Let’s consider total number of mobile nodes is: 3

Their X,Y positions are (10, 20), (20,30), (30,40)

Let’s split X,Y co-ordinates into separate list like this

Partx = (10, 20, 30)

Party = (20, 30, 40)

The formula to calculate fitness function, distance and hop count is

$$\sqrt{(x(i)-x(j))^2+(y(i)-y(j))^2}$$

Here total number of nodes is 3, So take i and j values as 3

The equation is,

for (i=0; i<3; i++)

{

for (j=0; j<3; j++)

{

$\sqrt{(x(i)-x(j))^2+(y(i)-y(j))^2}$

}

}

Total number if iterations are 3.

From partx, party list , we take

X(0)=10, X(1)= 20, X(2)=30

Y(0)=20,Y(1)= 30, Y(2)=40

Let’s do the iterations ….

Iteration 1:

In this iteration, execute the X, Y list with j values against I values

```
for (i=0; i<3; i++)
{
for (j=0; j<3; j++)
{
```

$$\sqrt{(x(i)-x(j))^2+(y(i)-y(j))^2}$$

```
}
}
```

Execute the for loop of value J first against values of i

Take **j=0, i=0** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$$\sqrt{\left(x(0)-x(0)\right)^2+\left(y(0)-y(0)\right)^2} \rightarrow \sqrt{(10-10)^2+(20-20)^2} \rightarrow$$

$$\sqrt{(0)^2+(0)^2} \rightarrow \underline{0}$$

Take **j=1, i=0** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$$\sqrt{\left(x(1)-x(0)\right)^2+\left(y(1)-y(0)\right)^2} \rightarrow \sqrt{(20-10)^2+(30-20)^2} \rightarrow$$

$$\sqrt{(10)^2+(10)^2} \rightarrow \sqrt{100+100} \rightarrow \sqrt{200} \rightarrow \underline{14}$$

Take **j=2, i=0** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$$\sqrt{\left(x(2)-x(0)\right)^2+\left(y(2)-y(0)\right)^2} \rightarrow \sqrt{(30-10)^2+(40-20)^2} \rightarrow$$

$$\sqrt{(20)^2+(20)^2} \rightarrow \sqrt{400+400} \rightarrow \sqrt{800} \rightarrow \underline{28}$$

In Iteration1, the fitness values are { 0, 14, 28 } The PBEST value is 28

Iteration 2:

In this iteration, execute the X, Y list with j values against I values

for (i=0; i<3; i++)

{

for (j=0; j<3; j++)

{

$\sqrt{(x(i)-x(j))^2+(y(i)-y(j))^2}$

}

}

Execute the for loop of value J first against values of i

Take **j=0, i=1** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$\sqrt{(x(0)-x(1))^2+(y(0)-y(1))^2}$ ➔ $\sqrt{(10-20)^2+(20-30)^2}$ ➔

$\sqrt{(-10)^2+(-10)^2}$ ➔ $\sqrt{100+100}$ ➔ $\sqrt{200}$ ➔ **14**

Take **j=1, i=1** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$\sqrt{(x(1)-x(1))^2+(y(1)-y(1))^2}$ ➔ $\sqrt{(20-20)^2+(30-30)^2}$ ➔

$\sqrt{(0)^2+(0)^2}$ ➔ **0**

Take **j=2, i=1** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$\sqrt{(x(2)-x(1))^2+(y(2)-y(1))^2}$ ➔ $\sqrt{(30-20)^2+(40-30)^2}$ ➔

$\sqrt{(10)^2+(10)^2}$ ➔ $\sqrt{100+100}$ ➔ $\sqrt{200}$ ➔ **14**

In Iteration2, the fitness values are { 14, 0, 14 } The PBEST value is 14

Iteration 3:

In this iteration, execute the X, Y list with j values against I values

for (i=0; i<3; i++)

{

for (j=0; j<3; j++)

{

$\sqrt{(x(i)-x(j))^2+(y(i)-y(j))^2}$

}

}

Execute the for loop of value J first against values of i

Take **j=0, i=2** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$\sqrt{(x(0)-x(2))^2+(y(0)-y(2))^2}$ ➔ $\sqrt{(10-30)^2+(20-40)^2}$ ➔

$\sqrt{(-20)^2+(-20)^2}$ ➔ $\sqrt{400+400}$ ➔ $\sqrt{800}$ ➔ **28**

Take **j=1, i=2** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$\sqrt{(x(1)-x(2))^2+(y(1)-y(2))^2}$ ➔ $\sqrt{(20-30)^2+(30-40)^2}$ ➔

$\sqrt{(-10)^2+(-10)^2}$ ➔ $\sqrt{100+100}$ ➔ $\sqrt{200}$ ➔ **14**

Take **j=2, i=2** X(0)=10, X(1)= 20, X(2)=30 Y(0)=20,Y(1)= 30, Y(2)=40

So,

$\sqrt{(x(2)-x(2))^2+(y(2)-y(2))^2}$ ➔ $\sqrt{(30-30)^2+(40-40)^2}$ ➔

$\sqrt{(0)^2+(0)^2}$ ➔ **0**

In Iteration3, the fitness values are { 28, 14, 0 } The PBEST value is 28

In Iteration1, the fitness values are {0, 14, 28} The PBEST value is 28

In Iteration2, the fitness values are {14, 0, 14} The PBEST value is 14

In Iteration3, the fitness values are {28, 14, 0} The PBEST value is 28

The GBEST value is 28

After PBEST, GBEST calculations, SPEED and WEIGHT are calculated as follows:

The Speed formula is:

V(t) = Vi(t)+ C1R1(PBEST - Xi)+C2R2(PBEST-Xi)

C1, C2 are learning factors. That can be any value. So in this example we consider C1, C2 as 10

R1, R2 are random numbers. So in this example we consider R1, R2 as 0.5

This should be updated for each iteration. For example, on iteration number 3, speed and weight of the nodes are

V(3) = 30 + (10*0.5)(28-30)+(10*0.5)(28-30)

➔ 30+5*(-2)+5*(-2) ➔ 30+(-10)+(-10) ➔ 10

So, the SPEED update of the nodes at Iteration 3 is 10.

The Weight formula is: W= wmax-(wmax-wmin)(t/T)+u*r

According to research paper, wmax is 0.9, wmin is 0.4, r is a random number. Here we take r is 0.5

For u, the formula is (1/2)*exponential of (-t/T)

T is total iterations, t is current iteration. In this example total iteration are 3, and we consider current iteration as 3

So u = (1/2) * exponential of (-1/1)

➔ 0.5 * exponential of (-1) ➔ 0.5 * (1/-1) ➔ 0.5 * (-1) ➔ -0.5

Let's apply this u value into weight formula,

W= 0.9 – (0.9 – 0.4) (3/3) + (-0.5) * 0.5

➔ 0.9 – 0.5 + (-0.5) * 0.5 ➔ 0.9 – 0.5 – 0.5 * 0.5 ➔ 0.15

So, after 3rd iteration, the weight of the nodes is 0.15

Learning Process

Let's consider p1j, p2j, p3j, p4j = 0.25

Consider the current iteration is 3.

Consider XX = 30. Random value is 0.15

According to Learning strategy 1

If *rand*<P1,j

xxj=2×*gbestj*− *xxj*; //learning strategy 1

Here, rand is less than p1j.

So X1 = 2*28-30 ➔ 56 – 30 ➔ 26

X2 = 2*28-20 ➔ 56-20 ➔ 36

X3 = 2*28-10 ➔ 56-10 ➔ 46

After learning process, the new X values are (26, 36, 46)

The previous X, Y coordinates are

Partx = (10, 20, 30)

Party = (20, 30, 40)

So, the new X, Y coordinates are

Partx = (26, 36, 46)

Party = (20, 30, 40)

Mutation

For mutation, take any random X,Y coordinates and replace the values randomly.

Here, we take 2nd X,Y position that is (36, 30).

Change these values to random value. For example, replace 36 with 52, 30 with 26.

Then the new and X,Y values are (52, 26).

We can take this PSO resultant value as GATEWAY position for our project.