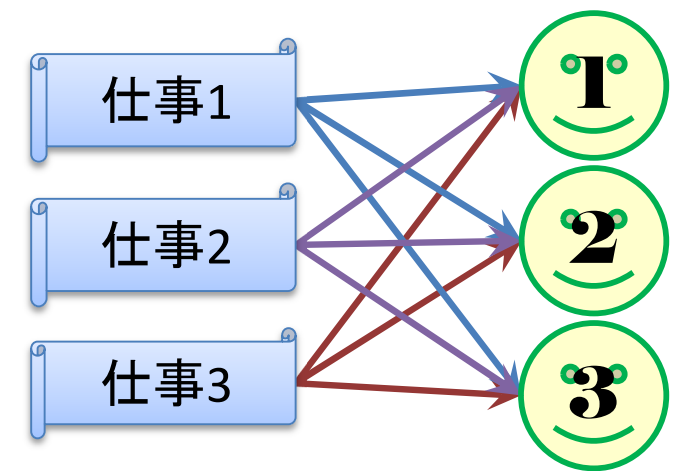


# 割当問題を解く

## ➤ 割当問題の最適化(例1)

- 今すべき仕事は3つあり, 3人の部下がいる
- 各部下は1つの仕事を引き受けられる
- 上司は各仕事を任せたときの部下の出来具合を5段階で評価済(右下表)
- この部署のパフォーマンスが最大になるように部下へ仕事を割り当てよ



## ➤ 最適化問題の定式化(変数設定)

- 0-1変数  $x_{ij} = \begin{cases} 1 & \dots \text{仕事}i\text{を部下}j\text{へ割り当てる} \\ 0 & \dots \text{仕事}i\text{を部下}j\text{へ割り当てない} \end{cases}$

- 変数行列  $X$

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$$

仕事 \ 部下	1	2	3
1	2	2	4
2	5	3	2
3	4	1	3

## ➤ 最適化問題の定式化(係数表記)

- 評価を表す係数行列  $C$

$$C = \begin{pmatrix} 2 & 2 & 4 \\ 5 & 3 & 2 \\ 4 & 1 & 3 \end{pmatrix}$$

# 割当問題を解く

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$$

$$C = \begin{pmatrix} 2 & 2 & 4 \\ 5 & 3 & 2 \\ 4 & 1 & 3 \end{pmatrix}$$

## ➤ 割当問題の最適化 (ベタ表記)

$$\max. 2x_{11} + 2x_{12} + 4x_{13} + 5x_{21} + 3x_{22} + 2x_{23} + 4x_{31} + 1x_{32} + 3x_{33}$$

$$\text{s. t. } x_{11} + x_{12} + x_{13} = 1$$

$$x_{21} + x_{22} + x_{23} = 1$$

$$x_{31} + x_{32} + x_{33} = 1$$

$$x_{11} + x_{21} + x_{31} \leq 1$$

$$x_{12} + x_{22} + x_{32} \leq 1$$

$$x_{13} + x_{23} + x_{33} \leq 1$$

$$x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23} \in \{0, 1\}$$

## ➤ 最適化問題の定式化 ( $\Sigma$ 表記)

$$\max. \sum_{i=1}^3 \sum_{j=1}^3 c_{ij} x_{ij}$$

$$\text{s. t. } \sum_{j=1}^3 x_{ij} = 1 \quad (i = 1, \dots, 3)$$

$$\sum_{i=1}^3 x_{ij} \leq 1 \quad (j = 1, \dots, 3)$$

$$x_{ij} \in \{0, 1\} \quad (i = 1, \dots, 3; j = 1, \dots, 3)$$

# How to use CPLEX?

## ➤ 新規プロジェクトの作成

- ① [ファイル(F)]－[新規(N)]－[OPLプロジェクト]を選択
- ② [プロジェクト名]を記入(例: **Assignment**)し, 3カ所にチェックする
  - デフォルトの実行構成の追加
  - モデルの作成
  - データの作成
- ③ [終了]をクリック

プロジェクト名は自由だが, **半角英数**で何の問題を解こうとしているのかが分かる名前が良い

## ➤ プロジェクト内のいくつかの名前を変更

- ✓ [構成1] → [config1] ※日本語を英語に変更しないと実行時エラーになる
- ✓ モデルファイル [Assignment.mod] → [as.mod]
- ✓ データファイル [Assignment.dat] → [asex1.dat]

## ➤ モデルファイル・データファイルを記述し保存(次ページ参照)

## ➤ [config1]にモデルファイルとデータファイルをセットする

# How to use CPLEX?

➤ as.mod

```
int i_max = ...; // 行の添え字の最大値
int j_max = ...; // 列の添え字の最大値

range I = 1..i_max; // 行の添え字の範囲 [1..i_max]を指定
range J = 1..j_max; // 列の添え字の範囲 [1..j_max]を指定

int c[I,J] = ...; // 評価値Cij(size:I×J)

dvar int+ x[I,J] in 0..1; // 変数宣言:0-1変数(size:I×J)

maximize
    sum(i in I) sum(j in J) c[i,j]*x[i,j];
subject to{
    forall(i in I) {
        sum(j in J) x[i,j] == 1;
    };
    forall(j in J) {
        sum(i in I) x[i,j] <= 1;
    };
};
```

# How to use CPLEX?

➤ asex1.dat

```
i_max = 3;  
j_max = 3;  
  
c = [  
[2 2 4]  
[5 3 2]  
[4 1 3]  
];
```

# How to use CPLEX?

## 計算結果の確認

[解]タブの中身

solution (optimal)  
解(最適)

最適解発見

[統計情報]タブの中身

目的 11 の解

名前	値
データ (5)	
c	[[2 2 4] [5 3 2] [4 1 3]]
l	1..3
i_max	3
J	1..3
j_max	3
決定変数 (1)	
x	[[0 0 1] [0 1 0] [1 0 0]]

```
// solution (optimal) with objective 11
// Quality Incumbent solution:
// MILP objective
// MILP solution norm |x| (Total, Max)
// MILP solution error (Ax=b) (Total, Max)
// MILP x bound error (Total, Max)
// MILP x integrality error (Total, Max)
// MILP slack bound error (Total, Max)
//
x = [[0
      0 1]
      [0 1 0]
      [1 0 0]];
```

最適値 = 11  
各定数(係数行列)  
最適解  $x = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$

LPではなく  
IPなので  
情報が少し  
違う

統計情報	値
solution (optimal) with objective 11	
Constraints	6
Variables	9
Binary	9
Non-zero coefficients	18
MIP	
Objective	11
Incumbent	11
Nodes	0
Remaining nodes	0
Iterations	2
Solution pool	
Count	2
Mean objective	9.5

■ 最良ノード ■ 最良整数 ■ 整数解

# 割当問題を解く2

## ➤ 割当問題の最適化(例2)

- 仕事は4つ, 部下は7人, 各部下は1つの仕事を引き受けられる
- 上司は部下を5段階[1,2,3,4,5]で評価済(下表)
- この部署のパフォーマンスが最大になるように部下へ仕事を割り当てよ

## ➤ 最適化問題の定式化(ベタ・Σ表記)

- 0-1変数  $x_{ij} = 1$  ... 仕事*i*を部下*j*へ割り当てる,  $x_{ij} = 0$  ... 割り当てない
- 係数matrix **C**:上司による部下評価値

仕事\部下	1	2	3	4	5	6	7
1	2	4	2	1	3	1	4
2	3	1	3	2	5	4	2
3	1	3	4	4	4	3	3
4	4	2	1	4	5	2	1

- CPLEXで解く(モデルファイル[as.mod]は共通で使えるので[asex2.dat]のみ作り解く)