HOMEWORK 5 FOR MODERN OPTIMIZATION METHODS

R09946006 | 何青儒 | HO, Ching-Ru | Oct. 19, 2020

1. Programming of Ant Colony Optimization.

- (a) State all inputs, data and output of ACO
  - Input:
    - N: Positive integer. The number of particles being used in the optimization.
    - n: Positive integer. The number of layers from home to destination (or how many nodes does particle should pass in one turn).
    - α: Float, between 0 and 1, Node probability calculation parameter.
    - p: Float, between 0 and 1. Pheromone evaporation parameter.
    - η: Positive number, can be integer of float. The scaling parameter in pheromone update process.
    - d: Positive number, can be integer or float. Consecutive permissible discrete value between range x_{min} and x_{max}.
    - x_{min} and x_{max}: Noticed that $x_{max} > x_{min}$. The range of particles' value.
    - T: Positive integer. Maximum iteration times we've given.
  - Data:
    - t: Iteration number. The program will be halt when convergence or reach the T times.
    - N_i^k: List. Neighborhood of ant k when at node i.
    - $X_i = \{x_{i1}, \cdots, x_{in}\}$: List. Each particle's path of n's ant.
    - $\tau_{ij} = \{\tau_{ij1}, \tau_{ij2}, \cdots, \tau_{ijn}\}$: List. Pheromone density of n's ant.
    - x_{best}: The best case of particle during one iteration.
    - x_{worse}: The worse case of particle during one iteration.
  - Output:
    - $x_{optimal}$: The most optimal case of particle we find during the iterations.
- (b) Algorithm pseudo-code of ACO

```
INPUT: ALL the PARAMETER MENTIONED ABOVE

// INITIALIZATION
Set: put N ants in n node.  // Particle position
Set: for all the edge set tau_{ij} <- 1     // Particle pheromone desity

FUNCTION SelectNode():
    FOR each ant DO:
        N_i^k <- Neighborhood nodes
    END FOR
    FOR each node DO:   // Calculate the probability
        IF node IN N_i^k THEN:
            P <- tau^alpha / SUM(tau^alpha)
        ELSE DOL
            P <- 0
        END IF
```

```
    END FOR
    RETURN P
END FUNCTION

FUNCTION ObectiveFunction():
    ans =   // Set Path
    RETURN ans
END FUNCTION

FUNCTION UpdatePhoremone():
    tau <- tau + DELTA(tau)
    RETURN tau
END FUNCTION

FUNCTION EvaporatePheromone():
    tau <- (1-p) * tau
    RETURN tau
END FUNCTION

WHILE t from 1 to T OR Convergence DO:
    FOR each ant DO:
        SelectNode()
        IF more than one edges has same amount of pheromone THEN:
            choose one randomly
        END IF
        Record this ant took this edge
        IF end of edge is terminal node THEN:
            Remove the loops from the path between source and terminal node just traced.
            UpdatePhoremone()
            EvaporatePheromone()
            Reset the ant back to the initial state.
            Clear its edge history and let it move as before.
        END IF
    END FOR
    t <- t + 1  //Update iteration number
END WHILE

OUTPUT: Best Record of Path
```

- (c) See `hw5ass1.py` (format: Python 3). Noticed that this program isn't completely because we don't know the objective function and other parameters.

2. Optimization of a Function via Ant Colony Optimization.

- (a) Show detailed calculations for 2 iterations.
  - INITIALIZATION (Iteration number $i = 0$):
    - Particle size $N = 4$. There is only one design variable, so $n = 1$.
    - Set $(x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}) = (0.0,\ 0.5,\ 1.0,\ 1.5,\ 2.0,\ 2.5,\ 3.0)$ in range $x \in \{0, 3\}$ with discrete value 0.5.
    - Assume equal pheromone along each of the pairs. $(\tau_{11},\ \tau_{12},\ \tau_{13},\ \tau_{14},\ \tau_{15},\ \tau_{16},\ \tau_{17}) = (1, 1, 1, 1, 1, 1, 1)$.
    - Set iteration number as $i = 1$.
  - Iteration number $i = 1$:

        - Since all $\tau_{1j} = 1$, $\forall j = 1, 2, \cdots, 7$, and $\alpha = 1$, the probability $p_{1j} = \frac{\tau_{1j}^{\alpha}}{\sum \tau_{1j}^{\alpha}} = \frac{\tau_{1j}}{\sum \tau_{1j}} = \frac{1}{7}$.
        - Apply the roulette-wheel selection method: $x_{11} : (0, 0.1428)$, $x_{12} : (0.1428, 0.2857)$, $x_{13} : (0.2857, 0.4286)$, $x_{14} : (0.4286, 0.5714)$, $x_{15} : (0.5714, 0.7143)$, $x_{16} : (0.7143, 0.8571)$, $x_{17} : (0.8571, 1)$.
        - Generate r_1, r_2, r_3, r_4 randomly to find the ant location and objective function value:
            - $r_1 = 0.3 \in \mathbb{P}(x_{13})$, $Ant_1 = x_{13} = 1$. $f(x_{13}) = f(1) = -19$
            - $r_2 = 0.5 \in \mathbb{P}(x_{14})$, $Ant_2 = x_{14} = 1.5$, $f(x_{14}) = f(1.5) = -5.59375$, worse path.
            - $r_3 = 0.2 \in \mathbb{P}(x_{12})$, $Ant_3 = x_{12} = 0.5$, $f(x_{12}) = f(0.5) = -34.28125$
            - $r_4 = 0.6 \in \mathbb{P}(x_{15})$, $Ant_4 = x_{15} = 2$, $f(x_{15}) = f(2) = -43$, best path.
        - Find the best and worse path are: $x_{best} = x_{15} = 2$, $x_{worse} = x_{12} = 0.5$.
        - Set scaling parameter $\eta = 2$, the change of pheromone $\sum \Delta\tau^{(k)} = \Delta\tau^{(k=4)} = \frac{\eta f_{best}}{f_{worse}} = \frac{2 \cdot (-43)}{-5.59375} = 15.3743$.
        - Set pheromone evaporation parameter $p = 0.5$, the pheromone in the nest step become $(\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}, \tau_{15}, \tau_{16}, \tau_{17}) = (0.5, 0.5, 0.5, 0.5, 15.8743, 0.5, 0.5)$.
        - Set iteration number $i = 2$.
    - Iteration number $i = 2$:
        - From $i = 1$, $(\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}, \tau_{15}, \tau_{16}, \tau_{17}) = (0.5, 0.5, 0.5, 0.5, 15.8743, 0.5, 0.5)$.
        - Calculate the probability. $\mathbb{P}_{1j} = \frac{0.5}{18.8743} = 0.0265$, $\forall j = 1, 2, 3, 4, 6, 7$. And $\mathbb{P}_{15} = \frac{15.8743}{18.8743} = 0.84105$.
        - Apply the roulette-wheel selection method: $x_{11} : (0, 0.0265)$, $x_{12} : (0.0265, 0.053)$, $x_{13} : (0.053, 0.0795)$, $x_{14} : (0.0795, 0.106)$, $x_{15} : (0.106, 0.94705)$, $x_{16} : (0.94705, 0.97355)$, $x_{17} : (0.97355, 1)$.
            - Generate r_1, r_2, r_3, r_4 randomly to find the ant location and objective function value:
                - $r_1 = 0.1 \in \mathbb{P}(x_{14})$, $Ant_1 = x_{14} = 1.5$. $f(x_{14}) = f(1.5) = -34.28125$
                    - $r_2 = 0.3 \in \mathbb{P}(x_{15})$, $Ant_2 = x_{15} = 2$, $f(x_{15}) = f(2) = -43$, best path.
                    - $r_3 = 0.75 \in \mathbb{P}(x_{15})$, $Ant_3 = x_{15} = 2$, $f(x_{15}) = f(2) = -43$, best path.
                    - $r_4 = 0.95 \in \mathbb{P}(x_{16})$, $Ant_4 = x_{16} = 2.5$, $f(x_{16}) = f(2.5) = -25.46875$, worse path
        - Find the best and worse path are: $x_{best} = x_{15} = 2$, $x_{worse} = x_{16} = 2.5$.
        - Set scaling parameter $\eta = 2$, the change of pheromone $\sum \Delta\tau^{(k)} = \Delta\tau^{(k=2)} + \Delta\tau^{(k=3)} = 2 \cdot \frac{\eta f_{best}}{f_{worse}} = 2 \cdot \frac{2 \cdot (-43)}{-25.46875} = 6.75337$
        - Set pheromone evaporation parameter $p = 0.5$, the pheromone in the nest step become $(\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}, \tau_{15}, \tau_{16}, \tau_{17}) = (0.25, 0.25, 0.25, 0.25, 14.49052, 0.25, 0.25)$.
        - Set iteration number $i = 3$.
- (b) Show detailed calculations for 2 iterations again by neglecting the pheromone evaporation
    - Assume that iteration number $i = 0$ (initialization) and $i = 1$ have the same situation as above until pheromone update process.
    - In the last stage of iteration $i = 1$, set pheromone evaporation parameter $p = 0$, the pheromone in the nest step become $(\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}, \tau_{15}, \tau_{16}, \tau_{17}) = (1, 1, 1, 1, 16.3743, 1, 1)$. Turn to $i = 2$.
    - Iteration number $i = 2$:
        - From $i = 1$, $(\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}, \tau_{15}, \tau_{16}, \tau_{17}) = (1, 1, 1, 1, 16.3743, 1, 1)$.

- Calculate the probability. $\mathbb{P}_{1j} = \frac{1}{22.3743} = 0.0447, \forall j = 1, 2, 3, 4, 6, 7$. And $\mathbb{P}_{15} = \frac{16.3743}{22.3743} = 0.7318$.
- Apply the roulette-wheel selection method: $x_{11} : (0, 0.0447)$, $x_{12} : (0.0447, 0.0894)$, $x_{13} : (0.0894, 0.1341)$, $x_{14} : (0.1341, 0.1788)$, $x_{15} : (0.1788, 0.9106)$, $x_{16} : (0.9106, 0.9553)$, $x_{17} : (0.9553, 1)$.
    - Generate the same r_1, r_2, r_3, r_4 as above to find the ant location and objective function value:
        - $r_1 = 0.1 \in \mathbb{P}(x_{13})$, $Ant_1 = x_{13} = 1.0$. $f(x_{13}) = f(1) = -19$
        - $r_2 = 0.3 \in \mathbb{P}(x_{15})$, $Ant_2 = x_{15} = 2$, $f(x_{15}) = f(2) = -43$, best path.
        - $r_3 = 0.75 \in \mathbb{P}(x_{15})$, $Ant_3 = x_{15} = 2$, $f(x_{15}) = f(2) = -43$, best path.
        - $r_4 = 0.95 \in \mathbb{P}(x_{16})$, $Ant_4 = x_{16} = 2.5$, $f(x_{16}) = f(2.5) = -25.46875$, worse path
- Find the best and worse path are: $x_{best} = x_{15} = 2$, $x_{worse} = x_{13} = 1$.
- Set scaling parameter $\eta = 2$, the change of pheromone $\sum \Delta\tau^{(k)} = \Delta\tau^{(k=2)} + \Delta\tau^{(k=3)} = 2 \cdot \frac{\eta f_{best}}{f_{worse}} = 2 \cdot \frac{2 \cdot (-43)}{-19} = 9.0526$
- Set pheromone evaporation parameter $p = 1$, the pheromone in the nest step become $(\tau_{11}, \tau_{12}, \tau_{13}, \tau_{14}, \tau_{15}, \tau_{16}, \tau_{17}) = (1, 1, 1, 1, 25.4269, 1, 1)$.
- Set iteration number $i = 3$.

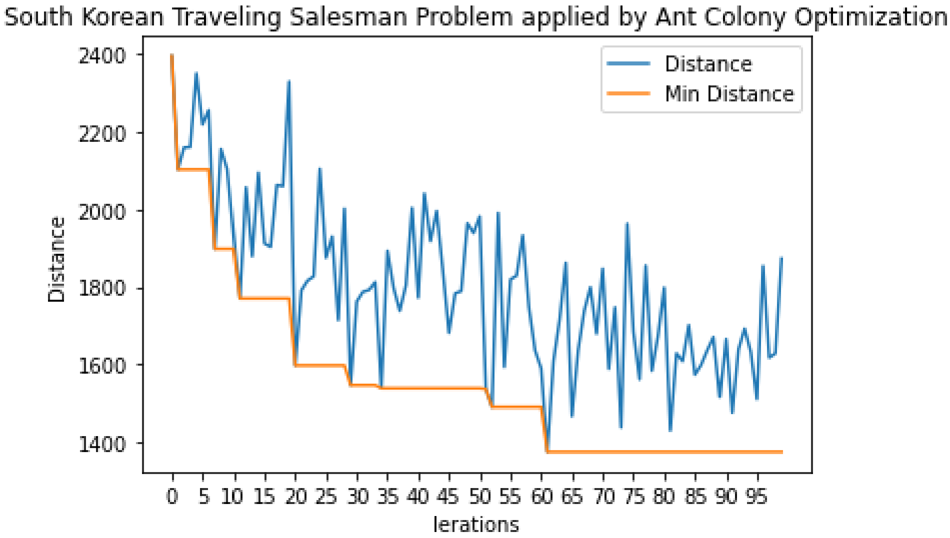
3. A Revisit of Traveling Salesman Problem

- (a) See `hw5ass3.ipynb` (format: Jupyter Notebook with Python).
- (b) I set `p=0.5` and $\eta = 1$. Because we have only 100 times of iteration, if I set a bigger evaporation parameter, it might converge slowly. I hope it will converge more quickly thus I let each nodes evaporates a half of pheromone in each iteration times. And the reason I set $\eta = 1$ is taking the lecture note as the reference.
- (c) Solve TSP by ACO:
    - Start from Incheon and back to Incheon finally (a loop).
    - City order: $[0, 1, 6, 13, 9, 4, 8, 5, 14, 11, 10, 2, 7, 3, 12, 0]$.
    - Path Distance: 1375 km.

```
>> shorted_path: ([(0, 1), (1, 6), (6, 13), (13, 9), (9, 4), (4, 8), (8, 5), (5, 14), (14, 11), (11, 10), (10, 2), (2, 7), (7, 3), (3, 12), (12, 0)], 1375.0)

# City Code as below
# (0)Incheon - (1)Seoul - (2)Busan - (3)Daegu - (4)Daejeon - (5)Gwangju - (6)Suwon-si - (7)Ulsan - (8)Jeonju - (9)Cheongju-si - (10)Changwon - (11)Jeju-si - (12)Chuncheon - (13)Hongsung - (14)Muan
```

- (d) The performance of ACO to the best distance vs iterations plot.



- (e) I found that some assumptions I used in `HW2` wasn't correct, I didn't set Incheon as the start and ending country. So I fixed the code in `HW2`.
    - Ant Colony Optimization: Better than Hill Climbing and Random Walk. Ants (particles) have the ability to try different solutions rather than trap in the local optimal solution (we can observe from the plot).
    - Hill Climbing: Though it is better than Random Climb, but this method still has a high probability trapping in the local optimal solution.
    - Random Walk: Search randomly, like buying the lottery.
    - Tabu Search: Better than Hill Climbing due to tabu list. This method can be avoid of trapping in the local optima. I think Tabu Search has the same "power" to find the global optimal solution as Ant Colony Optimization.
    - Simulated Annealing: Though Simulated Annealing is better than Hill Climbing and Random Walk. But comparing with Ant Colony Optimization and Tabu as above mentioned. I think Simulated Annealing doesn't show a good solution on this South Korea Traveling Salesman problem. Maybe because the iteration times is too small, or we need to try different parameters in algorithm.

Reference

- Salehinejad, H., Nezamabadi-pour, H., Saryazdi, S., & Farrahi-Moghaddam, F. (2015). Combined A*-ants algorithm: a new multi-parameter vehicle navigation scheme. *arXiv preprint arXiv:1504.07329*.