M20 - Introduction to Computer Programming with MATLAB
Instructor: Prof. Enrique López Droguett, Ph.D.
Teacher Assistants: C.Yen, G. San Martín, M. Pishahang, S. Malama
Fall 2022 – UCLA
Student: *YaoHsing – Tseng(605528017), Ashley Suh (105692029)*

FINAL PROJECT

(Group 12)

Introduction

The goal of this project is to create algorithms to mimic the actual ant's colony in a virtual environment with barrier and food source. We approach the problem by dividing the algorithms into two subtasks, one is simulation the ants with food, the other is ants without food. With technique we can clearly optimize and observe the behavior of ants during the simulation.

Model and Theory (Approach to solve the algorithm)

- Ant_Simulation routine (where we did the function call)
    - We initialize all the parameters according the read-in map structured data.
    - Initialize the to be tuned parameters with optimize value.
    - Initialize ants' angle in uniform distribution.
    - Initialize colony and food pheromones.
    - We treated ant with and without food in different routine:
        - With food: (goal is to find its way back to colony)
            - Check the surrounding blue pheromones and calculated its potential next move.
                - The Pheromones is mixed with colony's and blue pheromones left by ants without pheromones.
            - Check whether the potential next move is valid according to the information of map boundaries and walls.
            - Find out whether it reached the colony.
        - Without food: (goal is to find the food sources)
            - Check the surrounding red pheromones and calculated is potential next move.
                - The pheromones is mixed with food source and the red pheromones that ants with food left.
            - Check whether the potential next move is valid according to the information of map boundaries and walls.
            - Find out whether it reached the food sources.
        - Updated the pheromones: (filter, remove the useless or outdated information in the map)
            - According to different pheromones we use different rate of decay: deltaR and deltaB.
        - Plot the result and record it in the video format.
        - End

Method and Pseudo-code (function call)

- *Function* ComputerNewAngle:
    - *Inputs:*
        - *Ant's X and Y position, and it's angle*
        - *The specific type of pheromone that ant is looking for*

        - *Concentration of the pheromones*
        - *Range that ant can smell pheromones.*
        - *Sigma_1.*
            - *Parameters for the noise that follows normal distribution, while ant smells pheromones*
        - *Sigma_2*
            - *Parameters for the noise that follows normal distribution, while ant didn't smell any pheromones.*
    - *Outputs:*
        - *The potential new angle for the ant.*
    - *Implementation:*
        - *Initialize the parameters and return value.*
        - *Loop through all the pheromones:*
            - *If pheromones are outside angle rang:*
                - *Continue (find the next pheromone)*
            - *If Pheromones are outside the smell radius range:*
                - *Continue (find the next pheromone)*
            - *Store the index of this pheromone.*
        - *If there are pheromone inside the smelling range:*
            - *Loop through all the inside range pheromones:*
                - *Calculate the weighted position.*
            - *Use the result of the weighted position to find the new angle for the ants.*
            - *Update the current ant angle with the new angle plus noise(mean= 0, standard deviation= sigma_1)*
        - *If there are no pheromone inside the smelling range:*
            - *Update the current ant angle with original angle plus noise(mean = 0, standard deviation= sigma_2)*
- *Function MovemenValidationExecution:*
    - *Input*
        - *Ant's X position, Y position, angle, ant it's step size (speed)*
        - *Map boundaries.*
        - *Walls.*
    - *Output:*
        - *Validated ant's X and Y position, and it's angle.*
    - *Implementation:*
        - *Initialize parameters and return value.*
        - *Calculate the next position:*
            - *if updated angle is either 0 or pi (which will make it impossible to find sin(angle))*
                - *Ant moves in x direction Speed (step size) * cos(angle)*
            - *if updated angle is either pi/2 or 2*pi/3 (which will make it impossible to find cos(angle))*

        - *Ant moves in y direction Speed (step size) * sin(angle)*
      - *If updated angle isn't in above cases:*
        - *Ant moves in x direction Speed (step size) * cos(angle)*
        - *Ant moves in y direction Speed (step size) * sin(angle)*
    - *Check whether next position is in the boundary*
      - *If not turn pi angle. (current next step is not valid)*
      - *Return (current updated position is not valid)*
    - *Check whether next position will hit the wall:*
      - *If hit the wall turn pi angle*
      - *Return (current updated position is not valid)*
    - *If pass all above criteria:*
      - *Return the updated position.*
- *Function CheckFoodProximity:*
  - *Input*
    - *Ant's X position.*
    - *Ant's Y position.*
    - *All the food sources positions.*
    - *Maximum range that ant can grab the food.*
  - *Output:*
    - *Updated food sources*
    - *Indicator that whether ants grab the food.*
  - *Implementation:*
    - *Loop through all the food source*
    - *If the food source is inside the grab range:*
      - *Remove the food source*
      - *Return indicator that indicate the ant grab food*
- *Function CheckColonyProximity:*
  - *Input*
    - *Ant's X and Y position.*
    - *Colony's X and Y position.*
    - *Maximum range that define ant isn't in the colony.*
  - *Output:*
    - *Indicator that tells whether ant is in the colony or not.*
  - *Implementation:*
    - *Calculate the distance between ant and the colony*
    - *If the distance is larger than the maximum range*
      - *Return ants is not in colony*
    - *Else*
      - *Return ant is in the colony.*
- *Function PheromoneUpdate:*
  - *Input*
    - *Specific type of pheromones, and it's concentration.*

        - *Value of the decay of the pheromones.*
    - *Output:*
        - *Updated pheromones list and its concentration.*
    - *Implementation:*
        - *Loop through all the pheromones:*
            - *Decay it's concentration.*
        - *Loop through the decayed pheromones:*
            - *If the concentration is smaller than zero:*
                - *Remove the pheromone from list.*
                - *Remove the concentration from the list.*
        - *Return the updated pheromones and concentration.*

<u>Discussion (Tuning Parameters):</u>

- Delta blue: (the speed that blue pheromone will decay)
    - If delta blue is too large, which indicates blue pheromone will decay too fast, ant's that carries food will not find its way back to colony, and might keep wondering in the space. Also, this situation might trap bunch of the ants into some random coordinate, and no one in the group knows the way home.
    - If delta blue is too small, ants that carry food will be easily misguide by the lost ants that don't carry food, and will easily **overshoot** the angle to the way home.
- Delta red: (the speed that red pheromone will decay)
    - If it decays too fast, ants that still looking for food will not get the signal from its partners that has food.
    - If it decays too slow, once the ants carried with food get lost in the map, it will misguide lots of ants without food to wonder in the void and never get out.
- R smell: (bandwidth for the input data)
    - If R is too larger, ant will get too much information from the surrounding pheromone, which some of them are from the lost ants.
    - If R is too small, ants will get lost in the map very easily for lack of information, and might follow the only source of pheromone, which might come from the lost ants, and get lost in the map.
- Sigma 1: (ability to check whether it's on the right path)
    - If sigma 1 is too large, even the ants have already smell the right pheromone (path) to go home (back to colony), it might still detour to another paths. This will give the ants a high change to get lost in the map.
    - If sigma 1 is too small, the ant's will lose their ability to pull themselves out the wrong path or suboptimal path on its way home. This will also cause the ant's lost in the map.
- Sigma 2: (ability to explore the map)
    - If sigma 2 is too small, the ants will lose its ability to explore the map, which might take ants a long time to find the food sources.
    - If sigma 2 is too large, the ants will keep walking in a zig-zap pattern which will stop it from exploring the map in an efficient way.

- Ways to optimize the parameters: (priority of approach)
  - We divided this process into two different sub-problems:
    - Ant's exploring and tracking abilities.
      - Whether ant's can explore and find the food sources efficiently.
      - Whether ant's can track the right amount of pheromones to make the most efficient decision.
    - Signal filter:
      - Whether the information (the trace) are too much.
      - By using decay to filter out useless or outdated information.
  - Process:
    - Initialize the decay to a midsize number (around 0.2).
      - Make sure that we only deal with the ants tracking issue first.
    - Tuning Sigma 2 first
      - Find the best explore ability for ants to find the food sources as soon as possible.
    - Tuning Sigma 1 and r smell at the same time:
      - Since right now several ants need to find their way back to colony without too much detour, we need the ants smell right amount of information (proper size of r_smell) but also have the ability to follow the right path and pull itself out from the wrong route (sigma 1).
    - If we observe that though the ants can find the right path at home but might detour (overshoot) because there is too few information around the colony:
      - Then we started cut down the decay of the blue pheromones.
    - If we observe that though ants did follow the red pheromone to the food source, but always detour while it almost reached the food source
      - Then we started cut down the decay of the red pheromones.
    - If we find out though there are enough information around the right path but ants still got lost easily, we will go back to optimize the ant's tracking ability again.
  - Result: (the parameters we were using so far)
    - R_smell: **"8**.
    - Sigma_1: **"0.01"**
    - Sigma_2: **"0.6"**
    - DeltaR: **"0.08"**
    - DeltaB: **"0.1"**