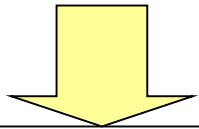


# Mathematical Programming (1)

問題発見・解決と数理モデル

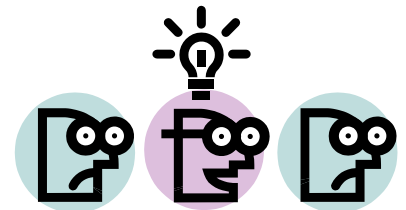
# ここで学ぶこと

- 数理計画とは
- システム的アプローチによる問題解決



次回

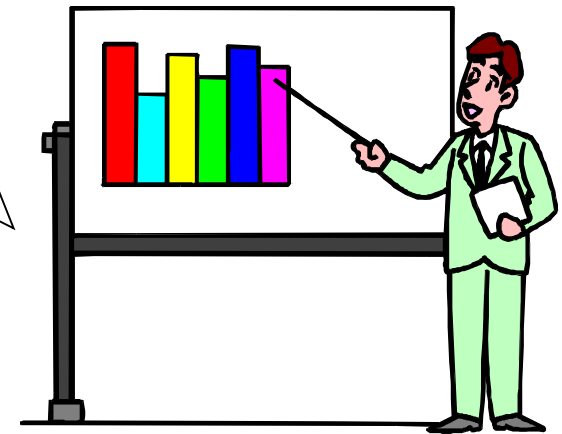
- 数理モデル化と表現方法(定式化)
- 数理計画問題の分類



# 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  
(一 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。



# 数理計画とORの深い関係

Operations Research

数理計画(問題)

与えられた**制約式**のもとで、  
ある**関数**を**最大(最小)**にする

ORとは？

比較

対象を数学的にモデル化し、有用な解決策を導く方法

ORの例: 経営の問題

ORは解決策を導く手法の宝庫

与えられた**資源内**で  
**利益**を**最大(費用・リスクを最小)**にする

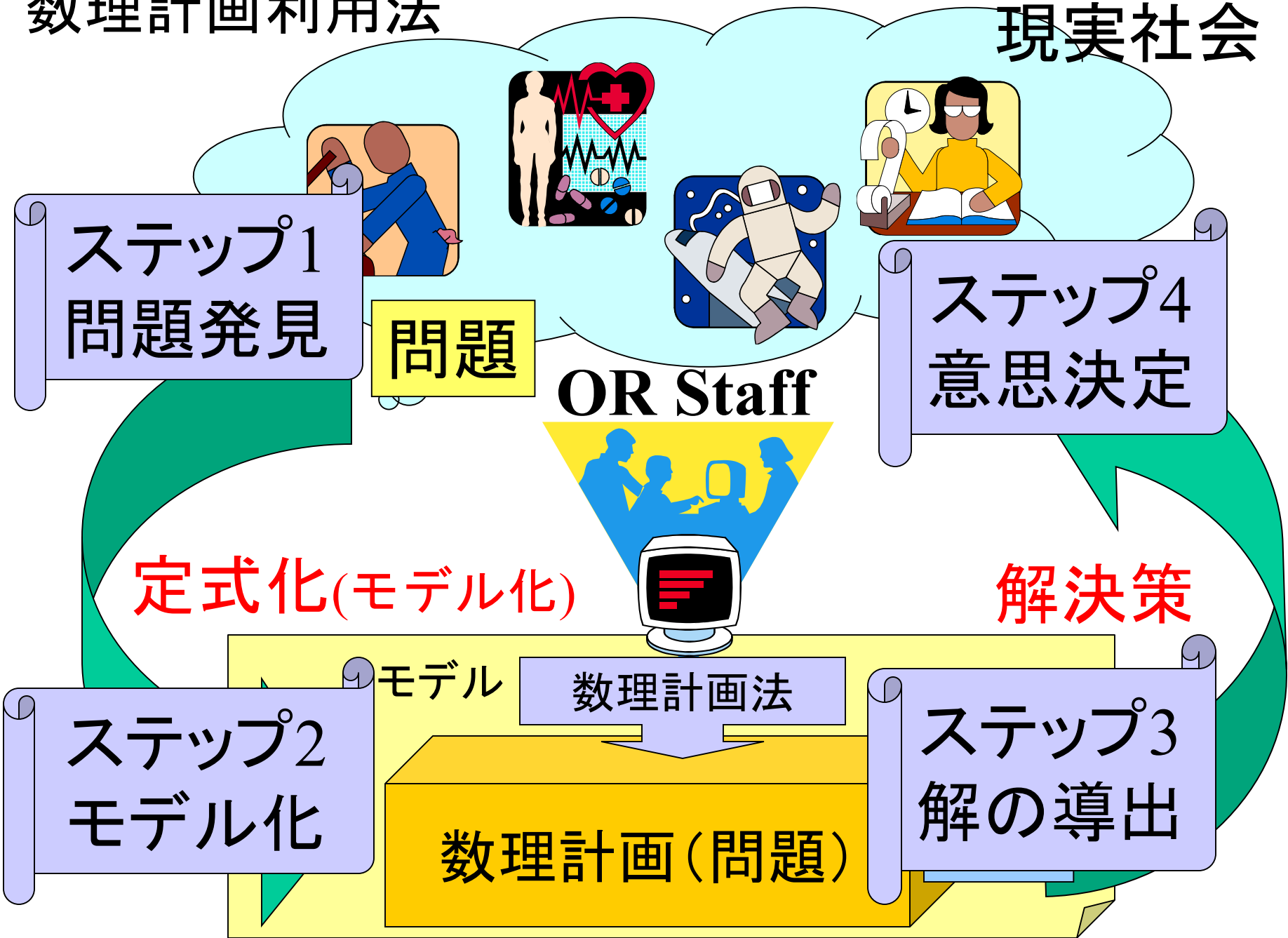
⇒数理計画は数理的な問題解決(OR)の中心的な技法として定着

# ORの仕組み



# 数理計画利用法


現実社会



# ステップ1 問題発見

何が問題？  
因果関係は？

問題発見技法

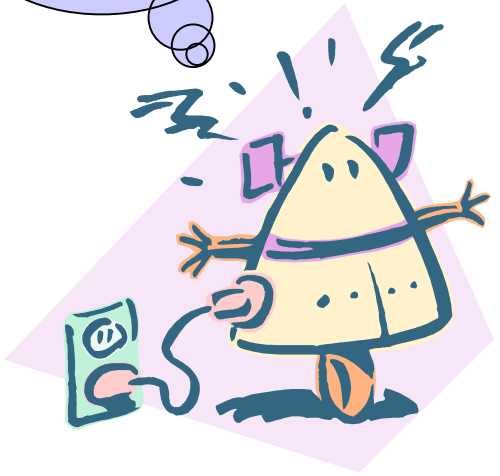
- ブレインストーミング法
- KJ法 
- QC7つ道具, 新QC7つ道具 など

Quality Control [品質管理]

問題の舞台をシステムで把握



問題は与えられることも多いが、発見することも重要！

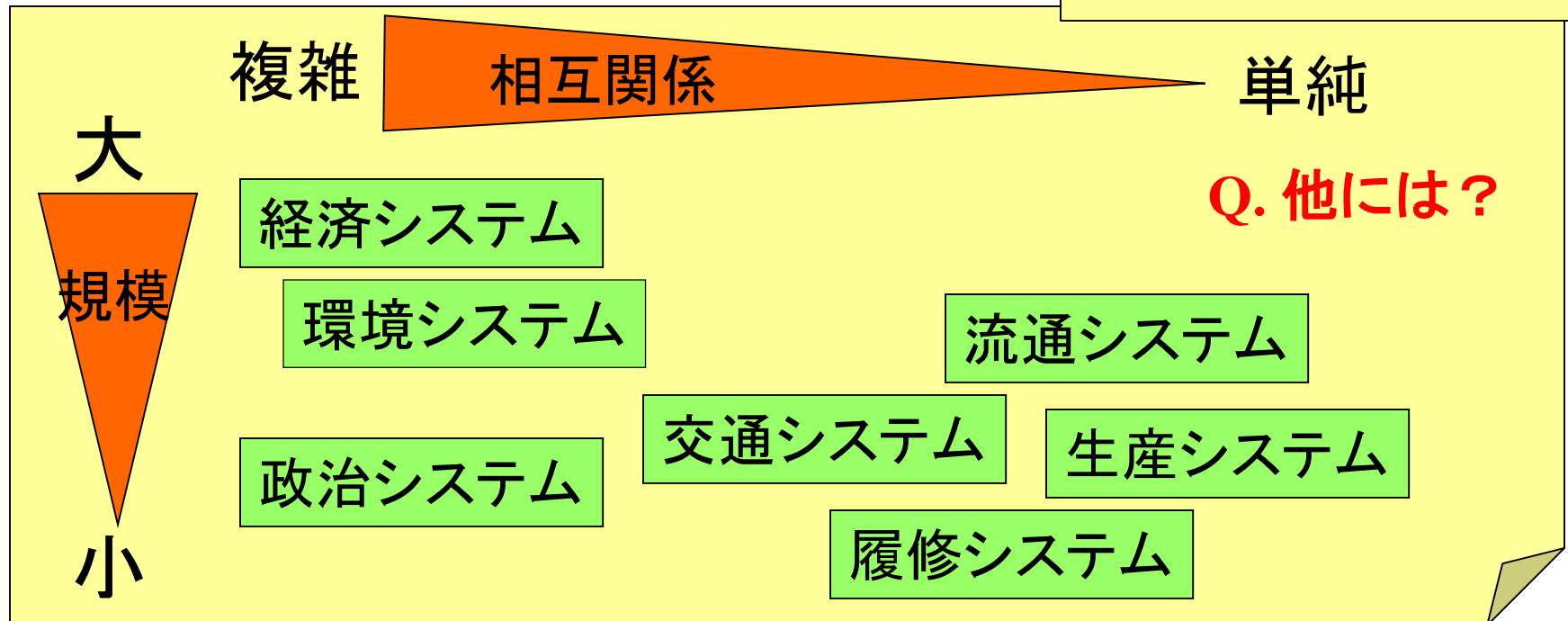


# システムとして問題把握

相互関係を持つ構成要素の集まり  
+  
構成要素は共通目的で機能

ポイント

- 構成要素の抽出
- 相互関係の明確化



※システムは小さなシステムの集まりの場合もある



# 問題を捉えるポイント

システムにおいて

- コントロールする(できる)構成要素は？
- コントロールに対する制約は？
- コントロール結果に対する  
良い・悪いの尺度は？



# 演習1 混雑する学食

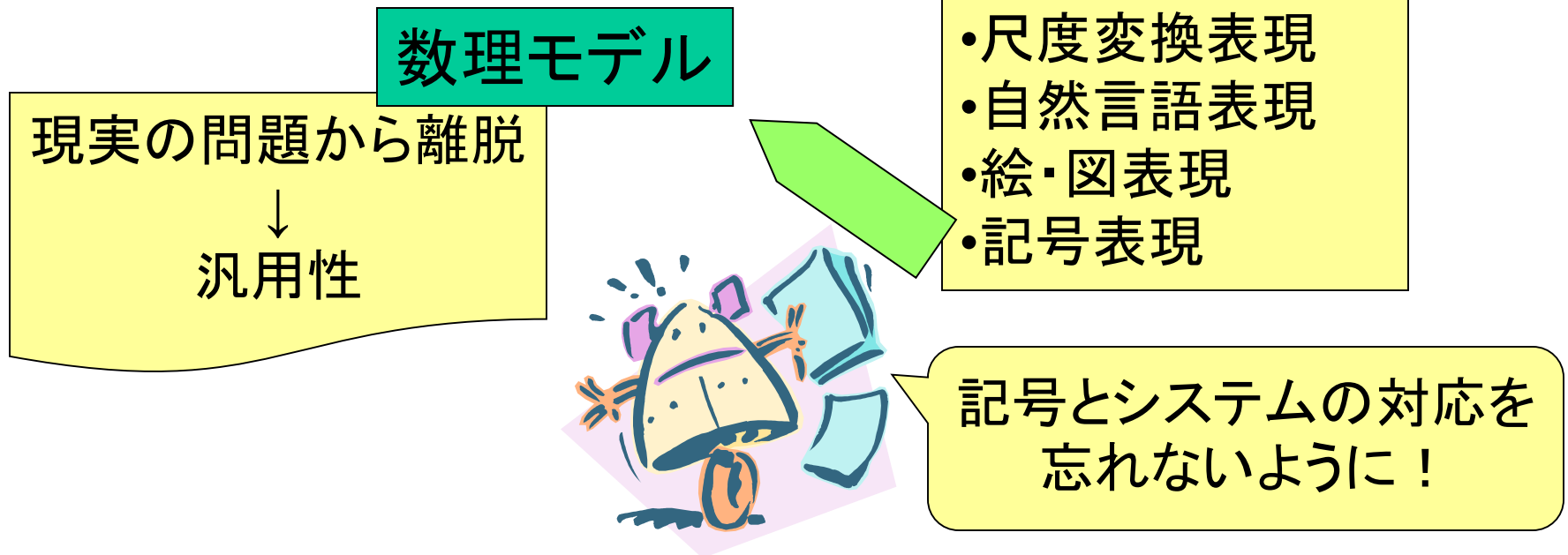
1. 何が問題？
2. システムの構成要素は？相互関係は？
3. コントロールする(できる)構成要素は？
4. コントロールに対する制約は？
5. コントロール結果に対する  
良い・悪いの尺度は？



# ステップ2 モデル化(定式化)



- 関係部分のみ抽出
- 抽出したシステムを抽象的な記号で表現



# 数理モデルは便利！

問題A



モデル化

問題B



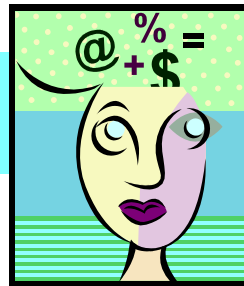
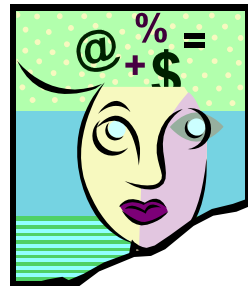
モデル化

問題Z

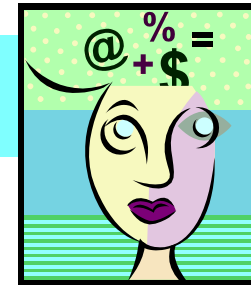


モデル化

様々な問題



同じ構造



数理モデル

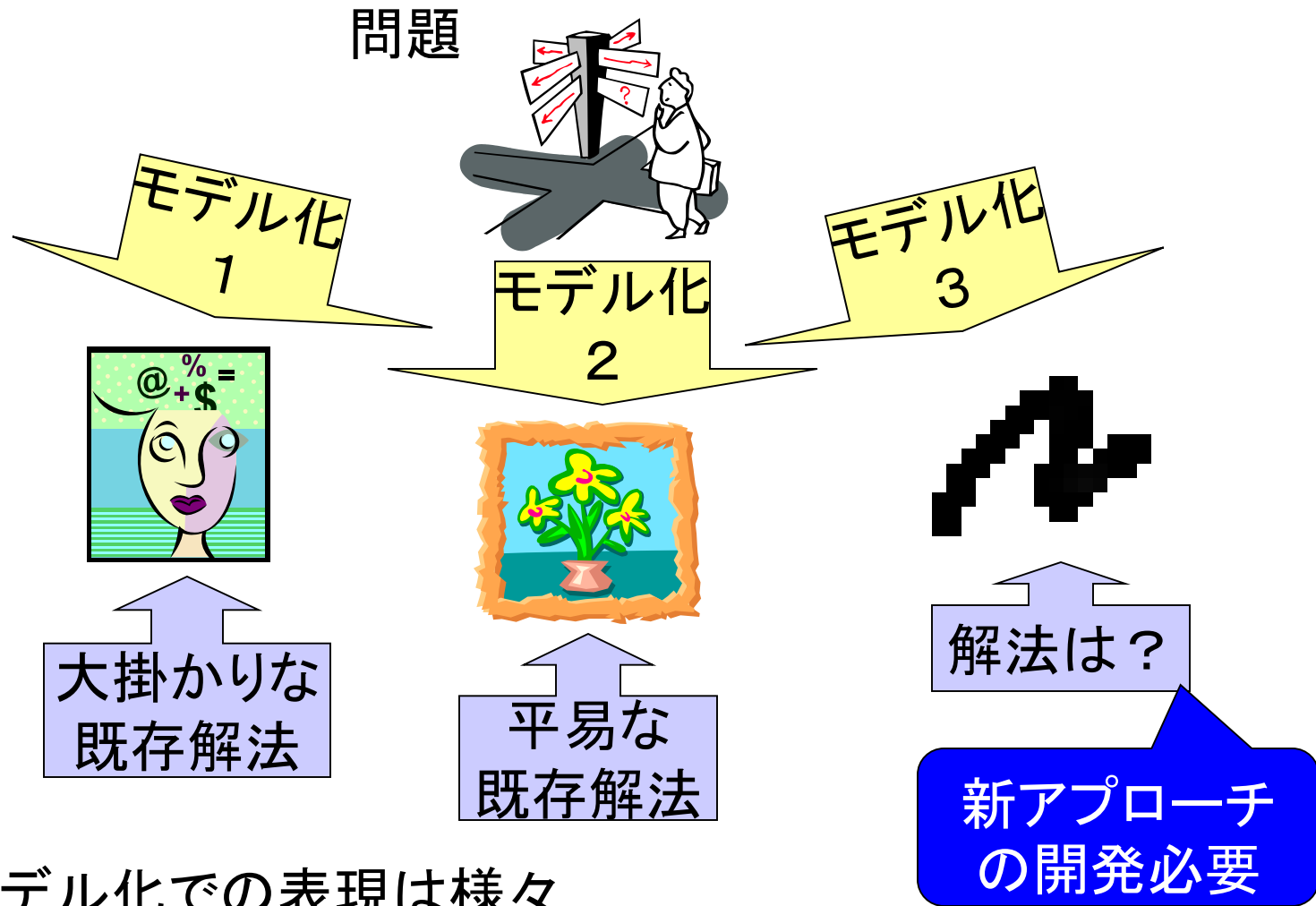
システムの  
アプローチ

共通アプローチで解決可

個々に解く必要はないんだね



# モデル化は芸術



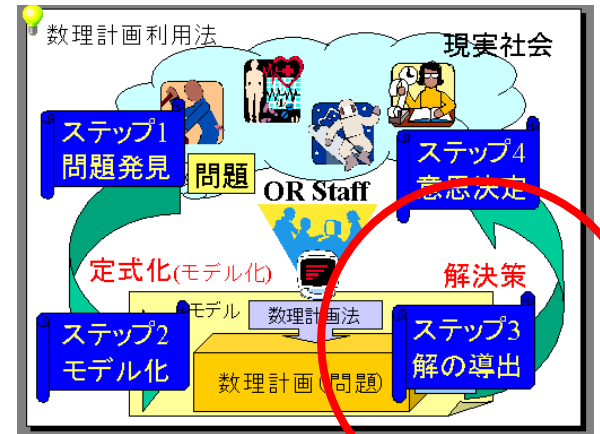
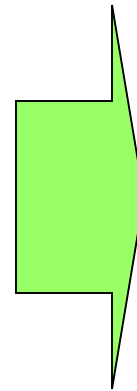
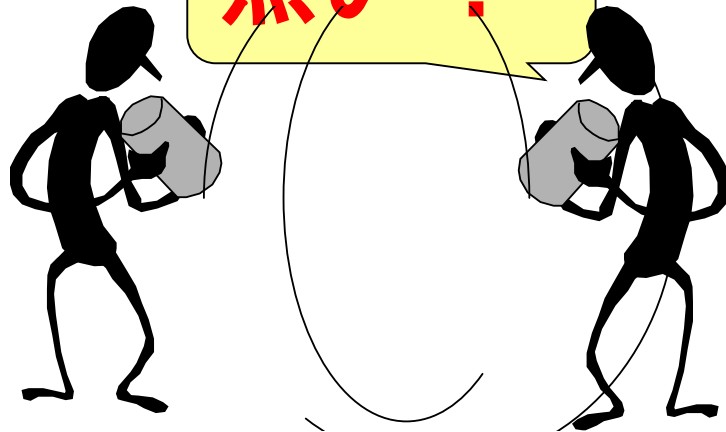
モデル化での表現は様々

⇒表現に応じて解き易い、解きにくい

# ステップ3 最適解の導出

数理モデル化された  
どんな問題でも解く  
万能な方法を教えて

無い！



- 問題タイプ別の解法
- やさしい手法
  - 難しい手法
  - 手間のかかる手法
  - 効率良い手法 等

数理計画法

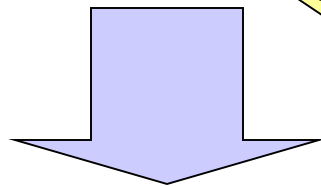
# ステップ4 意思決定

数理モデルの最適解

≠

問題解決の最良案

(∵ 元の問題  $\supseteq$  数理モデル)



ギャップがある  
場合が多い



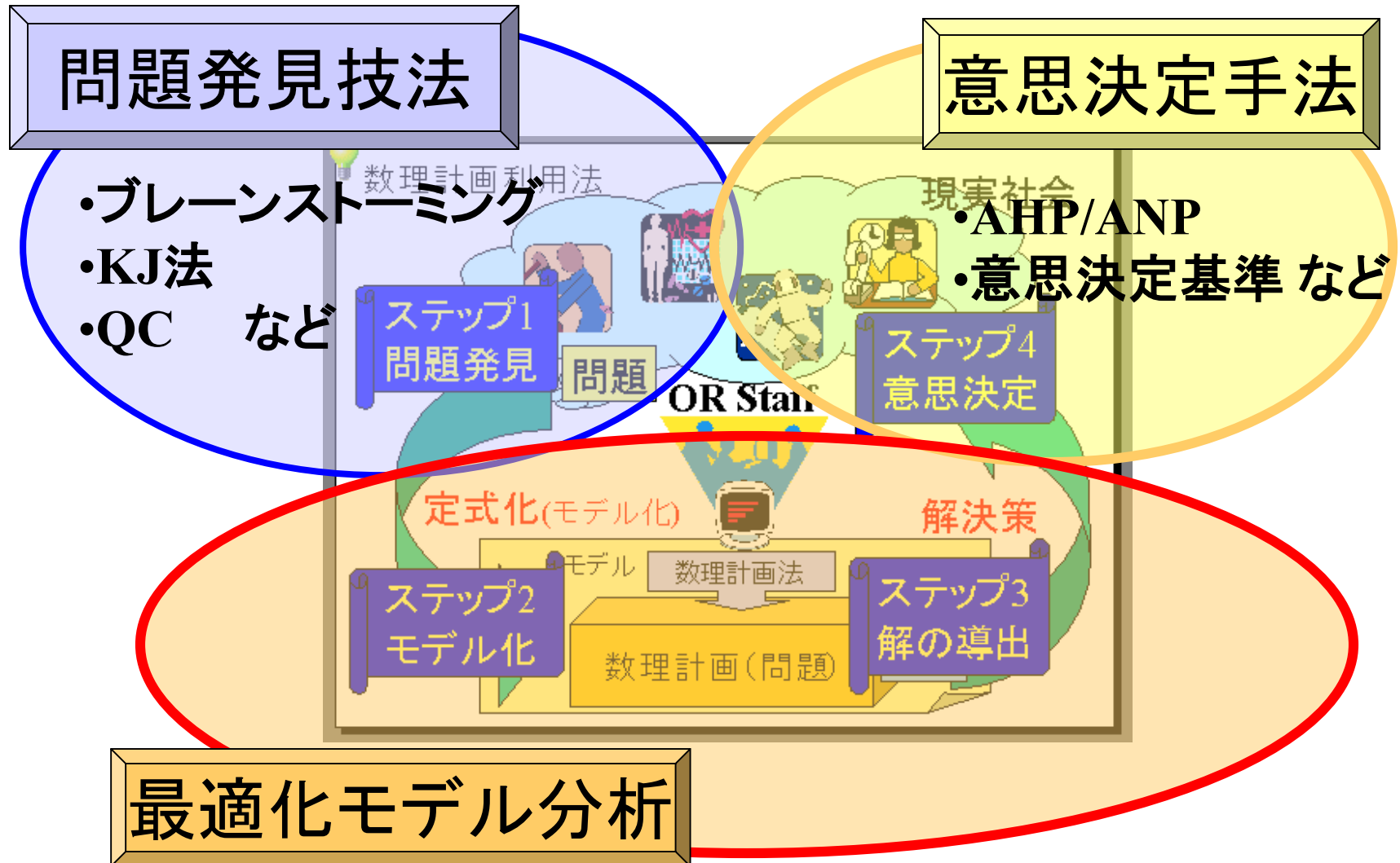
実際の解決策提案には  
意思決定が必要

意思決定法

- AHP/ANP
- 意思決定基準 など



# 講義「最適化モデル分析」での守備範囲

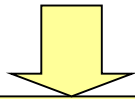




# この先で学ぶこと

- 数理計画とは
- システム的アプローチによる問題解決

学習済



- 数理モデル化と表現方法(定式化)
- 数理計画問題の分類

