知の探究

5. 整数計画法と組合せ最適化

堀田 敬介

2018年12月11日（火）

整数計画問題とは？

※ここでは，目的関数・制約とも線形（1次式）の場合のみ考える

- 整数計画問題(Integer Programming Problem)
  - 等式・不等式系であらわされる条件のもとで，目的関数を最大・最小化する形式の最適化問題で変数に整数条件が付く

$$
\begin{aligned}
\text{min.}\quad & 2x_1 + x_2 + 2x_3 + x_4 + 3x_5 \\
\text{s.t.}\quad & x_1 + 2x_3 + x_5 \geq 5 \\
& 9x_1 + 2x_2 + x_4 + 4x_5 \geq 1 \\
& x_2 + 5x_3 + x_5 \geq 3 \\
& x_2 + 5x_3 + x_5 \geq 2 \\
& x_1, x_2, x_3, x_4, x_5 \geq 0 \\
& x_1, x_2, x_3, x_4, x_5 \in Z
\end{aligned}
$$

目的関数 objective function

制約条件 constraints

非負条件 nonnegativity

整数条件 integrality

- 整数計画問題の最適解を求めるための主な手法
  - 分子限定法，切除平面法，分子カット法，etc.

※特殊ケース（右辺整数＆係数行列が全単模等）を除きNP困難なので，最適解を厳密に求めるのは大変

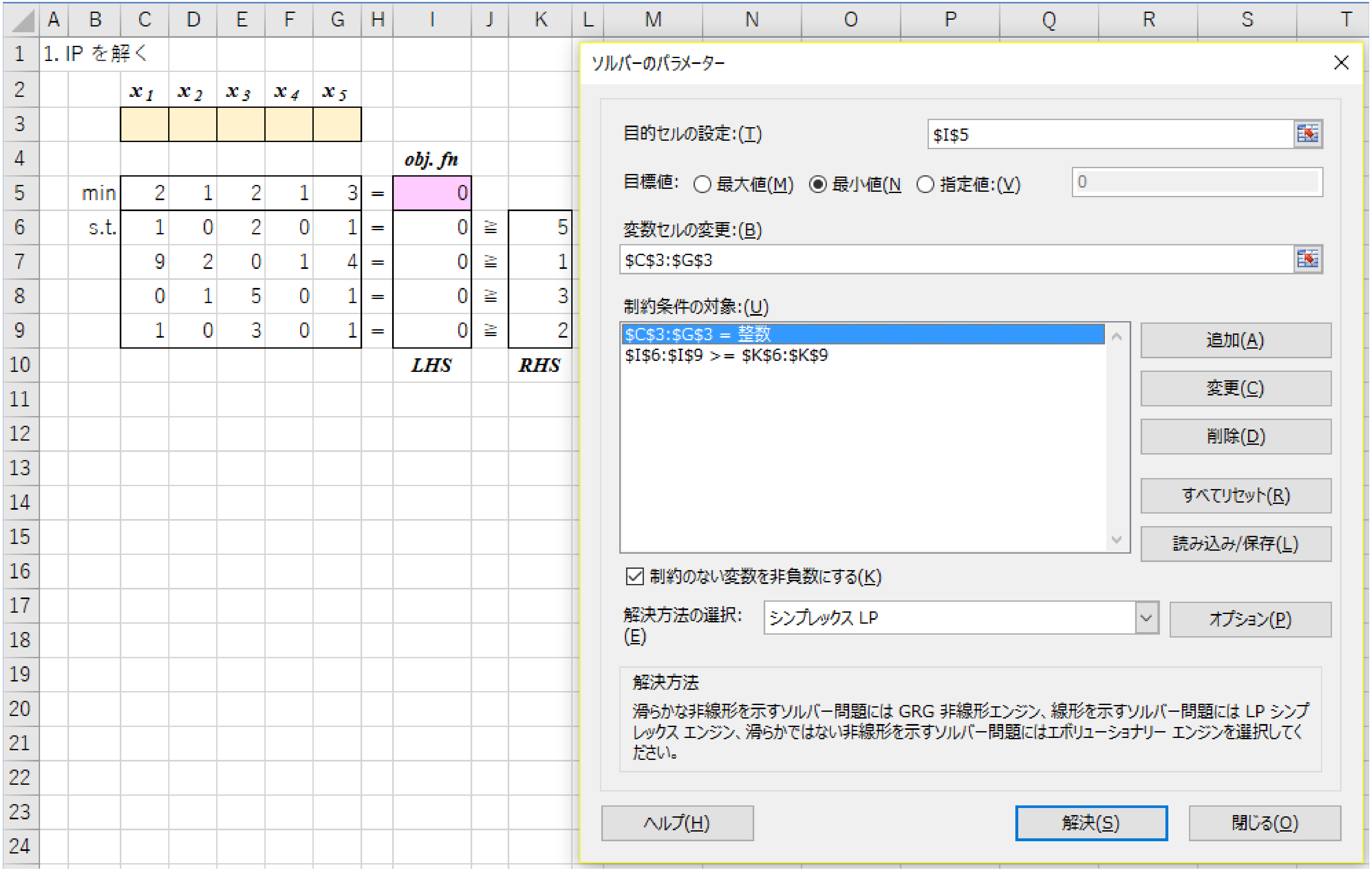
整数計画問題とは？

- 整数計画問題を Excel Solver で解く

$$
\begin{aligned}
\text{min.} \quad & 2x_1 + x_2 + 2x_3 + x_4 + 3x_5 \\
\text{s.t.} \quad & x_1 + 2x_3 + x_5 \geq 5 \\
& 9x_1 + 2x_2 + x_4 + 4x_5 \geq 1 \\
& x_2 + 5x_3 + x_5 \geq 3 \\
& x_1 + 3x_3 + x_5 \geq 2 \\
& x_1, x_2, x_3, x_4, x_5 \geq 0, {\color{red} x_1, x_2, x_3, x_4, x_5 \in Z}
\end{aligned}
$$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	1. IP を解く															
2			x_1	x_2	x_3	x_4	x_5									
3																
4									*obj. fn*				数式			
5		min	2	1	2	1	3	=	0				[I5] = SUMPRODUCT(C$3:G$3, C5:G5)			
6		s.t.	1	0	2	0	1	=	0	≧	5		↓			
7			9	2	0	1	4	=	0	≧	1		[I6]～[I9]へコピー			
8			0	1	5	0	1	=	0	≧	3		↓			
9			1	0	3	0	1	=	0	≧	2		↓			
10									*LHS*		*RHS*					

整数計画問題とは？


1. IP を解く
x1 x2 x3 x4 x5
obj. fn
min 2 1 2 1 3 = 0
s.t. 1 0 2 0 1 = 0 ≧ 5
9 2 0 1 4 = 0 ≧ 1
0 1 5 0 1 = 0 ≧ 3
1 0 3 0 1 = 0 ≧ 2
LHS
RHS
ソルバーのパラメーター
目的セルの設定:(T)
$I$5
目標値: 最大値(M) 最小値(N) 指定値:(V)
0
変数セルの変更:(B)
$C$3:$G$3
制約条件の対象:(U)
$C$3:$G$3 = 整数
$I$6:$I$9 >= $K$6:$K$9
追加(A)
変更(C)
削除(D)
すべてリセット(R)
読み込み/保存(L)
制約のない変数を非負数にする(K)
解決方法の選択:(E)
シンプレックス LP
オプション(P)
解決方法
滑らかな非線形を示すソルバー問題には GRG 非線形エンジン、線形を示すソルバー問題には LP シンプレックス エンジン、滑らかではない非線形を示すソルバー問題にはエボリューショナリー エンジンを選択してください。
ヘルプ(H)
解決(S)
閉じる(O)

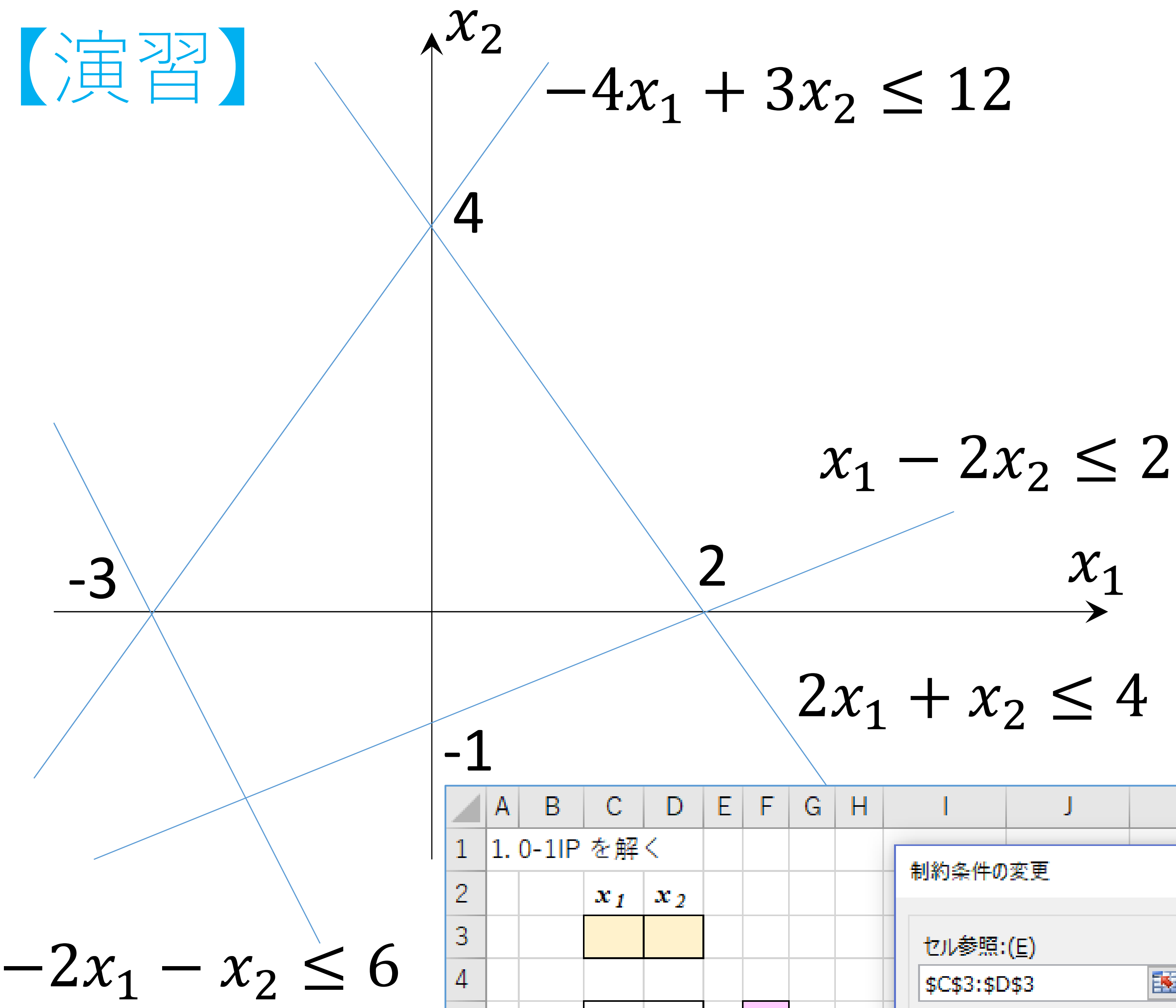
1. 0−1整数計画を解く

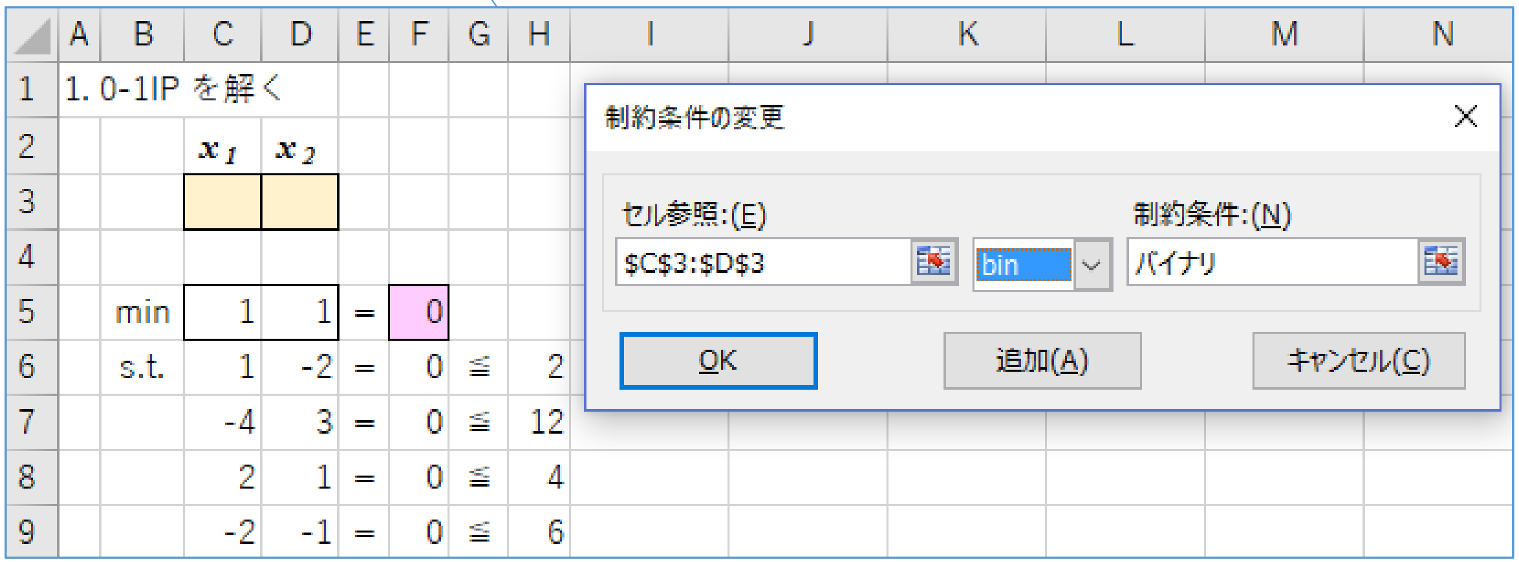
【演習】

- 以下の0-1IPについてExcel solverで最適解を求めよ

$$
\begin{aligned}
\text{min.} \quad & x_1 + x_2 \\
\text{s.t.} \quad & -4x_1 + 3x_2 \leq 12 \\
& x_1 - 2x_2 \leq 2 \\
& 2x_1 + x_2 \leq 4 \\
& -2x_1 - x_2 \leq 6 \\
& x_1, x_2 \geq 0 \\
& x_1, x_2 \in \{0,1\}
\end{aligned}
$$

【演習】





	A	B	C	D	E	F	G	H
1	1. 0-1IP を解く							
2			x_1	x_2				
3								
4								
5		min	1	1	=	0		
6		s.t.	1	-2	=	0	≦	2
7			-4	3	=	0	≦	12
8			2	1	=	0	≦	4
9			-2	-1	=	0	≦	6

2.ナップサック問題

- ナップサック問題 ***knapsack problem***

n個の品物があり，ナップサック（1つ）に入れたい

各品物には重さがあり，価値がある

ナップサックには b kg まで，品物を入れることができる

価値総和が最大になるようにするには，どの品物を持って行けばよいか？

【演習】 0-1IPに定式化せよ

- 品物 $i = 1,2,\ldots,n$
- 品物 i の価値：w_i
- 品物 i の重さ：c_i
- ナップサック容量 b
- 0-1変数 $x_i = \begin{cases} 1 & \ldots \quad \text{品物 } i \text{ をナップサックに入れる} \\ 0 & \ldots \quad \textit{otherwise} \end{cases}$

2.ナップサック問題

【定式化：解答例】

- ナップサック問題 ***knapsack problem***

$$
\begin{array}{ll}
\text{min.} & w_1x_1 + w_2x_2 + \cdots + w_nx_n \\
\text{s.t.} & c_1x_1 + c_2x_2 + \cdots + c_nx_n \leq b \\
& x_i \in \{0,1\} \quad (\mathrm{i} = 1, \cdots, \mathrm{n})
\end{array}
$$

2.ナップサック問題

- 例題：ナップサック問題

100個の品物があり，ナップサック（1つ）に入れたい

各品物には，重さがあり，価値がある

ナップサックには40kg まで，品物を入れることができる

価値総和が最大になるようにするには，どの品物を持って行けばよいか？

【演習】 Excel Solver で求解せよ

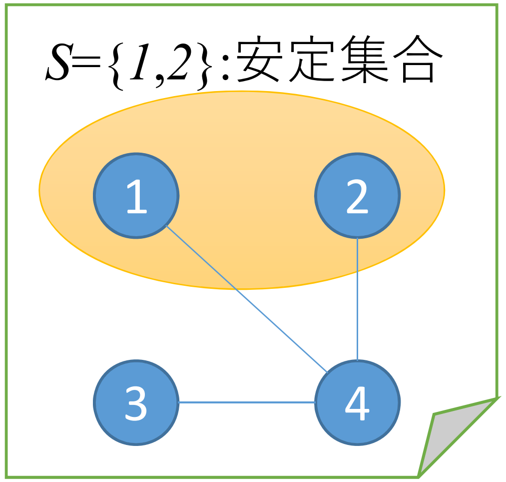
3.安定集合

- 最大安定集合問題 ***maximum stable set problem***

無向グラフ $G=(V,E)$　（V={1,2,...,n}）

について，要素数が最大となる安定集合 S を求めなさい

※点の部分集合 S（$S\subseteq V$）が安定集合（stable set）⇔ S内の任意の2点間に枝がない



【演習】 0-1IPに定式化せよ

- 点集合 $V=\{1,2,\ldots,n\}$

- 0-1変数 $x_i = \begin{cases} 1 & \cdots \quad \text{点 } i \text{ が安定集合 } S \text{ に含まれる} \\ 0 & \cdots \quad \textit{otherwise} \end{cases}$

3.安定集合

【定式化：解答例】

- 最大安定集合問題 ***maximum stable set problem***

$$
\begin{array}{ll}
\text{min.} & x_1 + x_2 + \cdots + x_n \\
\text{s.t.} & x_i + x_j \leq 1 \quad (\forall (i,j) \in E) \\
 & x_i \in \{0,1\} \quad (\forall \mathrm{i} \in V)
\end{array}
$$

3.安定集合

- 例題：最大安定集合問題

10人の学生がいる

人数が最大の仲良しグループをつくれ

※学生を点とし，仲が悪い学生間に枝を張ると，最大安定集合問題となる

【演習】 Excel Solver で求解せよ

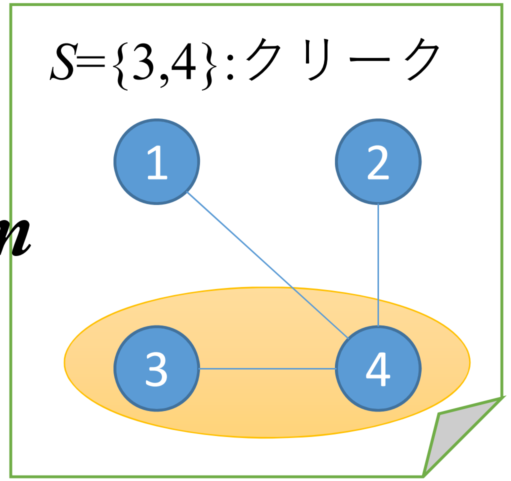
3.安定集合

- 最大クリーク問題 ***maximum clique problem***

無向グラフ $G=(V,E)$　（V={1,2,...,n}）

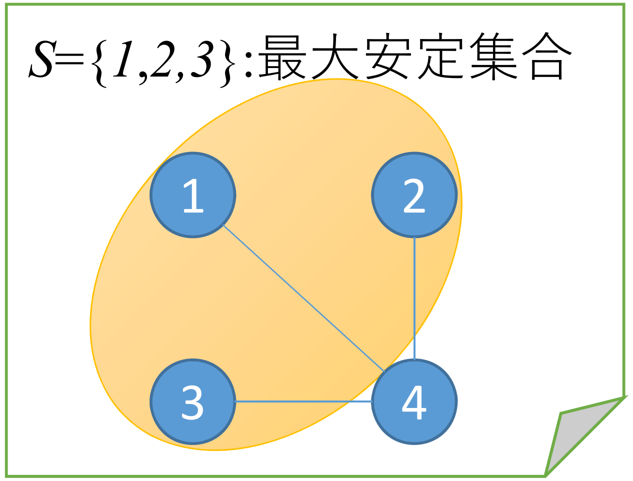
について，要素数が最大となるクリーク C を求めなさい

※点の部分集合 C（$C \subseteq V$）がクリーク ⇔ Cによる誘導部分グラフが完全グラフ



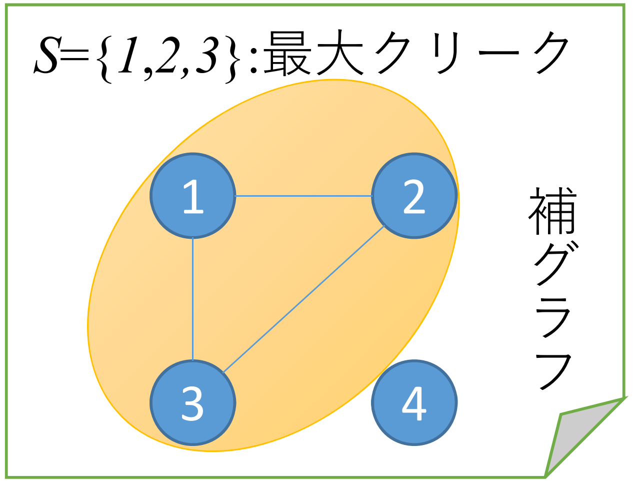
【演習】 0-1IPに定式化せよ

- 点集合 $V=\{1,2,\ldots,n\}$
- 0-1変数 $x_i = \begin{cases} 1 & \ldots \text{点 } i \text{ がクリーク } C \text{ に含まれる} \\ 0 & \ldots \textit{otherwise} \end{cases}$



【補足】

無向グラフ $G=(V,E)$ 上の最大安定集合問題
＝補グラフ$\bar{G}=(V,\bar{E})$ 上の最大クリーク問題



3.安定集合

【定式化例】

- 最大クリーク問題 ***maximum clique problem***

$$
\begin{aligned}
\text{min.} \quad & x_1 + x_2 + \cdots + x_n \\
\text{s.t.} \quad & \left.\begin{array}{l} y_{ij} \le x_i \\ y_{ij} \le x_j \end{array}\right. (\forall (i,j) \in E \; with \; i < j) \\
& \sum_{i<j} y_{ij} \ge k \sum_i x_i - \sum_{i=1}^{k-1} i \;\; (k = 2,3,\dots,n) \\
& x_i \in \{0,1\} \quad (\forall \mathrm{i} \in V) \\
& y_{ij} \in \{0,1\} \quad (\forall (\mathrm{i},\mathrm{j}) \in E \text{ with } \mathrm{i} < \mathrm{j})
\end{aligned}
$$

4.集合分割

- 集合分割問題 ***set partition problem***

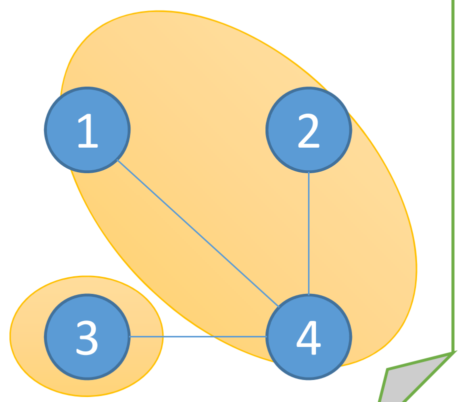
集合 V を m 個に分割しなさい

V={1,2,3,4}, m=2
→ S_1={1,2,4}:分割1
S_2={3}:分割2

※def) Vの部分集合族$\{V_1,\ldots,V_m\}$がVの分割 ⇔ $\bigcup_{i=1}^{m} V_i = V,\ V_i \cap V_j = \emptyset(\forall i,j; i \neq j)$

※連結成分分割（無向グラフ $G=(V,E)$ を m 個の連結成分に分割せよ）

S_1={1,2,4}:連結成分1
S_2={3}:連結成分2



【演習】 0-1IPに定式化せよ

- Vの部分集合 $V_i\ (i=1,2,\ldots)$
- V_i を表す特性ベクトル $a_i \in \boldsymbol{R}^{|V|}$
- 0-1変数 $x_i = \begin{cases} 1 & \ldots\ V_i \text{ を分割の構成要素として使う} \\ 0 & \ldots\ otherwise \end{cases}$

4.集合分割

【定式化：解答例】

- 集合分割問題 ***set partition problem***

min. or max. 問題による何らかの目的

s.t. $$a_1 x_1 + a_1 x_1 + \cdots = 1$$

$$x_i \in \{0,1\} \quad (i = 1,2,\cdots)$$

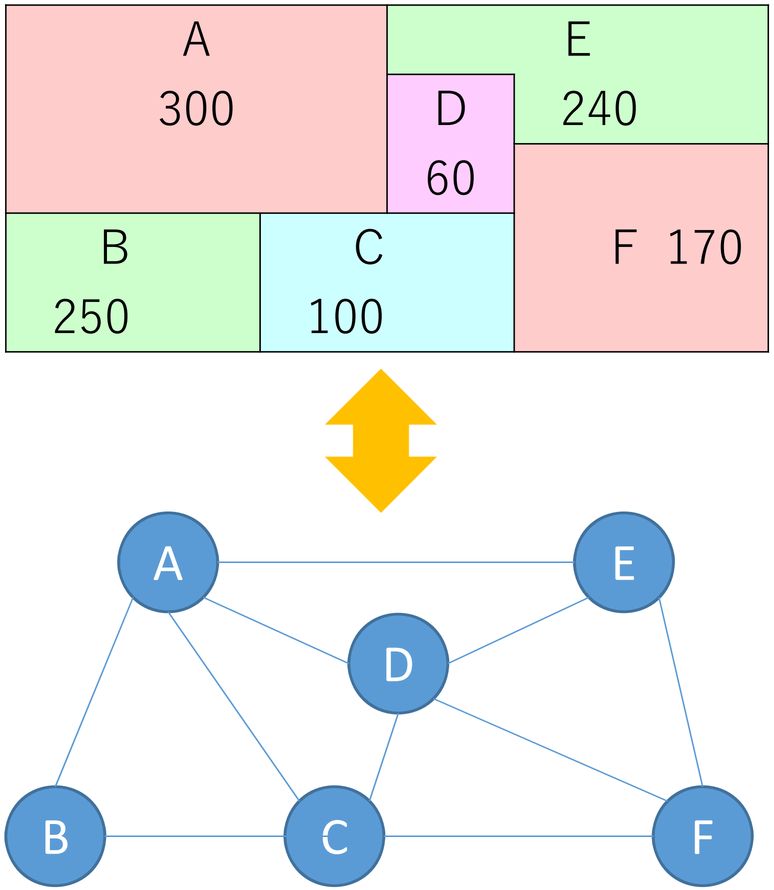
4.集合分割

- 例題：連結成分分割問題（飛び地なし集合分割問題）

6つの地区（A,B,...,F）があり，各地区の人口が与えられている

3つの地域に分割したい（ただし，分割後の各地域は連結であること）

最大人口の地域と最小人口の地域の差を最小とする分割を求めよ



【演習】 Excel Solver で求解せよ

- Vの部分集合 V_i $(i = 1,2,...)$
- V_i を表す特性ベクトル $a_i \in \boldsymbol{R}^{|V|}$

- V_i の人口 p_i
- 0-1変数 x_i
- 実数変数 u, l …分割人口の上限と下限

4.集合分割

【定式化：例題の解答例】

- 集合分割問題 ***set partition problem***

$$
\begin{aligned}
\text{min.} \quad & u - l \\
\text{s.t.} \quad & a_1 x_1 + a_2 x_2 + \cdots = 1 \\
& p_i x_i \le u \ (i = 1,2,\cdots) \\
& p_i x_i + \bar{p}(1 - x_i) \ge l \ (i = 1,2,\cdots) \\
& x_i \in \{0,1\} \quad (i = 1,2,\cdots)
\end{aligned}
$$

x_iは第i地域を採用するかどうかの0-1変数
1番目の制約式は，各地区は1回のみ使用，つまりたくさんある地域の内3つだけ採用（値が1になる. 他は全部0）
2番目の制約式は，3つの地域の最大人口をu以下とする
3番目の制約式は，3つの地域の最小人口をl以上とする
3番目が2番目と違い，余計な項がついているのは，x_iの値が0の場合について対処するため．この余分な項がないと，lの値が0以下になって意味をなさないことに注意せよ

$$a_1 = \begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix}, a_2 = \begin{bmatrix}1\\1\\0\\0\\0\\0\end{bmatrix}, \ldots$$

iは分割のパターン（地域）を表すことに注意
例えば
- ✓ a_1は地区Aのみを含む地域
- ✓ a_2は地区A,Bからなる地域

…
であり，各地域の人口は
p_1=300, p_2=550, …

$$\bar{p} := \frac{1120}{3} = 373$$

(1分割あたり平均人口)

5.巡回セールスマン

TSP = Traveling Salesman Problem

- 対称巡回セールスマン問題　***symmetric TSP***

無向グラフ $G = (V, E)$　（V={1,2,...,n}）

について，全ての点を丁度1度ずつ経由するコスト最小の巡回路を求めよ

【演習】0-1IPに定式化せよ

- 点集合 $V = \{1,2,\ldots,n\}$, 枝$e \in E$上のコストc_e
- 0-1変数 $x_e = \begin{cases} 1 & \cdots \quad \text{巡回路として枝 e} \in E \text{ を通る} \\ 0 & \cdots \quad \textit{otherwise} \end{cases}$

5.巡回セールスマン

- 定式化

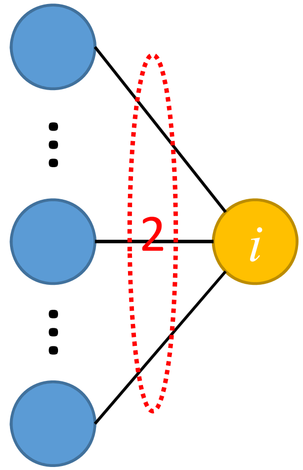
$$
\text{min.} \quad \sum_{e \in E} c_e x_e
$$

$$
\text{s.t.} \quad \sum_{e \in \delta(\{i\})} x_e = 2 \ (\forall i \in \{1, \ldots, n\})
$$

$$
x_e \in \{0,1\} \ \ (\forall e \in E)
$$

※ $\delta(\{i\})$：端点の一つがiとなる枝集合

この制約の意図



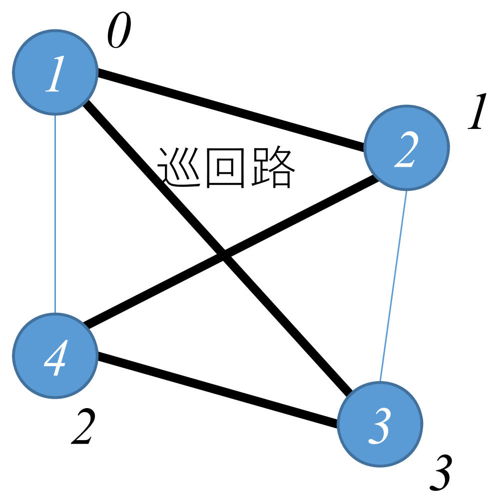
しかし，この制約だけでは部分巡回路が含まれる

部分巡回路をどう回避する？

- ✓ 部分巡回路除去定式化
- ✓ ポテンシャル定式化
- ✓ 単一品種流定式化
- ✓ 多品種流定式化
- ✓ etc.

5.巡回セールスマン

部分巡回路を除去する



- 部分巡回路除去定式化

$$\text{min.} \quad \sum_{e \in E} c_e x_e$$

$$\text{s.t.} \quad \sum_{e \in \delta(\{i\})} x_e = 2 \ (\forall i \in \{1, \dots, n\})$$

$$x_e \in \{0,1\} \ \ (\forall e \in E)$$

部分巡回路が
含まれる

$$\boxed{\sum_{e \in E(S)} x_e \leq |S| - 1 \ (\forall S \subset V, |S| \geq 2)}$$

※ S：点集合Vの任意の真部分集合（つまり$S \neq V$）
※ $E(S)$：両端点がSに含まれる枝の集合

5.巡回セールスマン

- 非対称巡回セールスマン問題　***asymmetric TSP***

有向グラフ $G=(V,E)$　（V={1,2,...,n}）

について，全ての点を丁度1度ずつ経由するコスト最小の巡回路を求めよ

【演習】 0-1IPに定式化せよ

- 点集合 $V=\{1,2,\ldots,n\}$, 枝$(i,j)\in E$上のコストc_{ij}
- 0-1変数 $x_{ij}=\begin{cases}1\ldots & \text{巡回路として枝 } (i,j)\in E \text{ を} i\rightarrow j \text{の順に通る} \\ 0\ldots & otherwise\end{cases}$

5.巡回セールスマン

- 定式化

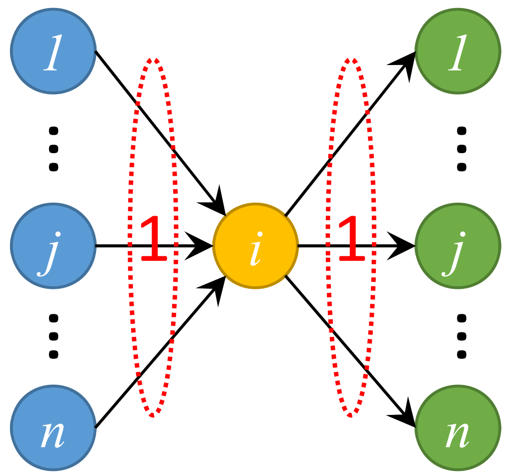
$$\text{min.} \quad \sum_{i \neq j} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{j:j\neq i} x_{ij} = 1 \ (\forall i \in \{1,\dots,n\})$$

$$\sum_{j:j\neq i} x_{ji} = 1 \ (\forall i \in \{1,\dots,n\})$$

$$x_{ij} \in \{0,1\} \ \ (\forall i \neq j)$$

この制約の意図



しかし，この制約だけでは部分巡回路が含まれる

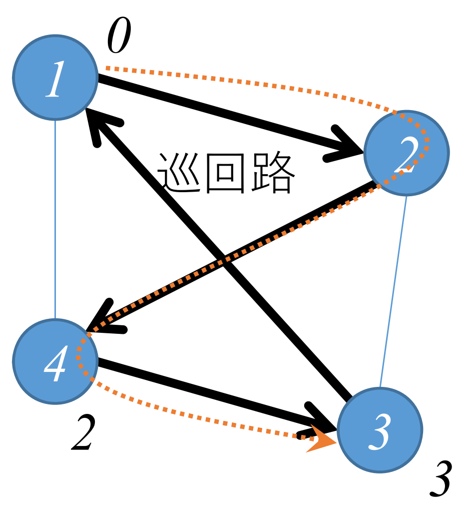
部分巡回路をどう回避する？

- ✓ 部分巡回路除去定式化
- ✓ ポテンシャル定式化
- ✓ 単一品種流定式化
- ✓ 多品種流定式化
- ✓ etc.

5.巡回セールスマン

- ポテンシャル定式化

巡回路に訪問順ラベルがつく



$$\text{min.} \quad \sum_{i \neq j} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{j:j \neq i} x_{ij} = 1 \ (\forall i \in \{1, \ldots, n\})$$

$$\sum_{j:j \neq i} x_{ji} = 1 \ (\forall i \in \{1, \ldots, n\})$$

$$x_{ij} \in \{0,1\} \ \ (\forall i \neq j)$$

部分巡回路が含まれる

$$u_i + 1 \ - (n-1)(1 - x_{ij}) \leq u_j \ (\forall i = 1, \ldots, n; j = 2, \ldots n; i \neq j)$$
$$1 \leq u_i \leq n - 1 \ (\forall i = 2, \ldots, n)$$

u_i：点iの訪問順序を表す実数変数

- u_1=0
- $i \rightarrow j$の順に訪問するとき $u_j = u_i$+1

5.巡回セールスマン

- 単一品種流定式化

$$\text{min.} \quad \sum_{i \neq j} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{j:j \neq i} x_{ij} = 1 \ (\forall i \in \{1, \dots, n\})$$

$$\sum_{j:j \neq i} x_{ji} = 1 \ (\forall i \in \{1, \dots, n\})$$

$$x_{ij} \in \{0,1\} \ \ (\forall i \neq j)$$

部分巡回路が
含まれる

$$\sum_j f_{1j} = n - 1$$

$$\sum_j f_{ji} - \sum_j f_{ij} = 1 \ (\forall i = 2, \dots, n)$$

$$f_{1j} \leq (n-1) x_{1j} \ (\forall j \neq 1)$$

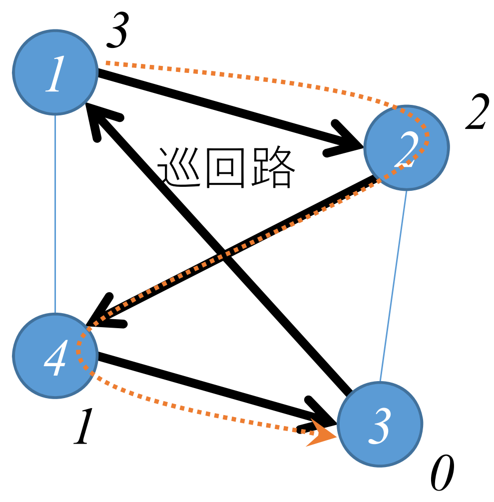
$$f_{ij} \leq (n-2) x_{ij} \ (\forall i \neq j, i \neq 1, j \neq 1)$$

$$f_{ij} \geq 0 \ (\forall i \neq j)$$

f_{ij}：点$i \rightarrow j$のフロー

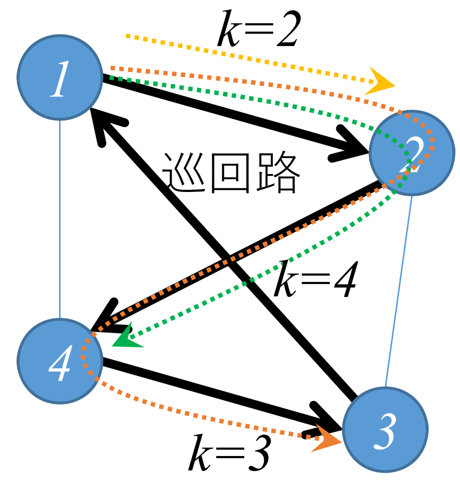
- 点1から n-1 のフローを流す
- 各点では 1 消費する
- フローは巡回路上のみ流れる

点1から3のフローを流す
各点は1ずつ消費



5.巡回セールスマン

- 多品種流定式化



$$\text{min.} \quad \sum_{i \neq j} c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{j:j \neq i} x_{ij} = 1 \ (\forall i \in \{1, \ldots, n\})$$

$$\sum_{j:j \neq i} x_{ji} = 1 \ (\forall i \in \{1, \ldots, n\})$$

$$x_{ij} \in \{0,1\} \ \ (\forall i \neq j)$$

部分巡回路が含まれる

$$\sum_j f_{ji}^k - \sum_j f_{ij}^k = \begin{cases} -1 (i = 1) \\ 0 (i \neq 1, k) \ \ (\forall k = 2, \ldots, n) \\ 1 (i = k) \end{cases}$$

$$f_{ij}^k \leq x_{ij} \ (\forall k, i, j)$$

$$f_{ij}^k \geq 0 \ (\forall i \neq j, j \neq 1, k)$$

f_{ij}^k：品種kの点i→jのフロー

- 点1から n-1種類のフローを流す
- 各品種のフローは全て1単位
- 各品種k=2,...,nは対応点2,...,nが受け取る（k=2は点2, k=3は点3,…, k=nは点nが受け取る）
- 各フローは巡回路上のみ流れる

6. 配送計画問題

VRP = *Vehicle Routing Problem*

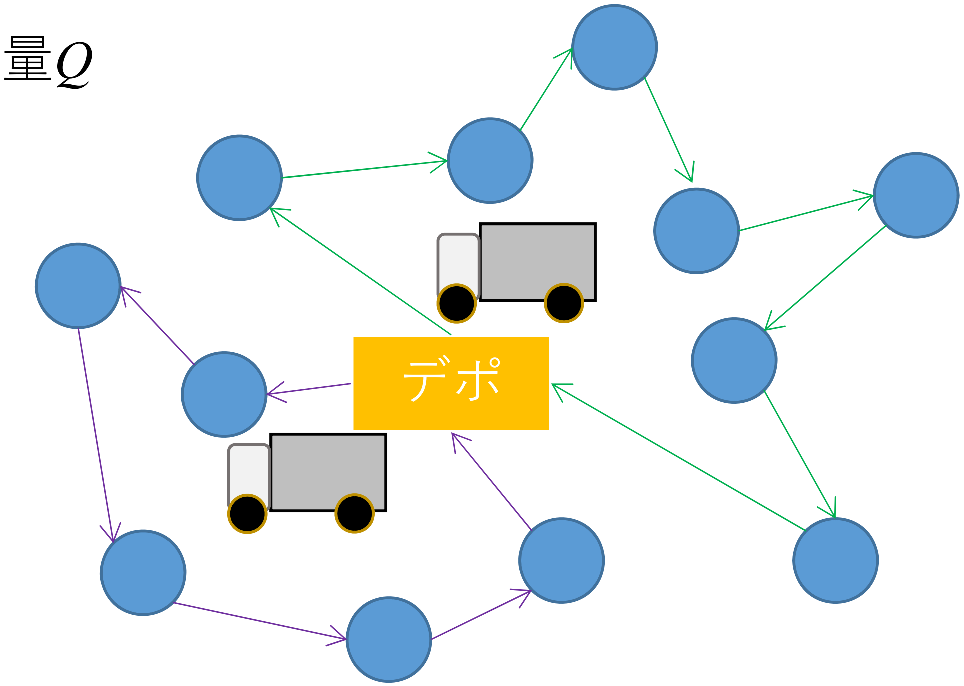
- 容量制約付き配送計画問題　***capacitated VRP***

無向グラフ $G = (V, E)$　（$V=\{1,2,...,n\}$）

について，デポ（配送センター(i=1)）から顧客(i=2,3,...,n) へ物資を輸送

配送車 k =1,2,...,m（m台）　各配送車の容量Q

顧客の需要q_i, 枝$(i, j)\in E$上の輸送コストc_{ij}



※ $\forall i, q_i \leq Q$として一般性を失わない
（もしq_i>Qとなる顧客がいたら
その顧客（点）を分割すればよい）
※ $\sum_i q_i \leq mQ$とする

【演習】 0-1IPに定式化せよ

- 0-1変数 $x_{ij} = \begin{cases} 1 \ldots & \text{輸送路として枝 } (i, j)\in E \text{ を使う（向きは不問）} \\ 0 \ldots & \textit{otherwise} \end{cases}$

6.配送計画問題

- 定式化

$$\text{min.} \quad \sum_i \sum_j c_{ij} x_{ij}$$

$$\text{s.t.} \quad \sum_{j:j \neq i} x_{ij} = b_i \ (\forall i \in \{1, \dots, n\})$$

$$\sum_{j:j \neq i} x_{ji} = b_i (\forall i \in \{1, \dots, n\})$$

$$x_{ij} \in \{0,1\} \ \ (\forall i \neq j)$$

部分巡回路が含まれる

- 点i=1はデポ（配送センター）
- 点i=2,…,nは顧客（顧客数n-1）
- デポから m台の配送車が往復
  → b_1=m
- 各顧客へは「届けて / 帰る」
  → b_i=1 (i=2,…,n)

$$\sum_{i,j \in S} x_{ij} \leq |S| - N(S) \ (\forall S \subset \{2, \dots, n\}; |S| \geq 3)$$

$$N(S) = \left\lceil \sum_{i \in S} q_i / Q \right\rceil$$

- 部分巡回路除去＆配送車容量制約
- S：顧客の部分集合
- $N(S)$：S内の顧客の総需要を満たすために必要な配送車の台数

7. スポーツ・スケジューリング

team/slot	1	2	3
A	B	C	D
B	A	D	C
C	D	A	B
D	C	B	A

- **single round-robin tournament**

nチームの総当たりリーグ戦

各チームは，他のチームと丁度1回ずつ対戦する．1チームあたりn-1試合(slot)

全てのチームがどちらかのホームで毎回試合を行う（1方がHome, 他方がAway）

何らかの目的（Break数最小化，巡回路長最小化, etc）のもとでスケジュールを組む

チームの集合 $T = \{ 1, 2, ..., n \}$　　※チーム数n は偶数のみと思って一般性を失わない

スロットの集合 $S = \{ 1, 2, ..., n\text{-}1 \}$　　※スロット数は「チーム数 - 1」

【演習】 0-1IPに定式化せよ

➢ 0-1変数 $x_{ijs} = \begin{cases} 1 & \dots \quad \text{チーム}\, i \text{ vs. } j \text{ の対戦を，} i \text{のHomeで，第 } s \text{ slotに実施} \\ 0 & \dots \quad otherwise \end{cases}$

7. スポーツ・スケジューリング

- 定式化

※Break数最小化 ＝ 非Break数最大化

min. $$\sum_{i}\sum_{j} c_{ij}x_{ij}$$

s.t. $$\sum_{j\in T/\{i\}} (x_{ijs} + x_{jis}) = 1 \ (\forall i \in T, \forall s \in S)$$

$$\sum_{s\in S} (x_{ijs} + x_{jis}) = 1 \ (\forall i, j \in T; i \neq j)$$

$$x_{ijs} \in \{0,1\} \ \ (\forall i, j, s; i \neq j)$$

- 制約1：各チームiはどのスロットsでも，Home/Awayのどちらかで1回戦う
- 制約2：任意の2チームi, j の試合 i vs. j は，どこかのスロットsで丁度1回行う

参考文献

1. 今野浩「線形計画法」日科技連（**1987**）
2. 藤田・今野・田邉「最適化法」岩波書店（**1994**）
3. 田村明久・村松正和「最適化法」共立出版（**2002**）
4. 坂和正敏「線形計画法の基礎と応用」朝倉書店（**2012**）
5. 小島・土谷・水野・矢部「内点法」朝倉書店（**2001**）
6. ***A. Schrijver: Theory of Linear and Integer Programming, John Wiley and Sons, 1986.***
7. ***L.A. Wolsey: Integer Programming, John Wiley and Sons, 1998.***
8. ***M. Conforti, G. Cornuejols and G.Zambelli: Integer Programming, Springer, 2014.***
9. 久保幹雄**, J.P.**ペドロソ**,** 村松正和**, A.**レイス：あたらしい数理最適化**,** 近代科学社**,2012.**
10. 久保幹雄**,** 小林和博**,** 斉藤努**,** 並木誠**,** 橋本英樹：**Python**言語によるビジネスアナリティクス**,** 近代科学社**, 2016.**
11. 藤澤克樹**,** 後藤順哉**,** 安井雄一郎：**Excel**で学ぶ**OR,** オーム社**, 2011.**