

【S8】 SDN 時代を生き抜く為の
グラフ理論とネットワークの
アルゴリズム入門

浅間 正和 @ 日本 Vyatta ユーザ会

このプログラムの概要

- 導入
 - 背景、用語の説明、グラフの表現方法と探索
浅間（30分）
- グラフ理論とネットワークのアルゴリズムの基礎
 - 最短路問題、最小木問題、アルゴリズムと計算量
伊波さん（50分）
- ネットワークフローとその代表的な問題
 - 最大流問題、多品種流問題
金子さん（50分）
- まとめ
 - システム最適化流問題、参考情報、まとめ
浅間（20分）

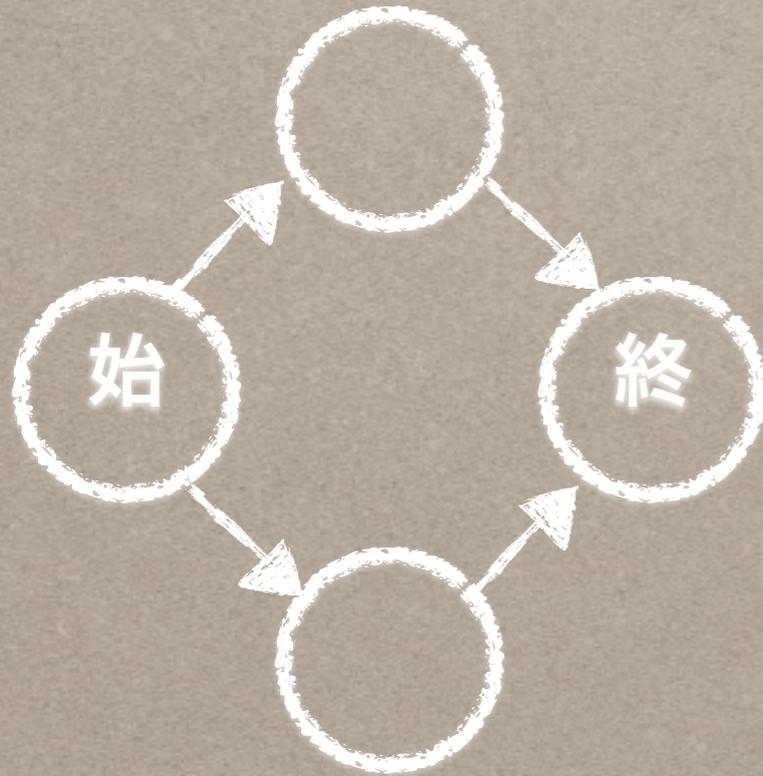
このプログラムの概要

- 導入
 - 背景、用語の説明、グラフの表現方法と探索
浅間（30分）
- グラフ理論とネットワークのアルゴリズムの基礎
 - 最短路問題、最小木問題、アルゴリズムと計算量
伊波さん（50分）
- ネットワークフローとその代表的な問題
 - 最大流問題、多品種流問題
金子さん（50分）
- まとめ
 - システム最適化流問題、参考情報、まとめ
浅間（20分）

計算してみよう! (1/5)

ごくシンプルなネットワークの例

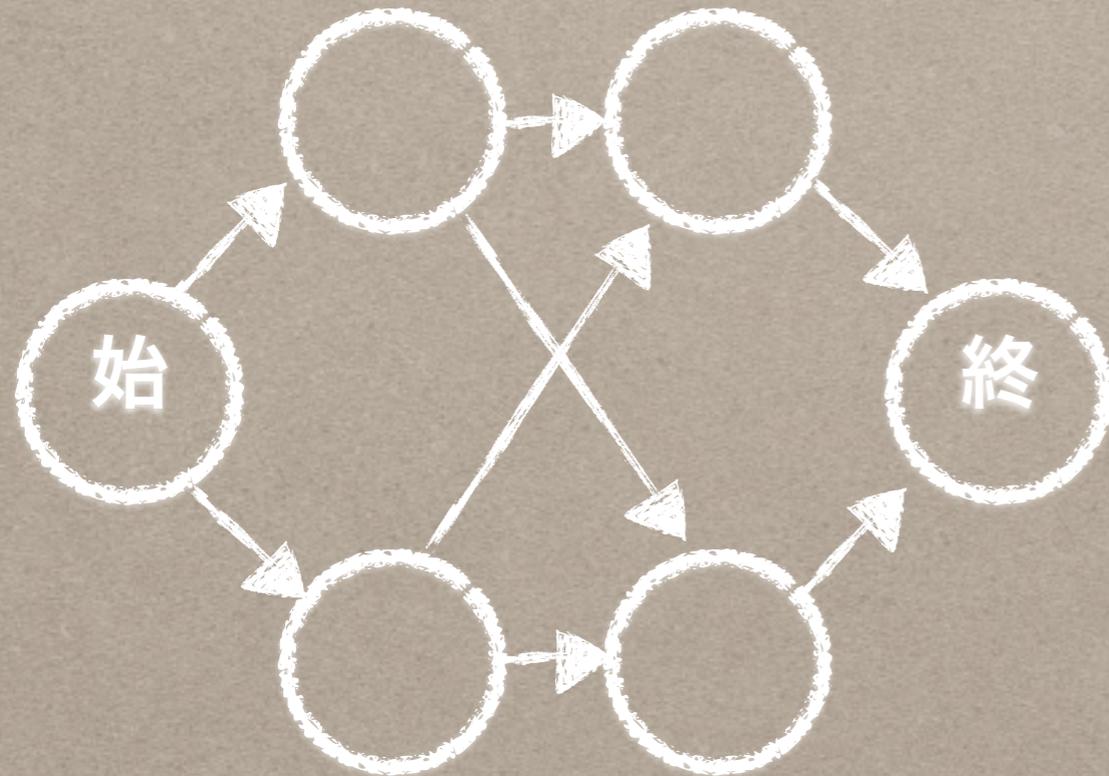
- ノード数4の場合の例: 全経路は2通り



計算してみよう! (2/5)

ごくシンプルなネットワークの例

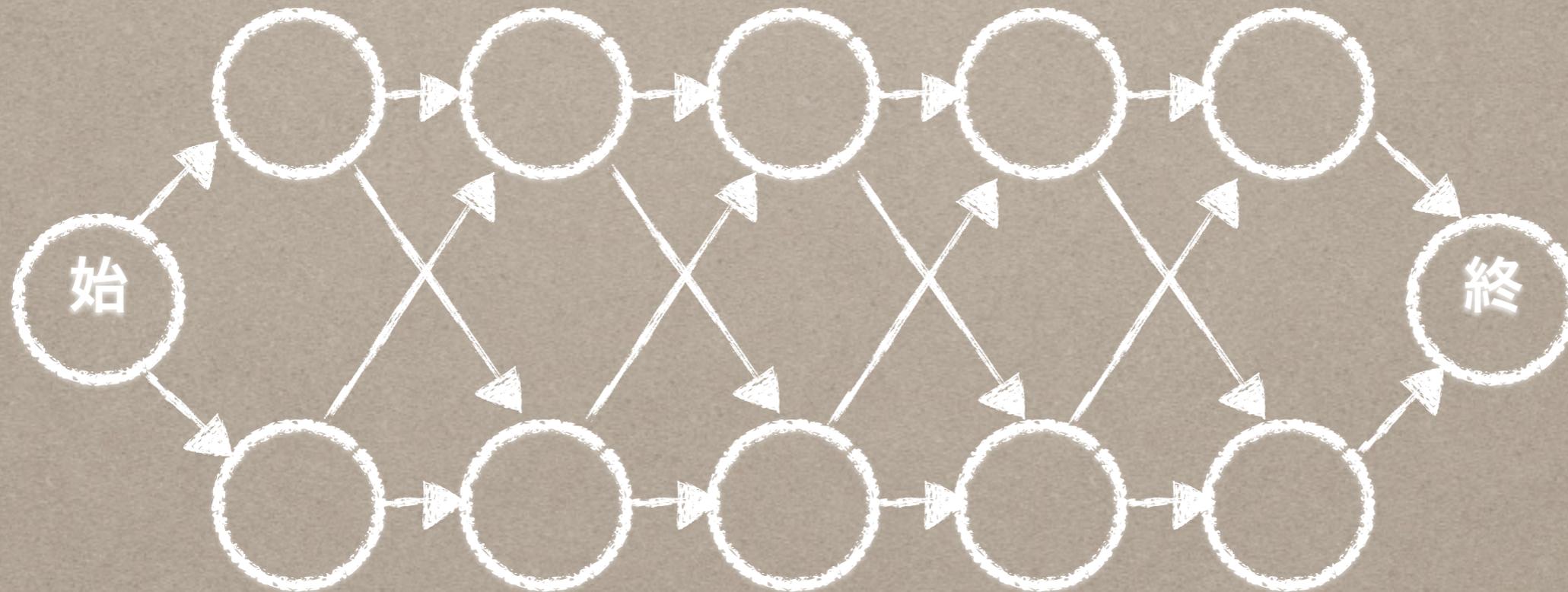
- ノード数6の場合の例: 全経路は4通り (2^2 通り)



計算してみよう! (3/5)

ごくシンプルなネットワークの例

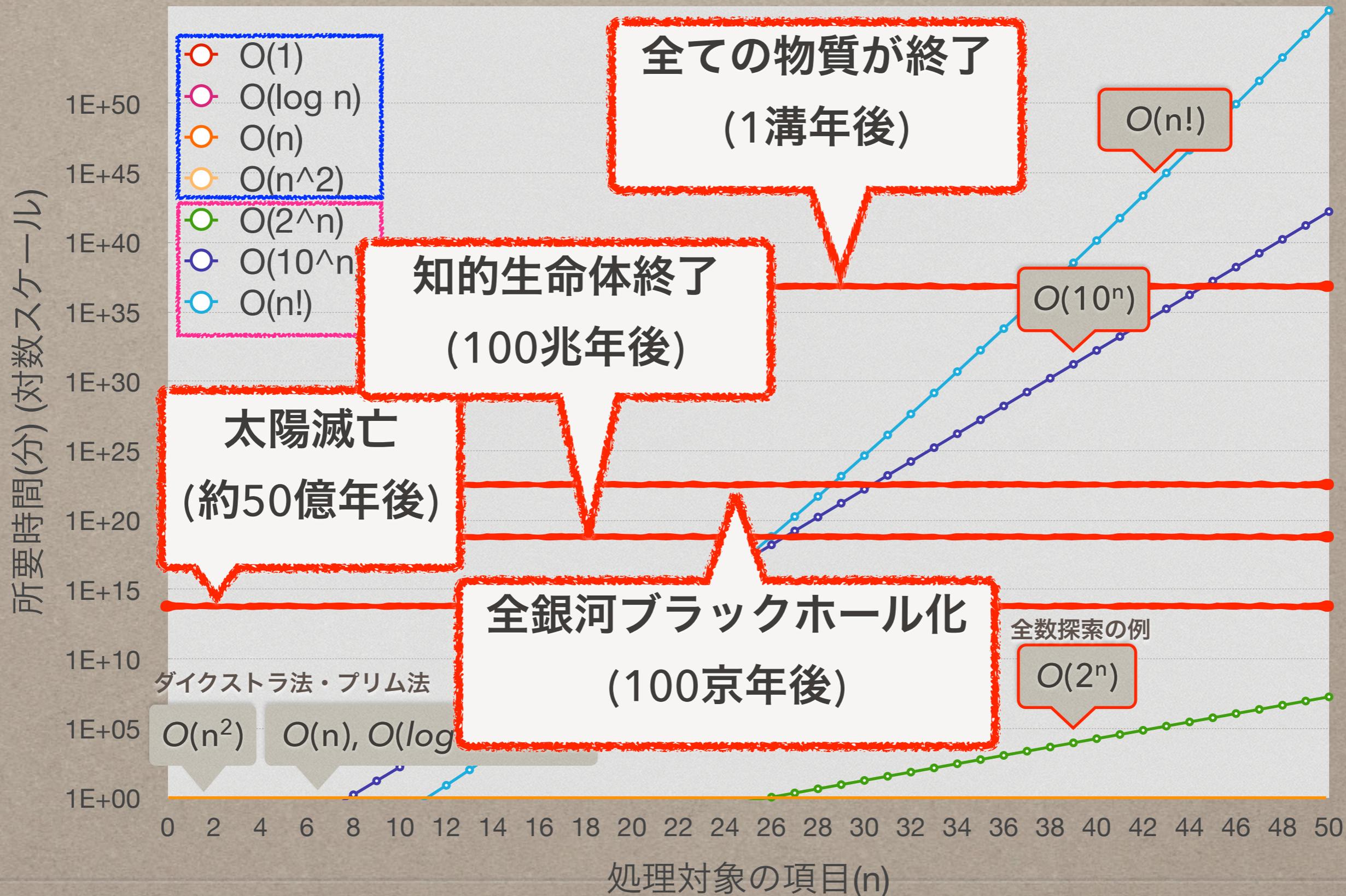
- ノード数12の場合の例: 全経路は32通り (2^5 通り)



計算してみよう! (4/5)

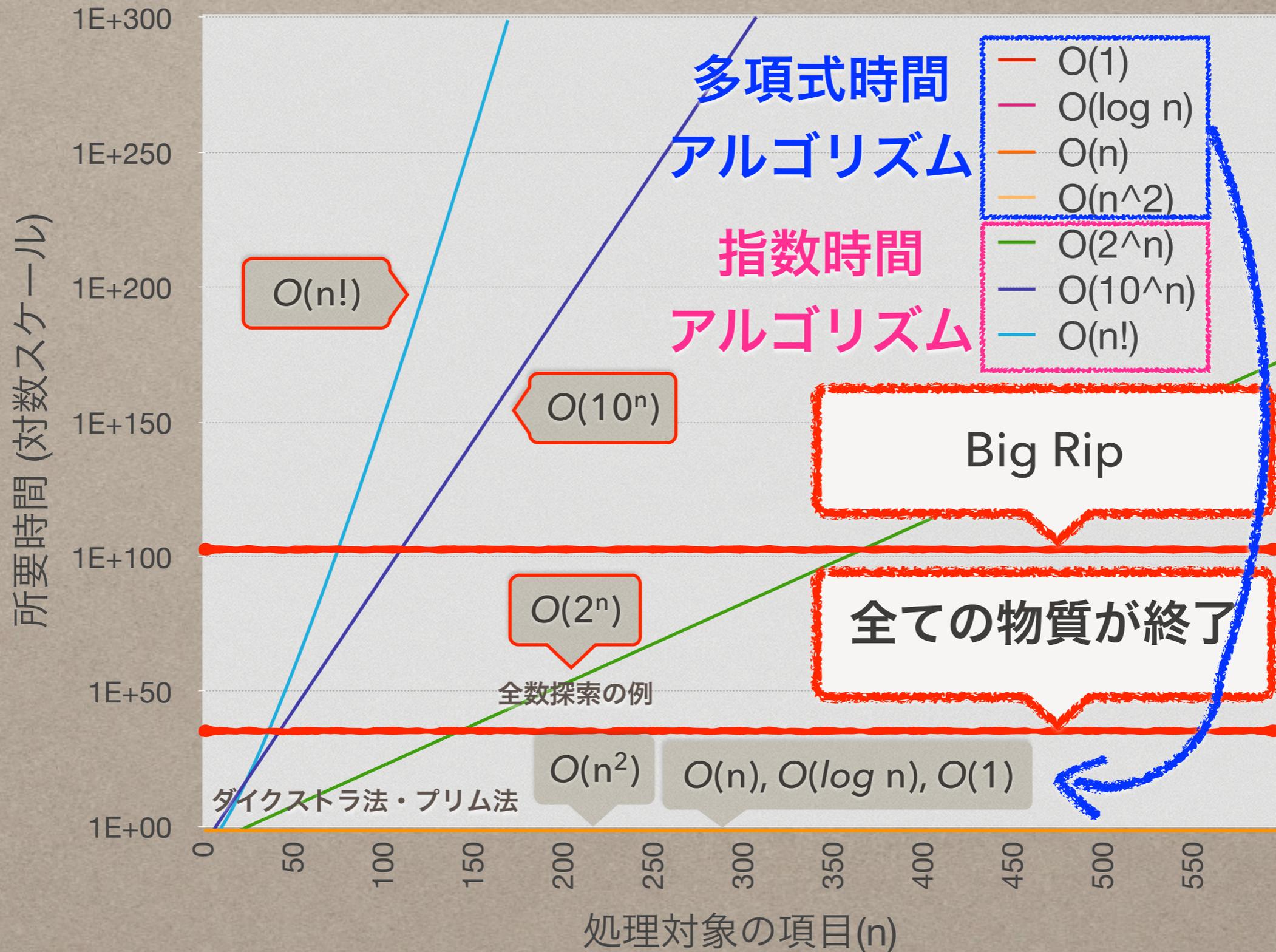
- ノード数22の場合: 2^{10} 通り (=1,024通り)
- ノード数42の場合: 2^{20} 通り (=1,048,576通り)
- ノード数102の場合: 2^{50} 通り
(= 1,125,899,906,842,624通り)
- ノード数202の場合: 2^{100} 通り
(=1,267,650,600,228,229,401,496,703,205,376通り)

アルゴリズム毎の所要時間のめやす



1処理を1μ秒で実行できるコンピュータを用いた際の見積もり時間

アルゴリズム毎の所要時間のめやす



このプログラムの概要

- 導入
 - 背景、用語の説明、グラフの表現方法と探索
浅間 (30分)
- グラフ理論とネットワークのアルゴリズムの基礎
 - 最短路問題、最小木問題、アルゴリズムと計算量
伊波さん (50分)
- ネットワークフローとその代表的な問題
 - 最大流問題、多品種流問題
金子さん (50分)
- まとめ
 - システム最適化流問題、参考情報、まとめ
浅間 (20分)

最大フロー問題

最大どれだけ流せるのか？を知りたい

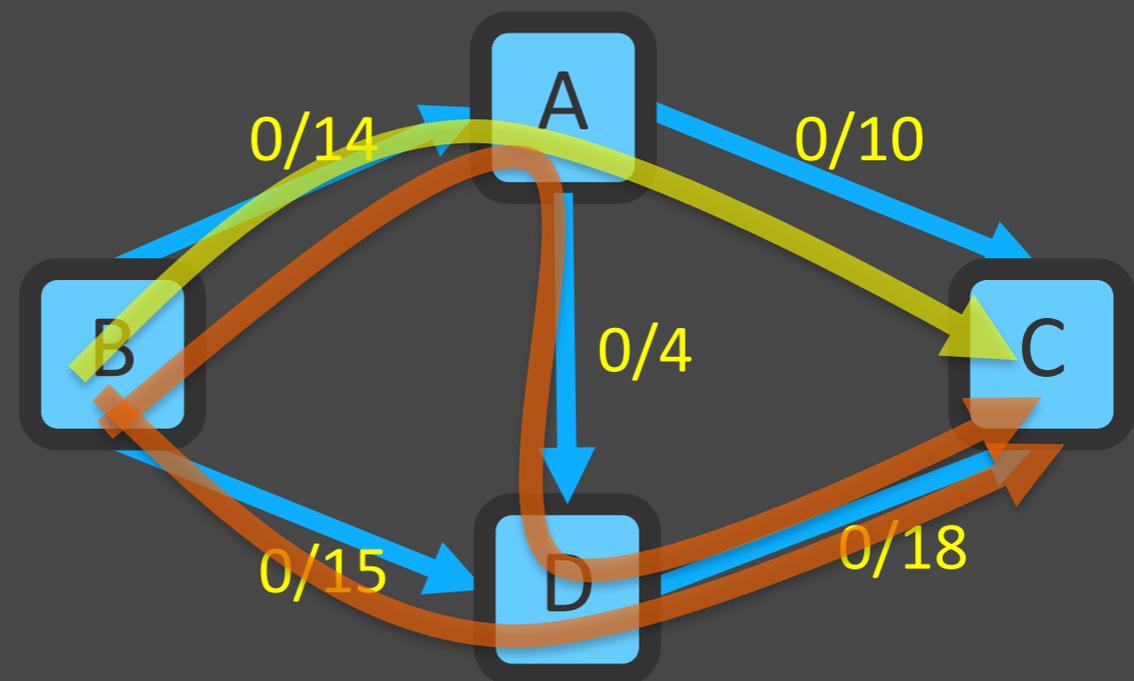
問：BからCまでの流量を最大化せよ

解：28

B→A→C :10

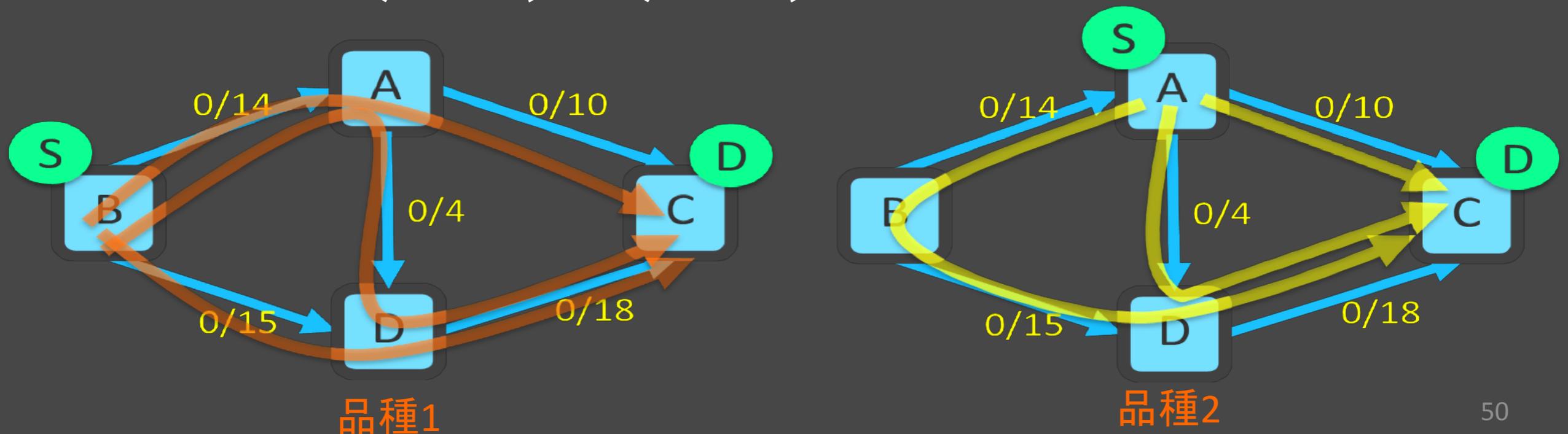
B→A→D→C:4

B→D→C :14



多品種フロー

- 品種とは発地, 着地のペアの事
- さきほど解いた多品種問題は1品種
 - 品種 (B→C)
- 複数の品種に拡張 (multi commodity)
 - 例えば (B→C) と (A→C) の2品種



最適な”割り当て”の問題

最適にネットワークを使いきれぬ条件

1品種の場合 品種1:28

2品種の場合 品種1:15, 品種2:13

要求

Aさん「品種1のパスに20流したい」

Bさん「品種2のパスに13流したい」

どうする??

Max-min Fairness Policy

Maximize the minimum (to allocate)

割り当て可能な帯域が最も小さいものを優先する

アルゴリズム

1. 全てのflowを同一ペースで増加させた場合に、最初に飽和するリンク(ボトルネックリンクと呼ぶ)を特定する
2. 全てのflowに対して、ボトルネックリンクが発生する流量分割り当てを行い、NWからボトルネックリンクを削除
3. 増加することのできるflowが存在する場合は1に戻る

Fairness Policy

Max-min fairness

– 合計帯域は減少したが割り当ては公平に

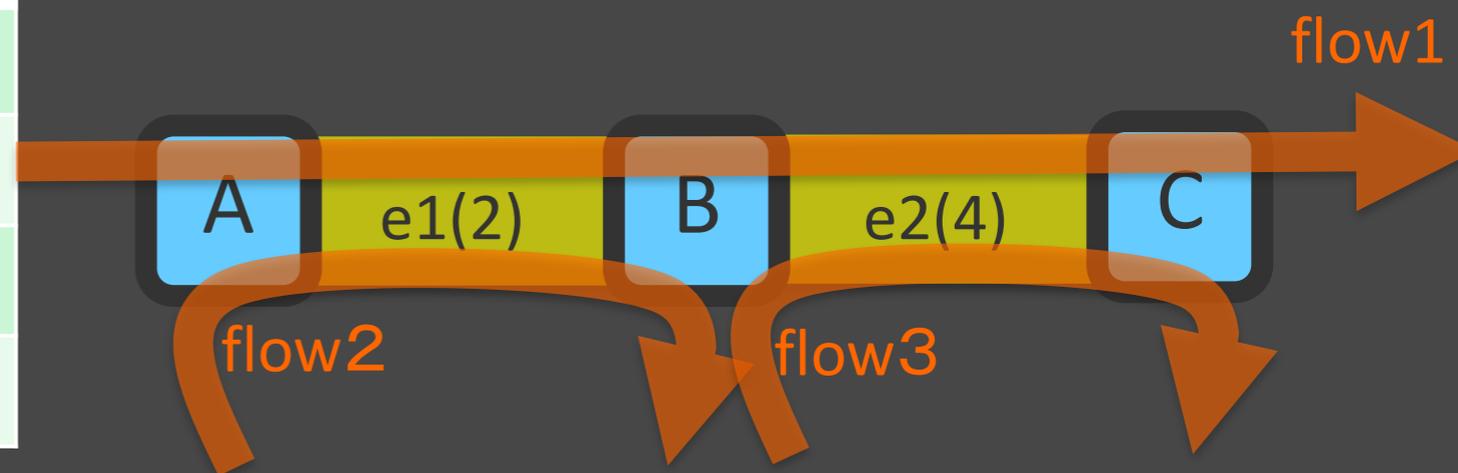
Fairness Policyはユースケース次第

最大流

flow	割り当て
1	0
2	2
3	4
Sum	6

Max-min

flow	割り当て
1	1
2	1
3	3
Sum	5



まとめ

- 紹介したアルゴリズム
 - Dijkstra 法（最短路問題）
 - Prim 法（最小木問題）
 - Ford-Fulkerson 法（最大流問題）
 - 線形計画法（最大流問題（多品種含む））
 - Max-Min Fairness 法（多品種最大流問題）
- アルゴリズムの計算量はとても重要
 - 適切なアルゴリズムを選択しないと大変なことに...
- 多品種流問題はとてもチャレンジングな領域
 - なにをもって“最適”とするかはひとそれぞれ...