```
%LICNACHAN, LANCE OLIVER C.
%2014-64880

format long;
% rng('shuffle');
% rune=1;
% while rune <= 3
    CurPopNum = 7;
%     if rune == 2
%         CurPopNum = 120;
%     elseif rune == 3
%         CurPopNum = 360;
%     end
%     Y = sprintf('\nCurPopNum: %d\n',CurPopNum);
%     disp(Y);
%
%     mun=1;
%     while mun <= 3
%     Y = sprintf('\nMaxIt: %d\n',(CurPopNum*(5*mun)));
%     disp(Y);


        ProbDim = 4;
        ConsNum = 0;
        RunMax = 20;

        convRuns=0;
        Ans = zeros(RunMax, ProbDim+ConsNum+1+1);
        timeRec = zeros(RunMax,1);

        for trials = 1:RunMax
        tic;
        Y = sprintf('\nTrial: %d\n',trials);
        disp(Y);

        % Variables specific to the problem

        DimMinMax = zeros(ProbDim, 2);
        DimMinMax(1, :) = [0 1];
        DimMinMax(2, :) = [0 1];
        DimMinMax(3, :) = [0 1];
        DimMinMax(4, :) = [0 1];

        % Variables specific to the algorithm
        AcceptThreshold = 1e-5;
        PopNum = CurPopNum;
```

```
% PopNum = 200;
PSO_Curr = 1;
PSO_Max = CurPopNum*5;
c1 = 1.49445;
c2 = 1.49445;
wmax = 0.9;
wmin = 0.4;
Tao = zeros(PopNum, 1);

GA_cross = 0.85;
GA_mut = 0.02;
GA_y = 10;
GA_B = 15;
GA_NumMax = 5;
GA_NumMin = 1;
GA_MinPS = 10;
GA_MaxPS = 5;
GA_MinItr = 10;
GA_MaxItr = 15;
GA_MaxItr = floor(GA_MinItr + ((PSO_Curr/PSO_Max)^GA_B)*(GA_MaxItr-GA_MinItr));
GA_Num = floor(GA_NumMax - ((PSO_Curr/PSO_Max)^GA_y)*(GA_NumMax-GA_NumMin));
GA_PS = floor(GA_MinPS + ((PSO_Curr/PSO_Max)^GA_y)*(GA_MaxPS-GA_MinPS));

TransPos = zeros(PopNum, ProbDim);
TransVel = zeros(PopNum, ProbDim);

% Initialization Step
[PosPop, VelPop] = PSO_InitPop(PopNum, ProbDim, DimMinMax);
FitVal = PSO_GetFitValues(PopNum, PosPop, ProbDim, DimMinMax);
[Pbest, Gbest]= PSO_GetInitPGBest(PopNum, PosPop, ProbDim, FitVal);

PrevDiff = 0;
while PSO_Curr <= PSO_Max
%     disp(Gbest(ProbDim+1));

    % Evaluate
    FitVal = PSO_GetFitValues(PopNum, PosPop, ProbDim, DimMinMax);
%
%       clf;    %clear frame
%       figure(1);
%       hold on;
%       posit = 1:PopNum;
%       plot(posit,FitVal,'.r','MarkerSize', 10);
%       M(PSO_Curr)=getframe(gca);

    if(PSO_Curr == 1)
```

```
    PrevDiff = max(FitVal) - min(FitVal);
else
    CurrDiff = max(FitVal) - min(FitVal);
    % disp(CurrDiff);
    % Check for population convergence
    if PrevDiff - CurrDiff < AcceptThreshold && CurrDiff < AcceptThreshold && PS
        for i = 1:PopNum
            if min(FitVal) == FitVal(i)
                minInd = i;
            end
            if max(FitVal) == FitVal(i)
                maxInd = i;
            end
        end
        mPos = mean(FitVal);
        disp(PosPop(minInd,:));
        seq_route = ObtainSequence(PosPop(minInd,:), ProbDim);
        disp(seq_route);
        X = sprintf('Population Converged!\nNumber of Iterations: %d\nBest Value
        disp(X);
        convRuns = convRuns + 1;
        break;
    end
    PrevDiff = CurrDiff;
end

if(PSO_Curr == PSO_Max)
    % if max gen reached
    break;
end

% Get best values
[Pbest, Gbest] = PSO_GetPGBest(PopNum, PosPop, ProbDim, FitVal, Pbest, Gbest);

% Change value according to how current iteration
w = wmax-(wmax-wmin)*(PSO_Curr/PSO_Max);

% Calculate new velocities and move
[Tao, TransPos, TransVel] = PSO_ChangeVel(PopNum, PosPop, VelPop, ProbDim, Pbest

% Evaluate
TransFitVal = PSO_GetFitValues(PopNum, TransPos, ProbDim, DimMinMax);

% GA Portion
PSO_Arranged = sort(TransFitVal);
GA_Num_Curr = 1;
```

```
while GA_Num_Curr <= GA_Num
    % Get one from best individuals
    for RowNum = 1:PopNum
        if TransFitVal(RowNum) == PSO_Arranged(GA_Num_Curr);
           Sel_Indiv = TransPos(RowNum, :);
           break;
        end
    end

    % Generate a population with the first indiv being the selected
    % chromosome
    GA_Chroms = GA_InitPop(GA_PS, ProbDim, DimMinMax);
    GA_Chroms(1, :) = Sel_Indiv;

    GA_Fit_Elite = PSO_Arranged(GA_Num_Curr);
    GA_Fit_Chrom = Sel_Indiv;
    GA_Curr = 1;
    while GA_Curr <= GA_MaxItr
        % Get Fitness
        GA_FitVal = PSO_GetFitValues(GA_PS, GA_Chroms, ProbDim, DimMinMax);
        TransPop = zeros(GA_PS, ProbDim);

        % Keep Elite
        for i = 1:GA_PS
            if GA_Fit_Elite < GA_FitVal(i)
                GA_Chroms(i,:)=GA_Fit_Chrom;
            end
        end

        % Create Wheel
        GA_RouWheel = GA_CreateWheel(GA_PS, GA_FitVal);

        % Create the population
        for i = 1:GA_PS
            % Select 2 Parents
            [Parent1, Parent2] = GA_Selection(GA_PS, GA_Chroms, ProbDim, GA_FitV
            % Cross-over
            SibRep = GA_CrossOver(Parent1, Parent2, GA_cross, ProbDim);
            % Mutate
            if rand() <= GA_mut
                SibRep = GA_Mutation(SibRep, DimMinMax, ProbDim);
            end
            % Place
            TransPop(i, :) = SibRep;
        end
```

```
            GA_Chroms = TransPop;
            GA_Curr = GA_Curr + 1;
        end
        % Obtain current best
        Arranged = sort(GA_FitVal);
        if Arranged(1) < GA_Fit_Elite
            GA_Fit_Elite = Arranged(1);
            for i = 1:GA_PS
                if Arranged(1) == GA_FitVal(i)
                    GA_Fit_Chrom = GA_Chroms(i,:);
                end
            end
        end
        % Replace the individual
        for RowNum = 1:PopNum
            if TransFitVal(RowNum) == PSO_Arranged(GA_Num_Curr);
                TransPos(RowNum,:) = GA_Fit_Chrom(1,:);
                break;
            end
        end
        GA_Num_Curr = GA_Num_Curr + 1;
    end

    % Update GA_Vars
    GA_MaxItr = floor(GA_MinItr + ((PSO_Curr/PSO_Max)^GA_B)*(GA_MaxItr-GA_MinItr));
    GA_Num = floor(GA_NumMax - ((PSO_Curr/PSO_Max)^GA_y)*(GA_NumMax-GA_NumMin));
    GA_PS = floor(GA_MinPS + ((PSO_Curr/PSO_Max)^GA_y)*(GA_MaxPS-GA_MinPS));

    PosPop = TransPos;
    VelPop = TransVel;
    PSO_Curr = PSO_Curr + 1;
end

if PSO_Curr >= PSO_Max
    for i = 1:PopNum
        if min(FitVal) == FitVal(i)
            minInd = i;
        end
        if max(FitVal) == FitVal(i)
            maxInd = i;
        end
    end

    disp(PosPop(minInd,:));
    seq_route = ObtainSequence(PosPop(minInd,:), ProbDim);
```

```
    disp(seq_route);
    mPos = mean(FitVal);
    X = sprintf('Did Not Converge!\nNumber of Iterations: %d\nBest Value: %0.15f\nW
    disp(X);
end
    %movie(M,1,120);

    if PSO_Curr >= PSO_Max
        Ans(trials,:) = [PosPop(minInd, :) FitVal(minInd) 0];
    else % Converged
        Ans(trials,:) = [PosPop(minInd, :) FitVal(minInd) 1];
    end

    timeRec(trials) = toc;
    X = sprintf('Running Time for this trial: %0.15e\n', timeRec(trials));
    disp(X);
end
if convRuns > 0
    % Get Best Fit
    ConvVals = zeros(convRuns,1);
    i=1;
    for o = 1:RunMax
        if(Ans(o,ProbDim+ConsNum+1+1) == 1)
            ConvVals(i) = Ans(o,ProbDim+1);
            i=i+1;
        end
    end

    Best = min(ConvVals);
    for o = 1:RunMax
        if min(ConvVals) == Ans(o,ProbDim+1)
            BesInd = o;
            break;
        end
    end

    % Generate Stats
    Mean = mean(ConvVals);
    StdDev = std(ConvVals);
    Median = median(ConvVals);
    Worst = max(ConvVals);

else
    % Get Best Fit
    Vals = zeros(RunMax,1);
    for o = 1:RunMax
```

```
                Vals(o) = Ans(o,ProbDim+1);
            end

            % Generate Stats
            Mean = mean(Vals);
            StdDev = std(Vals);
            Median = median(Vals);
            Worst = max(Vals);
            Best = min(Vals);

            % Get index of best run
            BesInd = 0;
            for o = 1:RunMax
                if min(Vals) == Ans(o,ProbDim+1)
                    BesInd = o;
                end
            end
        end

        ConvRatio = convRuns/RunMax;
        totalTime = sum(timeRec);
        aveTime = mean(timeRec);

        X = sprintf('\n\nBest OverAll Value: %0.15f\nMean: %0.15f\nMedian: %0.15f\nStandard
        disp(X);


%         mun=mun+1;
%     end
%     rune = rune+1;
% end
```

```
Trial: 1

  Columns 1 through 3

   0.729488991660910   0.889883185049873   0.202174932889289

  Column 4

   0.123900302222688

     0     4     3     1     2     0

Did Not Converge!
```

```
Number of Iterations: 35
Best Value: 420.000000000000000
Worst Value: 700.000000000000000
Mean: 4.842857142857143e+02

Running Time for this trial: 2.424780127906188e+00


Trial: 2

  Columns 1 through 3

   0.728640416021856   1.035541132610916   0.498963673466467

  Column 4

   0.243182446213001

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 9
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 5.696864297328766e-01


Trial: 3

  Columns 1 through 3

   0.702587821726192   0.887831151200571   0.531972871721281

  Column 4

   0.400698487978326

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 21
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 1.421963700037845e+00
```

```
Trial: 4

  Columns 1 through 3

   0.864163673586256   0.949248178468673   0.827680123818002

  Column 4

   0.361246400310343

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 28
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 2.412758362776902e+00


Trial: 5

  Columns 1 through 3

   0.462487376212944   1.058015558540304   0.418923240730751

  Column 4

   0.371732740567193

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 26
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 2.246303678422903e+00


Trial: 6

  Columns 1 through 3
```

```
   0.677315845089756   0.862901344198479   0.638893033748540

  Column 4

   0.943865791933924

     0     3     1     2     4     0

Population Converged!
Number of Iterations: 13
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 1.023755956802542e+00


Trial: 7

  Columns 1 through 3

   0.535113026119816   0.697107132034374   0.513162477197549

  Column 4

   0.070794937305635

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 16
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 1.307492397863423e+00


Trial: 8

  Columns 1 through 3

   0.973670308737321   0.993241011160684   0.882720441772713

  Column 4

   0.460068922399339
```

```
     0     4     3     1     2     0

Population Converged!
Number of Iterations: 18
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 1.467527452811713e+00


Trial: 9

  Columns 1 through 3

   0.483204104714239   0.654389351834870   0.529747840018292

  Column 4

   0.889522123409579

     0     1     3     2     4     0

Population Converged!
Number of Iterations: 10
Best Value: 450.000000000000000
Worst Value: 450.000000000000000
Mean: 450.000000000000000
Running Time for this trial: 7.644517394994088e-01


Trial: 10

  Columns 1 through 3

   0.421945959221046   0.493555656902096   0.155974196373974

  Column 4

   0.556564008333398

     0     3     1     2     4     0

Population Converged!
Number of Iterations: 8
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
```

```
Mean: 420.000000000000000
Running Time for this trial: 5.813775280544120e-01


Trial: 11

  Columns 1 through 3

   0.353335371827893   0.453373629607537   0.330242509600389

  Column 4

  -0.046398570736131

     0     4     3     1     2     0

Did Not Converge!
Number of Iterations: 35
Best Value: 420.000000000000000
Worst Value: 450.000000000000000
Mean: 4.328571428571428e+02

Running Time for this trial: 2.657062680346744e+00


Trial: 12

  Columns 1 through 3

   0.189996443333244   0.356128889957439   0.490030814977961

  Column 4

   0.697974633520915

     0     1     2     3     4     0

Population Converged!
Number of Iterations: 10
Best Value: 450.000000000000000
Worst Value: 450.000000000000000
Mean: 450.000000000000000
Running Time for this trial: 7.211023876703521e-01


Trial: 13
```

```
  Columns 1 through 3

   0.630072837066814   0.881286985131791   0.534584689847438

  Column 4

   0.034306309167810

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 12
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 9.060675865888369e-01


Trial: 14

  Columns 1 through 3

   0.356402743308675   0.755464173640597   0.332046667027070

  Column 4

   0.087442180090464

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 12
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 9.194764456432835e-01


Trial: 15

  Columns 1 through 3

   0.569505758715181   0.898357926661442   0.384284675375627

  Column 4
```

```
   0.911138017649488

     0     3     1     2     4     0

Did Not Converge!
Number of Iterations: 35
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 4.200000000000000e+02

Running Time for this trial: 2.632542563218161e+00


Trial: 16

  Columns 1 through 3

   0.610537436891293   0.758500488173235   0.578255357504541

  Column 4

   0.789019531366648

     0     3     1     2     4     0

Did Not Converge!
Number of Iterations: 35
Best Value: 420.000000000000000
Worst Value: 480.000000000000000
Mean: 4.285714285714286e+02

Running Time for this trial: 2.652718785160015e+00


Trial: 17

  Columns 1 through 3

   0.644663942379587   0.839127611340013   0.526955584234254

  Column 4

   0.447767104311468

     0     4     3     1     2     0
```

```
Population Converged!
Number of Iterations: 31
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 2.429775116704050e+00


Trial: 18

  Columns 1 through 3

   0.703075200833995   0.975675088993130   0.659941926674831

  Column 4

   0.176321123545787

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 16
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 1.223741108796207e+00


Trial: 19

  Columns 1 through 3

   0.570875996701824   0.813616787865936   0.531711535593876

  Column 4

   0.044928850625505

     0     4     3     1     2     0

Population Converged!
Number of Iterations: 19
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
```

```
Running Time for this trial: 1.611447300884182e+00


Trial: 20

  Columns 1 through 3

   0.689914328808482   0.703131362394839   0.266647045851136

  Column 4

   0.885256093023404

     0     3     1     2     4     0

Population Converged!
Number of Iterations: 18
Best Value: 420.000000000000000
Worst Value: 420.000000000000000
Mean: 420.000000000000000
Running Time for this trial: 1.385027577611766e+00



Best OverAll Value: 420.000000000000000
Mean: 423.750000000000000
Median: 420.000000000000000
Standard Deviation:10.246950765959598
Worst Best Overall Value: 450.000000000000000
Number of Converged Runs: 16
Ratio of Convergence: 8.000000000000000e-01
Total Running Time for all trials: 3.135905892653181e+01
Average running time: 1.567952946326590e+00
```