Lab 2: finding good solutions to an NP-hard problem

This document is divided into following sections :

A. Introduction to different techniques applied and problems faced
B. Table collating all the results for different number of teams
C. Code Testing for inner functions of swap in the simulated annealing algorithm
D. Result Files from different runs taken (I have attached all the different runs taken for different values of various parameters)

Lab Overview

The lab involved

1. Generating Initial Random Solutions with only hard constraints for the Travelling Tournament Problem
2. Use Simulated Annealing to generate optimal solutions when minimized with respect to the cost. The final solution should be satisfying both the hard and the soft constraints

Hard Constraints - Should be a round robin tournament.
Soft Constraints - Away and Home games should not be consecutive.
- No three consecutive home or away games for each team.

Run-time and quality of results tradeoffs when varying selected parameters

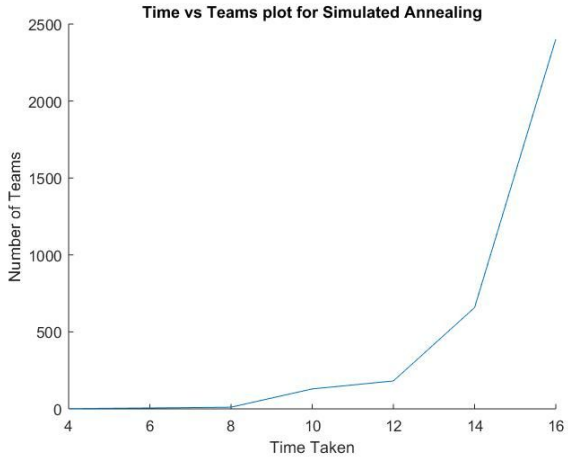
1. Varying maxR,maxP,maxC
   As we can see increasing these values considerably increases the time taken by the simulated annealing algorithm, this is because of the multiple iterations it has to go through before settling for the best feasible solution.

   Additionally, it is noted that increasing maxR, increases the time taken considerably, this is because the reheats increase and hence the solution iterates over a larger number of possible local optimas.

   Performance wise, the increase in number of iterations makes the explore the input space more thoroughly, hence we achieve a more optimal solution each time. (Please see the attached table for the improved solution values.)

2. Increasing number of teams

Increasing the number of teams has an adverse effect on time taken as shown in the plot below, this is because the code has non linear complexity:



3. Optimizations for High number of teams(>10)

It was observed that the solution was not converging for the default set of values, hence I tried changing the values of weights (w) and beta (temperature reheat value).

I used w = 9000 as compared to w = 4000 and changed beta from 0.9999 to 0.998, this helped the algorithm converge faster.

4. Optimizations for Low number of teams

```
maxR = 10,maxP = 10,maxC = 10
T = 400,theta = 1.04,sigma = 1.04
beta = 0.9999,w = 4000
```

5. Problems Faced

a. I used back propagation using recursion to build the initial solution, so for a large number of teams, the recursion was taking a lot of time to converge. To remove this particular problem, I added a soft constraint while creating the initial solution. This ensured that the choices are limited, hence the converging of the back-propagation algorithm was faster.
b. For a large number of teams I had to execute the initial solution creation part multiple number of times, to ensure it didn't get stuck in a bad position.
c. Each execution for the later part of the simulated annealing was taking a lot of time, so I used multiple terminals to execute the code remotely on the rlogin nodes of CS department, VT. These results were saved to a file to analyse later.

Parameter Values of TTSA on the TTP with varying maxR, maxP, maxC,n										
n	maxR	maxP	maxC	Best(Jan 29, 2016)	Initial	Final Feasible	Time	% Cost as compared to base case	% Time as compared to base case	% Cost with respect to the best solution
4	10	10	10	8276	8797	8276	1.832091808	100	100	1
	10	10	100		9166	8276	12.81700325	100	699.5830226	1
	10	100	10		10649	8276	13.9462533	100	761.2202203	1
6	10	10	10	22969	27844	26062	5.844084501	100	100	1.134659759
	10	10	100		27777	24910	42.71871161	95.57977131	730.9735444	1.084505203
	10	100	10		29481	25138	47.72842741	96.45460824	816.6963944	1.094431625
	10	100	100		31238	24075	358.3209333	92.37587292	6131.344153	1.048151857
	30	10	10		30774	26060	10.32293105	99.99232599	176.6389765	1.134572685
8	10	10	10	39721	54854.25476	48460	10.838763	100	100	1.220009567
	10	10	100		52376.14286	45825	125.4375732	94.56252579	1157.30525	1.153671861
	10	100	10		54855.83664	45232	127.922797	93.33883615	1180.234285	1.138742731
	10	100	100		54401.92806	43888	1165.497547	90.56541478	10753.04947	1.104906724
	30	10	10		56211.19185	47583	29.73136497	98.19026001	274.3058868	1.197930566
10	10	10	10	58831	89806	79847	129.56	100	100	1.357226632
	10	100	100		94222	72707	676	91.05789823	521.7659772	1.235862046
12	10	10	10	108629	167053.8358	136424.5822	181.2309577			1.25587626
14	10	10	10	183354	327734.1536	260150	657.9813654			1.418840058
16	10	10	10	249477	455100.8922	372590	2402.152			1.493484369

Solutions for 10,12,14 and 16 were obtained by increasing w, decreasing beta, hence they should not be directly compared with the other solutions

In [1]:

```
from util import Schedule
import hardcode
import config
```

Testing the working of the inner functions

In [2]:

```
# Prints a randomly generated schedule
TTSA = Schedule(n=6,hardcoded=False)
TTSA.printSchedule(TTSA.scheduleMap)
print(''' Violations = {} '''.format(TTSA.getViolations(S=TTSA.scheduleMap)))
```

```
Generated Schedule and Distance Map
Rounds - >
         1     2     3     4     5     6     7     8     9     10
Teams
   1     4    -3    6    5    -6    -4    -5    2    3    -2
   2     -6    -5    4    -3    5    3    6    -1    -4    1
   3     5    1    -5    2    -4    -2    4    6    -1    -6
   4     -1    6    -2    -6    3    1    -3    -5    2    5
   5     -3    2    3    -1    -2    6    1    4    -6    -4
   6     2    -4    -1    4    1    -5    -2    -3    5    3
 Violations = 0
```

In [3]:

```
# Generates the schdeule listed in the paper.
TTSA = Schedule(n=6,hardcoded=True)
TTSA.printSchedule(TTSA.scheduleMap)
print(''' Violations = {} '''.format(TTSA.getViolations(S=TTSA.scheduleMap)))
```

```
Generated Schedule and Distance Map
Rounds - >
         1     2     3     4     5     6     7     8     9     10
Teams
   1     6    -2    4    3    -5    -4    -3    5    2    -6
   2     5    1    -3    -6    4    3    6    -4    -1    -5
   3     -4    5    2    -1    6    -2    1    -6    -5    4
   4     3    6    -1    -5    -2    1    5    2    -6    -3
   5     -2    -3    6    4    1    -6    -4    -1    3    2
   6     -1    -4    -5    2    -3    5    -2    3    4    1
 Violations = 0
```

In [4]:

```
# Swap the Home and Away games of two teams
TTSA.swapHomes(teamA=2,teamB=4)
TTSA.printSchedule(TTSA.scheduleMap)
print(''' Violations = {} '''.format(TTSA.getViolations(S=TTSA.scheduleMap)))
```

```
Rounds - >
        1     2     3     4     5     6     7     8     9     10
Teams
   1     6    -2    4    3    -5    -4    -3    5    2    -6
   2     5    1    -3    -6    -4    3    6    4    -1    -5
   3     -4    5    2    -1    6    -2    1    -6    -5    4
   4     3    6    -1    -5    2    1    5    -2    -6    -3
   5     -2    -3    6    4    1    -6    -4    -1    3    2
   6     -1    -4    -5    2    -3    5    -2    3    4    1
 Violations = 0
```

In [5]:

```
# Swap two rounds completely
TTSA.swapRounds(roundA=2,roundB=4)
TTSA.printSchedule(TTSA.scheduleMap)
print(''' Violations = {} '''.format(TTSA.getViolations(S=TTSA.scheduleMap)))
```

```
Rounds - >
        1     2     3     4     5     6     7     8     9     10
Teams
   1     6    3    4    -2    -5    -4    -3    5    2    -6
   2     5    -6    -3    1    -4    3    6    4    -1    -5
   3     -4    -1    2    5    6    -2    1    -6    -5    4
   4     3    -5    -1    6    2    1    5    -2    -6    -3
   5     -2    4    6    -3    1    -6    -4    -1    3    2
   6     -1    2    -5    -4    -3    5    -2    3    4    1
 Violations = 2
```

In [6]:

```
# Swap the schedules of two teams
TTSA.swapTeams(teamA=2,teamB=4)
TTSA.printSchedule(TTSA.scheduleMap)
print(''' Violations = {} '''.format(TTSA.getViolations(S=TTSA.scheduleMap)))
```

```
Rounds - >
        1     2     3     4     5     6     7     8     9     10
Teams
   1     6    3    2    -4    -5    -2    -3    5    4    -6
   2     3    -5    -1    6    -4    1    5    4    -6    -3
   3     -2    -1    4    5    6    -4    1    -6    -5    2
   4     5    -6    -3    1    2    3    6    -2    -1    -5
   5     -4    2    6    -3    1    -6    -2    -1    3    4
   6     -1    4    -5    -2    -3    5    -4    3    2    1
 Violations = 2
```

In [7]:

```
# Partial Swap (with hard constraints) on two rounds
TTSA.partialSwapRounds(team=2,roundA=2,roundB=9)
TTSA.printSchedule(TTSA.scheduleMap)
print(''' Violations = {} '''.format(TTSA.getViolations(S=TTSA.scheduleMap)))
```

```
Rounds - >
         1     2     3     4     5     6     7     8     9     10
Teams
   1     6    4    2    -4    -5    -2    -3    5    3    -6
   2     3    -6    -1    6    -4    1    5    4    -5    -3
   3     -2    -5    4    5    6    -4    1    -6    -1    2
   4     5    -1    -3    1    2    3    6    -2    -6    -5
   5     -4    3    6    -3    1    -6    -2    -1    2    4
   6     -1    2    -5    -2    -3    5    -4    3    4    1
 Violations = 2
```

Distance Map

This is from CMU website containing the standard solution:http://mat.gsia.cmu.edu/TOURN (http://mat.gsia.cmu.edu/TOURN)

In [8]:

```
print(TTSA.distanceMap)
```

```
[[   0  745  665  929  605  521]
 [ 745    0   80  337 1090  315]
 [ 665   80    0  380 1020  257]
 [ 929  337  380    0 1380  408]
 [ 605 1090 1020 1380    0 1010]
 [ 521  315  257  408 1010    0]]
```

```
Summary
Hardcoded Solution = False
n = 4
Distance Map =
[[  0 745 665 929]
 [745   0  80 337]
 [665  80   0 380]
 [929 337 380   0]]

initial Solution =
[[ 4  2  3 -4 -2 -3]
 [-3 -1 -4  3  1  4]
 [ 2  4 -1 -2 -4  1]
 [-1 -3  2  1  3 -2]]

initial Cost = 8797

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 6377.638038145492
w = 3287.7084270374075
bestT = 3188.819019072746

bestInfeasible = 11225.56296139174
bestFeasible = 8276

time = 1.8320918083190918



Best Infeasible Solution ->

[[ 4  2  3 -3 -2 -4]
 [-3 -1  4 -4  1  3]
 [ 2  4 -1  1 -4 -2]
 [-1 -3 -2  2  3  1]]

Cost = 11430.524881030793
Violations = 4



Best Feasible Solution ->
```

```
[[-4 -2 -3  2  4  3]
 [ 3  1  4 -1 -3 -4]
 [-2 -4  1  4  2 -1]
 [ 1  3 -2 -3 -1  2]]

Cost = 8276
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 4
Distance Map =
[[  0 745 665 929]
 [745   0  80 337]
 [665  80   0 380]
 [929 337 380   0]]

initial Solution =
[[ 2  4  3 -4 -3 -2]
 [-1 -3 -4  3  4  1]
 [-4  2 -1 -2  1  4]
 [ 3 -1  2  1 -2 -3]]

initial Cost = 9166

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 100
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 1578.0705646503095
w = 3555.9854346836596
bestT = 789.0352823251548

bestInfeasible = 11731.22018156775
bestFeasible = 8276

time = 12.81700325012207



Best Infeasible Solution ->

[[ 4  2  3 -3 -4 -2]
 [-3 -1  4 -4  3  1]
 [ 2  4 -1  1 -2 -4]
 [-1 -3 -2  2  1  3]]

Cost = 11960.51518128771
Violations = 4



Best Feasible Solution ->
```

```
[[ 3  2  4 -3 -2 -4]
 [ 4 -1 -3 -4  1  3]
 [-1  4  2  1 -4 -2]
 [-2 -3 -1  2  3  1]]

Cost = 8276
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 4
Distance Map =
[[  0 745 665 929]
 [745   0  80 337]
 [665  80   0 380]
 [929 337 380   0]]

initial Solution =
[[-2 -4  3  4 -3  2]
 [ 1  3 -4 -3  4 -1]
 [ 4 -2 -1  2  1 -4]
 [-3  1  2 -1 -2  3]]

initial Cost = 10649

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 100
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 1579.3335893968533
w = 3161.258102920584
bestT = 789.6667946984267

bestInfeasible = 11032.677008483935
bestFeasible = 8276

time = 13.946253299713135



Best Infeasible Solution ->

[[ 4  2  3 -3 -2 -4]
 [-3 -1  4 -4  1  3]
 [ 2  4 -1  1 -4 -2]
 [-1 -3 -2  2  3  1]]

Cost = 11225.56296139174
Violations = 4



Best Feasible Solution ->
```

```
[[ 3  2  4 -3 -2 -4]
 [ 4 -1 -3 -4  1  3]
 [-1  4  2  1 -4 -2]
 [-2 -3 -1  2  3  1]]

Cost = 8276
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 4
Distance Map =
[[  0 745 665 929]
 [745   0  80 337]
 [665  80   0 380]
 [929 337 380   0]]

initial Solution =
[[-2 -4  2  3  4 -3]
 [ 1  3 -1 -4 -3  4]
 [-4 -2  4 -1  2  1]
 [ 3  1 -3  2 -1 -2]]

initial Cost = 10373

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 100
maxC = 100
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 797.9225979211958
w = 2922.7608200079358
bestT = 398.9612989605979

bestInfeasible = 10680.85346311717
bestFeasible = 8276

time = 109.29898071289062



Best Infeasible Solution ->

[[ 3  2  4 -4 -2 -3]
 [ 4 -1 -3  3  1 -4]
 [-1  4  2 -2 -4  1]
 [-2 -3 -1  1  3  2]]

Cost = 10851.293153138455
Violations = 4



Best Feasible Solution ->
```

```
[[ 3  4  2 -3 -2 -4]
 [-4 -3 -1  4  1  3]
 [-1  2  4  1 -4 -2]
 [ 2 -1 -3 -2  3  1]]

Cost = 8276
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 4
Distance Map =
[[  0 745 665 929]
 [745   0  80 337]
 [665  80   0 380]
 [929 337 380   0]]

initial Solution =
[[ 4 -3 -4  2  3 -2]
 [-3  4  3 -1 -4  1]
 [ 2  1 -2 -4 -1  4]
 [-1 -2  1  3  2 -3]]

initial Cost = 10417

initial Violations = 0


Initial Parameters

maxR = 30
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 3193.6060763535506
w = 3698.224852071006
bestT = 1596.8030381767753

bestInfeasible = 11960.51518128771
bestFeasible = 8276

time = 2.876742124557495



Best Infeasible Solution ->

[[ 2  4  3 -3 -2 -4]
 [-1 -3  4 -4  1  3]
 [ 4  2 -1  1 -4 -2]
 [-3 -1 -2  2  3  1]]

Cost = 12203.67171820849
Violations = 4



Best Feasible Solution ->
```

```
[[-4 -2 -3  4  2  3]
 [ 3  1 -4 -3 -1  4]
 [-2 -4  1  2  4 -1]
 [ 1  3  2 -1 -3 -2]]

Cost = 8276
Violations = 0
```

```
Summary
Hardcoded Solution = True
n = 6
Distance Map =
[[   0  745  665  929  605  521]
 [ 745    0   80  337 1090  315]
 [ 665   80    0  380 1020  257]
 [ 929  337  380    0 1380  408]
 [ 605 1090 1020 1380    0 1010]
 [ 521  315  257  408 1010    0]]

initial Solution =
[[ 6 -2  4  3 -5 -4 -3  5  2 -6]
 [ 5  1 -3 -6  4  3  6 -4 -1 -5]
 [-4  5  2 -1  6 -2  1 -6 -5  4]
 [ 3  6 -1 -5 -2  1  5  2 -6 -3]
 [-2 -3  6  4  1 -6 -4 -1  3  2]
 [-1 -4 -5  2 -3  5 -2  3  4  1]]

initial Cost = 27844

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 1595.0474328130322
w = 5263.727116943362
bestT = 797.5237164065161

bestInfeasible = 24065.14051997321
bestFeasible = 26062

time = 5.8440845012664795



Best Infeasible Solution ->

[[-5  3  5 -2 -3 -4 -6  4  2  6]
 [-3  6  3  1 -6  5  4 -5 -1 -4]
 [ 2 -1 -2 -4  1  6  5 -6  4 -5]
 [ 6 -5 -6  3  5  1 -2 -1 -3  2]
 [ 1  4 -1 -6 -4 -2 -3  2  6  3]
 [-4 -2  4  5  2 -3  1  3 -5 -1]]

Cost = 23967.15976417825
```

```
Violations = 2



Best Feasible Solution ->

[[ 4  3 -4 -3  5  2  6 -5 -2 -6]
 [-6 -5  6  5 -4 -1 -3  4  1  3]
 [-5 -1  5  1 -6  4  2  6 -4 -2]
 [-1 -6  1  6  2 -3 -5 -2  3  5]
 [ 3  2 -3 -2 -1  6  4  1 -6 -4]
 [ 2  4 -2 -4  3 -5 -1 -3  5  1]]

Cost = 26062
Violations = 0
```

```
Summary
Hardcoded Solution = True
n = 6
Distance Map =
[[   0  745  665  929  605  521]
 [ 745    0   80  337 1090  315]
 [ 665   80    0  380 1020  257]
 [ 929  337  380    0 1380  408]
 [ 605 1090 1020 1380    0 1010]
 [ 521  315  257  408 1010    0]]

initial Solution =
[[-4 -2  4  3 -5  6 -3  5  2 -6]
 [ 3  1 -3 -6  4  5  6 -4 -1 -5]
 [-2  5  2 -1  6 -4  1 -6 -5  4]
 [ 1  6 -1 -5 -2  3  5  2 -6 -3]
 [-6 -3  6  4  1 -2 -4 -1  3  2]
 [ 5 -4 -5  2 -3 -1 -2  3  4  1]]

initial Cost = 27777

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 100
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 797.3642196384719
w = 6157.816225260313
bestT = 398.68210981923596

bestInfeasible = 23672.824509386533
bestFeasible = 24910

time = 42.718711614608765



Best Infeasible Solution ->

[[-5 -3  5  3 -2 -4 -6  4  2  6]
 [ 3  4 -3 -4  1  6  5 -6 -1 -5]
 [-2  1  2 -1  6  5  4 -5 -6 -4]
 [ 6 -2 -6  2  5  1 -3 -1 -5  3]
 [ 1  6 -1 -6 -4 -3 -2  3  4  2]
 [-4 -5  4  5 -3 -2  1  2  3 -1]]

Cost = 23678.513103030604
```

```
Violations = 1



Best Feasible Solution ->

[[ 5 -4 -6 -5  6 -3 -2  3  4  2]
 [ 3  6  5 -3 -5  4  1 -4 -6 -1]
 [-2  5 -4  2  4  1 -6 -1 -5  6]
 [ 6  1  3 -6 -3 -2  5  2 -1 -5]
 [-1 -3 -2  1  2  6 -4 -6  3  4]
 [-4 -2  1  4 -1 -5  3  5  2 -3]]

Cost = 24910
Violations = 0
```

```
Summary
Hardcoded Solution = True
n = 6
Distance Map =
[[   0  745  665  929  605  521]
 [ 745    0   80  337 1090  315]
 [ 665   80    0  380 1020  257]
 [ 929  337  380    0 1380  408]
 [ 605 1090 1020 1380    0 1010]
 [ 521  315  257  408 1010    0]]

initial Solution =
[[ 4 -2 -4  3 -5  6 -3  5  2 -6]
 [ 3  1 -3 -6  4  5  6 -4 -1 -5]
 [-2  5  2 -1  6 -4  1 -6 -5  4]
 [-1  6  1 -5 -2  3  5  2 -6 -3]
 [-6 -3  6  4  1 -2 -4 -1  3  2]
 [ 5 -4 -5  2 -3 -1 -2  3  4  1]]

initial Cost = 29481

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 100
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 1555.6636266675025
w = 5061.276073984001
bestT = 777.8318133337513

bestInfeasible = 23451.51145573977
bestFeasible = 25138

time = 47.72842741012573



Best Infeasible Solution ->

[[-5 -2  6  5 -6 -3 -4  2  3  4]
 [ 4  1 -3 -4  3  5  6 -1 -5 -6]
 [ 6  4  2 -6 -2  1  5 -4 -1 -5]
 [-2 -3 -5  2  5  6  1  3 -6 -1]
 [ 1  6  4 -1 -4 -2 -3 -6  2  3]
 [-3 -5 -1  3  1 -4 -2  5  4  2]]

Cost = 23492.679636369347
```

```
Violations = 2



Best Feasible Solution ->

[[ 4  2 -6 -4  6  3  5 -3 -2 -5]
 [ 6 -1 -5 -6  5  4 -3 -4  1  3]
 [-5  6  4  5 -4 -1  2  1 -6 -2]
 [-1 -5 -3  1  3 -2 -6  2  5  6]
 [ 3  4  2 -3 -2  6 -1 -6 -4  1]
 [-2 -3  1  2 -1 -5  4  5  3 -4]]

Cost = 25138
Violations = 0
```

```
Summary
Hardcoded Solution = True
n = 6
Distance Map =
[[   0  745  665  929  605  521]
 [ 745    0   80  337 1090  315]
 [ 665   80    0  380 1020  257]
 [ 929  337  380    0 1380  408]
 [ 605 1090 1020 1380    0 1010]
 [ 521  315  257  408 1010    0]]

initial Solution =
[[-6 -2 -4  3 -5  6 -3  5  2  4]
 [-5  1 -3 -6  4  5  6 -4 -1  3]
 [ 4  5  2 -1  6 -4  1 -6 -5 -2]
 [-3  6  1 -5 -2  3  5  2 -6 -1]
 [ 2 -3  6  4  1 -2 -4 -1  3 -6]
 [ 1 -4 -5  2 -3 -1 -2  3  4  5]]

initial Cost = 31238

initial Violations = 0


Initial Parameters

maxR = 10
maxP = 100
maxC = 100
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 1577.9127575938444
w = 6157.816225260313
bestT = 788.9563787969222

bestInfeasible = 22980.813361811026
bestFeasible = 24075

time = 358.32093334198



Best Infeasible Solution ->

[[-5 -4  3  5 -3 -2 -6  4  2  6]
 [ 3  6  4 -3 -4  1  5 -6 -1 -5]
 [-2 -5 -1  2  1  6 -4  5 -6  4]
 [ 6  1 -2 -6  2  5  3 -1 -5 -3]
 [ 1  3  6 -1 -6 -4 -2 -3  4  2]
 [-4 -2 -5  4  5 -3  1  2  3 -1]]

Cost = 22980.813361811026
```

```
Violations = 1



Best Feasible Solution ->

[[ 5  2  4 -3 -4 -6  3  6 -2 -5]
 [-3 -1 -5  6  5 -4 -6  4  1  3]
 [ 2 -4 -6  1  6  5 -1 -5  4 -2]
 [ 6  3 -1 -5  1  2  5 -2 -3 -6]
 [-1 -6  2  4 -2 -3 -4  3  6  1]
 [-4  5  3 -2 -3  1  2 -1 -5  4]]

Cost = 24075
Violations = 0
```

```
Summary
Hardcoded Solution = True
n = 6
Distance Map =
[[   0  745  665  929  605  521]
 [ 745    0   80  337 1090  315]
 [ 665   80    0  380 1020  257]
 [ 929  337  380    0 1380  408]
 [ 605 1090 1020 1380    0 1010]
 [ 521  315  257  408 1010    0]]

initial Solution =
[[-6 -2 -4 -3 -5  6  3  5  2  4]
 [-5  1 -3 -6  4  5  6 -4 -1  3]
 [ 4  5  2  1  6 -4 -1 -6 -5 -2]
 [-3  6  1 -5 -2  3  5  2 -6 -1]
 [ 2 -3  6  4  1 -2 -4 -1  3 -6]
 [ 1 -4 -5  2 -3 -1 -2  3  4  5]]

initial Cost = 30774.704896174087

initial Violations = 4


Initial Parameters

maxR = 30
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 1598.400719808034
w = 6157.816225260313
bestT = 799.200359904017

bestInfeasible = 24702.68722867398
bestFeasible = 26060

time = 10.322931051254272



Best Infeasible Solution ->

[[-5 -2 -4  2  5 -3 -6  4  3  6]
 [-3  1 -5 -1  3 -6 -4  5  6  4]
 [ 2 -4  6  4 -2  1  5 -6 -1 -5]
 [-6  3  1 -3  6  5  2 -1 -5 -2]
 [ 1  6  2 -6 -1 -4 -3 -2  4  3]
 [ 4 -5 -3  5 -4  2  1  3 -2 -1]]

Cost = 24690.40902395892
```

```
Violations = 1



Best Feasible Solution ->

[[-5 -3 -4  5  4  2 -6 -2  3  6]
 [-6 -5 -3  6  3 -1  4  1  5 -4]
 [ 4  1  2 -4 -2  6  5 -6 -1 -5]
 [-3 -6  1  3 -1 -5 -2  5  6  2]
 [ 1  2  6 -1 -6  4 -3 -4 -2  3]
 [ 2  4 -5 -2  5 -3  1  3 -4 -1]]

Cost = 26060
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 8
Distance Map =
[[   0  745  665  929  605  521  370  587]
 [ 745    0   80  337 1090  315  567  712]
 [ 665   80    0  380 1020  257  501  664]
 [ 929  337  380    0 1380  408  622  646]
 [ 605 1090 1020 1380    0 1010  957 1190]
 [ 521  315  257  408 1010    0  253  410]
 [ 370  567  501  622  957  253    0  250]
 [ 587  712  664  646 1190  410  250    0]]

initial Solution =
[[ 7 -6  2  8 -7  5 -2  6 -5 -3 -4  3  4 -8]
 [-5 -7 -1 -3 -8 -6  1  5  3 -4  6  7  8  4]
 [-8  4  6  2  5  7 -4  8 -2  1 -5 -1 -6 -7]
 [ 6 -3 -5  7 -6 -8  3 -7  8  2  1  5 -1 -2]
 [ 2  8  4 -6 -3 -1 -8 -2  1 -7  3 -4  7  6]
 [-4  1 -3  5  4  2 -7 -1  7 -8 -2  8  3 -5]
 [-1  2  8 -4  1 -3  6  4 -6  5 -8 -2 -5  3]
 [ 3 -5 -7 -1  2  4  5 -3 -4  6  7 -6 -2  1]]

initial Cost = 54854.25476

initial Violations = 6


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 799.520119984
w = 7203.77402203
bestT = 399.760059992

bestInfeasible = 45271.0492159
bestFeasible = 48460

time = 10.8387629986



Best Infeasible Solution ->

[[ 7 -5  4  5  8 -7 -2 -4  3  6  2 -8 -6 -3]
 [ 6  8 -7 -8 -6  4  1 -3  7 -5 -1 -4  3  5]
 [-5 -7 -8  6  7 -6  4  2 -1  8 -4  5 -2  1]
 [ 8 -6 -1 -7 -5 -2 -3  1  6  7  3  2  5 -8]
```

```
 [ 3  1  6 -1  4  8 -6 -7 -8  2  7 -3 -4 -2]
 [-2  4 -5 -3  2  3  5 -8 -4 -1  8 -7  1  7]
 [-1  3  2  4 -3  1  8  5 -2 -4 -5  6 -8 -6]
 [-4 -2  3  2 -1 -5 -7  6  5 -3 -6  1  7  4]]

Cost = 45314.2693217
Violations = 2



Best Feasible Solution ->

[[ 6 -5  2  5  7 -3 -8 -2  8  4  3 -4 -6 -7]
 [ 4 -6 -1  8 -5 -8 -3  1  7  6 -4 -7  5  3]
 [-5 -7 -8  7 -6  1  2 -4  6  8 -1  5  4 -2]
 [-2 -8  7  6  8 -7 -6  3 -5 -1  2  1 -3  5]
 [ 3  1  6 -1  2 -6 -7 -8  4  7  8 -3 -2 -4]
 [-1  2 -5 -4  3  5  4 -7 -3 -2  7 -8  1  8]
 [-8  3 -4 -3 -1  4  5  6 -2 -5 -6  2  8  1]
 [ 7  4  3 -2 -4  2  1  5 -1 -3 -5  6 -7 -6]]

Cost = 48460
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 8
Distance Map =
[[   0  745  665  929  605  521  370  587]
 [ 745    0   80  337 1090  315  567  712]
 [ 665   80    0  380 1020  257  501  664]
 [ 929  337  380    0 1380  408  622  646]
 [ 605 1090 1020 1380    0 1010  957 1190]
 [ 521  315  257  408 1010    0  253  410]
 [ 370  567  501  622  957  253    0  250]
 [ 587  712  664  646 1190  410  250    0]]

initial Solution =
[[ 6  8 -6 -3 -8 -5 -4 -2  3  2  4  7  5 -7]
 [-4  3 -7 -5 -6 -8 -3  1  6 -1  5  4  7  8]
 [ 8 -2  4  1  7  6  2 -4 -1  5 -7 -8 -6 -5]
 [ 2  5 -3 -8 -5  7  1  3 -7  6 -1 -2  8 -6]
 [-7 -4  8  2  4  1  6  7 -8 -3 -2 -6 -1  3]
 [-1 -7  1  7  2 -3 -5  8 -2 -4 -8  5  3  4]
 [ 5  6  2 -6 -3 -4  8 -5  4 -8  3 -1 -2  1]
 [-3 -1 -5  4  1  2 -7 -6  5  7  6  3 -4 -2]]

initial Cost = 52376.1428565

initial Violations = 9


Initial Parameters

maxR = 10
maxP = 10
maxC = 100
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 799.040527824
w = 7791.60198223
bestT = 399.520263912

bestInfeasible = 43064.4056544
bestFeasible = 45825

time = 125.437573195



Best Infeasible Solution ->

[[ 8  4 -3 -8  7 -4 -6 -2  6 -7  5  3  2 -5]
 [ 3  6  7 -4 -6  5 -8  1 -3  4  8 -5 -1 -7]
 [-2  7  1 -7 -8  6  5 -4  2  8 -6 -1 -5  4]
 [-6 -1 -5  2  5  1 -7  3  8 -2  7  6 -8 -3]
```

```
 [ 7  8  4 -6 -4 -2 -3 -8 -7  6 -1  2  3  1]
 [ 4 -2  8  5  2 -3  1  7 -1 -5  3 -4 -7 -8]
 [-5 -3 -2  3 -1  8  4 -6  5  1 -4 -8  6  2]
 [-1 -5 -6  1  3 -7  2  5 -4 -3 -2  7  4  6]]

Cost = 43069.4124924
Violations = 1



Best Feasible Solution ->

[[ 5 -6 -7 -4  3 -2 -3  6  8  4 -8 -5  2  7]
 [ 3  5 -3  6  4  1 -6 -4  7 -8 -7  8 -1 -5]
 [-2  4  2 -5 -1 -8  1  7 -6 -7  5 -4  8  6]
 [ 7 -3  5  1 -2 -7 -8  2 -5 -1 -6  3  6  8]
 [-1 -2 -4  3 -8 -6 -7  8  4  6 -3  1  7  2]
 [-8  1  8 -2  7  5  2 -1  3 -5  4 -7 -4 -3]
 [-4 -8  1  8 -6  4  5 -3 -2  3  2  6 -5 -1]
 [ 6  7 -6 -7  5  3  4 -5 -1  2  1 -2 -3 -4]]

Cost = 45825
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 8
Distance Map =
[[   0  745  665  929  605  521  370  587]
 [ 745    0   80  337 1090  315  567  712]
 [ 665   80    0  380 1020  257  501  664]
 [ 929  337  380    0 1380  408  622  646]
 [ 605 1090 1020 1380    0 1010  957 1190]
 [ 521  315  257  408 1010    0  253  410]
 [ 370  567  501  622  957  253    0  250]
 [ 587  712  664  646 1190  410  250    0]]

initial Solution =
[[ 2 -6  5 -3 -2  4  7 -5  3 -7 -4 -8  6  8]
 [-1  4  6 -8  1 -6  8 -7 -4  5 -3 -5  7  3]
 [ 8  7 -4  1 -8 -7  5  4 -1  6  2 -6 -5 -2]
 [-7 -2  3  6  5 -1 -6 -3  2 -8  1  7  8 -5]
 [ 6 -8 -1 -7 -4  8 -3  1 -6 -2  7  2  3  4]
 [-5  1 -2 -4  7  2  4  8  5 -3 -8  3 -1 -7]
 [ 4 -3 -8  5 -6  3 -1  2  8  1 -5 -4 -2  6]
 [-3  5  7  2  3 -5 -2 -6 -7  4  6  1 -4 -1]]

initial Cost = 54855.8366362

initial Violations = 5


Initial Parameters

maxR = 10
maxP = 100
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 794.022146077
w = 11994.8132767
bestT = 397.011073039

bestInfeasible = 43535.0888587
bestFeasible = 45232

time = 127.922796965



Best Infeasible Solution ->

[[-5  2 -8  5 -7 -6  3  6  7 -4 -3 -2  4  8]
 [-7 -1 -5  6  4 -3  7  8  5 -8 -4  1  3 -6]
 [-4 -6 -7  4  6  2 -1 -5  8  5  1 -8 -2  7]
 [ 3  8  6 -3 -2  5 -8 -7 -6  1  2  7 -1 -5]
```

```
 [ 1  7  2 -1 -8 -4  6  3 -2 -3 -7 -6  8  4]
 [ 8  3 -4 -2 -3  1 -5 -1  4 -7 -8  5  7  2]
 [ 2 -5  3  8  1 -8 -2  4 -1  6  5 -4 -6 -3]
 [-6 -4  1 -7  5  7  4 -2 -3  2  6  3 -5 -1]]

Cost = 43546.8254703
Violations = 1



Best Feasible Solution ->

[[ 7  3  4 -7 -6  2 -5 -8  6  8 -3 -2 -4  5]
 [-6 -7 -5  8  4 -1 -8  3  7 -3  6  1  5 -4]
 [-5 -1 -6  5  8  7 -4 -2  4  2  1 -7 -8  6]
 [-8 -5 -1  6 -2  8  3  7 -3 -6 -7  5  1  2]
 [ 3  4  2 -3 -7 -6  1  6 -8  7  8 -4 -2 -1]
 [ 2 -8  3 -4  1  5 -7 -5 -1  4 -2  8  7 -3]
 [-1  2 -8  1  5 -3  6 -4 -2 -5  4  3 -6  8]
 [ 4  6  7 -2 -3 -4  2  1  5 -1 -5 -6  3 -7]]

Cost = 45232
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 8
Distance Map =
[[   0  745  665  929  605  521  370  587]
 [ 745    0   80  337 1090  315  567  712]
 [ 665   80    0  380 1020  257  501  664]
 [ 929  337  380    0 1380  408  622  646]
 [ 605 1090 1020 1380    0 1010  957 1190]
 [ 521  315  257  408 1010    0  253  410]
 [ 370  567  501  622  957  253    0  250]
 [ 587  712  664  646 1190  410  250    0]]

initial Solution =
[[ 4  7  3  6 -8  5  2  8 -2 -7 -4 -5 -3 -6]
 [-8 -3 -6  5  6 -7 -1  4  1 -4  8  7 -5  3]
 [ 6  2 -1  4  5 -6 -8  7  8 -5 -7 -4  1 -2]
 [-1 -6  8 -3  7 -8  5 -2 -5  2  1  3  6 -7]
 [ 7  8 -7 -2 -3 -1 -4  6  4  3 -6  1  2 -8]
 [-3  4  2 -1 -2  3  7 -5 -7 -8  5  8 -4  1]
 [-5 -1  5  8 -4  2 -6 -3  6  1  3 -2 -8  4]
 [ 2 -5 -4 -7  1  4  3 -1 -3  6 -2 -6  7  5]]

initial Cost = 54401.9280582

initial Violations = 4


Initial Parameters

maxR = 10
maxP = 100
maxC = 100
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 794.339834354
w = 9858.86217266
bestT = 397.169917177

bestInfeasible = 42071.9971848
bestFeasible = 43888

time = 1165.49754691



Best Infeasible Solution ->

[[ 6  4  5 -7 -8 -6  2  3 -4 -2 -3  8  7 -5]
 [-3  7 -6 -5  4  8 -1 -7 -8  1  5 -4  6  3]
 [ 2  8  7 -8 -5  4 -6 -1 -7  5  1  6 -4 -2]
 [-5 -1  8 -6 -2 -3  7  5  1 -7 -8  2  3  6]
```

```
 [ 4  6 -1  2  3 -7 -8 -4 -6 -3 -2  7  8  1]
 [-1 -5  2  4 -7  1  3 -8  5  8  7 -3 -2 -4]
 [-8 -2 -3  1  6  5 -4  2  3  4 -6 -5 -1  8]
 [ 7 -3 -4  3  1 -2  5  6  2 -6  4 -1 -5 -7]]

Cost = 41942.2441335
Violations = 1



Best Feasible Solution ->

[[ 5  4  2 -4 -2 -3  8 -5 -6 -8  6 -7  3  7]
 [-3  6 -1  5  1  7 -5  3 -8 -7  4  8 -6 -4]
 [ 2 -7 -8 -6  7  1  6 -2  5  4  8 -4 -1 -5]
 [ 6 -1 -5  1  5 -6 -7 -8  7 -3 -2  3  8  2]
 [-1 -8  4 -2 -4  8  2  1 -3 -6 -7  6  7  3]
 [-4 -2 -7  3  8  4 -3  7  1  5 -1 -5  2 -8]
 [ 8  3  6 -8 -3 -2  4 -6 -4  2  5  1 -5 -1]
 [-7  5  3  7 -6 -5 -1  4  2  1 -3 -2 -4  6]]

Cost = 43888
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 8
Distance Map =
[[   0  745  665  929  605  521  370  587]
 [ 745    0   80  337 1090  315  567  712]
 [ 665   80    0  380 1020  257  501  664]
 [ 929  337  380    0 1380  408  622  646]
 [ 605 1090 1020 1380    0 1010  957 1190]
 [ 521  315  257  408 1010    0  253  410]
 [ 370  567  501  622  957  253    0  250]
 [ 587  712  664  646 1190  410  250    0]]

initial Solution =
[[-5  8  5 -4  2 -6  4  6 -3  7 -8 -2  3 -7]
 [ 3  6  8 -3 -1  4  5 -7 -6 -8  7  1 -4 -5]
 [-2 -7  6  2 -5  8 -6 -8  1  4  5  7 -1 -4]
 [-7  5  7  1 -6 -2 -1 -5 -8 -3  6  8  2  3]
 [ 1 -4 -1  8  3 -7 -2  4  7 -6 -3  6 -8  2]
 [ 8 -2 -3  7  4  1  3 -1  2  5 -4 -5 -7 -8]
 [ 4  3 -4 -6  8  5 -8  2 -5 -1 -2 -3  6  1]
 [-6 -1 -2 -5 -7 -3  7  3  4  2  1 -4  5  6]]

initial Cost = 56211.1918526

initial Violations = 8


Initial Parameters

maxR = 30
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.9999
sigma = 1.04
w = 4000



Final Parameters

T = 798.880727709
w = 8103.26606151
bestT = 399.440363854

bestInfeasible = 45929.271412
bestFeasible = 47583

time = 29.7313649654



Best Infeasible Solution ->

[[ 8  6 -7 -8  7 -5 -3  5 -2 -4  2  4 -6  3]
 [-3  7  5  3 -8 -7  6  4  1 -6 -1 -5  8 -4]
 [ 2  8 -4 -2 -6  4  1  6 -7  5  7 -8 -5 -1]
 [ 7  5  3 -6 -5 -3  8 -2  6  1 -8 -1 -7  2]
```

```
 [ 6 -4 -2  7  4  1 -7 -1 -8 -3 -6  2  3  8]
 [-5 -1 -8  4  3  8 -2 -3 -4  2  5 -7  1  7]
 [-4 -2  1 -5 -1  2  5 -8  3  8 -3  6  4 -6]
 [-1 -3  6  1  2 -6 -4  7  5 -7  4  3 -2 -5]]

Cost = 45934.3490339
Violations = 1



Best Feasible Solution ->

[[ 8 -4 -2 -6  4  5  6 -5  3 -7 -8  2  7 -3]
 [-7  6  1  7 -3 -6 -8  4  5  8 -5 -1  3 -4]
 [-5 -7 -8  5  2  8 -4  7 -1 -6  4  6 -2  1]
 [ 6  1  7 -8 -1 -7  3 -2  8  5 -3 -5 -6  2]
 [ 3  8  6 -3 -6 -1  7  1 -2 -4  2  4 -8 -7]
 [-4 -2 -5  1  5  2 -1  8 -7  3  7 -3  4 -8]
 [ 2  3 -4 -2 -8  4 -5 -3  6  1 -6  8 -1  5]
 [-1 -5  3  4  7 -3  2 -6 -4 -2  1 -7  5  6]]

Cost = 47583
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 12
Distance Map =
[[   0  745  665  929  605  521  370  587  467  670  700 1210]
 [ 745    0   80  337 1090  315  567  712  871  741 1420 1630]
 [ 665   80    0  380 1020  257  501  664  808  697 1340 1570]
 [ 929  337  380    0 1380  408  622  646  878  732 1520 1530]
 [ 605 1090 1020 1380    0 1010  957 1190 1060 1270  966 1720]
 [ 521  315  257  408 1010    0  253  410  557  451 1140 1320]
 [ 370  567  501  622  957  253    0  250  311  325  897 1090]
 [ 587  712  664  646 1190  410  250    0  260   86  939  916]
 [ 467  871  808  878 1060  557  311  260    0  328  679  794]
 [ 670  741  697  732 1270  451  325   86  328    0 1005  905]
 [ 700 1420 1340 1520  966 1140  897  939  679 1005    0  878]
 [1210 1630 1570 1530 1720 1320 1090  916  794  905  878    0]]

initial Solution =
[[ -6   3  -5   9  10  12 -11   4  -3  -9  11   8   2 -12  -4  -7  -8   6
    7 -10  -2   5]
 [  7 -10  -6   4 -12 -11  10   6  -7   3   5  12  -1  11  -5   8  -4  -3
    9  -8   1  -9]
 [-10  -1  -9   5   9   7  12   8   1  -2  -8  -6  -5  -4  10 -11  -7   2
    6   4 -12  11]
 [-11 -12  10  -2   6  -8   9  -1  -5  12 -10  -9   7   3   1  -6   2   5
    8  -3  11  -7]
 [-12   9   1  -3   7   6  -7  10   4  -8  -2 -11   3 -10   2  -9  -6  -4
   11  12   8  -1]
 [  1   8   2  12  -4  -5  -8  -2 -10 -11 -12   3   9  -7  -9   4   5  -1
   -3  11   7  10]
 [ -2 -11 -12   8  -5  -3   5  11   2 -10   9  10  -4   6  -8   1   3  12
   -1  -9  -6   4]
 [ -9  -6  11  -7 -11   4   6  -3 -12   5   3  -1 -10   9   7  -2   1  10
   -4   2  -5  12]
 [  8  -5   3  -1  -3 -10  -4  12 -11   1  -7   4  -6  -8   6   5 -12  11
   -2   7  10   2]
 [  3   2  -4  11  -1   9  -2  -5   6   7   4  -7   8   5  -3  12 -11  -8
  -12   1  -9  -6]
 [  4   7  -8 -10   8   2   1  -7   9   6  -1   5  12  -2 -12   3  10  -9
   -5  -6  -4  -3]
 [  5   4   7  -6   2  -1  -3  -9   8  -4   6  -2 -11   1  11 -10   9  -7
   10  -5   3  -8]]

initial Cost = 167053.83582159772

initial Violations = 7


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.998
sigma = 1.04
w = 40000
```

Final Parameters

T = 797.0453217892804
w = 26282.112967005618
bestT = 398.5226608946402

bestInfeasible = 136406.5972302448
bestFeasible = 142025.0

time = 181.23095774650574

Best Infeasible Solution ->

```
[[  6  12   9 -12 -11  -2   4   8  10  -5   2  -7  -8  -9   3  -6   5  11
  -10  -3  -4   7]
 [-10  -9  -8   3  -7   1  10   9 -12 -11  -1   5  -4   8   6  12  -3  -6
   -5   7  11   4]
 [ -4 -10  12  -2   4   8   9  -6  11 -12 -11  -9   5  10  -1  -5   2  -7
   -8   1   7   6]
 [  3   5 -11  -5  -3   7  -1  -7   9   8  -9 -10   2  -6  10  11  -8 -12
    6  12   1  -2]
 [  7  -4  -7   4   8  -9 -11  10   6   1 -10  -2  -3  11  12   3  -1   9
    2  -6 -12  -8]
 [ -1  -7 -10   8 -12 -11   7   3  -5  -9  -8  12  10   4  -2   1  11   2
   -4   5   9  -3]
 [ -5   6   5  -9   2  -4  -6   4   8 -10  12   1  11 -12 -11  -8  10   3
    9  -2  -3  -1]
 [-12 -11   2  -6  -5  -3  12  -1  -7  -4   6  11   1  -2   9   7   4 -10
    3  -9  10   5]
 [-11   2  -1   7  10   5  -3  -2  -4   6   4   3 -12   1  -8 -10  12  -5
   -7   8  -6  11]
 [  2   3   6 -11  -9  12  -2  -5  -1   7   5   4  -6  -3  -4   9  -7   8
    1  11  -8 -12]
 [  9   8   4  10   1   6   5 -12  -3   2   3  -8  -7  -5   7  -4  -6  -1
   12 -10  -2  -9]
 [  8  -1  -3   1   6 -10  -8  11   2   3  -7  -6   9   7  -5  -2  -9   4
  -11  -4   5  10]]
```

Cost = 136424.58222316115
Violations = 1

Best Feasible Solution ->

```
[[  6  11  10 -11 -12   2   4  -9   8  -3  -2   9   7  12  -5  -6   5 -10
   -8   3  -4  -7]
 [  8 -10  -9   3  -7  -1  -8  10 -11 -12   1   5  -4   9   6  11  -3  -6
   -5   7  12   4]
 [ -4   8  11  -2   4   9  10  -6  12   1 -12 -10   5  -8 -11  -5   2  -7
   -9  -1   7   6]
 [  3   5  12  -5  -3   7  -1  -7  10   9 -10  -8   2  -6   8 -12  -9 -11
    6  11   1  -2]
 [  7  -4  -7   4  -9 -10 -12   8   6  11  -8  -2  -3  10   1   3  -1  12
    2   9  -6 -11]
 [ -1  -7  -8   9 -11 -12   7   3  -5 -10  -9  11   8   4  -2   1  12   2
   -4  10   5  -3]
```

```
 [ -5   6   5 -10   2  -4  -6   4   9  -8  11 -12  -1 -11  12  -9   8   3
   10  -2  -3   1]
 [ -2  -3   6  12 -10  11   2  -5  -1   7   5   4  -6   3  -4  10  -7   9
    1 -12 -11  -9]
 [-11 -12   2  -6   5  -3  11   1  -7  -4   6  -1  12  -2 -10   7   4  -8
    3  -5  10   8]
 [-12   2  -1   7   8   5  -3  -2  -4   6   4   3 -11  -5   9  -8  11   1
   -7  -6  -9  12]
 [  9  -1  -3   1   6  -8  -9  12   2  -5  -7  -6  10   7   3  -2 -10   4
  -12  -4   8   5]
 [ 10   9  -4  -8   1   6   5 -11  -3   2   3   7  -9  -1  -7   4  -6  -5
   11   8  -2 -10]]

Cost = 142025.0
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 14
Distance Map =
[[   0  745  665  929  605  521  370  587  467  670  700 1210 2130 1890]
 [ 745    0   80  337 1090  315  567  712  871  741 1420 1630 2560 2430]
 [ 665   80    0  380 1020  257  501  664  808  697 1340 1570 2520 2370]
 [ 929  337  380    0 1380  408  622  646  878  732 1520 1530 2430 2360]
 [ 605 1090 1020 1380    0 1010  957 1190 1060 1270  966 1720 2590 2270]
 [ 521  315  257  408 1010    0  253  410  557  451 1140 1320 2260 2110]
 [ 370  567  501  622  957  253    0  250  311  325  897 1090 2040 1870]
 [ 587  712  664  646 1190  410  250    0  260   86  939  916 1850 1730]
 [ 467  871  808  878 1060  557  311  260    0  328  679  794 1740 1560]
 [ 670  741  697  732 1270  451  325   86  328    0 1005  905 1846 1731]
 [ 700 1420 1340 1520  966 1140  897  939  679 1005    0  878 1640 1300]
 [1210 1630 1570 1530 1720 1320 1090  916  794  905  878    0  947  832]
 [2130 2560 2520 2430 2590 2260 2040 1850 1740 1846 1640  947    0  458]
 [1890 2430 2370 2360 2270 2110 1870 1730 1560 1731 1300  832  458    0]]

initial Solution =
[[-13 -11  -9   5 -12  10  12   9  -5 -10  11   6   4 -14  -3  14  -6  -7
    2   7   3  -4   8  13  -8  -2]
 [  4   6  -4  12 -14  -3  11   3  10   5  -7  14   9 -11  -6  13   8 -10
   -1  -9  -8   7 -13  -5 -12   1]
 [ 12  -7   8 -11  -6   2   4  -2   9 -13   6  13 -14  -4   1  -9  -5 -12
   -8 -10  -1  11   5   7  10  14]
 [ -2 -12   2   6  -8  -5  -3 -10   7  11   8   5  -1   3  13  -7 -11  -6
   -9  14 -13   1 -14  10   9  12]
 [  7  -8  12  -1  11   4  14  -6   1  -2   9  -4 -10 -13  -9 -11   3   8
   -7   6 -12  10  -3   2 -14  13]
 [ 14  -2 -14  -4   3  12   8   5  13   7  -3  -1  -8  10   2 -12   1   4
  -11  -5  -9 -13  -7   9  11 -10]
 [ -5   3  10  14 -10   8  13  -8  -4  -6   2  11 -12   9  12   4 -14   1
    5  -1 -11  -2   6  -3 -13  -9]
 [-10   5  -3 -13   4  -7  -6   7 -11  12  -4  -9   6 -12  14  10  -2  -5
    3  13   2   9  -1 -14   1  11]
 [ 11 -14   1 -10 -13 -11  10  -1  -3  14  -5   8  -2  -7   5   3  12  13
    4   2   6  -8 -12  -6  -4   7]
 [  8  13  -7   9   7  -1  -9   4  -2   1 -14  12   5  -6  11  -8 -13   2
  -12   3  14  -5 -11  -4  -3   6]
 [ -9   1 -13   3  -5   9  -2  14   8  -4  -1  -7  13   2 -10   5   4 -14
    6  12   7  -3  10 -12  -6  -8]
 [ -3   4  -5  -2   1  -6  -1  13 -14  -8 -13 -10   7   8  -7   6  -9   3
   10 -11   5  14   9  11   2  -4]
 [  1 -10  11   8   9 -14  -7 -12  -6   3  12  -3 -11   5  -4  -2  10  -9
   14  -8   4   6   2  -1   7  -5]
 [ -6   9   6  -7   2  13  -5 -11  12  -9  10  -2   3   1  -8  -1   7  11
  -13  -4 -10 -12   4   8   5  -3]]

initial Cost = 327734.1535945766

initial Violations = 18


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
```

theta = 1.04
beta = 0.998
sigma = 1.04
w = 40000

Final Parameters

T = 1593.2937576498962
w = 29563.80272051781
bestT = 796.6468788249481

bestInfeasible = 250334.57616023536
bestFeasible = 260150.0

time = 657.981365442276

Best Infeasible Solution ->

```
[[-12 -13 -14   6   5  -2 -10  11  10   8  -9  -8   3   9  -4   2  -5 -11
    4   7  -3  12  13  14  -6  -7]
 [  5   3  -6   8  14   1  -8 -10  13   6 -14  -9  10   4  -3  -1  12  -5
  -11  -4   7   9  11  -7 -12 -13]
 [ -6  -2   8  11   6 -13 -11  14 -12 -14  13   7  -1  10   2  -8  -7  12
   -5   9   1   4 -10  -9   5  -4]
 [ 10 -11 -12 -14 -13 -10   6  -5  14  13   8  -6  -7  -2   1  12   9  -8
   -1   2  11  -3   7   5  -9   3]
 [ -2 -12 -10  12  -1  -9  -7   4   7   9  -6  11   8 -11  -8 -14   1   2
    3 -13  10  13  14  -4  -3   6]
 [  3   8   2  -1  -3  14  -4   7   9  -2   5   4 -12 -13 -14  10  13 -10
   -7  12  -9  11  -8 -11   1  -5]
 [ 11 -14   9 -13 -12   8   5  -6  -5  12 -11  -3   4  -8 -10  -9   3  13
    6  -1  -2  10  -4   2  14   1]
 [  9  -6  -3  -2  10  -7   2  13  11  -1  -4   1  -5   7   5   3 -10   4
   12  14 -12 -14   6 -13 -11  -9]
 [ -8 -10  -7  10  11   5 -14  12  -6  -5   1   2  13  -1 -11   7  -4  14
  -13  -3   6  -2 -12   3   4   8]
 [ -4   9   5  -9  -8   4   1   2  -1  11  12  13  -2  -3   7  -6   8   6
   14 -11  -5  -7   3 -12 -13 -14]
 [ -7   4  13  -3  -9 -12   3  -1  -8 -10   7  -5  14   5   9 -13 -14   1
    2  10  -4  -6  -2   6   8  12]
 [  1   5   4  -5   7  11  13  -9   3  -7 -10 -14   6  14 -13  -4  -2  -3
   -8  -6   8  -1   9  10   2 -11]
 [ 14   1 -11   7   4   3 -12  -8  -2  -4  -3 -10  -9   6  12  11  -6  -7
    9   5 -14  -5  -1   8  10   2]
 [-13   7   1   4  -2  -6   9  -3  -4   3   2  12 -11 -12   6   5  11  -9
  -10  -8  13   8  -5  -1  -7  10]]
```

Cost = 249921.4491764946
Violations = 3

Best Feasible Solution ->

```
[[ 11 -14   8  13  -6  -8 -12  10   5 -11  12  -2   4 -10  -9  -5   2   6
  -13  -7   3   9  -4  -3  14   7]
```

```
[  6 -10  -3  11  14 -11  -8  13 -12 -14 -13   1  -7   9   3  10  -1  12
  -5   8   7   4  -9  -6   5  -4]
[ -5  -6   2  10   7 -10  14  -9  13   6 -14  -8   9   4  -2  -7 -12   5
  12  -4  -1   8  11   1 -11 -13]
[  8 -11 -12 -14   9  -7 -13  -5   6  13  10  -6  -1  -3   7  12  -9 -10
  14   3  11  -2   1   5  -8   2]
[  3  12  11 -12  -8 -14   7   4  -1   8   6  -9 -10 -11  10   1  -7  -3
   2 -13   9  13  14  -4  -2  -6]
[ -2   3  10  -7   1  -9 -11  14  -4  -3  -5   4 -12 -14 -13   9  13  -1
  -8  12   8  11 -10   2   7   5]
[-12 -13 -14   6  -3   4  -5  11   9 -10  -8  10   2   8  -4   3   5 -11
  -9   1  -2  12  13  14  -6  -1]
[ -4   9  -1  -9   5   1   2 -12  14  -5   7   3  13  -7  11 -14  10 -13
   6  -2  -6  -3  12 -11   4 -10]
[-10  -8  13   8  -4   6  10   3  -7  12  11   5  -3  -2   1  -6   4  14
   7 -11  -5  -1   2 -12 -13 -14]
[  9   2  -6  -3  13   3  -9  -1  11   7  -4  -7   5   1  -5  -2  -8   4
 -11  14 -12 -14   6 -13  12   8]
[ -1   4  -5  -2 -12   2   6  -7 -10   1  -9  13  14   5  -8 -13 -14   7
  10   9  -4  -6  -3   8   3  12]
[  7  -5   4   5  11 -13   1   8   2  -9  -1 -14   6  13  14  -4   3  -2
  -3  -6  10  -7  -8   9 -10 -11]
[ 14   7  -9  -1 -10  12   4  -2  -3  -4   2 -11  -8 -12   6  11  -6   8
   1   5 -14  -5  -7  10   9   3]
[-13   1   7   4  -2   5  -3  -6  -8   2   3  12 -11   6 -12   8  11  -9
  -4 -10  13  10  -5  -7  -1   9]]

Cost = 260150.0
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 16
Distance Map =
[[   0  745  665  929  605  521  370  587  467  670  700 1210 2130 1890
  1930 1592]
 [ 745    0   80  337 1090  315  567  712  871  741 1420 1630 2560 2430
  2440 2144]
 [ 665   80    0  380 1020  257  501  664  808  697 1340 1570 2520 2370
  2390 2082]
 [ 929  337  380    0 1380  408  622  646  878  732 1520 1530 2430 2360
  2360 2194]
 [ 605 1090 1020 1380    0 1010  957 1190 1060 1270  966 1720 2590 2270
  2330 1982]
 [ 521  315  257  408 1010    0  253  410  557  451 1140 1320 2260 2110
  2130 1829]
 [ 370  567  501  622  957  253    0  250  311  325  897 1090 2040 1870
  1890 1580]
 [ 587  712  664  646 1190  410  250    0  260   86  939  916 1850 1730
  1740 1453]
 [ 467  871  808  878 1060  557  311  260    0  328  679  794 1740 1560
  1590 1272]
 [ 670  741  697  732 1270  451  325   86  328    0 1005  905 1846 1731
  1784 1458]
 [ 700 1420 1340 1520  966 1140  897  939  679 1005    0  878 1640 1300
  1370 1016]
 [1210 1630 1570 1530 1720 1320 1090  916  794  905  878    0  947  832
   830  586]
 [2130 2560 2520 2430 2590 2260 2040 1850 1740 1846 1640  947    0  458
   347  654]
 [1890 2430 2370 2360 2270 2110 1870 1730 1560 1731 1300  832  458    0
   112  299]
 [1930 2440 2390 2360 2330 2130 1890 1740 1590 1784 1370  830  347  112
     0  358]
 [1592 2144 2082 2194 1982 1829 1580 1453 1272 1458 1016  586  654  299
   358    0]]

initial Solution =
[[ -7  10 -12  16   7  -2  -6 -11  -9 -15   4   3   2 -16  -8  13  15  -4
    9   8 -10  12   6  -3  -5 -14   5 -13  11  14]
 [  3   7 -16  15 -13   1  -3 -12  -4   9   5   4  -1  11 -10 -15 -11   8
   10  -6  -5  16  -7  13 -14  -8  -9  14   6  12]
 [ -2  13  -8 -11   9 -14   2  10  15 -10  16  -1  12  -6 -13 -12 -16 -15
    6  -9  -4  -5  11   1  -7   5  14   7   8   4]
 [ 10   5  13 -12  15 -10   7   8   2 -11  -1  -2  11  14  -5   9 -13   1
   12 -16   3 -15  -8   6  -9  -7  16  -6 -14  -3]
 [ -8  -4  11  10 -16  -7  14  15 -10  16  -2 -12  -6  12   4   8 -14 -13
    7  13   2   3  -9 -15   1  -3  -1 -11   9   6]
 [ 15  12  10 -14 -11 -15   1 -16 -12 -13  11   8   5   3  14 -10  -8   7
   -3   2  -9  -7  -1  -4  16   9  13   4  -2  -5]
 [  1  -2 -15  -9  -1   5  -4 -14  11  12  10 -16   9   8 -11  14 -10  -6
   -5  15  13   6   2  16   3   4  -8  -3 -12 -13]
 [  5 -16   3 -13 -10   9 -12  -4  16 -14  12  -6  13  -7   1  -5   6  -2
   14  -1  11  -9   4  10  15   2   7 -15  -3 -11]
 [ 12 -11 -14   7  -3  -8  16 -13   1  -2  13  14  -7  15 -16  -4 -12 -10
   -1   3   6   8   5  11   4  -6   2  10  -5 -15]
 [ -4  -1  -6  -5   8   4 -13  -3   5   3  -7 -11  15  13   2   6   7   9
   -2 -14   1  11  14  -8 -12 -16  12  -9 -15  16]
 [-14   9  -5   3   6 -16 -15   1  -7   4  -6  10  -4  -2   7  16   2  14
  -13 -12  -8 -10  -3  -9  13  12  15   5  -1   8]
```

```
 [ -9  -6   1   4 -14 -13   8   2   6  -7  -8   5  -3  -5 -15   3   9  16
   -4  11  15  -1  13  14  10 -11 -10 -16   7  -2]
 [ 16  -3  -4   8   2  12  10   9 -14   6  -9  15  -8 -10   3  -1   4   5
   11  -5  -7  14 -12  -2 -11 -15  -6   1 -16   7]
 [ 11 -15   9   6  12   3  -5   7  13   8  15  -9 -16  -4  -6  -7   5 -11
   -8  10  16 -13 -10 -12   2   1  -3  -2   4  -1]
 [ -6  14   7  -2  -4   6  11  -5  -3   1 -14 -13 -10  -9  12   2  -1   3
   16  -7 -12   4 -16   5  -8  13 -11   8  10   9]
 [-13   8   2  -1   5  11  -9   6  -8  -5  -3   7  14   1   9 -11   3 -12
  -15   4 -14  -2  15  -7  -6  10  -4  12  13 -10]]

initial Cost = 435424.0584315031

initial Violations = 25


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.998
sigma = 1.04
w = 40000



Final Parameters

T = 3192.0095926440476
w = 42078.509632211426
bestT = 1596.0047963220238

bestInfeasible = 359327.27700227284
bestFeasible = 404894.0

time = 5032.1633665561676



Best Infeasible Solution ->

[[ 10 -14 -13 -12   4 -16  -6  -8  15   9   3 -15   6  16   7  -2   8 -10
   12  11  13  -5 -11   2   5  -3  -4  -9  14  -7]
 [  5  -3   6   7   9  -6  10  -4  -8  -7  12 -10  -9  14 -11   1  13   8
  -14 -15 -16 -12 -13  -1   4  15  -5  11   3  16]
 [ -6   2  -5 -14 -16   9   8   7  -4  10  -1  12  -8 -13 -15  13  -9  11
    4  16 -10  15  -7 -12 -11   1  14   5  -2   6]
 [ 12   8   7 -11  -1 -10   9   2   3 -13 -14  -6  14   5  16  -7 -12  13
   -3  -8  -9  11   6  15  -2  -5   1 -16 -15  10]
 [ -2  10   3  -9  -7  -8  15   6  11 -12   7  14  13  -4 -13 -15 -16 -14
   16  12  -6   1   9   8  -1   4   2  -3 -10 -11]
 [  3  12  -2 -10  13   2   1  -5   7 -14 -16   4  -1 -11  10  16 -15 -12
  -13   9   5   8  -4  -7  15  -8  11  14  -9  -3]
 [  9 -15  -4  -2   5  12 -11  -3  -6   2  -5  -8  15 -12  -1   4  11  -9
   10  13  14 -10   3   6 -14 -16 -13   8  16   1]
 [ 16  -4   9 -15 -10   5  -3   1   2  15 -13   7   3  10  -9  14  -1  -2
  -11   4  12  -6 -14  -5  13   6 -16  -7  11 -12]
```

```
 [ -7  16  -8   5  -2  -3  -4  13  10  -1 -11 -16   2  15   8 -10   3   7
  -15  -6   4  14  -5  11 -12 -14  12   1   6 -13]
 [ -1  -5 -11   6   8   4  -2  11  -9  -3  15   2  12  -8  -6   9  14   1
   -7 -14   3   7 -16  13  16 -13 -15 -12   5  -4]
 [-14 -13  10   4  14 -15   7 -10  -5  16   9  13 -16   6   2 -12  -7  -3
    8  -1  15  -4   1  -9   3  12  -6  -2  -8   5]
 [ -4  -6  16   1  15  -7  13  14 -16   5  -2  -3 -10   7 -14  11   4   6
   -1  -5  -8   2 -15   3   9 -11  -9  10 -13   8]
 [ 15  11   1 -16  -6  14 -12  -9 -14   4   8 -11  -5   3   5  -3  -2  -4
    6  -7  -1  16   2 -10  -8  10   7 -15  12   9]
 [ 11   1 -15   3 -11 -13  16 -12  13   6   4  -5  -4  -2  12  -8 -10   5
    2  10  -7  -9   8 -16   7   9  -3  -6  -1  15]
 [-13   7  14   8 -12  11  -5  16  -1  -8 -10   1  -7  -9   3   5   6 -16
    9   2 -11  -3  12  -4  -6  -2  10  13   4 -14]
 [ -8  -9 -12  13   3   1 -14 -15  12 -11   6   9  11  -1  -4  -6   5  15
   -5  -3   2 -13  10  14 -10   7   8   4  -7  -2]]

Cost = 359327.27700227284
Violations = 2



Best Feasible Solution ->

[[ -8  -3  -5  10  -4   8  -6  -2   5 -11  13  -7  16  -9 -10   2   4   6
  -14   9  11  15 -12 -13  12  14 -15 -16   7   3]
 [ -6  12  -8   3  10  13  -7   1  14   9  -5 -11   6  15   8  -1  11 -12
  -15 -13   7   4  -3   5 -16  -4  16  -9 -14 -10]
 [-13   1  16  -2 -11  -6   4   5  10  -4   9  -8  14  -7 -14  15 -12 -16
   12 -15 -10  13   2   8  -9   7   6  11  -5  -1]
 [ -5  -9 -15  11   1   5  -3 -14 -16   3  -7  14   7  -6  16  -8  -1 -10
    6  12  13  -2   8 -12 -13   2 -11  10  15   9]
 [  4 -14   1   7  -9  -4  13  -3  -1  12   2 -15 -12  10   6 -13 -16 -11
    8  14  15  -7   9  -2  11  16 -10  -6   3  -8]
 [  2  11  12 -13 -15   3   1  -8  -7  13  16  10  -2   4  -5  -9  14  -1
   -4   8 -12 -16 -14   9   7  15  -3   5 -10 -11]
 [ 16   8  13  -5  -8  -9   2  11   6 -16   4   1  -4   3  12 -14 -13  14
  -11 -10  -2   5  10  15  -6  -3   9 -12  -1 -15]
 [  1  -7   2  15   7  -1 -11   6  -9 -15 -14   3  11  12  -2   4   9  13
   -5  -6 -16  10  -4  -3  14 -10 -12 -13  16   5]
 [-12   4 -14 -16   5   7 -10 -15   8  -2  -3  12  10   1 -13   6  -8  15
   16  -1  14  11  -5  -6   3 -11  -7   2  13  -4]
 [-11  16  11  -1  -2 -12   9  13  -3  14  12  -6  -9  -5   1 -16 -15   4
  -13   7   3  -8  -7 -14  15   8   5  -4   6   2]
 [ 10  -6 -10  -4   3 -14   8  -7 -13   1  15   2  -8  14 -15 -12  -2   5
    7  16  -1  -9  13 -16  -5   9   4  -3  12   6]
 [  9  -2  -6  14  16  10 -15 -16  15  -5 -10  -9   5  -8  -7  11   3   2
   -3  -4   6 -14   1   4  -1 -13   8   7 -11  13]
 [  3  15  -7   6  14  -2  -5 -10  11  -6  -1  16 -15 -16   9   5   7  -8
   10   2  -4  -3 -11   1   4  12 -14   8  -9 -12]
 [-15   5   9 -12 -13  11  16   4  -2 -10   8  -4  -3 -11   3   7  -6  -7
    1  -5  -9  12   6  10  -8  -1  13  15   2 -16]
 [ 14 -13   4  -8   6 -16  12   9 -12   8 -11   5  13  -2  11  -3  10  -9
    2   3  -5  -1  16  -7 -10  -6   1 -14  -4   7]
 [ -7 -10  -3   9 -12  15 -14  12   4   7  -6 -13  -1  13  -4  10   5   3
   -9 -11   8   6 -15  11   2  -5  -2   1  -8  14]]

Cost = 404894.0
Violations = 0
```

```
Summary
Hardcoded Solution = False
n = 16
Distance Map =
[[   0  745  665  929  605  521  370  587  467  670  700 1210 2130 1890
  1930 1592]
 [ 745    0   80  337 1090  315  567  712  871  741 1420 1630 2560 2430
  2440 2144]
 [ 665   80    0  380 1020  257  501  664  808  697 1340 1570 2520 2370
  2390 2082]
 [ 929  337  380    0 1380  408  622  646  878  732 1520 1530 2430 2360
  2360 2194]
 [ 605 1090 1020 1380    0 1010  957 1190 1060 1270  966 1720 2590 2270
  2330 1982]
 [ 521  315  257  408 1010    0  253  410  557  451 1140 1320 2260 2110
  2130 1829]
 [ 370  567  501  622  957  253    0  250  311  325  897 1090 2040 1870
  1890 1580]
 [ 587  712  664  646 1190  410  250    0  260   86  939  916 1850 1730
  1740 1453]
 [ 467  871  808  878 1060  557  311  260    0  328  679  794 1740 1560
  1590 1272]
 [ 670  741  697  732 1270  451  325   86  328    0 1005  905 1846 1731
  1784 1458]
 [ 700 1420 1340 1520  966 1140  897  939  679 1005    0  878 1640 1300
  1370 1016]
 [1210 1630 1570 1530 1720 1320 1090  916  794  905  878    0  947  832
   830  586]
 [2130 2560 2520 2430 2590 2260 2040 1850 1740 1846 1640  947    0  458
   347  654]
 [1890 2430 2370 2360 2270 2110 1870 1730 1560 1731 1300  832  458    0
   112  299]
 [1930 2440 2390 2360 2330 2130 1890 1740 1590 1784 1370  830  347  112
     0  358]
 [1592 2144 2082 2194 1982 1829 1580 1453 1272 1458 1016  586  654  299
   358    0]]

initial Solution =
[[  4 -10 -15   9  11  15   3 -13  -8  -2 -11  -7  10  -5   2   5  14  -9
    6 -14  16  -4   8  -3 -12   7  -6 -16  13  12]
 [-14 -12  -8  15 -16  -3  -9   3  12   1 -10  -5   7  11  -1  -6  -4   8
    4  10 -15   9   6 -13 -11  14  -7  13  16   5]
 [ 12 -16   7 -11  14   2  -1  -2  -6  16   6  -8   5   9  10  15  11 -14
  -12  -9  -4 -13  -7   1   8 -10  -5   4 -15  13]
 [ -1   8  -5   7  12   9 -12 -11  14  10 -15  16  13  15   6 -10   2 -13
   -2 -16   3   1 -14   5  -6  -8  -9  -3  -7  11]
 [ -7  -6   4 -12 -13  -8   6  16 -15  -9  14   2  -3   1 -16  -1  15 -10
   13  11   9  12  10  -4 -14 -11   3   7   8  -2]
 [ 11   5 -11  16  -7  12  -5   8   3 -12  -3 -13  -9  -8  -4   2   7 -15
   -1  13  14 -10  -2   9   4 -16   1 -14  10  15]
 [  5 -11  -3  -4   6 -10  13  12   9 -13 -12   1  -2  16  11  14  -6 -16
   -8  15   8 -14   3  10 -15  -1   2  -5   4  -9]
 [ -9  -4   2 -10 -15   5  11  -6   1  14 -16   3  15   6 -12  16 -13  -2
    7  12  -7 -11  -1 -14  -3   4  13   9  -5  10]
 [  8 -13 -10  -1  10  -4   2 -15  -7   5  13 -12   6  -3  15  11 -16   1
   14   3  -5  -2 -11  -6  16  12   4  -8 -14   7]
 [ 13   1   9   8  -9   7  16  14 -11  -4   2 -15  -1 -14  -3   4  12   5
  -16  -2 -12   6  -5  -7 -13   3  15  11  -6  -8]
 [ -6   7   6   3  -1  13  -8   4  10 -15   1 -14 -16  -2  -7  -9  -3 -12
   15  -5 -13   8   9  16   2   5  14 -10  12  -4]
```

```
 [ -3   2  14   5  -4  -6   4  -7  -2   6   7   9 -14 -13   8  13 -10  11
    3  -8  10  -5  16 -15   1  -9 -16  15 -11  -1]
 [-10   9  16  14   5 -11  -7   1 -16   7  -9   6  -4  12 -14 -12   8   4
   -5  -6  11   3 -15   2  10  15  -8  -2  -1  -3]
 [  2  15 -12 -13  -3  16 -15 -10  -4  -8  -5  11  12  10  13  -7  -1   3
   -9   1  -6   7   4   8   5  -2 -11   6   9 -16]
 [-16 -14   1  -2   8  -1  14   9   5  11   4  10  -8  -4  -9  -3  -5   6
  -11  -7   2  16  13  12   7 -13 -10 -12   3  -6]
 [ 15   3 -13  -6   2 -14 -10  -5  13  -3   8  -4  11  -7   5  -8   9   7
   10   4  -1 -15 -12 -11  -9   6  12   1  -2  14]]

initial Cost = 455100.8921775051

initial Violations = 24


Initial Parameters

maxR = 10
maxP = 10
maxC = 10
T = 400
theta = 1.04
beta = 0.998
sigma = 1.04
w = 40000



Final Parameters

T = 1596.9627344502203
w = 55372.44804790844
bestT = 798.4813672251101

bestInfeasible = 361027.0540110249
bestFeasible = 372590.0

time = 2042.304660320282



Best Infeasible Solution ->

[[  2  12  -3   7 -13  -2  -4   9 -14 -15   3  -7  11   6  14  -8  -9   4
   16 -10  -6  15  13   8 -11  -5  10 -16 -12   5]
 [ -1   6 -13 -10  -9   1   3  13  -7  -8  -6   9  15   8  -5  14  16 -14
  -11   5 -16  12  11 -15 -12  10   7   4  -3  -4]
 [-16 -14   1   8  16  -7  -2   4  13   9  -1  15 -13 -15  -8   7   6  10
   -4 -11  -5  11  12  -6  -9 -12  14   5   2 -10]
 [  6  -9  -8  -6  11   5   1  -3  12  16   9  -5 -12 -14   7 -15 -13  -1
    3  14 -11  10  -7 -16 -10   8  13  -2  15   2]
 [ -9  -7   6  11 -12  -4 -15  14   8   7  -8   4  10 -16   2  12 -11  16
  -10  -2   3  13  15 -13 -14   1   9  -3  -6  -1]
 [ -4  -2  -5   4   8 -14 -12  -7  16  13   2 -11  -9  -1  12  10  -3  11
   14 -13   1   9 -16   3   7 -15  -8 -10   5  15]
 [-11   5   9  -1  15   3 -10   6   2  -5  16   1 -16  13  -4  -3  12 -15
  -12  -9  10  -8   4  11  -6  14  -2 -14 -13   8]
 [ 12  11   4  -3  -6  13 -16 -11  -5   2   5 -10 -14  -2   3   1  10 -12
    9 -15 -13   7  14  -1  16  -4   6  15  -9  -7]
```

```
 [  5   4  -7  15   2  16 -14  -1  10  -3  -4  -2   6 -12 -15 -16   1 -13
   -8   7  14  -6 -10  12   3  13  -5 -11   8  11]
 [-13 -15 -16   2 -14  11   7 -12  -9  12  15   8  -5 -11  13  -6  -8  -3
    5   1  -7  -4   9  14   4  -2  -1   6  16   3]
 [  7  -8  15  -5  -4 -10  13   8 -15 -14  12   6  -1  10  16 -13   5  -6
    2   3   4  -3  -2  -7   1 -16 -12   9  14  -9]
 [ -8  -1 -14  16   5 -15   6  10  -4 -10 -11  13   4   9  -6  -5  -7   8
    7 -16  15  -2  -3  -9   2   3  11 -13   1  14]
 [ 10 -16   2  14   1  -8 -11  -2  -3  -6 -14 -12   3  -7 -10  11   4   9
  -15   6   8  -5  -1   5  15  -9  -4  12   7  16]
 [-15   3  12 -13  10   6   9  -5   1  11  13 -16   8   4  -1  -2  15   2
   -6  -4  -9  16  -8 -10   5  -7  -3   7 -11 -12]
 [ 14  10 -11  -9  -7  12   5 -16  11   1 -10  -3  -2   3   9   4 -14   7
   13   8 -12  -1  -5   2 -13   6  16  -8  -4  -6]
 [  3  13  10 -12  -3  -9   8  15  -6  -4  -7  14   7   5 -11   9  -2  -5
   -1  12   2 -14   6   4  -8  11 -15   1 -10 -13]]

Cost = 361027.0540110249
Violations = 1


Best Feasible Solution ->

[[  2  12  -3   7 -13  -2  -4   9 -14 -15   3  -7  11   6  14  -8  -9   4
   16 -10  -6  15  13   8 -11  -5  10 -16 -12   5]
 [ -1   6 -13 -10  -9   1   3  13  -7  -8  -6   9  15   8  -5  14  16 -14
  -11   5 -16  12  11 -15 -12  10   7   4  -3  -4]
 [-16 -14   1   8  16  -7  -2   4  -5   9  -1  15 -13 -15  -8   7   6  10
   -4 -11  13  11  12  -6  -9 -12  14   5   2 -10]
 [  6  -9  -8  -6  11   5   1  -3  12  16   9  -5 -12 -14   7 -15 -13  -1
    3  14 -11  10  -7 -16 -10   8  13  -2  15   2]
 [ -9  -7   6  11 -12  -4 -15  14   3   7  -8   4  10 -16   2  12 -11  16
  -10  -2   8  13  15 -13 -14   1   9  -3  -6  -1]
 [ -4  -2  -5   4   8 -14 -12  -7  16  13   2 -11  -9  -1  12  10  -3  11
   14 -13   1   9 -16   3   7 -15  -8 -10   5  15]
 [-11   5   9  -1  15   3 -10   6   2  -5  16   1 -16  13  -4  -3  12 -15
  -12  -9  10  -8   4  11  -6  14  -2 -14 -13   8]
 [ 12  11   4  -3  -6  13 -16 -11 -13   2   5 -10  14  -2   3   1  10 -12
    9 -15  -5   7 -14  -1  16  -4   6  15  -9  -7]
 [  5   4  -7  15   2  16 -14  -1  10  -3  -4  -2   6 -12 -15 -16   1 -13
   -8   7  14  -6 -10  12   3  13  -5  11   8 -11]
 [-13 -15 -16   2 -14  11   7 -12  -9  12  15   8  -5 -11  13  -6  -8  -3
    5   1  -7  -4   9  14   4  -2  -1   6  16   3]
 [  7  -8  15  -5  -4 -10  13   8 -15 -14  12   6  -1  10  16 -13   5  -6
    2   3   4  -3  -2  -7   1 -16 -12  -9  14   9]
 [ -8  -1 -14  16   5 -15   6  10  -4 -10 -11  13   4   9  -6  -5  -7   8
    7 -16  15  -2  -3  -9   2   3  11 -13   1  14]
 [ 10 -16   2  14   1  -8 -11  -2   8  -6 -14 -12   3  -7 -10  11   4   9
  -15   6  -3  -5  -1   5  15  -9  -4  12   7  16]
 [-15   3  12 -13  10   6   9  -5   1  11  13 -16  -8   4  -1  -2  15   2
   -6  -4  -9  16   8 -10   5  -7  -3   7 -11 -12]
 [ 14  10 -11  -9  -7  12   5 -16  11   1 -10  -3  -2   3   9   4 -14   7
   13   8 -12  -1  -5   2 -13   6  16  -8  -4  -6]
 [  3  13  10 -12  -3  -9   8  15  -6  -4  -7  14   7   5 -11   9  -2  -5
   -1  12   2 -14   6   4  -8  11 -15   1 -10 -13]]

Cost = 372590.0
Violations = 0
```