

-



Program Structures & Algorithm

Thermostat Controller

Final Project

* **Rishabh Vikramkumar Jain**

**NUID - 001226719**

**THERMOSTAT CONTROLLER**

**Problem Statement:-**

Thermostat works by depending on random air current. The thermostat controls both the heating and cooling systems of homes and etc. The thermostat detects the temperature of the room and compares it to the desired temperature setting. If the room is too warm or too cold, the thermostat requests heat or air depending on the senses.

But most common cause of a thermostat reading the wrong temperature is that it is poorly located. If your thermostat is in direct sunlight, near sources of drafts or not centrally located in your home, it will not give an accurate reading of your home’s temperature levels. And this in return will have erroneous temperature.

**Approach:-**

One of the solution can be to detach the sensors from thermostat and place it separately inside the room and let the sensors communicate with thermostat. The process starts by thermostat sensing the desired temperature of the region and later on that target position will be verified by the means of communication between sensors and later on, with thermostat.

**Algorithm:-**

Above approach can be implemented using Swarm Particle Optimisation where sensors can act as a particles. We can have many sensors and each sensor will start from different position of the room and will be communicating with each other to reach the target position and will communicate the target position temperature to thermostat for an action. We can make use of micro Robots to act as sensors.

Stepwise explanation for Algorithm execution.

1. Input all the desired constraints such as region, temperature, & PSO constants with no. of sensors (swarm size).
2. Initialize the sensors with random positions and velocities.
3. Evaluate the fitness value of system with unit step response.
4. Calculate system constraints for each sensor and total error.
5. Compare the individual fitness value of each sensor to its previous value, if it is better than previous one, replace with new value i.e. local best position otherwise don’t change.
6. The position of sensor having lowest fitness value is global best value.
7. Update position and velocity of sensors according to (1).
8. Go back to step (3) and repeat all steps until system constraints are met Here

For The optimization of parameters the following constants and equation has been used.

Number of Sensors = 30; Size of the Swarm

Iterations = 100; Maximum iterations for solution.

Dimension = 2; Dimensional space

C2 = 1.3; PSO parameter C2, importance of neighborhood best

C1 = 0.14; PSO parameter C1, importance of personal best

w = 0.9; PSO momentum or inertia

**Calculation for fitness Value:-**

**Fitness Value** = Math.sqrt((xTarget - x) \* (xTarget - x) + (yTarget - y) \* (yTarget - y));

Where

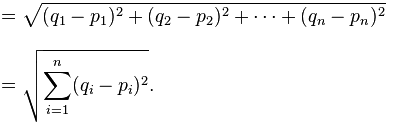
xTarget = X coordinate of Target position.

yTarget = Y coordinate of Target position.

x = X coordinate of current position of a sensor.

y = Y coordinate of current position of a sensor.

Fitness value we get in the equation is basically the distance between the target position and the current position of sensor. By looking at the fitness value we get to know about the proximity of the sensors from the target. Fitness value mentioned above is the **Euclidean Distance Formula i.e.**



**Calculation for Velocity:-**

**Velocity = (**W \* Velocity) + C1 \* ( R1 \* ( Local\_Best\_Coordinate – Current\_Coordinate )) + C2 \* ( R2 \* (Global\_Best\_Coordinate –Current\_Coordinate))

**Next\_Coordinate** = Current\_Coordinate + Velocity;

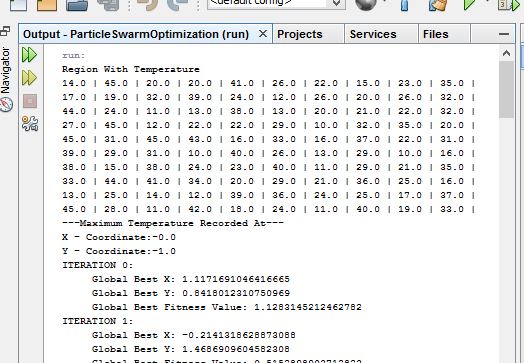
**Iteration v/s Global Fitness Value Trend:-**

**Scatter Plot of X-Y of Global Fitness Values**

**Console Output:-**

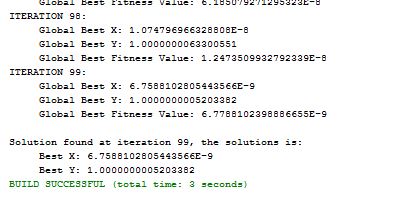
**Start of the program:** shows the Coordinate of highest temperature recorded.

Which is then target for all the sensors for verification.



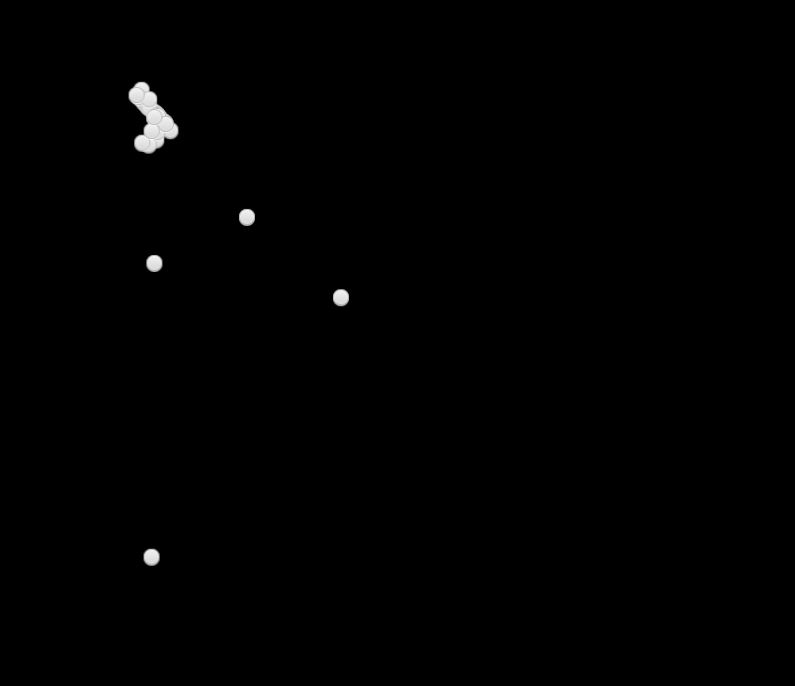
**End of the program:** shows the solution at target position (0,1)

i.e. all the particles nearly converge at that point



**Visualization:-**

Sensors Trying to converge towards the target based on fitness value and velocity:-



**Conclusion:-**

The sensors present in the swarm were able to communicate with each other and were able to find the desired room temperature. This temperature is then sent to the thermostat for verification, on basis of which thermostat can maintain the temperature with no errors.

**References:-**

* Design and Experimental Evaluation of PSO and PID Controller based Wireless Room Heating System. International Journal of Computer Applications (0975 – 8887) Volume 107 – No 5, December 2014

Link: <https://pdfs.semanticscholar.org/3517/5bef8ef8a9285065f4fbd70eb62351d1abc0.pdf>

* Particle Swarm Optimization (PSO) by Gandhi Manalu.
* Wikipedia Particle swarm optimization: <https://en.wikipedia.org/wiki/Particle_swarm_optimization>
* <http://www.swarmintelligence.org/>
* Math Work : https://www.mathworks.com/help/gads/particle-swarm-optimization-algorithm.html?requestedDomain=www.mathworks.com