

Memetic Algorithm With Population Management (MA|PM) For Multi-Objective Network Design

***R. O. Oladele**

Department of Computer Science
University of Ilorin
P.M.B. 1515, Ilorin, NIGERIA
roladele@yahoo.com

O. T. Oladele

Department of Computer Science
Federal University of Technology
P. M. B. 704, Akure, NIGERIA.
oladelemrs@yahoo.com

**Corresponding author*

ABSTRACT

A Memetic Algorithm with Population Management (MA|PM) is employed to solve Multi-objective Network Design Problem. The algorithm was tested with three randomly generated networks of varying sizes. Results obtained were compared with the results obtained when MA was used. Overall, it was observed that MA|PM outperformed MA in terms of efficiency (computation time) for the three test problems. In addition, the results' quality of MA|PM is superior to that of MA for 10-node network problem while it is inferior to that of MA for 36-node network problem. The results' qualities of MA|PM and MA rank the same for 21-node network problem. The implication of these results is that MA|PM is always more efficient than MA regardless of problem size. However, the impact of population management on the effectiveness of MA is inversely proportional to the size of the problem.

Keywords: Memetic Algorithm, Population Management, efficiency.

African Journal of Computing & ICT Reference Format:

R. O. Oladele & O. T. Oladele (2014): Memetic Algorithm With Population Management (MA|PM) For Multi-Objective Network Design. Afr J. of Comp & ICTs. Vol 7, No. 1. Pp 137-146 .

1. INTRODUCTION

Population diversity is an important issue that should be considered in the design of an effective MA [1, 2]. If the population diversity is ill-managed, the algorithm will converge prematurely to poor local optima. This is particularly true when a small population is used by the MA.

MA|PM was proposed in [2] as a variant of classical MA to mitigate some of its drawbacks such as premature or slow convergence and a lack of strategy to control the diversity of a small population of individuals. Population management controls the diversity of a small population of high-quality solutions. It uses an input function that determines whether an offspring solution is added to the population or not. In a typical MA|PM, the input function takes cognizance of the following two factors:

- (i) the quality of the solution, and
- (ii) the diversity of the population after adding a solution, i.e. the contribution that the solution makes to the diversity of the population; this is measured as the "distance" of the solution to the population.

A Memetic Algorithm with Population Management (MA|PM) is structured much like a standard MA, but differs in the use of population management. Its main distinguishing features are:

- (i) a small population (typically 10 to 30 solutions),
- (ii) population management to control the diversity of the population

In [3], Zarei et al proposed an Adaptive Memetic Algorithm with Population Management (AMAPM) for designing HIV multidrug therapies. The authors claimed that the optimal solution given by the AMAPM is very satisfactory. They also submitted that numerical comparisons show that the AMAPM outperforms three other standard methods for the class of problems considered. Pagacz et al [4] designed an MA with advanced crossover operators, local improvement methods, and two population management techniques for solving a generalized minimum vertex-biconnected network problem (GMVBCNP).

Their numerical experiments clearly indicate that better results are obtained when the two management techniques are used, but more computation time is required. Park and Ryu [5] developed a Dual-population GA (DPGA). DPGA is a type of multipopulation GA (MPGA) that uses an additional population as a reservoir of diversity. Their empirical results on various classes of problems reveal that DPGA quite often outperforms both the standard GA and other GAs with additional diversity preservation mechanisms. A self-adaptation mechanism based on the competition of preference characteristic in mating to control the diversity of population in GA is proposed in [6]. The authors observed that the adaptive method is comparable to traditional GA with the best parameter setting.

MAIPM has not been extensively investigated for real life problems. Aside the work in [3] and [4], there is scarcely any other paper in literature that applied it to problems of practical significance. This paper therefore considers application of MAIPM for solving network design problem. To allow for comparison between MAIPM and the basic MA, the MA in [7] was equally used.

2. PROBLEM FORMULATION

We consider a multi-objective network design problem of the form:

$$\text{Minimize } y = f(x) = (f_1(x), f_2(x)) \quad (1)$$

$$\text{Subject to: } LFLOW_{ij} \leq CAP_{ij} \quad (2)$$

$$R(x) \geq R_0 \quad (3)$$

Where:

$x = (x_1, x_2, \dots, x_n) \in X$ is the decision vector

$y = (f_1, f_2) \in F$ is the objective vector

$f_1(x)$ is the cost function of the configuration x

$f_2(x)$ is the average delay on all the links in the configuration x

$LFLOW_{ij}$ refers to the traffic flowing along link (i, j)

CAP_{ij} is the capacity of link (i, j)

$R(x)$ is the reliability of the configuration x

R_0 is the minimum acceptable reliability ($R_0 = 0.95$)

The reliability calculation is done via Monte Carlo simulations.

Other network design parameters used are the followings:

N denotes the total number of nodes in the network

D_{ij} denotes the physical distance between every pair of nodes i and j

C_{ij} represents the cost of the link between nodes i and j

C_i is the cost of network equipment at node i

P_{ij} is selection status of link (i, j) : $P_{ij} = 1$ if link (i, j) is selected, else $P_{ij} = 0$

L = maximum distance for which the signal is sustained without amplification (We fix $L = 15\text{km}$)

A = cost of each amplifier unit (#6.00)

Poisson process was used to model the traffic delay

The objective functions; network cost and average delay are approximated by the following formulation.

i Network Cost:

$$NetCost = NodeCost + LinkCost + AmpCost \quad (4)$$

Where;

$$NodeCost = \sum_i C_i \quad (5)$$

$$LinkCost = \sum_i \sum_j C_{ij} \quad (6)$$

$$AmpCost = \frac{\sum_i \sum_j D_{ij} \times A}{L} \quad (7)$$

ii Average Delay:

$$AvDelay = \frac{\sum_i \sum_j [DELAY_{ij} \times LFLOW_{ij}]}{\sum_i \sum_j LFLOW_{ij}} \quad (8)$$

$$DELAY_{ij} = \frac{1}{[CAP_{ij} - LFLOW_{ij}]} \quad (9)$$

$DELAY_{ij} = 0$ if there is no link between nodes i and j

$DELAY_{ij} = \infty$ if the network cannot handle the traffic load with the existing links' capacities and routing policy.

Monte Carlo Simulation is used to estimate network reliability. The network is simulated t times, given the design and the links' reliabilities. The method is outlined below.

initialize $i = 0, c = 0$

Step C0: while $i < t$ Repeat.

Step C1: Randomly generate network

(a): $i = i + 1$.

Step C2: Check to see if the network forms a spanning tree

(a): if YES, increment c by 1 and go to Step C0

(b): if NO, go to Step C0

Step C3: $R(x) = c / t$.

Breadth First Search (BFS) is used for routing. The metric used for this purpose is the length of the link. The following assumptions were made in the problem formulation.

The location of each network node is given

Each C_{ij} is fixed and known

Each link is bidirectional i.e. a path can be traversed in either direction. There is no redundant link in the network

3. ALGORITHM DESIGN AND IMPLEMENTATION DETAILS

3.1 MAIPM DESIGN

- 1: initialize population P
- 2: set population diversity parameter Δ
- 3: repeat
- 4: select: p_1 and p_2 from P
- 5: crossover: $p_1 \times p_2 \rightarrow o_1, o_2$
- 6: local search: on o_1 and o_2
- 7: for each offspring o do
- 8: if $d_p(o) \geq \Delta$ then
- 9: remove solution: $P \leftarrow P \setminus o$
- 10: add solution: $P \leftarrow P \cup o$
- 11: end if
- 12: end for
- 13: update diversity parameter Δ
- 14: until stopping criterion is met

3.2 IMPLEMENTATION DETAILS

This section presents relevant details concerning the implementation of MAIPM

3.2.1 Encoding Scheme

The chosen encoding scheme is such that every chromosome codes a possible network, which corresponds to an individual in a set of feasible solutions of the problem. This set of feasible solutions constitutes a population. The chromosome is represented by a constant length integer string representation. The chromosome consists of two parts, the first part contains details of NE's at the nodes and the second part consists of details of the links. For example, if there are H types of nodes, then $\log_2 H$ bits are required to encode a node. Therefore the first part of the chromosome consists of $N \cdot \log_2 H$ bits, where N is the number of nodes in the network. If a link exists between nodes 1 and 2 then the first bit position in the link part is set to 1. Hence the second part of the chromosome consists of $(N(N-1))/2$ bits.

3.2.2 Initial Population

The two algorithms start by creating an initial population. There are two ways of generating initial population namely heuristic process and random process. A random process of generating initial population is adopted.

3.2.3 Fitness Evaluation

Fitness of a chromosome is evaluated based on principle of Pareto ranking. Pareto-rank of each individual is equal to one more than the number of individuals dominating it. All non-dominated individuals are assigned rank one. Network cost and average delay are used to evaluate the rank of an individual chromosome using the principle of Pareto dominance. The fitness of an individual is given by:

$$Fitness = \frac{1}{Rank^2} \quad (10)$$

3.2.4 Selection

Roulette Wheel Selection Process is used. In roulette wheel, individuals are selected with a probability that is directly proportional to their fitness values i.e. an individual's selection corresponds to a portion of a roulette wheel. The probabilities of selecting a parent can be seen as spinning a roulette wheel with the size of the segment for each parent being proportional to its fitness. Obviously, those with the largest fitness (i.e. largest segment sizes) have more probability of being chosen. The fittest individual occupies the largest segment, whereas the least fit have correspondingly smaller segment within the roulette wheel. The circumference of the roulette wheel is the sum of all fitness values of the individuals. The proportional roulette wheel algorithm procedure is depicted by the algorithm below. Let $f_1, f_2, \dots, f_n$ be fitness values of individuals 1, 2, ...n. Then the selection probability, P_i , for an individual i , is given as

$$P_i = \frac{f_i}{\sum_{j=1}^n f_j} \quad (11)$$

The template of the roulette wheel selection procedure is shown below.

Procedure: Roulette Wheel Selection

```

while population size < pop_size do
    generate pop_size random number r
    calculate cumulative fitness, total fitness
    ( $P_i$ ) and sum of proportional fitness
    (sum)
    Spin the wheel pop_size times
    if sum < r then
        select the first chromosome
    else
        select the jth chromosome
    Endif
Endwhile
return chromosomes with fitness values
proportional to the size of the selected
wheel section

```

End Procedure

3.2.5 Crossover

3.2.6 Local Search

The local search technique used is the hill climbing search algorithm. It is essentially an iteration that continuously proceeds in the direction of increasing quality value. The algorithm is as shown below

```
While (termination condition is not satisfied) do
    New solution ← neighbours(Best solution);
    If new solution is better than actual solution then
        Best solution ← actual solution
    End if
End while
```

Distance measure: The proposed MAIPM uses a distance measure, d that determines for each pair of solutions their relative distance (or similarity) in the solution space. Then, by extension, the distance of a solution o to population P is expressed as

$$d_P = \min_{s \in P} d(o, s)$$

For problems of which the solutions are most naturally represented as a binary string, the Hamming distance seems to be the only distance measure to use. For problems represented as real valued vectors, Euclidean, Manhattan, or Chebychev distance metrics can be used. Hamming distance [2] is used in our implementation. The distance of a given solution s_k to the population can be calculated as follows:

$$d_P(s_k) = \min_{s_i \in P} d(s_k, s_i) \quad (13)$$

Input function and diversity parameter Δ

It is obvious that a solution that has a small distance to another solution already in the population, will not contribute much to the diversity of a population. Therefore, a solution is not added to the population if its distance to the population is below a certain threshold Δ . We call Δ the diversity parameter. It is initialized to 1, 3, and 5 for 10-node network, 21-node network and 36-node network respectively and then incremented by 1 at each iteration.

4. COMPUTATIONAL RESULTS AND DISCUSSION

This section presents the results obtained when MAIPM and MA were applied to solve three instances of multi-objective network design problem. Comparative analyses of MA and MAIPM results are as well discussed. To enhance easy referencing and comprehensive analysis, results are coupled accordingly. All computations were performed on a HP 630 NOTEBOOK PC with the following configuration: 2.13GHz Processor Speed, 3.0GB RAM and 64 BIT OS.

The two algorithms were implemented in Java. Each of the algorithms was run 3 times for each problem instance. The results in bold (objective vectors) are the best pareto-ranked vectors for each run (that is, part of the pareto optimal front). Other algorithm design parameters used are:

Population size (MA) - 100 (250 for 36-node network problem)

Population size (MAIPM) - 30

Mutation probability - 0.02 (Mutation was not used in MAIPM)

Two- point crossover was used

Number of Node Type - 4

Test Problems:

10-node network design

21-node network design

36-node network design.

The test data for each of these problems were randomly generated and they are presented in appendix 1. The units of Cost, Delay and CPU Time are Naira, Seconds and Seconds respectively.

4.1 MA Results versus MAIPM Results

Table 1: MA and MAIPM Results of Run 1 for 10-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	98	619.6	0.06	145	5	100	595.5	0.04	100
10	100	619.6	0.05	410	10	100	523.5	0.03	389
15	100	619.6	0.05	548	15	100	528.5	0.03	429
20	100	619.6	0.05	643	20	100	523.5	0.03	510

Table 2: MA and MAIPM Results of Run 2 for 10-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	98	619.6	0.06	145	5	100	535.6	0.04	109
10	100	619.6	0.05	410	10	100	528.8	0.03	384
15	100	619.6	0.05	548	15	100	528.8	0.04	421
20	100	619.6	0.05	643	20	100	528.8	0.03	511

Table 3: MA and MAIPM Results of Run 3 for 10-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	98	619.6	0.06	145	5	100	615.7	0.03	101
10	100	619.6	0.05	410	10	100	552.7	0.03	289
15	100	619.6	0.05	548	15	100	523.1	0.03	359
20	100	619.6	0.05	643	20	100	523.1	0.03	410

Table 4: MA and MAIPM Results of Run 1 for 21-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	97	1259	0.06	954	5	100	1098.9	0.06	410
10	100	1167.4	0.05	1892	10	100	1111.2	0.08	999
15	100	1167.4	0.04	4467	15	90	1111.2	0.09	1340
20	100	1158.6	0.04	5665	20	100	1111.2	0.08	2736

Table 5: MA and MAIPM Results of Run 2 for 21-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	97	1259	0.06	910	5	100	1169.0	0.04	410
10	100	1167.4	0.05	1989	10	100	1169.4	0.05	991
15	100	1167.4	0.04	4320	15	100	1122.8	0.07	1240
20	100	1167.4	0.03	5766	20	100	1122.8	0.06	2136

Table 6: MA and MAIPM Results of Run 3 for 21-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	97	1259	0.06	954	5	100	1196.6	0.04	410
10	100	1167.4	0.05	1892	10	100	1199.0	0.08	1003
15	100	1167.4	0.04	4467	15	100	1150.2	0.07	1440
20	100	1158.6	0.04	5665	20	100	1119.9	0.08	1936

Table 7: MA and MAIPM Results of Run 1 for 36-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	97	1259	0.06	3987	5	90	1259.0	0.06	1598
10	100	1167.4	0.05	5673	10	98	1267.1	0.05	2890
15	100	1167.4	0.05	9210	15	100	1247.4	0.05	4673
20	100	1167.4	0.05	14900	20	100	1247.5	0.05	5897

Table 8: MA and MAIPM Results of Run 2 for 36-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	97	1167.4	0.06	3598	5	97	1219.8	0.06	1543
10	100	1167.4	0.05	5890	10	100	1267.4	0.07	2673
15	100	1167.4	0.05	9673	15	100	1259.4	0.06	4062
20	100	1167.4	0.05	15897	20	100	1237.4	0.05	5537

Table 9: MA and MAIPM Results of Run 3 for 36-Node Network Design Problem

MA					MAIPM				
No of Gen	C/Ratio	Cost	Delay	Comp. Time	No of Gen	C/Ratio	Cost	Delay	Comp. Time
5	97	1259	0.06	3987	5	97	1350.4	0.06	1298
10	100	1167.4	0.05	5673	10	100	1263.4	0.07	2490
15	100	1167.4	0.05	9210	15	100	1220.4	0.07	4673
20	100	1167.4	0.05	14900	20	100	1220.4	0.05	5597

4.2 Comparative Analysis of MA Results and MAIPM Results

Results show that MAIPM is always more effective and more efficient than MA in the case of 10-node network design problem while MAIPM is as effective as MA and more efficient than MA in the case of 21-node network. In the case of 36-node network however, MAIPM is less effective than MA though more efficient than MA. In general, population management enhances the efficiency of MA for the three instances of the design problem. However, the impact of population management on MA's effectiveness is a function of the network's size.

In this paper, the population size is set as a parameter and remains constant throughout execution. A further research avenue can be creating a new algorithm that can vary population size periodically and then do a series of performance tests comparing static and periodic population management.

5. CONCLUSION

Population management resulted in increased efficiency of MA for the three test problems. However, population management's impact on MA effectiveness is inversely proportional to the size of the network. That is for 10-node network, it improves results' quality, for 21-node network; it neither improves nor degrades quality, for 36-node network; it degrades quality. Our results do not totally agree with the results in [4] because MAIPM results are better than MA results for 10-node network problem only and these better results were obtained using less computational efforts (computing time). The discrepancy may be due to the fact that mutation was not implemented in our own MAIPM. Also our own problem domain/nature is different from that in [4].

REFERENCES

- [1] P. Galinier and J.K. Hao. Hybrid evolutionary algorithms for graph coloring. *Journal of Combinatorial Optimization*, 3(4):379–397, 1999.
- [2] K. Sørensen and M. Sevaux. MA|PM: memetic algorithms with population management. *Computers and Operations Research*, 33(5):1214–1225, 2006
- [3] H. Zarei, A. V. Kamyad, and S. Effati. An Adaptive Memetic Algorithm with Dynamic Population Management for Designing HIV Multidrug Therapies. *World Academy of Science, Engineering and Technology* 79: 986-993, 2011
- [4] Anna Pagacz, Bin Hu, Günther R. Raidl. A Memetic Algorithm with Population Management for the Generalized Minimum Vertex-Biconnected Network Problem. In *Proceedings of International Conference on Intelligent Networking and Collaborative System*. 356-361, 2010
- [5] T. Park, K. R. Ryu (2010). A Dual-Population Genetic Algorithm for Adaptive Diversity Control. *IEEE Transactions on Evolutionary Computation* 14(6): 865-884, 2010.
- [6] C. Jassadapakorn and P. Chongstitvatana (2011). Self-Adaptation Mechanism to control the Diversity of the Population in Genetic Algorithm. *International Journal of Computer Science & Information Technology (IJCSIT)* Vol 3, No 4, pp. 111-127.
- [7] R. O. Oladele (2013). Comparative Study of Memetic Algorithm and Genetic Algorithm for Multi-objective Optimization of Network Design. *International Journal of Emerging Trends and Technology in Computer Science*, Vol.2, Issue 3, pp.435 -441.
- [8] K. Watcharasitthiwat, P. Wardkein (2009): Reliability Optimization of topology communication network design using an improved ant colony optimization”. *Computers and Electrical Engineering*, vol. 35, pp. 730 – 747. Elsevier.
- [9] C. S. Wang and C. T. Chang (2008): Integrated genetic algorithm and goal programming for network topology design problem with multiple objectives and multiple criteria,” *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 680 –690.
- [10] E. Szlachcic and J. Mlynek (2009): Efficiency analysis in communication networks topology design,” *Fourth International Conference on Dependability of Computer Systems*”. *DepCos-RELCOMEX '09.*, pp. 184 –191.
- [11] M. Tsuji, M. Munetomo, and K. Akama (2003): Metropolitan Area Network Design Using GA Based on Hierarchical Linkage Identification”. *LNCS*, vol. 2724, pp. 1616-1617.

APPENDIX – TEST DATA

10-NODE NETWORK

Node Details (NodeType, C_2) = { (01,42) ,(0, 78) ,(10,33) ,(00,53) ,(01,42) ,(00,13) ,(10,9) ,(11,23) ,(10,57) ,(10,25) }

Link Details (D_{ij} , C_{ij} , CAP_{ij} , $LFLOW_{ij}$) = { (28,47,60,46) ,(20,43,90,72) ,(28,12,54,28) ,(62,39,61,46) ,(42,23,24,9) ,(42,30,16,14) ,(36,3,44,16) ,(40,18,75,54) ,(10,36,29,8) ,(44,30,79,53) ,(44,45,54,35) ,(36,18,66,51) ,(32,13,78,25) ,(14,16,96,54) ,(16,13, 84,74) ,(21,28,76,17) ,(22,3,80,71) ,(3,39,55,54) ,(47,12,66,62) ,(26,11,89,56) ,(13,42,77,47) ,(46,22,45,39) ,(28,6,53,16) ,(5,38,89,57) ,(28,40,16, 9) ,(48,49,49,40) ,(18,34,37,9) ,(34,35,11,8) ,(11,41,39,31) ,(46,20,32,9) ,(11,3,50,35) ,(70,1,54,41) ,(18,6,8,65) ,(35,42,91,66) ,(14,33,10,26) ,(11,33,60,9) ,(43,16,79,49) ,(20,43,88,56) ,(16,13,96,68) ,(6,30,91,67) ,(34,49,16,7) ,(37,21,57,49) ,(20,12,79,62) ,(33,46,81,70) ,(48,25,8,7) }

21-NODE NETWORK

Node Details = { (01,42) ,(01,78) ,(10,33) ,(00,53) ,(01,42) ,(00,13) ,(10,9) ,(11,23) ,(10,57) ,(10,25) ,(01,53) ,(00,55) ,(01,11) ,(00,34) ,(10,33) ,(00,32) ,(01,51) ,(10,38) ,(10,15) ,(10,57) }

Link Details = { (29 20 87 74) , (4 15 75 43) , (13 47 50 35) , (41 16 69 52) , (32 25 72 54) , (43 42 89 63) , (31 1 75 21) , (29 38 76 70) , (1 33 39 11) , (16 37 59 56) , (30 44 70 42) , (6 47 89 77) , (10 12 81 18) , (30 6 66 37) ,(26 3 80 54) ,(17 45 98 33) ,(12 10 49 39) ,(32 9 61 31) ,(6 45 35 24) ,(14 42 67 24) ,(19 10 89 31) ,(6 31 73 28) ,(31 19 29 22) ,(12 21 80 3) ,(49 35 60 13) ,(15 30 95 39) ,(15 3 29 9) ,(4 2 82 64) ,(26 27 27 12) ,(24 42 57 2) ,(25 46 68 66) ,(23 18 47 14) ,(5 28 94 65) ,(30 18 44 26) ,(7 18 78 59) ,(20 44 80 38) ,(33 29 30 7) ,(18 10 99 5) ,(25 43 18 2) ,(30 30 94 13) ,(0 26 82 46) ,(22 0 87 52) ,(40 6 63 15) ,(10 41 46 3) ,(25 45 35 10) ,(15 22 35 56) ,(46 28 32 5) ,(13 8 31 47) ,(17 18 29 35) ,(5 24 89 70) ,(36 25 90 76) ,(32 20 94 75) ,(40 34 84 73) ,(7 29 94 53) ,(39 35 33 21) ,(37 42 57 4) ,(43 41 71 60) ,(20 28 85 45) ,(36 17 51 19) ,(22 19 83 48) ,(44 17 28 19) ,(36 37 40 19) ,(32 36 40 14) ,(4 12 88 78) ,(32 47 88 8) ,(48 19 27 7) ,(26 7 73 60) ,(28 13 32 1) ,(20 21 47 19) ,(41 28 84 54) ,(30 28 78 66) ,(20 38 92 67) ,(21 27 88 27) ,(37 21 63 56) ,(27 22 57 35) ,(3 48 39 38) ,(20 32 62 56) ,(17 33 74 60) ,(41 24 60 14) ,(11 4 93 44) ,(20 44 75 74) ,(49 30 73 52) ,(39 16 64 57) ,(12 40 62 54) ,(33 16 12 70) ,(43 20 83 48) ,(0 16 93 71) ,(23 29 40 8) ,(2 35 81 36) ,(11 38 78 62) ,(7 11 93 63) ,(0 33 94 74) ,(9 48 88 54) ,(9 46 86 69) ,(15 44 87 32) ,(12 18 51 43) ,(16 24 79 43) ,(28 8 68 7) ,(41 49 67 27) ,(29 24 78 60) ,(48 5 63 12) ,(18 22 23 1) ,(22 31 17 7) ,(14 45 58 4) ,(3 45 73 64) ,(16 28 89 71) ,(5 8 59 26) ,(16 49 65 50) ,(6 25 39 9) ,(48 35 76 73) ,(1 30 35 63) ,(19 29 82 34) ,(35 27 57 20) ,(43 10 73 70) ,(17 28 25 15) ,(44 30 14 3) ,(5 20 63 12) ,(19 40 46 59) ,(8 30 82 50) ,(7 9 5 54) ,(7 10 74 66) ,(30 14 18 55) ,(7 5 83 15) ,(30 33 69 64) ,(15 10 64 61) ,(33 11 14 2) ,(18 31 79 75) ,(2 38 66 0) ,(47 0 29 17) ,(20 29 52 48) ,(46 38 10 4) ,(5 45 75 40) ,(39 17 83 66) ,(18 3 94 4) ,(30 25 60 43) ,(31 32 84 71) ,(34 45 74 74) ,(5 19 68 42) ,(27 48 72 69) ,(13 6 45 33) ,(20 17 37 23) ,(41 26 97 30) ,(34 42 54 22) ,(5 42 63 47) ,(39 26 71 47) ,(18 28 20 15) ,(18 4 70 10) ,(16 12 87 32) ,(1 13 97 11) ,(27 39 71 62) ,(41 14 94 59) ,(1 33 63 57) ,(12 2 70 43) ,(37 4 77 51) ,(25 16 85 21) ,(8 17 40 25) ,(6 11 87 42) ,(48 47 97 21) ,(28 39 42 21) ,(18 21 19 15) ,(46 12 99 51) ,(1 27 99 76) ,(8 31 21 11) ,(6 13 91 36) ,(27 17 69 21) ,(18 16 80 38) ,(42 20 92 19) ,(38 33 66 33) ,(47 5 57 27) ,(39 3 45 19) ,(30 4 88 78) ,(39 14 60 30) ,(28 40 92 41) ,(2 48 63 22) ,(17 10 83 42) ,(30 17 71 50) ,(14 20 66 79) ,(10 27 83 65) ,(43 27 89 40) ,(5 2 97 78) ,(8 13 67 57) ,(42 21 68 47) ,(5 20 84 23) ,(41 28 61 51) ,(41 38 70 31) ,(8 48 80 59) ,(9 10 97 25) ,(40 38 86 55) ,(31 20 65 60) ,(39 39 41 40) ,(22 40 95 77) ,(44 16 72 56) ,(21 45 88 53) ,(29 2 74 37) ,(21 45 60 40) ,(4 37 37 7) ,(12 43 48 38) ,(3 43 63 48) ,(46 25 63 43) ,(4 31 53 51) ,(18 9 70 22) ,(47 9 76 70) ,(38 13 75 44) ,(20 49 83 52) ,(21 15 90 25) ,(9 45 66 49) ,(42 40 55 0) ,(1 37 80 27) ,(40 8 88 28) ,(9 27 73 38) }

36-NODE NETWORKS

Node Details = { (10 52) ,(10 69) ,(00 67) ,(01 35) ,(00 77) ,(10 62) ,(10 41) ,(00 62) ,(10 77) ,(01 62) ,(11 20) ,(01 69) ,(10 70) ,(10 25) ,(11 41) ,(01 31) ,(01 5) ,(01 70) ,(00 39) ,(10 6) ,(10 30) ,(01 40) ,(00 8) ,(01 52) ,(10 69) ,(00 65) ,(01 78) ,(01 77) ,(00 12) ,(10 18) ,(11 69) ,(01 70) ,(11 33) ,(00 15) ,(10 54) ,(01 57) }

Link Details = { (8 32 89 20) ,(43 48 83 9) ,(31 21 65 7) ,(48 37 36 6) ,(10 19 86 24) ,(30 21 15 15) ,(17 37 33 6) ,(1 41 58 26) ,(46 42 94 22) ,(10 32 40 13) ,(5 18 21 17) ,(12 33 80 29) ,(28 8 21 11) ,(4 40 31 16) ,(46 3 75 9) ,(43 40 30 2) ,(36 39 34 1) ,(7 44 14 5) ,(7 25 78 29) ,(26 22 98 6) ,(17 0 60 29) ,(20 26 19 15) ,(2 18 78 16) ,(1 2 29 21) ,(46 34 35 22) ,(47 37 20 5) ,(38 7 66 21) ,(29 0 81 4) ,(18 20 15 12) ,(28 37 93 5) ,(3 19 40 28) ,(0 35 26 22) ,(42 38 34 21) ,(36 13 86 5) ,(31 30 68 29) ,(25 28 69 21) ,(21 3 81 28) ,(13 11 48 19) ,(2 21 87 2) ,(20 13 71 1) ,(38 11 71 21) ,(12 3 12 4) ,(13 36 48 20) ,(19 5 34 16) ,(22 47 28 19) ,(35 2 50 3) ,(2 40 66 25) ,(40 38 87 20) ,(20 16 99 23) ,(31 21 91 8) ,(24 29 57 6) ,(16 37 34 10) ,(3 24 42 28) ,(28 33 75 13) ,(38 42 67 18) ,(41 27 99 3) ,(23 40 98 27) ,(32 18 32 22) ,(0 24 37 29) ,(28 24 45 9) ,(13 8 54 11) ,(14 13 86 13) ,(12 43 65 15) ,(24 4 77 26) ,(28 31 35 29) ,(5 4 92 0) ,(24 6 79 18) ,(0 15 23 26) ,(27 16 54 22) ,(32 14 5 2) ,(49 28 90 5) ,(37 7 79 19) ,(12 38 84 28) ,(19 35 58 8) ,(31 4 52 26) ,(44 15 92 25) ,(46 8 48 9) ,(21 12 99 4) ,(13 30 24 16) ,(19 4 3 3) ,(25 21 82 9) ,(24 10 49 2) ,(29 29 85 19) ,(39 15 47 16) ,(34 0 22 9) ,(29 42 52 1) ,(30 25 48 26) ,(2 29 40 19) ,(12 22 51 17) ,(5 28 47 12) ,(37 20 92 29) ,(40 26 96 12) ,(44 43 52 28) ,(21 12 32 23) ,(34 31 37 9) ,(33 20 47 21) ,(13 48 50 15) ,(0 12 50 23) ,(30 8 70 25) ,(1 30 67 21) ,(28 9 16 5) ,(5 29 61 15) ,(17 31 31 7) ,(46 31 45 26) ,(16 13 1 0) ,(36 40 48 21) ,(46 35 61 21) ,(12 43 44 18) ,(41 47 30 8) ,(24 45 29 27) }

,(45 2 28 11),(9 47 91 27),(26 2 20 20),(24 49 34 15),(8 44 39 14),(46 40 79 11),(18 43 66 3),(7 4 80 2),(32 2 57 22),(28 17 78 7),(45 36 44 14),(11 35 26 5),(12 26 46 26),(20 43 24 27),(15 48 87 5),(29 15 23 20),(39 31 88 2),(12 48 46 14),(17 6 78 19),(8 34 26 16),(16 31 61 25),(17 19 41 18),(3 19 52 16),(41 22 25 3),(17 45 83 13),(4 8 59 14),(32 21 76 26),(17 1 95 24),(9 40 20 6),(23 44 43 7),(35 18 13 2),(8 1 40 9),(29 36 92 2),(20 2 79 8),(47 12 39 21),(26 7 60 7),(22 5 62 27),(26 30 99 7),(15 26 82 5),(8 10 18 10),(26 27 74 14),(6 43 90 20),(14 5 40 7),(21 0 41 2),(21 43 64 27),(23 9 67 11),(42 22 62 10),(49 6 55 20),(34 20 37 24),(7 7 44 21),(26 5 78 29),(49 28 68 24),(47 1 94 27),(47 34 64 9),(5 22 19 14),(41 22 36 13),(13 23 51 15),(40 30 38 21),(22 37 51 5),(8 20 55 15),(29 25 99 2),(1 45 40 20),(46 26 75 28),(41 9 52 26),(16 19 63 2),(40 33 55 8),(3 7 57 7),(37 38 90 21),(4 0 40 25),(4 38 97 14),(35 38 12 7),(47 39 60 13),(21 18 14 10),(17 49 50 24),(17 23 27 16),(23 29 57 6),(0 46 13 11),(5 7 17 13),(7 39 82 25),(17 19 31 19),(21 4 86 18),(15 21 76 3),(49 25 51 10),(26 5 99 9),(35 31 59 26),(6 43 28 25),(13 29 20 2),(43 21 64 13),(24 9 70 13),(48 4 78 6),(31 33 60 27),(5 21 28 9),(26 43 89 14),(43 47 87 15),(30 25 30 19),(41 22 32 27),(19 37 23 14),(42 36 28 17),(5 3 81 27),(19 27 35 6),(7 25 74 2),(41 3 91 5),(37 28 29 29),(49 9 70 28),(9 8 84 15),(33 20 50 14),(8 0 86 2),(30 24 35 19),(16 28 34 19),(19 18 10 25),(2 47 99 4),(47 39 63 13),(42 15 29 24),(40 18 38 20),(13 27 37 24),(25 35 25 2),(41 16 30 22),(15 20 50 20),(31 41 47 24),(25 25 19 16),(23 40 50 25),(43 3 80 6),(37 23 66 19),(48 46 59 6),(30 43 50 6),(23 23 36 28),(31 21 31 20),(30 20 30 8),(22 30 90 2),(24 34 60 19),(9 20 19 16),(46 43 57 24),(10 39 65 5),(38 0 24 10),(34 8 90 26),(9 26 47 10),(7 46 51 3),(20 33 38 13),(49 16 42 16),(24 31 87 18),(24 7 38 5),(19 20 37 13),(34 18 71 27),(31 3 65 7),(31 13 38 27),(43 30 88 15),(13 2 89 22),(17 48 27 13),(23 36 27 10),(12 2 22 17),(40 45 90 14),(7 4 35 16),(37 39 46 20),(6 19 35 8),(49 13 56 14),(24 7 36 22),(32 22 80 20),(36 21 38 20),(30 15 20 3),(32 10 53 16),(21 42 14 11),(19 30 20 19),(15 34 35 21),(47 36 66 10),(15 6 58 16),(28 39 64 22),(37 2 58 11),(33 37 76 29),(12 46 40 7),(39 4 71 13),(9 18 20 19),(7 41 83 8),(12 21 20 15),(7 42 38 5),(25 49 68 26),(38 0 61 8),(12 36 28 21),(39 43 49 3),(16 23 17 23),(6 28 68 0),(36 5 7 1),(45 35 13 20),(46 13 88 9),(47 42 35 19),(47 21 39 10),(37 11 21 18),(8 5 44 0),(14 37 91 1),(31 22 72 22),(4 9 46 5),(0 13 17 19),(23 23 64 28),(32 17 82 18),(39 8 59 14),(17 25 35 1),(13 38 19 13),(43 1 82 12),(35 17 30 4),(10 47 30 21),(44 25 72 6),(3 38 36 27),(41 46 63 6),(15 23 63 3),(5 41 42 25),(5 29 40 6),(34 8 21 4),(7 13 15 1),(15 18 79 0),(3 1 94 1),(48 49 4 2),(7 44 97 23),(30 18 27 21),(34 21 50 5),(22 27 66 14),(32 8 35 27),(44 36 69 16),(37 44 34 22),(2 3 30 23),(37 43 68 9),(49 39 99 22),(39 33 61 29),(22 32 11 11),(49 17 19 19),(33 18 84 11),(35 5 13 18),(25 28 28 22),(5 34 81 28),(43 15 45 15),(41 7 33 26),(17 15 59 1),(16 40 75 1),(29 15 36 6),(34 14 82 3),(22 10 30 15),(0 15 31 19),(13 12 40 19),(13 13 48 28),(3 41 68 13),(27 7 27 11),(34 49 54 0),(12 49 40 2),(47 43 10 8),(43 0 60 26),(9 19 40 21),(18 15 35 19),(14 46 97 24),(6 26 79 8),(35 47 44 6),(47 2 94 7),(38 36 35 25),(42 14 33 0),(3 35 81 28),(26 40 47 12),(10 5 87 25),(24 30 3 2),(43 30 87 12),(29 7 62 10),(1 32 40 26),(12 2 34 17),(41 16 40 24),(29 17 71 22),(21 0 75 26),(22 42 54 14),(40 16 49 4),(39 37 84 23),(19 39 47 0),(3 47 97 29),(49 31 70 29),(30 25 42 14),(13 13 27 11),(46 1 7 2),(48 32 28 2),(37 4 75 4),(19 24 47 23),(47 5 53 13),(49 38 18 16),(37 4 96 3),(5 22 50 26),(2 4 29 22),(1 0 20 4),(18 31 90 16),(48 19 57 12),(11 0 9 5),(48 18 99 13),(23 25 53 21),(10 17 25 24),(11 4 22 18),(36 21 15 7),(27 26 18 2),(20 41 35 26),(17 30 48 22),(26 19 68 9),(46 33 28 23),(29 25 29 24),(41 26 31 26),(44 29 71 2),(32 29 40 23),(15 20 47 11),(22 41 53 25),(0 18 64 5),(35 40 58 17),(14 43 19 9),(38 42 85 14),(34 12 50 5),(20 14 94 21),(49 47 56 29),(14 6 60 4),(39 41 50 27),(37 37 63 16),(40 42 44 7),(29 14 11 2),(7 34 86 7),(49 8 29 26),(4 29 97 14),(13 24 33 13),(0 1 61 14),(21 17 25 2),(41 31 78 1),(34 41 15 14),(48 26 29 24),(9 20 50 24),(16 4 23 17),(43 23 66 5),(20 39 63 24),(8 16 98 13),(10 28 84 10),(40 26 75 4),(24 45 47 27),(5 32 66 26),(33 30 17 7),(13 37 6 1),(25 40 30 21),(45 3 39 4),(4 49 33 20),(38 1 22 10),(49 33 72 10),(44 7 75 15),(32 20 42 10),(8 31 76 27),(1 1 29 27),(12 48 63 7),(24 8 70 22),(23 16 60 12),(14 0 64 24),(38 1 29 27),(45 6 71 17),(49 5 60 29),(23 15 79 24),(23 42 29 8),(46 32 99 28),(6 44 78 16),(17 10 22 19),(23 49 81 26),(3 1 88 16),(7 43 20 8),(6 20 29 27),(1 35 45 2),(20 33 88 11),(38 40 56 28),(25 40 19 17),(30 2 51 26),(42 38 60 8),(35 42 77 8),(5 8 64 28),(10 31 10 3),(33 46 9 2),(49 22 40 6),(27 40 57 25),(31 36 80 13),(15 5 22 21),(13 37 61 25),(40 47 41 28),(6 5 29 25),(27 29 98 20),(46 6 27 7),(18 14 70 15),(41 15 43 23),(28 10 83 20),(5 3 96 11),(28 12 66 24),(40 21 55 22),(40 29 39 14),(26 13 14 10),(35 13 29 28),(31 24 38 20),(26 41 85 0),(22 7 37 5),(26 40 37 0),(34 18 80 26),(3 0 74 22),(1 26 48 25),(9 31 39 25),(6 9 20 14),(0 6 99 1),(8 34 40 10),(41 28 21 11),(14 14 30 14),(46 48 38 4),(33 25 46 26),(27 29 70 5),(49 35 57 17),(3 20 17 1),(32 14 30 13),(46 17 58 17),(6 11 57 9),(28 27 21 18),(35 18 25 16),(8 7 98 15),(17 48 82 0),(31 16 22 7),(45 24 35 23),(19 17 43 10),(43 22 14 9),(15 2 53 8),(35 49 86 1),(45 49 57 19),(14 10 40 7),(12 11 93 0),(44 39 30 26),(30 9 27 5),(41 5 59 23),(38 24 17 12),(49 7 29 9),(4 23 49 13),(14 39 19 9),(13 37 17 4),(7 7 41 1),(5 2 49 16),(30 12 87 12),(42 18 60 21),(16 6 94 24),(37 48 85 26),(2 13 38 7),(5 36 47 17),(38 32 61 10),(43 7 85 24),(1 2 90 26),(18 34 27 16),(2 24 20 16),(11 22 54 10),(27 38 54 18),(45 19 69 7),(17 15 34 26),(48 20 31 23),(15 11 38 17),(36 7 92 16),(10 3 38 13),(42 12 12 8),(18 27 40 11),(15 28 30 6),(24 16 79 29),(27 19 91 14),(25 41 86 18),(41 35 93 27),(1 14 79 22),(1 11 99 0),(11 45 64 0),(25 25 20 4),(16 25 58 2),(42 25 90 0),(12 6 30 29),(15 14 65 18),(41 33 38 29),(25 46 82 21),(40 10 39 28),(36 29 99 23),(23 40 44 16),(37 19 52 25),(25 46 35 3),(3 41 30 23),(38 32 33 28),(45 15 29 28),(38 12 88 1),(34 27 90 2),(2 17 26 24),(15 35 58 4),(48 48 83 16),(42 21 10 1),(15 26 95 27),(23 20 59 1),(6 44 42 15),(22 40 19 3),(9 26 64 0),(34 47 61 18),(16 32 29 29),(16 0 38 18),(29 45 83 10),(5 26 39 0),(35 42 57 3),(27 8 70 16),(23 20 39 9),(17 38 31 11),(1 0 81 14),(12 46 95 24),(9 1 20 16),(32 24 47 19),(2 40 60 10),(17 28 19 11),(48 22 7 5),(19 14 11 5),(11 0 30 28),(46 1 38 28),(6 36 36 23),(18 8 78 22),(21 39 20 4),(25 16 37 20),(12 17 72 8),(12 43 94 26),(44 42 35 26),(46 46 59 6),(6 18 59 17),(41 17 25 25),(6 35 40 25),(0 22 57 25),(38 8 88 3),(27 0 29 2),(12 44 36 10),(2 21 89 0),(3 6 28 7),(39 42 30 29) }

Authors' Brief



Dr. R. O. Oladele is a Faculty at the Department of Computer Science, University of Ilorin, Nigeria. He obtained a Bachelor of Science Degree in Mathematics at the University of Ilorin, Ilorin, Nigeria in 1992, a Master of Science Degree in Mathematics at the University of Ilorin, Ilorin in 1998 and a PhD

Degree in Computer Science from the University of Ilorin, Ilorin, Nigeria in 2013. His research interests are Algorithms, Combinatorics, Optimization, Computational Complexity Theory, and Software Engineering.



Mrs. O. T. Oladele is a Program Analyst in the service of Ado-Ekiti Local Government, Ado Ekiti, Nigeria. She obtained a Bachelor of Science Degree in Computer Science at the University of Ilorin, Ilorin, Nigeria in 2000; she is currently working towards a Master Degree at Federal

University of Technology, Akure. Her research interests are Computer Networks, Traffic Engineering and Performance Evaluation of Computing Systems.
