Rapport knapsack

December 2022

1 Ant colony algorithm

The ACO(Ant colony algorithm) is a optimization algorithm by using probabilistic, this algorithm inspired from the behavior of ants. The idea is to simulate the behavior of the ant for finding the best solution of a difficult optimism problem.

For implementing this algorithm we use this pseudo-code as a base :

begin

while (a cycle exists) **do**

while (an ant k, which has not yet worked, exists) **do**

while ($V_c \geq 0$) **do**

select a next object o_j from N_i with probability $p_j = \begin{cases} \dfrac{\tau_j^\alpha \mu_j^\beta}{\sum_{j \in N_i} \tau_j^\alpha \mu_j^\beta}, & \text{for } j \in N_i \\ 0, & \text{for } j \notin N_i \end{cases}$

add a selected object to a partial solution $S = S + \{o_j\}$

update the current knapsack load capacity $V_C = V_C - w_j$

update the profit $Z = Z + z_j$

update the neighbourhood of the current state $N_i = \{o_i : w_i \leq V_C\}$

end

remember the best solution if a better solution has been found

end

remember a global best solution if a better solution has been found

use an evaporation mechanism $\tau = \rho\tau$

update pheromone trails $\tau = \tau + \Delta\tau$

end

end.

where:

C	–	is the total knapsack load capacity,
V_c	–	is the current knapsack load capacity, $V_C = C - \sum_{g \in S_i} (w_g)$,
S_i	–	is a partial solution,
$\sum_{g \in S_i} (w_g)$	–	is the weight of all objects which were included in the partial solution S_i,
w_j	–	is the weight of selected object j,
z_j	–	is the profit of selected object j,
μ_j	–	is the attractiveness of selecting an object j.

To implement this program we have used lists to represent all the elements such as supplies or arrays to represent the deposits of pheromones left by the ants when they pass over an object.
In order to simulate the cycles and the ants we use 2 loops while.

The choice to take an object by an ant and give by the probabilistic:

$$p_j = \begin{cases} \dfrac{\tau_j^\alpha \mu_j^\beta}{\sum_{j \in N_i} \tau_j^\alpha \mu_j^\beta}, & \text{for } j \in N_i \\ 0, & \text{for } j \notin N_i \end{cases}$$

When an ant finds a better solution it can leave a drop of pheromone with the function :

$$\Delta\tau = f(Q) = \frac{1}{1 + \frac{z_{best} - z}{z_{best}}}$$

To calculate the attractiveness of an object we have set up 3 formulas :

1) AKA1, – with the attractiveness μ_j of the move expressed as:

$$\mu_j = \frac{\frac{z_j}{w_j}}{V_C}$$

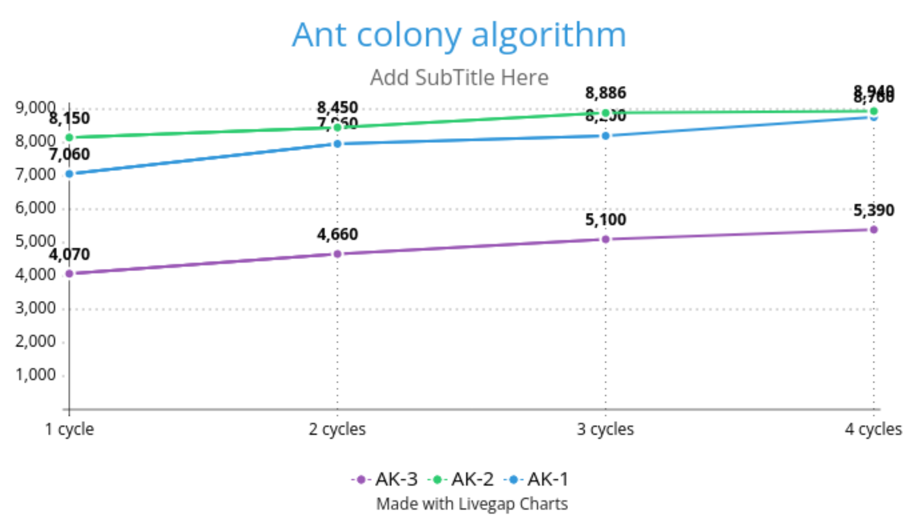
2) AKA2, – with the attractiveness μ_j of the move expressed as:

$$\mu_j = \frac{z_j}{w_j^2}$$

3) AKA3, – with the attractiveness μ_j of the move expressed as:

$$\mu_j = \frac{\frac{z_j}{w_j}}{C}$$

Test with different calculation formula of attractiveness



Test performed with the dataset : knapPI$_{11}00_{1}000_{1}$