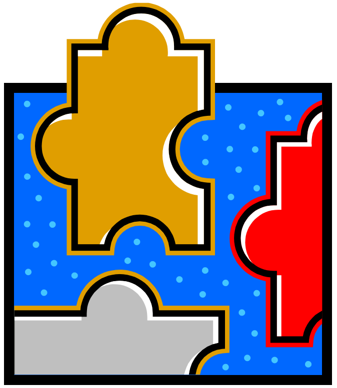


Mathematical Programming（2）

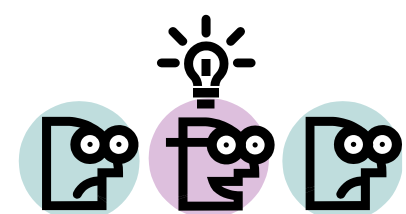
最適化手法の王様
数理計画法

ここで学ぶこと

前回

- 数理計画とは
- システム的アプローチによる問題解決

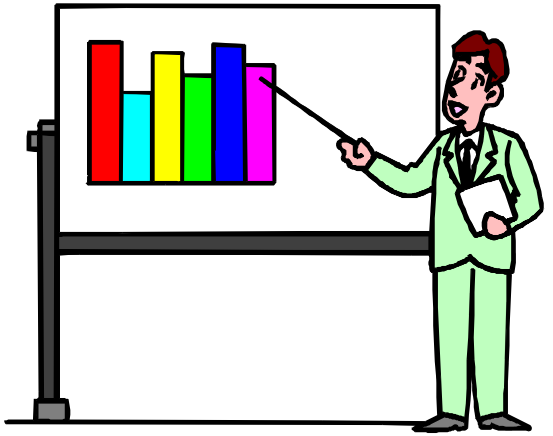
- 数理モデル化と表現方法（定式化）
- 数理計画問題の分類



（復習）数理計画とは
Mathematical Programming

与えられた**制約式のもと**で,
ある**関数を最大化**する応用数学の問題
（最小化）

•数理計画＝数理計画問題
(ー problem)
•数理計画問題とそれを解く手法
全般を「**数理計画法**」とよぶ.



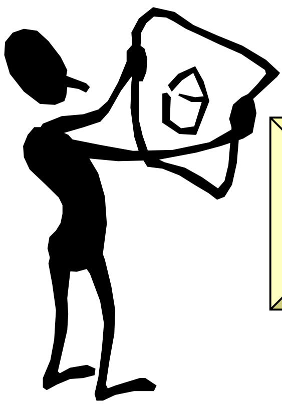
例題1　数式での表現

3種類の原液A,B,Cから,
2つの粉末製品P,Qを製造

	製品P 1(kg)	製品Q 1(kg)	使用可能量
原液A	15(kl)	11(kl)	1650(kl/日)
原液B	10(kl)	14(kl)	1400(kl/日)
原液C	9(kl)	20(kl)	1800(kl/日)
利益	5(万円)	4(万円)	

利益が最大になる製品P,Qの1日の生産量は？
問題を数理モデル化しなさい.

数理モデル作成　2つの段階



数理モデル化 → 定式化 formulation

問題理解 → 問題表現

観察力･言語理解力

システムとしての把握

- 構成要素は？
  - コントロール可能な要素
  - コントロールできない要素
- 相互関係は？
- コントロール結果の評価方法は？

変数として表現
例：x_1, x_2

定数として表現

数式として表現
例：等式，不等式

関数として表現

例題1(続) 定式化してみよう

·コントロールできる要素 : 製品P,Qの生産量
→製品Pの生産量をx_1, 製品Qの生産量をx_2とおく.

·コントロールの制約 : 原液A,B,Cの使用可能量

·コントロール結果の評価 : 利益

•制約を表す不等式は？
•利益を表す関数は?

数理計画問題の書き方

目的関数　Objective function

最大化
(最小化の時はminimize)

$$\begin{aligned}
&\text{maximize } z=5x_1+4x_2\\
&\text{subject to } 15x_1+11x_2 \leqq 1650\\
&\qquad 10x_1+14x_2 \leqq 1400\\
&\qquad 9x_1+20x_2 \leqq 1800\\
&\qquad x_1 \geqq 0\\
&\qquad x_2 \geqq 0
\end{aligned}$$

又は制約条件式
subject to:〜の条件の下で

制約式　Constraints

省略表記

$$\begin{aligned}
&\text{max. } z=5x_1+4x_2\\
&\text{s.t.} \quad 15x_1+11x_2 \leqq 1650\\
&\qquad 10x_1+14x_2 \leqq 1400\\
&\qquad 9x_1+20x_2 \leqq 1800\\
&\qquad x_1, x_2 \geqq 0
\end{aligned}$$

目的関数の$z=$も
省略される時あり

練習 生産計画

- 2つの液体製品P,Qは機械A,Bを用いて加工される
- 利益が最大になる1週間の液体P,Qの加工量は？

	液体P 1ml	液体Q 1ml	使用可能時間
機械A	3(h)	1(h)	45(h/週)
機械B	1(h)	2(h)	40(h/週)
利益	6(万円)	5(万円)	

⇨ 定式化してみよう

練習 解答例

練習を定式化

x_1(ml) : 液体Pの加工量
x_2(ml) : 液体Qの加工量

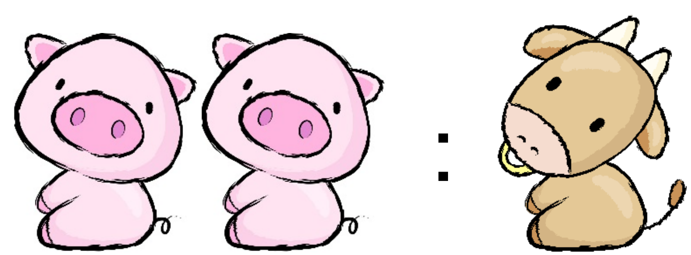
$$
\begin{aligned}
\text{max.}\ & z=6x_1+5x_2 \\
\text{s.t.}\ & 3x_1+\ \ x_2 \leqq 45 \\
& x_1+2x_2 \leqq 40 \\
& x_1,\ x_2 \geqq 0
\end{aligned}
$$

練習2　(1)

ハンバーグは脂肪が全体の25%を超えない範囲で
牛肉と豚肉を合わせて作るとおいしいらしい.
できる限り安い値段で,
おいしいハンバーグ1kgを作るには牛肉と豚肉を
何グラムずつ合わせると良い？

[基礎データ]
- 牛肉　赤身80%, 脂肪20%, 100gあたり150円
- 豚肉　赤身60%, 脂肪40%, 100gあたり120円

[変数のヒント]
牛肉の量をx_1(g), 豚肉の割合をx_2(g)とおいてみよう.



練習2　(2)

3箇所の漁場で収穫された海苔は高級と上級に分けられる.
今週(7日間)で高級54トン, 上級65トンを収穫費用最小で出荷したい.
各漁場で何日間操業(収穫)を実施すればよいか？

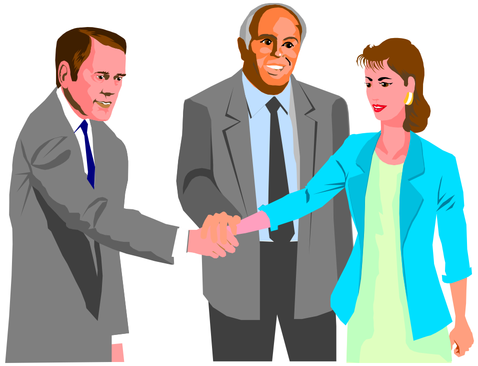
[基礎データ]

- 漁場A　収穫量:高級5トン/日, 上級5トン/日 ;収穫費用20万円/日
- 漁場B　収穫量:高級6トン/日, 上級4トン/日 ;収穫費用22万円/日
- 漁場C　収穫量:高級1トン/日, 上級6トン/日 ;収穫費用18万円/日

[変数のヒント]

各漁場の操業日数をx_A(日), x_B(日) , x_C(日)とおいてみよう.

演習2 原料奪取作戦

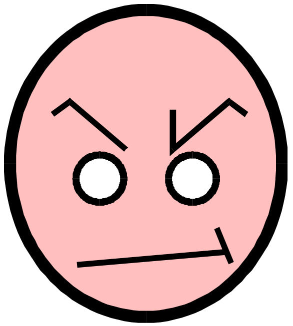


例題1で登場した会社から

- 原液A, B, Cの1日の使用可能量をすべて買い取りたい.
- 支払総額は少なくしたい.
- 問題:各原液1klにいくらで提示する？

この問題を数理モデルで表現しなさい

演習2(続) ヒント



- 変数(コントロールできるもの)
  - 原液Aの買取提示価格　y_1(円/kl)
  - 原液Bの買取提示価格　y_2(円/kl)
  - 原液Cの買取提示価格　y_3(円/kl)
- 制約(交渉成立の条件):
  売主は自製造で得る利益以下では売らない
  - 自製造で得る利益以上の金額を提示すべき

数理モデルで表現してみよう！

用語:実行可能解と最適解

optimal solution

最適解:最適値を達成する実行可能解

$$
\begin{aligned}
\text{max. } & z = 5x_1 + 4x_2 \\
\text{s.t. } & 15x_1 + 11x_2 \leqq 1650 \\
& 10x_1 + 14x_2 \leqq 1400 \\
& 9x_1 + 20x_2 \leqq 1800 \\
& x_1, x_2 \geqq 0
\end{aligned}
$$

最適値:目的関数の最大値

optimal value

feasible solution

実行可能解:制約式を満たす(x_1, x_2)

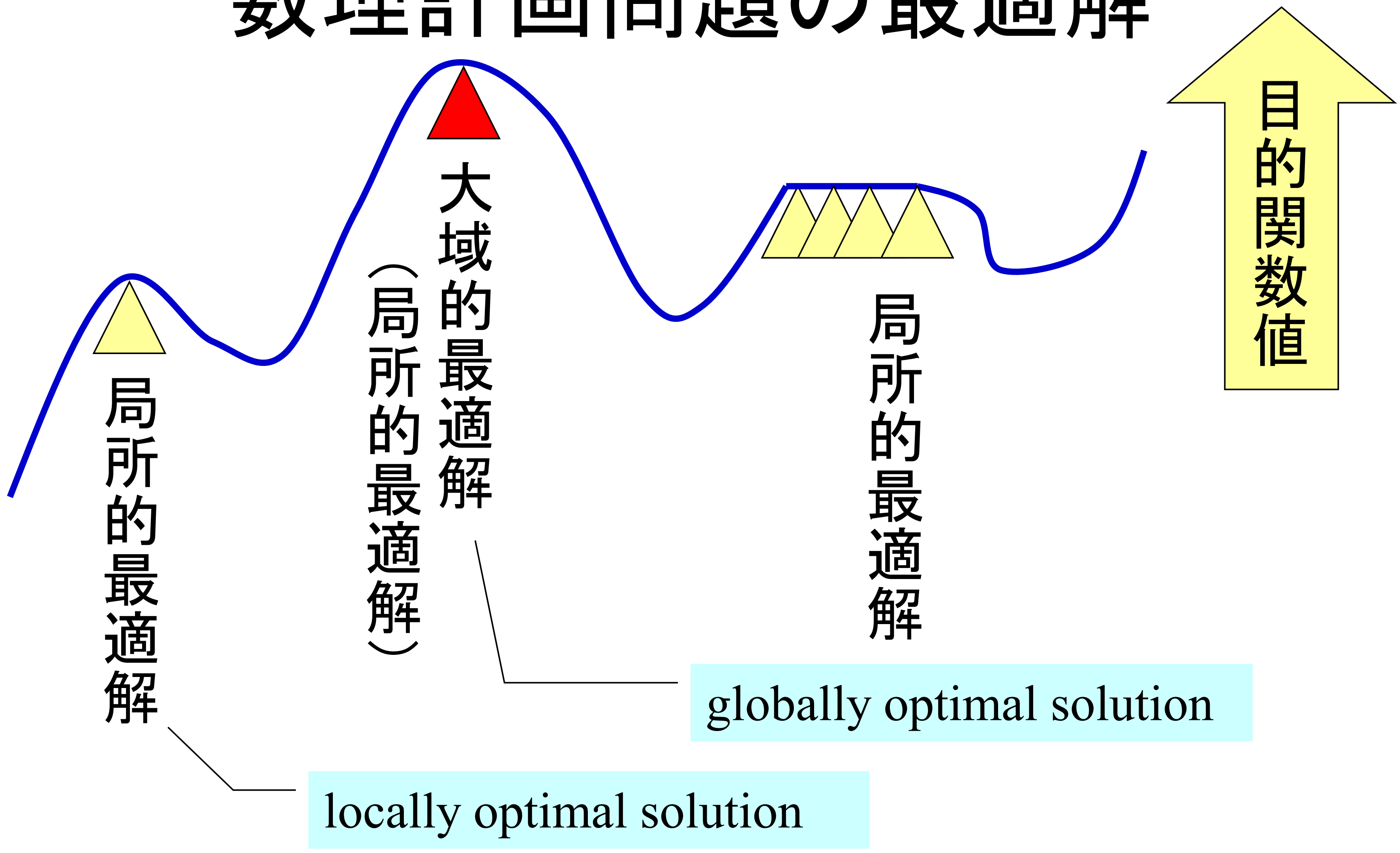
feasible region

実行可能領域:実行可能解の集合

※実行可能解が存在しない場合もある→実行不能な問題

※実行可能でも最適解が存在しない場合がある→例題2

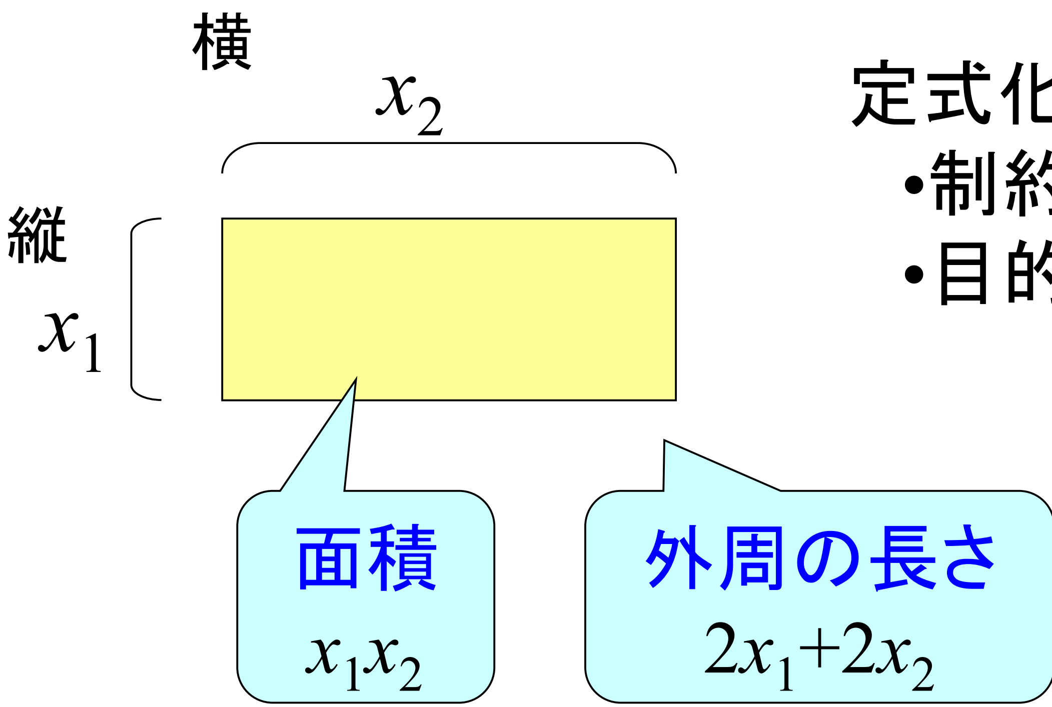
数理計画問題の最適解



※最適解が複数存在する場合もある→通常1つだけ求めればよい

例題2

面積が4以上で，外周の長さ最小の長方形は?



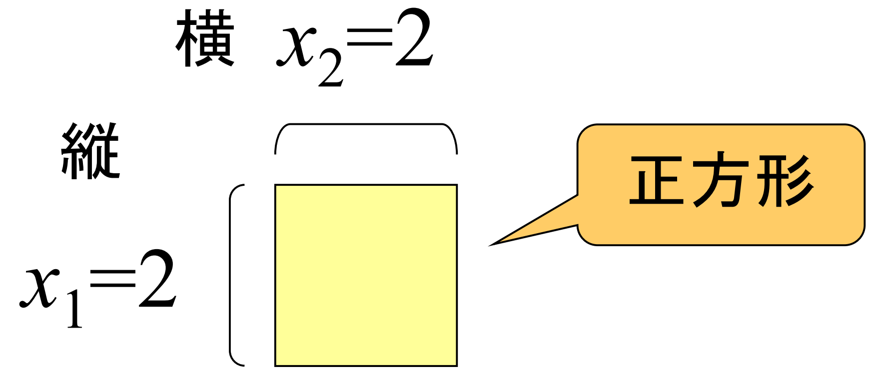
定式化してみよう

- 制約条件は?
- 目的関数は?

例題2 解答例

- 最適解は $x_1=2,\ x_2=2$
- 最適値は 8

$$
\begin{aligned}
\text{min. } & z=2x_1+2x_2 \\
\text{s.t. } & x_1x_2 \geqq 4 \\
& x_1,\ x_2 \geqq 0
\end{aligned}
$$



Q. 面積が4以上で縦の長さ最小の長方形は?

$$
\begin{aligned}
\text{min. } & z=x_1 \\
\text{s.t. } & x_1x_2 \geqq 4 \\
& x_1,\ x_2 \geqq 0
\end{aligned}
$$

限りなく0に近い値?

⇒最適解はない

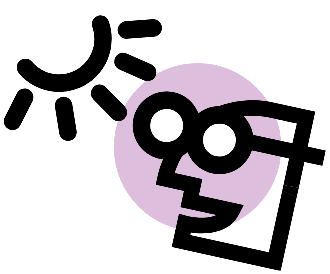
最適解が存在する･しない

実行可能領域が存在 ⟺ 実行可能領域が空

最適値は有限

最適解存在

複数存在も有



最適解無し

例題2
後半の場合

最適値が発散

目的関数値を
いくらでも
大きくできる

最適解無し

非有界
unbounded

実行可能解がない
↓
最適解無し

実行不能
infeasible

実行可能解の存在判定

実行可能性問題 feasibility problem

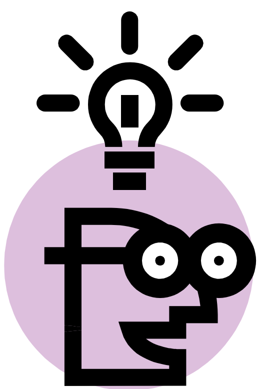
> 実行可能解が存在するのかを判定する問題

解法:いつでも値が0になる関数を目的関数にする

⇒実行可能解があれば，最適値は0

例

$$
\begin{array}{ll}
\text{max.} & z = \mathbf{0}x_1 + \mathbf{0}x_2 \\
\text{s.t.} & 15x_1 + 11x_2 \leqq 1650 \\
 & 10x_1 + 14x_2 \leqq 1400 \\
 & 9x_1 + 20x_2 \leqq 1800 \\
 & x_1,\ x_2 \geqq 0
\end{array}
$$

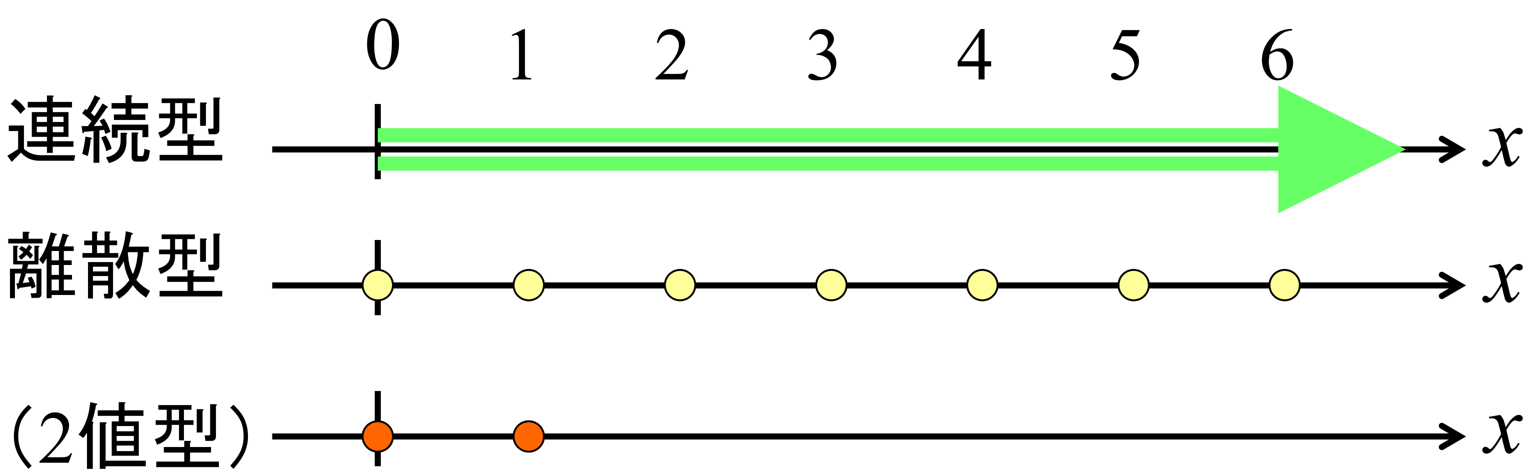


実行可能性の判定も
最適化問題なんだ

定式化の分類法(1)

利用する変数が取れる値の型で分類

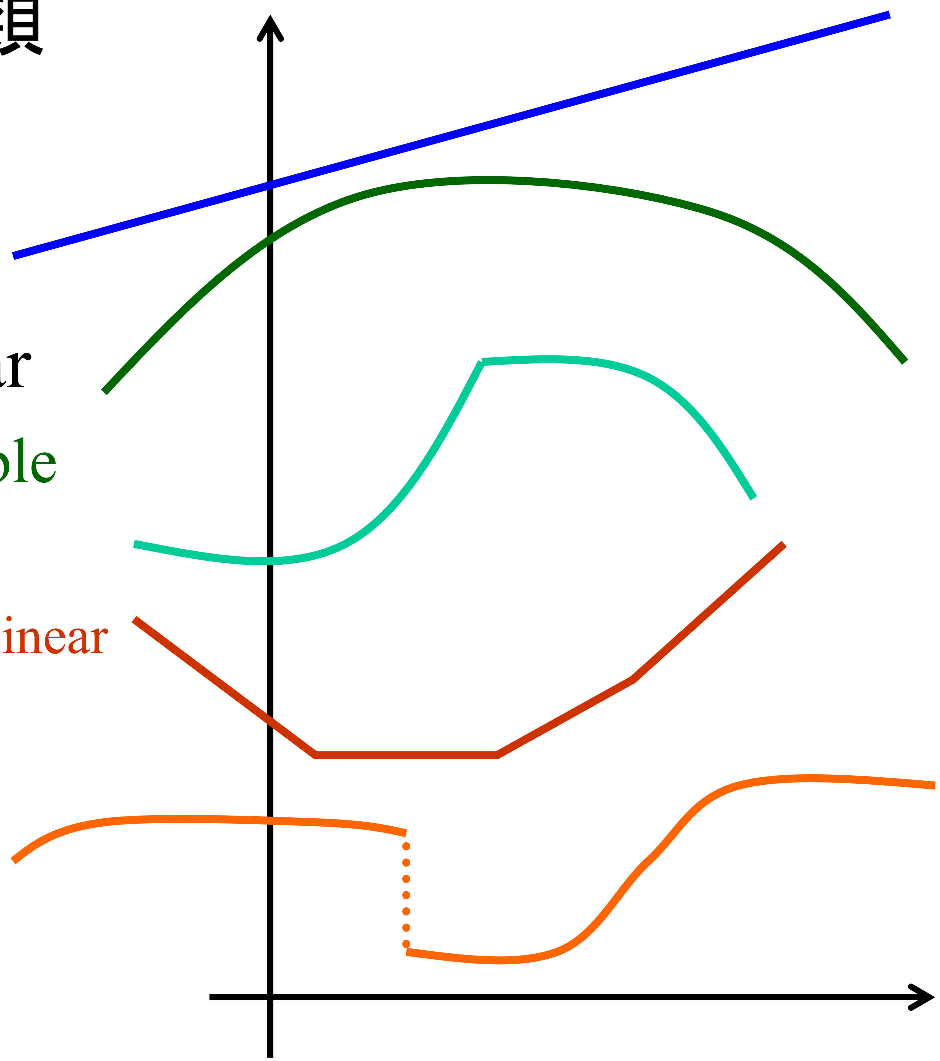
- 連続型 continuous ⇒ 例：実数 real
- 離散型 discrete ⇒ 例：整数 integer（整数計画）
  - 2値型 binary ⇒ 例：0または1（0-1整数計画）

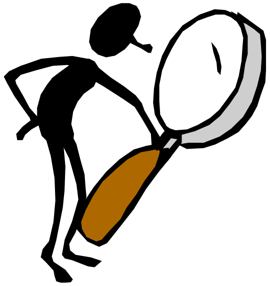


定式化の分類法(2)

使用関数の種類で分類

- 連続関数
  - 線形関数 linear
  - 非線形関数 nonlinear
    - 微分可能 differentiable
    - 微分不能 non-
      - 区分線形 piecewise linear
- 非連続
- 凸関数 convex
- 凹関数 concave





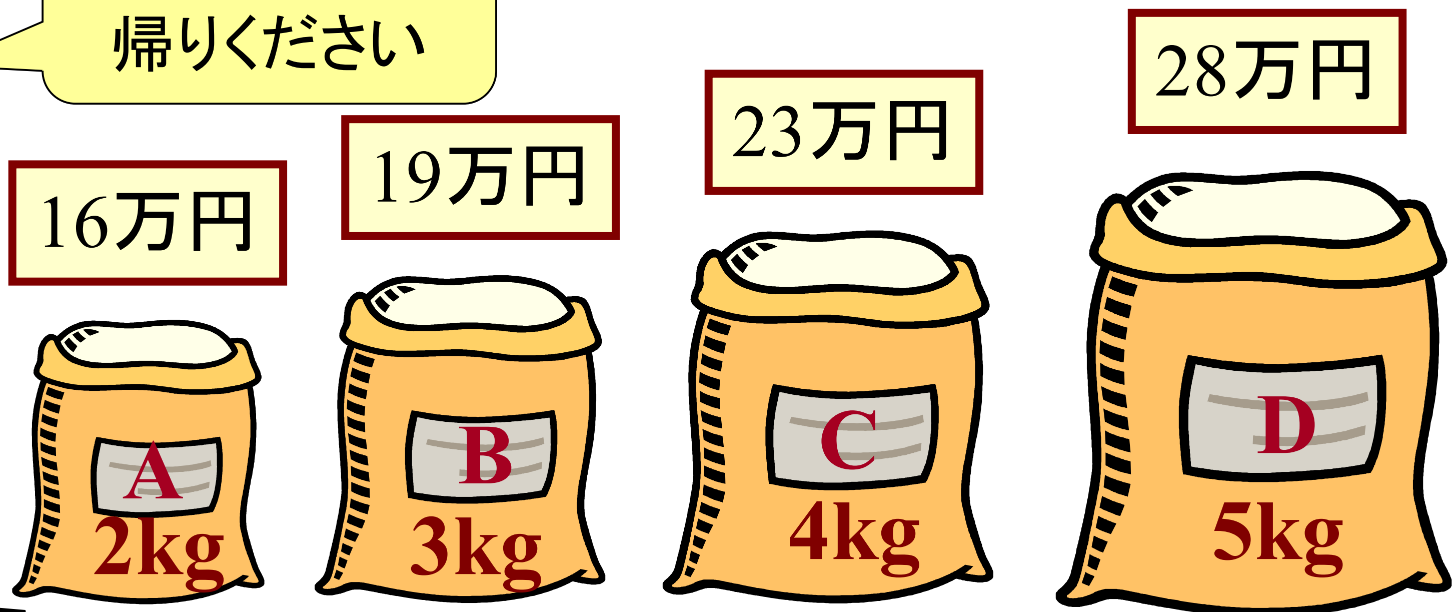
分類後の主な問題名

変数型	目的関数	条件式	問題名	+Programming	略称
連続型	線形	線形	線形計画	Linear	LP
	2次関数	線形	2次計画	Quadratic	QP
	凸関数	凸関数	凸計画	Convex	CP
	非線形	非線形	非線形計画	Nonlinear	NLP
離散型	ー	ー	整数計画	Integer	IP
	凸	凸	離散凸計画	Discrete Convex	
2値型	ー	ー	0-1整数計画	Binary Integer	BIP
混合	ー	ー	混合整数計画	Mixed Integer	MIP

※ 凸計画の等式制約は線形

例題3 ナップザック問題

自由にお持ち
帰りください

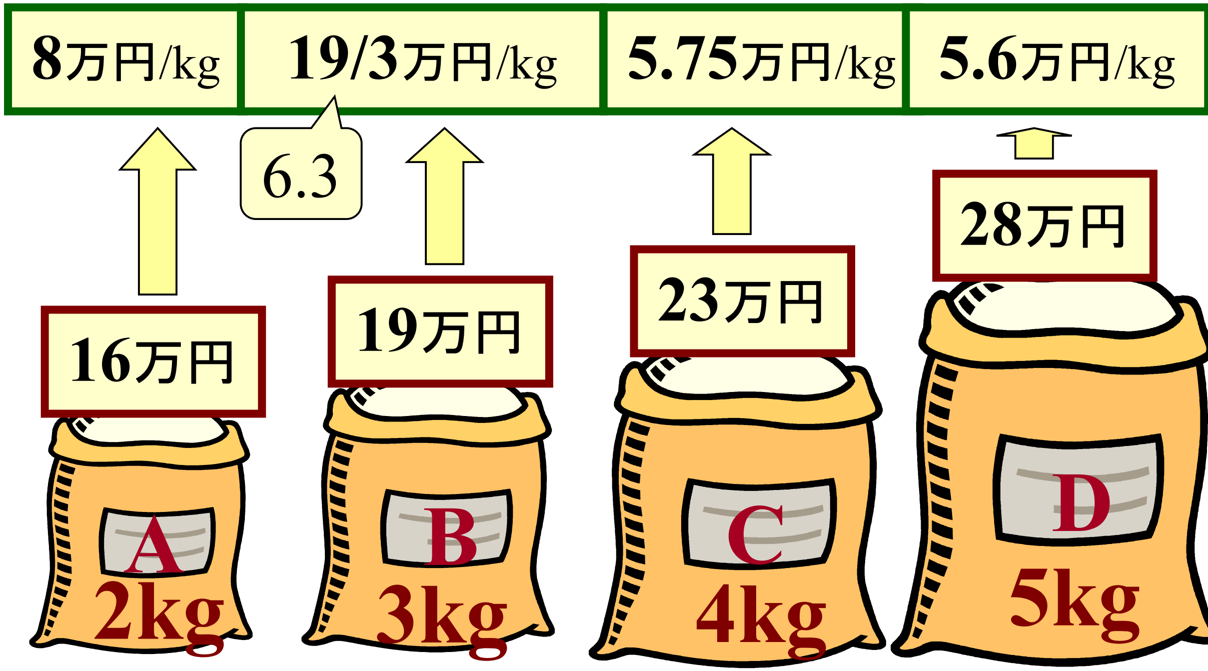


重量制限: 7kg

なるべく総価値を高く持って帰りたい.
どれを何Kg持って帰る?

⇒定式化してみよう

例題3（続）
分割可の時



単位価値額

x_1 kg　x_2 kg　x_3 kg　x_4 kg

変数:積む量

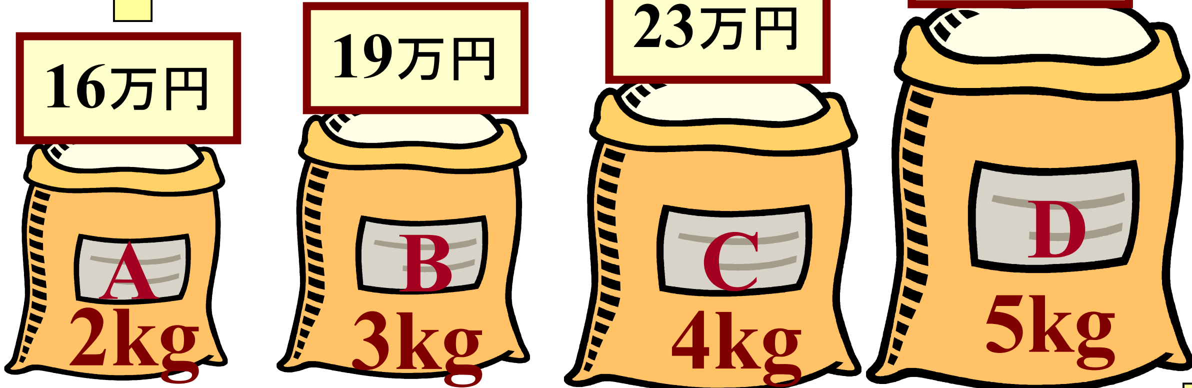
線形計画

$$
\begin{aligned}
\text{max.}\quad & z = 8x_1 + 19/3x_2 + 5.75x_3 + 5.6x_4 \\
\text{s.t.}\quad & x_1 + x_2 + x_3 + x_4 \leqq 7 \\
& x_1 \leqq 2,\ x_2 \leqq 3,\ x_3 \leqq 4,\ x_4 \leqq 5, \\
& x_1, x_2, x_3, x_4 \geqq 0
\end{aligned}
$$

~~8万円/kg | 6.3万円/kg | 5.7万円/kg | 5.6万円/kg~~

~~単位価値額~~

16万円 19万円 23万円 28万円



例題3（続）
分割不可の時

x_1 x_2 x_3 x_4

2値（0-1）変数

積む時：$x=1$

積まない時：$x=0$

0-1整数計画

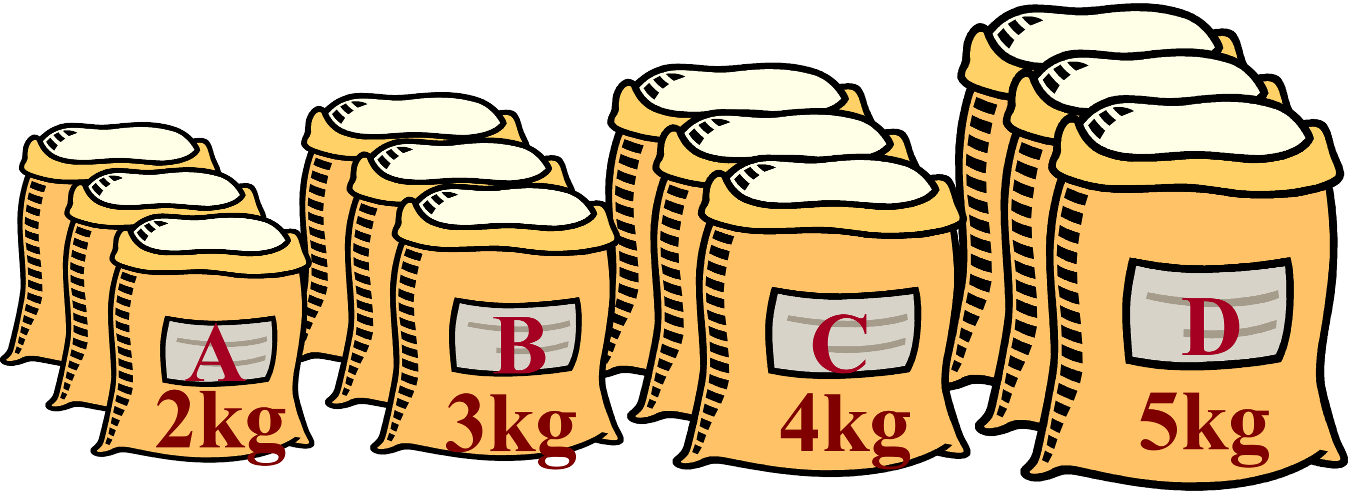
$$\text{max. } z=16x_1+19x_2+23x_3+28x_4$$

$$\text{s.t.} \quad 2x_1+3x_2+4x_3+5x_4 \leqq 7$$

$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

記号∈
元として含まれる

16万円/袋 19万円/袋 23万円/袋 28万円/袋



x_1 袋 x_2 袋 x_3 袋 x_4 袋

例題3（続）
分割不可
複数可の時

変数:いくつ積む?

整数計画

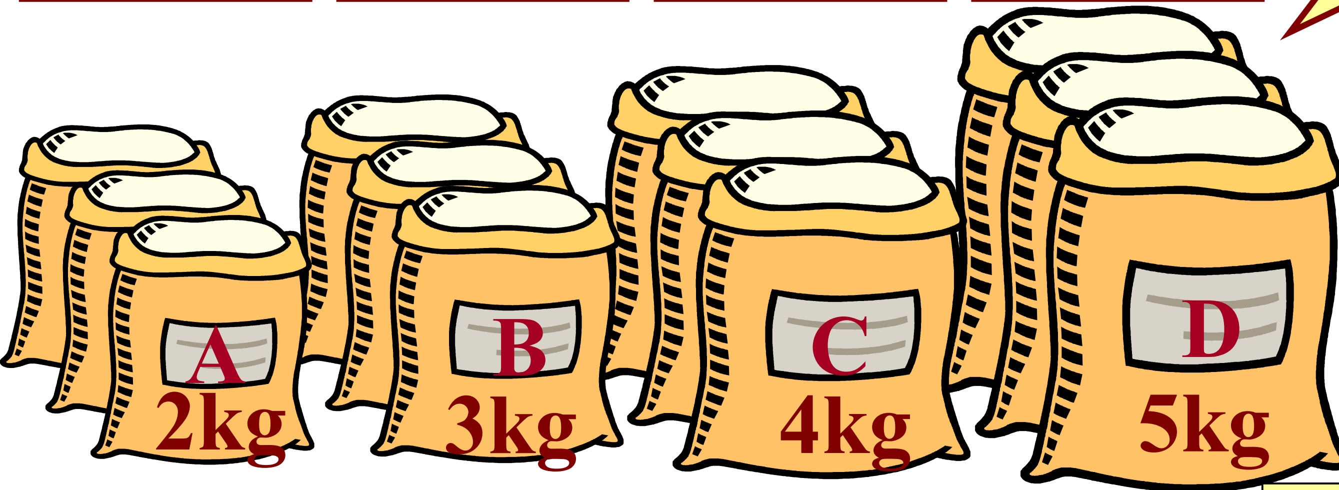
$$\begin{aligned} \text{max.}\ & z=16x_1+19x_2+23x_3+28x_4 \\ \text{s.t.}\ & 2x_1+3x_2+4x_3+5x_4 \leqq 7 \\ & x_1, x_2, x_3, x_4 \in \mathrm{Z}_+ \end{aligned}$$

記号 Z_+
非負整数の集合

（参考）R: 実数
Z_{++}: 正の整数

Dのみ分割可

16万円/袋 19万円/袋 23万円/袋 28万円/袋



例題3（続）
分割一部可
複数可の時

x_1 袋 x_2 袋 x_3 袋 x_4 袋

変数:何袋分積む?

変数:何袋積む?

混合整数計画

$$
\begin{aligned}
\text{max.}\ & z=16x_1+19x_2+23x_3+28x_4 \\
\text{s.t.}\ & 2x_1+3x_2+4x_3+5x_4 \leqq 7 \\
& x_1, x_2, x_3 \in \mathrm{Z}_+,\ x_4 \geqq 0
\end{aligned}
$$

例題3（続）分割不可の時(別表現1)

2kg,3kg,4kg,5kg

カートの重量制限（kg）

対象の粉を順に増やす

積む　積まない　+0　+16　+19　+23　+28

	0	1	2	3	4	5	6	7
なし	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	**0**	**0**	**16**	**16**	**16**	**16**	**16**	**16**
	**0**	**0**	**16**	**19**	**19**	**35**	**35**	**35**
	**0**	**0**	**16**	**19**	**23**	**35**	**39**	**42**
	**0**	**0**	**16**	**19**	**23**	**35**	**39**	**44**

16万,19万,23万,28万

例題3（続）動的計画法

カートの重量制限

	0	1	2	3	4	5	6	7
	0	**0**	**16**	**19**	**19**	**35**	**35**	**35**
4kg	**0**	**0**	**16**	**19**	**23**	**35**	**39**	**42**

+23

粉がk種類，カートの制限重量がαkgの時の最適値

制限重量αが粉kの重み以下のとき

粉kを積まない

粉kを積む

$$f(k,\alpha) = \begin{cases} f(k-1,\alpha) \\ \max\{f(k-1,\alpha),\ (k\text{の価値})+f(k-1,\alpha-(k\text{の重み}))\} \end{cases}$$

比較して，価値の高い方を採用

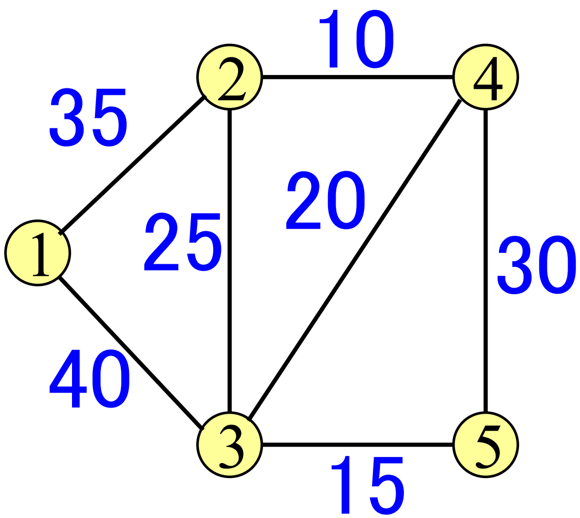
再帰方程式

動的計画法 Dynamic Programming（DP）

例題4 ガス管配置

5軒の家にガスを供給したい

設置費用が最小になるガス管の設置方法は?



定式化してみよう

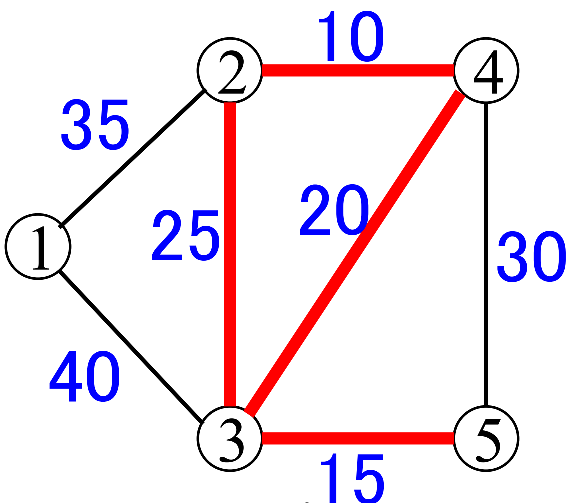
目的 設置費用合計→最小
制約 5軒にガスを供給

ガス管が繋がっている＋5軒を張っている

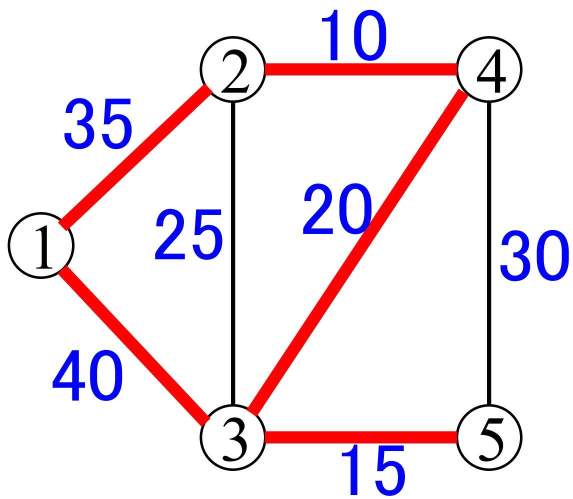
枝：設置可能路線
数字:設置費用

例題4（続）最適解でない例

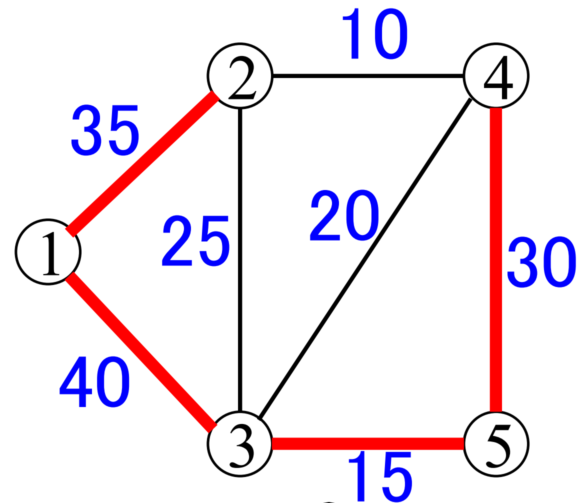
なぜ最適でないのか？



条件を満たしていない



自明な無駄がある



他に良いプランがある

実行不能

改善策

閉路は無駄
×閉路上の最大重み枝

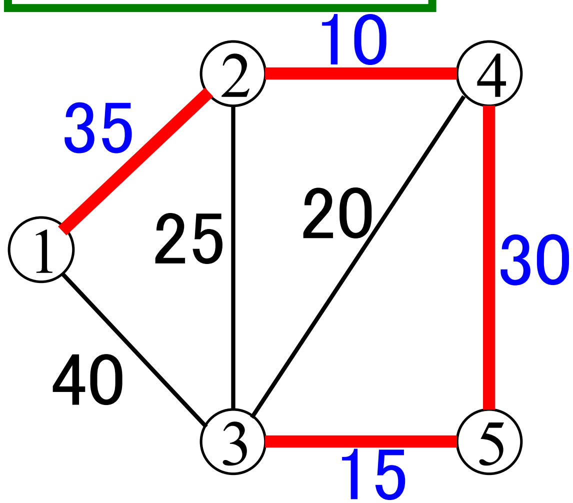
非連結部分を繋げる
○最小重み枝

例題4（続） 実行可能解が持つ性質

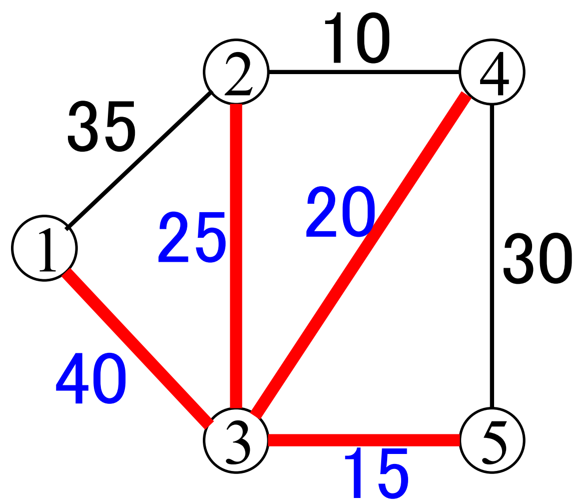
閉路は無駄 ⇒ 閉路の無いグラフ＝木
全点を結ぶ ⇒ 全張（spanning;スパンする）

全張木
spanning tree

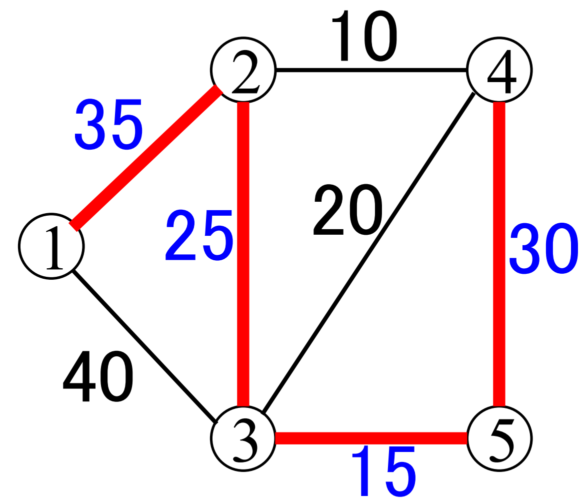
様々な全張木



35+10+30+15=90



40+25+20+15=100



35+25+30+15=105

問題の本質 重み和最小の全張木（最小木）を見つけよ

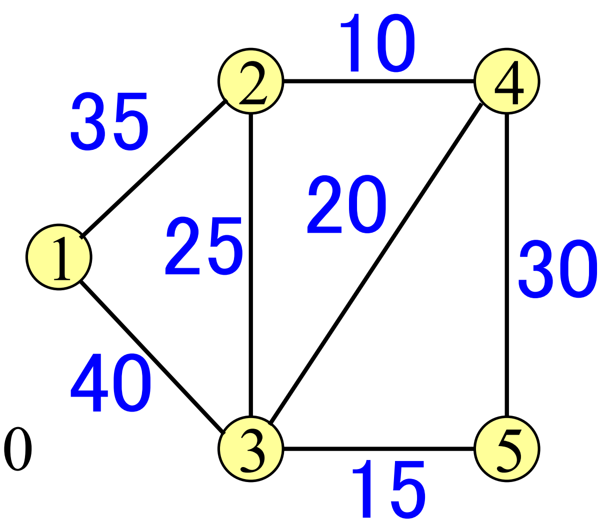
⇔ 最小木問題

Minimum spanning tree problem

例題4（続）最小木問題の定式化

解きたい問題

目的 利用枝の重みの和→最小
制約 利用枝は全点を結ぶ
利用枝に閉路がない



使用変数

x_{ij}:枝(i,j)を利用する時1，利用しない時0

目的関数

$$\min.\ z=35x_{12}+40x_{13}+25x_{23}+10x_{24}+20x_{34}+15x_{35}+30x_{45}$$

制約条件式は?

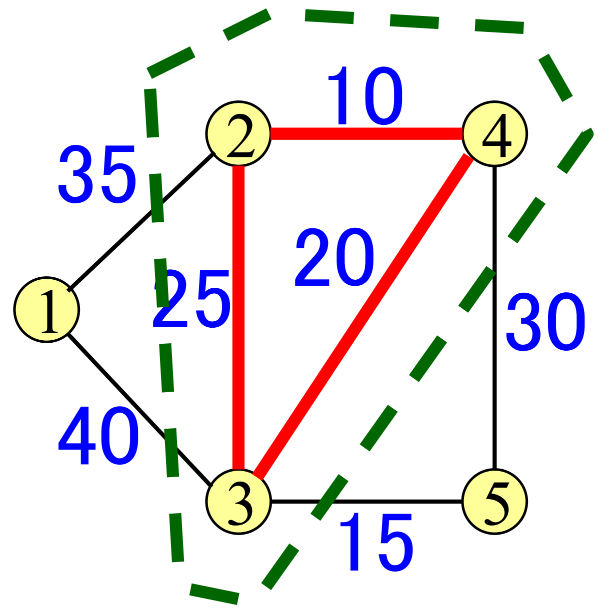
例題4(続)「閉路がない」の表現

閉路がない　⟷　閉路がある

（部分点集合内での使用枝数）＜（部分点集合の大きさ）

（部分点集合内での使用枝数）＝（部分点集合の大きさ）

（部分点集合内での使用枝数）≦（部分点集合の大きさ）－1



例 点部分集合{②,③,④}に対して

$$x_{23}+x_{24}+x_{34} \leqq 2$$

例題4（続）定式化

$$\min.\ z=35x_{12}+40x_{13}+25x_{23}+10x_{24}+20x_{34}+15x_{35}+30x_{45}$$

s.t. $x_{12}+x_{13}+x_{23}+x_{24}+x_{34}+x_{35}+x_{45}=4$

全部分集合に対して

（使用枝本数）≦（部分集合の大きさ）－１

$x_{12}, x_{13}, x_{23}, x_{24}, x_{34}, x_{35}, x_{45} \in \{0,1\}$

部分集合は$2^{（点数）}$個存在

定式化は可能だが
サイズ大で実際には扱えない

⇨ 使用枝の組合せを決める問題

⇨ 組合せ最適化問題 combinatorial optimization problem

離散最適化問題 discrete optimization problem

例題4（続）定式化

$$\min.\ z=35x_{12}+40x_{13}+25x_{23}+10x_{24}+20x_{34}+15x_{35}+30x_{45}$$

s.t. $x_{12}+x_{13}+x_{23}+x_{24}+x_{34}+x_{35}+x_{45}=4$

$x12+x13+x23 \leqq 2$

$x23+x24+x34 \leqq 2$

$x34+x35+x45 \leqq 2$

$x12+x13+x23+x24+x34 \leqq 3$

$x12+x13+x23+x35 \leqq 3$

$x13+x34+x35+x45 \leqq 3$

$x23+x24+x34+x35+x45 \leqq 3$

$x_{12}, x_{13}, x_{23}, x_{24}, x_{34}, x_{35}, x_{45} \in \{0,1\}$

定式化を利用しない

最小木の見つけ方：アイディア(1)

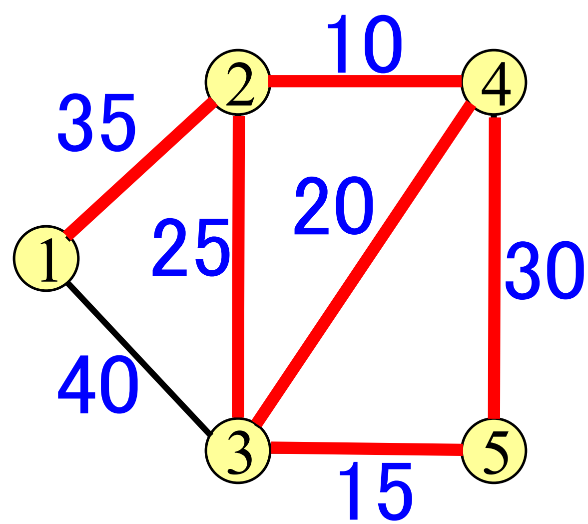
閉路⇒最大重みの枝を消去

実現方法例

重みの小さい順に枝を選択し
閉路になる時は選ばない
全点がつながったら終了

クラスカル法

(Kruskal)



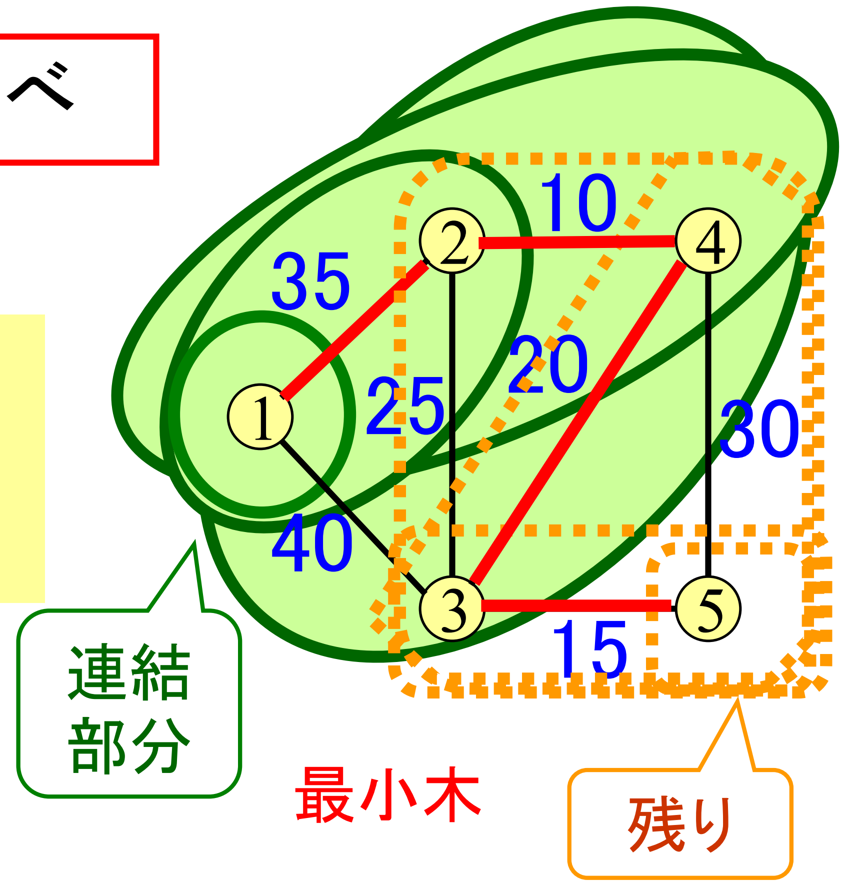
定式化を利用しない

最小木の見つけ方：アイディア(2)

非連結⇒最小重みの枝で結べ

実現方法例

1点から連結部分を1点ずつ
最小重みの枝で増やす
全点が連結になったら終了



プリム法

(Prim)

最小木問題に対する解法の計算量

n:グラフの点数
m:グラフの枝数

クラスカル法

- 閉路の発見に集合の併合操作 $O(m+n\log n)$
- データ構造の改造で $O(m\alpha(m,n))+O(m\log n)$

Ackermann逆関数

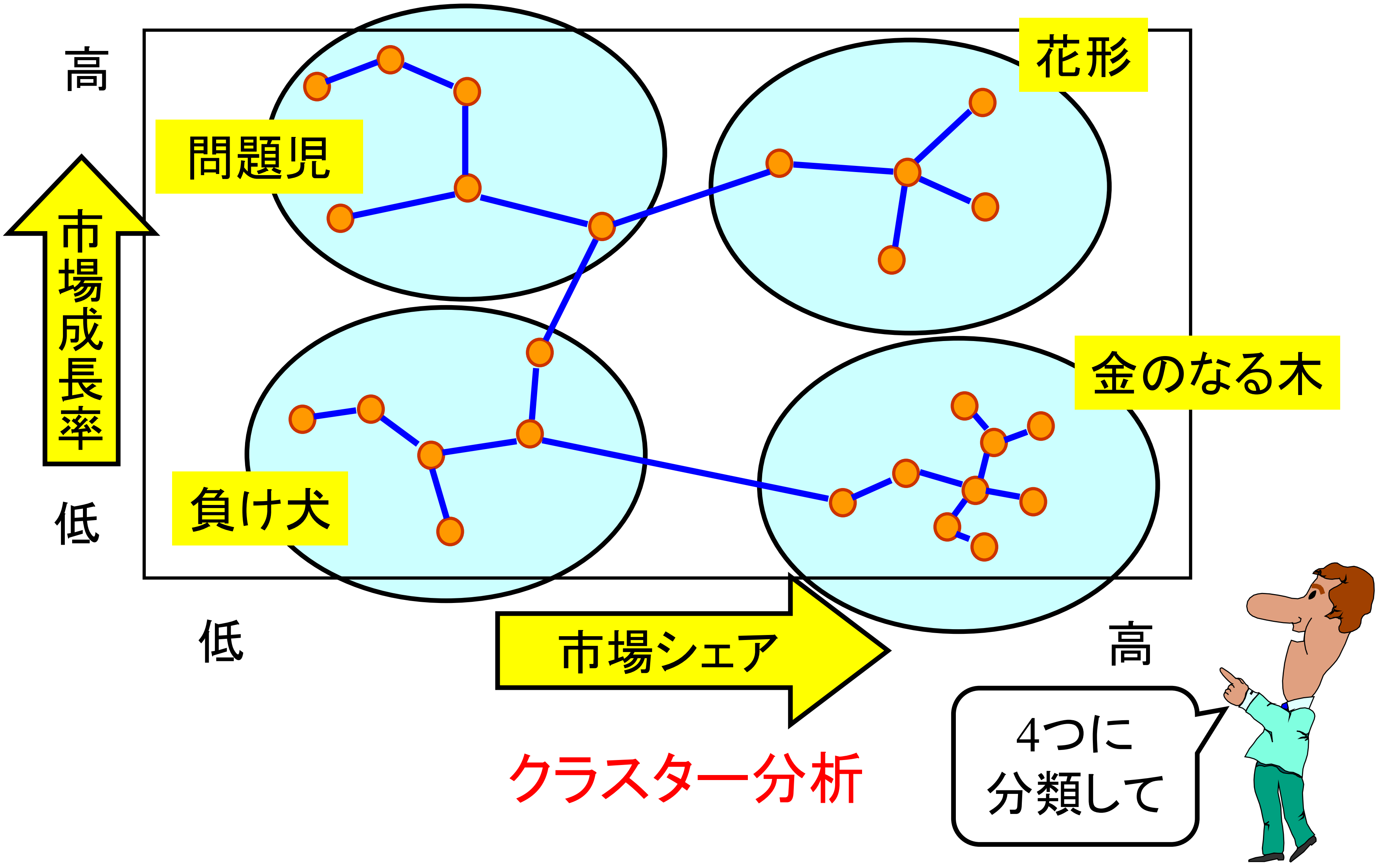
プリム法

- $O(n^2)$ ⇒Fibonacciヒープの利用 $O(m+n\log n)$
- 改良 $O(m\beta(m,n))$ ⇒さらに改良$O(m\log\beta(m,n))$

O(m) 線形時間解法はあるか?

◎存在確認
◎平面グラフ

最小木問題の利用例

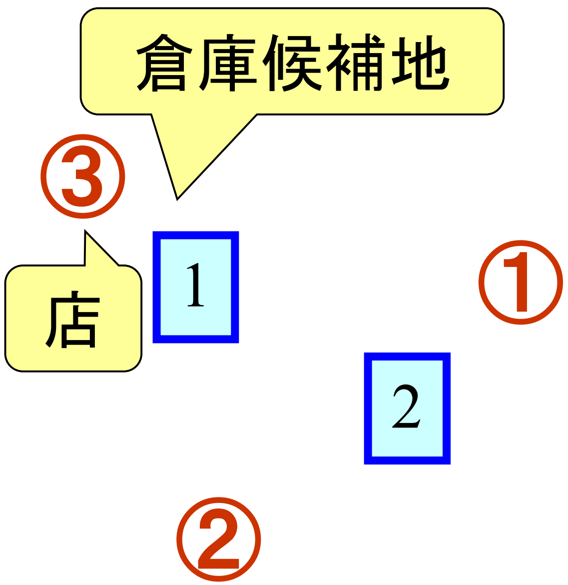


クラスター分析

演習3 定式化せよ

施設配置問題

（建設費）＋（配送費）を最小にしたい．
どこに倉庫を建設し，
どのように配送すればよいか．
この問題を定式化せよ．



ヒント

コントロールできるもの

$0 \leqq x_{ij} \leqq 1$

倉庫iから店jへの配送量⇒0～1の値

倉庫iを建設する･しない⇒2値

$y_i \in \{0,1\}$

- 倉庫iの建設費用 f_i (i=1,2)
- 倉庫iと店j間の配送費用 c_{ij} (i=1,2 ; j=1,2,3)
- 各店の需要は1．分割配送可能．

さて今後の展開は

最もシンプルな数理計画問題

＝線形計画問題の最適解の導出法を考える

寄り道 組合せ最適化

◎組み合わせる（動詞）

意味が違う

× 組み合わせ最適化

× 組合わせ最適化

× 組合最適化