HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.

ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

PLANNING OPTIMIZATION

Metaheuristic methods

ONE LOVE. ONE FUTURE.

CONTENT

- Introduction to Metaheuristics
- Tabu Search (TS)
- Simulated Annealing (SA)
- Genetic algorithms (GA)
- Ant Colony Optimization (ACO)

Overview

- Metaheuristics
  - Generic framework, problem-independent
  - Aim to find high-quality solutions to optimization problems in reasonable time
  - Do not specify (like heuristic methods) how close are the obtained solution from the optimal solutions
  - Single-solution based: Tabu Search, Simulated Annealing, Variable Neighborhood Search, etc.
  - Population-based: Genetic algorithms, Ant Colony Optimization, etc.

Tabu Search

- Based on neighborhood search
- Use memory structure to avoid revisiting solutions explored so far

```
function TabuSelect(N, f, s, T){
    eval = ∞; s = NULL;
    for x in N(s) do
        if x ∉ T and f(x) < eval then {
            eval = f(x); s = x;
        }
    return s;
}
```

```
function TabuSearch(N, f){
  // N: neighborhood
  // f: quality function
  s = Generate an initial solution;
  s* = s;
  T = initialize a tabu list
  while termination not reach do {
    s = TabuSelect(N(s), f, s, T); //neighbor selection
    if s = NULL then break;
    if f(s) > f(s*) then   s* = s;
    AddLast(s, T); // add s to the end of T
  }
  return s*;
}
```

Simulated Annealing

- Probabilistic technique based on neighborhood search
- Accept poor solutions with a probability which reduces over time

```
function S-Metropolis(N, f, s, t){
    select n∈ N with probability 1/|N|;
    if f(n) > f(s) then return n;
    else with probability e^((f(s)−f(n))/t) ;
    else return s;
}
```

```
function SimulatedAnnealing(N, f){
  // N: neighborhood
  // f: quality function
  s = Generate an initial solution;
  s* = s;
  t = initial templature;
  while termination not reach do {
    s = S-Metropolis(N(s), f, s, t); //neighbor selection
    if s = NULL then break;
    if f(s) > f(s*) then   s* = s;
    t = update(t); // reduce temperature
  }
  return s*;
}
```

Genetic algorithms

- Population-based method
- Generate initial population of solutions
- Perform cross over between 2 parent solutions from the population for generating new child solutions
- Perform mutation on child solutions with some probability
- Replace some poor solutions in the population by new good child solutions generated

Ant Colony Optimization (ACO)

- Nature-inspired metaheuristic
- Each iteration
  - *m* ants construct *m* solutions exploiting a pheromone model
    - Solution can be seen a sequence of components: component (*i*, *j*) means (for instance) the assignment of the decision variable x_i by the value *j*
    - Select component (*i*, *j*) with the probability

$$p(i,j) = \frac{\tau(i,j)^{\alpha} * \eta(i,j)^{\beta}}{\sum_{(p,q) \in Candidate} \tau(p,q)^{\alpha} * \eta(p,q)^{\beta}}$$

$\tau(i,j)$: pheromone on component (*i*, *j*)

$\eta(i,j)$: heuristic function for selecting component (*i*, *j*)

  - Some obtained solutions will be used to update pheromone
    - Example: $\tau(i,j) = \rho^{*}\tau(i,j) + \sum_{k=1}^{m} \Delta(i,j)^{k}$

in which $\Delta(i,j)^{k} = \begin{cases} \frac{1}{f(s^{best})}, & \text{if } (i,j) \text{ is selected in the solution constructed by ant } k \\ 0, & \text{otherwise} \end{cases}$

HUST

hust.edu.vn fb.com/dhbkhn

THANK YOU !