組み合わせ最適化

組み合わせ最適化って？

- 決められた条件の中で，最も良い答えを求める

問題例

- 最短経路問題
- ナップサック問題
- 巡回セールスマン問題　　などなど

このような問題へのアプローチ

- DPのような効率よく解くアルゴリズムを用いる
- 全部の組み合わせをチェックする←今回取り扱うのはこっち

整数線形計画問題

- 解きたい問題を目的関数と制約式を用いてモデル化する.
  - 整数を使うことで現実問題をモデル化できる（こともある）
- 分枝限定法を使って探索する範囲を限定しながら，全てのパターン探索する.

意外と整数線形計画問題を使えるシーンはおおい

- LT会の当番表とかもいろいろ制約付けたら作れるかも
- アルゴリズム考えるの面倒なときとかもいいかも
- スライド作るの面倒なのであとは，デモします

自作PC最適化ツール（コストとおすすめ度合と消費電力で）

- 使った目的関数と制約式

```
param N_GPU integer, > 0;
param N_CPU integer, > 0;
param N_POW integer, > 0;
param N_MOT integer, > 0;

set V_GPU := 1..N_GPU ;
set V_CPU := 1..N_CPU ;
set V_POW := 1..N_POW ;
set V_MOT := 1..N_MOT ;

param COST_CPU{V_CPU} ;
param COST_GPU{V_GPU} ;
param COST_POW{V_POW} ;
param COST_MOT{V_MOT} ;
```

```
param POW_CPU{V_CPU} ;
param POW_GPU{V_GPU} ;
param POW_MOT{V_MOT} ;
param POW{V_POW} ;

param RCD_CPU{V_CPU} ;
param RCD_GPU{V_GPU} ;
param RCD_POW{V_POW} ;
param RCD_MOT{V_MOT} ;

var cpu{V_CPU} binary ;
var gpu{V_GPU} binary ;
var pow{V_POW} binary ;
var mot{V_MOT} binary ;

var satisfy ;
var cost ;
```

```
maximize SATISFY : satisfy ;

s.t. CPU_Select:
    sum {i in V_CPU} cpu[i] = 1 ;

s.t. GPU_Select:
    sum {j in V_GPU} gpu[j] = 1 ;

s.t. POW_Select:
    sum {k in V_POW} pow[k] = 1 ;

s.t. MOT_Select:
    sum {l in V_MOT} mot[l] = 1 ;

s.t. Power_Limit:
    sum {i in V_CPU} POW_CPU[i] * cpu[i]
    + sum {j in V_GPU} POW_GPU[j] * gpu[j]
    + sum {l in V_MOT} POW_MOT[l] * mot[l] <= sum {k in V_POW} POW[k] * pow[k] ;
```

```
s.t. Total_Cost:
    cost = sum {i in V_CPU} COST_CPU[i] * cpu[i]
         + sum {j in V_GPU} COST_GPU[j] * gpu[j]
         + sum {k in V_POW} COST_POW[k] * pow[k]
         + sum {l in V_MOT} COST_MOT[l] * mot[l] ;

s.t. Limit_Cost:
    cost <= 20 ;

s.t. Satisfaction:
    satisfy = sum {i in V_CPU} RCD_CPU[i] * cpu[i]
            + sum {j in V_GPU} RCD_GPU[j] * gpu[j]
            + sum {k in V_POW} RCD_POW[k] * pow[k]
            + sum {l in V_MOT} RCD_MOT[l] * mot[l] ;
```

結果

```
Problem:    modell
Rows:       9
Columns:    11 (9 integer, 9 binary)
Non-zeros:  40
Status:     INTEGER OPTIMAL
Objective:  SATISFY = 6 (MAXimum)

   No. Column name       Activity     Lower bound   Upper bound
------ ------------    ------------- ------------- -------------
     1 cpu[1]       *              0             0             1
     2 cpu[2]       *              0             0             1
     3 cpu[3]       *              1             0             1
     4 gpu[1]       *              1             0             1
     5 gpu[2]       *              0             0             1
     6 pow[1]       *              0             0             1
     7 pow[2]       *              1             0             1
     8 mot[1]       *              0             0             1
     9 mot[2]       *              1             0             1
    10 satisfy                     6
    11 cost                     16.2
```