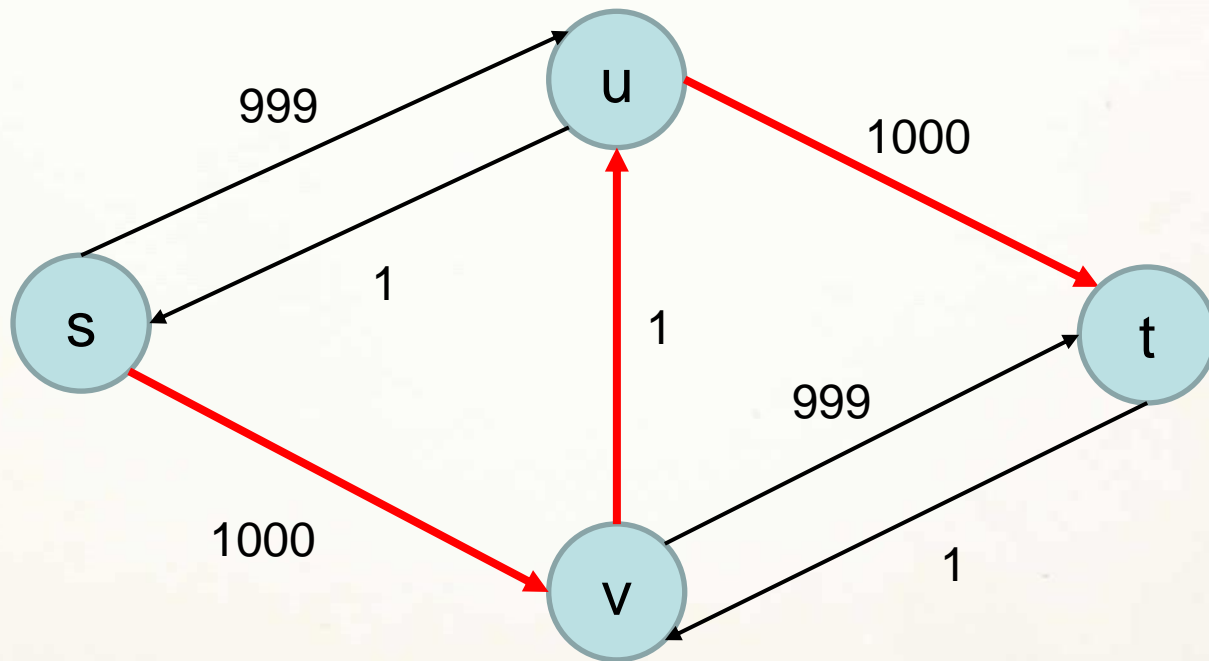
# Kitas pavyzdys (nepalankus atvejis)



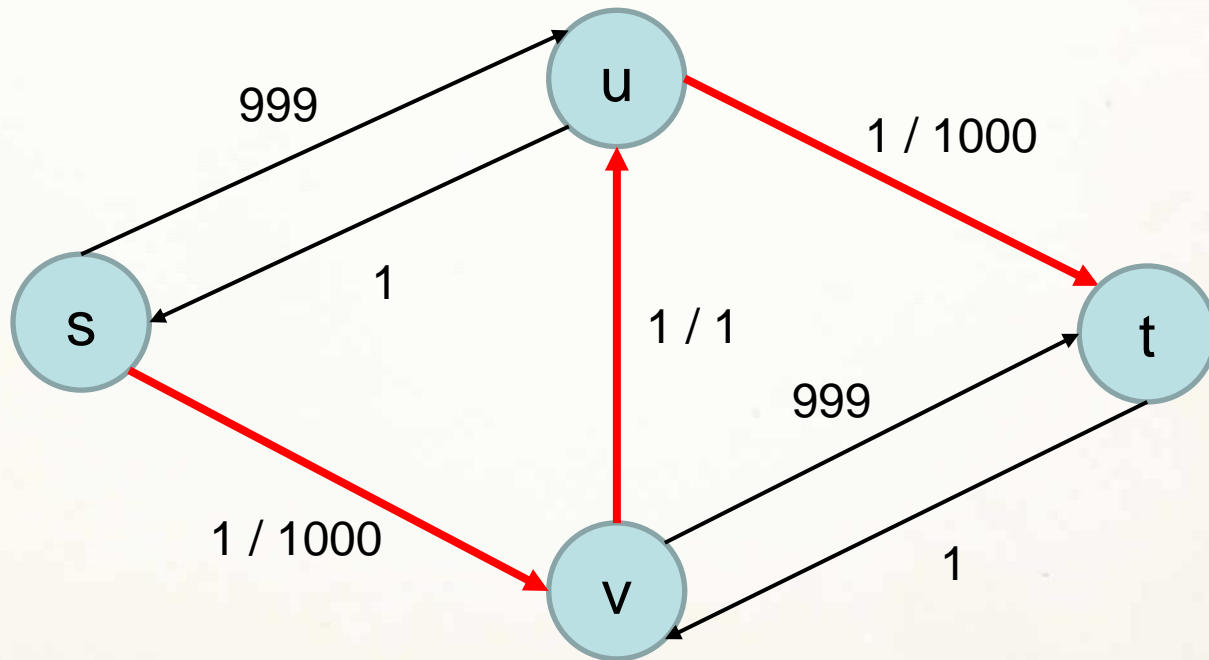
# Kitas pavyzdys (nepalankus atvejis)



# Kitas pavyzdys (nepalankus atvejis)

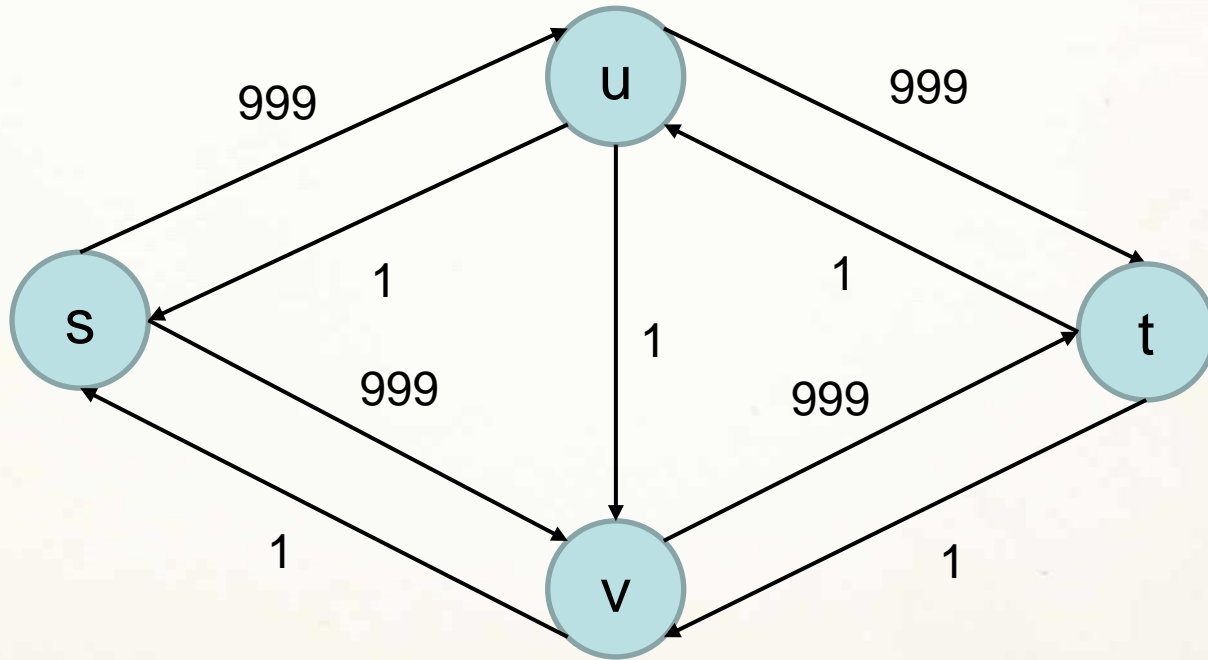


# Kitas pavyzdys (nepalankus atvejis)

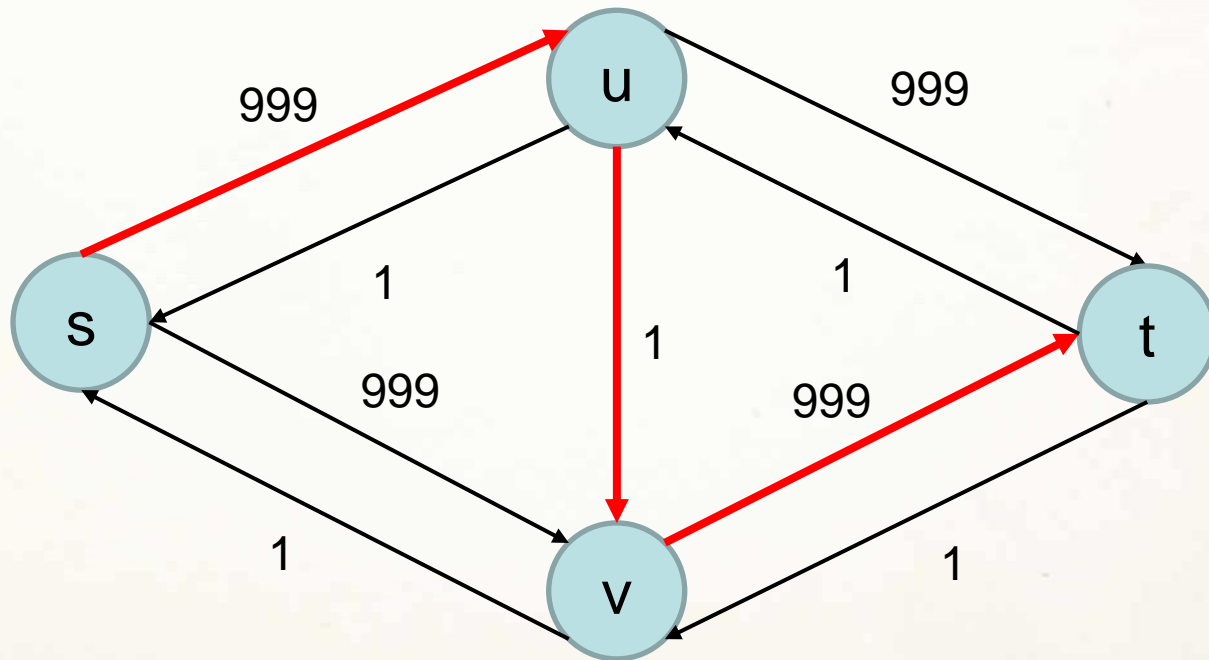




# Kitas pavyzdys (nepalankus atvejis)

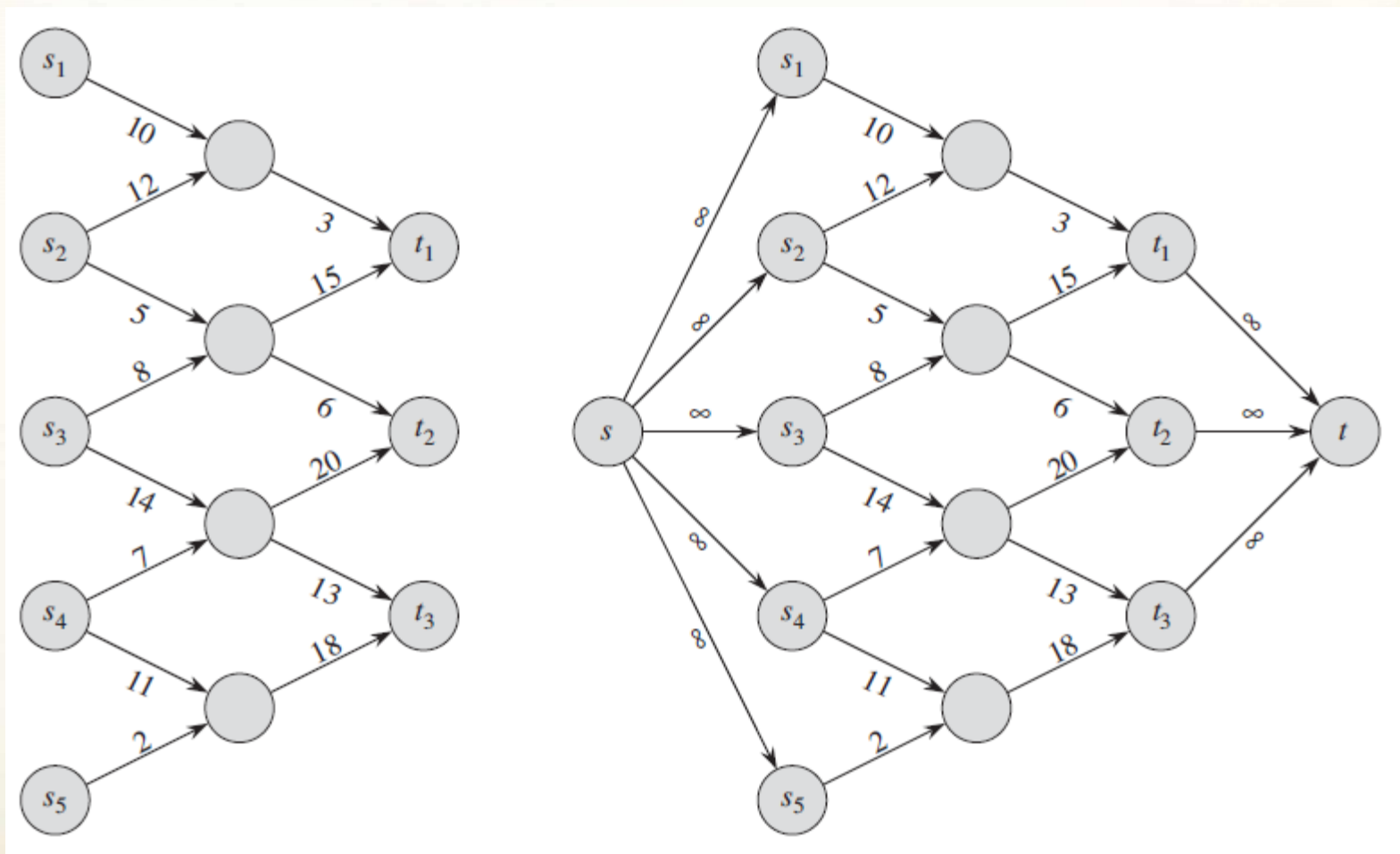


# Kitas pavyzdys (nepalankus atvejis)



ir t. t...

# Apibendrinto maksimalaus srauto uždavinio sprendimas



# Priešsraučio stūmimo algoritmas (*angl. Preflow-Push algorithm*)

- Šiuo algoritmu taip pat išsprendžiamas maksimalaus srauto apskaičiavimo uždavinys.
- Priešsraučio stūmimo algoritmo sudėtingumas –  $O(|V|^3)$ .
- Pagrindinės operacijos – stumk( $uv$ ) ir iškrauk( $u$ ).

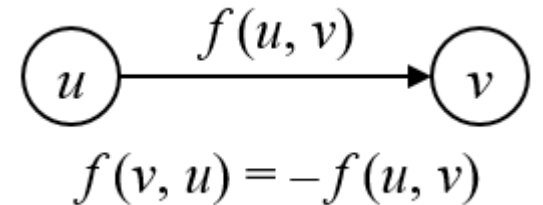
# Priešsraučio stūmimo algoritmas (*angl. Preflow-Push algorithm*)

- Priešsraučių tinkle vadinama funkcija  $f : V \times V \rightarrow \mathbb{R}$ , tenkinanti sąlygas:

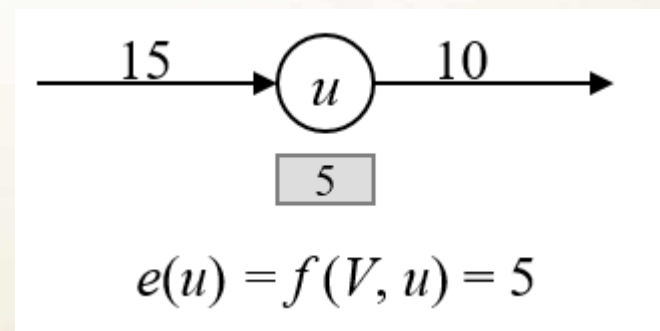
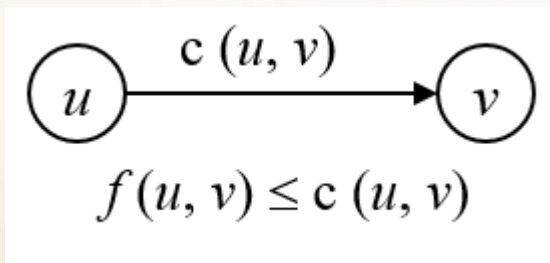
1)  $f(uv) = -f(vu)$ ,

2)  $f(uv) \leq c(uv)$ ,

3)  $e(u) = f(V, u) \geq 0$  kiekvienai  $u \in V \setminus \{s\}$ .



$c(uv)$  – briaunos  $uv$  talpa,  $f(vu)$  – briaunos  $uv$  svoris,  $e(u)$  – perteklinis srautas viršūnėje  $u$ .



# Priešsraučio stūmimo algoritmas

- **Likutinė talpa**  $c_f$  parodo, kokį svorį galima perkelti iš viršūnės  $u$  į viršūnę  $v$ :

$$c_f(u, v) = c(u, v) - f(u, v)$$

- **Aukščių** vadinama funkcija  $h : V \rightarrow \mathbb{N}$ , tenkinanti sąlygas:

$$h(s) = |V|$$

$$h(t) = 0$$

$$h(u) \leq h(v) + 1, \quad \forall (u, v) \in E_f$$

- Likutiniu tinklu vadinamas grafas  $G_f = (V, E_f)$  :

$$E_f = \{ (u, v) \in V \times V : c_f(u, v) > 0, h(u) \leq h(v) + 1 \}$$

# Stumk(uv) ir Kelk(u)

Stumk(uv) bus vykdoma tada, kai:

$$c_f(uv) > 0,$$

$$h(u) = h(v) + 1.$$

**Stumk(uv):**

1.  $d_f(uv) \leftarrow \min\{e[u], c_f(uv)\}$
2.  $f[uv] \leftarrow f[uv] + d_f(uv)$
3.  $f[vu] \leftarrow -f[uv]$
4.  $e[u] \leftarrow e[u] - d_f(uv)$
5.  $e[v] \leftarrow e[v] + d_f(uv)$

Vykdymo metu briauna uv išstumiamas srautas

$$d_f(uv) = \min\{e(u), c_f(uv)\}.$$

Kelk(u) bus vykdoma tada, kai:

$$c_f(uv) > 0 \text{ ir } h(u) \leq h(v)$$

visoms  $uv \in E_f$ .

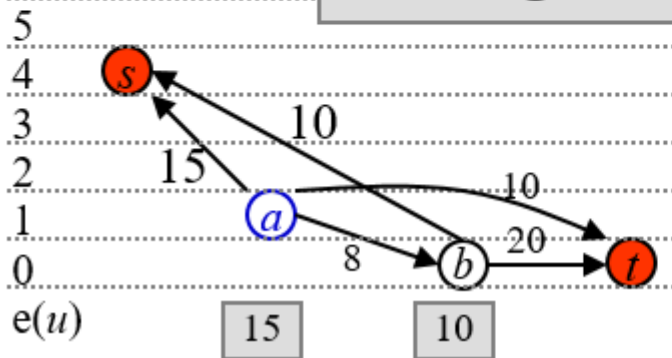
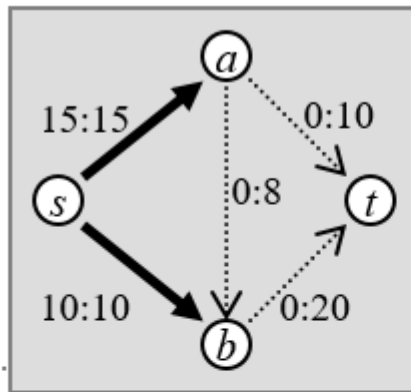
**Kelk(u):**

1.  $h[u] \leftarrow 1 + \min\{h[v] : uv \in E_f\}$

Kelk(u) atliekama, kai viršūnė u yra perteklinė visoms briaunoms uv. Šaltinio s ir tikslo t viršūnės nėra keliamos.

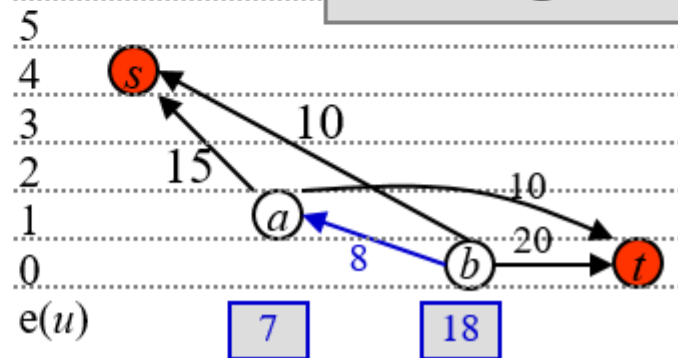
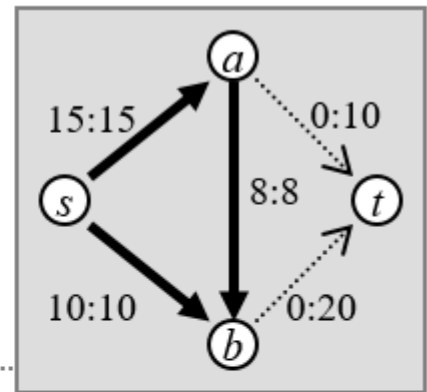
# Stumk(uv) ir Kelk(u) realizavimo pavyzdys

Kelk( $a$ )



Stumk( $a, b$ )

$$d_f = 8$$





# Priešsraučio stūmimo algoritmas

INIT( $G, s$ ):

1. for  $\forall u \in V$
  2.     do  $h[u] \leftarrow 0$
  3.      $e[u] \leftarrow 0$
  4. for  $\forall uv \in E$
  5.     do  $f[uv] \leftarrow 0$
  6.      $f[vu] \leftarrow 0$
  7.  $h[s] \leftarrow |V|$
  8. for  $\forall u \in Adj[s]$
  9.     do  $f[su] \leftarrow c(su)$
  10.      $f[us] \leftarrow -c(su)$
  11.      $e[u] \leftarrow c(su)$
- END

**Bendra Pr-St procedūra:**

1. INIT( $G, s$ )
2. **while** egzistuoja galimybė naudoti  
    **Kelk(u)** ar **Stumk(uv)**
3.     **do** tai.

# Maksimalaus srauto taikymai

- Maksimalus dvidalis suporavimas (Halo vedybų problema, *angl. Hall's marriage problem*).
- Ištrūkimo uždavinys (arba kitaip – išsigelbėjimo problema, *angl. escape problem*).

# Maksimalus dvidalis suporavimas

- *Suporavimu* grafe  $G = (V, E)$  vadinamas nepriklausomų briaunų poaibis  $M$ .
- $M \subset E$  yra suporavimas, jei kiekviena viršūnė  $v \in V$  turi ne daugiau kaip vieną incidentiją briauna, priklausančią aibei  $M$ .
- Maksimalus suporavimas turi didžiausią briaunų skaičių, t. y.  $|M| \rightarrow \max$ .
- Visiškas suporavimas dvidaliame grafe  $G = (L \cup R, E)$  yra toks poravimas, kai visos vienos dalies viršūnės yra incidentčios suporavimo briaunoms.

# Halo vedybų problema (1)

- *Visiško suporavimo  $M$  atveju  $|M| = \min\{|L|, |R|\}$ .*
- *Toks suporavimas ne visada egzistuoja, tačiau galima apskaičiuoti maksimalų dvidalį suporavimą.*
- *Šis uždavinys išsprendžiamas taikant maksimalaus srauto apskaičiavimo algoritmą, pavyzdžiui, Fordo–Fulkersono metodą.*

## Halo vedybų problema (2)

- Tegu  $G = (V, E)$  – pradinis dvidalis grafas.
- Sudarykime asocijuotą srauto tinklą  $G' = (V', E')$ , kur

$$V' = L \cup R \cup \{s, t\},$$

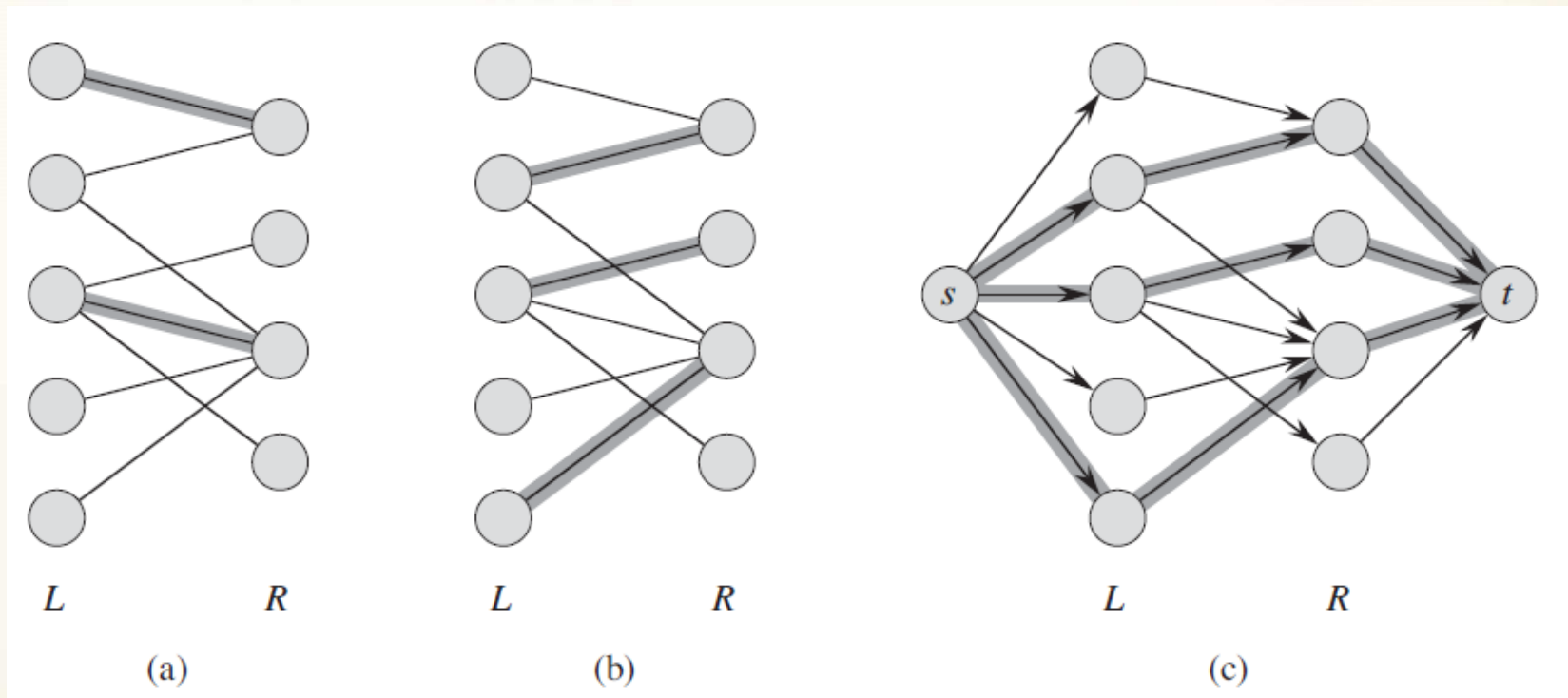
$$E' = \{uv \in E\} \cup \{su : u \in L\} \cup \{vt : v \in R\},$$

visoms briaunoms suteikime kryptis ir vienetines talpas.

**Teorema.** Jei  $M$  yra dvidalis suporavimas, tai asocijuotame tinkle egzistuoja srautas  $f$ , kurio didumas yra  $|f| = |M|$ .

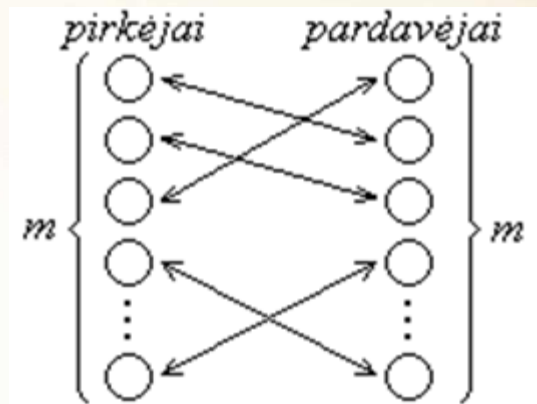
Atvirkščiai, turėdami tinkle  $G'$  sveikareikšmį srautą  $f$ , turime ir suporavimą  $M$ , kurio didumas yra  $|M| = |f|$ .

# Halo vedybų problemos pavyzdys



- (a) – nemaksimalus suporavimas,
- (b) – maksimalus suporavimas,
- (c) – maksimalus suporavimas taikant Fordo–Fulkersono metodą.

# Priskyrimo lošimas



- Tai panašus uždavinys dvidalio grafo suporavimui, tik formuluojamas lošimų teorijoje:

Tegu  $N = M \cup M'$ , kur  $M$  ir  $M'$  nerikertančios aibės, turinčios po  $m$  elementų:  $M = \{1, 2, \dots, m\}$  ir  $M' = \{m+1, m+2, \dots, 2m\}$ . Lošėjas  $\{i\} \in M$ ,  $i = 1, 2, \dots, m$ , ( $i$ -tasis pardavėjas) turi namą, kurio vertė  $a_i$  dolerių. Lošėjas  $\{m+j\} \in M'$ ,  $j = 1, 2, \dots, m$ , ( $j$ -tasis pirkėjas) nori pirkti namą maksimaliai skirdamas  $b_{ij}$  dolerių. Tokiu atveju koalicija tarp dviejų lošėjų  $\{i, m+j\}$  gauna pelną, kuris lygus:

$$v(\{i, m+j\}) = c_{ij} = \begin{cases} b_{ij} - a_i, & \text{kai } b_{ij} \geq a_i, \\ 0, & \text{kai } b_{ij} \leq a_i. \end{cases}$$

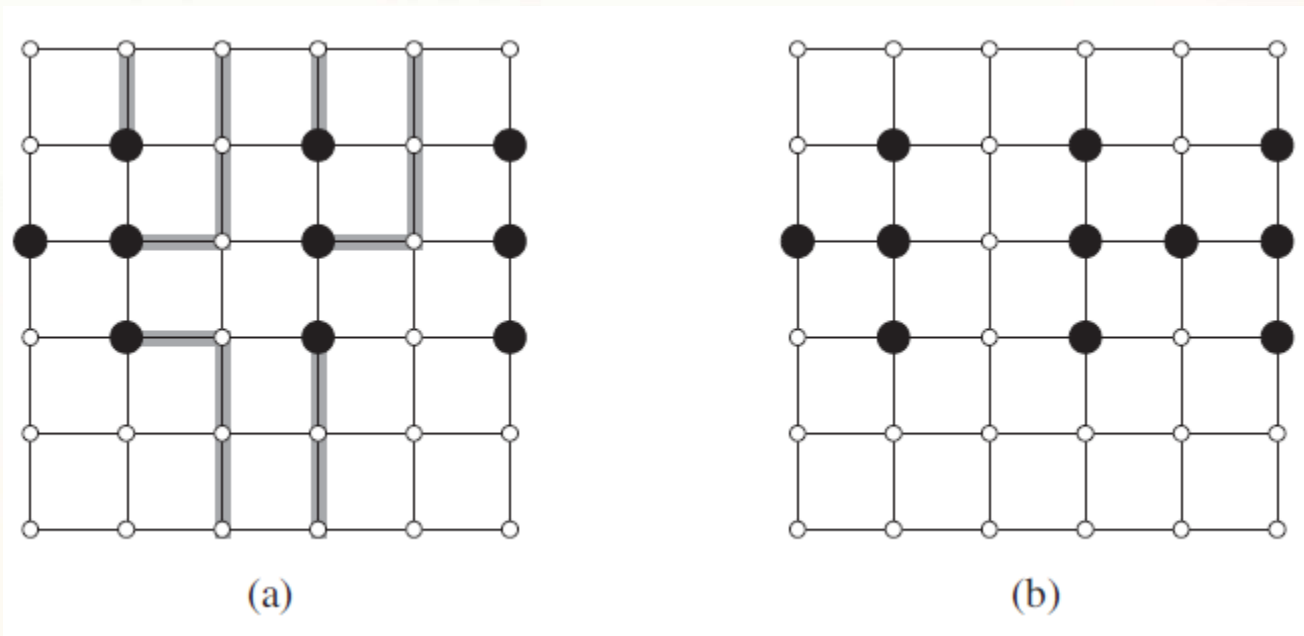
- Priskyrimo lošimo uždavinio tikslas – maksimizuoti visų koalicijų pelną.
- Klausimas diskusijai.** Ar galima priskyrimo lošimo uždavinį išspręsti taikant maksimalaus srauto paieškos algoritmą?

# Ištrūkimo uždavinys

- *Tegu languoto  $n \times n$  sąsiuvinio lapas atitinka grafa*  
$$G = (V, E).$$
- *Šiame grafe pažymėta nepriklausomų  $m \leq n^2$  viršūnių, kuriose tupi kiškiai.*
- *Visiems kiškiams leidžiama keliauti grafo briaunomis sudarant skirtingus takus, tačiau bet kurie 2 takai negali turėti bendrų viršūnių.*
- ***Ištrūkimo uždavinys.*** *Ar gali visi kiškiai pabėgti į grafo pakraštį?*



# Ištrūkimo uždavinys



- (a) – kiškių pabėgimas įmanomas,
- (b) – kiškių pabėgimas neįmanomas, kodėl?

# Ištrūkimo uždavinio sprendimas

- Tegū  $x_{ij} \in V$  – grafo  $G = (V, E)$  viršūnės,  $1 \leq i, j \leq n$ .
- Tada  $x_{ij} \in V$ , o  $x_{ij}x_{i+1,j} \in E$  ir  $x_{i+1,j}x_{i,j} \in E$ , jei  $i \leq n - 1$ .
- Tegū grafo briaunos ir viršūnės turi vienetines talpas.
- Tegū  $V' = V \cup \{s, t\}$ , čia  $s$  – šaltinio ir  $t$  – tikslo viršūnės.
- Jei  $x_{ij}$  tupi kiškis, pridėkime briauną  $sx_{ij}$ , jei  $x_{kl}$  – kraštinė viršūnė, pridėkime briauną  $x_{kl}t$ . Gauname grafą  $G' = (V', E')$ .
- **Teorema.** Ištrūkimo uždavinio sprendinys egzistuoja tada ir tik tada, jei sukonstruotame tinkle  $G' = (V', E')$  maksimalus srautas  $|f|$  lygus  $m$ .
- **Irodymas.** Kaip ir dvidalio grafo suporavimo uždavinyje maksimalus srautas lygus nepriklausomų  $s \Rightarrow t$  takų skaičiui, t. y.  $|f| = m$ . To užtenka  $m$  kiškių pabėgti.

***Ačiū už dėmesį.***

***Klausimai?***