A Course on Meta-Heuristic Search Methods for Combinatorial Optimization Problems

Santosh Kumar Mandal, *Ph.D research fellow*

AutOrI LAB,
DIA, Roma Tre
Email: mandal@dia.uniroma3.it

January 20, 2014

Outline

- Multi-parent crossover: `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.56.5147&rep=rep1&type=pdf`
- Cellular GA: `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.65.7785&rep=rep1&type=pdf`
- EA for VRP: `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.3.7230&rep=rep1&type=pdf`

ROMA TRE UNIVERSITÀ DEGLI STUDI

Developed by Glover (1977)

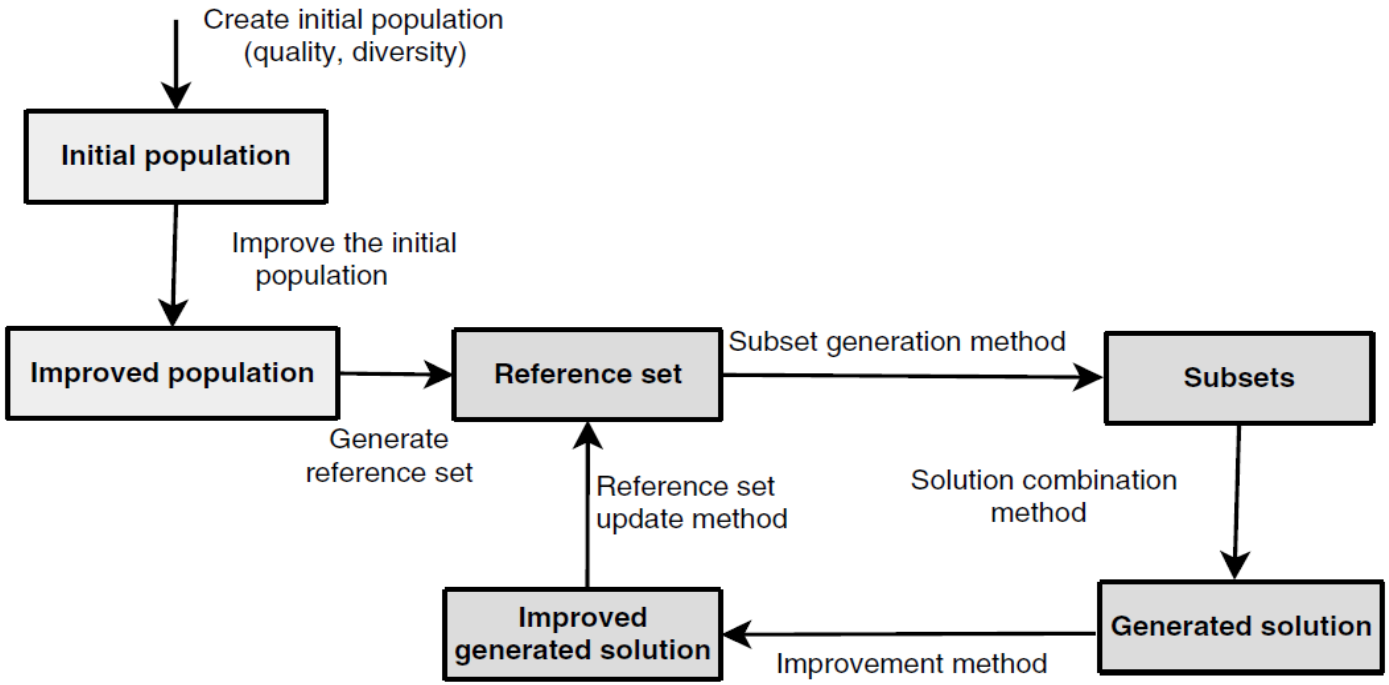


Figure: Flow chart of scatter search

components

Initial population:

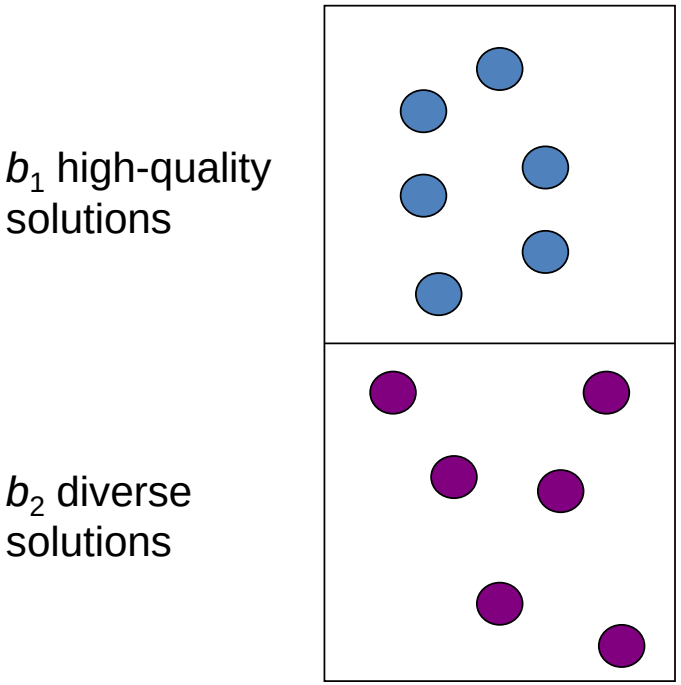
- Generate a set of diverse and high quality solutions.
- Use a greedy procedure

Improve initial population:

- Use a local search procedure.
- A Single-objective meta-heuristic can also be applied.

components

Reference set:



RefSet of size *b*

Figure: Reference set

components

Subset generation method:

- Each subset should contain t solutions (t: size of subsets $= 2$).
- Each subset should include atleast one new solution.

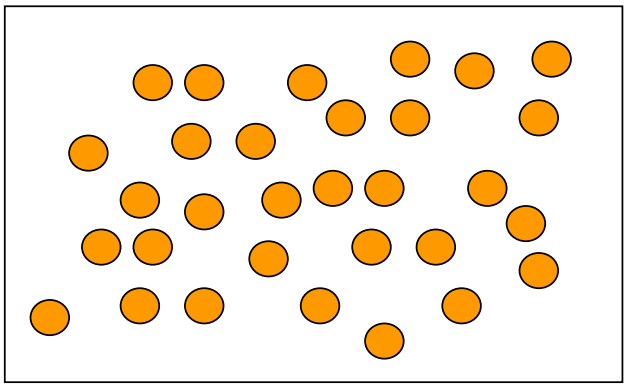
Recomibination method:

- Recombine solutions in each subset.
- Any crossover operator can be used.

components

Reference set update: Combine sets and select the top best individuals.

RefSet of size *b*



Pool of new trial solutions

Table of Contents

Overview

- It protects organisms from *pathogens* with several layers of defence.
- Surface barrier:
  - Skin, Hair
  - Body fluids: tears, sweat, etc.
- Innate immune system (non-specific response):

First arm	A group of 20 proteins called the *complement system*. These proteins flow freely in the body. They kill invaders or recruit eater cells such as macrophages (a type of Phagocyte cell).
Second arm	Phagocyte cells: They patrol the body searching for pathogens. If found, they eat and destroy them.
Third arm	NK-cells: They kill virus-infected cells.

ROMA TRE UNIVERSITÀ DEGLI STUDI

continue...

- Adaptive immune system (specific response):
  - Lymphocytes: T-cells and B-cells
  - T-cells:
    - Helper T-cells: Help B-cells and activate killer T-cells
    - Killer T-cells: kills infected cells.
  - B-cells: Once a B-cell finds an *antigen* matching its receptor, it becomes active. With the help of proteins of helper T-cells, it becomes fully active and proliferates into *plasma cells* and *memory cells*.

continue...

- B-cell:
  - Plasma cells: They produce specific *antibodies* that lock onto the *antigen*. *Eater cells* then chew up bacteria marked with antibodies.
  - Memory cells: They memorizes specific intruders and helps the immune system to respond faster if the same intruder enters again.

Basic principles

- Fit B-cells are allowed to grow in number.
- Unfit B-cells are slowly removed.
- Cloning is directly proportional to the fitness.
- Mutation is inversely proportional to the fitness.

Working process

[redacted] Template of the CLONALG.

Input: Initial population P_0.
$P = P_0$; /* Generation of the initial population of random antibodies */
Repeat
  Evaluate all existing antibodies and compute their affinities ;
  Select $N\%$ of antibodies with highest affinities ;
  Clone the selected antibodies ;
  Maturate the cloned antibodies ;
  Evaluate all cloned antibodies ;
  Add $R\%$ of best cloned antibodies to the pool of antibodies ;
  Remove worst members of the antibodies pool ;
  Add new random antibodies into the population ;
Until Stopping criteria satisfied
Output: Best population found.

$$N_c = \lfloor \frac{\beta \times |P_o|}{i} + 0.5 \rfloor \tag{3.1}$$

$\beta \in [0, 1]$
$|P_o|$: size of the population.
i: rank of an antibody in the sorted population according to ascending values of affinities.

Assignment-II

- Form a group of 2 new students.
- Implement *Genetic Algorithm* algorithm on the single-variable Ackley function using C++/MATLAB/Java.
- Prepare a report of 3-5 pages (extend if needed):
  - Compare its performance against the *Simulated Annealing* algorithm.
  - Support your claims by:
    - Providing tables showing results on different values of algorithmic parameters (population size and crossover/mutation probability).
    - Providing graphs showing convergence of algorithm (w.r.t. number of iterations/computational time).
- Use population of binary strings.
- Use one/two point crossover.
- Use flip operator mutation.
- 10 experiments with different values of GA parameters.
- 20 runs for each parameter settings.
- Show results of the best run only.

ROMA TRE UNIVERSITÀ DEGLI STUDI

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences*, 8(1):156–166.