

Network Programming I

ものをなるべく多く流す

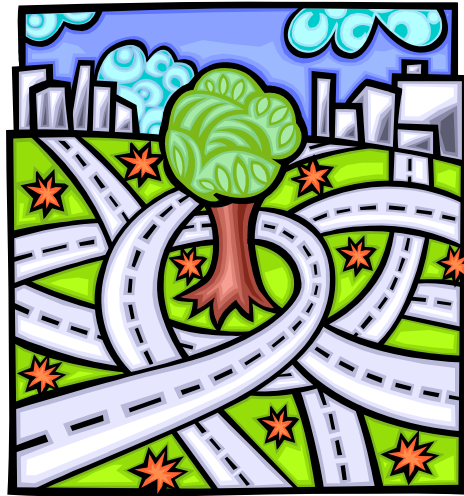
最大フロー問題

ネットワーク上に“モノ”を流す



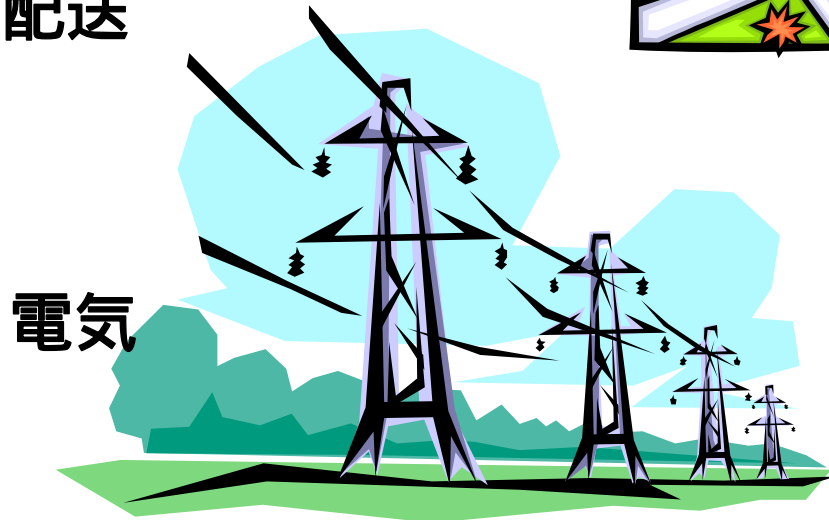
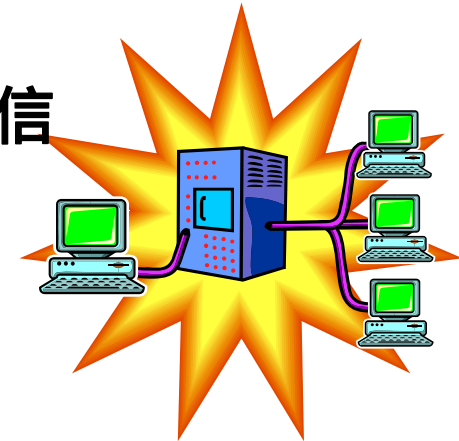
配送

道路



パイプライン

通信

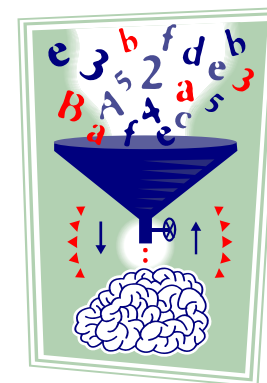


電気

モデル化

- グラフ + 容量 = ネットワーク
- フロー(流れ)

モノの流れのモデル化

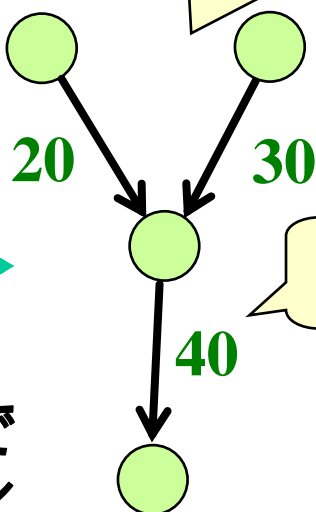


流れる量には
限界が有る

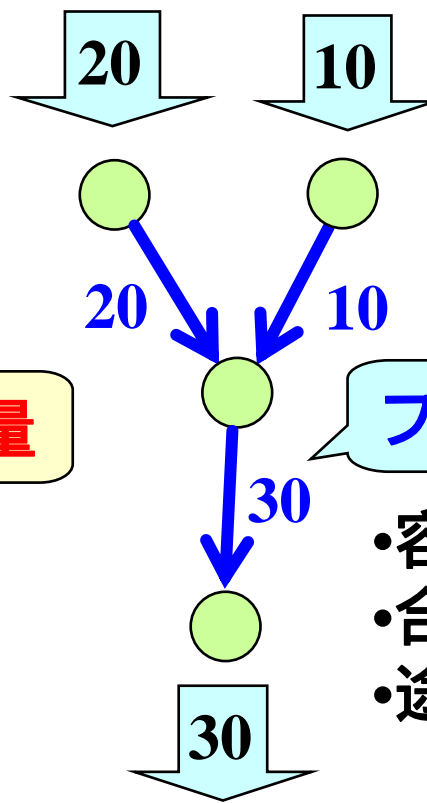
各枝に
容量情報を追加



舞台を
グラフで
モデル化



ネットワーク



フロー

- 容量以内
- 合流・分岐可
- 途中で増減無

流れを数字でモデル化

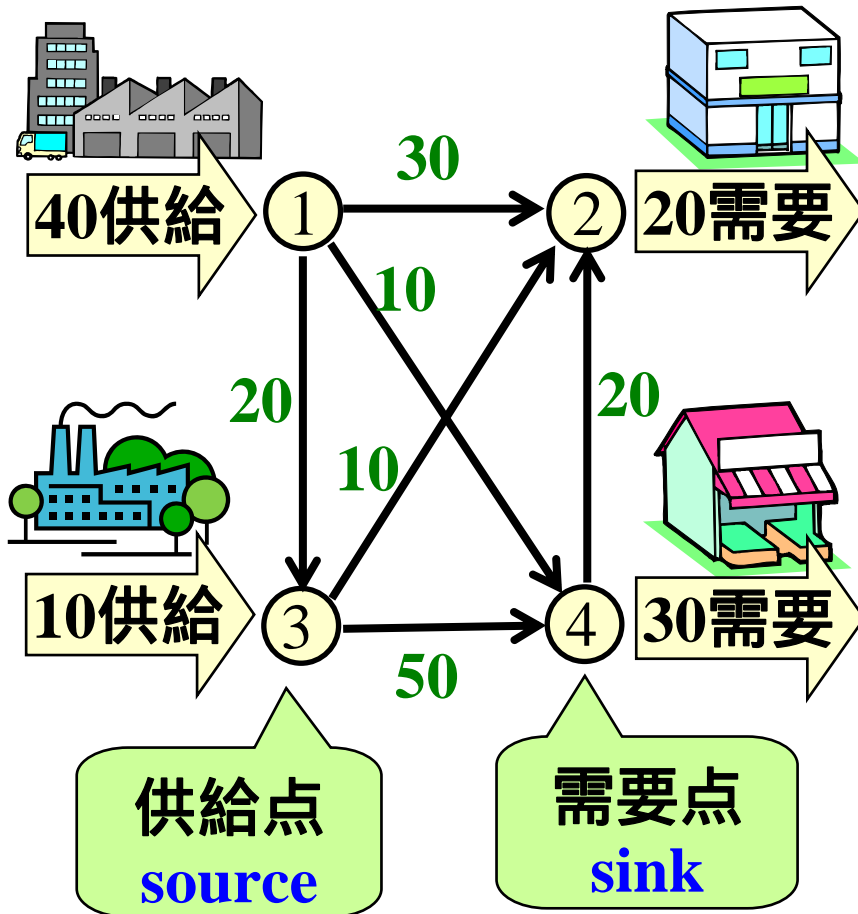
最適なフローの
制御方法は?

ネットワークフロー計画問題

2端子ネットワーク

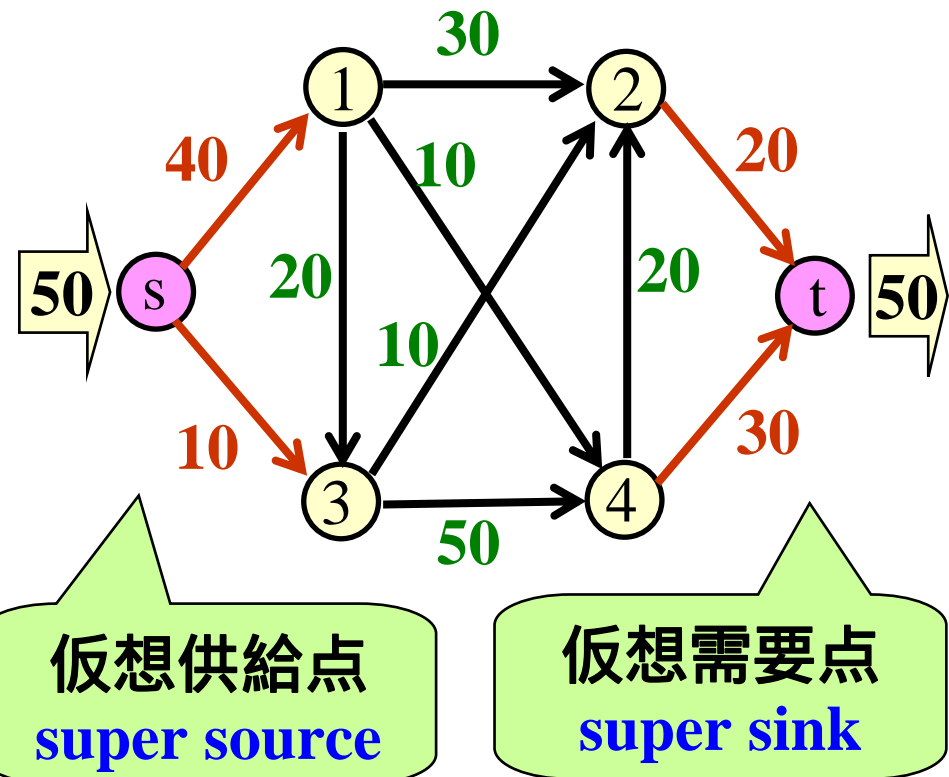
モノの流れをモデル化した
ネットワーク

仮定 (総供給量) = (総需要量)



変換 2端子ネットワーク

需要・供給点が各1つ



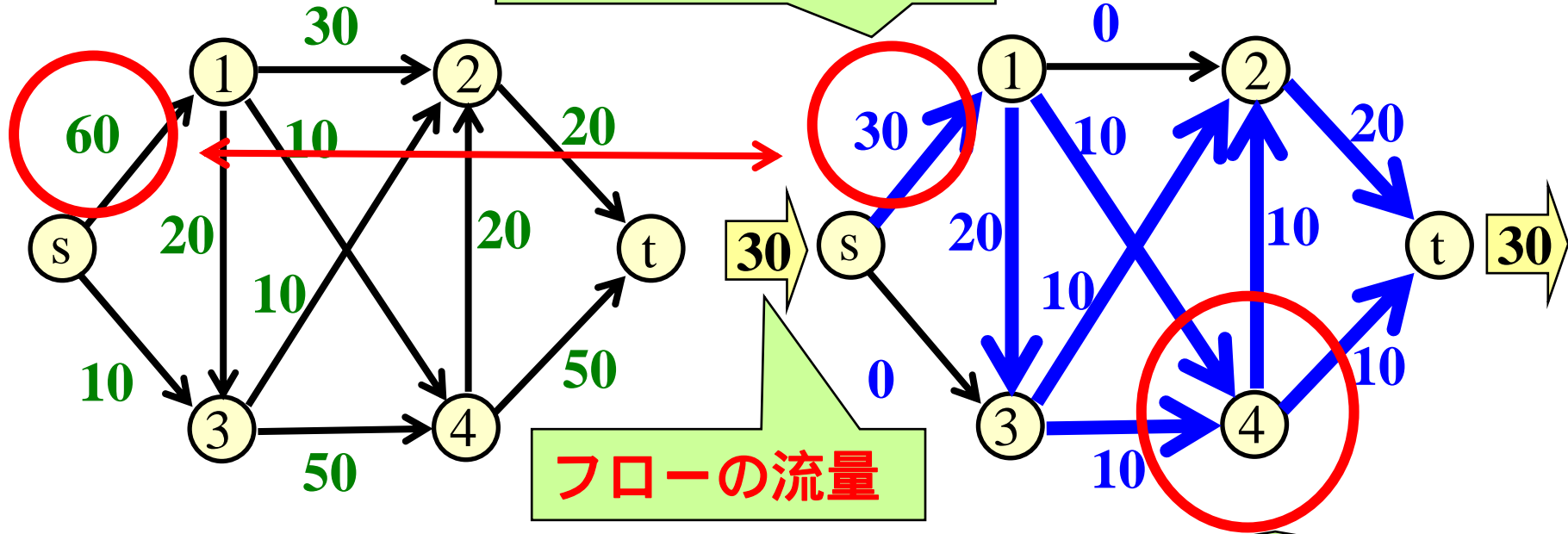
フロー

容量条件と流量保存条件の両方を満たす枝に付いた数字

ネットワーク

容量条件
フローは容量以下

フロー



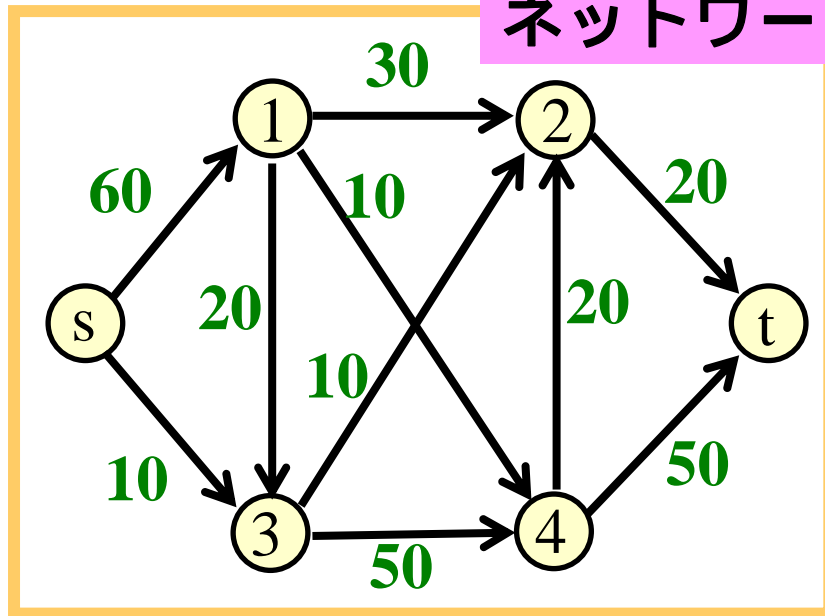
一般化
フロー

条件緩和
多少増減可

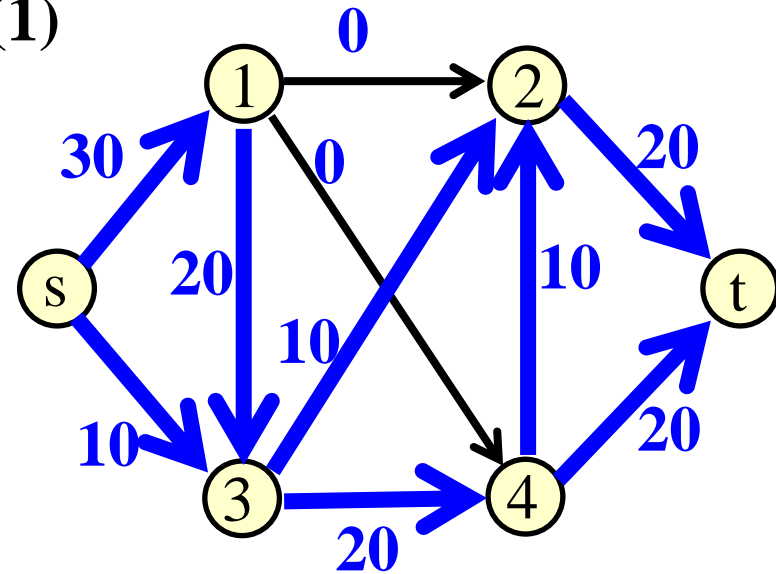
流量保存条件
s,t以外の全点で,
(流入フロー合計) = (流出フロー合計)

練習 フローはどれ？

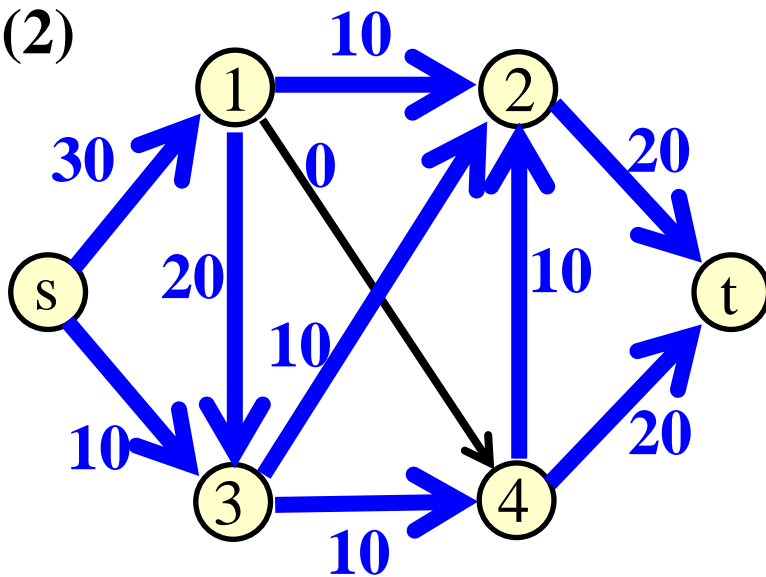
ネットワーク



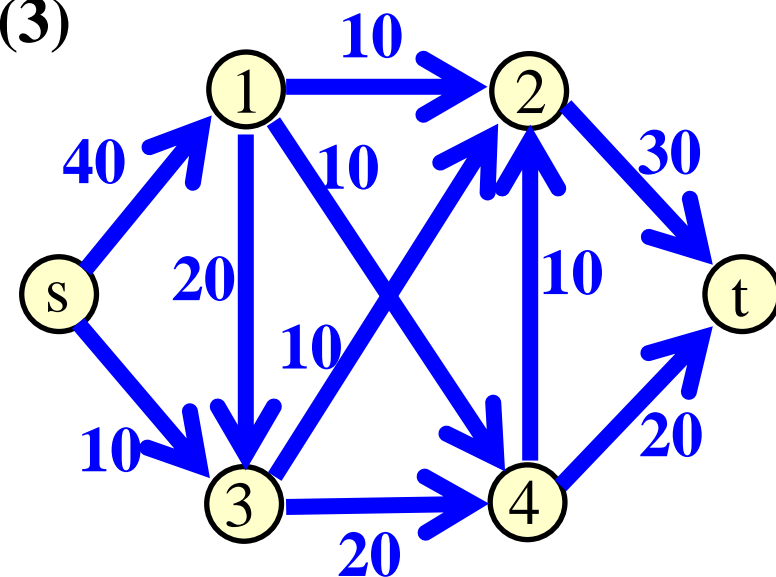
(1)



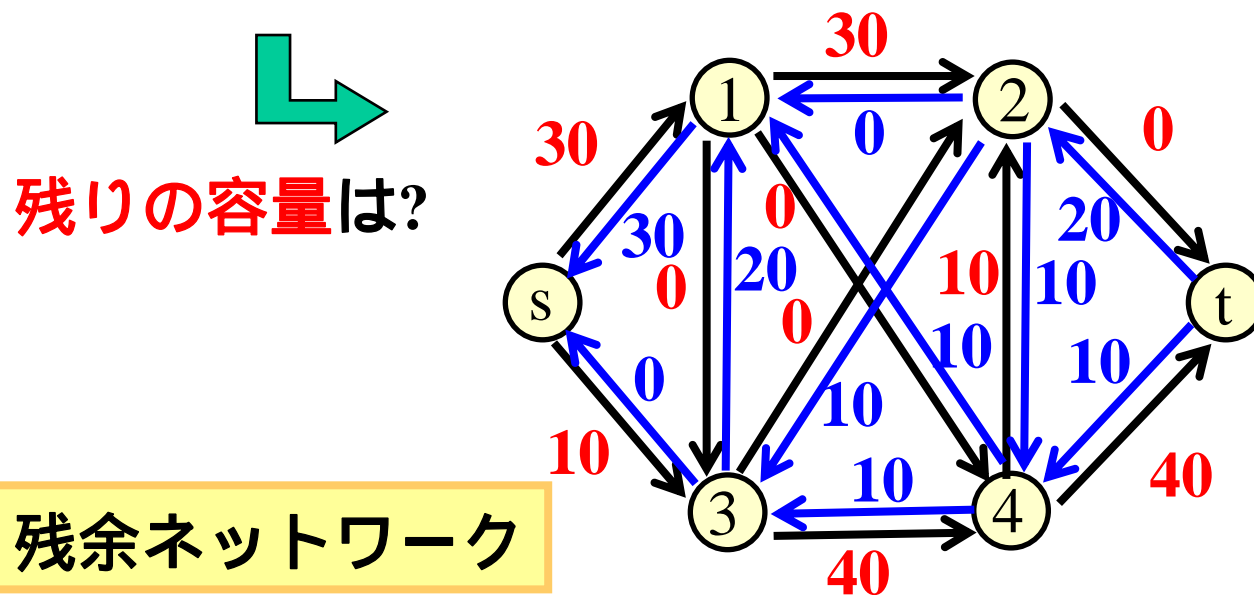
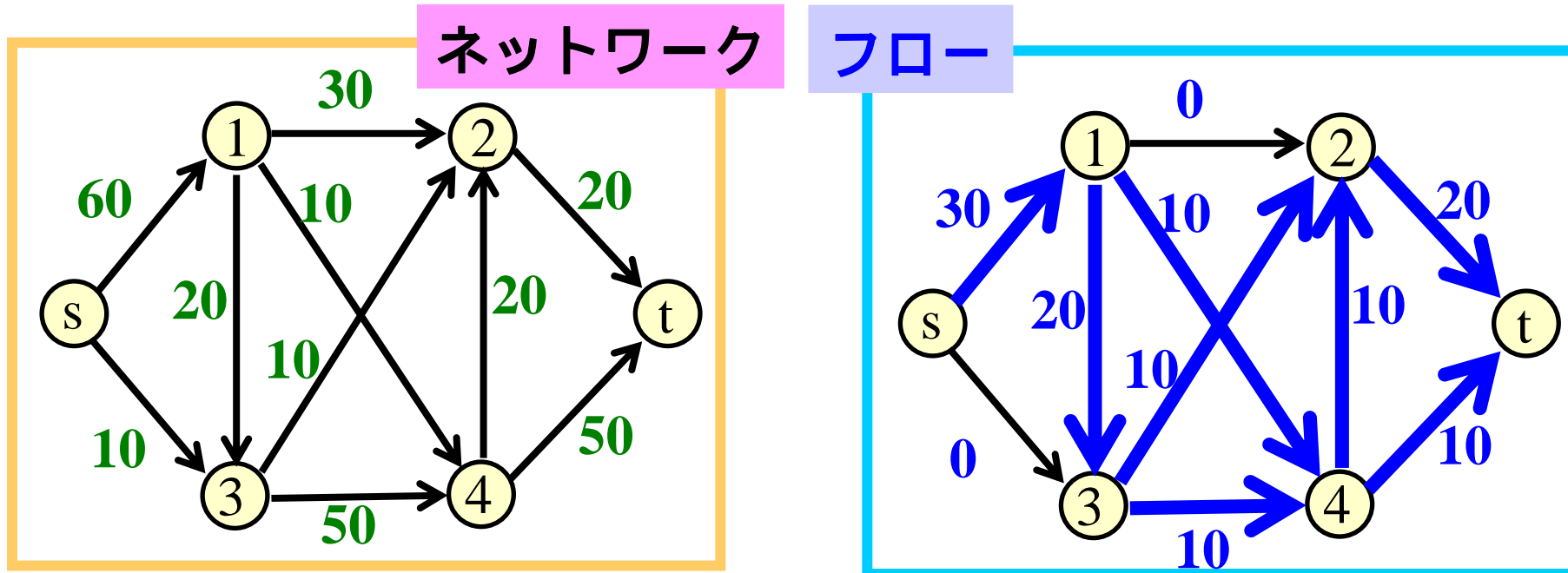
(2)



(3)



容量とフローを同時に表現するアイデア



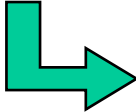
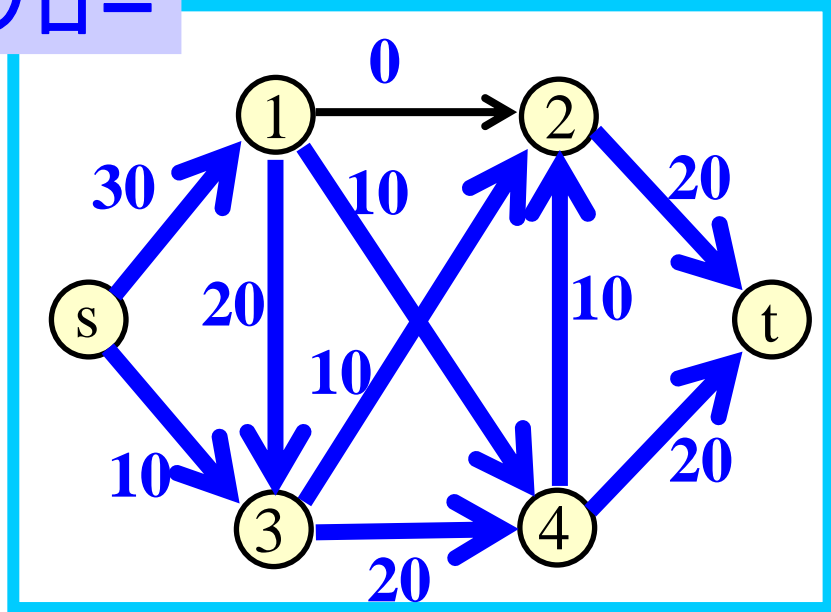
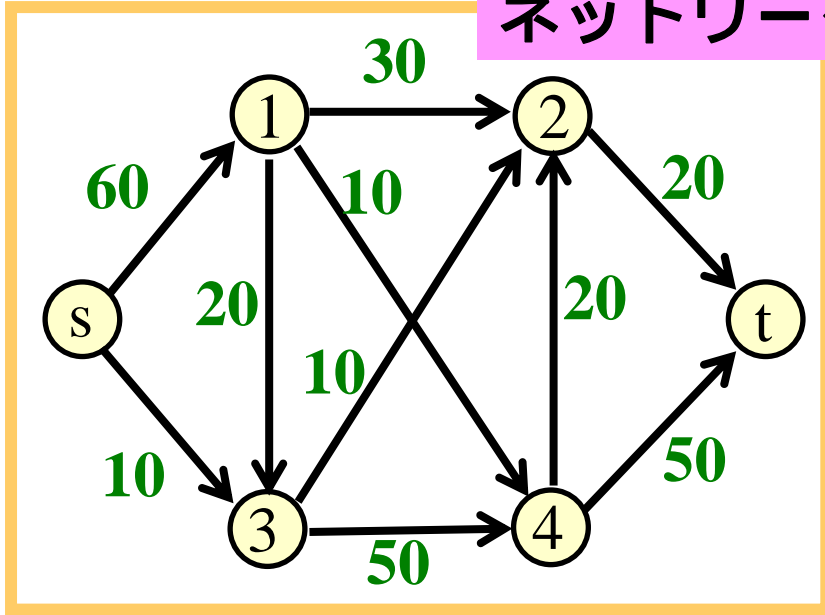
逆向き枝で
表現

0の枝は省略

練習 残余ネットワークで表現してみよう

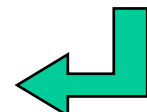
ネットワーク

フロー



①

②



s

t

残余ネットワーク

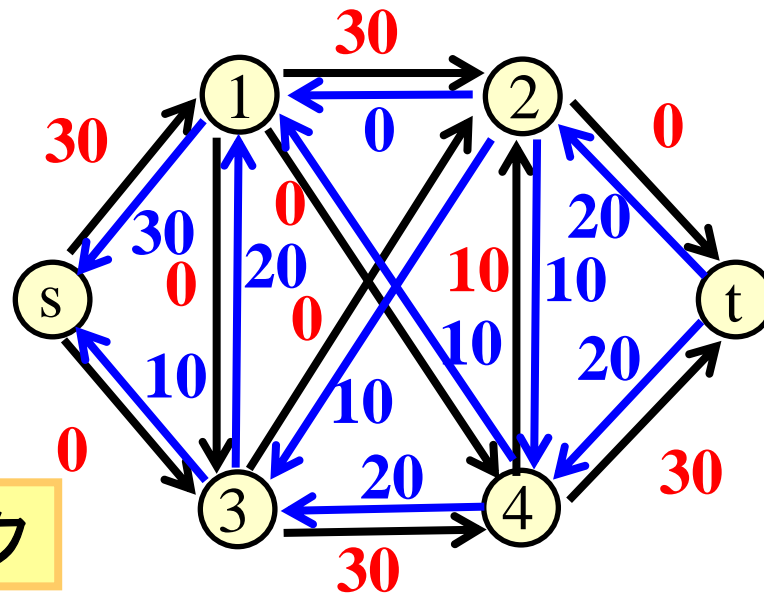
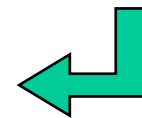
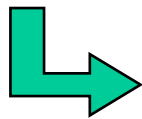
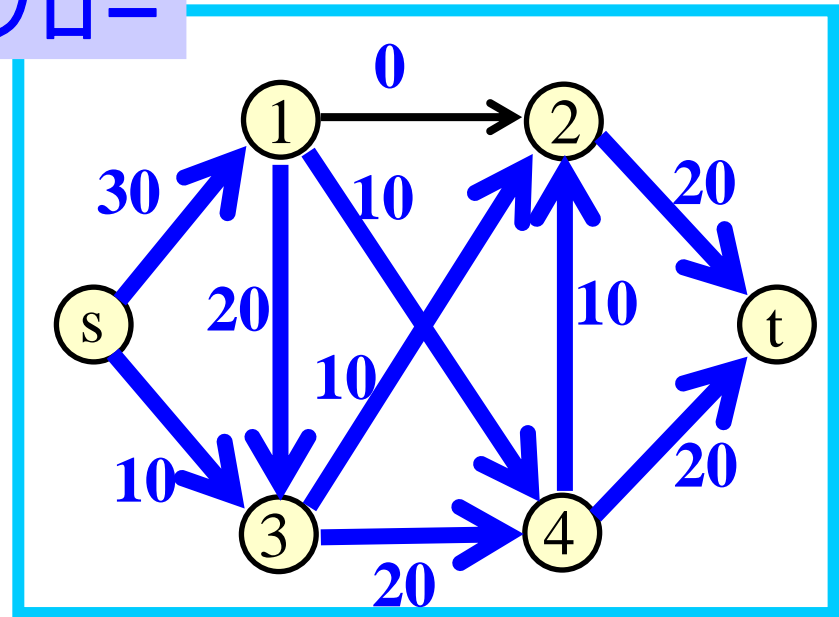
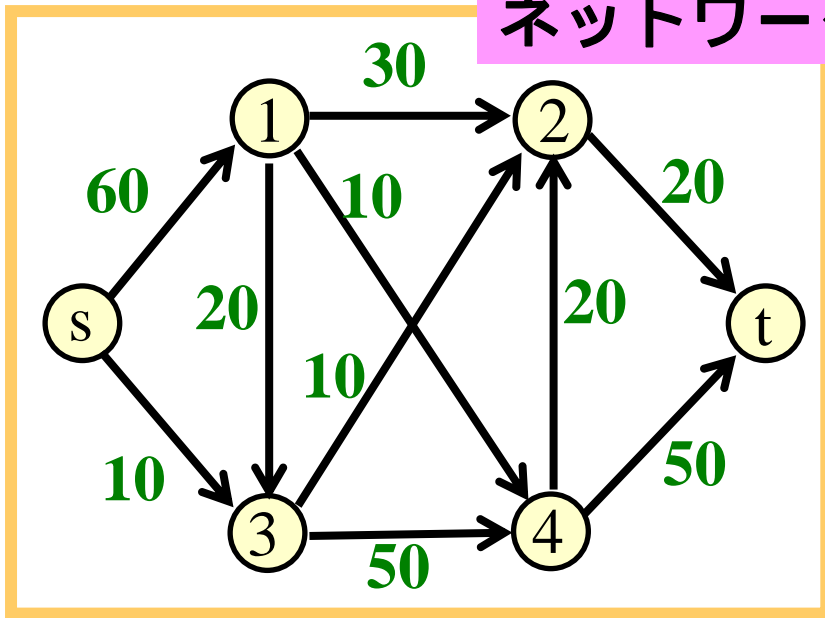
③

④

練習 解答例

ネットワーク

フロー

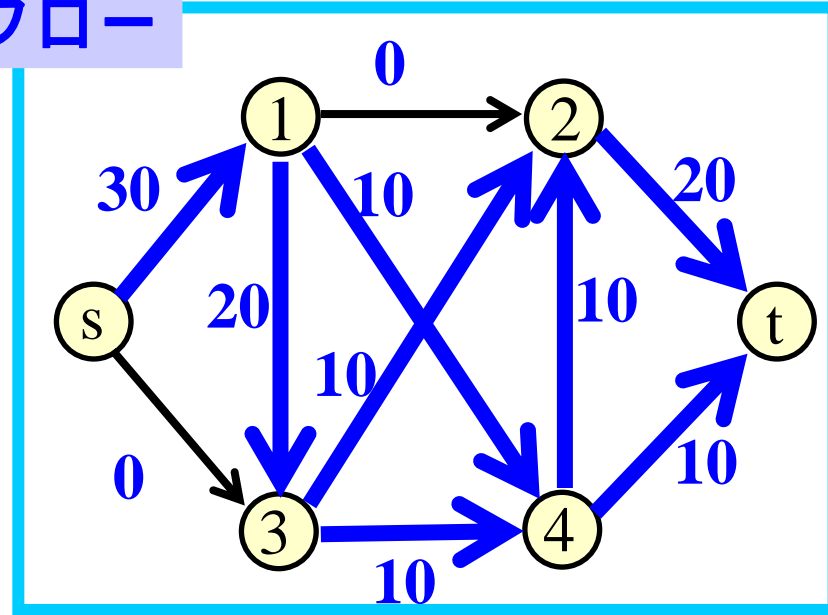
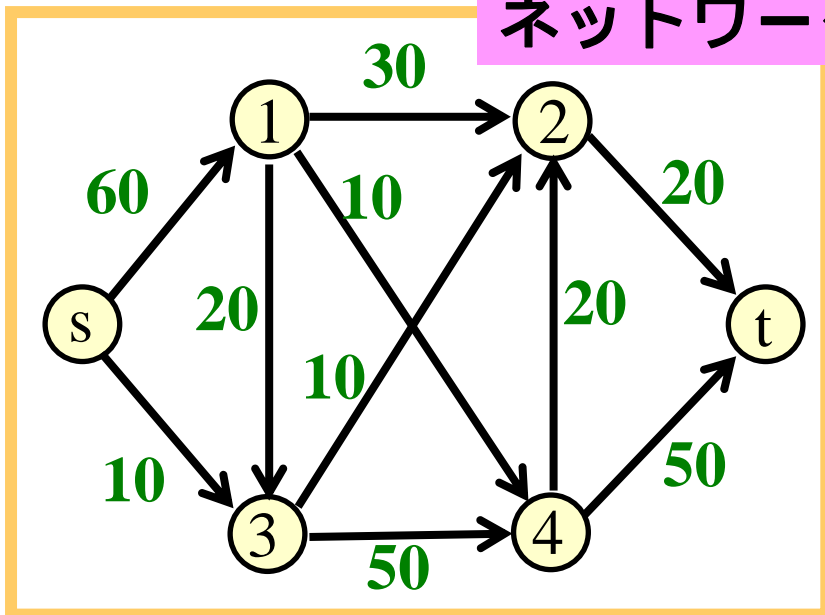


残余ネットワーク

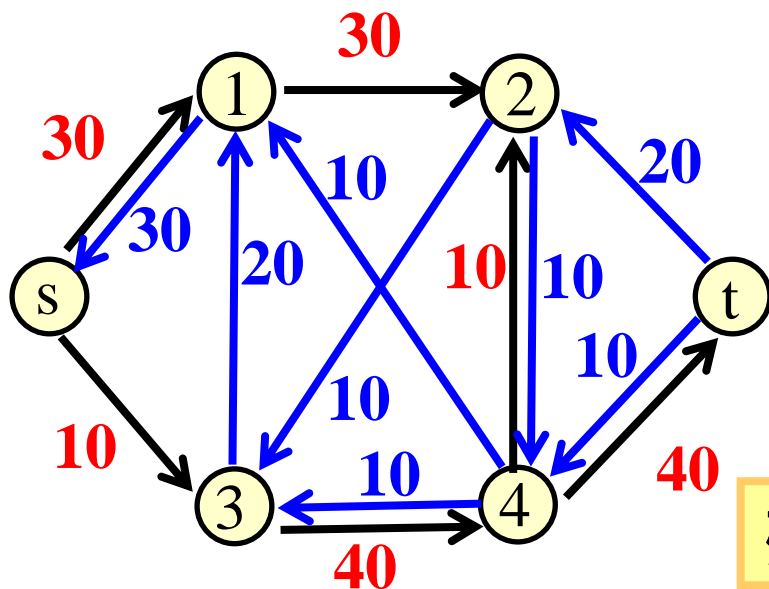
残余ネットワーク表現の利点

ネットワーク

フロー



質問：フローの流量を増やす余地がある？

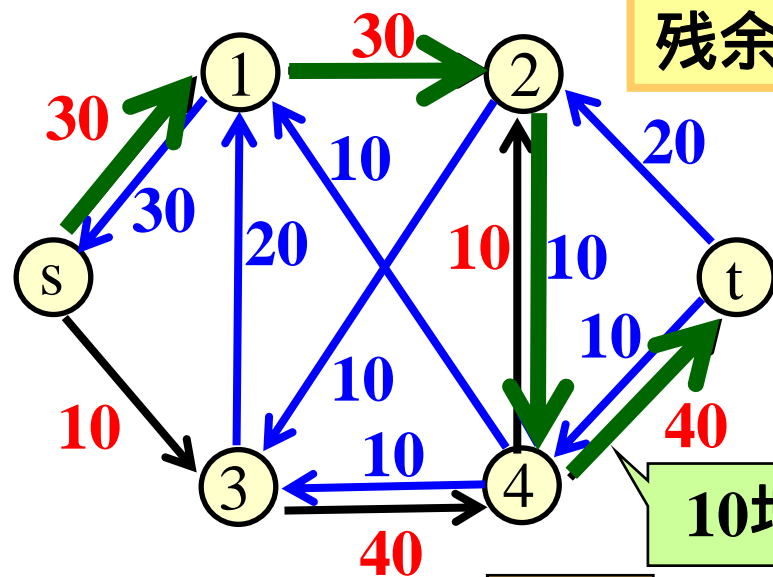


sからtへパスがある
余地がある

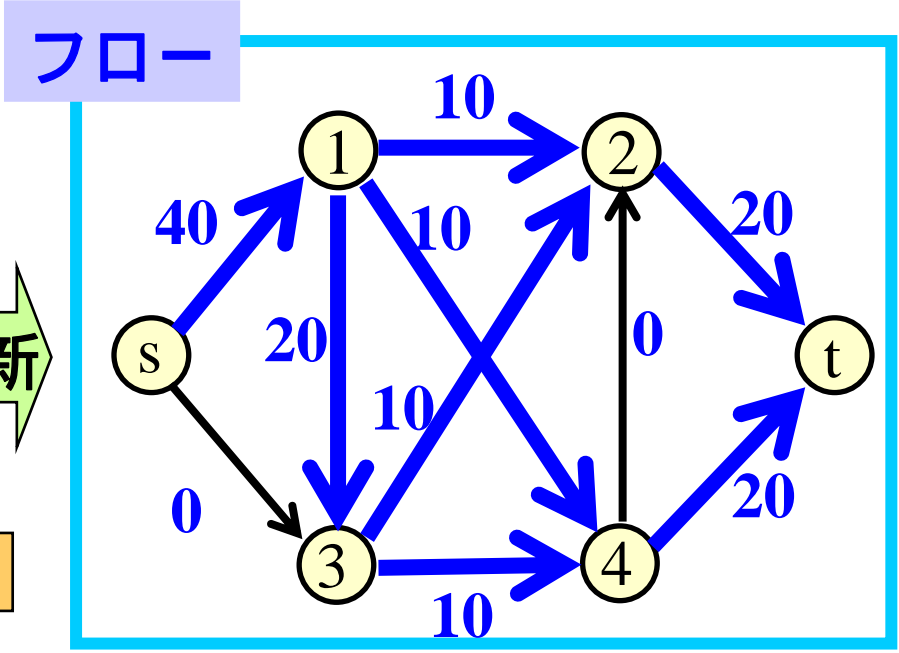
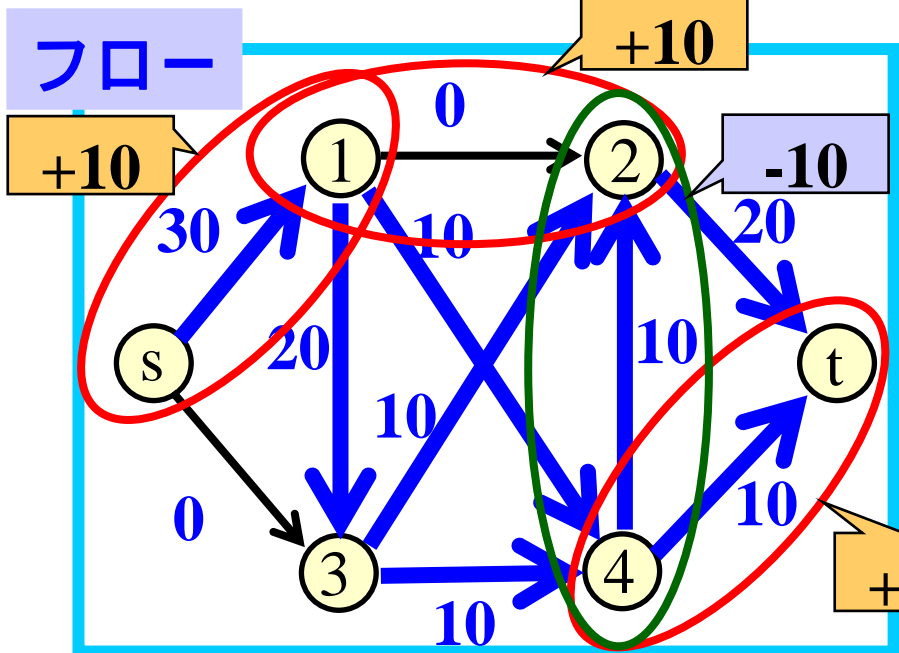
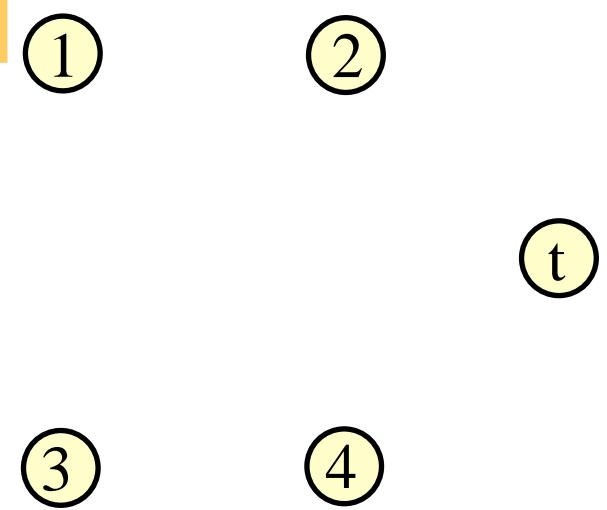
探索で
発見可

残余ネットワーク

練習 残余ネットワーク上でのフローの変化

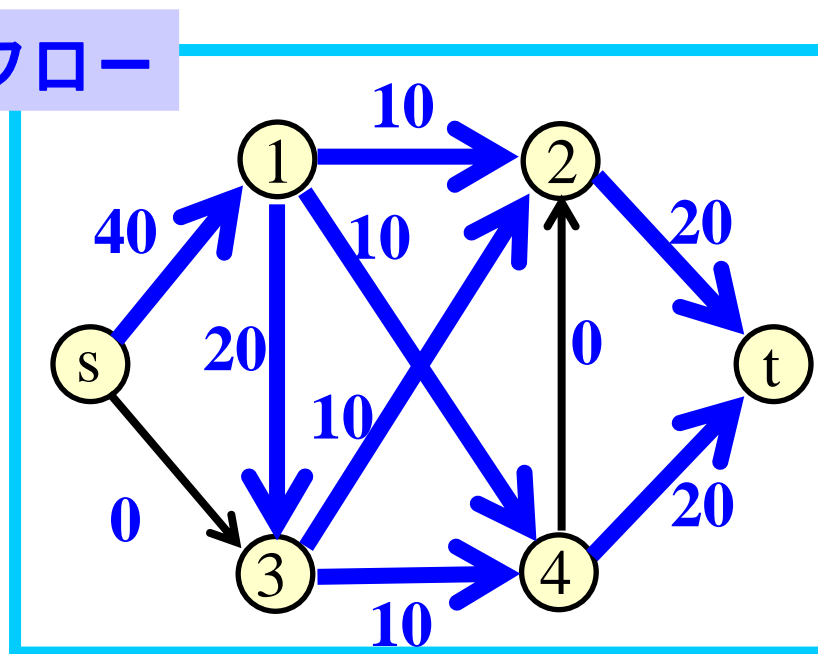
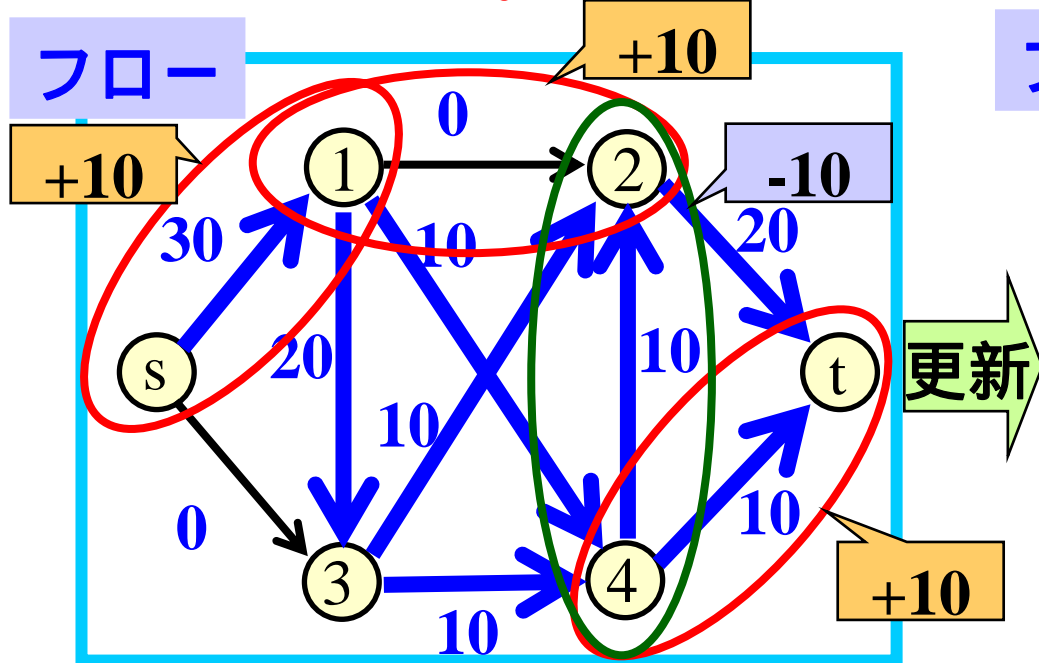
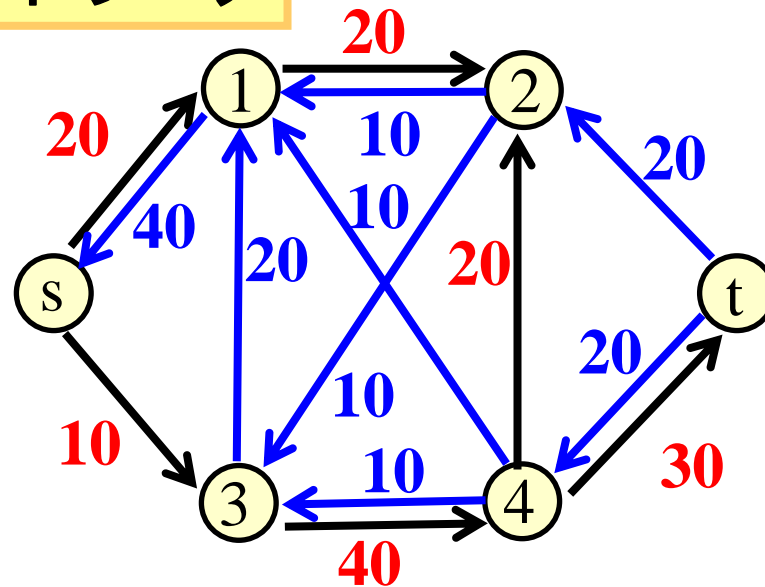
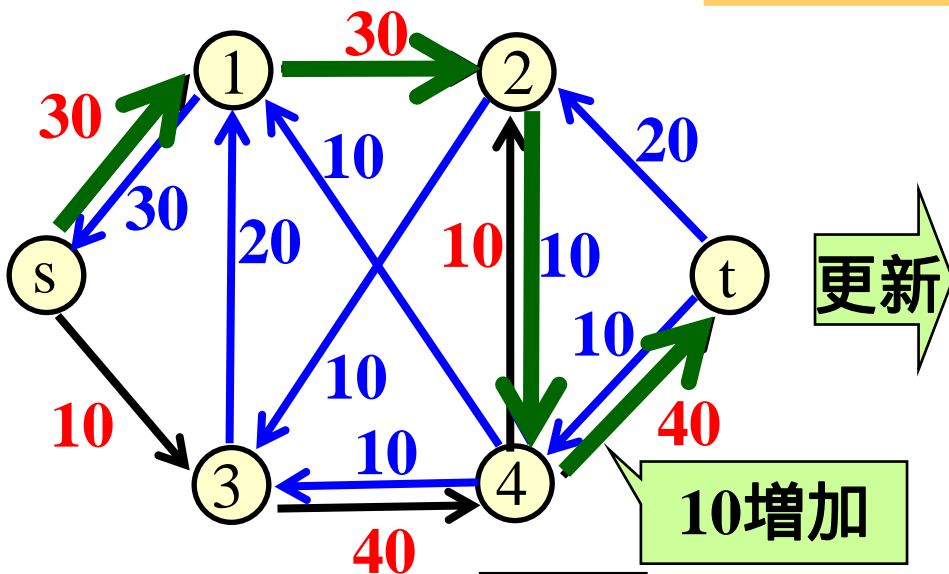


残余ネットワーク



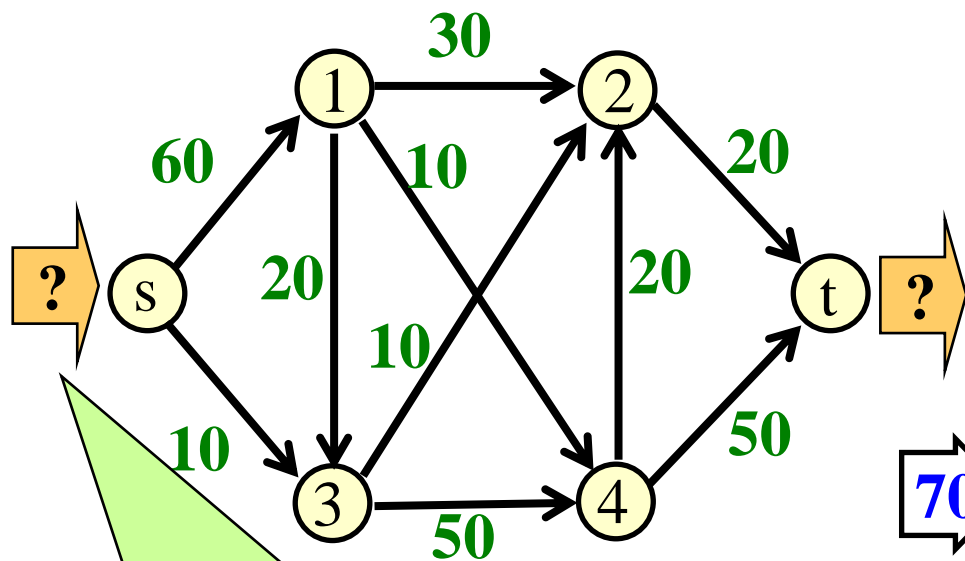
練習 解答例

残余ネットワーク



フローに関する基本的な最適化問題

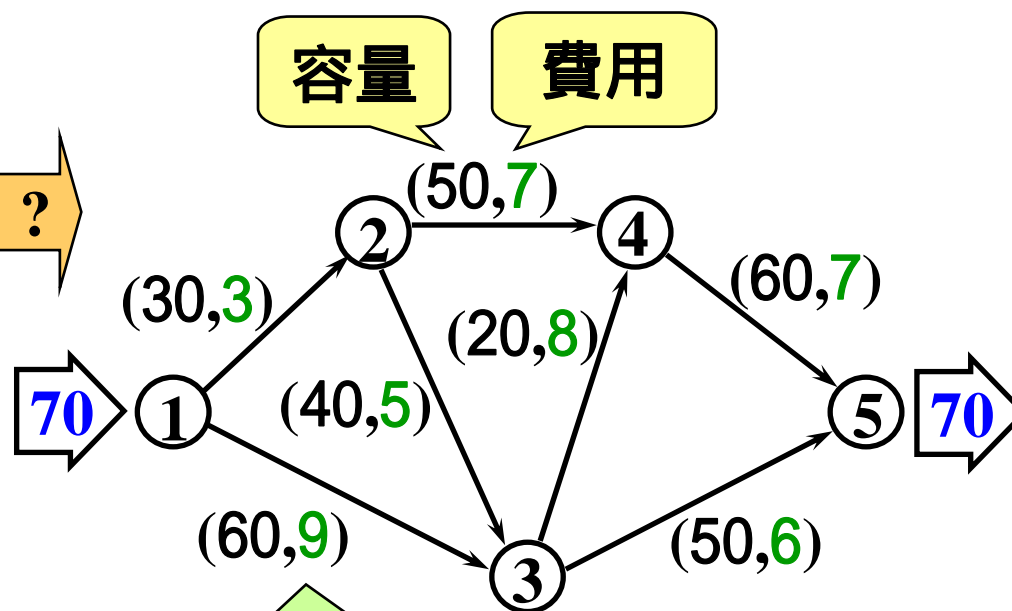
最大フロー問題



フロー流量の最大は?

ネットワークの能力を計りたい

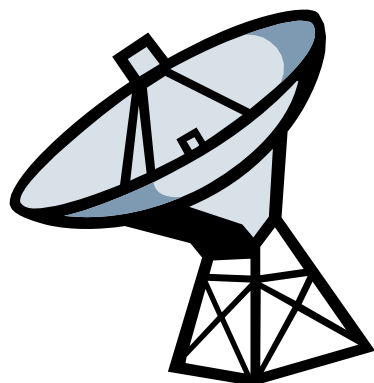
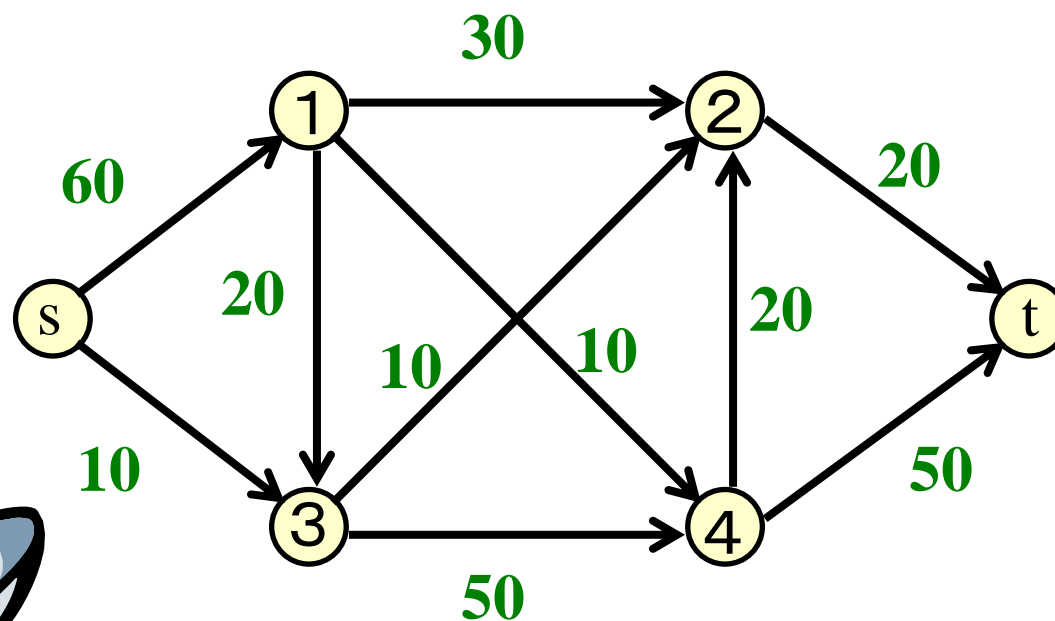
最小費用フロー問題



費用最小の流し方は?

例題1 最大フロー問題

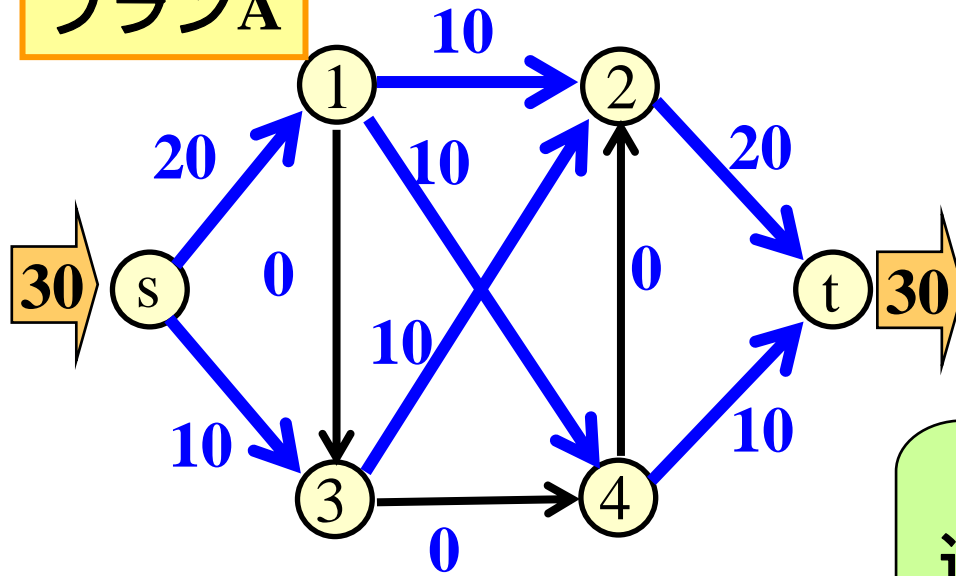
通信所sからtまで同時に送信できる最大データ量は？



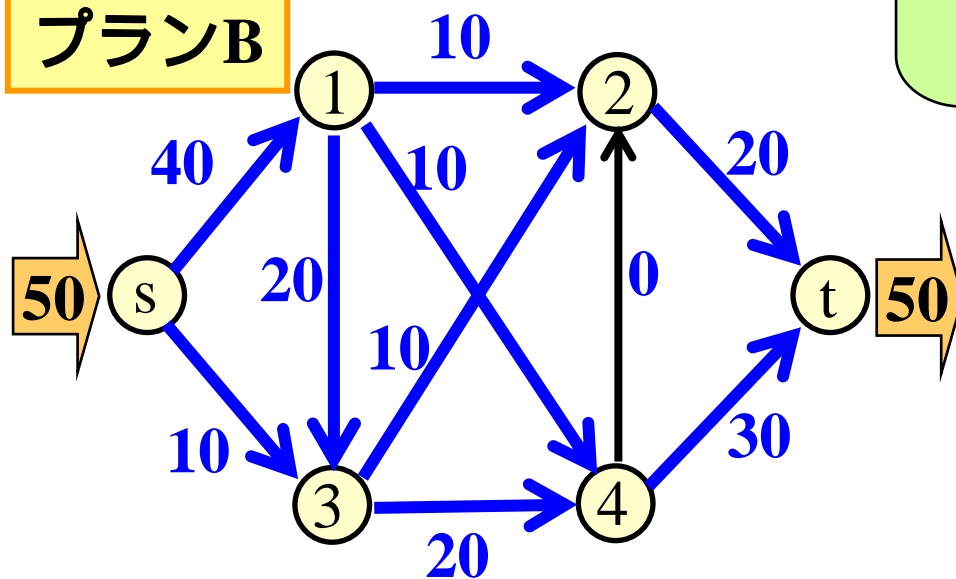
各枝の数字:送信可能容量

実行可能な通信計画

プランA



プランB



どちらが
送信データ
が多い？



最大フロー問題(定式化)



- **フローの流量:**
始点 s から流出するフローの和

目的 フローの流量→最大
条件 実行可能フロー



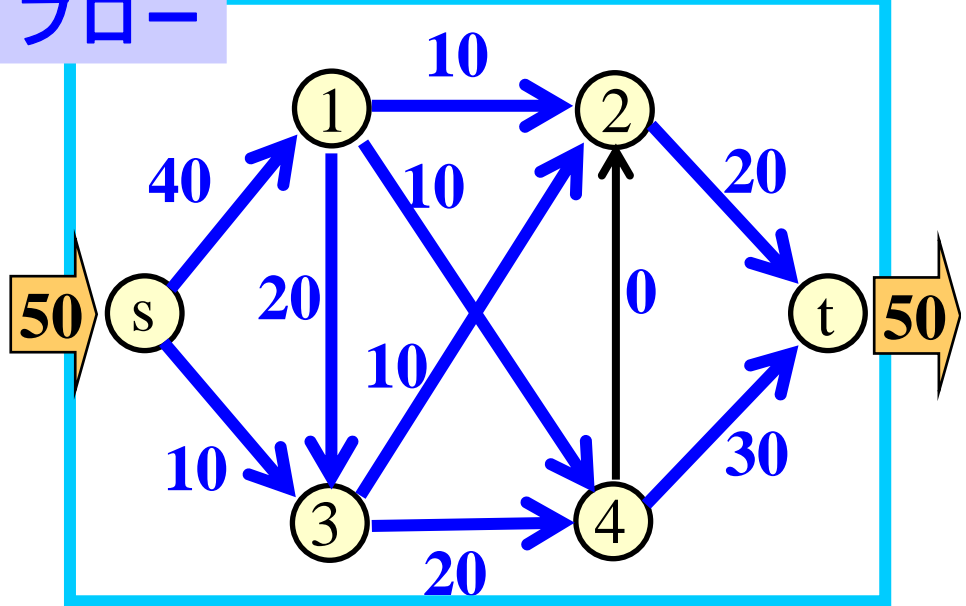
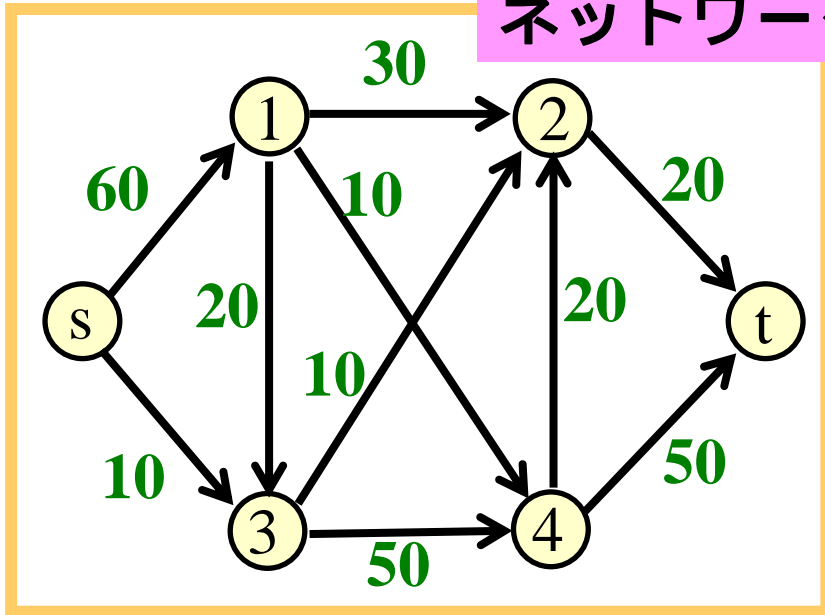
最大フロー問題

- **最大フロー:** フローの最大流量を達成する
実行可能フロー

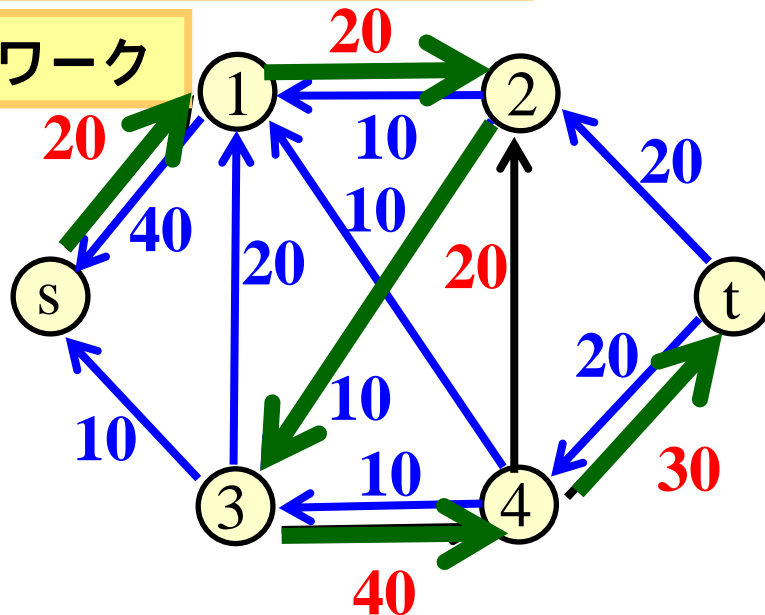
最大フロー?

ネットワーク

フロー



残余ネットワーク



残余ネットワーク上で
sからtへのパスが存在



最大フローではない

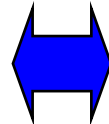
最大フローの求め方のアイデア

観察された事実

残余ネットワーク上で
 s から t へのパスが存在



最大フローではない



導かれる事実

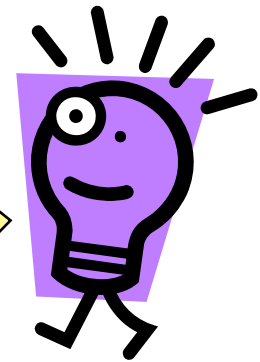
残余ネットワーク上で
 s から t へのパスが存在しない



最大フローである

増加道
(増加パス)

残余ネットワーク上で
 s から t へのパスがなくなるまで
フローを増やし続けよう

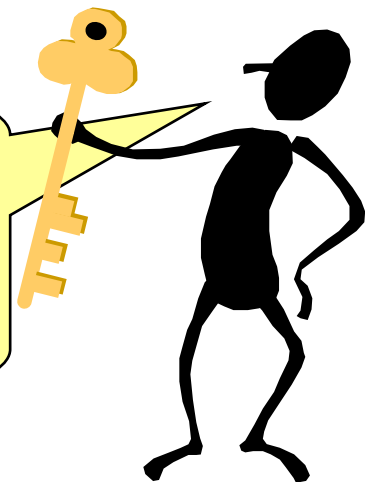


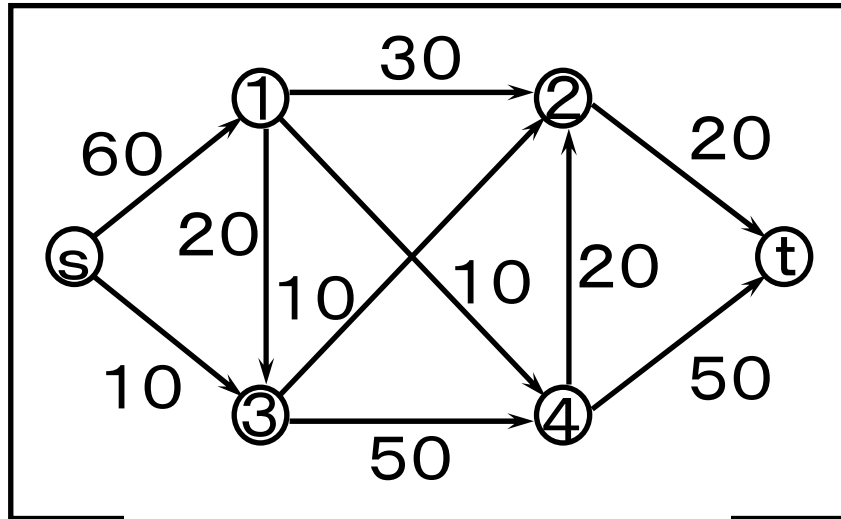
 **増加道法**

増加道法

- 手順1 各枝のフローを0にする.
- 手順2 **増加道**がある限り以下を繰り返す.
 - 増加道をひとつ見つける.
 - その増加道上の枝容量の最小値分のフローを、
残余ネットワーク上で増加道に沿って流す.

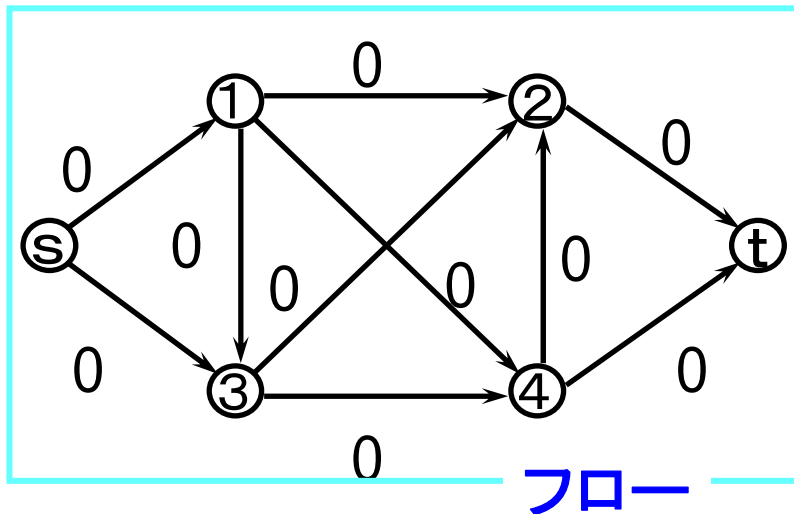
残余ネットワーク上で流すので、
実際のネットワーク上ではフロー
が減る枝も出てくることに注意！





容量付きネットワーク

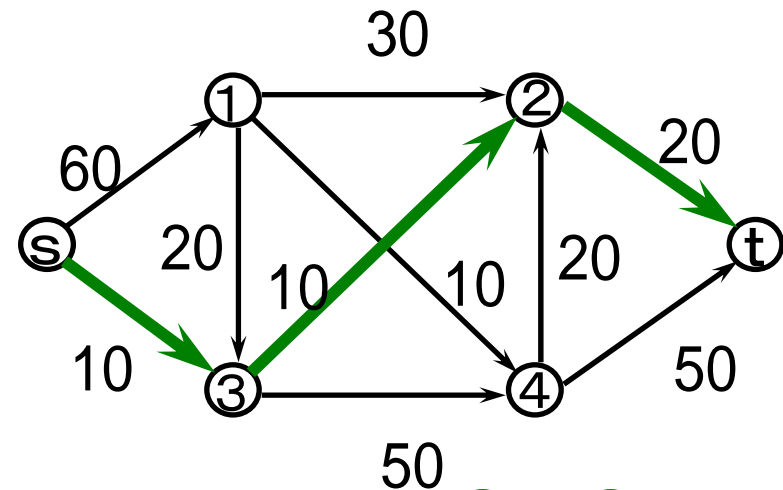
手順1



フロー

例題1(続) 増加道法

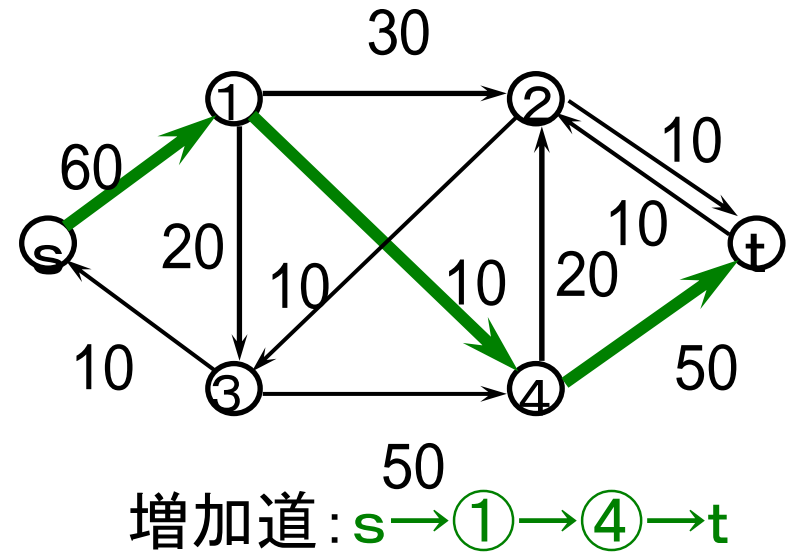
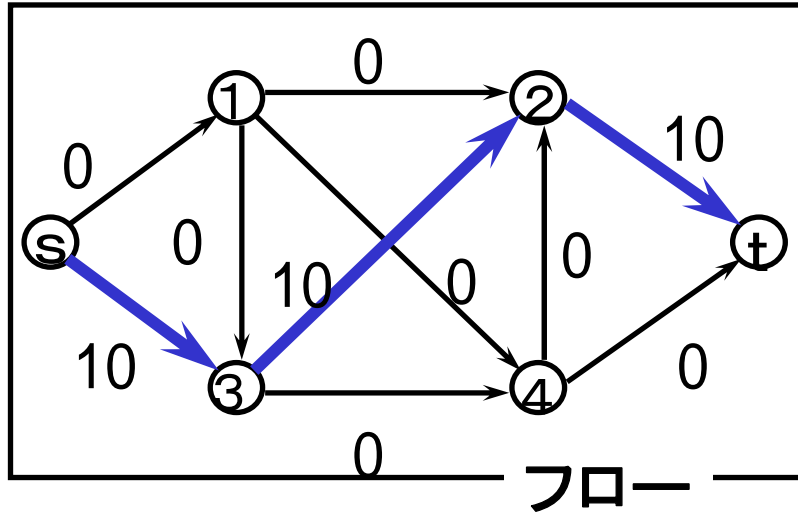
左側のフローに対する
残余ネットワーク



増加道: $s \rightarrow ③ \rightarrow ② \rightarrow t$

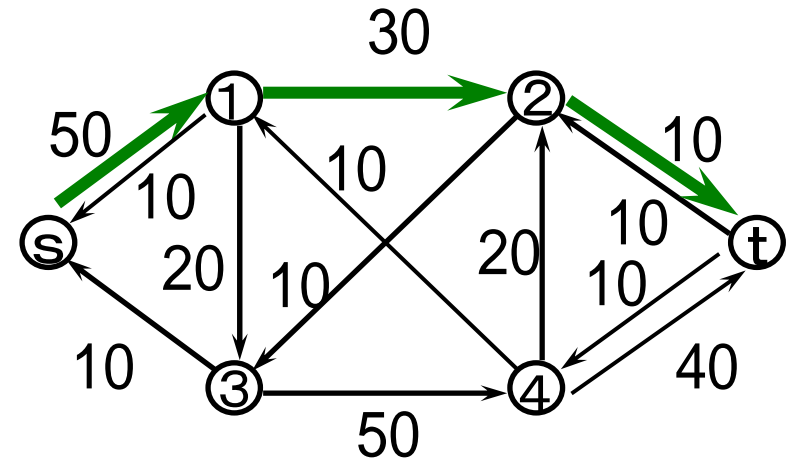
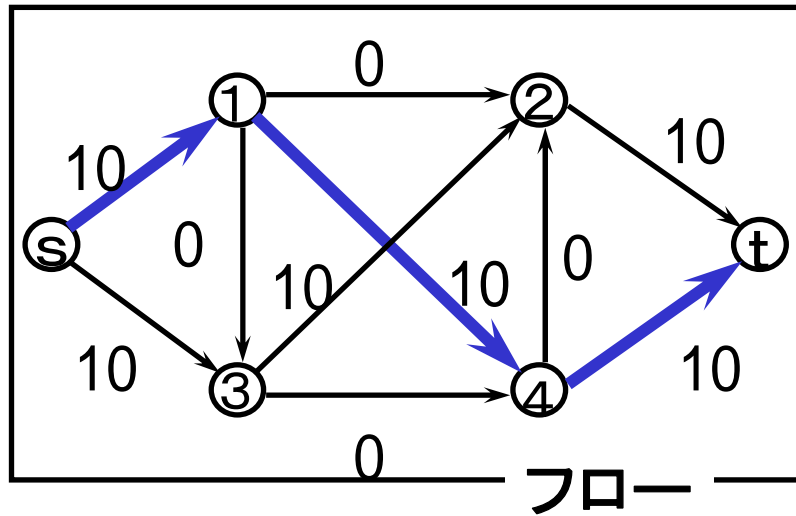
最小容量: 10

手順2 繰返し1回目



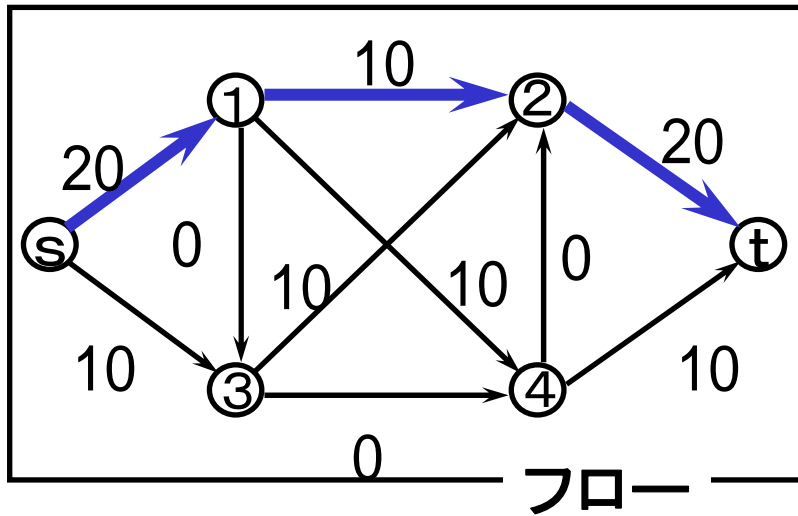
増加道: $s \rightarrow ① \rightarrow ④ \rightarrow t$
 最小容量: 10

手順2 繰返し2回目

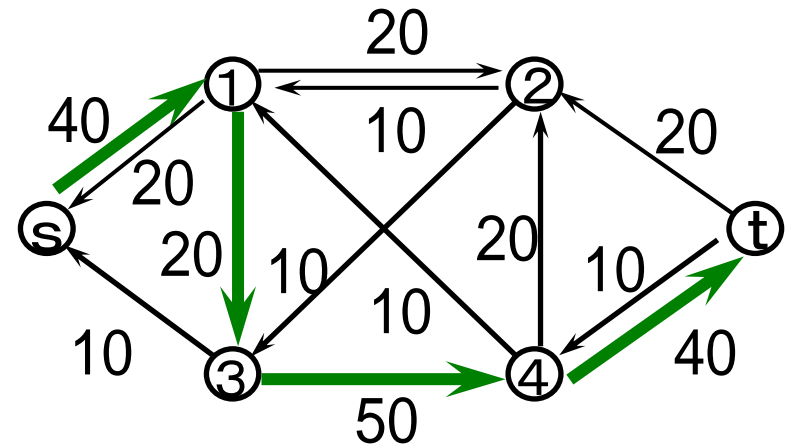
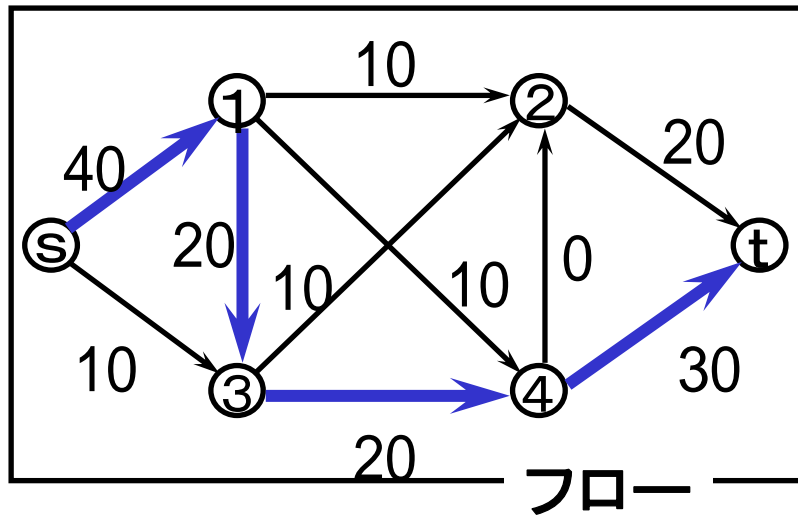


増加道: $s \rightarrow ① \rightarrow ② \rightarrow t$
 最小容量: 10

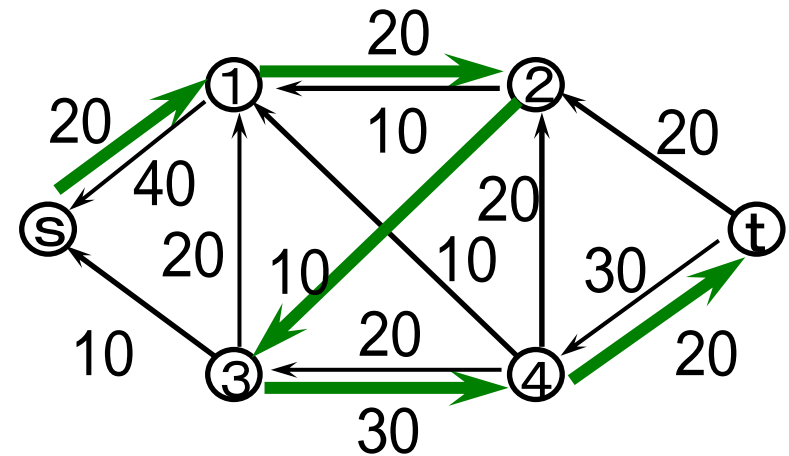
手順2 繰返し3回目



手順2 繰返し4回目

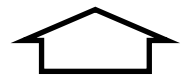
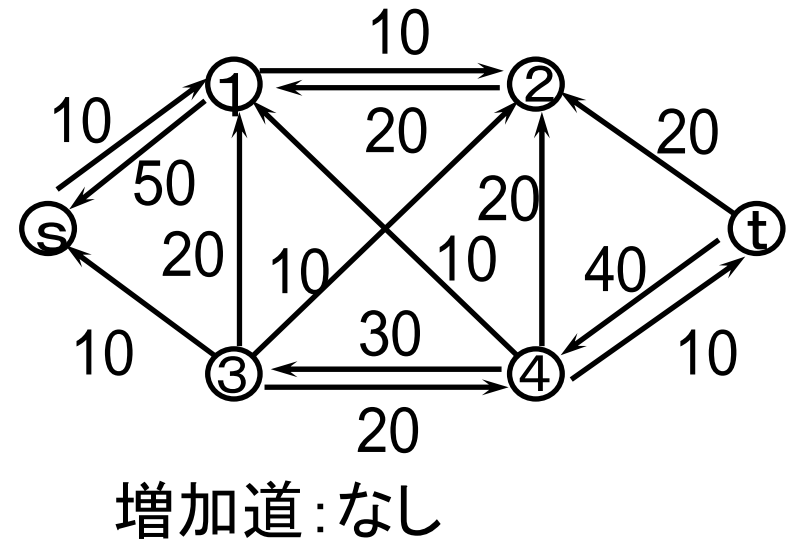
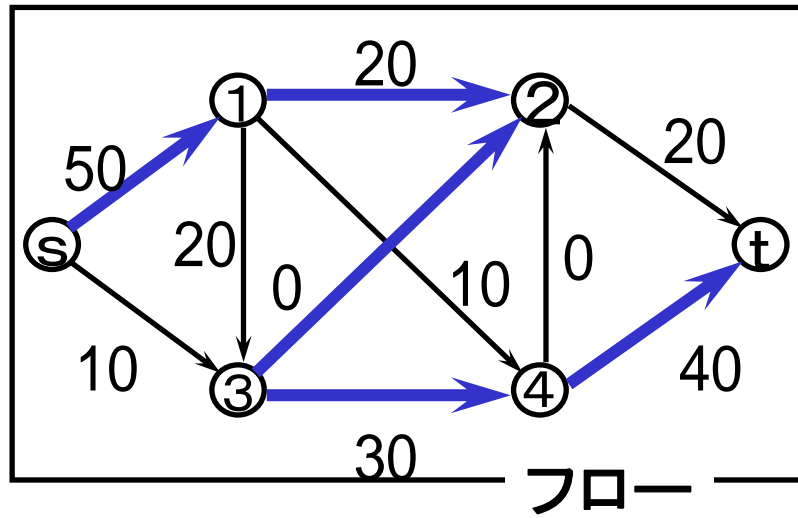


増加道: $s \rightarrow ① \rightarrow ③ \rightarrow ④ \rightarrow t$
 最小容量: 20



増加道: $s \rightarrow ① \rightarrow ② \rightarrow ③ \rightarrow ④ \rightarrow t$
 最小容量: 10

手順2 繰返し5回目

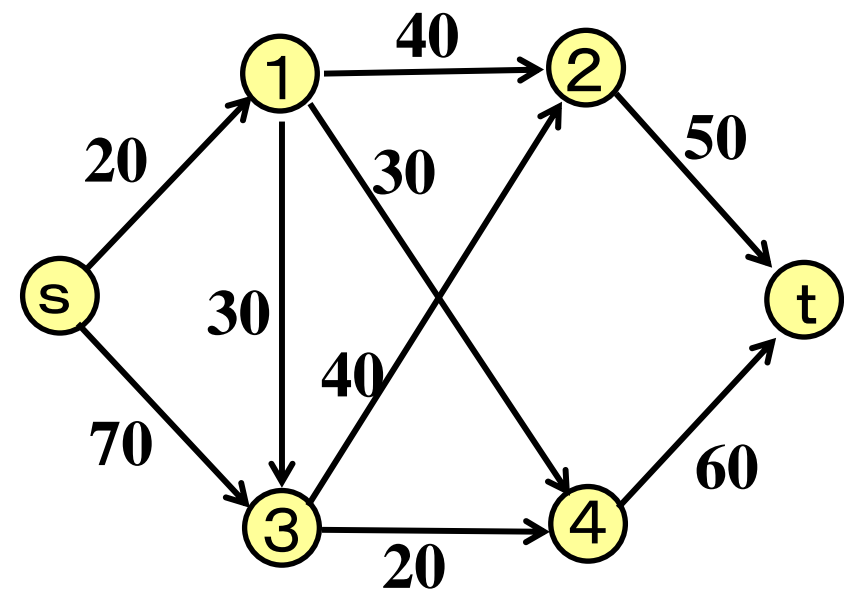


最大フロー
流量:60

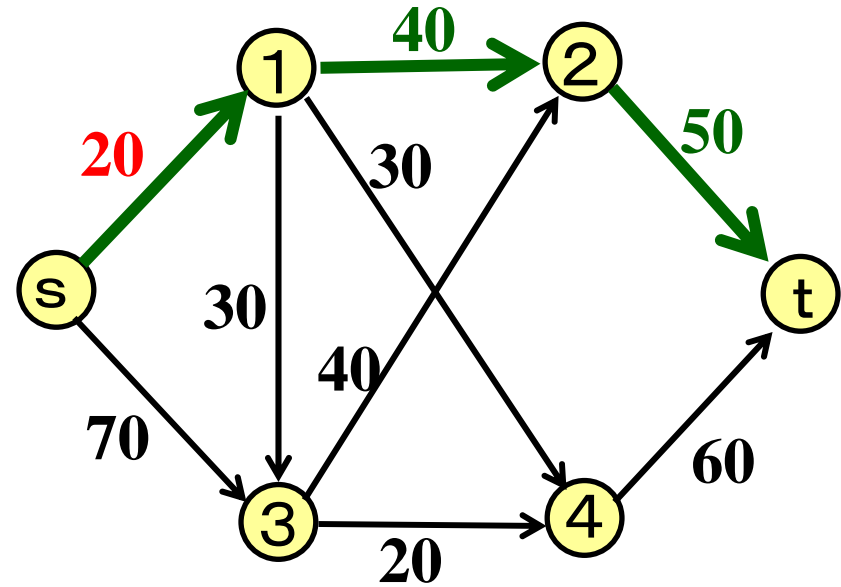
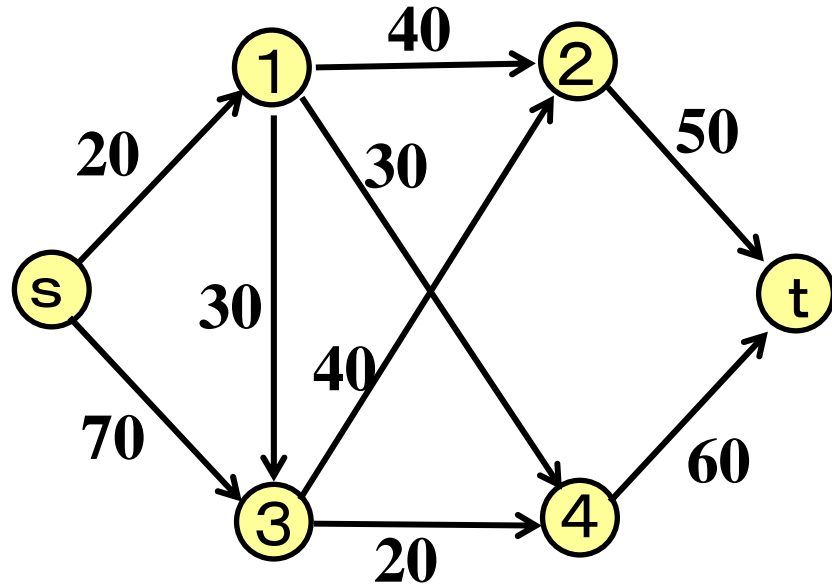


練習 増加道法

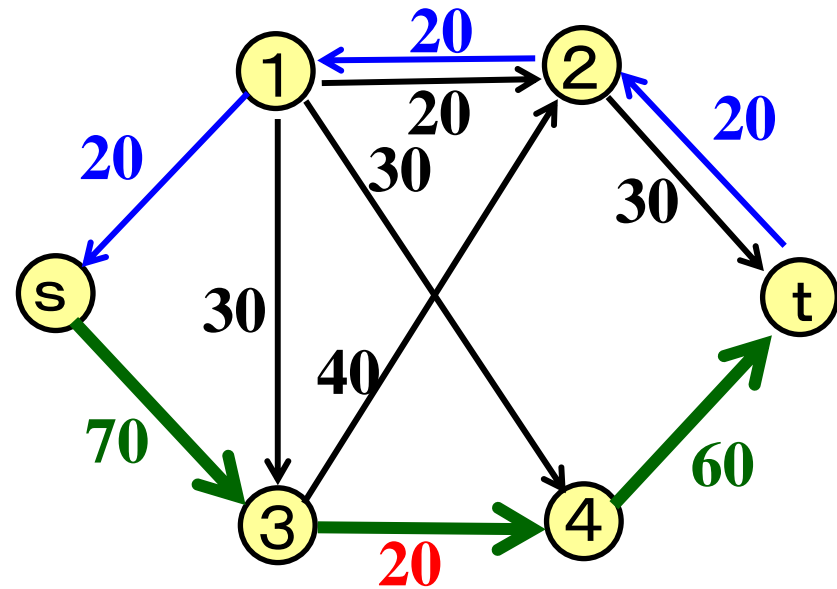
増加道法で最大フローを求めよ



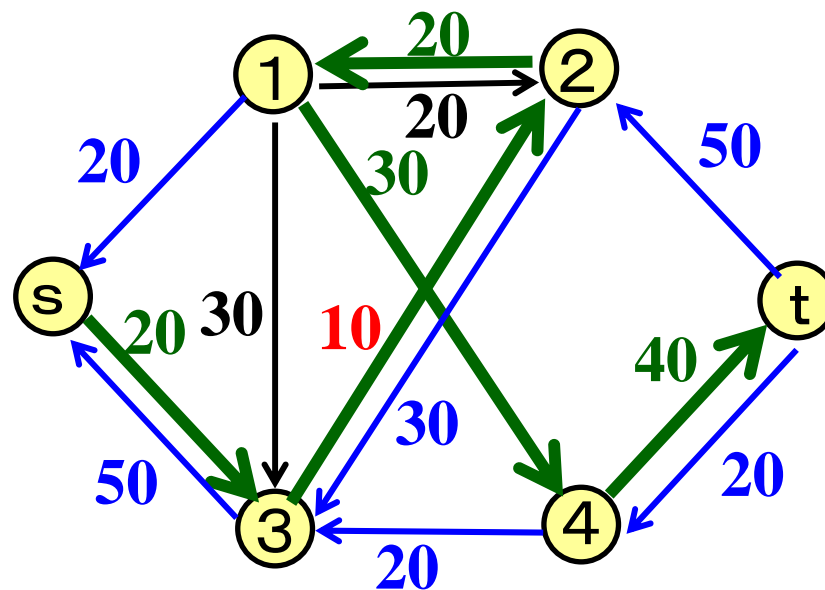
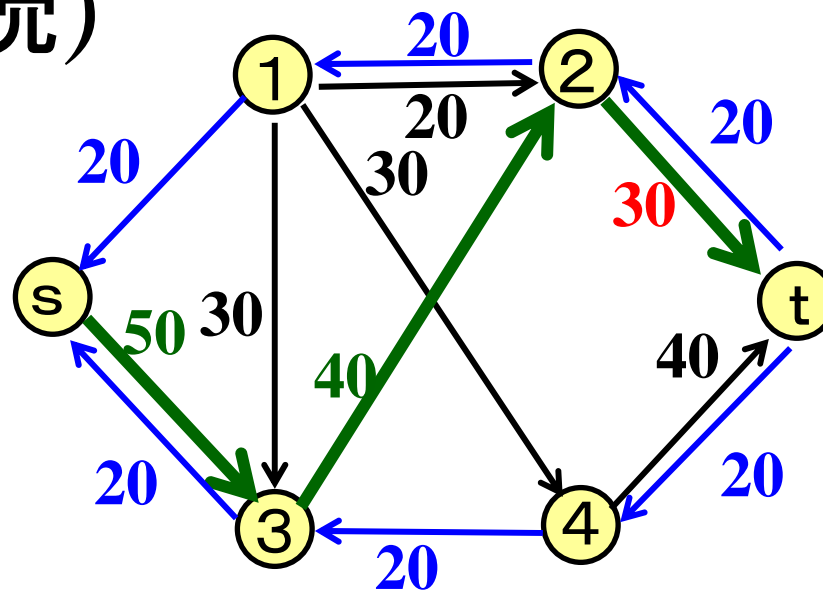
練習 解答例



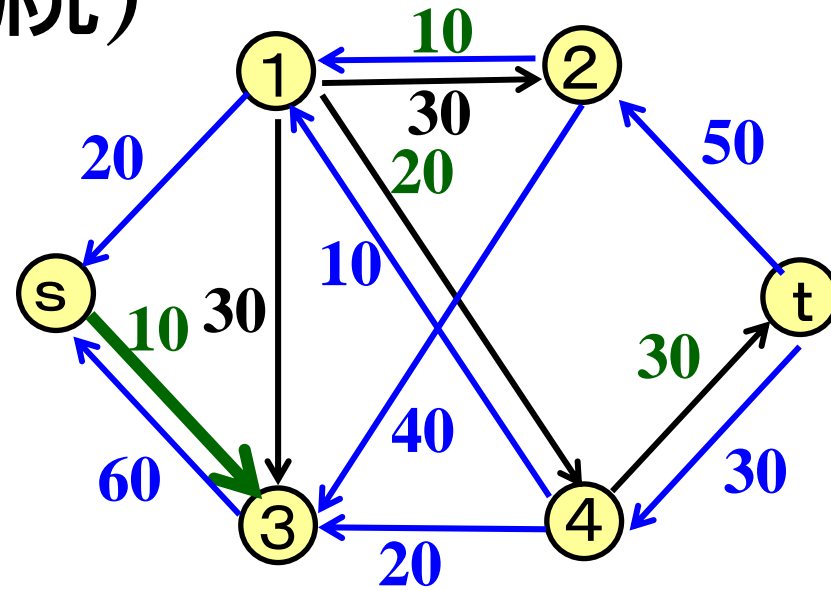
初期の残余ネットワーク
(フローの流量は0)



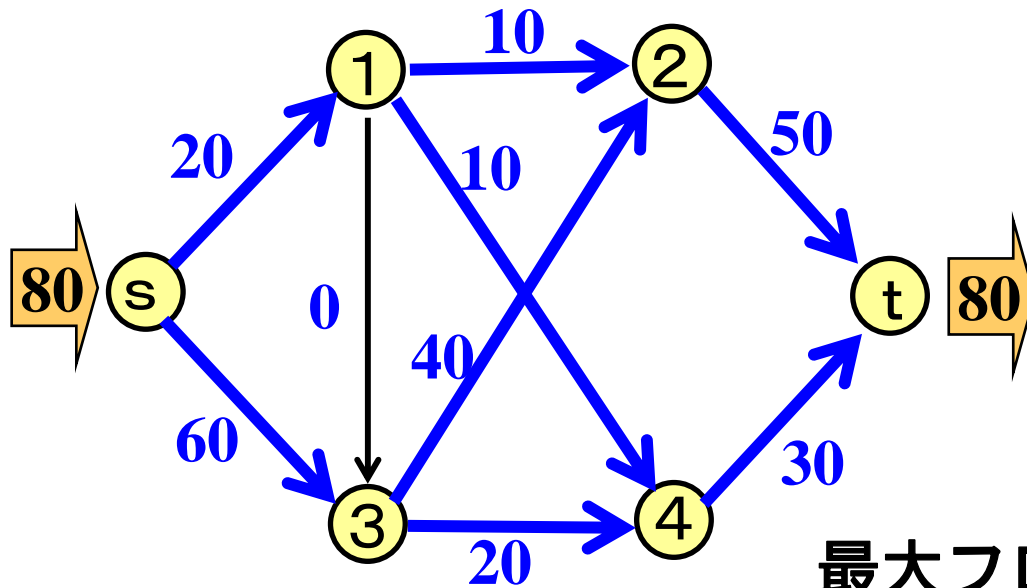
練習 解答例(続)



練習 解答例(続)



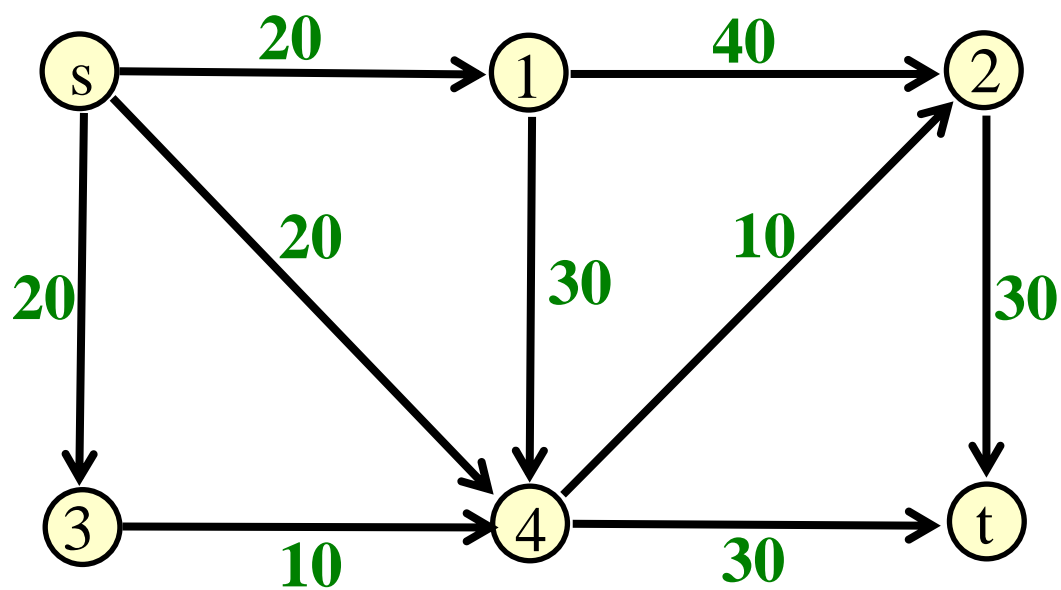
最大フロー



増加道がない

最大フローの流量は80

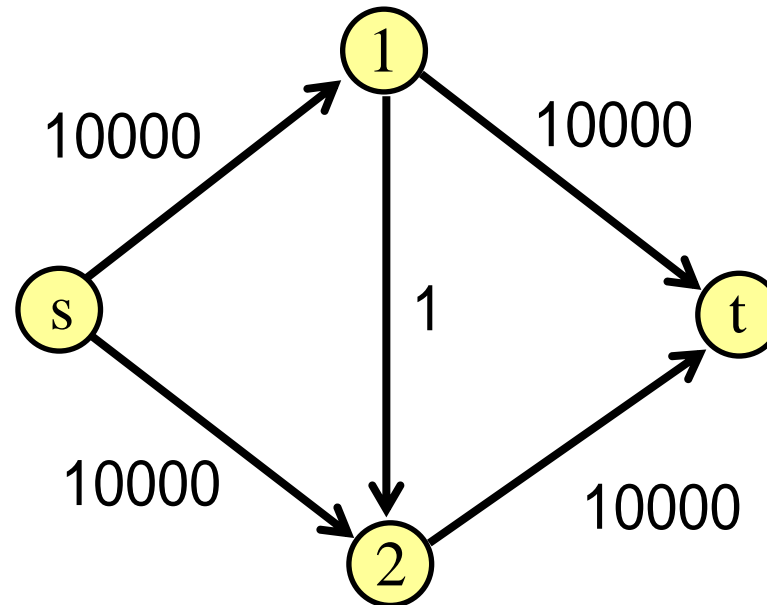
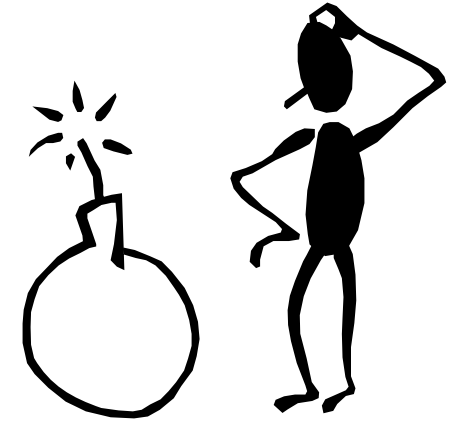
演習1 増加道法



最大フローとその流量を求めよ

増加道法の欠点

sからtへの最大フローを求めよ.
手順2を何回繰り返す？



演習2 増加道法を改良せよ

最大フロー問題の二大基本解法

- **増加道法** (Ford-Fulkerson)

- 簡単な手順の繰り返し. 直感的に妥当性が理解できる. 計算時間が多くかかる.



改良: Dinicの解法

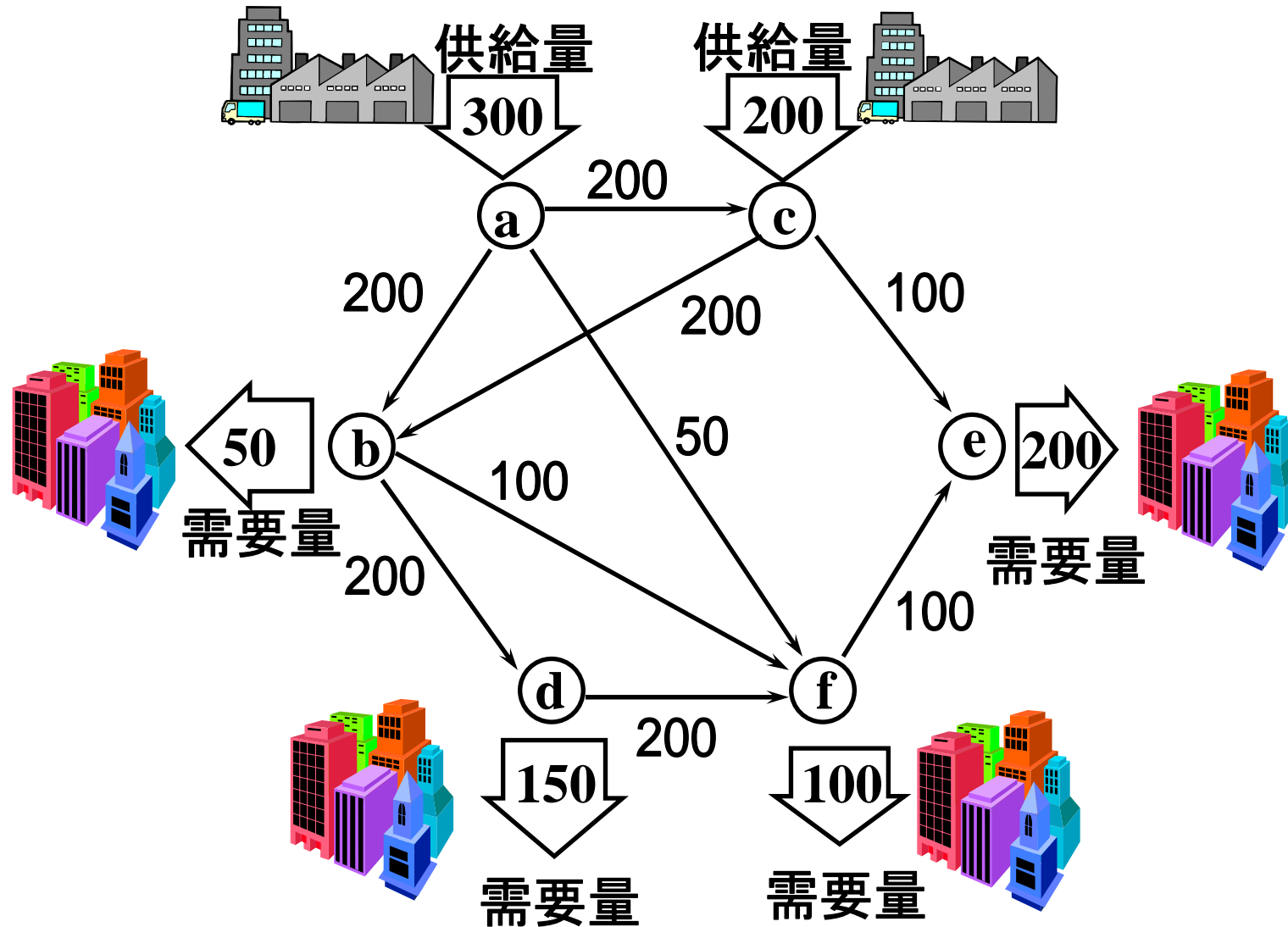
- **Preflow-Push**解法

- 工夫を加えることで高速に最大フローを求める.



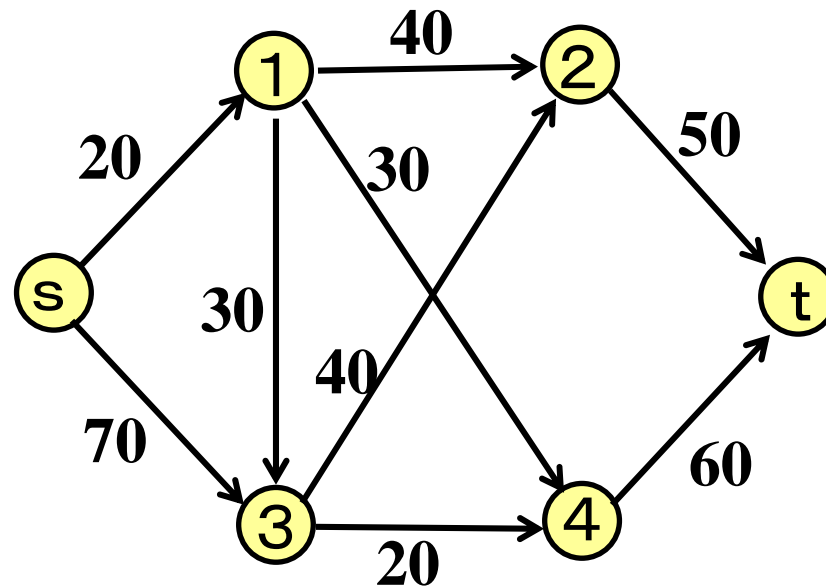
仮定: ネットワークの容量は整数で与えられる.

演習3 需要を満たすことは可能？



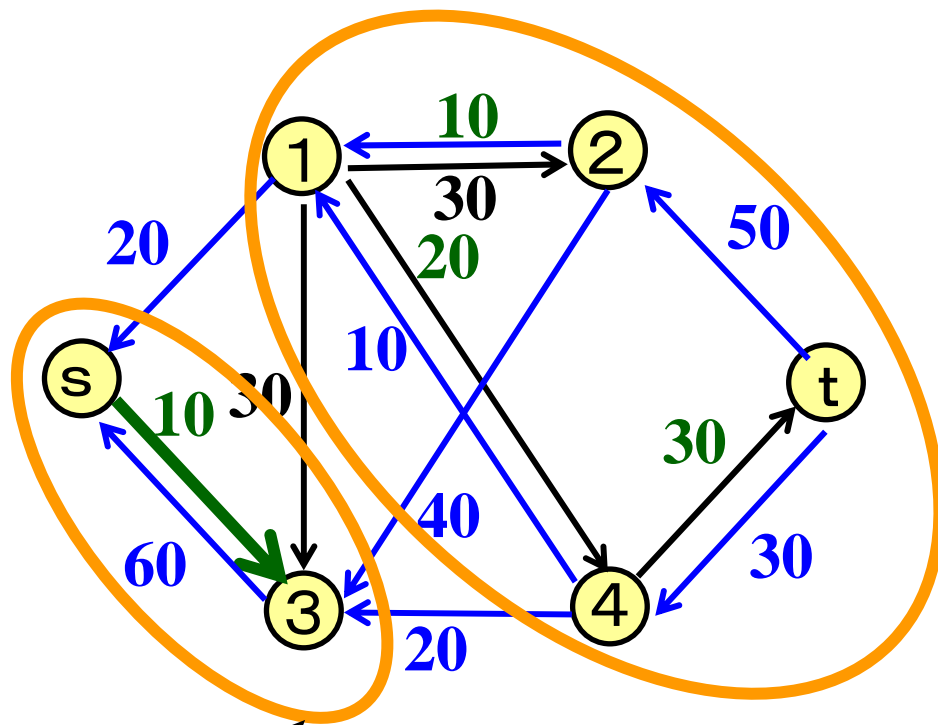
例題2

sからtへより多くのモノを流したい。
どの枝の容量を増やすのが効果的?



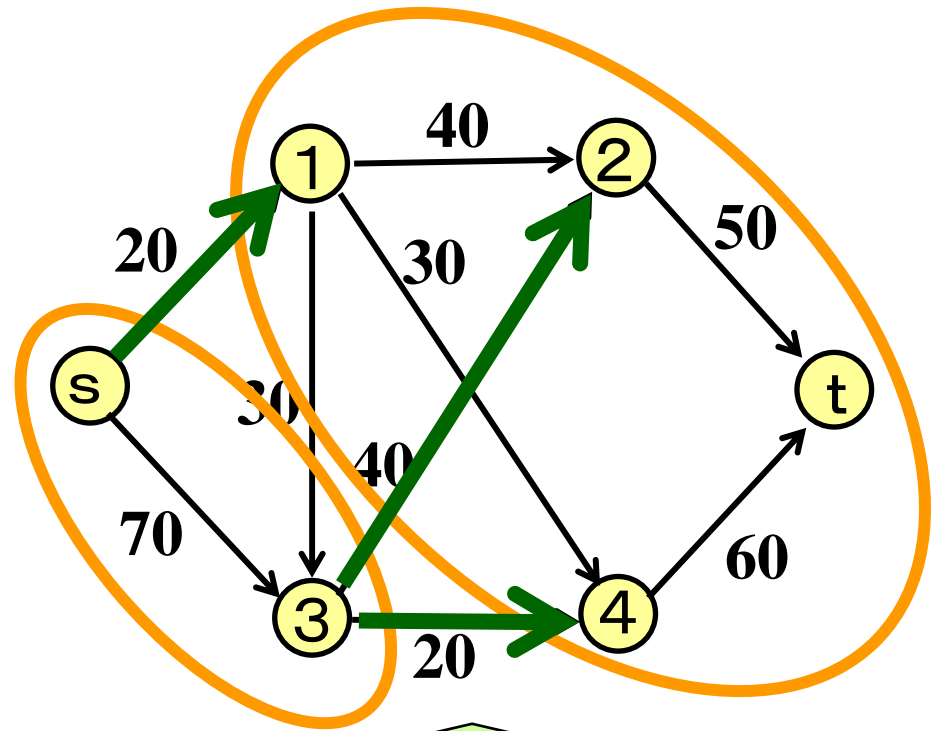
例題2 解答例

増加道法で最後の場面
(増加道が存在しない)



sから到達可能

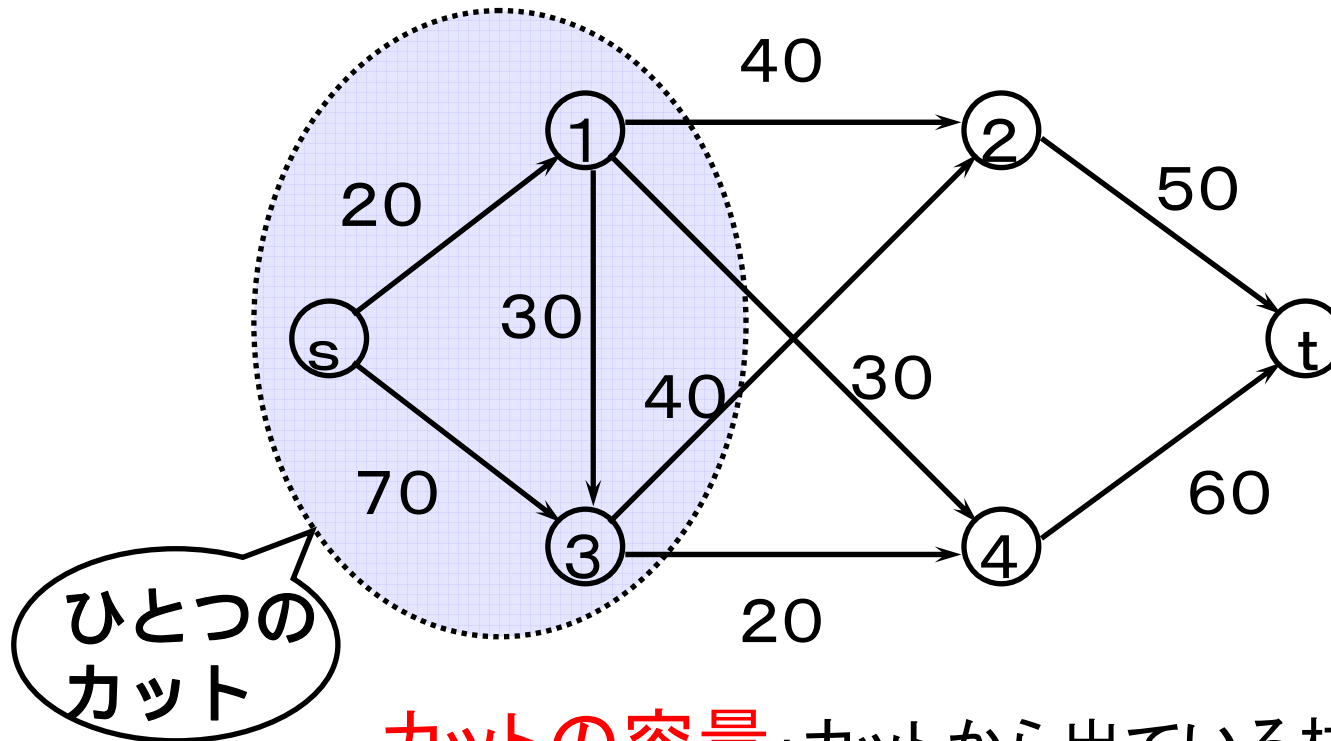
強連結成分分解



ボトルネック

カット

sを含み, tを含まない点の部分集合を**カット**という.
※ネットワーク上にカットはたくさんある



カットの容量: カットから出ている枝の容量の総和

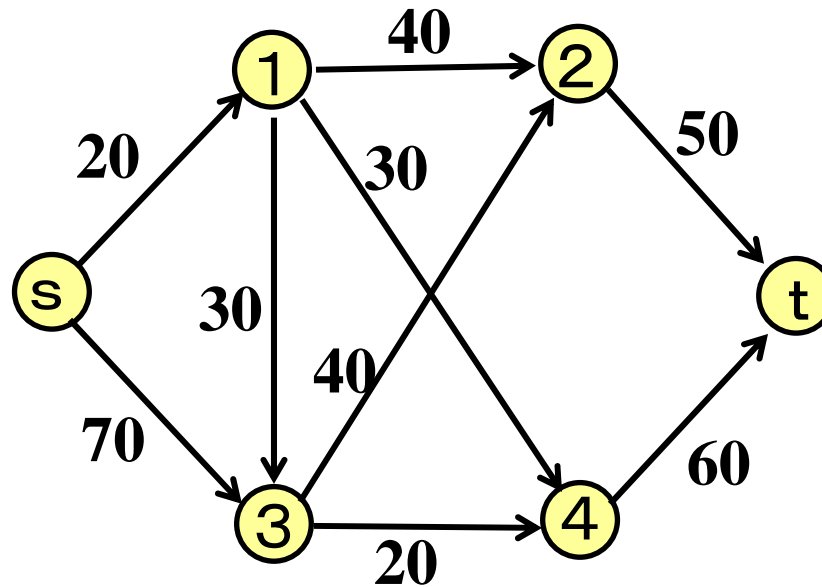
※上記のカットを特に「s-tカット」と呼ぶ場合もある

最小カット

最小カット: 容量最小のカット

演習7-4: 以下のネットワークの最小カットを見つけよう

➡ **最小カット問題**

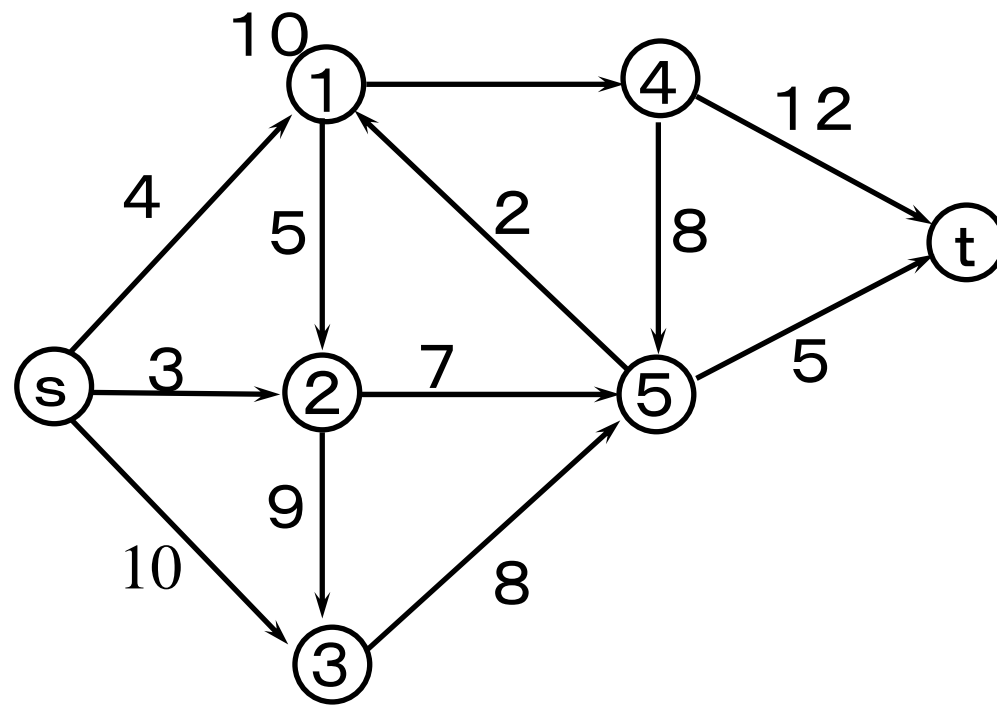


最大フローと最小カットの関係

- **最大フローの流量 = 最小カットの容量**
(最大フロー・最小カット定理)
 - 最小カットは最大フロー問題から導出可能
 - 導出概要: 容量いっぱいの流れ, 始点sと終点tを分割する枝集合 → 最小カット.
- ⇒ CPMを実行する時の最小カットは最大フロー問題に帰着することにより得られる.

演習 4

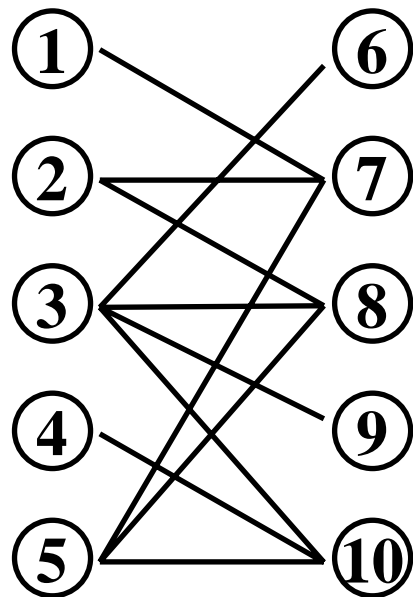
- 最小カットを求めよ



例題3 Shall we dance?



社交ダンスパーティーに男性・女性5人ずつ集まった。幹事がアンケートをとったところ、パートナーになってもよいとお互い思っているペアは以下の組合せであることがわかった。



男性

女性

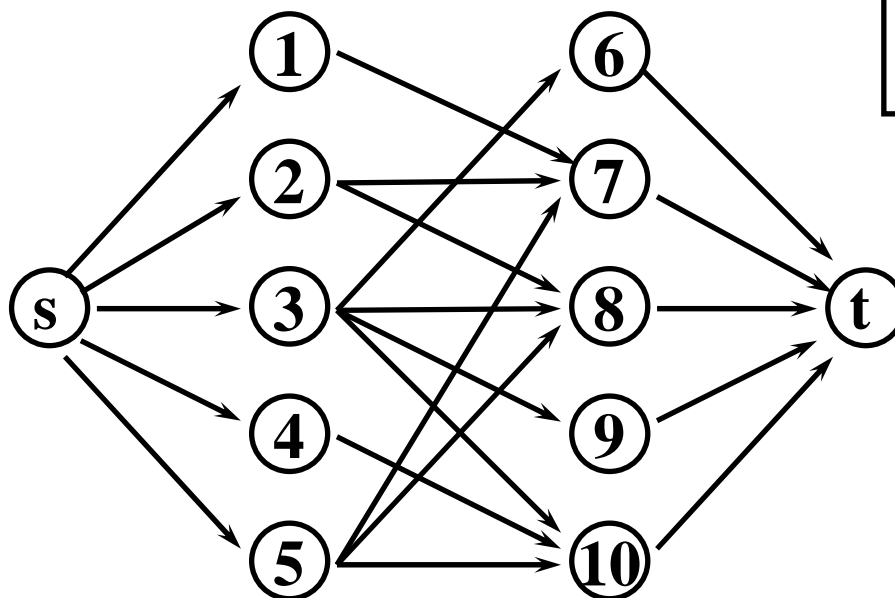
さて、なるべく多くのペアを組みたいが最大で何組できるか？その組み方は？

同時にペアになれる組合せを「マッチング」、このような問題を「マッチング問題」とよぶ。

マッチング問題の解法

以下のように変形し最大フローを求める.

演習5
求めてみよう!



各枝の容量はすべて1

- 「フローが流れている元の枝 \Leftrightarrow マッチング」 \leftarrow なぜか？
- 「最大フロー \Leftrightarrow 最大マッチング」 \leftarrow なぜか？

演習6 最大マッチングを求めよう

