ENVIRONMENTAL ORIENTED OPTIMIZATION MODELS FOR VEHICULAR ROUTING PROBLEMS: GREEN COMPUTING APPROACH

A THESIS

Submitted by

M.SHANMUGAM

in partial fulfilment for the award of the degree

of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF ENGINEERING AND TECHNOLOGY

PONDICHERRY UNIVERSITY

PUDUCHERRY 605 014

INDIA

DECEMBER 2015

ENVIRONMENTAL ORIENTED OPTIMIZATION MODELS FOR VEHICULAR ROUTING PROBLEMS: GREEN COMPUTING APPROACH

A THESIS

Submitted by

M.SHANMUGAM

in partial fulfilment for the award of the degree

of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF ENGINEERING AND TECHNOLOGY

PONDICHERRY UNIVERSITY

PUDUCHERRY 605 014

INDIA

DECEMBER 2015

PONDICHERRY UNIVERSITY
PUDUCHERRY 605 014

SCHOOL OF ENGINEERING AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

Certified that this thesis titled **"ENVIRONMENTAL ORIENTED OPTIMIZATION MODELS FOR VEHICULAR ROUTING PROBLEMS: GREEN COMPUTING APPROACH"** is the bonafide work of **M. SHANMUGAM** who carried out the research under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion of this or any other candidate.

Place : Puducherry
Date :

Dr. M.S. SALEEM BASHA
RESEARCH SUPERVISOR,
DEPARTMENT OF COMPUTER SCIENCE,
PONDICHERRY UNIVERSITY.

ABSTRACT

The Vehicle Routing Problem is one of the most studied combinatorial optimization problems because of its practical relevance and complexity. Though, there are several techniques have been proposed to solve the Vehicle Routing Problems and its variants effectively, each technique has its own tradeoff values in terms of the performance factors. Apart from the traditional performance criteria such as optimal route with respect to the distance, time and scalability, it is very difficult to find the techniques that considered environment related performance factors such as air pollution, sound pollution, etc,. From this perspective, the work presented in this thesis proposed an intelligent routing strategy for Vehicle Routing Problem based on air pollution intensity values between the cities. The proposed strategy uses an enhanced model of Genetic Algorithm to find the optimal tour paths among the cities under two different scenarios: distance based optimized tour path estimation and air-pollution intensity based optimized tour path estimation. For distance based optimization approach, experiments were performed on the standard benchmark Travelling Salesman Problem instances obtained from Travelling Salesman Problem Library. For air pollution based optimization approach, pollution specific data set has been generated and accompanied with benchmark Travelling Salesman Problem instances of Travelling Salesman Problem Library A set of fine grained result analyses demonstrated that the proposed model of routing strategies performed comparatively better with respect to the existing relevant approaches.

By considering these environmental problem as the base, three distinct models are developed as a set of assistive modules for Genetic Algorithms, which are aimed at improving the overall efficiency of the typical Genetic Algorithms, particularly for optimization problems. The capability of the proposed environmental oriented optimization models for Vehicle Routing Problem is demonstrated at various levels, particularly at the population initialization stage, using a set of well-defined experiments.

The fitness of the proposed models are validated in terms of standard valuation parameters, which could discover the abilities of any combinatorial optimization algorithm in various scopes along with appropriate trade-offs. The list of assessment criteria used for investigations include convergence rate, error rate, average convergence and average error rate. The controlled experimental study analyzed the exalted performance of the proposed techniques at the anticipated hierarchies. The outcomes and promising results of the experiments proved the ability of the proposed model. In addition to that, it is also outlined that further research work can be carried out to promote the proposed environmental oriented optimization models for Vehicle Routing Problem to integrate with Vehicular Ad-hoc Network to provide efficient intelligent transportation system.

ACKNOWLEDGEMENTS

First of all, I thank the Almighty for blessing, wisdom and everything

At first, I would like to express my sincere gratitude to my supervisor, **Dr. M. S. Saleem Basha**, for being a great motivator, advisor and supporting me throughout my research with his patience and knowledge. One simply could not wish for a better or friendlier supervisor.

My special thanks to **Prof. R. Subramanian**, HOD & Dean, **Prof. P. Dhavachelvan,** Ph.D. Coordinator, Department of Computer Science, School of Engineering and Technology, Pondicherry University, Puducherry, **Prof. K. Porsezian**, Professor, Department of Physics, Pondicherry University, **Prof. R. Baskaran**, Professor, Department of Computer Science and Engineering, College of Engineering, Anna University, Chennai, **Dr. T. Vengattaraman** and **Dr. P. Sujatha,** Assistant Professor, Department of Computer Science, Pondicherry University for their great support in my research and making quality publications.

I also wish to thank my fellow research scholars and all the teaching & nonteaching members of the Department of Computer Science, Pondicherry University and all my dear friends for their cooperation and moral support throughout my research.

I would like to thank my Parents, **Mr. V. Munisamy** and **Mrs. M. Baby Saroja** and my lovable Sisters **Mrs. M. Vijayalakshi and Miss. M. Umamaheswari.**

[M. Shanmugam]

TABLE OF CONTENTS

LIST OF FIGURES & TABLES

FIGURE No.	**TITLE**	**PAGE No.**

LIST OF ABBREVIATIONS

EVRP	-	Environmental Vehicle Routing Problem
IGA	-	Improved Genetic Algorithm
MO-VRP	-	Multi-Objective Vehicle Routing Problems
CVRP	-	Capacitated Vehicle Routing Problem
VRPTW	-	Pick and Delivery with Time Window Problem
M-TSP	-	*m*- Traveling Salesman Problem
SPPTWCC	-	Shortest Path Problem with Time Window and Capacity Constraints
ESPPTWCC	-	Elementary Shortest Path Problem with Time Window and Capacity Constraints
DARP	-	Dial-a-Ride Problem
EA	-	Evolutionary Algorithm
HybPSO	-	Hybrid Particle Swarm Optimization Algorithm
B&C	-	Branch and Cut
ACO	-	Ant colony Optimization
ODV	-	Ordered Distance Vector
EV	-	Equi-Begin with Variable Diversity
NN	-	Nearest Neighbor
ER	-	Elitism Ratio
TCC	-	Total Cost Convergence
VANET	-	Vehicular Ad-Hoc Network
ITS		Intelligent Transportation System
TPC		Total Pollution Convergence

CHAPTER -1

INRTODUCTION

1.1 OVERVIEW

Transportation plays a crucial role in our daily life because the transportation expenses are the major shares of many leading companies. The scope of transportation, supply chain management and logistics management plays a vital role within the delivery of products and services. Among others, it permits for the suitable distribution between the providers, manufacture divisions, depositories, vendors, and final clients. Generally, functioning a fleet of vehicle is a base problem that arises both in the service industries, the scheduling of school buses, or the on-site maintenance activities; and in the goods industries. Transportation of raw materials between providers and manufacture divisions, the replacement of vehicles in transport companies, or the pickup and delivery of goods in the trade industry become more complicated. Transportation conjointly has a crucial footprint in the trade economy and on the environment. Consequently the intention is to reduce the transportation cost and these problems are transformed into real world problems.

The objective of these problems is to minimize the transportation cost and achieve efficient routing. The Vehicle Routing Problem (VRP) is a (NP-hard) combinatorial optimization and integer programming problem in the fields of transportation, distribution, and supply chain management. Hence due to the importance of these real world problems, researchers are paying attention towards the VRP and its variant problems.

The VRP are important and most studied combinatorial optimization problems in relevance of transport logistics and distribution systems. The main objective is that the fleet of vehicles having capacity has to serve the customers in order to minimize the total cost of all the vehicles. Various constraints are given to the vehicles to reach the feasible solution; the customer should be visited exactly once by the vehicle. The vehicle has to start from the depot and visit the customers one by one; however the capacity of the vehicle exceeds then the vehicle must return to the depot [Ren et al, 2010]. In general, optimization algorithms are used to find the optimal routes for VRPs effectively. Initially, the VRP was solved using dynamic programming method [Thierry et al, 2010] by focusing on time oriented route elimination, based on the demand transportation. Though several parameters have been proposed to ensure the optimal route for VRP [Yannis et al, 2013], selection of an optimal route based on the specific time of travel attracted the attention of many researchers.

Later, VRP was solved by using Particle Swarm Optimization (PSO), which of course yielded efficient results [Kennedy et al, 1995]. In [Yannis Marinakisa 2010], [Marinakis et al 2005] proposed a hybrid PSO algorithm (HybPSO) for VRP by modifying each phase of the PSO using different algorithms: The Multiple Phase Neighborhood Search–Greedy Randomized Adaptive Search Procedure (MPNS–GRASP) algorithm [Marinakis et al, 2005a] was applied to enhance the initialization of routes; as an enhancement, Expanded Neighborhood Search (ENS) algorithm [Marinakis et al, 2005b] was used to improve the quality of routes and the computational cost; Path Relinking (PR) strategy [Marinakis et al, 2009] was used to take care of the local or global minimum constraints appropriately.

One of the popular variants of VRP is Dynamic Vehicle Routing Problem (DVRP), in which new customers arrives dynamically when the vehicles have already started executing their routes, which consequently have to be re-planned at runtime in order to include these new customers [Ghiani et al, 2003]. In [Mostepha et al, 2012] authors proposed a set of dynamic approaches to solve DVRP using PSO and VNS paradigms and demonstrated that the PSO perform better than VNS in terms of complexity and scalability criteria. Heterogeneous Fleet Vehicle Routing Problem (HFVRP) is another variant of the VRP, where the vehicles do not necessarily have the same capacities and the cost. Subramanian et al. [Subramanian et al, 2012], [Eduardo et al, 2013] and [Eduardo et al, 2005] proposed a hybrid algorithm, to solve HFVRP, composed of Iterated Local Search (ILS) based heuristic and the Set Partitioning (SP) formulation. The experimental results demonstrated that the method performs better with small scale problem instances and the efficiency diminishes with the increase in instance size.

A new hybrid algorithm based on PSO is proposed to find the suitable mapping between the vehicle routing problems with stochastic demands [Yannis et al, 2013]; this method produces the initial population of routes is generated randomly and ranked based on their fitness value. Different types of PSO are applied to assess the performance of the VRPs in terms of velocity of vehicle and justified that the constriction PSO with local search strategies performs better than other PSO variants [Yannis et al, 2013]. [Cordeau et al, 2012] proposed a parallel algorithm to solve different types of VRP using the tabu search with iterated local search framework. Authors proved that the proposed method performs well for different types of VRP and it can also be applied to solve the VRP with time windows. But, the fact of resultant solution of one stage to be given as the input for the next stage, destroy the potential sequence created by the previous stages. In [Yiyo et al, 2012], an effective Variable Neighborhood Search (VNS) method was proposed to solve the Multi-Depot Vehicle Routing Problem with Loading Cost (MDVRPLC), which is a combination of two complex problems namely Multi-Depot Vehicle Routing Problem (MDVRP) and Vehicle Routing Problem with Loading Cost (VRPLC).

1.2 INTRODUCTION AND EXPANSION OF VRPS

In combinatorial optimization problems Travelling Salesman Problem (TSP) plays an important role. It is very simple and however challenging task to solve. These problems are considered as NP complete problems, states that it cannot resolve in polynomial time. The objective of TSP is that the salesman has to visit $'n'$ number of customers exactly once in order to minimize the distance. Let $G = \{C_n , A_n\}$ be a complete undirected graph, C represents the number of customers $C \in \{C_1, C_2, C_3, \dots C_n\}$ and A represents the Arcs $A \in \{(C_1, C_2), (C_1, C_3), \dots (C_1, C_n) \dots (C_2, C_3), (C_2, C_4), \dots (C_2, C_n) \dots (C_{n-1}, C_n,), \}$.

The objective function of the problem is generalized as follows:

$$Objective\ Function = min\{\textstyle\sum_{k=1}^{n} Distance(C_k , C_{k+1})\} , \quad k + 1 \equiv 1 \quad \rightarrow 1$$

However many significant variants of TSP are available and solved using different approaches; let’s see another variant of TSP in detail. Multiple travelling salesman problem (M-TSP) which is a variant of TSP and considered as a combinatorial optimization problem. The objective is that the multiple salesmen has to visit numbers of customers exactly once and return to the stating place; finally the distance should be minimized. As like before Let $G = \{C_n , A_n, \; SP_m\}$ be a complete undirected graph, C represents the number of customers $C \in \{C_1, C_2, C_3, \dots C_n\}$, A represents the Arcs $A \in \{(C_1, C_2), (C_1, C_3), \dots (C_1, C_n) \dots (C_2, C_3), (C_2, C_4), \dots (C_2, C_n) \dots (C_{n-1}, C_n,),\}$ and SP_m represents the salesman $SP \; \in SP_1, SP_2 \dots SP_m$. Each salesperson has a separate path in order to achieve the objective function all path cost should be added. The objective function of the problem is modified as follows:

$$Objective\ Function = min\left\{\sum_{i=1}^{m} \sum_{k=1}^{n} Distance({C_k}^{i} , {C_{k+1}}^{i})\right\}, \quad k+1 \equiv 1 \rightarrow 2$$

Now let’s see another important variant of TSP, namely Travelling Salesman Problem with Time Window (TSPTW) widely used for scheduling problems. In this the sales person has to visit all n number of customers exactly once and return to the same place in order to minimize the total distance. Additionally a time window has been set for each customer and the sales person has to visit the customer within the time window. Let $G = \{C_n , A_n, \; TW_n\}$ be a complete undirected graph, C represents the number of customers $C \in \{C_1, C_2, C_3, \dots C_n\}$, TW_nrepresents the time window of each customer $TW \in TW_1, TW_2 \dots TW_n$ and Arepresentsthe Arcs.

$$A \in \{(C_1, C_2), (C_1, C_3), \dots (C_1, C_n) \dots (C_2, C_3), (C_2, C_4), \dots (C_2, C_n) \dots (C_{n-1}, C_n,)\}$$

Here the salesperson has to visit the customers with in the predefined time window, the objective function of the problem is modified as follows:

$Objective\ Function = min\{\sum_{k=1}^{n} Distance(C_k{}^i, C_{k+1}{}^i), TW_{k+1} \leq Cur\_TW\},\quad k+ \equiv 1 \rightarrow 3$

As discussed above VPR is one of the most important problems in transportation logistic, initially it has been developed by G. B. Dantzig' and J. H. Ramser and named as "Truck Dispatching Problem". VRP is designed by the influence of travelling salesman problem and its variants. The objective of TSP is that the salesman has to visit $'n'$ numbers of customers exactly once in order to minimize the distance. Likewise the objective of VRP is that the more than two vehicles have to visit $'n'$ number of customers exactly once in order to minimize the distance. Let $G = (C, E)$ be a complete undirected graph, where $'C$' is the vertex set represents the customers, and E is an edge set. They are expressed as $C = \{c_0, c_1, . . ., c_n\}$, and $E = \{(c_i, c_j)|c_j, c_j \in C, i \neq j\}$. Let $C' = C\backslash\{c_0\}$ be used as the set of $'n'$ customers, c_0 represent the depot. E a non-negative distance, travel time or distance matrix $DM\ (i, j)$ between customers c_i and c_j . Furthermore, a set V of m homogeneous vehicles originate from a single depot $V = \{v_0, v_1, . . ., v_m\}$, where $'\mathrm{m}'$ represents the number of vehicles $m = \lceil \sum_{i=0}^{n} Demand_i\ / \delta \rceil$.

Let $Indiv$ be a solution, $Indiv = \{R_1, R_2, ... R_m\}$ a partition of R_m representing the routes of the vehicles visited all the customers. A route R_k is characterized by the set of customers it contains, given by $R_k = \{c_o, c_1, ... c_{l+1}\}, 1 \leq i \leq l$ where $c_l \in C$ and $c_0 = c_{l+1}$ (denote the depot).

Where,

l Represents the number of customers in the route R_k

Evaluating the cost of all the routes,

$\forall[1, \leq k \leq m], R_k \leftarrow \sum_{j=1}^{l} DM(Indiv(j)\ , Indiv(j+1))$ →4

Evaluating the solution cost of the problem,

$$TC \leftarrow \sum_{i=1}^{m} R_i \rightarrow 5$$

In CVRP each customer is in need of some services or goods, each customer has its own demand and the capacity of the vehicles has been fixed. Initially the vehicle has to start from a single depot, hence the vehicle has to visit the customers one by one; certainly the capacity of the vehicle exceeds at some point then the vehicle has to return to the same depot. Yet again another vehicle has to start from the depot and visits the unvisited customers, this process continued until the last customer has been visited. Each vehicle has its own route, suppose for m number of vehicles m numbers of routes are generated. The summation of all the routes provides the total cost of the problem and also it should be minimized.

Correspondingly the VRPTW is an extension of CVRP; an additional constraint is that the each customer C_i has to be served in predefined time window$[e_i,\ l_i\ ]$ and the capacity of the vehicle is fixed. The vehicle has to start from a single depot and visits the customers individually, suppose the time window of the current customer is exceeds the vehicle time window then move towards the next customer. It is mandatory that the capacity of the vehicle should not exceeds the total capacity, then the vehicle has to return to the same depot and again another vehicle starts the same process until it visit all the customers. The overall cost (i.e.) the cost of all the routes affords the feasible solution.

In MDVRP, Multiple depots are available each vehicle has to start from each depot while constructing the routes. The capacity of the vehicle is fixed or may differ and each customer has its own demand. Initially the vehicle has to start from a depot, visit the customers after constructing the routes the vehicle has to return to the same depot. At this point, another vehicle has to start from different depot, visit the customers and return to the same depot; this process has been executed until all the customers visited exactly once by any one of the vehicles. The route of each vehicle provides the feasible solution, additionally the constraints should be satisfied.

As like in MDVRP, CVRP has multiple depots and each customer has its own predefined time window $[e_i,\ l_i]$ and the capacity of the vehicle is fixed or may differ. The vehicle has to start from a depot and visits the customers separately, assume the time window of the current customer is exceeds the vehicle time window then vehicle move towards the next customer. It is necessary that the capacity of the vehicle should not exceeds the total capacity, then the vehicle has to return to the same depot and again another vehicle starts from different depot the same process until visit all the customers. The routes of each vehicle provides the feasible solution, additionally the constraints should be satisfied.

1.3 GENETIC ALGORITHM

Genetic Algorithm is a heuristic search optimizing technique that is based on the natural selection of any biological evolution. Genetic Algorithms have been applied in almost all the fields of today's evolution which has a higher degree of complexity. The algorithm first randomly generates a set of population which it depicts as the set of solutions. Then it computes a solution to be the parent population at each step from which the other random solutions known as the children population are generated. Successive iteration of these steps will lead to an optimized solution in the search space. The genetic algorithm consists of two functions namely the Genetic representation and the fitness function. The genetic representation is usually of an array of bits represented as 0's and 1's. The fitness function is used to evaluate the genetic representation. These two functions are carried out to obtain an optimal solution from the variable set of solutions available in the search space. The genetic algorithms are usually represented in the form of a tree structure. With the help of the information available in the previous step, the algorithm chooses a solution from the search space. The search space plays a major role in the genetic technology which is a set or a group of solutions through which the process goes through to select the next solution. It may be any kind of data structure and is mostly used during the decision process. The researchers are most probably working based on the genetic algorithm because of its sturdy characteristics. The system does not crash if there are any slight variations in the inputs like the other evolutionary algorithms. It is also more advantageous in searching an optimal solution through a large number of solutions available in a state space and also in n- dimensional area it is found to be more efficient.

1.4 BASIC OF GENETIC ALGORITHMS

The genetic algorithm is based on the behavior of the chromosomes. The chromosomes consist of the genes which is associated with a fitness function. The nature of the chromosomes is that the mates are chosen at random. The second step is that the individuals mates with a high degree of competition to depict which produces a higher population and those mates that produces low population. The genes are then checked such that some of the parents produce an offspring that is better than either of the individuals. Thus each of the offspring generated are adoptable to the successive environments.

1.5 TYPES OF GENETIC ALGORITHMS

Genetic algorithm is basically a heuristic search technique which is used to generate optimized solutions for various search problems. This algorithm uses the search techniques which are derived from natural evolution such as inheritance, mutation, selection and crossover.

Inheritance:

Inheritance in GA is the one in which some behavior or characteristics of parent are present in the child. It’s a heredity process through which few or many behaviors of parent may be transferred to child either through sexual or asexual reproduction.

Selection:

The selection process in GA is the one in which a certain set of genome are selected for breeding the next generation. This can be implemented by evaluating the fitness function for each individual. These evaluated fitness function will provide some fitness values. These values are then normalized which is the process in which the fitness value of each individual is divided by the addition of all fitness values such that the resulting fitness value will be equal to 1. The current population is arranged in descending order of fitness values. The normalized fitness values are computed such that the value of last individual should be one. A number should be chosen between 0 and 1. An individual should be selected whose normalized value should be greater than the value chosen between 0 and 1. This process is repeated until enough individuals are selected for the next generation breeding. In some cases the fitness value cannot be computed. In this situation other selection process can be used such as stochastic universal sampling which does not uses multiple values for breeding rather than is chooses single value. The next is that tournament selection in which a best individual is selected repeatedly from a subset. Likewise many selection processes have been used for breeding the next generation with the reference to the current generation.

Cross over:

The cross over is said to be a genetic operator in GA which uses the process of recombination. The cross over is the most widely used genetic operator. As the name says cross over is what combining two or more parent solutions together to form a single child solution. In general, various numbers of methods are used to select the chromosomes. Such selection methods are fitness proportionate, Boltzmann, tournament, rank, steady state, truncation and local techniques. On choosing any one of these techniques, chromosomes needed to perform this crossover can be selected.

Various techniques are available under this category of crossover operation. Such common types of techniques used are:

1. *One-point crossover:*

   In this type of crossover technique, one point is selected on the parent’s organism strings.
2. *Two-point crossover:*

   Here, two points on the parent’s organism strings are selected and what are in between these points are exchanged which will render a child organism.

3. *Cut and splice:*

   Separate crossover points on parent organism string are chosen so that different varying in length based child string is formed.

4. *Uniform crossover :*

   Here, a fixed level of combining ratio is been selected in order to point on either of the parent's organism strings.

5. *Flat crossover :*

   Either of any one parent's organism string, one is chosen on a flat basis like a major usage of that one to form child's one.

6. *Three parent crossover :*

   A new offspring is chosen from the other three string set available. First parent's bits are checked with other two bit set so that the common bit is chosen based on the position accordingly.

Crossover bias tells about the formation of new sections from the ordering of contiguous sections of chromosomes will lead to a true solution which can be distracted by non-respectful cross over operator. Yet another new case in crossover operators in this ordered chromosomes. This Ordered chromosomes is what the given order should be preserved till the end of operation of crossover operator. Various common examples are available where the given order will be retained. Such are,

a) Partially matched cross over-PMX
b) Cycle crossover-CX
c) Order crossover operator-OX1
d) Order-based crossover- OX2
e) Position based crossover operator-POS
f) Voting recombination crossover operator-VR
g) Alternation –position Crossover operator-AP

h) Sequential constructive crossover operator-SCX

Mutation in Genetic Algorithm:

Mutation is another kind of genetic operator which is used to maintain a diversity of generation and take it to another next coming generation through chromosomes. From initial level to the completed level of the mutation process there occurs a formation new different solution. By using a random variable, for each bit in sequence so that generation of new series is possible. A mutation probability is followed according to the user defining way.

Various types are available in case of mutation based on the genetic type we consider. Such types are:

1. *Bit string Mutation:*

   Based on the random positions we use, the bit string ensure for mutation.

2. *Flip Bit:*

   Here, the genome type will be initially chosen and based on that genome value this corresponding bit is inversed and used for mutation.

3. *Boundary:*

   Either the integer or floating value can be used here. Thus the mutation operator will exchange the lower or upper bound value along with the genome value.

4. *Uniform:*

   A uniform random value will be used by mutation operator every time here.

5. *Non-Uniform:*

   The amount of mutation regarding the probability will be close to the value 0. As the generation number increases for any particular type or model which in turn increases with the probability value. This type will make use of both integer and floating type value.

6. *Gaussian:*

   Here, a unit Gaussian distributed value will be added to the genome that is been selected. If in case the new genome value falls outside the upper bound or lower bound of the user specified value then clipping occurs. Only integer and floating type values are used here.

Random Mutation:

In general, Mutation preserves a diverged search. So the mutation operator that is been chosen was the random mutation. Among the offspring values of the population that we consider for the probability uses this operator. Depending upon the genome we consider, the value lies between the intervals 0 to 1.

1.6 IMPLEMENTATION OF GENETIC ALGORITHM

Three operations that are used to implement the task are selection, cross over and mutation. Selection process is the one in which the individuals are directly related to their fitness value. If the fitness value is higher, then the chance of selecting the individual is higher. If it is used alone, it can fulfill the population with their copies. Next is cross over where the most typical solution is used. It takes place between the individual. The position of gene is chosen, where swapping operation may be carried out for possible set of solution. The point at which it is broken depends on the unusual selection of cross over point. This process represents the combinational operation of the individuals. The combination of both selection and cross over will generate the less quality solution. The last process of implementation is mutation. After the computation of selection and cross over, we may obtain a solution with same or different characteristics, done by the means of swapping operation (cross over) or by normally obtained one. In this process, the possible setting of an individual has to be changed. Using this mutation alone, an unusual walk to search space has been generated.

1.7 APPLICATIONS OF GENETIC ALGORITHM IN VANET

Vehicular routing problem is the complex functional problem. Many authors described it with the facts that they faced. Generally the problem is stated as, the group of vehicles that are engaged in a task or activity has to find efficient use of making any number of stops for taking in or delivering the goods or travelers. Picking or delivering operation has to be assigned to only one vehicle at a time. The overall routing problem minimizes the cost. It should have the parameters such as number of travelers, transportation cost, capacity of the vehicle and the demands made by the customer in order to solve a problem. All these values are considered to be either positive or equal to zero. It leads to a more complicated task because of the heterogeneous environment. This kind of solution has been computed by performing the heuristic search. The population size that is the collection of individuals is clearly defined among the bounds in order to discover the value of solution. In order to compute the routing problem efficiently and effectively, it combines with genetic algorithm and other local algorithms which are used for search. The locally defined search algorithm computes a task or evaluates the task by means of heuristic search process.

Nowadays rapid increase in vehicles has caused a higher degree of difficulty in transportation. Hence dynamic path finding is a key factor to find an optimal path in order to reduce the traffic congestion and to improve the efficiency of the vehicle movement. The path planning can be done by selecting the minimum cost and the minimum distance travelled by the vehicles. Here the available routes are selected in random and is stored as the search space. The single node is chosen to the parent node and the mutation occurs such that the other routes are generated. From the next possible set of solutions again the same process is carried out until the optimal path is generated. Hence the heuristic genetic algorithm is used to find the optimal path.

Another application of the genetic algorithm in VANET is the management of data stream. In order to normalize the traffic flow and to move across safely transmission of data is very much necessary. The necessary condition for the transfer of information is the selection of the vehicles to deliver the packets with a minimal of bandwidth jitter. Here the application of the genetic algorithm plays a crucial role. The vehicle selection is based on the fitness function. The fitness function consists of an effect coefficient, latency; the rate of messages received by the vehicles and the process how the vehicles transfer the messages. This algorithm is found to be more efficient based on the implementation strategy may be the mutation or the cross over.

1.8 GREEN COMPUTING

Energy is a phenomenal factor that is inevitable in the day-to-day life of people. It is being used in different forms, at different situations, to perform various operations. Vehicles are the key notion that is focused here, where energy is an indispensable component. The use of energy in diverse environments at high rate, exhaust their availability for future use. To overcome this, the concept of green computation is instigated. Green computing deals with the process of reducing the energy consumption of computer modules and provides methods for recycling the hardware components and electronic gadgets. This initially began as an intended program to advertise the use of energy in an efficient manner. This technique reduces the consumption of paper and the disposal of equipment's are minimized. The changes that take place using this may or may not affect the environment. Green computing not only involves energy conservation, but can also be applied to numerous other domains to attain a global eco- friendly environment. The use of non- hazardous waste in computational field may also be classified under green computing.

1.9 APPROACHES IN GREEN COMPUTING

Various approaches have been followed in green computing. Each approach is different from one another.

Virtualization: when two or more computers connected on to single hardware to perform certain operations is called virtualization. Administrator encompasses several systems on to a virtual system. It minimizes power and cooling consumption. Virtualization provides greater impact in Centralized repository. It has been implemented in green computing by the process of combining several servers and increasing the speed of processor.

Terminal servers: the end users of any computing system are enabled to use the central servers by means of an operating system. Here the clients are said to perform less amount of work through a request to server to perform the heavy duty. Thus it benefits the environment by consuming low energy and is cost efficient.

Power supply and management: among the total unit of power that is supplied to the computers only 75% is utilized by it, the remaining as dissipated in the form of heat. Later, systems where designed to use optimal power, by automatic on and off of monitors and hardware drives. The operating system controls the performance and use of power through direct control of hardware components.

Storage: hard disk drives are said to be static and hence the manipulation and saving of information in it does not require large value of power. The flash drives are also used in storage. The array of data storage is done to reduce the energy loss for data storage online.

Display: the power consumption of any display device depends on the components that are used. The use of LED for a Fluorescent bulb may lower the utility of power to large rate.

These approaches are being applied for providing a better use of energy. Green computing not only includes the utility bases of energy, but also helps in reducing the pollution caused in the environment through computing elements and guides the users to avoid contact to such contaminated surroundings.

1.10 APPLICATIONS IN VEHICULAR NETWORKS:

The application of green computing in vehicular networks allows any user to prohibit the contact to contaminated atmosphere and enable the minimum use of energy. The main theme is to route the vehicles through smog free shortest alleyway that is proliferated through the examination of the all available routes that lack of toxicity. Moreover, the research may also need self-analysis of routing vehicles to measure the pollution rate cause by it, as the use of vehicles may lead to emission of toxic gas that contaminates the surroundings. Intelligent transporting enables the scrutiny fuel usage and discharge of surplus components. The vehicles are fitted with sensor units that are used to sense the external atmosphere. Based on the values that are credited through the analysis of the surroundings, the contamination level of the environment is examined. Depending on the outlook of the gathered information the path that is to be traced by the user to reach the destined location is formulated. The lane which the user needs to follow is intimated through the application units that are embedded into the vehicle.

The routing of information to the vehicles is done by using the routing protocols for transferring data through the network channel. The data that is given to the user may contain information about the distance, type of route that is traversed, amount of contamination, obstacle control, diverse positions, parking lanes, alternative pathway. The alternative path is provided for the vehicles to make a choice to reach the destination if any unexpected obstacles are faced on the specified path. The implementation of green computing in vehicular networks promotes a secure and conscious travel. The avoidance of polluted routes helps the user to have a safe and healthy travel. The network devices are designed in such a way that the data are transmitted along the channel without any packet loss or delay, as it may affect the optimal route. The implementation of this concept onto the real world may be a bit tricky as the amount of contamination in any area does not remain constant all the time. Hence the path that is framed needs to be updated dynamically depending on the evaluated rate of pollution. At the same time the outlets from one's own vehicle should be controlled either by parsing it to reduce the level of effluence released into the atmosphere or find an alternative method to release it rather than discharging it into the environment directly, as it may increase the rate of the presence of toxic components in air. Since the main concept of this research is to concentrate on routing the vehicles in a shortest path that is free from the contaminations and the path that is traversed may be less polluted, hence this trend is said to be classified under the facet of green computing

1.11 MOTIVATIONS

Huge growth in transportation system and the development over a number of industries the pollution is increased tremendously. There are many modes of transportation such as cycle, bus, train, car, etc. the Vehicular Ad-Hoc network aims to provide intelligent transportation system. The vehicular communication is greatly affected by polluted path. The ITS (Intelligent Transportation System) avoids the polluted path, and it identifies the optimal path to vehicles. The vehicles choose the optimum path, so that they can avoid congestion and collision. Due to increase in fuel consumption of vehicles the CO_2 emission is more. Because of the increase in CO_2 emission health hazards occur. More vehicles choosing the same path causes air pollution due to excessive CO_2 emissions. The Intelligent Transportation System (ITS) of vehicles is achieved by equipping the devices such as On Board Unit, Application Unit, which is used for communication purposes. The application unit is used for navigation system. Onboard unit is a device for communicating with vehicles. Global warming is increased nowadays due to air pollution.

How to route vehicles by avoiding the polluted path/route

Vehicles emit CO_2 which is a major cause of air pollution. So to avoid the polluted path Intelligent Transportation System in the Vehicular Ad-Hoc Network is used to direct the best route for the vehicles. It directs the vehicles to identify the shortest alternative path to reach the destination and avoids congestion. Pollution in urban areas became more so toxic level increases in the air. There are several pollutants such as nitrous oxide, carbon dioxide, sulphur dioxide, nitrogen dioxide, nitric oxide, carbon monoxide. To avoid this pollution is not easy one but by avoiding the polluted routes which have more air pollution is possible. To predict the best routes and to avoid polluted route several devices such as sensors are equipped with vehicles. Using geo-cast routing protocol the locations are identified and send messages to the moving vehicles. Geo-cast routing protocol deals with vehicle fuel consumption and emission of CO_2. This routing protocol alerts the drivers who are in the highway city environment to avoid collision, congestion, etc. so it deals with less fuel consumption and CO_2 emissions. Vehicle acceleration, deceleration, speed factors determine to choose the best path.

To identify the emission of CO_2 microscopic and macroscopic fuel consumption models are available. The microscopic consumptions of fuel are determined in the vehicle periodically every second. Macroscopic consumptions of fuel of vehicles are predicted by their link speed. VT-micro model is used to reduce the emission and fuel consumption. If there are more vehicles in the same route, there will be more chances of CO_2 emissions, which cause an increase in air pollution. To avoid this the vehicle should get the alert message, and they can choose the finest path and predict the best route. By choosing the alternative path accidents is avoided. To choose the optimal route Global Positioning system (GPS) is used in vehicles, which contain the map-related information, routes can be identified easily and to choose the best type of roads.

The polluted area is identified by vehicles by choosing the best route from source to the destination using the GPS systems. Pollution can be regulated by measuring devices, which are equipped with vehicles. It is placed in moving vehicles, so that the pollution-related information can be identified easily. Through the Internet, the vehicles get the polluted related information up to date periodically. Map database is very helpful to identify the map information and roadways in a geographic region. Using the analyzer called as pollution analyzer program help vehicles to select the pollution fewer routes. When the vehicle density is high congestion occurs, so the only solution to avoid air pollution and fuel consumption is to avoid congestion by re-routing the vehicles. The Vehicular Ad-Hoc Network plays an important role to route vehicles effectively and directs the vehicle to the efficient fuel route. Traffic will be reduced by avoiding the crowding of vehicles. Traffic management is effectively done by VANET so that smooth traffic avoids congestion.

Proactive routing protocols play an important role in Re-routing of vehicles. It identifies the shortest path. When the vehicle is moving at high speed, the vehicle should get an alert message to decrease the speed of the vehicles this helps a lot to avoid air pollution. When the vehicle speeds are at 30kph, the air pollution can be avoided. The air pollution detecting device is placed in vehicles to identify the path which contains low air pollution. In VANET, the information of the effective path is identified by wireless networks such as GPRS, WiMax, Bluetooth, GPS, etc. the navigation systems present in the vehicles accepts the request from the user and analyzes the optimum route path and sends back the requested information. Traffic management application which comes under safety application of VANET helps to reduce the polluted paths by avoiding the congestion, collision, fuel consumption, and air pollution. So using VANET the polluted paths can be avoided in vehicular networks. To avoid congestion safety messages is sent to drivers through the navigation system so the driver can choose the alternative path to reach the destination, Geocast routing protocol is used for predicting the optimal route.

1.12 OUR CONTRIBUTIONS

As reflected in the previous sections, from the perspectives of GA, the work EVRP described in this thesis is motivated by TSP. The proposed system computational may offer a better component for the classical GA and hence the overall performance may definitely be uplifted to expected level.

For achieving the above, the research methodology of the work presented in this thesis is being organized into separate segments and the key contributions are itemised as follows:

- A well-focused survey has been conducted with the current advances in the related domain. This logical study ends with the predictability for developing the EVRP.
- Inspired by environmental problem three different scenario is being modelled and transformed into three set of computation models.
- Based on the TSP data set, we generate air pollution data set computation models, three different sets have been proposed for enhanced search space exploration and improved solution quality.
- Well-defined research are conducted to validate the complete performance of the proposed algorithmic models and the importance of the proposed models are verified using standard set of performance evaluation criteria.

1.13 SUMMARY

In summary, though several techniques have been proposed to solve the VRPs and its variants effectively, each technique has its own tradeoff values in terms of the performance factors. In accordance to the no free lunch theorem, there is no single technique to solve the whole family of problems effectively. On the other hand, apart from the traditional performance criteria such as optimal route w.r.t. the distance, time, complexity and scalability, it is very difficult to find the techniques to consider socially inspired performance factors such as air pollution, sound pollution, etc. In this perspective, the work reported in this thesis proposes an intelligent routing strategy for VRP based on the air pollution between the corresponding cities.

1.14 OUTLINE OF THE DISSERTATION

The document of the thesis contains introduction, literature survey, 3 main chapters, conclusions and references. At the end of each chapter summary is delivered (except conclusions). The total space of this thesis is 152 pages and 49 figures.

Chapter 1 deals with Introduction describes research background and inspiration, presents the vehicle routing problem, discusses types of genetic algorithm and its implementation tasks, genetic algorithm applications in VANET and objectives of the research in the stream of green computing.

Chapter 2 provides the literature survey and overview of vehicle routing problems and solutions, analyses genetic algorithms for solving vehicle routing problems in details.

Chapter 3 describes the experimental setup phases in detail about genetic algorithm with respect to vehicle routing problem.

Chapter 4 provides experimental evaluation of the proposed algorithms in first module that is, optimal distance based routing strategy.

Chapter 5 provides experimental valuation of the proposed algorithms in second module that is, optimal pollution based routing approach.

Chapter 6 proposed the hybrid model which combines both the first two modules optimal distance as well as optimal pollution for finding best route vehicle routing problem.

Chapter 7 presents the conclusions of the thesis.

Chapter 8 and chapter 9 describe the research references and publications details respectively.

CHAPTER 2

LITERATURE SURVEY

2.1 VEHICLE ROUTING PROBLEM.

The Vehicle Routing Problem (VRP) has got much consideration in recent years. Because of the effectiveness in real life and improvement of the transportation system as well as logistics, VRP attracts many researchers attention. VRP organize a class of combinatorial optimization problems concerned with the distribution of goods to customers. Different types of exact and heuristic models have been studied to resolve the VRP that is known to be NP-hard. Even though exact approaches give the optimal solution, their problem complexity increases as the size of the problem improves. Many classes of VRP are there but, in this chapter we chosen and studied five best and well known NP-hard problems.

To solve the Multi-Objective Vehicle Routing Problems (MO-VRP) [Pisinger et al, 2005] proposed a fast approximation heuristics and the heuristic depends on the savings approach. The solutions are enhanced by the local search against the pareto-front in iterative process. Based on the savings heuristic the initial solutions are generated and the solution is approximated by the pareto-front and then enhanced by the local search. This method has been tested on the beach mark it improves the initial approximation.

The quality of the individual in the current generation is sent to the next generation, influenced by the Lamark's method. This method has been used to maintain the best solution throughout the process and many proposals are used to solve the application local search operator [Schleuter et al, 1997], [Merz et al, 1997] and [Ross et al, 1999]. In [Milthers et al, 2009] they enhanced the genetic operators (crossover and mutation) using the feasible solution and proposed an Improved Genetic Algorithm (IGA). To improve the effectiveness they established three optimization strategies: immigration, local optimization and global optimization. Random population method is used, while generating the initial population, the chance of finding the optimal solution is very less and also the computation cost will be more.

[Milthers et al, 2009] The ALNS heuristic framework was further preceded by the influence of [Pisinger et al, 2005], [Pisinger et al, 2007] and [Ropke et al, 2006a] to find optimal route and least distance in different types of VRP. Instead of using a single large neighborhood as in LNS, this executes many removal and insertion operators in the solution. Depend on the previous performance these operators are chosen during the execution and set a score to the operators, which will boost the solution.

In our survey we discussed, how the VRP is solved in using different methods, with different parameters and different constrains. We proposed a socially inspired transportation problem, in this based on the pollution in the path, we are routing the vehicle. The experiments are done using the slandered TSP bench mark instances and then analyzed the performance with different initialization techniques.

2.1.1 Traveling Salesman Problem

The Traveling Salesman Problem (TSP) is a well-known NP-hard problem and considered as a standard test bed for different combinatorial optimization techniques. The aim of TSP is to find least cost route to visit each city exactly once and returns to the starting city. The problem can be formalized as follows:

Let $G = \{CITY_n , A_n\}$ be a complete undirected graph such that

$CITY \in \{City_1, City_2, City_3, \ldots City_n\}$ and

$A \in \{(City_1, City_2), (City_1, City_3), \ldots (City_1, City_n), (City_2, City_3),$

$(City_2, City_4), \ldots (City_2, City_n) \ldots (City_{n-1}, City_n)\}$. In the Graph $G, City$ represent the cities and A represents the arcs (i.e.) the path between the cities. The distance between the cities $City_i$ and $City_j$ can be given as,

$DM(City_i , City_j)$, where the $City_i \neq City_j$. TSP aims to find the optimal solution with minimum distance tour between cities which can be given as,

$$Optimal\ Solution = \sum_{k=1}^{n} DM(City_{k,k+1}) , \quad n+1 \equiv 1 \qquad \rightarrow 6$$

The size of the TSP search space with n cities is $\frac{(n-1)!}{2}$. Since the initial city of the solutions of proposed intelligent routing strategy remains same, the search space size for the problem would be reduced to $\frac{(n-2)!}{2}$.

To find the best shortest route or path among the n possible routes is not as like easy, because route complexity will keep on increase when the number of cities or nodes increases. The TSP comes in two different types depending upon the distance. If the distance between two cities *i* and *j* is the same distance from city *j* and city *i* then the problem is said to be symmetric. If this condition fails then it is said to be asymmetric.

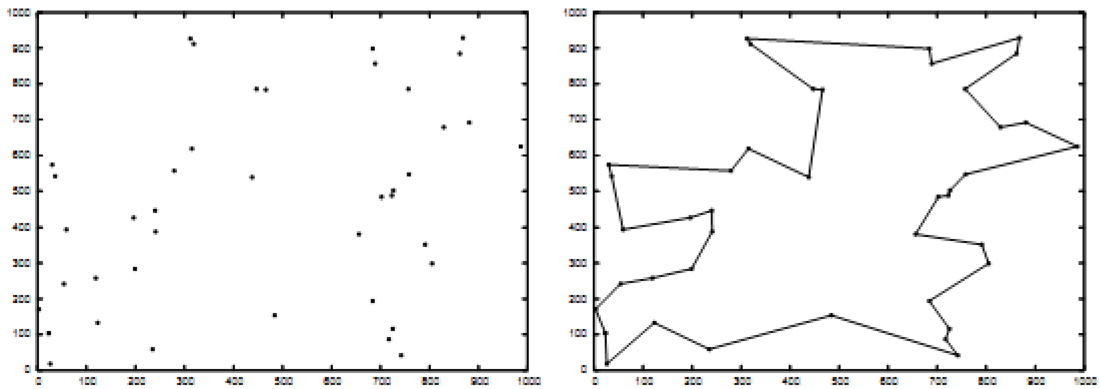


Figure 2.1.1(a) & 2.1.1(b): TSP illustration

So far the highest Euclidean instances of nodes or cities taken to solve TSP is 24,978 which was solved to optimally by the research team Applegate, Bixby, Cvatal, Cook and Helsgaun by using branch and cut technique. In Figure 2.1.1(a) shows an example of TSP with dots represents the cities or instances and Figure 2.1.1(b) shows the optimal shortest path between all instances. In VRP most common problem like Capacitated Vehicle Routing Problem (CVRP) or Pick and Delivery with Time Window Problem (VRPTW) are harder to solve in both methods Heuristic and Exact, when compared to TSP.

2.1.2 *m*- Traveling Salesman Problem

m- Traveling Salesman Problem (m-TSP) is an abstract principle model of the general TSP problem. Here they employees more than one salesman to visit all the cities exactly once and return back the stating depot or home at low cost. For example consider n cities, m salesman and one depot or home. The problem is to visit all the cities exactly once on one of *m* tours, with constrain both the starting and ending at same depot. The tour should not be empty. If distance or cost satisfy triangular inequality then it is easy to get that the cost of shortest TSP tour on the *n* cities plus the depot always is less than or equal to the distance of the shortest *m*-TSP result for any m salesman.

The m-TSP problem is not studied commonly, because it is closely connected the general TSP problem. The recent works about heuristic and exact method has been done by Bektas [2006]. Another alternative of the problem is min-max m-TSP which was introduced by Franca et al. [1995] the length of the longest salesman tour has to be reduced. In [2002] Applegate et al. solved the min-max m-TSP problem for the first time using 188 processors with computing time of 10 days.

2.1.3 Capacitated Vehicle Routing Problem

In Capacitated Vehicle Routing Problem (CVRP) is one of the most studied VRPs problem. The finest algorithms for the Capacitated Vehicle Routing Problem have been based on either branch-and-cut or Lagrangean relaxation/column generation. Ricardo Fukasawa et al. [2004], Lysgaard et al. [2004] and Blasum and Hochtattler [2000]. CVRP is also more or less same, related to TSP but the terminology which are used in this problem is different that is customers are employed instead of cites, fleet of vehicles are introduced as an alternative for salesman and with an extra performance factor of capacity or load is included.

The objective of CVRP is same as m-TSP that is, a set of n customers is served by a set of m vehicles with the extra constraint that every vehicles has a capacity Q and each customer i ∈ {1, 2,…., n} has a request qi The assignment in the CVRP is to build the vehicle routes such that all the customers are served exactly once and also the capacity of the vehicle constraint is followed. This should be done while total distance traveled. The CVRP was introduced by Dantzig and Ramser [1959] since then it is most interesting research in the field of Vehicle Routing Problem. Many heuristic methods have been proposed in the past 55 years. Heuristic introduced until around 1980 are surveyed in Christofides et al. [1979], whereas the successful heuristic until new period was surveyed in Laporte and Semet [2002] and Gendreau et al. [2002]. The most current advances in metaheuristics have been surveyed in Cordeau et al. [2004] and Bramel and Simchi- Levi [2002]. The best heuristic for the problem at present is the metaheuristic introduced by Mester and Braysy. [2005]. Many more exact methods are introduced through the years for the CVRP. The best of them are surveyed by Cordeau et al. [2005], Toth et al. [2002] and Naddef et al. [2002].

2.1.4 The Vehicle Routing Problem with Time Windows.

The Vehicle Routing Problem with Time Windows (VRPTW) generalizes the CVRP by combining travel time t_{ij}, service time s_i, route (i, j) with customers i and depot i. The vehicle should reach in advance or within the time stamp of a customer. If it arrives earlier the start of the time stamp, it has to wait until the time stamp before the service at the customer can start.

In recent years many researchers were proposed and surveyed a lot for both heuristic and exact methods for VRPTW. Various metaheuristics have been proposed for the VRPTW. A very simple metaheuristics proposed by Cordeau et al. [2002]. While the most successful and broad literature was given by Braysy et al. [2005]. It very difficult is predicate which metaheuristic is best for the VRPTW because of the several different parameters like robustness, speed and exactness. Mester et al. [2005] proposed hybrid evolutionary algorithm which is best fit for minimizing the number of vehicles being employed to serve all customers who place orders for goods.

Exact technique for the VRPTW has been proposed by several mathematicians. Many instances from the Solomon test set (Solomon [1987]) with 50 customers were still unsolved. But the researchers Kallehauge et al. [2005] and Jepsen et al. [2005] reported that the unsolved instances with 25 and 50 customers were solved by their new inequalities approach. The exact methods for VRPTW are done by column generation, with the branch and cut technique by kallehauge et al. [2005]. The improvements over the column methods were archived by solving pricing problem Irnich et al. [2003] and Feillet et al. [2004]. The pricing problem that was solved by column generation approaches was the Shortest Path Problem with Time Window and Capacity Constraints. (SPPTWCC) and Elementary Shortest Path problem with Time Window and Capacity Constraints (ESPPTWCC) that permitted cycles of length 3 or more in the shortest path. Irnich et al. [2003] and Boland [2005] proposed new algorithm which reduced the cycle length k value to 2. Righini [2005] proposed an ESPPTWCC algorithm for bidirectional search which shows improvements in shortest path.

2.1.5 The Vehicle Routing Problem with Pickup and Delivery

The Vehicle Routing Problem with Pickup and Delivery (VRPPD) or Pick and Delivery Problem with Time Windows (PDPTW) simplifies the VRPTW. Here the problem is not related to deliver the goods central depot to customer doorsteps. The customer need goods to be transported form the pickup location to a delivery location. The problem is defined as the $2n+2$ problem, that is pickup and delivery called request. Where n is the number of request. Another variant of VRPPD is Dial-a-Ride problem (DARP). Gendreau et al. [1998] proposed a tabu search for dynamic type of the VRPPD which follows metaheuristic methods. They used a neighborhood search technique to remove the visited request i from the route r_1 and reinserted into another route r_2 while ejecting another request j from r_2. Li et al. [2001] tested tabu search heuristic with 50 instances. But Hentenryck et al. [2006] proposed two stage heuristic to the VRPPD and produced good results than Li et al. [2001].

2.2 HEURISTICS FOR VRP

Vehicular Routing Problem has got much consideration nowadays towards optimization. Because of convenience, and advancements in the transportation and logistics, VRP keeps on drawing specialists' consideration. Various diverse and heuristic systems have been concentrated on to the VRP that is known to be NP-hard. Although precise routines give the ideal arrangement, their calculation time impressively increments with the expanding size of the issue.

Branch and bound (B&B): It is an advancement method which pursuit of every single conceivable arrangement while disposing of (pruning) an extensive number of non-promising solutions by arranging upper and lower limits of the amount to be enhanced. Depth first technique is utilized to look the tree, where nodes whose objective values are lower/higher than the current best are not investigated. Algorithm requires branching operator for part arrangements set into the littler ones and bounding operator for processing lower/ higher headed for the target capacity to be streamlined. Branch and cut (B&C) is a B&B method, where search space is lessened by including new requirements (cuts). Branch and bound calculation is suitable to unravel VRP of little occurrences with just couple of hubs. (Toth and Vigo, 2001; Toth and Vigo, 2002; Lysgaard et al., 2004; Yeun et al., 2008; Bektas et al., 2011; Vidal et al., 2013).

Local Search and heuristic methodologies frequently deliver a close ideal arrangement inside of a sensible computation time. These systems may be delicate to information sets given or require extra preparing information amid the learning process.

Metaheuristic is another methodology for tackling an intricate issue that may be excessively troublesome or tedious by traditional procedures. The metaheuristics that are connected to the VRP are taking after:

Simulated Annealing (SA): Simulated Annealing methodology mirrors the tempering procedure in metallurgy. To get away from the local neighborhood, the likelihood of tolerating crumbled move for the arrangement depends on the supposed "temperature". The higher temperature, the higher likelihood to acknowledge degraded solution. Temperature parameter is advanced amid the hunt, accordingly mimicking the cooling process in metallurgy. (Černý, 1985; Misevičius, 2003; Cordeau et al., 2005; Vidal et al., 2013).

Ant Colony Optimization (ACO): ACO methodology is roused by the conduct of the ants. In the nature, at first every insect meanders haphazardly and when the sustenance is found, the subterranean insect comes back to the state by setting down pheromone trails. At the point when different ants discover the way with pheromone trails they pass by that way with higher likelihood contrasting with go haphazardly. When pheromone trails dissipate, so more ways will vanish more than shorter ones due to time expected to go down the way and back once more. Dissipation method of the pheromone trails lead to improvement of the way length. (Rizzoli et al., 2007; Yeun et al., 2008; Jančauskas et al., 2012; Vidal et al., 2013).

Construction heuristics are deficient and productive systems. The objective of a Construction heuristic is to construct an answer for an issue. This is finished by iteratively adding elements to an at first purge arrangement until a complete arrangement is gotten. Heuristics go for acquiring great quality arrangements.

Construction heuristics for VRP build an arrangement of routes. Amid development they attempt to keep the aggregate separation of the arrangement as little as could be allowed. To accomplish this, the most ideal approach to expand the present route or arrangement is picked (this is called insatiable conduct) at every stride. Such choices are nearsighted as they consider just the present circumstance, taking choices that may be great now however awful in the master plan. Note that while Construction heuristics for the VRP are ensured to give an arrangement of attainable routes, it is conceivable that these routes can't be consolidated into a practical arrangement if the quantity of courses got is higher than the quantity of accessible vehicles.

Local Search is a deficient and perturbative system. The essential thought of Local Search is genuinely straightforward. The hunt considers arrangements one by one and records the best quality possible arrangement ever experienced. The space containing every conceivable arrangement (doable and infeasible) to an issue is known as the arrangement space. Local Search depends on the perception that by altering a given (plausible or infeasible) arrangement another, diverse (practical or infeasible) arrangement is gotten. Local Search in this way travels through the arrangement space, every stride comparing to an annoyance of the flow arrangement. Local Search will break down every arrangement it experiences to check its plausibility (if necessary) and records the best quality possible arrangement experienced in this way (called the officeholder arrangement). The procedure is halted once a given ceasing rule is come to (e.g. execution time or cycles without change to the occupant). At every emphasis in Local Search the area of the ebb and flow arrangement is developed and assessed. One of the neighbor arrangements is then chosen as the new current arrangement. Note that regularly more than one Local operator is utilized as a part of Local Search. The choice of the Local solution for move to is ordinarily done utilizing either a First Improvement or a Best change technique. In First Improvement the inquiry assesses the Local arrangements amid the development of the area and when a Local arrangement enhancing the nature of the momentum arrangement is found, the hunt moves to it. In Best change the complete Local is built and assessed.

Two vital ideas in Local Search are Intensification and Diversification. The arrangement space might contain ideal arrangements and locally ideal arrangements. Ideal arrangements are called all around ideal to separate from locally ideal arrangements. A nearby ideal is an answer that is ideal just in its Local (i.e. none of the Local arrangements enhances it). Escalation implies that the hunt is amassed in a particular region of the arrangement space as this zone appears to be most encouraging (commonly with the objective of winding up at a nearby ideal). Enhancement implies that the hunt is compelled to investigate diverse parts of the arrangement space with a specific end goal to ensure distinctive ranges are secured and the pursuit does not continue coming back to the same Local optimum. Increase and Diversification measures are usually executed in the assessment and choice of Local arrangements. Irregular choices can be incorporated into the decision of the area operator to use in the present cycle, the parameters used to develop this area, the choice of the Local arrangement or the restart from a haphazardly created arrangement.

Faster Heuristics, Larger Instances: The journey for faster Heuristics has been going ahead since the start of mechanized arrangement of Vehicle Routing issues, however improvements are as yet occurring and will keep on doing as such later on. A standout amongst the most imperative advantages of quicker Heuristics is that it will permit us to tackle bigger occasions, and this is without a doubt required in this present reality - genuine issues are regularly bigger than the 1000 client examples that ordinarily are the biggest occurrences considered by heuristic techniques. Some late research merits bringing up, Toth and Vigo [2003] portrayed an approach to diminish the running time of Tabu Inquiry, a technique they called Granular Tabu Search. The key thought in the Granular Tabu Search is to limit the area look by disposing of the most unpromising moves. It is possible by taking a gander at the circular segment lengths and classify the arc as either encouraging or unpromising, taking into account its length additionally on different elements like on the off chance that it is occurrence to the warehouse or has been utilized as a part of one of the best arrangements experienced as such. While doing the area seek, just moves that include no less than one promising arc are endeavored. The methodology was tried on CVRP occurrences with up to around 500 clients and demonstrated that the heuristic was quick considering the PC utilized.

More precise Heuristics: Heuristics that convey arrangements of astounding is a point that got a great deal of consideration, particularly since the landing of Meta Heuristics. It appears like the best of today's Heuristics are reliably ready to achieve arrangements whose expense is inside 1–1.5% of the Optimal or best known arrangement cost. For some uses of VRP this is adequate, as the information that can be gathered, in actuality, will be affected by blunders or clamor at any rate. Therefore, the thought of an Optimal Solution is not that essential when managing genuine occurrences by and large. Heuristics that create superb arrangements are by the by going to get consideration later on - one reason is that there dependably will be a sure individual fulfillment in seeing your heuristic produce arrangements superior to the beforehand best known! Another reason is that arrangement quality is anything but difficult to gauge and thusly an undeniable method for contrasting Heuristic.

2.3 GENETIC ALGORITHMS AND VRP

This section presents a brief introduction to a very recent population seeding technique of the GA and Traveling Salesman Problem (TSP) to improve the understandability of this research. A variety of Ordered Distance Vector (ODV) based population seeding techniques has been proposed to effectively generate the initial population for permutation coded GA [Victer et al, 2013].

2.3.1 Ordered Distance Vector (ODV)

Let ‘n’ be the total number of cities in the problem $City_1, City_2, City_3 .. City_n$. The ODV for any city can be formulated by sorting the $(n-1)$ number of cities based on the distance between corresponding cities. The ODV of the $City_x$ can be given as,

$$ODV(City_x) = City_y, City_{y+1}, City_{y+2} .. City_{n-1} \rightarrow 7$$

then,

$$DM\left(City_{x,y}\right) \leq DM\left(City_{x,y+1}\right) \leq DM\left(City_{x,y+2}\right) \ldots \leq DM\left(City_{x,n-1}\right) \rightarrow 8$$

where,

$DM\left(City_{x,y}\right)$ is the distance between the $City_x$ and $City_y$ using the distance matrix of the problem

The Ordered Distance Vector Matrix (ODM) is a constructed by arranging the ODV of each city in the problem. The ODM for the problem can be represented as,

$$ODM = \begin{vmatrix} \mathrm{ODV}(City_1) \\ \mathrm{ODV}(City_2) \\ \mathrm{ODV}(City_3) \\ \cdots \\ \mathrm{ODV}(City_\mathrm{n}) \end{vmatrix} = \begin{vmatrix} City_{1(y)} & City_{1(y+1)} & City_{1(y+2)} & \cdots & City_{1(n-1)} \\ City_{2(y)} & City_{2(y+1)} & City_{2(y+2)} & \cdots & City_{2(n-1)} \\ City_{3(y)} & City_{3(y+1)} & City_{3(y+2)} & \cdots & City_{3(n-1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ City_{n(y)} & City_{n(y+1)} & City_{n(y+2)} & \cdots & City_{n(n-1)} \end{vmatrix} \rightarrow 9$$

A novel factor Best Adjacent (ba) number is introduced in [26] which can be expressed as follows: in an optimal solution, any city $City_i$ is connected to city $City_j$ such that $City_j$ is one of the $City_i$’s nearest ‘ba’ number of cities.

2.3.2 ODV based Equi-begin with Variable diversity (EV) Method

The ODV based Equi-begin with Variable diversity (EV) method is preferred in this paper, since the initial city in the solutions of the intelligent routing strategy for VRP is assumed to remain the same. In EV method, a new random number within 'ba' number is generated before adding each city into every individual and each individual starts with the same city. The individuals in the population have high potential sequence of cities and the maximum number of feasible individuals that can be generated using EV method can be given as,

$$\max\big(tot(Pop_{ODM})\big) = ba^{n-1} \qquad \rightarrow 10$$

where,

$tot(Pop)_{ODM}$ is the total number of individuals in the population,

n is the total number of cities in the problem instance and

ba is the Best Adjacent number based on the n of the problem.

The initial population generated using EV method can be represented as,

$$Pop_{ODM} = \begin{vmatrix} \delta_1(1) & \delta_1(City_2) & \delta_1(City_3) & \cdots & \delta_1(City_n) \\ \delta_2(1) & \delta_2(City_2) & \delta_2(City_3) & \cdots & \delta_2(City_n) \\ \delta_3(1) & \delta_3(City_2) & 3(City_3) & \cdots & \delta_3(City_n) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \delta_n(1) & \delta_n(City_2) & \delta_n(City_3) & \cdots & \delta_n(City_n) \end{vmatrix} \qquad \rightarrow 11$$

where, the initial city remains same for each individual in the population, $\delta_1(1) \equiv \delta_2(1) \equiv \delta_3(1) \ldots, \delta_n(1)$

2.4 SHORTEST PATH SEARCH

At initial stage, the shortest path problem look like to be very simple. Global Positioning System (GPS) devices and other various systems find the shortest path between two locations somewhat faster than mathematical approach. When the result is set within a time limit or in few seconds, it does not seem very slow and the result is satisfactory. However, VRP systems deal with much wider route development tasks. The objective of the VRP is to find the optimal path, when a number of customers are serviced. The real-world VRP deals with road network and a job to reach distinct locations in the road graph from the starting location which is the shortest path problem. In order to solve the logistic task made up of k+1 nodes, the (k+1)*k shortest paths need to be calculated.

In VRP when searching for the shortest path in the road network, a graph with non-negative weights is generally used. An edge in the graph can be labeled by any numerical factors like distance, time, speed, etc. Generally Dijkstra's algorithm is commonly used for finding shortest path in the graph.

Bidirectional Dijkstra’s algorithm is one of the most frequently used speed-up modification algorithms. (Goldberg et al., [2006]; Koehler et al., [2006], Berrettini et al., [2009]). This technique computes a path starting a search operation from both sides at the same time.

Since Dijkstra's algorithm is static and solutions are made with a static graph, several preprocessing methods are used for speeding up the method. The oldest technique is to calculate the shortest paths between all k nodes was proposed by Romeijn and Smith, [1999]. The resulting k*k matrix is used to find next level route planning system. But, the use of such method in combination with road data of the real world would be very inefficient. Speed-up factors using highway hierarchies approach only searches the highway routes around the neighborhood between the source and destination. Koehler et al., [2006]; Knopp et al.,[2007].

2.4.1 Dijkstra's Algorithm

Dijkstra's Algorithm gives to every node j a pair of labels (pj, dj), where pj is the node preceding node j in the existing shortest path from 1 to j, dj is the length of this shortest path. Some of the labels are temporary, that is it will change at a upcoming period; some labels are permanent, that is they are fixed and the shortest path from 1 to other all nodes are permanently labeled.

We denote by djk the length of arc (j,k).

Step 1. Label node 1 with the permanent labels (∅,0). Label every node j, such that (1,j) is an arc in the graph, with temporary labels (1, dj). Label all other nodes in the graph with temporary labels (∅,∞).

Step 2. Let j be a temporarily labeled node with the minimum label dj, i.e. dj=min{dl: node l is temporarily labeled}. For every node k, such that (j,k) is in the graph, if dk> dj+djk then relabel k as follows:

pk=j, dk=dj+djk.

Consider the labels of node j to be permanent.

Step 3. Repeat step 2 until all nodes in the graph are permanently labeled. The shortest paths can be found by reading labels pj.

An example graph to find the shortest paths from node 1 to all other nodes is given below. The permanent nodes are marked with red color.

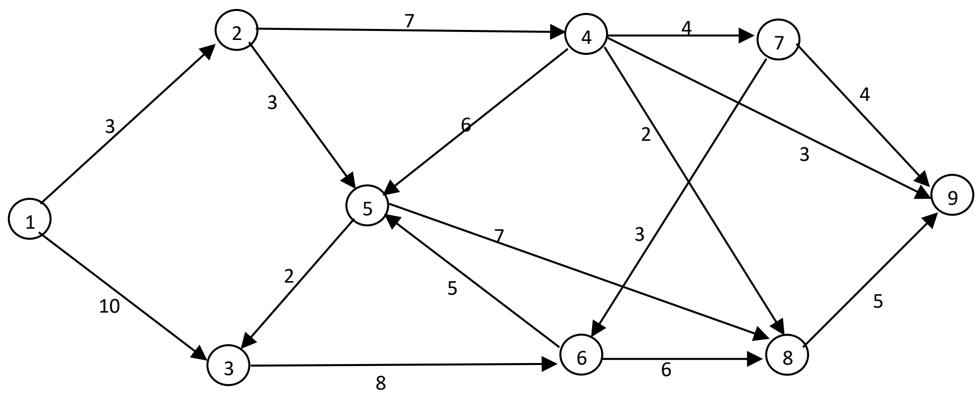


Fig 2.4.1 (a) Dijkstra's Algorithm graph illustration.

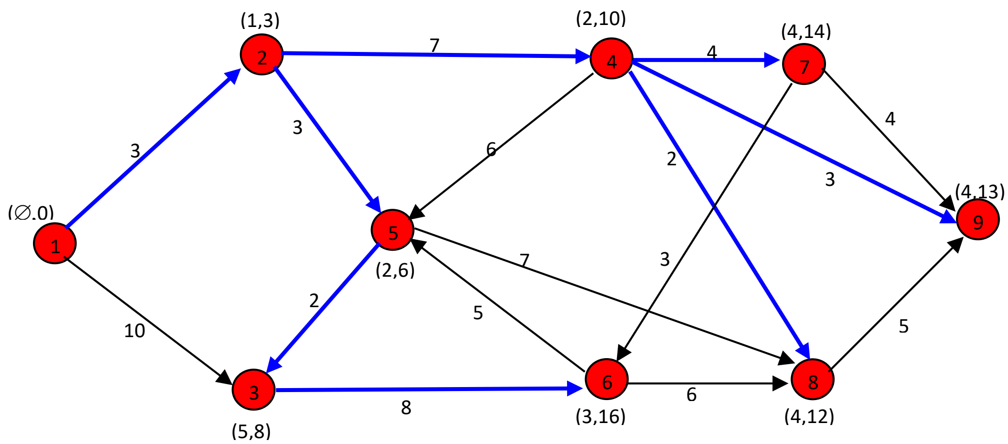


Fig 2.4.1(b) Dijkstra's Algorithm Shortest Path graph illustration.

2.4.2 Bidirectional Algorithm

In search of the shortest path between the node 1 to all other specific nodes of the graph, a modified Dijkstra's algorithm technique called Dijkstra's bidirectional method is used. This technique executes the searches starting from the start and end nodes (Goldberg et al., 2006; Berrettini et al., 2009).

This algorithm just split the graph into two part run by executing one step on each side in a single period. The algorithm is simply run by executing one step on each side in a single period. At first, the starting node is selected and processing step performed from the start side, and then the same process is done from the end node. During each step a distinct node is removed from the priority queue, marked as visited and all the equivalent edges are left to inner loop. Such process will be determined by the number of edges of all the visited nodes.

To implement such algorithm, unique data buckets must be used, so each search must have its own sets for labeled and visited nodes. In the subsequent pseudo-code, a forward search technique with priority queue QS and the set of visited nodes NSr and a backward search with priority queue QD and the set of visited nodes NDr.

Algorithm BidirectionalDijkstra*($Gr = (N^r, E^r), n_s^r, n_d^r$)*

$Q_S = \varnothing$ // labeled nodes in search from start
$Q_D = \varnothing$ // labeled nodes in search from end
$N_S^r = \varnothing$ // visited nodes in search from start
$N_D^r = \varnothing$ // visited nodes in search from end
…
while *Q_S is not empty* ***and*** *Q_D is not empty // outer loop calc from start*
$n_u^r = Q_S$. extractMin()
N_S^r.addNode()
…
    if *stoppingCriterion(n_u^r) is* ***true*** *// stopping criterion*
        break
    end if
for *each n_v^r adjacent to n_u^r // inner loop*
…
end for
// calc from end
$n_u^r = Q_D$.extractMin()
N_D^r.addNode(n_u^r)
…
    if *stoppingCriterion() is* ***true*** *// stopping criterion*
        break

end if
***for** each n_v^r adjacent to n_u^r // inner loop*
…
end for
end

The Bidirectional Dijkstra search algorithm stops when stopping criterion is satisfied. And this condition will takes place at the middle of the process because the algorithm starts execute at both the ends.

2.5 SUMMARY

As discussed earlier VRP is one of the most complicated NP- hard problem because of its dynamic route selection process. We analyzed various literature papers and reputed articles to understand and solve this VRP efficiently. Many researchers tried different level of approaches to solve VRP. But none of them combine VRP with environmental related problems like air pollution, sound pollution, etc., we took it as a challenge and relate the VRP with environmental problem. Nowadays air pollution is one of the major threat to our universe. So we considered one of our core objective is to reduce air pollution on the roadsides of both inside and outside of the cities.

To simulate such a huge problem Genetic Algorithm is one of the metaheuristic technique which can solve VRP effectively. To solve such a combinatorial optimization problem GA is generally used to produce good results towards NP-hard problems. GA works based on the evolutionary theory technique. The search solution of GA is depends upon two factors that is population seeding and selection technique. While seeding process initialize the population and the selection process selects the best individual from the entire population to reproduce next generation individual effectively. This process will continuous until the termination condition satisfies generally the termination condition will be generation limit.

In the real world scenario finding the shortest path between the nodes or cities in VRP are very important. We studied and reviewed many shortest path algorithms likes Dijkstra algorithm, bidirectional search algorithm, Ant colony optimization technique and some of the speed up and modified Dijkstra algorithm for finding the better solution in short time window.

CHAPTER-3

SYSTEM MODELLING AND EXPERIMENTATION METHODOLOGY

3.1 INTRODUCTION

The TSP is an NP-hard problem in combinatorial optimization, the objective is to find least possible route to visit each city exactly once and returns to the starting city. It organized into symmetric travelling salesman problem, asymmetric travelling salesman problem, and multi travelling salesman problem. In our thesis we are using symmetric travelling salesman problem,

Let $G = \{City_n , A_n\}$ be a complete undirected graph such that the cities $City \in \{City_1, City_2, City_3, \dots City_n\}$ and the arcs

$$A \in \{(City_1, City_2), (City_1, City_3), \dots (City_1, City_n) \dots$$

$$(City_2, City_3), (City_2, City_4), \dots (City_2, City_n) \dots$$

$$(City_{n-1}, City_n,), \}$$

In Graph $G, City_n$ represent the cities and A_n represents the arcs (i.e.) the path between the cities. The distance between the $City_i$ and the $City_j$ is represented as $DM(City_i , City_j)$, here the $City_i \neq City_j$. The TSP is formulated as follows,

$$Optimal\ Distance = \sum_{k=1}^{n} DM(City_k , City_{k+1}) , \quad i + 1 \equiv 1 \qquad \rightarrow 12$$

Since it is symmetric TSP, it gives $n(n-1)!/2$ possible solutions for any permutation of $n$cities. In our proposed system, the Starting city is same for the entire problem, so it gives the possible solutions $n(n-2)!/2$

3.2 PROPOSED SYSTEM

3.2.1 Problem Formulation

The main objective of vehicle routing problem is to find the route with least distance, as we discussed in the [Chapter 2] many proposals are there to solve this problem. We are contemplate this problem in different perspective and proposed a new socially inspired transportation problem. In this we routing the vehicle efficiently with respect to the pollution in the path. A vehicle wants to reach a destination; it is possible there may be any number of paths in between the starting point and the destination. Before going to enter into a path, the vehicle has to check the pollution in the path is within the limit or not. The pollution in the path is more than the limit, it has to take the alternative path otherwise it can proceed through the path. We applied this in the symmetric Travelling salesman Problem, the objective is to find least possible route to visit each city exactly once and returns to the starting city with respect to the pollution. (i.e.) our main objective is to minimize the pollution in the path and also the distance.

We generated a Pollution matrix for TSP bench mark instances. As we discussed in the background work, for the complete undirected graph $G = \{City_n, A_n\}$, the distance between the $City_i$ and the $City_j$ is represented as $DM(City_i, City_j)$, here the $City_i \neq City_j$ and $DM(City_i, City_j) = DM(City_j, City_i)$. The pollution between the $City_i$ and the $City_j$ is represented as $Poll(City_i, City_j)$and $POll(City_i, City_j) = Poll(City_j, City_i)$. In the entire problem the starting city is constant; because the vehicle should start from the dispatcher. We will discuss our proposed system with a diagrammatic example,

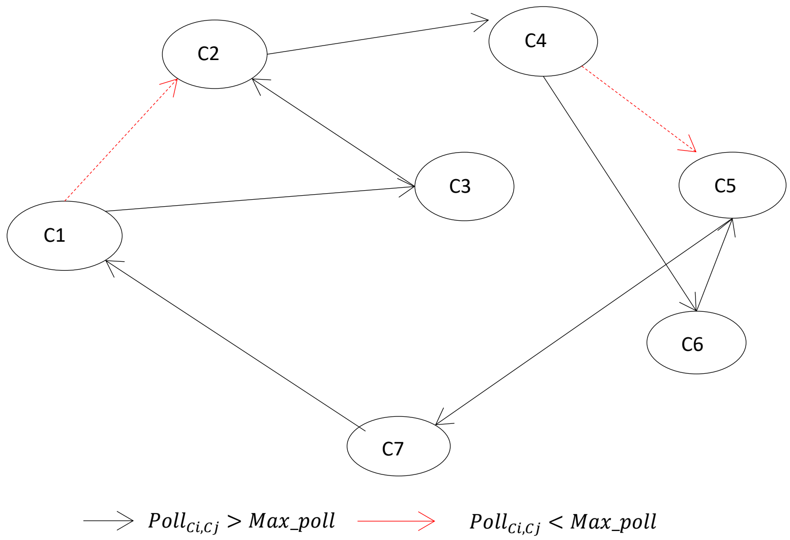


Fig.3.2. Show an example of proposed system

Explanation: In this example we have 7 cities$(C1, C2, C3 \ldots . C7)$, assume that the cities are arranged based on the distance$(d(C1) \leq d(C2) \leq C3 \ldots \leq C7)$. The objective of TSP is to visit all the cities once and then return to the stating city. Our objective is to visit the least distance city and the pollution should be less than the maximum pollution $Poll_{Ci,Cj} < Max\_poll$. The starting city is $c1$ and its target is to visit the next city $c2$ which is having the minimum distance, Now we are supposed to check the pollution between the cities $c1$ and $c2$. The pollution is less than the maximum pollution we can visit that city, otherwise we have to move to the next city with least distance. This will continue until we reach the starting city. Note that the pollution in the path between the $c7$and $c1$, the vehicle have to wait and then it can take that path.

To improve the efficiency of the optimal tour we are going for the GA. In GA many methods are used to generate the initial population, in this proposal we are using the ODV based population seeding technique, the initialization of the initial population is done through Equi-begin with Variable diversity (EV) method discussed in section 2.

3.2.2 Algorithm

Variables Used
n ←No. of cities
G←Maximum No. of Generations
$PopSize$←Population size / no. of feasible solutions
Max_Poll←Maximum pollution
$Pop_{Popsize \times n}$←Possible solutions / Search space
$Indiv_{m \times n}$←Single solution in the possible solutions / Search space
$City_n$←Subset of each individual
$POLL_{n \times n}$←Pollution Matrix, for all cities
$POLL(u, v)$←Pollution between the u^{th} city and v^{th} city
$DM_{n \times n}$← Distance Matrix
$DM(x, y)$←Cost between x^{th} City and y^{th}city
ba← Best Adjacent value
bax← Random value based on ba value $bax = RAND(1, ba)$
$ODM_{n \times ba}$←Ranked Matrix of Distance
ER← Elitist Rate for selecting best Individual from the Population
TR←Tradeoff rate between Distance and Pollution
P_Indiv_k←Parent Individuals, Randomly generated from the Population based on crossover rate $where\ \ 0 < k \ \leq 4$
TC_i ←Total cost of the Population,$where\ \ 0 < i \ \leq Popsize$
TP_i ←Total Pollution of the Pollutio$where\ \ 0 < i \ \leq Popsize$n,
Ω_k ← Normalized values of different cities, $where\ \ 0 < k \ \leq 4$

Assumptions
The current Pollution will be update periodically, between the city paths $City_i\ to\ City_j$, where $0 < i, j \leq n$
Limit of Maximum Pollution Max_Poll for all the cities are Constant.
The initial city is fixed in population generation.
TR Trade off Rate is set to 0.5.
ER Elitist Rate is set to 0.4.

Fig.3.2.2(a) List of variables used in the Algorithm

In our proposal we are dealing this problem with three scenarios. One is Optimal Distance based routing, second one is optimal pollution based routing and the third one is hybrid optimal routing based on the pollution and the distance. The initialization phase is same for all three method, now we will see the algorithm and its explanation.

Population Initialization Algorithm

Step 1: Set the parameters n ,Max_Gen,$PopSize$,Max_Poll

Step 2: Generate the Initial Population using ODV-EV method

Step 2.1: Initialize the initial city $Cur\_city,\ Len \leftarrow 1, Size \leftarrow 1$

Step 2.2: Add the initial city in the Individual, $Indiv[Len] = Init\_city$

Step 2.2: Repeat through Step 2.8 **Until** $Size \leq PopSize$, else goto Step 3

Step 2.3: Repeat through Step 2.4 **Until** $Len \leq n$, else goto Step 2.6

Generate a random number $'R' where\ 1 \leq\ R \leq ba$

$Cur\_City \leftarrow Indiv[Len]$*// current city in the individual is assigned as current city*

$Next\_city \leftarrow ODM(Cur\_city, R)$*//extort the next city from the ODM*

Step 2.5: **IF**$Next\_City\ \notin\ Indiv\ and\ POLL_{Cur\_City,Next\_City} < Max\_Poll$, else goto Step 2.3

Step 2.6 $Len \leftarrow\ Len + 1$,Update$Indiv[Len] \leftarrow next_City$

Step 2.7: Update $POP_{Size} \leftarrow\ Indiv$

//move the completed individual to the population

Step 2.8: $Size\ \leftarrow Size + 1$ *// goto the next individual*

Step 3: Evaluate the cost of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TC_i\ \leftarrow \sum_{j=1}^{n} DM(Indiv_i(j)\ , Indiv_i(j+1))\ ,\quad j+1\ \equiv 1$$

Step 4: Evaluate the Pollution of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TP_i\ \leftarrow \sum_{j=1}^{n} POLL(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1\ \equiv 1$$

Step 5: **If**$scenario =\ 1$ then, $Pop\ \leftarrow Distance\_Algorithm(Pop, TC, G, n)$

Else If$scenario = 2\ then,\ Pop\ \leftarrow Pollution\_Algorithm(Pop, TP, G, n)$

Else $Pop\ \leftarrow Tradeoff\_Algorithm(Pop, TC, TP, G, n)$

End If

Step 5: **Return**Pop

Fig. 3.2.2(b) Algorithm for Population Initialization using EV method

3.2.3 Algorithm Explanation

Population Initialization

Set the parameters, number of cities as n ($City_1, City_2, City_3, \ldots City_n,$), maximum number of generation G , population size $PopSize$ and maximum pollution in the path between the two cities $poll_{x,y}$. The cities are arranged in the increasing order using the Distance Matrix (DM) and then rank the cities move to the Ordered Division Matrix (ODM) matrix. Bubble sort is used for arranging the cities. Depending on the ODM, the initial population is generated using equi-begin with the variable diversity (EV) Method. In that first get the initial city and then make it as current city in the individual. Generate the next city randomly from the ODM with in the limit of 'ba' value (i.e.) $'R'\ where\ 1 \leq\ R \leq ba$ (Depending on the population size the 'ba' value will change). Check whether the next city is already present in the Individual and also the Pollution between the current city and the next city should be minimum (i.e. the pollution between the cities should be less than the maximum pollution)

$Next\_City\ \notin\ Indiv\ and\ POLL_{Cur\_City,Next\_City} < Max\_Poll$ and then add the city in the Individual. Do these procedures until we get a complete individual and then add the individual to the population $POP_{Size} \leftarrow\ Indiv$ do this procedure till we reach the maximum number of population, this process is population initialization. After initialized the Population, evaluate the total cost and total pollution of all the individuals from the whole population.

$$\forall[1, \leq i \leq PopSize],\ TC_i\ \leftarrow \textstyle\sum_{j=1}^{n} DM(Indiv_i(j)\ , Indiv_i(j+1))\ ,\quad j+1\ \equiv 1 \rightarrow 13$$

$$\forall[1, \leq i \leq PopSize], TP_i\ \leftarrow \textstyle\sum_{j=1}^{n} POLL(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1\ \equiv 1 \rightarrow 14$$

Now we are going for Genetic Algorithm, to improve the effectiveness of the population. In this we are analyzing three scenarios, the first one is based on the Cost, second one is based on the pollution and the third one is tradeoff between both distance and pollution.

All the implementations are done using MATLAB with TSP benchmark datasets obtained from TSPLIB [35]. The TSP instances that have been chosen

Fig 3.2.3- GA Configuration parameters

S.No	Parameter	Value / Technique
1	Population Size	100
2	Generation Limit	200
3	Crossover Method	Greedy crossover
4	Crossover Probability	0.7
5	Mutation Method	Swap Mutation
6	Mutation Probability	0.1
7	Elitism	True (4 individuals)
8	Termination Condition	Generation Limit

for experimentation are ulysses16, ulysses22, bays29, att48, eil56, eil76 and kroA100. A pollution matrix, similar to distance matrix, is generated for every instance with fixed optimal pollution route and this pollution matrix is available for validation.

The GA parameters and the corresponding values are depicted in the Fig. 3.4. For each technique, the executions are carried out for 50 times and the average of each case has been considered for experimental analyses.

3.3 EXPERIMENTAL PHASES

The experiments in this research are carried out in three different phases based on the population initialization technique used to generate the initial population. The performance of two different scenarios of intelligent routing strategy for VRP, as discussed in Section 3, is assessed using three population seeding techniques namely random, nearest neighbor and ODV based EV method.

3.3.1 Random population seeding technique

Randomly a finite set of individuals which are generated by choosing random adjacent cities is called a population. To improve the search space exploration an uniform random number generator has been used. The time taken to generate the initial population is less in random population seeding technique.

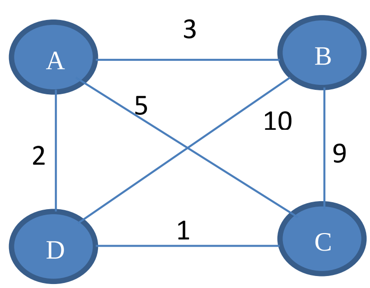
Steps

Step 1. Calculate the total number of tours.

Step 2. Draw and list all the possible tours.

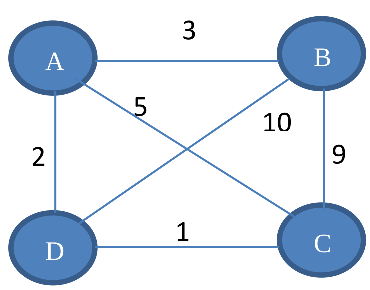
Step 3. Calculate the distance of each tour.

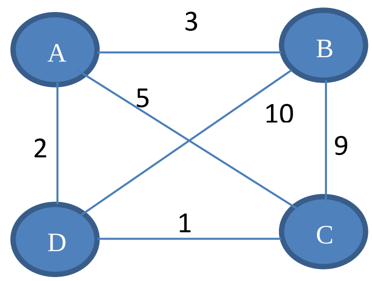
Step 4. Choose the shortest tour, this is the optimal solution.



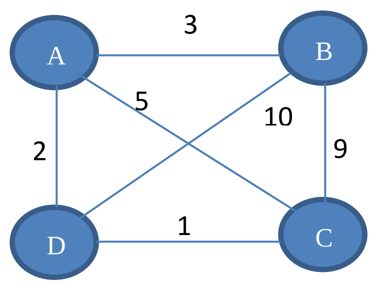
Here, there are four nodes. There is a possibility of the following 3 paths.

$$A \to B \to C \to D \to A = 15$$





$$A \to C \to B \to D \to A = 26$$



The shortest distance path is $A \to B \to C \to D \to A = 15$

3.3.2 Nearest Neighbor (NN) technique

Nearest neighbor (NN) tour construction heuristic is a common choice, in alternative for random population initialization, to construct the initial population. In NN technique, individuals in the population seeding are constructed that the gene 'y' can be selected as adjacent gene for the gene 'x' such that it would be the nearest unallocated gene of the individual of gene 'x'.

<u>Steps:</u>

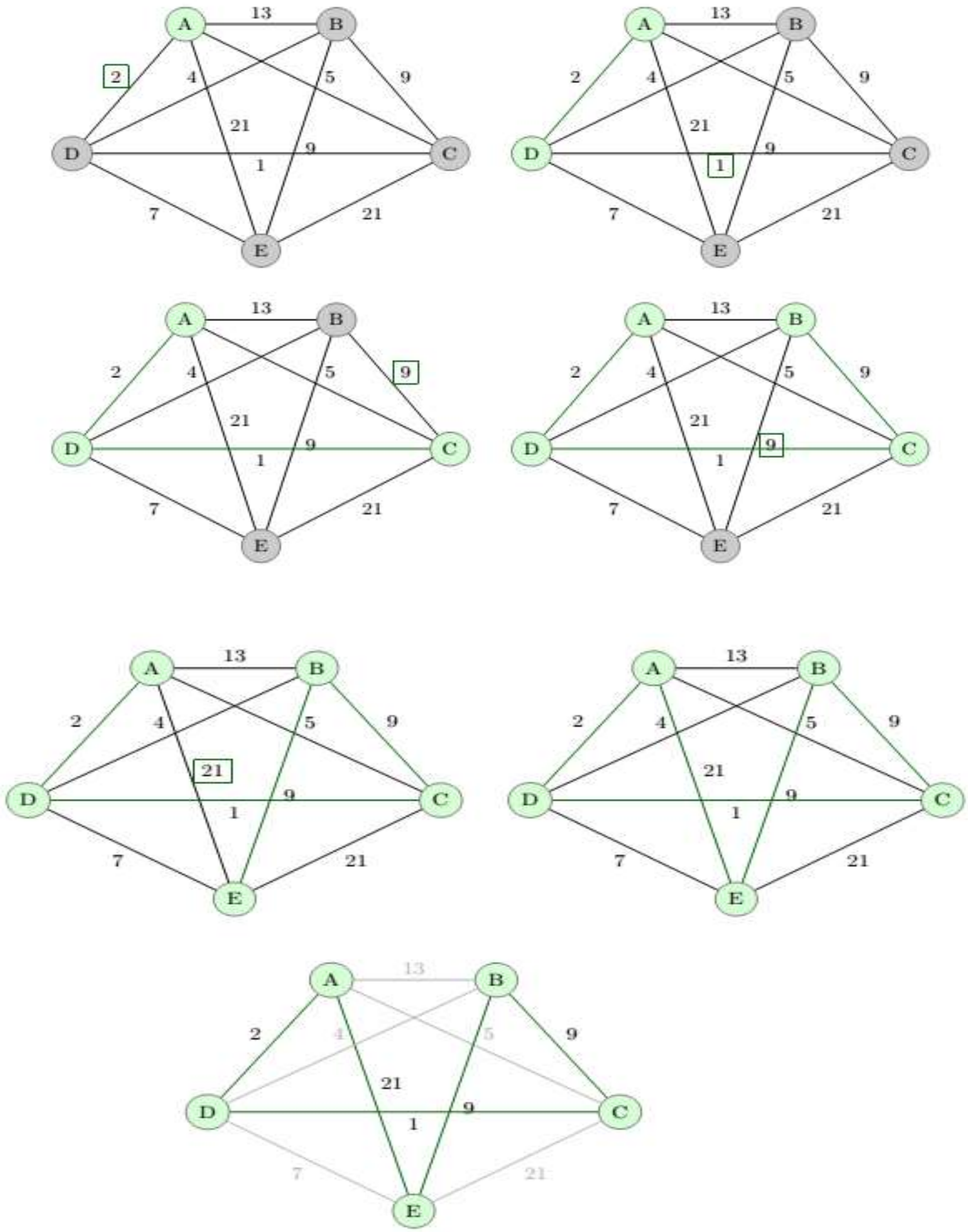
Step 1. Select a random city.

Step 2. Find the nearest unvisited city and go there.

Step 3. Are there any unvisited cities left? If yes, repeat step 2.

Step 4. Return to the first city.

This is the step-wise approximate solution by nearest neighbor method. This case has 5 nodes. We start with the node A and perform the nearest neighbor algorithm.



3.3.3 ODV based EV Technique

3.3.3.1 Equi-begin with Variable diversity (EV):

To initialize the initial population we using an effective population initialization technique, since the starting city is fixed in our proposal, we are moving to The EV (Equi-begin with Variable diversity) based ODV (Ordered Distance Vector) population seeding technique based on the ODV matrix [Victer et al, 2013].

3.3.3.2 Ordered Distance Vector (ODV)

In ODV the cities are sorted based on the distance that is computed by the permutation of $C_1, C_2, C_3 .. C_n$ cities. The ODV of a C_x is,

$$ODV(C_x) = C_y, C_{y+1}, C_{y+2} .. C_{n-1} \qquad \rightarrow 15$$

Then,

$$D(C_{x,y}) \leq D(C_{x,y+1}) \leq D(C_{x,y+2}) \ldots \leq D(C_{x,n-1}) \qquad \rightarrow 16$$

$$Where\ D(C_{x,y}) is\ the\ distance\ between\ the\ C_x and\ C_y$$

For each city, the ODV generates corresponding least distance cities in sorted order and rank the cities based on the distance, then it will moved to the ODM (Ordered Division Matrix) that is given by $n(n-1)$ matrix,

$$ODM = \begin{vmatrix} \mathrm{ODV}_{C_1} \\ \mathrm{ODV}_{C_2} \\ \mathrm{ODV}_{C_3} \\ \cdots \\ \mathrm{ODV}_{C_n} \end{vmatrix} = \begin{vmatrix} C_{1(y)} & C_{1(y+1)} & C_{1(y+2)} & \cdots & C_{1(n-1)} \\ C_{2(y)} & C_{2(y+1)} & C_{2(y+2)} & \cdots & C_{2(n-1)} \\ C_{3(y)} & C_{3(y+1)} & C_{3(y+2)} & \cdots & C_{3(n-1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ C_{n(y)} & C_{n(y+1)} & C_{n(y+2)} & \cdots & C_{n(n-1)} \end{vmatrix} \qquad \rightarrow 17$$

- Best adjacent value (ba):

This method is efficient because of the ba value; it is used to select the next city in population generation. The other techniques a random city is added as next city. Based on the size of the population, size of the ba value has been changed. The size for best adjacent (ba) value is assigned as follows:

- $If\ 1 \leq n \leq 10, then\ 2 \leq ba \leq 3.$
- $If\ 11 \leq n \leq 100, then\ 2 \leq ba \leq 4.$
- $If\ 101 \leq n \leq 1000, then\ 2 \leq ba \leq 5.$
- $If\ n \geq 1001, then\ 2 \leq ba \leq 6.$

- Equi-begin (Eb):

The starting city of the each individual is always same (i.e.) $'C_1'$ is fixed for all the individuals in the population. In our proposal, the starting city of the individuals is fixed so we are applying this method.

- Variable diversity (Vd) :

The Next city in the individual is added based on the $'bax'$ value, $'bax'$ is an integer that selected within the range $'ba'$ value. The city in the position of bax value is moved to the next city location of the individual.

3.3.3.3 ODV - EV Method

As we discussed above, Using EV method we generate the population, in that the starting city of each individual is fixed and based on the $'bax'$ value the next city of the individual is chosen and added. The individuals in the population have high permutation of cities and the time complexity can be reduced. Number of maximum individuals in the population during initialization is,

$$\max\left(tot(Pop_{ODM})\right) = ba^{n-1} \qquad \rightarrow 18$$

Where, $tot(Pop)_{ODM}$ is the total number of individuals in the population is, ba is the best adjacent value and n is the number of cities.

The initial population is generated using EV method is given by,

$$Pop_{ODM} = \begin{vmatrix} \rho_1(1) & \rho_1(C_2) & \rho_1(C_3) & \cdots & \rho_1(C_n) \\ \rho_2(1) & \rho_2(C_2) & \rho_2(C_3) & \cdots & \rho_2(C_n) \\ \rho_3(1) & \rho_3(C_2) & \rho_3(C_3) & \cdots & \rho_3(C_n) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \rho_n(1) & \rho_n(C_2) & \rho_n(C_3) & \cdots & \rho_n(C_n) \end{vmatrix} \quad \rightarrow 19$$

3.2 PERFORMANCE CRITERIA

There are four performance factors used to investigate the significance of the experimentation techniques and they are summarized as follows:

3.3.4 Convergence Rate

Convergence rate of an individual of a population set is defined as the percentage of fitness value obtained by the individual according to the optimal fitness value.

$$Convergence\ Rate(\%) = \left(1 - \frac{Fitness\ cost - Optimal\ cost}{Optimal\ cost}\right) * 100 \quad \rightarrow 20$$

3.2.3 Average convergence rate

Average convergence rate of a population set is defined as the average percentage of fitness value obtained by the individual according to the optimal fitness value

$$Average\ Convergence(\%) = \left(1 - \frac{Average\ fitness\ cost - Optimal\ cost}{Optimal\ cost}\right) * 100 \quad \rightarrow 21$$

3.2.4 Error rate

Error rate of an individual of a population set is defined as the percentage of difference between fitness obtained by the individual and optimal fitness value

$$Error\ Rate\ for\ Cost = \left(\frac{Fitness\ \ - Optimal\ fitness}{Optimal\ fitness}\right) * 100 \qquad \rightarrow 22$$

$$Error\ Rate\ for\ Cost = \left(\frac{Fitness\ cost - Optimal\ cost}{Optimal\ cost}\right) * 100 \qquad \rightarrow 24$$

$$Error\ Rate\ for\ Pollution = \left(\frac{Fitness\ Pollution - Optimal\ Pollution}{Optimal\ Pollution}\right) * 100 \qquad \rightarrow 25$$

3.2.5 Convergence Diversity

Distinct individual are the different individuals from the population. It is also an important factor that the increase and decrease in distinct individual reduces the performance.

The maintenance of a diverse solution population is required to ensure that the solution space is adequately searched, especially in the earlier stages of the optimization process. Population Diversity is considered as the primary reason for premature convergence. Hence a very homogeneous Population is found i.e. little Population Diversity is considered as the major reason for a Genetic Algorithm to premature converge. Premature convergence occurs when the population of a GA reaches such a suboptimal state that the genetic operators can no longer produce offspring that outperform their parents.

$$Convergence\ diversity\ (\%) = CR_{Highest} - CR_{Least} \qquad \rightarrow 26$$

$$Convergence\ diversity\ for\ Cost\ (\%) = CRC_{Highest} - CRC_{Least} \qquad \rightarrow 27$$

$$Convergence\ diversity\ for\ Pollution(\%) = CRP_{Highest} - CRP_{Least} \qquad \rightarrow 28$$

3.3 SUMMARY

This chapter describes various Genetic algorithm operators and parameters which are used to analysis the performance of the proposed Genetic algorithm with respect to Vehicle Routing Problem. The experimental was carried out in three different phases based on the GA population initialization technique that is Random population seeding, Nearest Neighbor (NN) and EV (Equi-begin with Variable diversity) based ODV (Ordered Distance Vector). Finally the last section of this chapter delivers the various performance factor that is Convergence rate, Average Convergence Rate, Error Rate and Convergence Diversity. In brief this chapter covers overall experimentation flow of the research with three different phases.

CHAPTER - 4

SCENARIO-1: OPTIMAL DISTANCE BASED ROUTING

4.1 INTRODUCTION

Optimal Distance Based Routing in VRP is intended to find the optimal route based on “the total distance of the route” as a primary factor. The total air pollution of the derived route is considered as the secondary factor for the optimal solution selection. The initial population for the problem is generated using random or heuristic method and the quality of individuals is improved in terms of distance of the route in each generation. The crossover and mutation operators are chosen in such a way to minimize the total distance of the parent routes in every generation.

In Optimal Distance Based Routing, the problem has been observed towards a single objective w.r.t distance. Depend on the minimum cost tour (distance), the selection and the crossover operation has been performed. In selection process, the initial population consists of possible solution for the problem defined. The distance cost of every individual within the population is determined and then the individuals those having least tour cost w.r.t distance have selected as an elitist individual. ER is the Elitism Rate, depending on that the number of elitist individuals are selected and passed to next generation population. This elitism transfer technique avoids the replacement of best fit individuals with poor individuals in the successive generations and also improves the performance of crossover operation, if the parent is selected from the elitist individuals.

The crossover is done through greedy crossover technique then selected two random individuals from the total population as parent individual. Since the starting city is same for all individuals, the starting city is the initial city of the offspring. Find the position of the current city in both the parent individuals and then locate the right side and left side city to the current city in both the parent individuals. If the position of current city is the starting city then location of left city to the current city is last city and if the position of current city is last city then location of Right city to the current city is last city. Check the cities in the locations are less than the maximum pollution $POLL_{Cur\_City\ ,Next\_City} < Max_{Poll}$. Based on the minimum distance cost the cities are added in offspring individual, after generated the complete individual move to the Mutation process, two random locations are generated and then swap the offspring individual cities in the locations and vice versa. The offspring is added into the population. Stop this process until we reach the total population.

4.2 PROBLEM DESCRIPTION AND FORMULATION

In Optimal Distance Based Routing, we are moving towards a single objective, distance. From the generated initial population we achieved the optimal distance base routing based on the distance. Depending the minimum distance, the selection and the crossover operation has been performed. In selection process, the elitist individuals have been selected from the population.

$$TC_j = \begin{vmatrix} \mathrm{TC}_1 \\ \mathrm{TC}_2 \\ \mathrm{TC}_3 \\ \cdots \\ \mathrm{TC}_{PopSize} \end{vmatrix} = \begin{vmatrix} DM(\mathrm{Indiv}_1) \\ DM(\mathrm{Indiv}_2) \\ DM(\mathrm{Indiv}_3) \\ \cdots \\ DM(\mathrm{Indiv}_{PopSize}) \end{vmatrix} = \begin{vmatrix} \mathrm{Indiv}_1 \\ \mathrm{Indiv}_2 \\ \mathrm{Indiv}_3 \\ \cdots \\ \mathrm{Indiv}_{PopSize} \end{vmatrix} =$$

$$\begin{vmatrix} City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \end{vmatrix} = CPOP_{PopSize \times n} \qquad \rightarrow 29$$

Here $CPOP_{PopSize\times n}$ is the current population, which consists of possible solutions.$PopSize$ is the population size and n is the number of cities. Each tour or possible solution in the population is represented as individual $Indiv_{\,j}$ and TC_j is the total cost of each individual in the population. The individuals having best fitness or minimum tour costs are selected and send to the next generation.

$$POP_{ER\times n} = \begin{vmatrix} City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \end{vmatrix} = \begin{vmatrix} \mathrm{Indiv}_1 \\ \mathrm{Indiv}_2 \\ \mathrm{Indiv}_3 \\ \cdots \\ \mathrm{Indiv}_{ER} \end{vmatrix} = \begin{vmatrix} \mathrm{TC}_1 \\ \mathrm{TC}_2 \\ \mathrm{TC}_3 \\ \cdots \\ \mathrm{TC}_{ER} \end{vmatrix}$$
$$= \min(TC_{ER}) \quad \rightarrow 30$$

Where ER is the Elitism Rate, depending on that the number of elitist individuals are selected and passed to next generation populationPOP. $TC_1, TC_2, TC_3, \ldots TC_{ER}$are the minimum cost of each individual arranged in the increasing order and $Indiv_1, Indiv_2, Indiv_3, \ldots Indiv_{ER}$ are the corresponding individuals to the minimum cost.

The crossover is done through greedy crossover. Selected two random individuals from the total population as parent individual. Since the starting city is same for all individuals, the starting city is the initial city of the offspring. Find the position of the current city in both the parent individuals $Pos1 \leftarrow find(P\_indiv1\ (Cur\_city)),\ \ Pos2 \leftarrow find(P\_indiv2\ (Cur\_city))$ and then locate the right side and left side city to the current city in both the parent individuals. If the position of current city is the starting city then location of left city to the current city is last city $IF\ Pos1 = 1\ then\ \ LLoc1 \leftarrow n$ and if the position of current city is last city then location of Right city to the current city is last city $IF\ Pos1 = n\ then\ RLoc1 \leftarrow 1$. Check the cities in the locations are less than the maximum pollution $POLL_{Cur\_City\ ,Next\_City} < Max_{Poll}$. Based on the minimum cost the cities are added in offspring individual, after generated the complete individual move to the Mutation process, two random locations are generated and then swap the offspring individual cities in the locations and vice versa. The offspring is added into the population. Stop this process until we reach the total population.

$$GeneC1 \leftarrow RAND\ (1, n,), GeneC2 \leftarrow RAND\ (1, n) \qquad \rightarrow 31$$

$$\text{Swap}\ \ Indiv(GeneC1)\ \leftarrow Indiv(GeneC2) \qquad \rightarrow 32$$

In this work, the standard VRP is contemplated in a different perspective to propose a new environment concerned transportation problem in which the optimal path should be of least distance and also minimum air-pollution along the route. A pollution matrix of TSP, similar to distance matrix, is formulated to specify the pollution between each pair of cities. A pollution limit between the cities is the maximum allowed pollution value between any two cities in any feasible solution for the problem. During the formulation of solution, at each stage, inclusion of a new city is allowed only if the pollution value between the previous and the new city is less than that of maximum allowed pollution limit between the cities else, it would try to select the alternate city. The intelligent routing strategy for VRP in Optimal Distance Based Routing can be represented as follows:

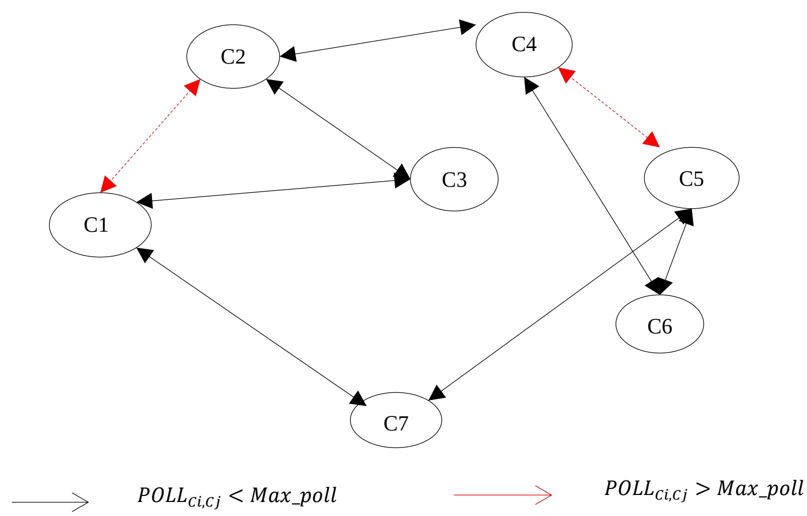


Fig.4.2. Sample intelligent routing strategy for VRP

4.3 ALGORITHM DEVELOPMENT

In this work, the intelligent routing strategy for VRP has been proposed using genetic algorithm for VRP in Optimal Distance Based Routing. In Optimal Distance Based Routing, our main objective is to minimize the distance and should satisfy the primary condition, the pollution between two adjacent cities should be lesser than the maximum pollution. Based on the distance alone, genetic operators applied to improve the effectiveness of the distance based routing of the population in each generation.

In optimal distance based routing, our goal is to reduce the Total Cost(TC) ,based on the cost of the individuals we are applying the genetic Algorithm to improve the effectiveness of the Distance based population. Get the Population, evaluated cost of each individual in the population, number of generation, number of cities, maximum population and Assign the population into a temporary population.

Selection: In selection, our main idea is to pass the best individuals to the next generation. Select ER number of best individuals from the temporary population, whose cost should be minimum than the other individuals $position \leftarrow \min(TC)$. The best individuals are moved to the population $Pop_i \leftarrow TPop_{position}$, we are following elitism in the selection process.

Crossover: In this choose any two random Parent individuals within the total population. The first city in the parent individual is moved to the offspring individual first city. Assign the current city as current city in the offspring. Now find the position of the current city in both the parent individuals $Pos1 \leftarrow find(P\_indiv1\ (Cur\_city))$, $Pos2 \leftarrow find(P\_indiv2\ (Cur\_city))$ and then locate the right side and left side city to the current city in both the parent individuals. If the position of current city is the starting city then location of left city to the current city is last city $IF\ Pos1 = 1\ then\ LLoc1 \leftarrow n$ and if the position of current city is last city then location of Right city to the current city is last city $IF\ Pos1 = n\ then\ RLoc1 \leftarrow 1$. Evaluate the cost of all the locations from the current city and choose the next city which is having the minimum cost. Check whether the next city is already present in the offspring Individual and also the Pollution between the current city and the next city should be minimum (i.e. the pollution between the cities should be less than the maximum pollution) $Next\_City\ \notin\ Indiv\ and\ POLL_{Cur\_City\ ,Next\_City} < Max\_Poll$ and then add the city in the offspring Individual. Do these procedures until we are getting a complete offspring individual.

Mutation: In Mutation, Generate two random locations $GeneC1 \leftarrow RAND\ (1, n,), GeneC2 \leftarrow RAND\ (1, n)$and then swap the offspring individual cities in the locationsSwap $Indiv(GeneC1) \leftarrow Indiv(GeneC2)$ and vice versa. Now add the offspring individual to the population, do this procedure till we reach the maximum number of temporary population. Evaluate the total cost and total pollution of all the individuals from the whole population.

4.3.1 Algorithm

Fig 4.3.1(a). - List of variables used in the algorithm

n	Total number of cities in the problem
G	Generation limit
$PopSize$	Total number of solutions in a population
Max_Poll	Maximum pollution limit between any two cities
Pop	Population in the generation with size $PopSize$
$Indiv$	Individual in the population
$City_n$	Subset of each individual
$POLL_{n \times n}$	Pollution Matrix for the problem of size n
$POLL(x, y)$	Pollution value between the city x and city y
$DM_{n \times n}$	Distance Matrix
$DM(x, y)$	Distance between city x and city y
ba	Best Adjacent number
bax	Select the value randomly within the range ($1 \leq bax \leq ba$)
$ODM_{n \times ba}$	Ordered Distance Matrix for the problem
ER	Elitist Rate
P_Indiv	Parent Individual selected from the Population
TC_i	Total cost of the i^{th} individual in the population, $where\ 0 < i \leq Popsize$
TP_i	Total pollution of the i^{th} individual in the population, $where\ 0 < i \leq Popsize$

<u>Assumptions</u>

- The current pollution is updated periodically, between the cities
- Maximum Pollution Limit Max_Poll between any two cities is constant.
- The initial city is fixed for all the solutions in the n population.

Fig 4.3.1(b) – Algorithm for Optimal Distance Based Routing

<u>Optimal Distance based routing Algorithm (</u>Pop, TC, G**,** n**)**
Step 1: Initialize$Gen \leftarrow 1, i \leftarrow 0, k \leftarrow 0$
Step 2: Storethe Population into a temporary variable, $CPop_{n\times n} \leftarrow Pop_{n\times n}$,
Step 2: Repeat through Step 10 **Until**$Gen \leq G$, else go to Step 10
Step 3: Select ER number of elitist individual which is having minimum cost
Step 3.1: Repeat through Step 3.3Until $i < ER$, else goto Step 11
Step 3.2: $position \leftarrow \min(TC)$ *// Position of the Individual with Minimum cost value will be acquired.*
Step 3.3 $Pop_i \leftarrow CPop_{position}$*// the individual in the position in temporary population is moved to the population*
Step 4: Repeat through Step 5.6 **Until**$k \leq Popsize$, else goto Step 6 $where\ ER < k \leq PopSize$
Step 5: Choose the random parents Individuals,$P\_Indiv1\ and\ P\_Indiv2$
Step 5.1:Select the initial City $Init\_City,\ Length \leftarrow 1, Size \leftarrow 1$
Step 5.2: $Indiv[Length] \leftarrow Init\_City$*//the first city ofparent individual is the initial city in the offspring individual*
Step 5.3:Repeat through Step 5.5 **Until**$Length \leq n$, else goto Step 5
$Cur\_City \leftarrow Indiv[Length]$ *//the current city in the offspring individual assigned as current city*
Find the Position $Pos1\ and\ Pos2$ of the Current City in the Parent Individuals
$Pos1 \leftarrow find(P\_indiv1\ (Cur\_city)),\ \ Pos2 \leftarrow find(P\_indiv2\ (Cur\_city))$
IF$Pos1 = 1,\ LLoc1 \leftarrow n$
Else IF$Pos1 = n,\ \ RLoc1 \leftarrow 1,$
IF$Pos2 = 1\ , LLoc2 \leftarrow n$
Else IF$Pos2 = n\ , RLoc2 \leftarrow 1,$
Evaluate the Distances $d1, d2, d3, d4$ from Previous City to Current City and Current City to Next City from the Parent Individuals

$$d_1 \leftarrow DM(P\_Indiv1(LLoc1), P\_Indiv1(Pos1))$$
$$d_2 \leftarrow DM(P\_Indiv1(Pos1), P\_Indiv1(RLoc1))$$
$$d_3 \leftarrow DM(P\_Indiv1(LLoc2), P\_Indiv1(Pos2))$$
$$d_4 \leftarrow DM(P\_Indiv1(Pos2), P\_Indiv1(RLoc2))$$

Step 5.4: Repeat through Step 5.6**Until**$k < 4\ ,\ else\ goto\ Step\ 6\ \ where\ 0 < k \leq 4$
$LNext\_City \leftarrow min(d)$ *//the location of the city with minimum cost will be acquired*
$Next\_City \leftarrow Indiv(LNext\_city)$
Step 5.5:**IF**$Next\_City \notin Indiv\ and\ POLL_{Cur\_City\ ,Next\_City} < Max\_Poll$, else goto Step 4.2
$Length \leftarrow Length + 1$, Update $Indiv[Length] \leftarrow next\_City$
Step 5.6: $k \leftarrow k + 1$ *// goto the next individual*
Step 6: Generate $GeneC1 \leftarrow RAND\ (1, n,), GeneC2 \leftarrow RAND\ (1, n)$
Swap $Indiv(GeneC1) \leftarrow Indiv(GeneC2)$, Swap $Indiv(GeneC2) \leftarrow Indiv(GeneC1)$
Step 7:, $Pop_{size} \leftarrow Indiv\ , Size \leftarrow Size + 1$ *// goto the next individual*
Step 8: Evaluate the cost of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TC_i \leftarrow \sum_{j=1}^{n} DM(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1\ \equiv 1$$

Step 9: Evaluate the Pollution of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TP_i \leftarrow \sum_{j=1}^{n} POLL(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1\ \equiv 1$$

Step 10: $Gen \leftarrow Gen + 1$*//Current generation is completed, increment the Gen for next generation*
Step 11: Return Pop

4.3.2 Algorithm Explanation

The algorithm for Optimal Distance based routing has the following arguments; Pop is the initial population generated using random or heuristic technique, TC is the total cost of each individual in the initial population using Equation (7), G is the generation limit for termination of GA and n is the size of the problem instances. Elitism Rate ER is the number of high quality/elitist individuals are transferred from the current generation to the next without any modification. This elitism transfer technique avoids the replacement of best fit individuals with poor individuals in the successive generations and also improves the performance of crossover operation, if the parent is selected from the elitist individuals. Elitist individuals are identified by finding the individual with least value in the TC of the current population.

After the elitism transfer, the size of the next population would be ER and the remaining $PopSize - ER$ individuals arederived using the greedy crossover and swap mutation operators. First, two parent solutions $P\_Indiv1\ and\ P\_Indiv2$ are chosen randomly from the current population and the first city of the parents is copied as the first city of the offsprings, thus the $Length = 1$. The construction of a complete offspring $Indiv$ of length n using the greedy crossover is explained in the subsequent discussion:

The position of the current cityCur_City of the partially built offspring $Indiv$ in the two selected parents is identified using the following conditions,

$$Pos1 \leftarrow find\left(P_{indiv1}(Cur\_City)\right) \rightarrow 33$$

$$Pos2 \leftarrow find(P_{indiv2}(Cur\_City)) \rightarrow 34$$

The position of current city in the parents is used to identify the location of left $LLoc$ and right $RLoc$ adjacent cities of Cur_city in the concerned parent solutions and the corresponding location value can be acquired by following the following heuristic:

IF $Pos1 = 1$

$LLoc1 \leftarrow n, RLoc1 \leftarrow Pos1 + 1$

Else IF $Pos1 = n$

$LLoc1 \leftarrow Pos1 - 1, RLoc1 \leftarrow 1$

Else

$LLoc1 \leftarrow Pos1 - 1,\ RLoc1 \leftarrow Pos1 + 1$

IF $Pos2 = 1$

$LLoc2 \leftarrow n, RLoc2 \leftarrow Pos2 + 1$

Else IF $Pos2 = n$

$LLoc2 \leftarrow Pos2 - 1, RLoc2 \leftarrow 1$

Else

$LLoc2 \leftarrow Pos2 - 1, \qquad RLoc2 \leftarrow Pos2 + 1$

The location of adjacent cities in the parent solutions are used to find the city with the least distance from the Cur_City,

$$d_1 \leftarrow \ DM(P\_Indiv1(LLoc1), P\_Indiv1(Pos1)) \rightarrow 35$$

$$d_2 \leftarrow \ DM(P\_Indiv1(Pos1), P\_Indiv1(RLoc1)) \rightarrow 36$$

$$d_3 \leftarrow \ DM(P\_Indiv1(LLoc2), P\_Indiv1(Pos2)) \rightarrow 37$$

$$d_4 \leftarrow \ DM(P\_Indiv1(Pos2), P\_Indiv1(RLoc2)) \rightarrow 38$$

The least distance value among the four $d_1, d_2, d_3 and\ d_4$ is selected and the city at the corresponding location of the concerned parent is chosen as the next city $Next\_City$. The chosen city $Next\_City$ is verified for two conditions,

Condition 1: The chosen city should not present in the partially built offspring i.e. $Next\_City \notin Indiv$.

Condition 2: The pollution value between the current city Cur_City and the chosen next city $Next\_City$ should be within the maximum pollution limit$POLL_{Cur_City\ ,Next_City} < Max_Poll$.

If the chosen city satisfies both the conditions, it is added as the next city in the offspring $Indiv$ and the length of the offspring is incremented $Length \leftarrow Length + 1$, otherwise the city with next least distance is chosen and verified. If all the possible cities are checked, next city is added randomly. The same steps are repeated until the length of the offspring $Indiv1$ is n which indicates that the offspring is a feasible solution/route of n cities. The similar procedures are followed to construct the second offspring $Indiv2$. The swap mutation is applied at the resultant offspring's by exchanging the randomly chosen cities,

$$GeneC1 \leftarrow RAND\ (1, n,), GeneC2 \leftarrow RAND\ (1, n) \rightarrow 39$$

within the offspring as,

$$Indiv(GeneC1)\ \leftarrow Indiv(GeneC2)\ and\ Indiv(GeneC2)\ \leftarrow Indiv(GeneC1) \rightarrow 40$$

This stage confirms that the construction of offspring is completed and it is included in the next population and the size of the population is incremented $Size \leftarrow Size + 1$. The generation of next population Pop of individuals is said to be completed if the $Size = PopSize$ and the population generation is repeated for G number of times, then the execution stops. The final population is assessed for best solution in terms of distance and pollution using Eq. (9) and (10) respectively.

$$\forall i\ [1 \leq i \leq PopSize], TC_i \leftarrow \sum_{j=1}^{n} DM(Indiv_i(j), Indiv_i(j+1)), \quad n+1 \equiv 1 \rightarrow 41$$

$$\forall i\ [1, \leq i \leq PopSize], TP_i \leftarrow \sum_{j=1}^{n} POLL(Indiv_i(j), Indiv_i(j+1)), \quad n+1 \equiv 1 \rightarrow 42$$

The final population of individuals is considered for validation to bring out the best in at last.

4.4 EXPERIMENTAL ANALYSIS

As discussed in the previous sections, the experimental analyses have been carried out in two scenarios: Optimal Distance Based Routing in VRP and Air Pollution Based Optimal Routing in VRP. Each scenario has been evaluated w.r.t. three population seeding techniques: random, nearest neighbor and ODV based EV method. The performance evaluation of the proposed intelligent routing strategies for VRP is performed in the following sections.

4.4.1 Optimal Distance Based Routing in VRP

The optimal Distance Based Routing in VRP, as discussed in the Section 3, is intended to find the optimal route based on "the total distance of the route" as a primary factor. The total air pollution of the derived route is considered as the secondary factor for the optimal solution selection. In each of the performance criteria associated with this scenario, the cost refers to the total distance of the solution obtained.

4.4.2 Experiments

In this scenario of experiments, the intelligent routing in VRP has been performed by optimizing the total distance of the route and the total air pollution is also noted for the corresponding optimized route. Experimental results for the scenario 1 of analyses with random, nearest neighbor and ODV based EV population seeding techniques are shown in the Table 4.4.2. From the Table 4.4.2, the following observations can be made:

Observation 1: The computation time increases based on the problem instances, each technique has its own computation time for every problem instances. For every problem instance, the computation time for ODV-EV technique is lesser than the other population seeding techniques considered. The computation time for ODV-EV technique ranges from 10.800 to 56.210 sec for the instance 16 cities to 100 cities. The minimum computation time of NN technique is same as the ODV technique and the maximum computation time is 57.680. The random technique ranges from 10.660 to 60.820 sec.

Observation 2: For all the problem instances, the ODV-EV population seeding technique yields better convergence rate % for both the best and the worst individual in the population. The maximum and minimum convergence rates obtained are 97.38% for uysses16 using the ODV-EV technique and 59.26% for the instance KroA100 using random technique respectively.

Observation 3: it is observed from the result that the worst convergence rate or the worst individuals in the population of ODV-EV technique showed better performance. In the worst convergence rate, the maximum and minimum convergence rate are obtained in ODV-EV and Random technique with 48.110% and -37.546. Each technique has its own maximum and minimum values for the worst convergence, the ODV-EV technique showed maximum of 48.110 for the instance swiss42 and minimum of -7.752% for the instance bays29. The output of NN technique has revealed maximum of 39.057 in swiss42 and minimum of -

14.317 in bays29. The result of random technique acquired higher worst convergence rate for many instances while compared to the NN technique, the random technique has maximum of 43.420 in eil51 and minimum of -37.546 in KroA100.

Observation 4: The error rate of the best individuals in the population increases randomly with an increase in the size of the problem instance despite the population technique used. Since the ODV-EV technique yielded good result for the best convergence rate, it is obvious that the best error rate lesser for ODV-EV technique. It is observed from the figure 4.4.2, the ODV-EV technique has the least value as 2.617 for the instance uysses16 and the highest value as 14.802 for the instance bays29. Whereas NN and random techniques has had maximum of 29.806% and 40.738% in terms of error rate for the best individuals in the population respectively for the instance KroA100. The minimum error rate obtain from the best individuals are 3.585 % in NN technique for the instance uysses16 and 3.084 % in random technique for the instance uysses26.

Observation 5: It is understandable that the error rate of the worst individuals in the population is contradictory to the convergence rate of the worst individuals in the population. The performance of ODV-EV technique is better than the NN and random technique for all the instances. The worst error rate of the worst individuals in the population ranges from 51.890% to 137.546% for the instance swiss42 in ODV-EV and KroA100 and random technique. Each technique has its own minimum and maximum error rate for different instances; however the minimum value of worst error rate for all the techniques have been acquired from the instance swiss42.

Observation 6: Population Diversity is considered as the primary reason for premature convergence. Premature convergence occurs when the population of a GA reaches such a suboptimal state that the genetic operators can no longer produce offspring that outperform their parents. For all the instances, the ODV-

EV technique has lesser convergence diversity w.r.t. other population seeding techniques which shows that the quality of individuals is improved as a population rather than the single individual. The convergence diversity of all the techniques consumes higher values for the instance bays29, whereas random technique showed extreme result of 106.990.

Observation 7: For most of the instances, the ODV-EV technique has at least 50% of average convergence followed by NN technique which has between 30% and 50%. The maximum and minimum convergence rates obtained are 66.92% for the instance swiss42 using ODV-EV technique and 5.22% using the random technique for the instance bays29 respectively. Similar trends were also observed in NN technique has the maximum of 66.260% in swiss42 and minimum of 11.912% in bays29. This result clearly shows that the average convergence for the instance bays29 obtained lesser value for all techniques.

Observation 8: from the figure 4.4.2, the data clearly indicate that the average error rate is contradictory to the average convergence rate, it is evidentially proves that the bays29 has higher values for all the techniques. The average error rate ranges from 33.074 to 94.740 for the instances swiss42 and bays29. It is also pertinent to suggest that, the NN technique showed a reasonable values fluctuates from 33.740 to 88.088.

Observation 9: The ODV-EV technique performs better than NN and random techniques in terms of pollution based convergence rate % for the distance optimized routes. The maximum pollution based convergence rate obtained is 55.37% for the instance eil76 using ODV-EV technique.

Instance			Optimal Solution	Computation Time	Quality of the Solution			Convergence Rate (%)		Error Rate (%)		Convergence Diversity	Average Convergence
					Best	Worst	Average	Best	Worst	Best	Worst		
uysses16	EV	Pollution	2.5596	10.800	5.078	7.148	6.400	1.627	-79.277	98.373	179.277	80.904	-50.027
		distance	74.1087		76.048	124.436	110.259	97.383	32.091	2.617	67.909	65.293	51.219
	NN	Pollution	2.5596	10.800	5.657	7.866	6.623	-21.007	-107.331	121.007	207.331	86.324	-58.743
		distance	74.1087		76.766	133.315	115.263	96.415	20.109	3.585	79.891	76.306	44.468
	Random	Pollution	2.5596	10.660	4.875	7.614	6.417	9.523	-97.483	90.477	197.483	107.006	-50.692
		distance	74.1087		76.394	125.972	108.641	96.916	30.017	3.084	69.983	66.899	53.403
uysses22	EV	Pollution	3.194	16.420	6.113	8.852	8.181	8.612	-77.130	91.388	177.130	85.742	-56.126
		distance	75.6615		80.084	144.880	127.428	94.154	8.515	5.846	91.485	85.639	31.581
	NN	Pollution	3.194	16.670	7.133	9.970	8.363	-23.323	-112.143	123.323	212.143	88.820	-61.822
		distance	75.6615		83.463	151.628	126.768	89.689	-0.403	10.311	100.403	90.092	32.454
	Random	Pollution	3.194	16.550	6.637	10.140	8.508	-7.810	-117.463	107.810	217.463	109.653	-66.380
		distance	75.6615		83.814	151.434	128.305	89.225	-0.147	10.775	100.147	89.372	30.423
bays29	EV	Pollution	5.1614	20.650	8.996	12.061	10.972	25.697	-33.678	74.303	133.678	59.375	-12.584
		distance	2020		2319.000	4196.600	3791.736	85.198	-7.752	14.802	107.752	92.950	12.290
	NN	Pollution	5.1614	22.070	9.033	12.559	11.029	24.999	-43.324	75.001	143.324	68.322	-13.678
		distance	2020		2420.000	4329.200	3799.374	80.198	-14.317	19.802	114.317	94.515	11.912
	Random	Pollution	5.1614	21.910	9.623	12.233	11.303	13.553	-37.002	86.447	137.002	50.554	-18.986
		distance	2020		2558.200	4719.400	3933.750	73.356	-33.634	26.644	133.634	106.990	5.260
swiss42	EV	Pollution	6.2613	22.250	11.045	12.351	11.793	23.606	2.743	76.394	97.257	20.863	11.657
		distance	1273		1324.222	1933.556	1694.034	95.976	48.110	4.024	51.890	47.866	66.926
	NN	Pollution	6.2613	22.500	12.229	13.487	12.828	4.682	-15.406	95.318	115.406	20.088	-4.873
		distance	1273		1339.600	2048.800	1702.510	94.768	39.057	5.232	60.943	55.711	66.260
	Random	Pollution	6.2613	22.820	12.005	13.402	12.625	8.267	-14.040	91.733	114.040	22.307	-1.630
		distance	1273		1365.000	2021.800	1758.182	92.773	41.178	7.227	58.822	51.595	61.887
EIL51	EV	Pollution	7.6588	28.750	12.343	14.074	13.541	38.844	16.237	61.156	83.763	22.607	23.199
		distance	426		442.062	833.087	603.879	96.230	4.440	3.770	95.560	91.790	58.244
	NN	Pollution	7.6588	29.980	12.825	14.488	13.752	32.551	10.827	67.449	89.173	21.724	20.441
		distance	426		464.658	699.113	579.155	90.925	35.889	9.075	64.111	55.037	64.048
	Random	Pollution	7.6588	28.860	13.397	15.323	14.425	25.079	-0.067	74.921	100.067	25.146	11.649
		distance	426		454.848	667.031	585.117	93.228	43.420	6.772	56.580	49.808	62.649
eil76	EV	Pollution	11.3454	42.980	16.408	19.302	18.258	55.374	29.869	44.626	70.131	25.504	39.069
		distance	538		612.614	908.158	769.134	86.131	31.197	13.869	68.803	54.934	57.038
	NN	Pollution	11.3454	43.940	19.670	21.621	20.778	26.624	9.430	73.376	90.570	17.194	16.861
		distance	538		636.223	928.683	800.029	81.743	27.382	18.257	72.618	54.361	51.296
	Random	Pollution	11.3454	46.100	23.275	25.538	24.475	-5.151	-25.095	105.151	125.095	19.943	-15.729
		distance	538		707.045	1025.886	904.237	68.579	9.315	31.421	90.685	59.264	31.926
kroA100	EV	Pollution	14.5057	56.210	23.539	25.755	24.956	37.728	22.449	62.272	77.551	15.279	27.956
		distance	21285		24432.547	37271.678	33334.141	85.212	24.892	14.788	75.108	60.320	43.391
	NN	Pollution	14.5057	57.680	26.727	28.518	27.700	15.748	3.404	84.252	96.596	12.345	9.043
		distance	21285		27629.297	45023.254	35956.651	70.194	-11.526	29.806	111.526	81.719	31.070
	Random	Pollution	14.5057	60.820	29.892	31.921	30.866	-6.069	-20.058	106.069	120.058	13.990	-12.789
		distance	21285		29955.978	50561.596	40854.846	59.262	-37.546	40.738	137.546	96.808	8.058

Table.4.4.2 Experimental results of optimal distance based routing

Observation 10: The ODV-EV population seeding yields better pollution based average convergence rate than the NN and random techniques. The negative average convergence rate obtained for instances uysses16, uysses22 and bays29 suggests that the resultant population is a collection of low quality solutions.

4.4.3 Result Analyses

Computational Time: As shown in the Fig.4.4.3(h) expressively proves that the computation time increases based on the problem instances, each technique has its identifiable computation time for every problem instances. In terms of computation time, it is obvious that the random technique showed good result in classical TSP or any other problem. In this case, each technique must validate the pollution between the corresponding cities before adding the next city. Hence, the computation time of each technique for different instance has had minor changes. Furthermore, analyzed from the Fig.4.4.3(h) the random technique has showed an unbalanced change for the problem instances, for all of the instances the ODV-EV technique performs better than NN and random technique. The performance of random technique has got acceptable computation time for the smaller size instance. The NN technique has showed increase in decrease while moved towards the smaller size instances to higher size instances.

Analysis on Convergence Rate w.r.t distance: Convergence of an individual refers to the quality of the individual w.r.t. the known optimal distance quality for the problem as given in Eq. 20. The convergence of individual in the population can be assessed in ways, the best and worst convergence rate; the best convergence rate (%) refers to the highest distance based fitness attained by an individual in the population and similarly, worst convergence rate (%) refers to the lowest distance based fitness attained by an individual in the population. Figs. 4.4.3(a) and 4.4.3(c) show the best and worst convergence rates (%) attained using different population seeding techniques for the problem instances. From the Figs. 4.4.3(a) and 4.4.3(c), it can be observed that ODV-EV technique has high convergence rate % in terms of both best and worst individuals in the population than the random and NN techniques. The NN technique performs better than random but the performance of both the techniques diminishes with increase in the size of the problem instances.

Analysis on Error Rate w.r.t distance: The error rate of an individual refers to the percentage of difference in the quality of the individual w.r.t. the known optimal quality for the problem as given in Eq. 25. The error rate of individual at the population can also be evaluated in ways, similar to the convergence rate, the best and worst error rate; the best error rate (%) refers to the error rate attained by the highest fit individual, based on the distance factor, in the population and similarly, worst error rate (%) refers to the lowest distance based fitness of the individual in the population. Figs.4.4.3(b) and 4.4.3(d) show the best and worst error rates (%) attained using different population seeding techniques for the problem instances. From the Figs.4.4.3(b) and 4.4.3(d), it can be observed that the ODV-EV technique performs excellent for both best and worst error rate %, closely followed by the NN technique in all the test instances. The performance of both random and NN techniques degrades with increase in the size of the problem instances.

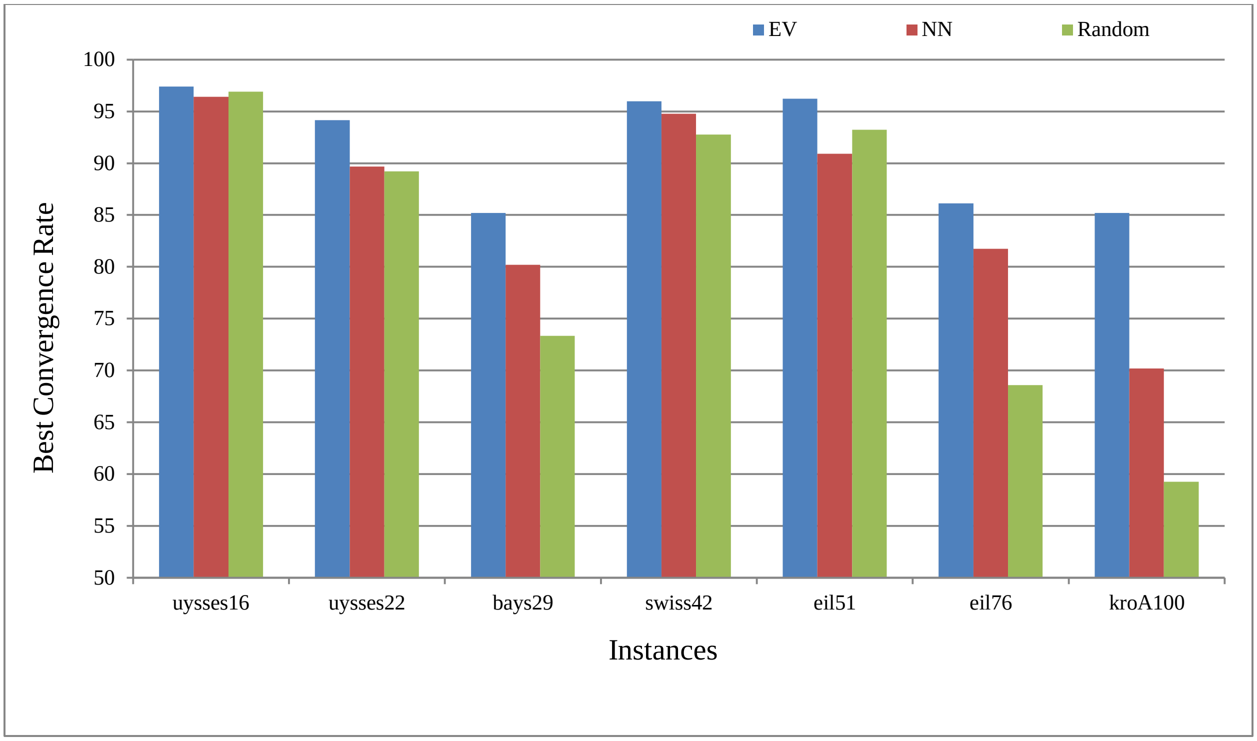


Fig.4.4.3 (a) Best convergence rate for optimal distance based routing

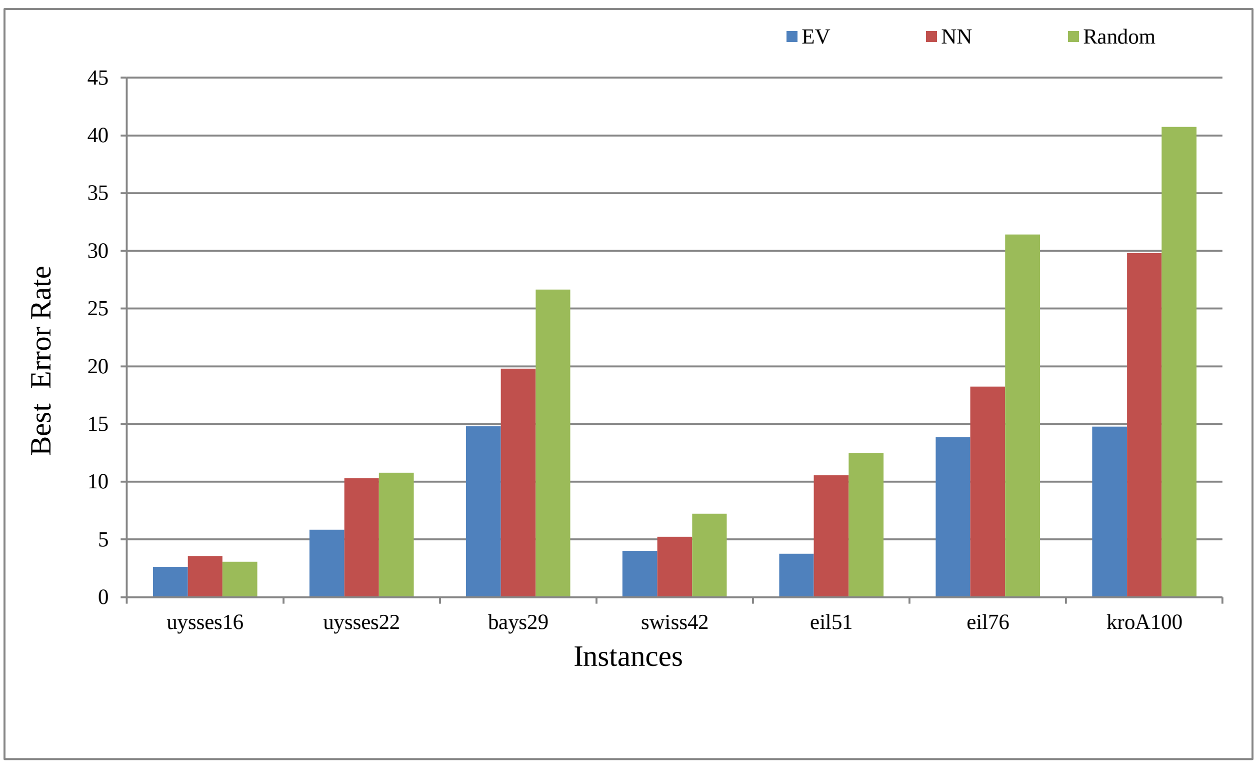


Fig.4.4.3(b) Best error rate for optimal distance based routing

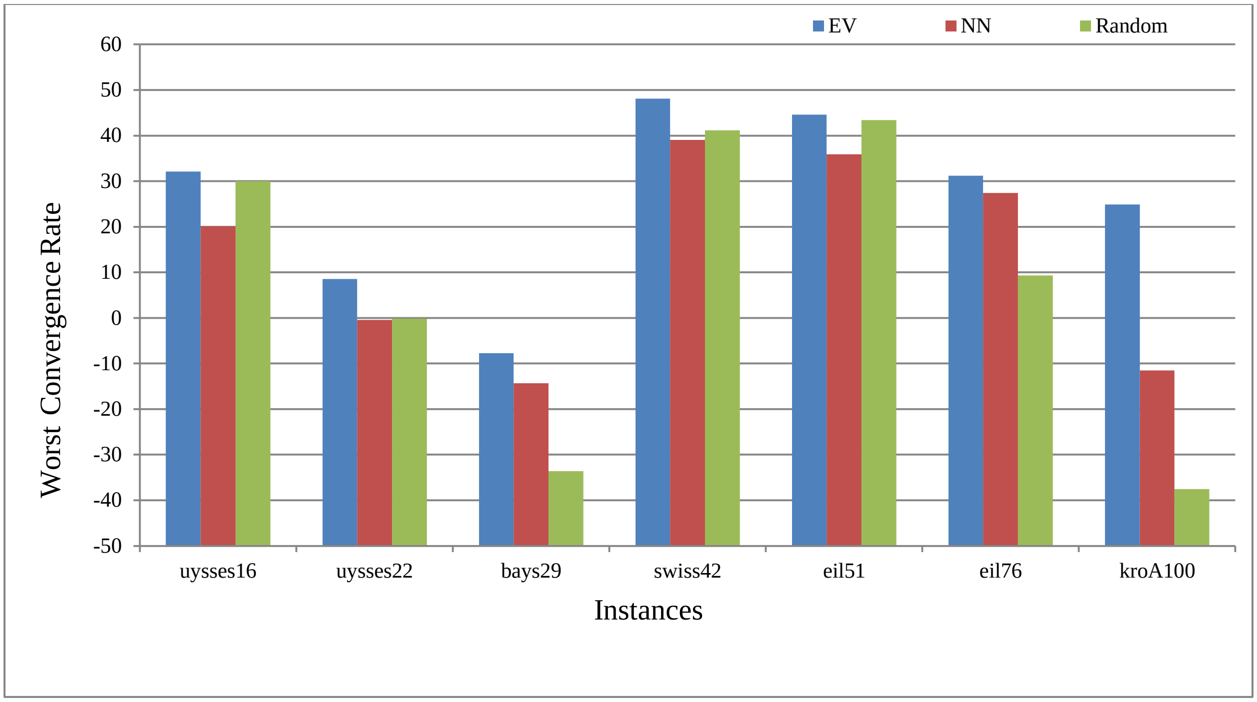


Fig.4.4.3(c)Worst convergence rate for optimal distance based routing

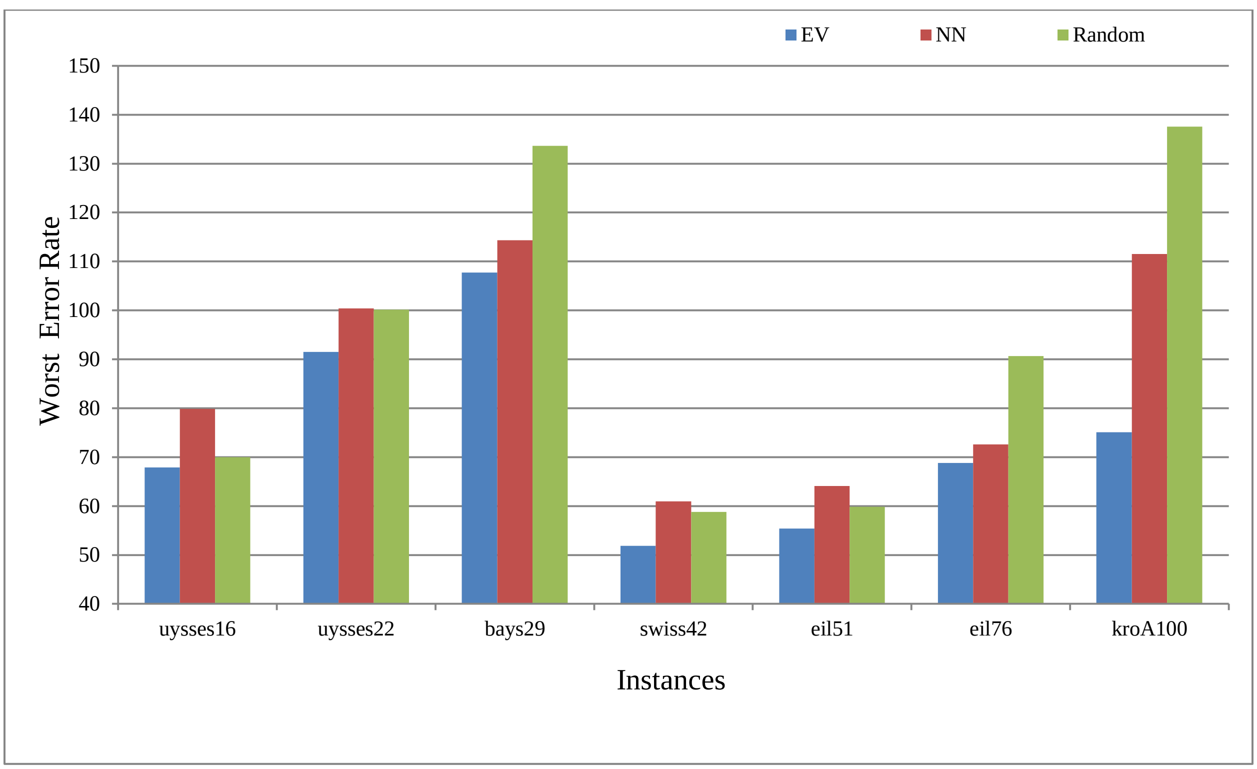


Fig. 4.4.3(d) Worst error rate for optimal distance based routing

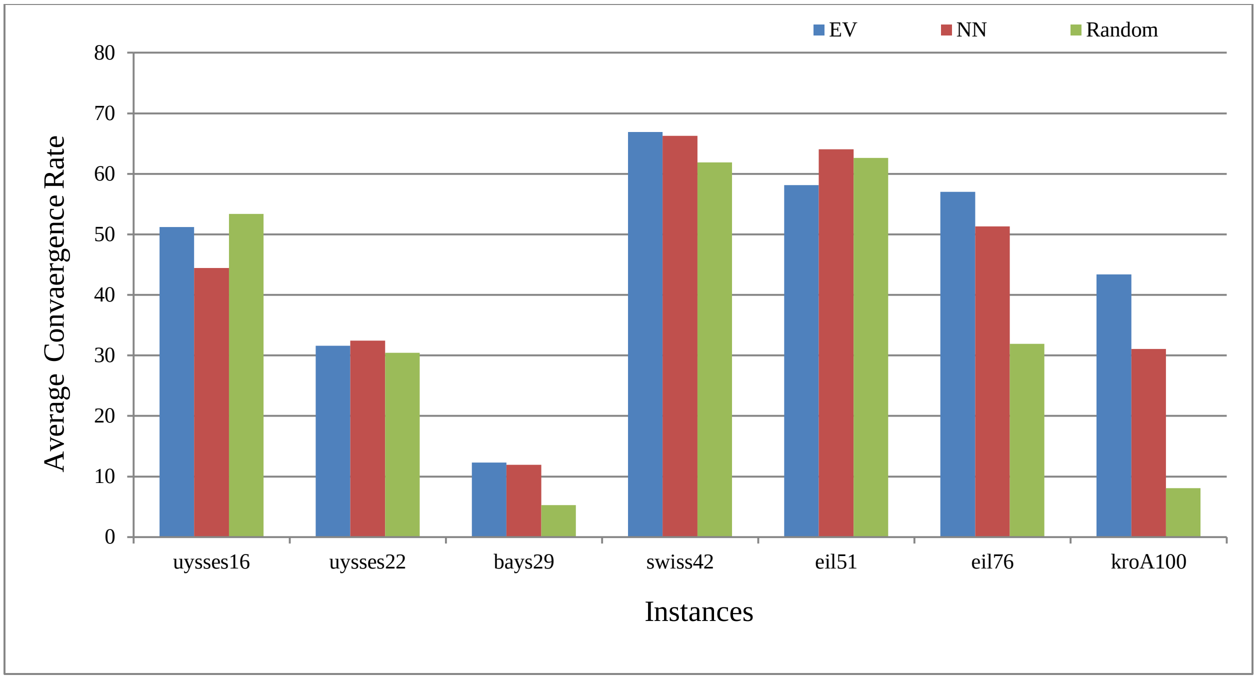


Fig. 4.4.3(e) Average convergence rate for optimal distance based routing

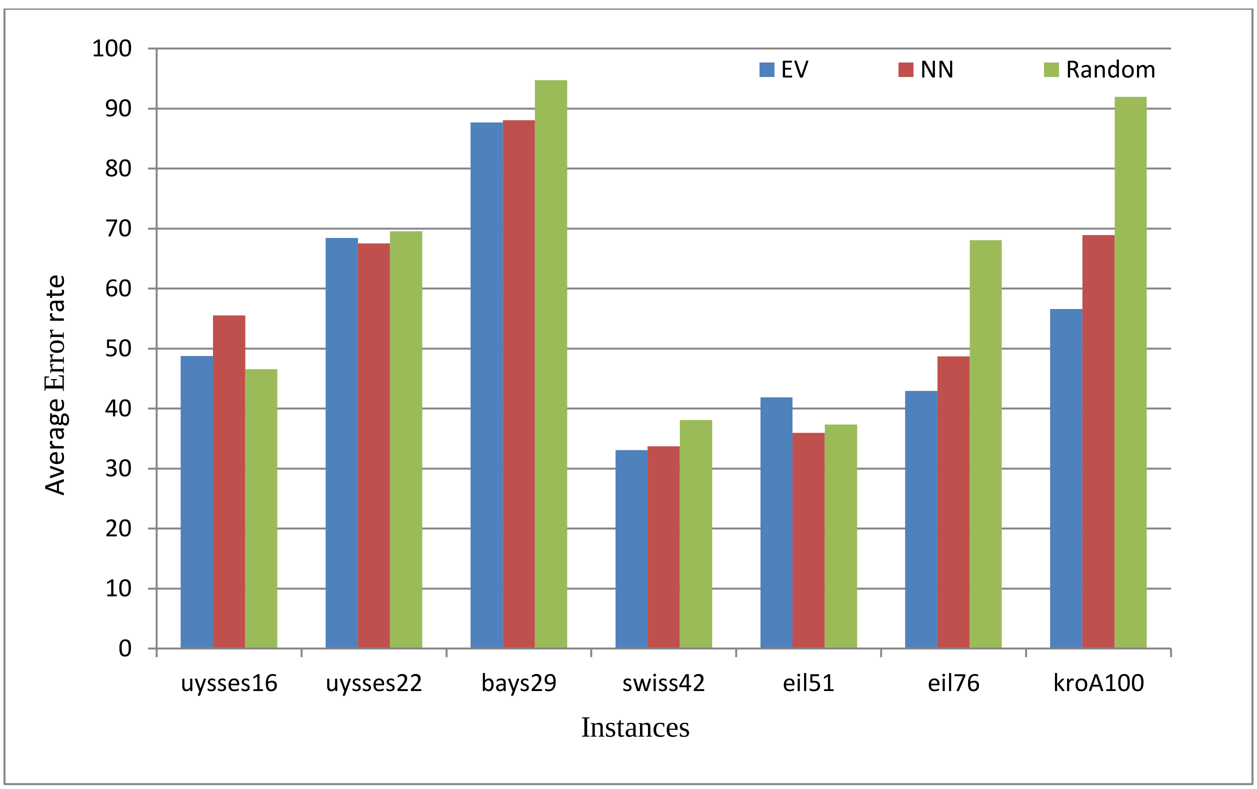


Fig. 4.4.3(f)Average error rate for optimal distance based routing

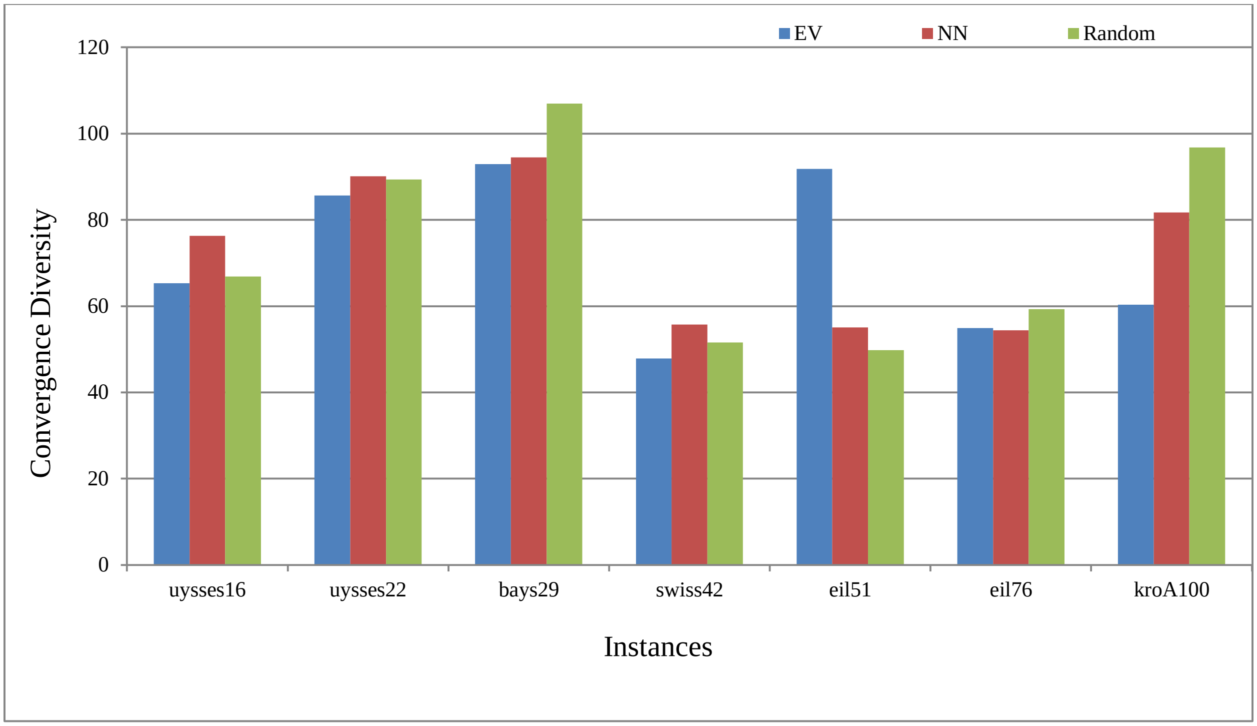


Fig. 4.4.3(g)Convergence diversity for optimal distance based routing

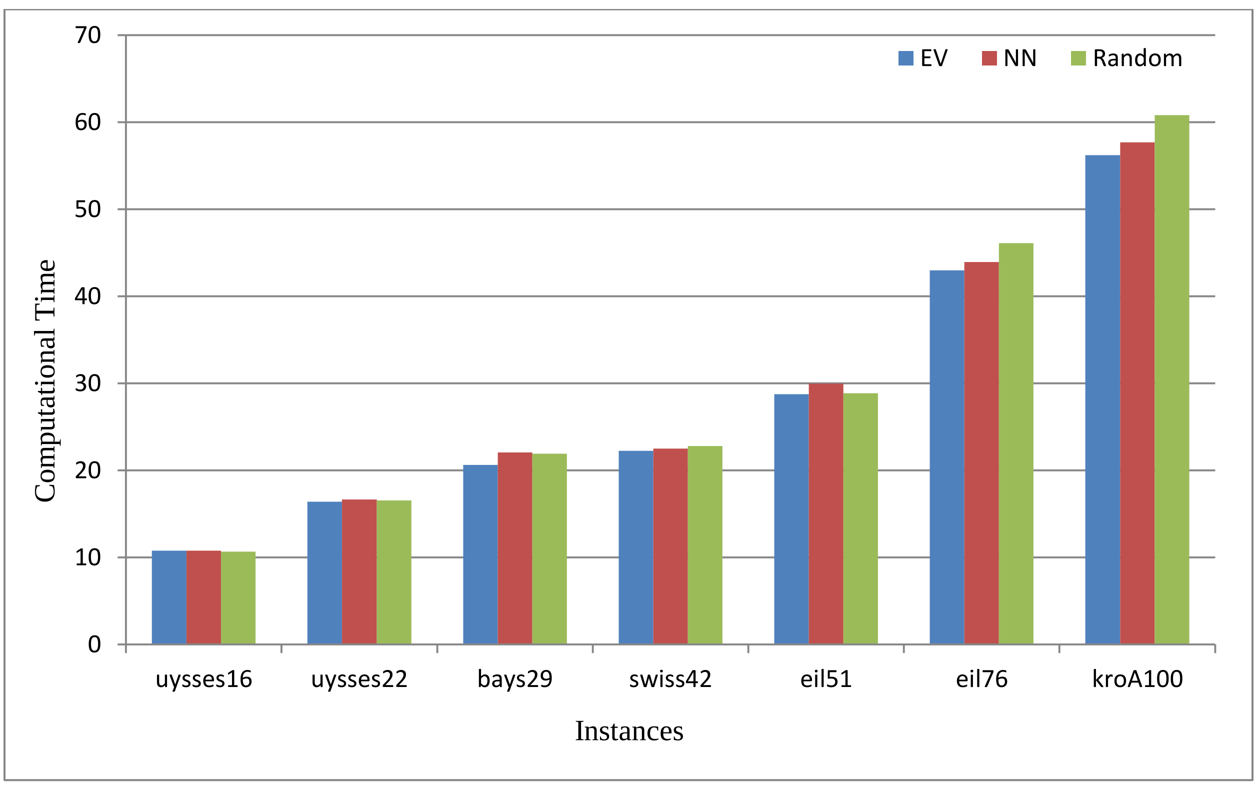


Fig. 4.4.3(h)Computational Time for optimal distance based routing

Analysis on Average Convergence w.r.t distance: The Average convergence of a population is used to measure the quality of the population generated by finding the average of fitness of individuals in the population as given in Eq. 21. Fig. 4.4.3(e) shows the average convergence rate for optimal distance based routing using different population seeding techniques for the problem instances. From the Fig. 4.4.3(e) it can be observed that every population seeding technique yields better average convergence rate for some of the large size problem instances than the small size instances. For most of the instances, the ODV-EV technique outperforms other population initialization techniques and random performs worst for the larger size instances. For the instance bays29, performance of random, NN and ODV-EV techniques are very poor; this possibly because of the peculiarity of the instance with small size and large distance based fitness value.

Analysis on Convergence diversity w.r.t distance: The convergence diversity is a factor that explicates the distribution of good and bad quality individuals among the population. It plays a critical role to increase the chance of evolving optimal solutions and to avoid premature convergence. Fig. 4.4.3(g) shows the convergence diversity of the optimal distance based routing scenario using different population seeding techniques for the problem instances. From the Fig. 4.4.3(g), it is understood that the ODV-EV technique has lesser convergence diversity w.r.t. other population seeding techniques which shows that the quality of individuals is improved as a population rather than the single individual. For most of the instances, random and NN techniques have nearly equal convergence diversity.

4.5 SUMMARY

In summary this chapter describes the first module of the research that is Optimal Distance Based Routing in TSP. This chapter mainly focused on to find the optimal route based on "the total distance of the route" as a primary factor. The total air pollution of the derived route is considered as the secondary factor for the optimal solution selection. Heuristic method is used to initialize the initial population for the problem that is NN, ODV-EV, Random population seeding technique and in each generation the quality of individuals was improved in terms of distance of the route. The greedy crossover and swap mutation technique is used to maintain the best parent individual or route in every generation. From the table 4.4.2 it is observed that in all instances ODV-EV technique performs better than that of NN and Random respectively. So it is concluded that our proposed GA performs better.

CHAPTER - 5

SCENARIO-2: OPTIMAL POLLUTION BASED ROUTING IN VRP

5.1 INTRODUCTION

Optimal pollution based routing in VRP is supposed to seek out the efficient path supported "the whole pollution of the path" by way of the main issue. The optimal result is selected based on the entire distance of the resulting route which is taken as a secondary factor. In chapter 4, by heuristic method or random methodology in every generation, the value of individuals is improved by pollution and initial population has also been created. To decrease the entire pollution of the root path in every generation, the mutation operator and crossover operator are changed toward the pollution.

In optimum pollution based Routing, the issue is focused on a distinct objective called as pollution and to minimize the cost of the path. The selection process is started by computing the cost of the paths or individuals exist in the population. The individuals which have lowest pollution costs are chosen by Elitism Rate (ER) and lastly followed by mutation process and crossover process. Two arbitrary individuals are chosen as parent individuals and initiated to create the offspring individual using the crossover process. The locations of the present individuals are calculated and the minimal polluted city which has low pollution cost is relocated to the offspring individual. The procedure is continued up to the entire offspring individual has created, and mutation process continues. Likewise, in the mutation process, two arbitrary cities are chosen among the offspring individual and at that moment, the cities are relocated. At that point, the current offspring individuals are repositioned to the new population, and the entire process continues till it attains the end state.

5.2 PROBLEM DESCRIPTION AND FORMULATION

In this process, VRP is anticipated in an alternative view, which is explored to suggest a novel atmosphere. It concerns over the issue of transportation; the best path must be of low distance and to minimize the pollution alongside by the path. The pollution among the two cities is calculated using TSP by pollution matrix. The pollution matrix is identical to the matrix which is calculated for distance. The permissible pollution among the two cities is the extreme allowable value of pollution in best solution for the issue. While calculating the results, at every step, the insertion of a new city is permitted if the value of pollution among the prior and the new city is lesser than the maximum permitted pollution range between the cities, otherwise it should choose the another city. The efficient routing approach for VRP using Optimal Pollution Based Routing is signified as follows.

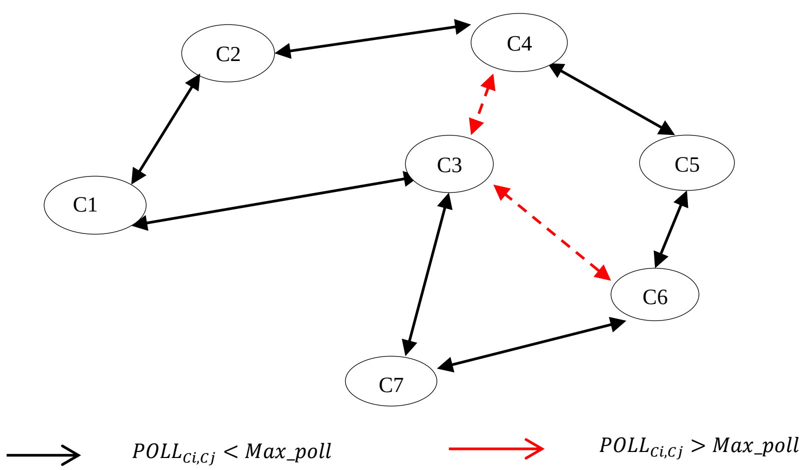


Fig.5.2. Sample intelligent routing strategy for VRP w.r.t Pollution

Let, the whole undirected graph $G = \{City_n, A\}$ and $DM(City_i, City_j)$ is the distance between the cities $City_i$ and $City_j$ such that $City_i \neq City_j$ and $DM(City_i, City_j) = DM(City_j, City_i)$. The pollution matrix for the TSP problem of size 'n'can be represented as $POLL(n \times n)$and $POLL(City_i, City_j)$ is the pollution between the $City_i$ and the $City_j$. In the proposed work, the initiating city must be similar to the reason that the vehicle had to initiate from the fixed source. As we discussed in the chapter 4, the same IRS standard is followed in this optimal pollution based routing scenario. The main working of the IRS is described with an example as follows. Let us consider the IRS for VRP with the size of 7 cities $(C1, C2, C3 \ldots. C7)$ as shown in the Figure 5.1. In the Figure 5.1, the red dashed line shows the air-pollution is higher than the pollution limit and the black line shows the pollution rate is normal which may be consider for optimal solution selection. Assume that $C1$ is the starting city and the neighboring cities are organized in ascending order of their distance such that $(d(C1, C2) \leq d(C1, C3) \leq \cdots \leq d(C1, C7))$.

The main aim of intelligent routing strategy for VRP in Optimal Pollution Based Routing is to identify the neighboring city of less pollution and also to identify whether the pollution is inside the limit between the cities. Initiating from the city $C1$, the nearby city called as adjacent city is 2 , the $C2$ has the less pollution among the two cities $C1$ and $C2$ goes over the pollution limit. i.e. $POLL_{C1,C2} < Max\_poll$. Therefore, the insertion of city $C2$ adjacent to the city $C1$ along the route is canceled, so the subsequent city of $C1$ city $C3$ is selected. The available routes are $(C1, C3, C2, C4, C6, C5, C7, C1)$ and the similar process is repeated up to the entire route through the $n$number of cities are generated. To reduce the pollution of the individuals, is the main objective of intelligent routing strategy for VRP in Optimal pollution Based Routing in the population. The individuals within the population do not guarantee the best solution to the issue for pollution. The genetic procedures are applied to improve the individuals within the population; so various procedures must be adopted.

5.3 ALGORITHM DEVELOPMENT

The intelligent routing strategy for VRP is proposed by genetic algorithm for VRP in Optimal Pollution Based Routing. The key objective is to reduce the pollution using the Optimal Pollution Based Routing and to fulfill the main conditions. The pollution among the two cities must be lesser than the maximum pollution limit. The genetic procedures are applied to enhance the efficiency of the pollution based routing of the population in every generation by pollution only.

In Optimal Pollution Based Routing, we are exploring this as a single objective, because the goal is to minimize the pollution. We achieved the optimal pollution base routing depending on the pollution from the initial population. The selection and the crossover operation have been performed depending on the minimum pollution. In selection process, the elitist individuals have been selected from the population.

$$TP_j = \begin{vmatrix} \mathrm{TP}_1 \\ \mathrm{TP}_2 \\ \mathrm{TP}_3 \\ \cdots \\ \mathrm{TP}_{PopSize} \end{vmatrix} = \begin{vmatrix} POLL(\mathrm{Indiv}_1) \\ POLL(\mathrm{Indiv}_2) \\ POLL(\mathrm{Indiv}_3) \\ \cdots \\ POLL(\mathrm{Indiv}_{PopSize}) \end{vmatrix} = \begin{vmatrix} \mathrm{Indiv}_1 \\ \mathrm{Indiv}_2 \\ \mathrm{Indiv}_3 \\ \cdots \\ \mathrm{Indiv}_{PopSize} \end{vmatrix}$$
$$= \begin{vmatrix} City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \end{vmatrix} = CPOP_{PopSize\times n} \qquad \rightarrow 43$$

TP_j represent the total pollution of each individual in the population. $CPOP_{PopSize\times n}$ is the current population, which consists of possible solutions. Each tour or possible solution in the population is represented as individual $Indiv_j$ the individuals having best fitness or minimum tour pollution are selected and send to the next generation.

$$POP_{ER\times n} = \begin{vmatrix} City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \end{vmatrix} = \begin{vmatrix} \mathrm{Indiv}_1 \\ \mathrm{Indiv}_2 \\ \mathrm{Indiv}_3 \\ \cdots \\ \mathrm{Indiv}_{ER} \end{vmatrix} = \begin{vmatrix} \mathrm{TP}_1 \\ \mathrm{TP}_2 \\ \mathrm{TP}_3 \\ \cdots \\ \mathrm{TP}_{ER} \end{vmatrix}$$

$$= \min(TP_{ER}) \qquad \rightarrow 44$$

Where ER is the Elitism Rate, depending on that the number of elitist individuals are selected and passed to next generation population POP. $TP_1, TP_2, TP_3, \dots TP_{ER}$ are the minimum pollution of each individual arranged in the increasing order and $Indiv_1, Indiv_2, Indiv_3, \dots Indiv_{ER}$ are the corresponding individuals to the minimum pollution.

The crossover is done through greedy crossover. From the total population two random individuals are selected as parent individual. All the individuals should have the same starting city, so the starting city is the initial city of the offspring. Find the position of the current city in both the parent individuals $Pos1 \leftarrow find(P\_indiv1\ (Cur\_city))$, $Pos2 \leftarrow find(P\_indiv2\ (Cur\_city))$ and then locate the right side and left side city to the current city in both the parent individuals. If the position of current city is the starting city then location of left city to the current city is last city $IF\ Pos1 = 1\ then\ LLoc1 \leftarrow n$ and if the position of current city is last city then location of Right city to the current city is last city $IF\ Pos1 = n\ then\ RLoc1 \leftarrow 1$. Check the cities in the locations are less than the maximum pollution $POLL_{Cur\_City\ ,Next\_City} < Max_{Poll}$.

5.3.1 Algorithm

<u>Optimal Pollution based routing Algorithm (</u>Pop, TP, G, n)

Step 1: Initialize $Gen \leftarrow 1, i \leftarrow 0, k \leftarrow 0$

Step 2:Storethe Population into a temporary variable, $CPop_{n\times n} \leftarrow Pop_{n\times n}$,

Step 2: Repeat through Step 10 **Until**$Gen \leq G$, repeat through Step 3 else go to Step 12

Step 3: Select the best individual which is having minimum Pollution

Step 3.1: Repeat through Step 3.3 Until $i < ER$, else goto Step 4

Step 3.2: $position \leftarrow \min(TP)$*//Position of the Individual with Minimum Pollution value will be acquired.*

Step 3.3 $Pop_i \leftarrow CPop_{position}$*// the individual in the position in temporary population is moved to the population*

Step 4: Repeat through Step 5.6 **Until**$k \leq PopSize$, else goto Step 6 $where\ ER < k \leq PopSize$

Step 5: Choose the random parents Individuals,$P\_Indiv1\ and\ P\_Indiv2$

Step 5.1:Select the initial City $Init\_City$, $Length \leftarrow 1, Size \leftarrow 1$

Step 5.2: $Indiv[Length] \leftarrow Init\_City$*//the first city ofparent individual is the initial city in the offspring individual*

Step 5.3:Repeat through Step 5.5 **Until**$Length \leq n$, else goto Step 5

$Cur\_City \leftarrow Indiv[Length]$ *//the current city in the offspring individual assigned as current city*

Find the Position $Pos1\ and\ Pos2$ of the Current City in the Parent Individuals

$Pos1 \leftarrow find(P\_indiv1\ (Cur\_city))$, $Pos2 \leftarrow find(P\_indiv2\ (Cur\_city))$

IF$Pos1 = 1$, $LLoc1 \leftarrow n$

Else IF$Pos1 = n$, $RLoc1 \leftarrow 1$

IF$Pos2 = 1$, $LLoc2 \leftarrow n$

Else IF$Pos2 = n$, $RLoc2 \leftarrow 1$

Evaluate the Pollutionp_1, p_1, p_1, from Previous City to Current City and Current City to Next City from the Parent Individuals

$p_1 \leftarrow POLL(P\_Indiv1(Pos1-1), P\_Indiv1(Pos1))$

$p_2 \leftarrow POLL(P\_Indiv1(Pos1), P\_Indiv1(Pos1+1))$

$p_3 \leftarrow POLL(P\_Indiv2(Pos2-1), P\_Indiv2(Pos2))$

$p_4 \leftarrow POLL(P\_Indiv2(Pos2), P\_Indiv2(Pos2+1))$

Step 5.4: Repeat through Step 5.6 **Until**$k < 4$, $else\ goto\ Step\ 6\ \ where\ 0 < k \leq 4$

$Next\_City \leftarrow min(p)$ *//the location of the city with minimum pollution will be acquired*

Step 5.5:**IF**$Next\_City \notin Indiv\ and\ POLL_{Cur\_City\ ,Next\_City} < Max\_Poll$, else goto Step 4.2

$Length \leftarrow Length + 1$, Update$Indiv[Length] = next_City$

Step 5.6: $k \leftarrow k + 1$ *// increment the individual in the population*

Step 6: Generate Random values $GeneC1, GeneC2$, $where\ 0 < GeneC1, GeneC2 \leq n$

Swap $Indiv(GeneC1) \leftarrow Indiv(GeneC2)$, Swap$Indiv(GeneC2) \leftarrow Indiv(GeneC1)$

Step 7: $Pop_{size} \leftarrow Indiv, Size \leftarrow Size + 1$*//goto next individual*

Step 8: Evaluate the cost of each Individual in the Population

$\forall[1, \leq i \leq PopSize], TC_i \leftarrow \sum_{j=1}^{n} DM(Indiv_i(j)\ , Indiv_i(j+1)\ )$, $j+1 \equiv 1$-

Step 9: Evaluate the Pollution of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TP_i \leftarrow \sum_{j=1}^{n} POLL(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1 \equiv 1$$

Step 10: $Gen \leftarrow Gen + 1$*//Current generation is completed, increment the Gen for next generation*

Step 11: Return Pop

Fig.5.3.1. Algorithm for Optimal Pollution Based Routing

Based on the minimum pollution the cities are added in offspring individual, after generated the complete individual move to the Mutation process, in that two random locations are generated and then swap the offspring individual cities in the locations and vice versa using equation (31) and (32). The offspring's are added into the population, Stop this process until we reach the total population.

5.3.2 Algorithm Explanation

Air Pollution Based Optimal Routing in VRP is proposed to identify the best route, based on "the total air pollution TP in the route" as a primary issue. The whole distance of the resultant path is taken as a secondary issue for the best solution selection. The initial population Pop for the scenario 2 is generated, which is similar to scenario 1, using random method or heuristic method and the value of individuals in terms of air pollution of the route is enhanced in each generation. The overall air pollution is minimized using crossover operation in each generation of the parent route. The algorithm for optimal pollution based routing is presented in the Fig. 5.3.1

The Scenario 2 works similar to Scenario 1 in elitism transfer, parent selection, position identification of current city in the parent solutions and mutation operations. But in this case, elitist individuals are chosen based on the air pollution value and in the offspring construction stage, the next city is decided based on the air pollution value between the current city and the cities identified in the parent solutions.

The algorithm for Optimal pollution based routing has the following arguments; Pop is the initial population generated using random or heuristic technique, TP is the total pollution cost of each individual in the initial population using Equation (30), G is the generation limit for termination of GA and n is the size of the problem instance.

Elitism Rate ER is the number of high quality/elitist individuals are moved from the current generation to the next without any alteration. This elitism transfer technique avoids the replacement of best fit individuals with poor individuals in the successive generations and also improves the performance of crossover operation, if the parent is selected from the elitist individuals. The individual having less value in the TP of the present population are recognized by Elitist individual.

As like in chapter 4, here also same elitism transfer technique is followed. After the elitism transfer, select the best parent individual from the whole population which is having minimum pollution value by applying greedy crossover and swap mutation operators. To proceed GA operation further, from the current population the initial two parent solutions $P\_Indiv1 \; and \; P\_Indiv2$ are randomly selected and the first city of the offspring is selected from any one of the selected parent individual, thus we consider the $Length = 1$. The remaining n length of offspring individual is constructed by applying greedy crossover technique.

To construct the remaining subsequent cities of the offspring individual the following equations are used to find the position of the current city of the selected two parent individuals. This process continuous until the length offspring individual limit completes.

$$Pos1 \leftarrow \; find\left(P_{indiv1}(Cur\_City)\right) \rightarrow 45$$

$$Pos2 \leftarrow \; find(P_{indiv2}(Cur\_City)) \rightarrow 46$$

As like we discussed in the previous chapter the same heuristics is followed. The position of current city in the parent individuals is used to identify the location of left $LLoc$ and right $RLoc$ adjacent cities of Cur_city .

IF $Pos1 = 1$

$$LLoc1 \leftarrow n, RLoc1 \leftarrow Pos1 + 1$$

Else IF $Pos1 = n$

$$LLoc1 \leftarrow Pos1 - 1, RLoc1 \leftarrow 1$$

Else

$LLoc1 \leftarrow Pos1 - 1,\ RLoc1 \leftarrow Pos1 + 1$

IF $Pos2 = 1$

$$LLoc2 \leftarrow n, RLoc2 \leftarrow Pos2 + 1$$

Else IF $Pos2 = n$

$$LLoc2 \leftarrow Pos2 - 1, RLoc2 \leftarrow 1$$

Else

$$LLoc2 \leftarrow Pos2 - 1, \qquad RLoc2 \leftarrow Pos2 + 1$$

In this scenario population is selected from the newly constructed pollution matrix. The location of near cities in the parent solutions are used to find the city with the slightest pollution from the Cur_City,

$$p_1 \leftarrow POLL(P\_Indiv1(LLoc1), P\_Indiv1(Pos1)) \rightarrow 47$$
$$p_2 \leftarrow POLL(P\_Indiv1(Pos1), P\_Indiv1(RLoc1)) \rightarrow 48$$
$$p_3 \leftarrow POLL(P\_Indiv1(LLoc2), P\_Indiv1(Pos2)) \rightarrow 49$$
$$p_4 \leftarrow POLL(P\_Indiv1(Pos2), P\_Indiv1(RLoc2)) \rightarrow 50$$

As like in previous chapter instead of choosing minimum cost, here the minimum pollution value among the four $p_1, p, p_3 and\ p_4$ is selected and the city at the adjacent location of the concerned parent is chosen as the next city $Next\_City$. The selected city $Next\_City$ is tested for two situations,

Situation 1: The selected city should not be present in the partially constructed offspring i.e. $Next\_City \notin Indiv$.

Situation 2: The pollution value between the current city Cur_City and the chosen next city $Next\_City$ should be within the maximum pollution limit.

$$POLL_{Cur\_City\ ,Next\_City} < Max\_Poll$$

If the selected city fulfills both the situations, it is added as the next subsequent city in the offspring Indiv and the length of the offspring is incremented $\text{Length} \leftarrow \text{Length} + 1$ otherwise the city with subsequent least pollution is chosen and verified. If all the possible cities are checked, next city is added randomly. The similar procedure is continuous until the length of the offspring Indiv1 is n which specifies that the offspring is a feasible solution/route of n cities. The similar procedures are followed to construct the second offspring Indiv2 as well. The swap mutation technique is applied to the newly constructed offspring's by exchanging the randomly chosen cities,

$$GeneC1 \leftarrow RAND\ (1, n,), GeneC2 \leftarrow RAND\ (1, n) \rightarrow 51$$

Within the offspring as,

$$Indiv(GeneC1)\ \leftarrow Indiv(GeneC2)\ and\ Indiv(GeneC2)\ \leftarrow Indiv(GeneC1) \rightarrow 52$$

This phase confirms that the building of offspring is completed and it is added in the following population and the size within the population is incremented $\text{Size} \leftarrow \text{Size} + 1$. The generation of subsequent population Pop of individuals is said to be completed if the $\text{Size} = \text{PopSize}$ and the population generation are repeated for G number of times, then the execution stops. The final population is assessed for best solution in terms of distance and pollution using the following equations respectively.

$$\forall i\ [1 \leq i \leq PopSize], TC_i \leftarrow \sum_{j=1}^{n} DM(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\ \ n+1\ \equiv 1 \rightarrow 53$$

$$\forall i\ [1, \leq i \leq PopSize], TP_i \leftarrow \sum_{j=1}^{n} POLL(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\ \ n+1\ \equiv 1 \rightarrow 54$$

The final population of individuals is considered for validation to bring out the best in at last.

5.4 EXPERIMENTAL ANALYSIS

As discussed in the Section 3, the optimal air pollution based optimal routing in TSP is intended to find the optimal route based on “the total air pollution of the route” as a primary factor. The total distance of the derived optimal route is considered as the secondary factor for the final solution selection. In each of the performance criteria associated with this scenario, the cost refers to the total air pollution of the route obtained.

5.4.1 Optimal Pollution Based Routing in VRP

In this scenario of experiments, the intelligent routing in VRP has been performed by optimizing the total air pollution of the route. Experimental results for the scenario 1 of analyses with random, nearest neighbor and ODV based EV population seeding techniques are shown in the Table.5.4.1

Observation 11: The computation time increases based on the problem instances, each technique has its own computation time for every problem instances. For most of the instances, the computation time for NN technique outperforms random and ODV-EV population seeding techniques. The computation time of NN technique has ranges from 10.540 to 55.340 and the ODV-EV technique ranges from 10.930 to 55.340. The random technique have additional computation time for all the problem instances, since the cities are added randomly and it should satisfy our primary criteria (i.e.) the pollution between adjacent cities should be less than maximum pollution.

Observation 12: For all the problem instances, the ODV-EV population seeding technique performs better in terms of convergence rate % for the best individual in the population. The maximum and minimum convergence rates obtained are 98.29% for eil76 using the ODV-EV technique and 79.79% for the instance bays29 using random technique respectively. The NN technique has the maximum and minimum convergence rate of 91.259 for the instance eil51 and 79.667 for the instance bays29. The random technique has the maximum and minimum convergence rate of 90.775 for the instance eil51 and 79.798 for the instance bays29.

Observation 13: It is observed from the result that the worst convergence rate or the worst individuals in the population of ODV-EV technique showed better performance. In the worst convergence rate, the maximum and minimum worst convergence rate is obtained for the instance KroA100 and uyssess16 in Random technique with 61.329 % and -75.681%. The performance of NN technique is better than random in terms of worst convergence, the maximum and minimum convergence rate of 64.907 for the instance KroA100 and -84.008 for the instance bays29.

Observation 14: It is obvious that the convergence rate is contradictory to the error rate, hence the result of convergence rate is higher, and then the error rate has driven lesser. Since the Performance analyses based on the best convergence rate %

reveals that the ODV-EV technique performs outstandingly means that the best error rate of ODV-EV technique have good performance and has maximum of 7.43% for the instance swiss42 where as NN and random techniques have maximum of 20.33% and 20.20% respectively for the instance bays29. From table 3. It is notable that the eil51 has acquired the minimum values 8.741% in and 9.245% for the NN and random technique respectively.

Observation 15: The worst error rate for all the techniques have higher values for the small instances and moving towards the larger instances the worst error rate values decreases progressively. The results of ODV-EV technique showed lesser values for most of the instances and have a minimum of 23.769% in KroA100. The NN technique has had the maximum worst error rate of 184.008% in uysses16, when compare to the ODV-EV and random technique maximum worst error rate 175.681% for the same instance uysses16.

Observation 16: For all the population seeding techniques, the average convergence rate improves with increase in the size of the problem instance. The ODV-EV technique gives at least 80% of average convergence for the instances eil51, eil76 and kroA100 whereas random and NN techniques offer at least 60% and 70% convergence for the same set of instances. The ODV-EV technique has showed extreme result and it acquired the lower average convergence values as well as the upper average convergence values.

Observation 17: From the fig.5.3, the data clearly indicate that the average error rate is contradictory to the average convergence rate, it is evidentially proves that the KroA100 has obtained lower values for all the techniques. The average error rate ranges from 16.698 to 122.340 for the instances KroA100 and uysses22. It is also pertinent to suggest that, the NN technique showed a reasonable outcome fluctuates from 28.082 to 114.445.

Observation 18: For the instances eil51 and eil76, ODV-EV technique yields more than 90% convergence rate in terms of distance for the pollution optimized route.

Instance			Optimal Solution	Computation Time	Quality of the Solution			Convergence Rate (%)		Error Rate (%)		Convergence Diversity	Average Convergence
					Best	Worst	Average	Best	Worst	Best	Worst		
uysses16	**EV**	Pollution	2.5596	10.930	2.721	6.603	5.373	93.701	-57.975	6.299	157.975	151.676	-9.915
		distance	74.1087		110.467	130.195	121.894	50.939	24.319	49.061	75.681	26.620	35.520
	NN	Pollution	2.5596	10.540	2.827	7.269	5.489	89.569	-84.008	10.431	184.008	173.577	-14.445
		distance	74.1087		107.843	136.305	122.800	54.480	16.074	45.520	83.926	38.406	34.298
	Random	Pollution	2.5596	10.940	2.880	7.056	5.363	87.479	-75.681	12.521	175.681	163.159	-9.527
		distance	74.1087		108.616	137.641	122.359	53.437	14.272	46.563	85.728	39.165	34.892
uysses22	**EV**	Pollution	3.194	12.850	3.363	8.003	7.102	94.699	-50.571	5.301	150.571	145.270	-22.340
		distance	75.6615		127.328	154.466	143.602	31.714	-4.154	68.286	104.154	35.868	10.204
	NN	Pollution	3.194	12.700	3.611	7.092	5.789	86.945	-22.043	13.055	122.043	108.987	18.744
		distance	75.6615		101.422	136.734	120.030	65.953	19.283	34.047	80.717	46.670	41.359
	Random	Pollution	3.194	12.510	3.753	6.836	5.894	82.492	-14.030	17.508	114.030	96.522	15.465
		distance	75.6615		100.520	139.907	118.247	67.145	15.088	32.855	84.912	52.057	43.716
bays29	**EV**	Pollution	5.1614	20.950	5.351	9.741	8.612	96.331	11.269	3.669	88.731	85.061	33.139
		distance	2020		4084.200	5141.800	4814.686	-2.188	-54.545	102.188	154.545	52.356	-38.351
	NN	Pollution	5.1614	21.260	6.211	11.700	9.590	79.667	-26.687	20.333	126.687	106.354	14.207
		distance	2020		3974.000	4772.400	4408.302	3.267	-36.257	96.733	136.257	39.525	-18.233
	Random	Pollution	5.1614	21.690	6.204	11.885	9.886	79.798	-30.270	20.202	130.270	110.067	8.459
		distance	2020		3824.400	4931.200	4515.756	10.673	-44.119	89.327	144.119	54.792	-23.552
swiss42	**EV**	Pollution	6.2613	20.930	6.727	11.245	8.721	92.565	20.402	7.435	79.598	72.164	60.708
		distance	1273		1516.867	2321.533	2128.454	80.843	17.633	19.157	82.367	63.210	32.800
	NN	Pollution	6.2613	20.540	7.195	12.909	9.466	85.080	-6.175	14.920	106.175	91.255	48.824
		distance	1273		1773.800	2825.000	2385.476	60.660	-21.917	39.340	121.917	82.577	12.610
	Random	Pollution	6.2613	21.460	7.099	11.251	8.998	86.615	20.305	13.385	79.695	66.310	56.293
		distance	1273		1672.200	2745.000	2317.868	68.641	-15.632	31.359	115.632	84.273	17.921
eil51	**EV**	Pollution	7.6588	27.970	7.659	10.322	9.165	98.2	65.223	1.800	34.777	32.977	80.331
		distance	426		461.478	633.706	571.335	91.672	51.243	8.328	48.757	40.429	65.884
	NN	Pollution	7.6588	27.740	8.328	10.831	9.915	91.259	58.586	8.741	41.414	32.672	70.545
		distance	426		576.452	775.829	691.079	64.683	17.880	35.317	82.120	46.802	37.775
	Random	Pollution	7.6588	27.940	8.367	10.567	9.725	90.755	62.023	9.245	37.977	28.733	73.024
		distance	426		620.006	777.902	710.770	54.459	17.394	45.541	82.606	37.065	33.153
eil76	**EV**	Pollution	11.3454	43.910	11.539	14.174	13.340	98.290	75.071	1.710	24.929	23.219	82.418
		distance	538		588.272	827.159	731.312	90.656	46.253	9.344	53.747	44.403	64.068
	NN	Pollution	11.3454	44.660	12.633	15.767	14.550	88.652	61.028	11.348	38.972	27.624	71.752
		distance	538		970.029	1176.016	1070.756	19.697	-18.590	80.303	118.590	38.288	0.975
	Random	Pollution	11.3454	45.590	13.211	16.462	15.180	83.558	54.902	16.442	45.098	28.656	66.206
		distance	538		1148.684	1338.700	1240.676	-13.510	-48.829	113.510	148.829	35.319	-30.609
kroA100	**EV**	Pollution	14.5057	56.910	15.145	17.954	16.928	95.596	76.231	4.404	23.769	19.365	83.302
		distance	21285		33840.446	46944.547	41074.316	41.013	-20.552	58.987	120.552	61.565	7.027
	NN	Pollution	14.5057	55.340	16.914	19.596	18.579	83.400	64.907	16.600	35.093	18.493	71.918
		distance	21285		58035.021	68783.612	64182.120	-72.657	-123.155	172.657	223.155	50.498	-101.537
	Random	Pollution	14.5057	58.410	16.938	20.115	18.960	83.230	61.329	16.770	38.671	21.902	69.291
		distance	21285		64750.406	76127.476	69835.489	-104.207	-157.658	204.207	257.658	53.451	-128.097

Table.5.4.1 Experimental results of optimal pollution based routing

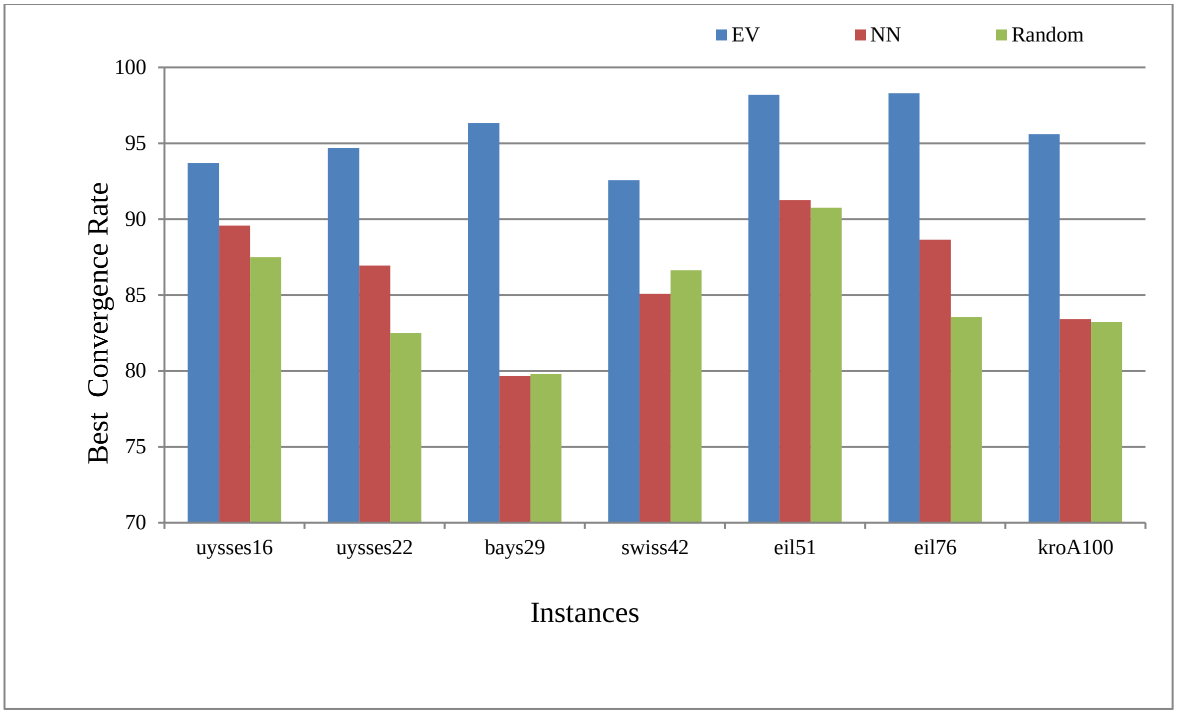


Fig. 5.4.1(a). Best convergence for optimal pollution based routing

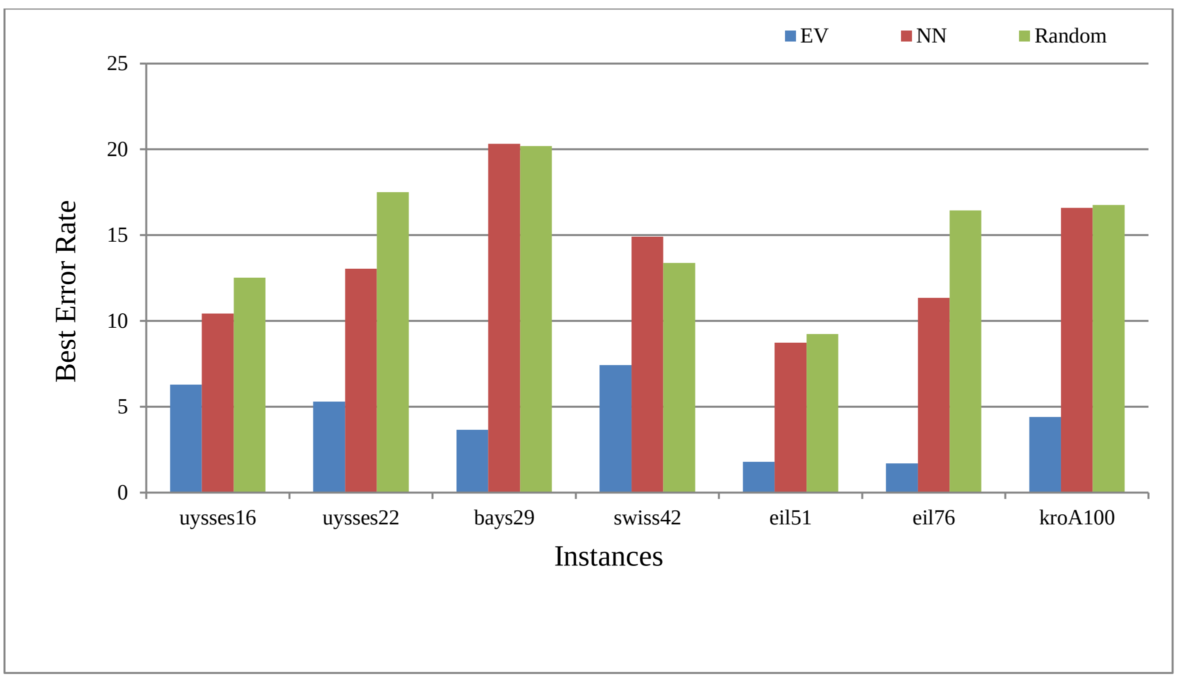


Fig. 5.4.1(b). Best error rate for optimal pollution based routing

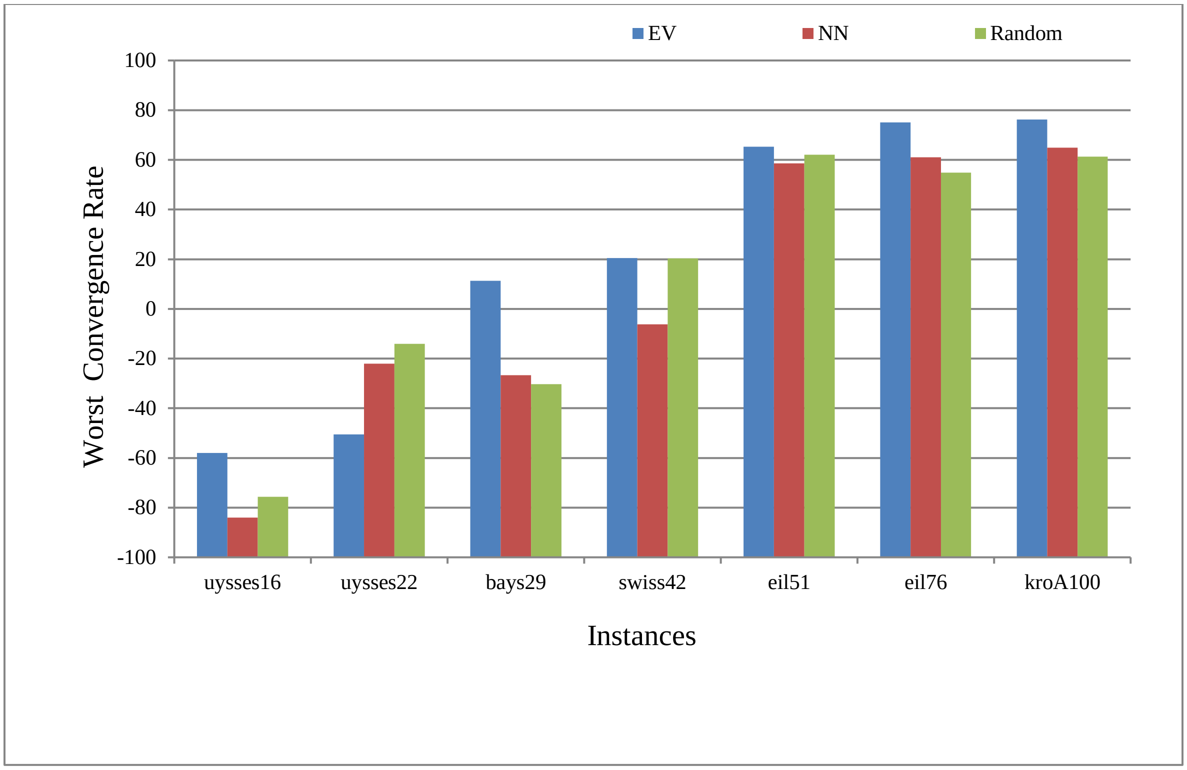


Fig. 5.4.1(c). Worst convergence for optimal pollution based routing

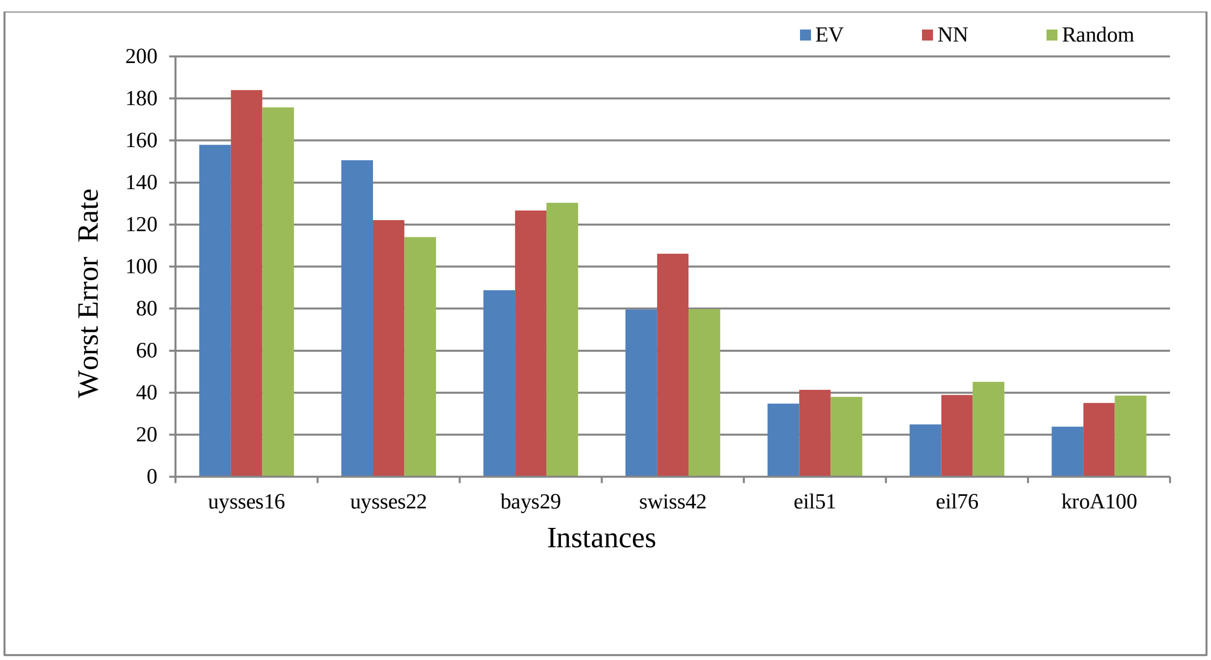


Fig. 5.4.1(d). Worst error rate for optimal pollution based routing

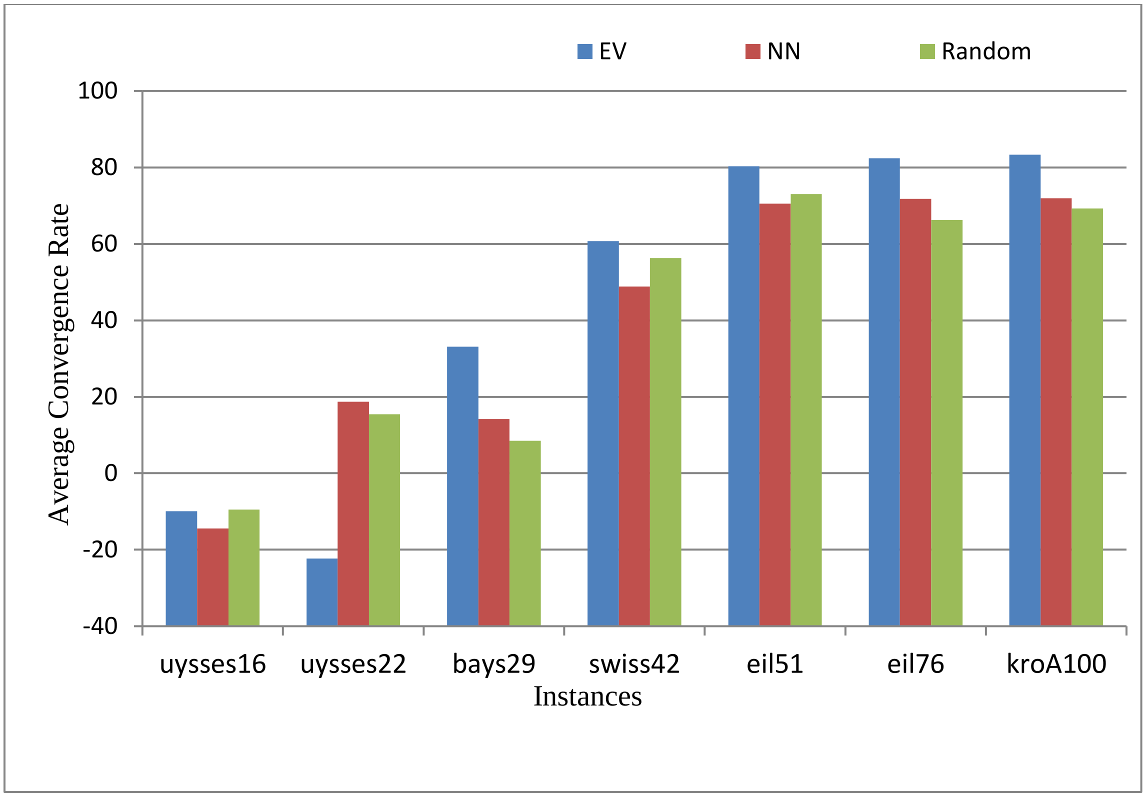


Fig. 5.4.1(e). Average convergence rate for optimal pollution based routing

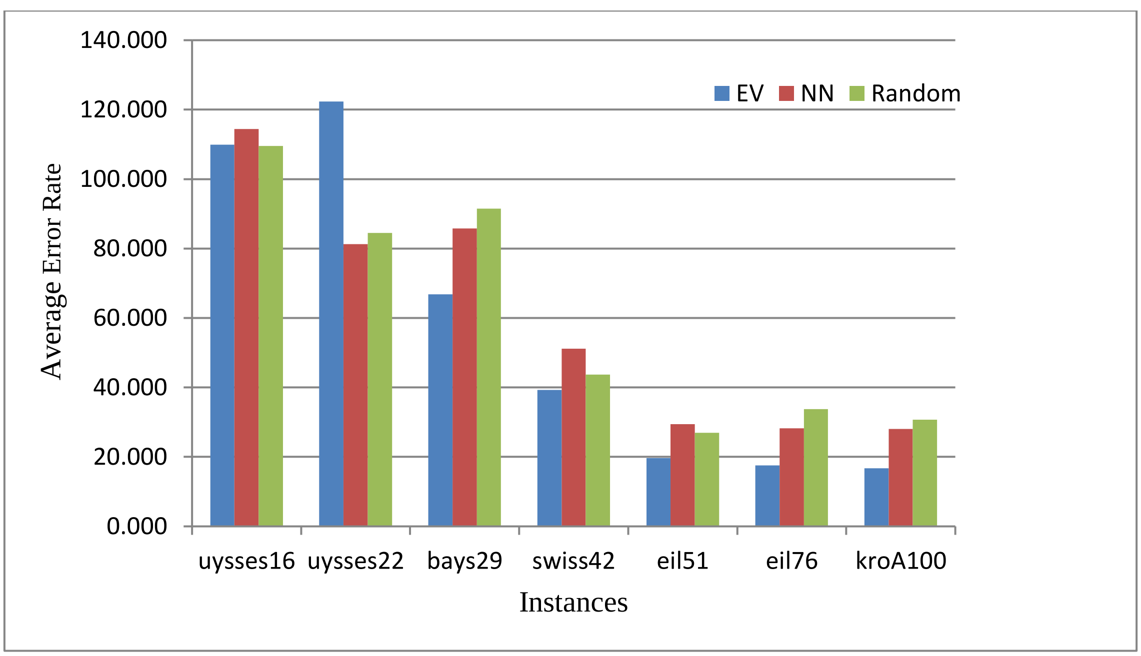


Fig. 5.4.1(f). Average error rate for optimal pollution based routing

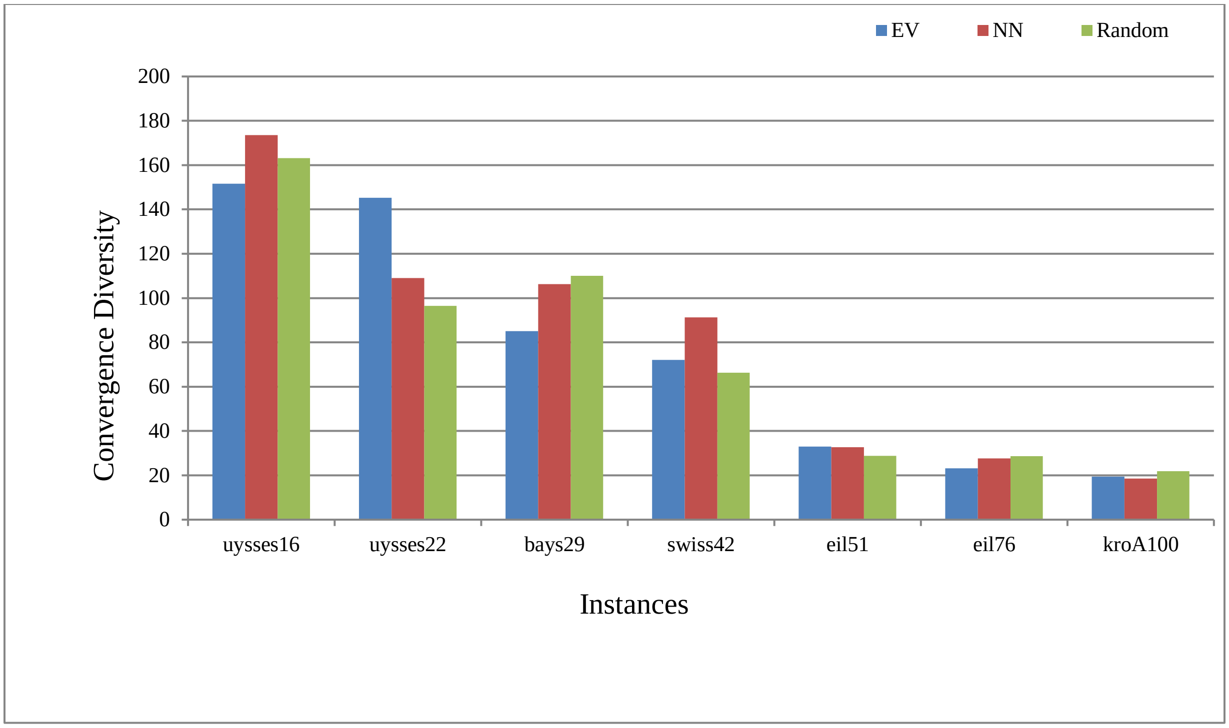


Fig. 5.4.1(g). Convergence diversity for optimal pollution based routing

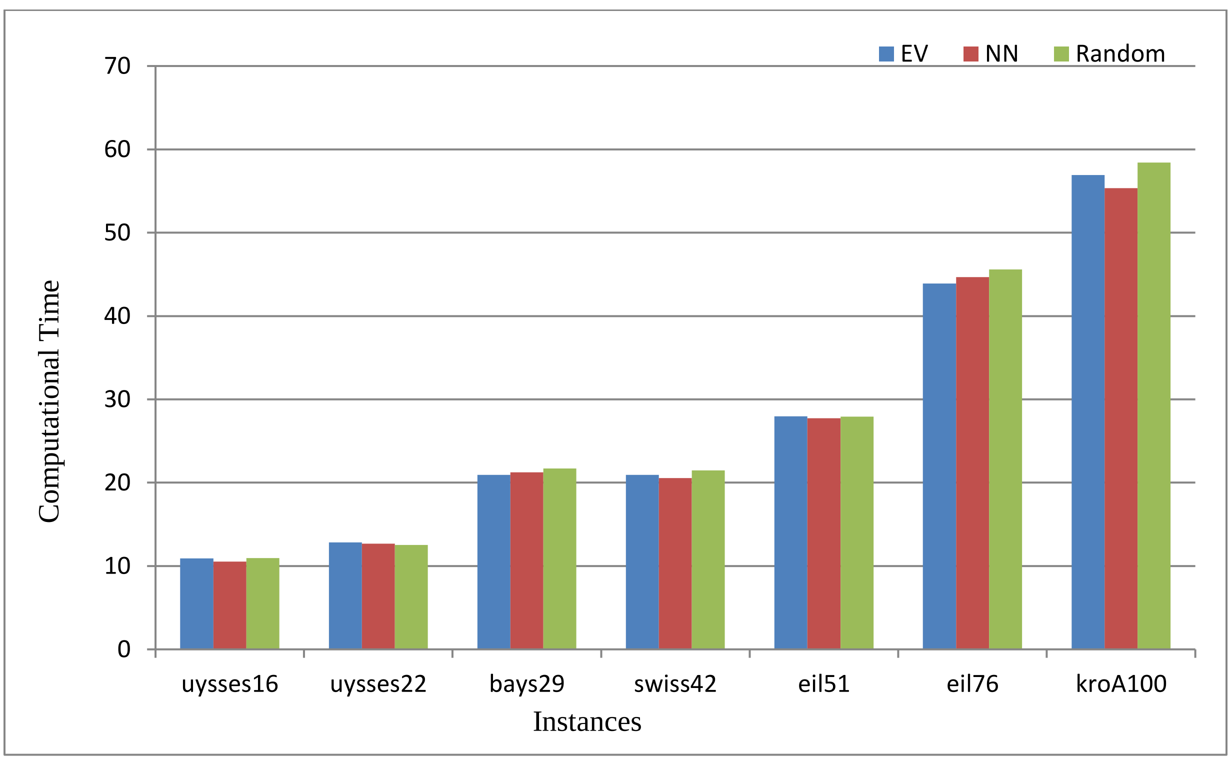


Fig. 5.4.1(h). Computational Time for optimal pollution based routing

5.4.2 Result Analyses

Computational Time: As shown in the Fig.5.4.1(h) significantly proves that the computation time increases based on the problem instances, each technique has its own computation time for every problem instances. In terms of computation time, it is obvious that the random technique showed good result in classical TSP or any other problem. In this case, the each technique must validate the pollution between the corresponding cities before adding the next city. Hence, the computation time of each technique for different instance has had slight changes. Furthermore, analyzed from the Fig.5.4.1(h)the random technique has showed an irregular change for the problem instances, for most of the instances the NN technique is better than ODV-EV and random technique. The performance of ODV-EV has got a acceptable computation time for all the instance.

Analysis on Convergence Rate w.r.t Air Pollution: Fig.5.4.1(a) and Fig.5.4.1(c) show the best and worst convergence rates (%), of the pollution optimized solution, achieved for the problem instances using different population seeding techniques. From the Fig.5.4.1(a) and Fig.5.4.1(c), it can be observed that ODV-EV technique has high convergence rate % in terms of both best and worst individuals in the population. The random and NN techniques perform with nearly equal convergence rate and the deviation in the convergence rate between ODV-EV and random/NN techniques increases with the increase in the size of the problem instance. Every population seeding technique offer non-positive convergence rate for small sized instances and positive convergence rate for the larger size problems.

Analysis on Error Rate w.r.t Air Pollution: Fig.5.4.1(b) and Fig.5.4.1(d) shows the pollution of the route based best and worst error rates (%) reached using different population seeding techniques for the problem instances. In the case of best error rate %, the ODV-EV technique outperforms other techniques though the NN technique values are much closer to the ODV-EV technique. From the Fig.5.4.1(f), it can be observed that every population seeding technique offers nearly same worst error rate % and it linearly improves with increase in the size of the problem instances.

Analysis on Average Convergence w.r.t Air Pollution: Fig.5.4.1(e) shows the average convergence rate for air pollution based optimal routing using different population seeding techniques for the problem instances. From the Fig.5.4.1(e) it can be understood that average convergence rate increases with increase in the size of the problem instances regardless of the population technique used. In the case of average convergence rate, all the population seeding techniques perform nearly equal though ODV-EV technique yields marginally better result than other techniques.

Analysis on Convergence diversity w.r.t. Air Pollution: The convergence diversity of the air pollution based optimal routing scenario using different population seeding techniques for the problem instances in shown in the Fig.5.4.1(g). From the Fig.5.4.1(g), it is observed that the convergence diversity of the instances decreases with increase in the problem size despite the population technique applied.

5.5 SUMMARY

This chapter describes the second module of the proposed work, which is optimal pollution based routing in VRP. The objective of this phase is to find the best optimal path based on less pollution of the path. The optimal result is selected based on the entire distance of the resulting route which is taken as a secondary factor. Heuristic method or random approach in every generation, the value of individuals is improved by pollution and initial population has also been created by using ODV-EV, NN, and Random technique. To decrease the entire pollution of the root path in every generation, the swap mutation operator and greedy crossover operator are used to maintain the best parent individual throughout the process. From the table 5.4.1 and graph it is observed that ODV-EV seeding technique produced good result compared to NN and Random. Hence it is proved that our proposed GA performs well.

CHAPTER 6

SCENARIO-3: HYBRID OPTIMAL ROUTING IN VRP

6.1 INTRODUCTION

In hybrid optimal Based routing, we are looking this problem as a multi objective, the goal is to minimize the distance and also the pollution. The hybrid optimal base routing is achieved depending on the average convergence of distance and the pollution from the initial population. The selection operation has been done through the average convergence of the distance and the pollution. The crossover operation has been performed depending on the normalization of both distance and pollution. Using equation (7) and (12), the convergence rate of the distance and pollution has been estimated from the individual in the population (i.e.) the cost convergence and the pollution convergence. The average convergence of each individual is calculated from the average of cost convergence and the pollution convergence, and the average convergence of each individual arranged in the increasing order. In selection process, the elitist individuals have been selected from the average of convergence rate of the population. The individuals having best fitness or maximum average convergence tour are selected and send to the next generation.

$$Cost\ Convergence(\%) = \left(1 - \frac{Fitness\ cost - Optimal\ cost}{Optimal\ cost}\right) * 100 \qquad \rightarrow 55$$

$$Pollution\ Convergence(\%) = \left(1 - \frac{Fitness\ Pollution - Optimal\ Pollution}{Optimal\ Pollution}\right) * 100 \qquad \rightarrow 56$$

$$Average\ Convergence\ for\ hybrid(\%) = \frac{Cost\ Convergence + Pollution\ Convergence}{2} \qquad \rightarrow 57$$

$$TCC_j = \begin{vmatrix} \mathrm{TCC}_1 \\ \mathrm{TCC}_2 \\ \mathrm{TCC}_3 \\ \cdots \\ \mathrm{TCC}_{PopSize} \end{vmatrix} = \begin{vmatrix} \mathrm{TC}_1 \\ \mathrm{TC}_2 \\ \mathrm{TC}_3 \\ \cdots \\ \mathrm{TC}_{PopSize} \end{vmatrix} = TC_j \qquad \rightarrow 58$$

$$TPC_j = \begin{vmatrix} \mathrm{TPC}_1 \\ \mathrm{TPC}_2 \\ \mathrm{TPC}_3 \\ \cdots \\ \mathrm{TPC}_{PopSize} \end{vmatrix} = \begin{vmatrix} \mathrm{TP}_1 \\ \mathrm{TP}_2 \\ \mathrm{TP}_3 \\ \cdots \\ \mathrm{TP}_{PopSize} \end{vmatrix} = TP_j \qquad \rightarrow 59$$

$$Avg\_Con_j = TCC_j + TPC_j/2 \qquad \rightarrow 60$$

TC_jandTP_j represents the total cost and total pollution each individual in the population. The total cost convergence and the total pollution convergence is represented as$\mathrm{TCC}_j$and TCP_j. The average convergence of each individual is represented asAvg_Con_j.The cost convergence and pollution convergence of each individual in the population is calculated using equation (14) and (15).using equation (16) the average convergence of the each individual in the population is calculated. The individuals having best fitness or maximum average convergence tour are selected and send to the next generation.

$$POP_{ER\times n} = \begin{vmatrix} City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ City_1 & City_i & City_i & \cdots & City_{n-1} & City_1 \end{vmatrix} = \begin{vmatrix} \mathrm{Indiv}_1 \\ \mathrm{Indiv}_2 \\ \mathrm{Indiv}_3 \\ \cdots \\ \mathrm{Indiv}_{ER} \end{vmatrix} =$$

$$\begin{vmatrix} Avg\_Con_1 \\ Avg\_Con_2 \\ Avg\_Con_3 \\ \cdots \\ Avg\_Con_{ER} \end{vmatrix} = \min(Avg\_Con_{ER}) \qquad \rightarrow 61$$

Where ER is the Elitism Rate, $Avg\_Con_1, Avg\_Con_2, Avg\_Con_3 \dots Avg\_Con_{ER}$, are the maximum average convergences of each individual arranged in the increasing order and $Indiv_1, Indiv_2, Indiv_3, \dots Indiv_{ER}$ are the corresponding individuals to the maximum average convergence $Avg\_Con_2, Avg\_Con_3 \dots Avg_{Con_{ER}}$, are the maximum average convergences of each individual arranged in the increasing order and $Indiv_1, Indiv_2, Indiv_3, \dots Indiv_{ER}$ are the corresponding individuals to the maximum average convergence. Depending on ER, the number of elitist individuals are selected and passed to next generation populationPOP

The crossover is done through greedy crossover. From the total population two random individuals are selected as parent individual. All the individuals should have the same starting city, so the starting city is the initial city of the offspring. Find the position of the current city in both the parent individuals $Pos1 \leftarrow find(P\_indiv1\ (Cur\_city))$, $Pos2 \leftarrow find(P\_indiv2\ (Cur\_city))$ and then locate the right side and left side city to the current city in both the parent individuals. If the position of current city is the starting city then location of left city to the current city is last city $IF\ Pos1 = 1\ then\ \ LLoc1 \leftarrow n$ and if the position of current city is last city then location of Right city to the current city is last city $IF\ Pos1 = n\ then\ RLoc1 \leftarrow 1$. Estimate the total cost TC and the pollution TP of the corresponding cities in the location to normalize the distance and the pollution.

The crossover is done through greedy crossover. From the total population two random individuals are selected as parent individual. All the individuals should have the same starting city, so the starting city is the initial city of the offspring. Find the position of the current city in both the parent individuals and then locate the right side and left side city to the current city in both the parent individuals. Estimate the cost and the pollution of the corresponding cities in the location using equation (21 and 22), then normalize the distance value and the pollution value using equation (23).

$$d'_t = \frac{\sum_{t=1}^{size(d)} d_t}{TC} \qquad \rightarrow 62$$

$$p'_t = \frac{\sum_{t=1}^{size(p)} p_t}{TP} \qquad \rightarrow 63$$

$$\forall[1, \leq z \leq Size(Loc)], \Omega_z \leftarrow TR * p'_z + (1 - TR) * d'_z \qquad \rightarrow 64$$

According to distance and pollution, a normalized value has been calculated for all the adjacent cites located in different locations w.r.t the current city. The city which is having minimum normalized value has been checked for the primary constrain, it should be less than the maximum pollution and the added in offspring individual. Repeat this process until a complete individual has generated and pass the individual to the Mutation process, here two random locations are generated and then swap the offspring individual cities in the locations and vice versa. The offspring’s are added into the population, Stop this process until we reach the total population.

6.2 PROBLEM DESCRIPTION AND FORMULATION

In this thesis, the standard VRP is considered in a different perspective to propose a new environment concerned transportation problem in which the optimal path should be of least distance and also minimum air-pollution along the route. A pollution matrix of TSP, similar to distance matrix, is formulated to specify the pollution between each pair of cities. A pollution limit between the cities is the maximum allowed pollution value between any two cities in any feasible solution for the problem. During the formulation of solution, at each stage, inclusion of a new city is allowed only if the pollution value between the previous and the new city is less than that of maximum allowed pollution limit between the cities else, it would try to select the alternate city. The intelligent routing strategy for VRP in Hybrid Optimal routing can be represented as follows:

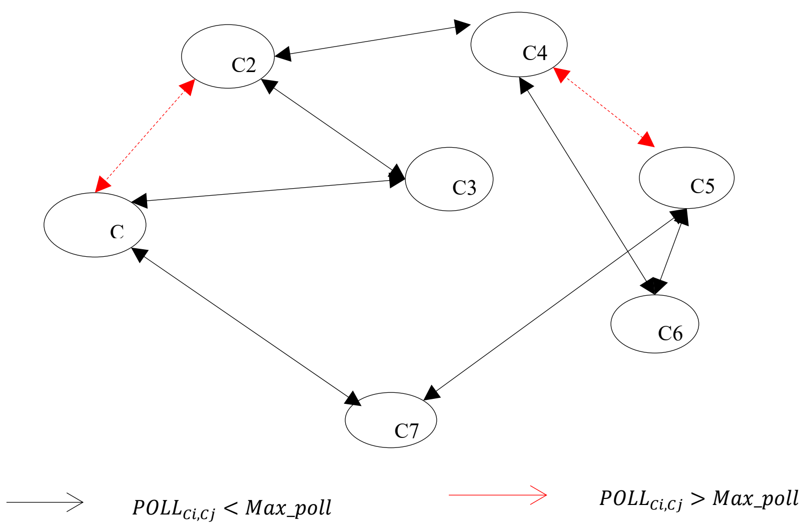


Fig.6.2. Sample intelligent routing strategy for VRP *w.r.t* Hybrid model

Let, the complete undirected graph $G = \{City_n , A\}$ and $DM(City_i , City_j)$ is the distance between the cities $City_i$ and $City_j$ such that $City_i \neq City_j$ and

$DM(City_i, City_j) = DM(City_j, City_i)$.The pollution matrix for the TSP problem of size 'n'can be represented as $POLL(n \times n)$and $POLL(City_i, City_j)$ is the pollution between the $City_i$ and the $City_j$. In this proposal, the starting city remains same because the vehicle should start from a single predetermined source. The working principle of the proposed intelligent routing strategy with an example is illustrated below:

Consider that the intelligent routing strategy for VRP with the size of 7 cities $(C1, C2, C3 \ldots . C7)$ as shown in the Figure 1. In the Figure 1, the red dashed line shows the path between any cities, which current air-pollution is higher than the pollution limit of the problem and the black line shows the path with pollution within the limit. Assume that $C1$ is the starting city and the neighboring cities are organized in ascending order of their distance such that $(d(C1, C2) \leq d(C1, C3) \leq \cdots \leq d(C1, C7))$. The objective of the intelligent routing strategy for VRP in hybrid optimal routing is to choose the adjacent city devising lowest normalized value and the pollution between the cities are within the pollution limit has selected as next city. Starting from the city $C1$, the adjacent city is $C2$ with least normalized value but the pollution between the cities $C1$ and $C2$ exceeds the pollution limit. i.e. $POLL_{C1,C2} < Max\_poll$. So, the process of inclusion of the city $C2$ adjacent to the city $C1$ in the route is aborted and the condition is verified with the next least normalized value city of $C1$ which is $C3$. The same procedure is repeated until the complete route is generated with $n$number of cities and the possible route would be $(C1, C3, C2, C4, C6, C5, C7, C1)$. It not guaranteed that the individuals in the population yields optimal solution to the problem for both air pollution and distance, based on the genetic operations the individuals in the populations are improved. As like in the previous chapters, the objective of intelligent routing strategy for VRP in Hybrid Optimal routing is to minimize the air pollution and the distance of the individuals in the population, for that different tradeoff method should be provided.

6.3 ALGORITHM DEVELOPMENT

In this work, the intelligent routing strategy for VRP has been proposed using Genetic Algorithm in Hybrid Optimal routing. In hybrid optimal based routing, our ultimate aim is to tradeoff between the air Pollution and the Cost. Based on the Pollution and the cost of the individuals, the genetic operators applied to improve the effectiveness of the tradeoff based population in each generation.

Selection: In selection, our main idea is to pass the best individuals to the next generation. As discussed in chapter 1, in selecting process the individuals which is having minimum tour cost w.r.t distance values is selected and passed to the next generation. Likewise, in chapter 2, the individuals which are having minimum tour cost w.r.t pollution values is selected and passed to the next generation in the selection process. In hybrid optimal based routing, the goal is to minimize both the air Pollution and the distance. Hence, the convergence for the cost and the pollution for all the individuals in the population has been calculated using the equations (1) and (2) given in section (4) and then calculate the average for both the distance and pollution convergence of the entire population. Select ER number of best individuals, whose average convergence should be maximum than the other individuals. The best individuals are moved to the new population. The elitist individuals are selected based on the average convergence of the individuals in the population in selection process.

6.3.1 Algorithm

Hybrid Optimal based routing Algorithm (Pop, TP, Tc, G**,** n**)**

Step 1: Initialize$Gen \leftarrow 1, i \leftarrow 1, Size \leftarrow 1, TTC \leftarrow 0, TTP \leftarrow 0$

Step 2: Set optimal Distance, optimal Pollution and Maximum pollution Opt_Dist , Opt_Poll and Max_Poll

Step 3: Store the Population into a temporary variable,$CPop_{n\times n} \leftarrow Pop_{n\times n}$,

Step 4: Repeat through Step 13 **Until** $Gen \ \leq G$, Step 3 else go to Step 14

Step 5: Repeat through step 5.3 **Until** $l <= Popsize$, else goto Step 6

Step 5.1:$TCC \leftarrow (1-(TC_l - Opt\_Dist)/Opt\_Dist) * 100$ //calculating the cost convergence of the individual

Step5.2:$TPC \leftarrow (1-(TP_l - Opt\_Poll)/Opt\_Poll) * 100$//calculating the pollution convergence of the individual

Step5.3: $TradeOff\_Con_l \leftarrow TCC + TPC/2$// calculating the average of pollution and distance convergence the individual

Step 6: Select the best individual which is having maximum tradeoff Convergence and pass the best Individual to the next generation

Step 6.1: Repeat through Step 6.3 **Until** $i < ER$, else goto Step 7

Step 6.2: $position \leftarrow max(TradeOff\_Con)$// Position of the Individual with Maximum tradeoff convergence value will

// be acquired.

Step 6.3 $Pop_i \leftarrow CPop_{position}$// the individual in the position in temporary population is moved to the population

Step 7: Repeat through Step 8.6 **Until**$Size \leq PopSize$, else goto Step 9 $where\ ER < Size \leq PopSize$

Step 8: Choose the random parents Individuals,P_Indiv1 and P_Indiv2

Step 8.1:Select the initial City $Init\_City,$ and $Length$ ←1, Size←1

Step 8.2: $Indiv[Length] \leftarrow Init\_City$ // the first city in the parent individual is selected as initial city

Step 8.3: Repeat through Step 8.5 **Until**$Length \ \leq n$, else goto Step 5

$Cur\_City \leftarrow Indiv[Length]$ //the current city in the offspring individual assigned as current city

Find the Position $Pos1$ and $Pos2$ of the Current City in the Parent Individuals

$Pos1 \leftarrow find(P\_indiv1\ (Cur\_City)),\ Pos2 \leftarrow find(P\_indiv2\ (Cur\_City))$

IF$Pos1 = 1,\ LLoc1 \leftarrow n$

Else IF$Pos1 = n,\ \ RLoc1 \leftarrow 1$

IF$Pos2 = 1$, $LLoc2 \leftarrow n$

Else IF$Pos2 = n$, $RLoc2 \leftarrow 1$

Evaluate the Tradeoff$Treadeoff_z$ from Previous City to Current City and Current City to Next City from the Parent Individuals using Normalization

$d_1 \leftarrow DM(P\_Indiv1(LLoc1), P\_Indiv1(Pos1))$

$d_2 \leftarrow DM(P\_Indiv1(Pos1), P\_Indiv1(RLoc1))$

$d_3 \leftarrow DM(P\_Indiv1(LLoc2), P\_Indiv1(Pos2))$

$d_4 \leftarrow DM(P\_Indiv1(Pos2), P\_Indiv1(RLoc2))$

$p_1 \leftarrow POLL(P\_Indiv1(Pos1-1), P\_Indiv1(Pos1))$

$p_2 \leftarrow POLL(P\_Indiv1(Pos1), P\_Indiv1(Pos1+1))$

$p_3 \leftarrow POLL(P\_Indiv2(Pos2-1), P\_Indiv2(Pos2))$

$p_4 \leftarrow POLL(P\_Indiv2(Pos2), P\_Indiv2(Pos2+1))$

$TC = \sum_{t=1}^{size(d)} {d_t}^2 , TP = \sum_{t=1}^{size(p)} {p_t}^2$ //sum of pollution and distance for the locations

$d'_t = \sum_{t=1}^{size(d)} d_t /TC$, $p'_t = \sum_{t=1}^{size(p)} p_t /TP$// normalizing the pollution and distance

$\forall[1, \leq z \leq Size(Loc)], \Omega_z \leftarrow TR * p'_z + (1-TR) * d'_z$

Step 8.4: Repeat through Step 8.6 **Until**$k < 4$, $else\ goto\ Step\ 9$ $where\ 0 < k \leq 4$

$Next\_City \leftarrow min(Tradeoff)$ //the location of the city with minimum cost will be acquired

Step 8.5: **IF**$Next\_City \notin Indiv\ and\ POLL_{Cur\_City\ ,Next\_City} < Max\_Poll$, else goto Step 8.4

$Length \leftarrow Length + 1$, Update$Indiv[Length] \leftarrow next_City$

Step 8.6: $k \leftarrow k + 1$// increment the individual in the population
Step 9: Generate Random values $GeneC1, GeneC2$, $where\ 1 \leq GeneC1, GeneC2 \leq n$
$Swap\ Indiv(GeneC1) \leftarrow Indiv(GeneC2)$, $SwapIndiv(GeneC2) \leftarrow Indiv(GeneC1)$
Step 10:$Pop_{size} \leftarrow Indiv$, $Size \leftarrow Size + 1$ // move to the next individual
Step 11: Evaluate the cost of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TC_i \leftarrow \sum_{j=1}^{n} DM(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1\ \equiv 1$$

Step 12: Evaluate the Pollution of each Individual in the Population

$$\forall[1, \leq i \leq PopSize], TP_i \leftarrow \sum_{j=1}^{n} POLL(Indiv_i(j)\ , Indiv_i(j+1)\ )\ ,\quad j+1\ \equiv 1$$

Step 13: $Gen \leftarrow Gen + 1$//Current generation is completed, increment the Gen for next generation
Step 14: Return Pop

Fig.6.3.1. Algorithm for Hybrid Optimal Routing

Crossover: Next, the crossover process is done through greedy crossover. In crossover, choose any two random Parent individuals within the total population. The first city in the parent individual is moved to the offspring individual's first city. Assign the current city as current city in the offspring. Now estimate the position of the current city in both the parent individuals and then locate the right side and left side city to the current city in both the parent individuals. If the position of current city is starting city then location of left city to the current city is last city and if the position of current city is last city then location of Right city to the current city is last city. In Optimal Distance Based Routing, the adjacent city which is having less distance w.r.t current city is added as next city in the offspring. In case of Optimal pollution Based Routing, the adjacent city which is having less pollution w.r.t current city is added as next city in the offspring.

Nevertheless, in Hybrid Optimal routing normalize the Pollution and distance of all the locations from the current city has been calculated and normalize the pollution and distance values. The adjacent city which is having minimum normalized values in different location is selected as next city in the offspring individual. Check whether the next city is already present in the offspring Individual and also the Pollution between the current city and the next city should be minimum (i.e. the pollution between the cities should be less than the maximum pollution) and then add the city in the offspring Individual. Do these procedures until we are getting a complete offspring individual.

Mutation: In Mutation, Generate two random locations and then swap the offspring individual cities in the locations and vice versa. Now add the offspring individual to the population do this procedure till we reach the maximum number of temporary population. Evaluate the total cost and total pollution of all the individuals from the whole population.

6.3.2 Algorithm Explanation

The algorithm for Hybrid Optimal routing has the following arguments; $Pop$is the initial population generated using random or heuristic technique, TC is the total cost of each individual in the initial population using Equation (7), G is the generation limit for termination of GA and n is the size of the problem instance. Elitism Rate ER is the number of high quality/elitist individuals are transferred from the current generation to the next without any modification. This elitism transfer technique avoids the replacement of best fit individuals with poor individuals in the successive generations and also improves the performance of crossover operation, if the parent is selected from the elitist individuals. The total cost and total pollution of each individual in the population is Determined and represented as TC_j and TP_j. The total cost convergence rate TCC and the total pollution convergence rate TPC of the individuals in the population is derived through $TCC \leftarrow (1 - (TC_l - Opt\_Dist)/Opt\_Dist) * 100$ and $TPC \leftarrow (1 - (TP_l - Opt\_Poll)/Opt\_Poll) * 100$. The average of both pollution and distance convergence has been computed (i.e.) $TradeOff\_Con_l$ of the each individual in the population. The ER numbers of individuals having best fitness (i.e.) maximum tradeoff convergence value are hand-picked based on the position and send to consecutive generation.

First, two parent solutions P_Indiv1 and P_Indiv2 are chosen randomly from the current population and the first city of the parents is copied as the first city of the offsprings, thus the $Length = 1$. The construction of a complete offspring $Indiv$ of length $'n'$ using the greedy crossover is explained in the subsequent discussion:

The position of the current city Cur_City of the partially built offspring $Indiv$ in the two selected parents is identified using the following conditions,

$$Pos1 \leftarrow find\left(P_{indiv1}(Cur\_City)\right) \quad \rightarrow 65$$

$$Pos2 \leftarrow find(P_{indiv2}(Cur\_City)) \quad \rightarrow 66$$

The position of current city in the parents is used to identify the location of left $LLoc$ and right $RLoc$ adjacent cities of Cur_city in the concerned parent solutions and the corresponding location value can be acquired by following the following heuristic:

IF $Pos1 = 1$

$LLoc1 \leftarrow n, RLoc1 \leftarrow Pos1 + 1$

Else IF $Pos1 = n$

$LLoc1 \leftarrow Pos1 - 1, RLoc1 \leftarrow 1$

Else

$LLoc1 \leftarrow Pos1 - 1,\ RLoc1 \leftarrow Pos1 + 1$

IF $Pos2 = 1$

$LLoc2 \leftarrow n, RLoc2 \leftarrow Pos2 + 1$

Else IF $Pos2 = n$

$LLoc2 \leftarrow Pos2 - 1, RLoc2 \leftarrow 1$

Else

$LLoc2 \leftarrow Pos2 - 1,\ RLoc2 \leftarrow Pos2 + 1$

The location of adjacent cities in the parent solutions are used to find the city with the least distance from the Cur_City is determined,

$$d_1 \leftarrow DM(P\_Indiv1(LLoc1), P\_Indiv1(Pos1)) \quad \rightarrow 67$$

$$d_2 \leftarrow DM(P\_Indiv1(Pos1), P\_Indiv1(RLoc1)) \quad \rightarrow 68$$

$$d_3 \leftarrow DM(P\_Indiv1(LLoc2), P\_Indiv1(Pos2)) \quad \rightarrow 69$$

$$d_4 \leftarrow DM(P\_Indiv1(Pos2), P\_Indiv1(RLoc2)) \quad \rightarrow 70$$

The location of adjacent cities in the parent solutions are used to find the city with the least air pollution from the Cur_City is determined,

$$p_1 \leftarrow POLL(P\_Indiv1(LLoc1), P\_Indiv1(Pos1)) \quad \rightarrow 71$$

$$p_2 \leftarrow POLL(P\_Indiv1(Pos1), P\_Indiv1(RLoc1)) \quad \rightarrow 72$$

$$p_3 \leftarrow POLL(P\_Indiv1(LLoc2), P\_Indiv1(Pos2)) \quad \rightarrow 73$$

$$p_4 \leftarrow POLL(P\_Indiv1(Pos2), P\_Indiv1(RLoc2)) \quad \rightarrow 74$$

Normalizing the calculated adjacent cities distance and pollution values using equation (21) and (22).Ω_{z} Represents the tradeoff values for distance and pollution of each adjacent city that is estimated using equation (23).

$$\forall[1, \leq t \leq 4], \mathrm{D}_{\mathrm{t}}' = \frac{D_t}{\sum_{x=1}^{4} D_x} \rightarrow 75$$

$$\forall[1, \leq t \leq 4], \mathrm{P}_t' = \frac{P_t}{\sum_{x=1}^{4} P_x} \rightarrow 76$$

$$\forall[1, \leq \mathrm{z} \leq 4], \Omega_{\mathrm{z}} \leftarrow D_{\mathrm{z}}' + \mathrm{P}_{\mathrm{z}}' \rightarrow 77$$

The least tradeoff value among the four $\Omega_1, \Omega_2, \Omega_3 and\ \Omega_4$ is selected and the city at the corresponding location of the concerned parent is chosen as the next city $Next\_City$. The chosen city $Next\_City$ is verified for two conditions,

Condition 1: The chosen city should not present in the partially built offspring i.e. $Next\_City\ \notin\ Indiv$.

Condition 2: The pollution value between the current city Cur_City and the chosen next city $Next\_City$ should be within the maximum pollution limit$POLL_{Cur_City\ ,Next_City} < Max_Poll$.

If the chosen city satisfies both the conditions, it is added as the next city in the offspring$Indiv$ and the length of the offspring is incremented $Length \leftarrow\ Length + 1$ otherwise the city with next least distance is chosen and verified. If all the possible cities are checked, next city is added randomly. The same steps are repeated until the length of the offspring $Indiv1$ is n which indicates that the offspring is a feasible solution/route of n cities. The similar procedures are followed to construct the second offspring $Indiv2$. The swap mutation is applied at the resultant offsprings by exchanging the randomly chosen cities, $GeneC1 \leftarrow RAND\ (1, n,), GeneC2 \leftarrow RAND\ (1, n)$ within the offspring as,

$$Indiv(GeneC1)\ \leftarrow Indiv(GeneC2)\ and\ Indiv(GeneC2)\ \leftarrow Indiv(GeneC1) \rightarrow 78$$

This stage confirms that the construction of offspring is completed and it is included in the next population and the size of the population is incremented $Size \leftarrow Size + 1$. The generation of next population Pop of individuals is said to be completed if the $Size = PopSize$ and the population generation is repeated for G number of times, then the execution stops. The final population is assessed for the best solution in terms of distance and pollution using Eq. (9) and (10) respectively.

6.4 EXPERIMENTS AND ANALYSIS

As discussed in section 3, The Hybrid Optimal Based Routing in TSP is based on the tradeoff between the distance and air pollution; exploring this problem as a multi objective. The intension is to find the optimal route based on “the total distance of the route” and “the total air pollution of the route”. In each of the performance criteria associated with this scenario, the cost refers to the total distance of the solution obtained.

6.4.1 Hybrid Optimal Routing in VRP

In this scenario of experiments, the intelligent routing in VRP has been performed by optimizing the total distance of the route and the total air pollution of the route. Experimental results for the scenario 3 of analyses with random, nearest neighbor and ODV based EV population seeding techniques are shown in the Figure 6.3. From the figure 6.3, the following observations can be made:

Observation 19: For all the problem instances, the ODV-EV population seeding technique yields higher convergence rate for the best individual within the population w.r.t air pollution and distance. In best convergence rate, The Maximum convergence rate for distance obtained in ODV-EV technique is 98.830% for eil51 and for the air pollution the Maximum of 100% obtained in ODV-EV technique for eil51. The minimum convergence rate for distance and air pollution are 40.407% and 57.213% obtained in random technique for the instance KroA100.

Observation 20: it is observed from the result that the worst convergence rate or the worst individuals in the population of ODV-EV technique showed better performance. In the worst convergence rate, the maximum and minimum convergence rate are obtained in NN and Random technique with 57.139% and -42.884 for distance. The worst individuals acquired for Air pollution with the maximum rate of 76.864 and the minimum rate of -53.138.

Instance			Optimal Solution	Computation Time	Quality of the Solution			Convergence Rate (%)		Error Rate (%)		Convergence Diversity	Average Convergence
					Best	Worst	Average	Best	Worst	Best	Worst		
uysses16	**EV**	Pollution	2.5596	11.230	2.849	6.120	5.092	88.686	-39.100	11.314	139.100	127.786	1.044
		distance	74.1087		82.465	124.907	109.448	88.725	31.454	11.275	68.546	57.270	52.315
	NN	Pollution	2.5596	11.370	2.915	6.418	5.070	86.134	-50.737	13.866	150.737	136.871	1.923
		distance	74.1087		84.086	130.483	113.127	86.537	23.930	13.463	76.070	62.606	47.350
	Random	Pollution	2.5596	11.270	2.887	6.378	4.918	87.211	-49.187	12.789	149.187	136.399	7.863
		distance	74.1087		83.914	135.927	110.031	86.768	16.585	13.232	83.415	70.184	51.528
uysses22	**EV**	Pollution	3.194	16.700	3.462	8.002	6.972	91.603	-50.537	8.397	150.537	142.140	-18.298
		distance	75.6615		80.566	129.462	129.719	93.518	28.893	6.482	71.107	64.624	28.553
	NN	Pollution	3.194	17.190	3.285	8.085	6.529	97.152	-53.138	2.848	153.138	150.290	-4.429
		distance	75.6615		93.653	155.851	131.725	76.221	-5.984	23.779	105.984	82.205	25.902
	Random	Pollution	3.194	16.850	2.976	7.872	6.136	83.762	-46.474	16.238	146.474	130.236	7.878
		distance	75.6615		95.554	148.302	132.271	73.708	3.993	26.292	96.007	69.715	25.181
bays29	**EV**	Pollution	5.1614	21.860	5.609	9.234	9.429	91.330	21.095	8.670	78.905	70.235	17.317
		distance	2020		2186.000	3842.800	4024.220	91.782	9.762	8.218	90.238	82.020	0.781
	NN	Pollution	5.1614	23.830	6.511	10.454	8.932	73.861	-2.544	26.139	102.544	76.406	26.955
		distance	2020		2800.000	4622.600	3990.376	61.386	-28.842	38.614	128.842	90.228	2.457
	Random	Pollution	5.1614	23.510	6.186	10.773	9.331	80.153	-8.724	19.847	108.724	88.877	19.209
		distance	2020		2946.600	4795.600	4149.330	54.129	-37.406	45.871	137.406	91.535	-5.412
swiss42	**EV**	Pollution	6.2613	23.550	7.466	8.762	9.561	80.766	60.061	19.234	39.939	20.705	47.297
		distance	1273		1330.600	1918.800	1642.133	95.475	49.269	4.525	50.731	46.206	71.003
	NN	Pollution	6.2613	22.730	8.301	9.925	9.219	67.416	41.489	32.584	58.511	25.928	52.766
		distance	1273		1474.400	2134.400	1807.974	84.179	32.333	15.821	67.667	51.846	57.975
	Random	Pollution	6.2613	23.040	8.120	9.785	9.028	70.314	43.727	29.686	56.273	26.587	55.808
		distance	1273		1486.400	2050.400	1777.124	83.236	38.932	16.764	61.068	44.305	60.399
EIL51	**EV**	Pollution	7.6588	28.290	7.659	10.669	8.835	100.000	60.697	0.000	39.303	39.303	84.636
		distance	426		430.983	683.409	516.770	98.830	39.575	1.170	60.425	59.255	78.693
	NN	Pollution	7.6588	28.850	8.257	11.268	10.016	92.183	52.872	7.817	47.128	39.311	69.227
		distance	426		471.989	725.836	606.292	89.205	29.616	10.795	70.384	59.589	57.678
	Random	Pollution	7.6588	29.550	7.974	10.073	9.285	95.879	68.484	4.121	31.516	27.395	78.768
		distance	426		443.793	608.587	543.201	95.823	57.139	4.177	42.861	38.684	72.488
eil76	**EV**	Pollution	11.3454	48.020	11.554	15.132	13.993	98.163	66.620	1.837	33.380	31.543	76.667
		distance	538		551.174	850.545	761.941	97.551	41.906	2.449	58.094	55.645	58.375
	NN	Pollution	11.3454	46.190	13.157	16.341	15.162	84.036	55.971	15.964	44.029	28.064	66.356
		distance	538		632.503	901.407	798.463	82.434	32.452	17.566	67.548	49.982	51.587
	Random	Pollution	11.3454	30.470	13.199	16.510	15.207	83.658	54.477	16.342	45.523	29.181	65.959
		distance	538		638.987	950.171	805.874	81.229	23.388	18.771	76.612	57.841	50.209
kroA100	**EV**	Pollution	14.5057	55.690	15.172	17.862	17.096	95.408	76.864	4.592	23.136	18.545	82.143
		distance	21285		21918.398	33873.150	30479.214	97.024	40.859	2.976	59.141	56.165	56.804
	NN	Pollution	14.5057	60.120	19.222	22.244	21.007	67.490	46.652	32.510	53.348	20.838	55.184
		distance	21285		29886.677	46054.165	40052.584	59.588	-16.369	40.412	116.369	75.957	11.827
	Random	Pollution	14.5057	64.220	20.712	23.666	22.315	57.213	36.849	42.787	63.151	20.364	46.162
		distance	21285		33969.432	51697.878	43604.118	40.407	-42.884	59.593	142.884	83.291	-4.858

Table.6.4.1 Result Analysis of Hybrid Optimal based Routing

Observation 21: Performance analyses in distance, based on the error rate % reveal that the ODV-EV technique performs outstandingly and has maximum of 19.234% for the instance Swiss42 where as NN and random techniques have maximum of 40.412% and 59.593% respectively for the instance KroA100. The minimum and maximum worst error rate in terms of air pollution obtain from that worst individuals are 23.136 % in ODV-EV for the instance KroA100 and 153.138% in NN technique for the instance uysses22.

Observation 22: The average convergence is working better in ODV – EV technique, Average Convergence for Hybrid Optimal routing (Pollution) is less in small cities, and it increases gradually when we are moving towards the large instances. The average convergence of NN technique is less than the random technique. As it is the evident from table.3, both the minimum and maximum average convergences for pollution is obtained in ODV-EV technique ranges from -18.298 to 84.636 in uysses22 and eil51 respectively.

Observation 23: The Convergence diversity of distance as well as pollution values of all the instances is better in ODV-EV technique. The minimum and maximum values of convergence rate w.r.t distance are acquired in random technique are 38.684 and 91.535.The convergence diversity w.r.t pollution values are obtained; minimum value is 18.545 in ODV-EV technique and the maximum value is 150.29.

Observation 24: It is observed from the table 6.4.1 that, the computational time varies for each problem instances. Based on the Problem Instance, ODV-EV and NN techniques showed a gradual increase. The performance of random technique is irregular and it showed unexpected changes in the computation time for the instance eil76. The ODV-EV technique performs well and has less computational time for the entire instance except eil76.

6.4.2 Result Analyses

Best Convergence rate w.r.t distance: The best convergences of distance in ODV-EV technique of are performing well when compared to the best convergence of distance in other techniques. The NN technique is performing better than the random technique in many instances as showed in the fig.6.4.2(a) The performance of random technique for each instance is uneven, gradually decreased, then increased and then decreased.

Best Convergence rate w.r.t pollution: From the Graph it is analyzed that, the best convergence of pollution in NN and Random techniques have showed lower performance, when compare to the best convergences of pollution in ODV-EV technique. Most of the time the performance of random technique is superior to the NN technique in many instances is showed in the Fig. 6.4.2(g). Moving towards the higher instances the convergence rate is gradually decrease in ODV-EV and NN technique, in case of random technique sudden decrease in best convergence rate.

Best Error rate w.r.t distance: the best convergence rate is high in ODV-EV; obviously the best error rate is less in ODV-EV technique. Because, the pollution in the path between current city and the next minimum distance city is high. It will move to the next minimum distance city. So the convergence rate is high and the error rate is less. The random technique has higher error rate than the NN technique in many instances is showed in the fig 6.4.2(b), shows that the performance of NN technique is better than the random technique.

Computational Time: Fig. 6.4.2(o) significantly proves that the computation time increases based on the problem instances, each technique has its own computation time for every problem instances. In terms of computation time, it is obvious that the random technique showed good result in classical TSP or any other problem. In this case, each technique should validate the pollution between the corresponding cities before adding the next city. Hence, the computation time of each technique for different instances has slight changes. Furthermore, analyzed from the Fig. 6.4.2(o) the random technique has showed an abnormal change for the instance eil51, except that the ODV-EV technique shows good performance.

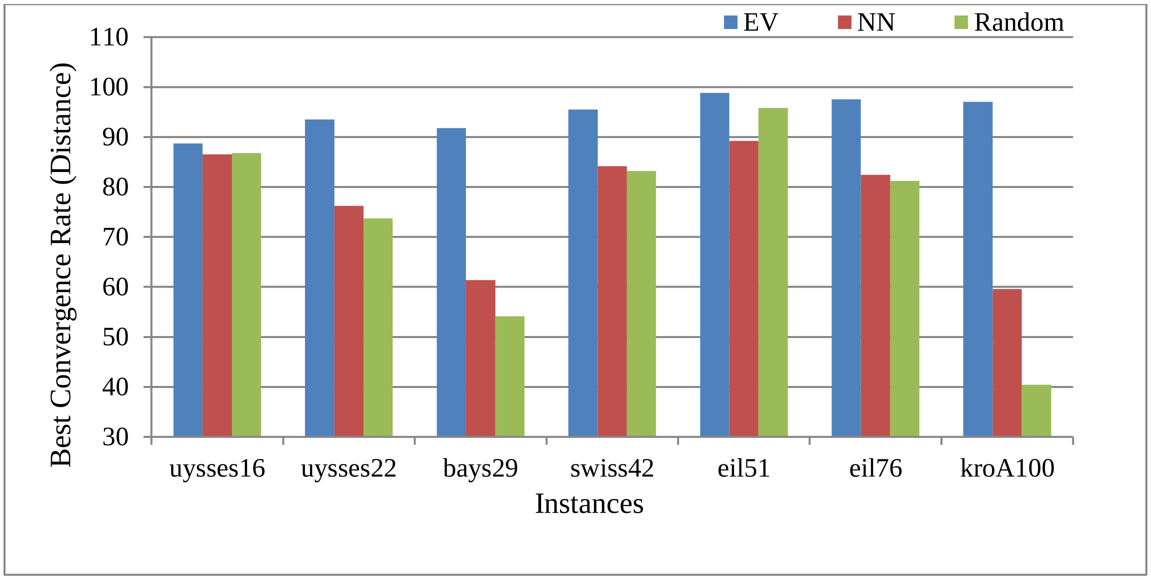


Fig.6.4.2(a) Best convergence Rate for Hybrid Optimal routing (Distance)

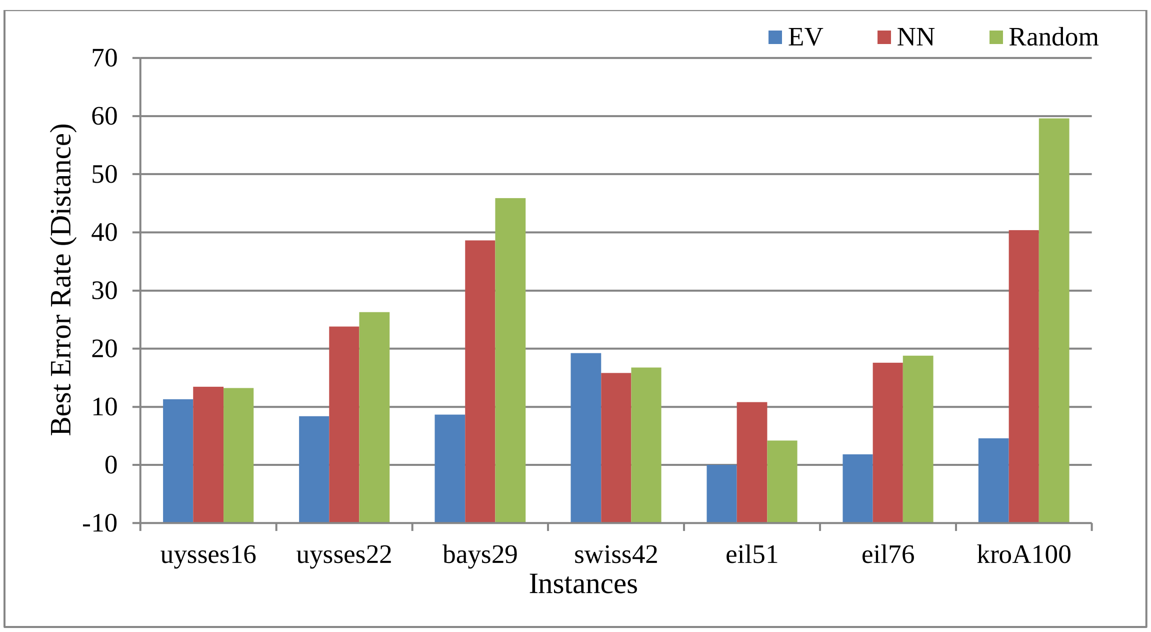


Fig.6.4.2(b). Best Error Rate for Hybrid Optimal routing (Distance)

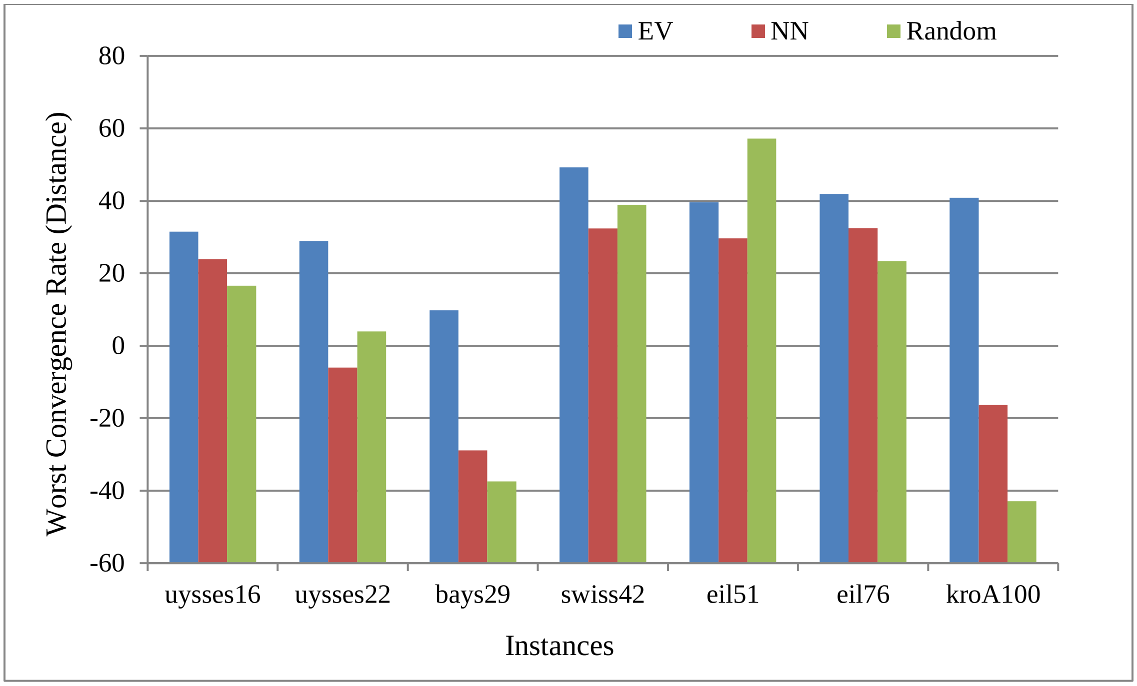


Fig.6.4.2(c) Worst Convergence Rate for Hybrid Optimal routing (Distance)

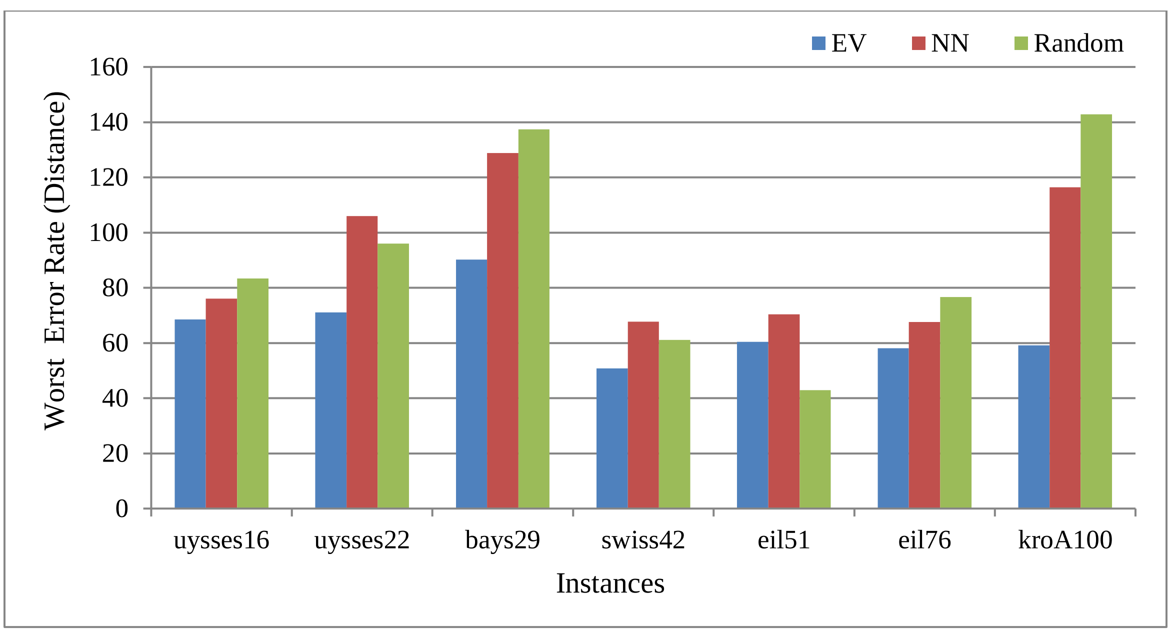


Fig.6. 4.2(d) Worst Error Rate for Hybrid Optimal routing (Distance)

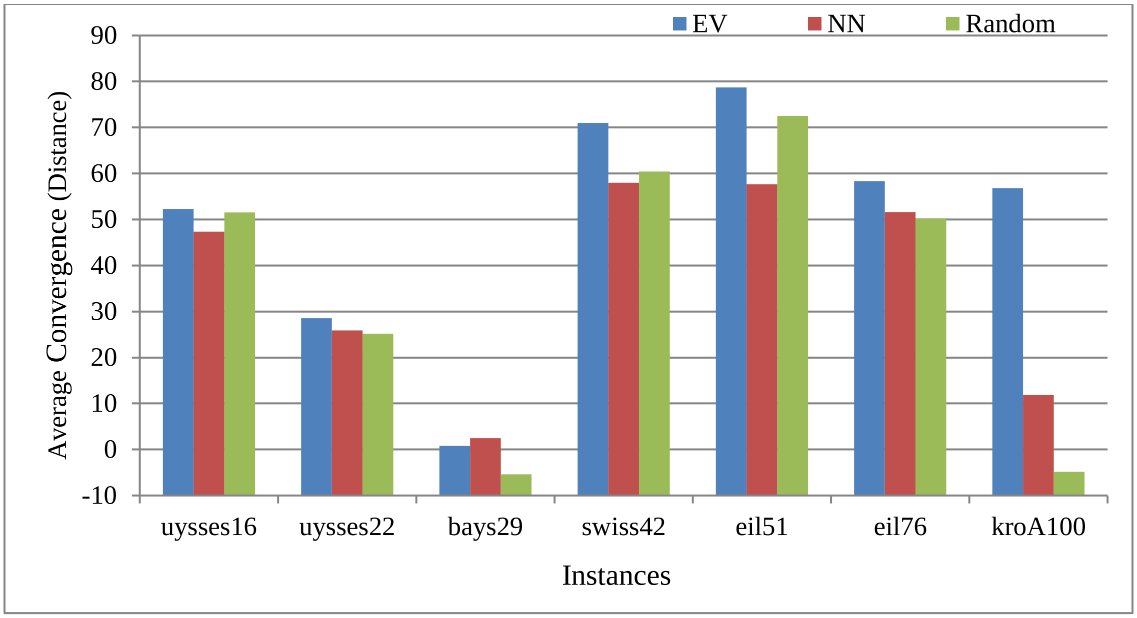


Fig.6. 4.2(e) Average Convergence Rate for Hybrid Optimal routing (Distance)

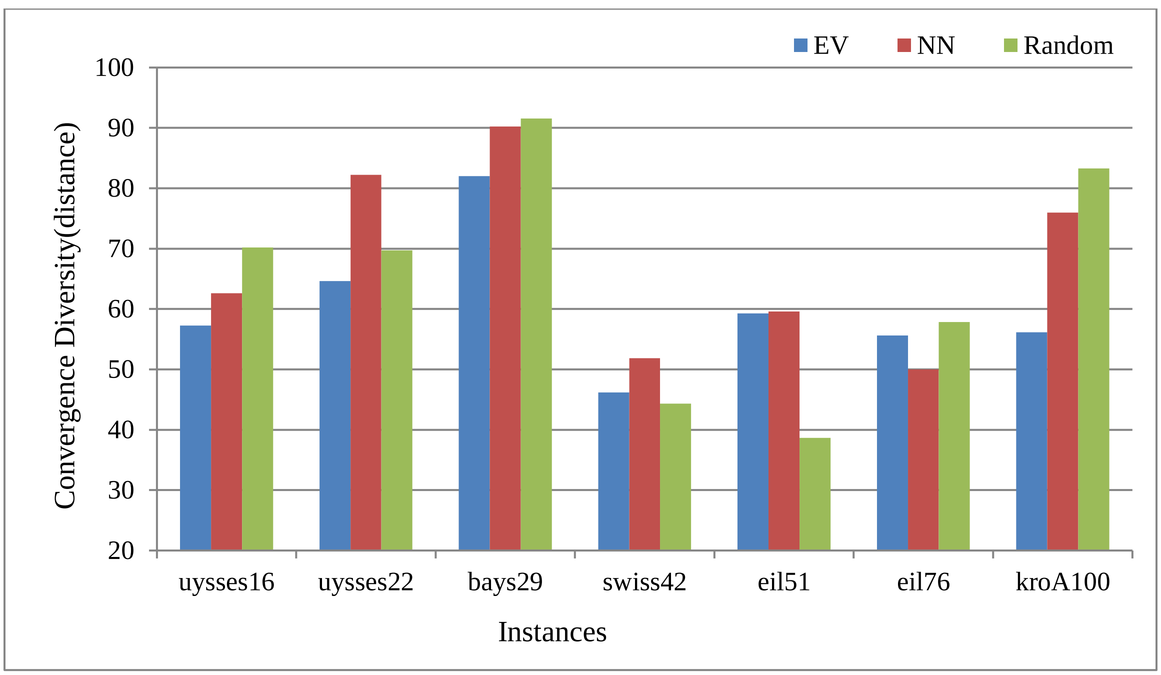


Fig.6. 4.2(f) Convergence Diversity for Hybrid Optimal routing (Distance)

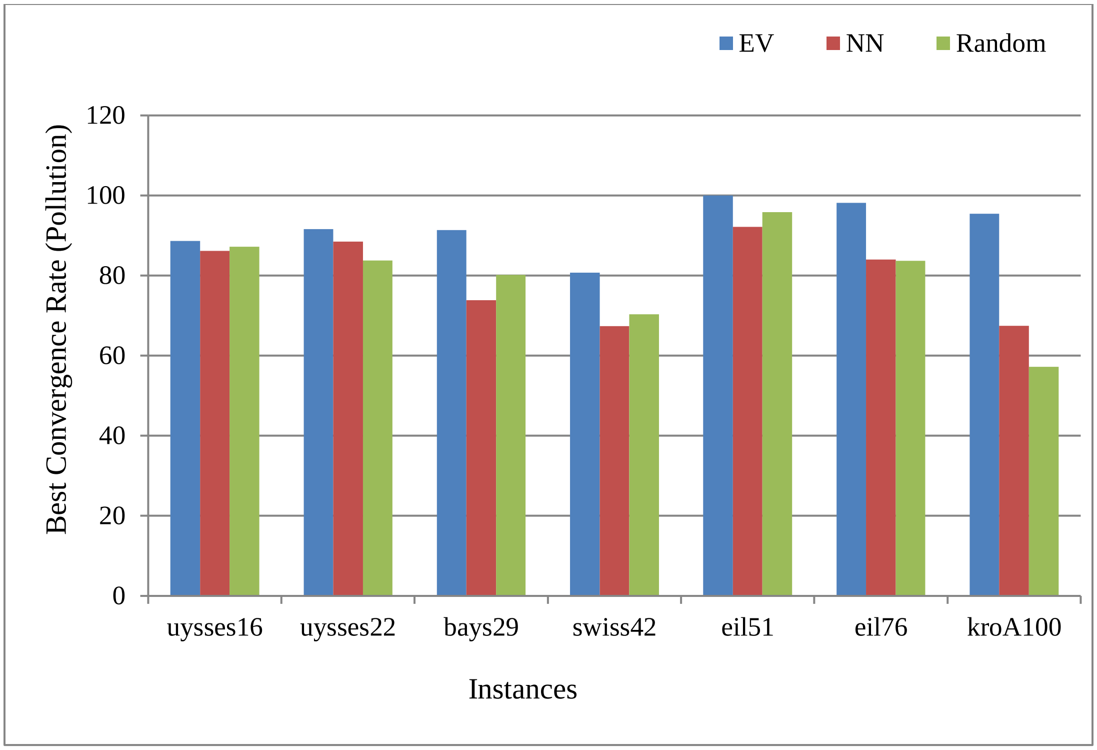


Fig.6. 4.2(g) Best convergence Rate for Hybrid Optimal routing (Pollution)

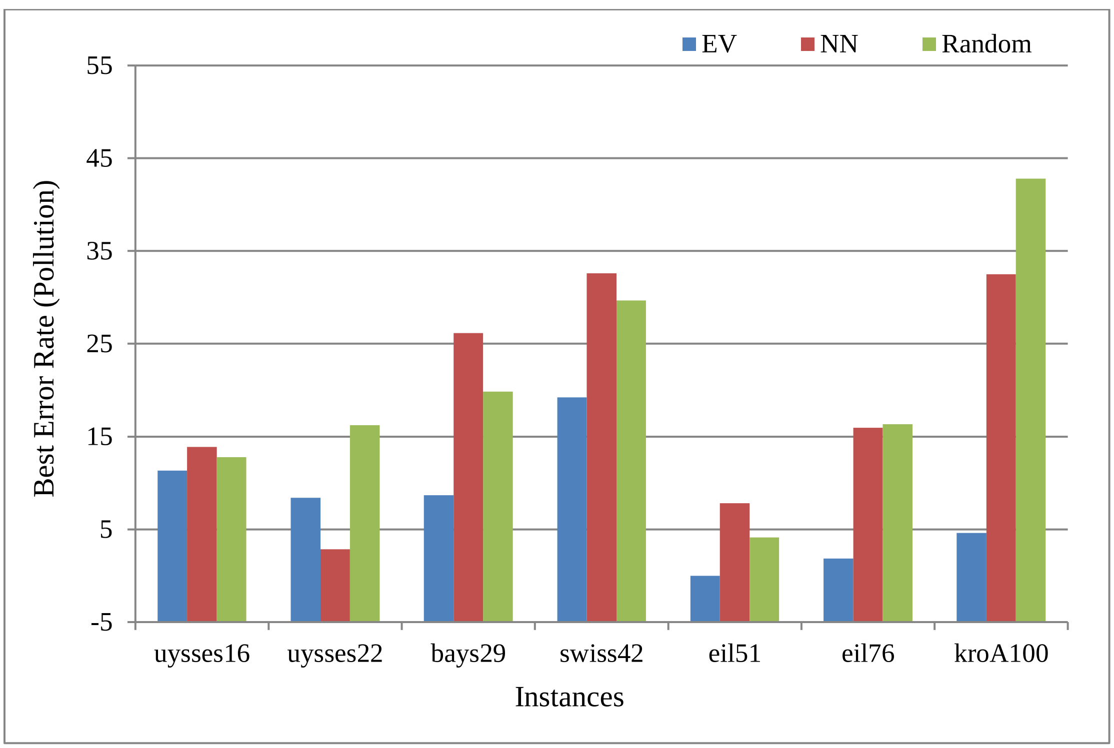


Fig.6. 4.2(h) Best Error Rate for Hybrid Optimal routing (Pollution)

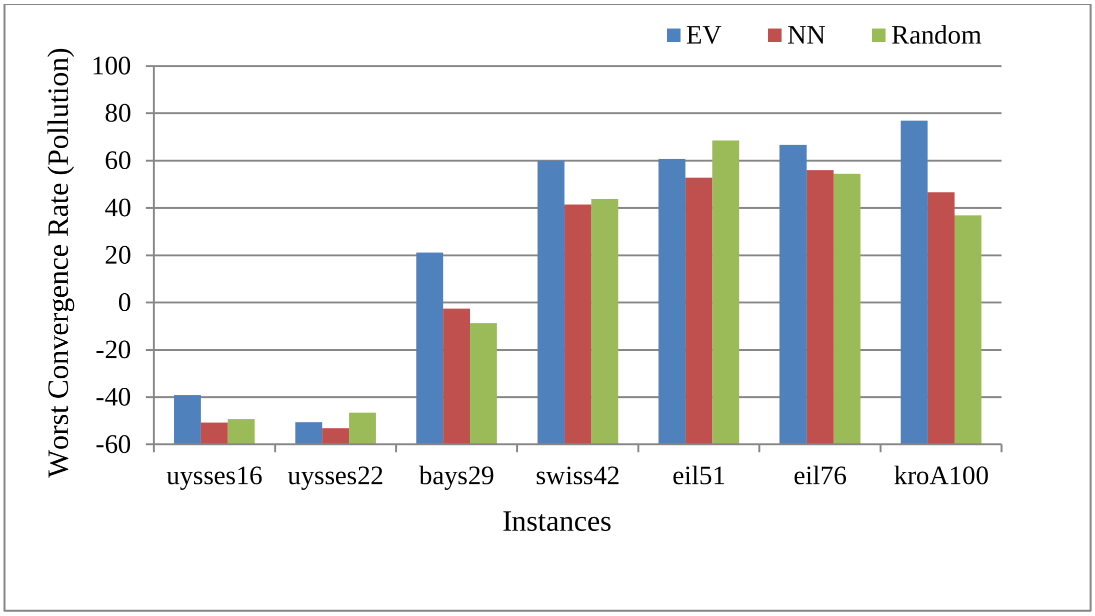


Fig.6. 4.2(i) Worst Convergence Rate for Hybrid Optimal routing (Pollution)

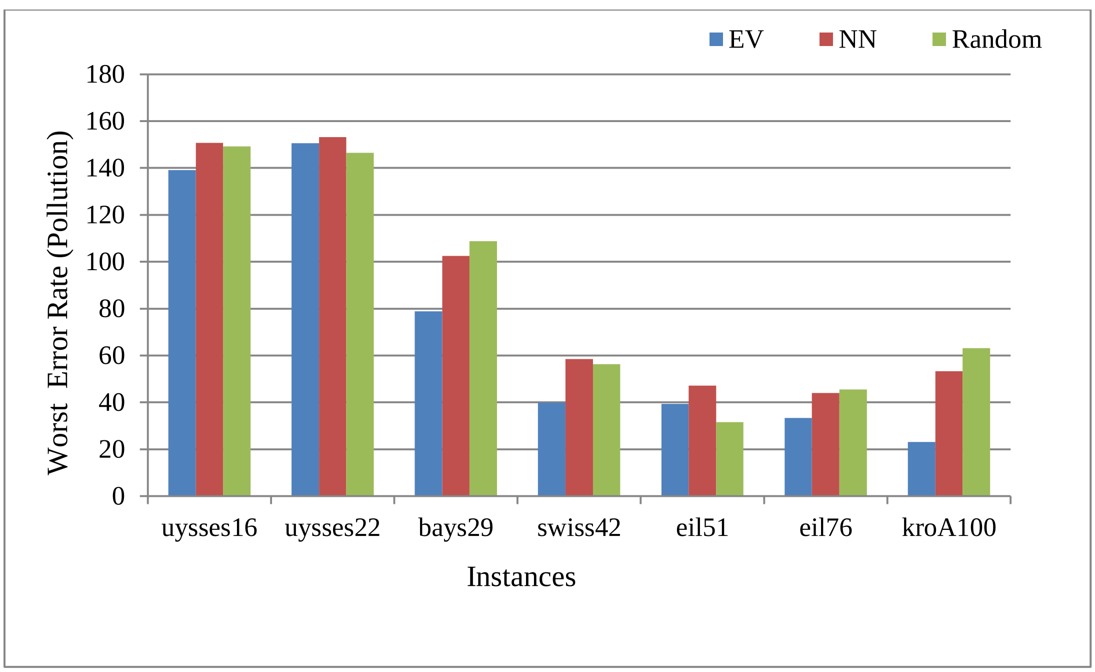


Fig.6. 4.2(j) Worst Error Rate for Hybrid Optimal routing (Pollution)

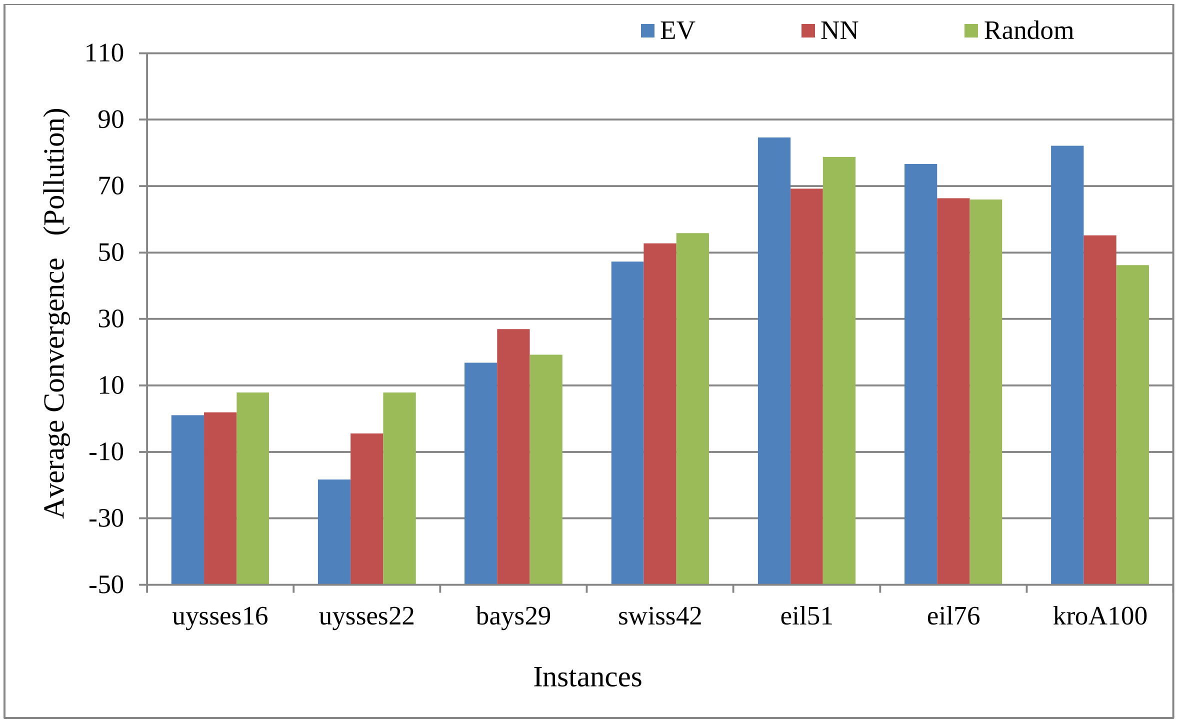


Fig.6. 4.2(k) Average Convergence Rate for Hybrid Optimal routing (Pollution)

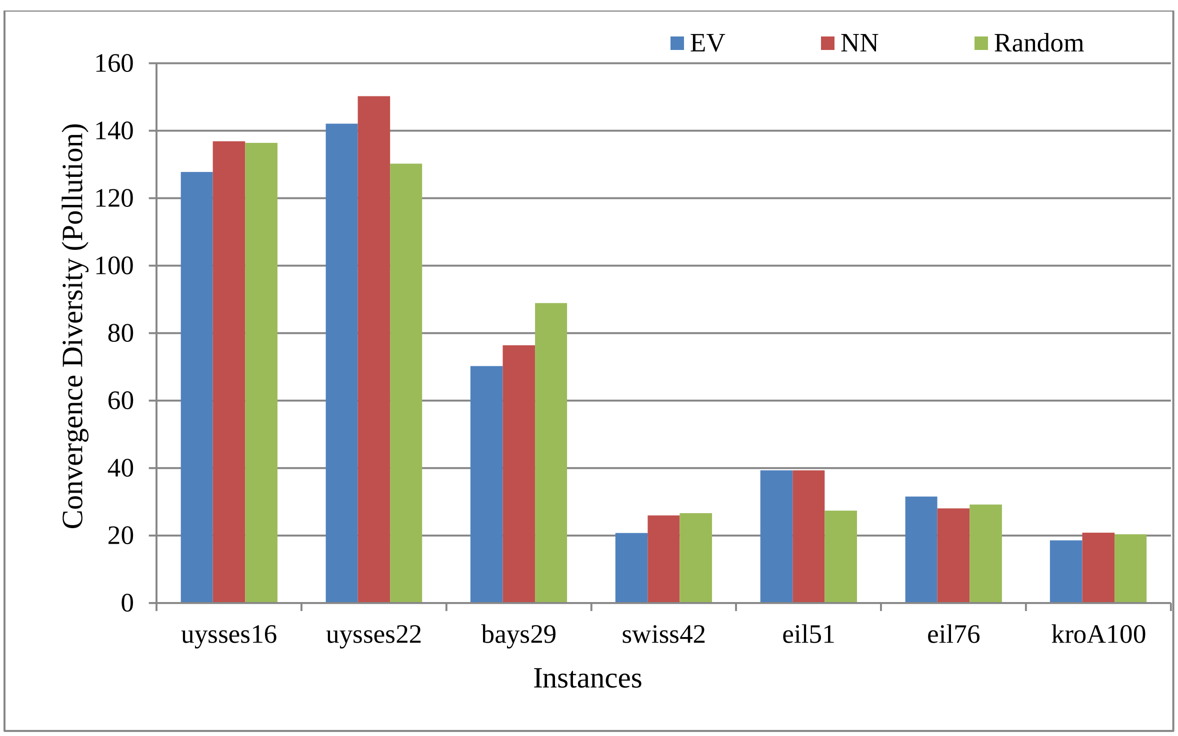


Fig.6. 4.2(l) Convergence Diversity for Hybrid Optimal routing (Pollution)

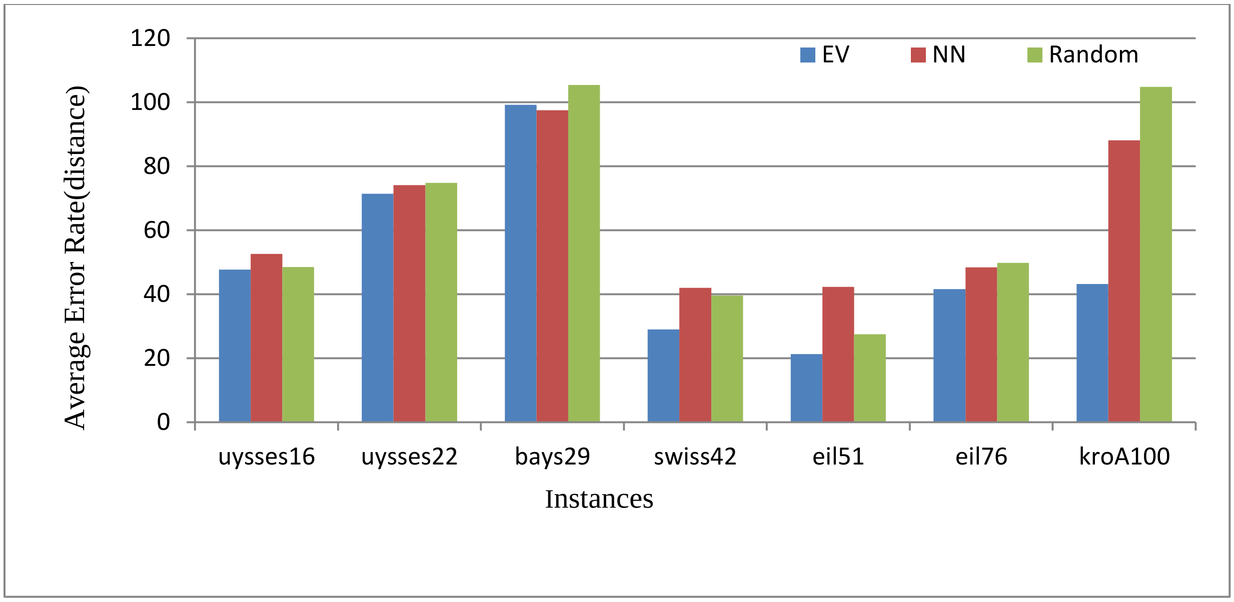


Fig.6. 4.2(m). Average Error Rate for Hybrid Optimal routing (Distance)

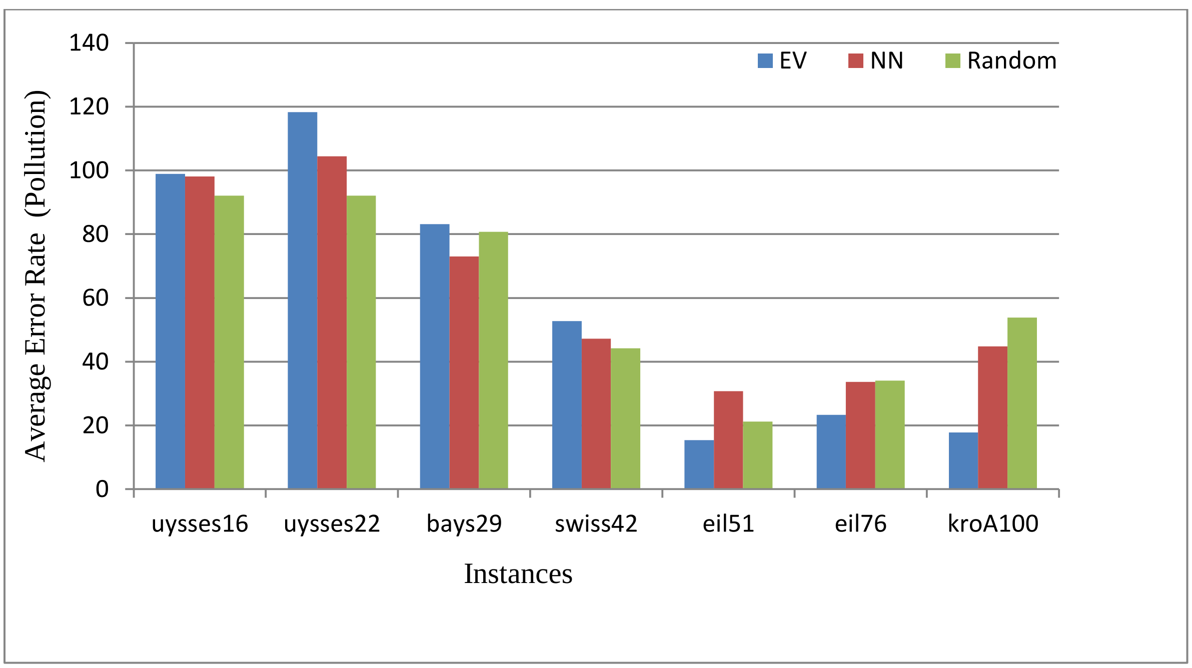


Fig.6. 4.2(n) Average Error Rate for Hybrid Optimal routing (Pollution)

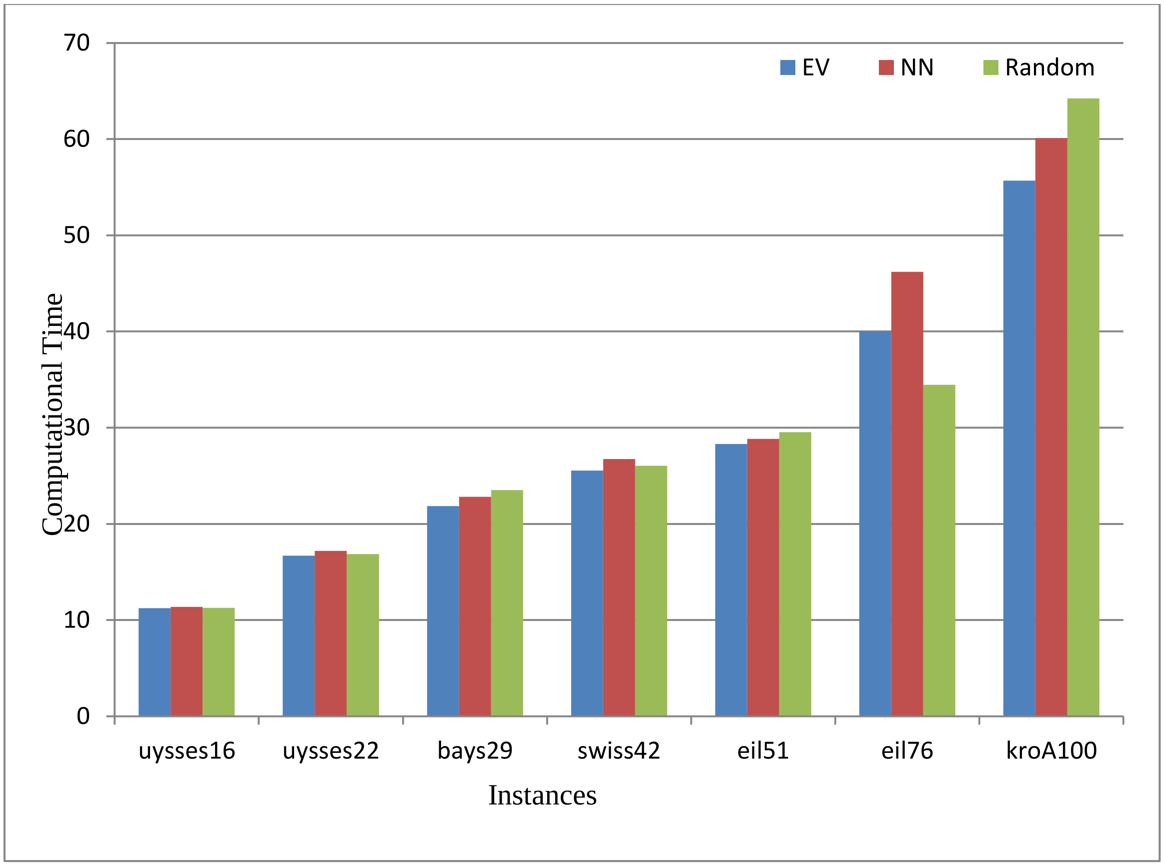


Fig.6. 4.2(o) Computational Time for Hybrid Optimal routing

Best Error rate w.r.t pollution: The Fig. 6.4.2(h) showed that the Best convergence rate is higher for every instance, which clearly states that the best error rate of ODV-EV is lower than other techniques. This openly implies the quality of the individuals in the population of the desire technique. Based on comparison and analysis of NN and Random technique, random has satisfactory performance projected in Fig 6.4.2(h).

Convergence diversity w.r.t distance: The convergence diversity is an aspect that illuminates the distribution of good and bad quality individuals among the population. It plays a vital role to increase the chance of evolving optimal solutions and to avoid premature convergence. Fig. 6.4.2(f) shows the convergence diversity of the optimal distance based routing scenario using different population seeding techniques for the problem instances. From the Fig. 6.4.2(f), it is understood that the ODV-EV technique has lesser convergence diversity w.r.t. other population seeding techniques which shows that the quality of individuals is improved as a population rather than the single individual. For most of the instances, random and NN techniques have nearly equal convergence diversity.

Convergence diversity w.r.t. Air Pollution: The convergence diversity of the air pollution based optimal routing scenario using different population seeding techniques for the problem instances in shown in the Fig. 6.4.2(l). From the Fig. 6.4.2(l), it is observed that the convergence diversity of the instances decreases with increase in the problem size despite the population technique applied.

Average convergence w.r.t distance: The Average convergence of a population is used to measure the quality of the population generated by finding the average of fitness of individuals in the population as given in Eq. 1. Fig. 6.4.2(e) shows the average convergence rate for hybrid optimal routing (distance) using different population seeding techniques for the problem instances. From the Fig. 6.4.2(e), it can be observed that every population seeding technique yields better average convergence rate for some of the large size problem instances than the small size instances. For most of the instances, the ODV-EV technique outperforms other population initialization techniques and random performs worst for the larger size instances. For the instance bays29, performance of random, NN and ODV-EV techniques are very poor; this is possibly because of the peculiarity of the instance with small size and large distance based fitness value.

Average convergence w.r.t pollution: Fig. 6.4.2(k) shows the average convergence rate for air pollution based optimal routing using different population seeding techniques for the problem instances. From the Fig. 6.4.2(k), it can be understood that average convergence rate increases with increase in the size of the problem instances regardless of the population technique used. In the case of average convergence rate, all the population seeding techniques perform nearly equal though ODV-EV technique yields marginally better result than other techniques.

Average error rate w.r.t distance: The average error rate is working better in ODV –EV technique, compare to other techniques. Average error rate of NN technique is less than the random technique is showed in Fig. 6.4.2(m). The performance of random technique is unpredictable; it shows huge variation for each problem instance, this evidently indicates that the quality of the individuals in the population is less.

Average error rate w.r.t pollution: the Average error rate w.r.t pollution, the ODV-EV technique shows high values in smaller instances and performance is increases as increase in problem instance. The Fig. 6.4.2(n) exposed, that the average convergence of NN technique showed a reasonable output for all the instances. The analysis shows that performance of NN technique in terms of average convergence is better than other techniques.

Worst Convergence rate w.r.t distance: Fig. 6.4.2(c) shows the worst convergence rate for Worst Convergence Rate for Hybrid Optimal routing (Distance) using different population seeding techniques for the problem instances. From the Fig. 6.4.2(c), it can be observed that ODV-EV technique yields better results than the NN and random technique. For the instance eil51, the random technique outperforms than the other techniques.

Worst Convergence rate w.r.t Pollution: The worst convergence rate of distance and pollution in ODV- EV technique is good, than the other two techniques. Fig. 6.4.2(i) shows the worst convergence rate for optimal pollution based routing using different population seeding techniques for the problem instances. From the Fig. 6.4.2(i), it is observed that every population seeding technique yields better, worst convergence rate for some of the large size problem instances than the small size instances.

Worst Error rate w.r.t distance: Fig. 6.4.2(d) indicate that the ODV-EV technique performing better than the NN and random technique and it got lower values for all the instances. Although the performance of ODV-EV technique showed good result in worst error rate, the random technique showed a least value for the instance eil51. For most of the instances the worst error rate of NN technique is lesser than the random technique, infers the NN performance is better than the random technique.

Worst Error rate w.r.t Pollution: As shown in the figure Fig. 6.4.2(j), it is clearly perceptible that the worst convergence rate of pollution in ODV- EV technique is virtuous, apart from the other two techniques. It has been observed that, the performance of worst convergence rate in terms of pollution in NN and random technique are inversely proportional to the performance of worst convergence rate in terms of distance. In pollution, the performance of random technique is superior to the NN technique.

6.5 SUMMARY

In summary, we investigated the three routing algorithms 1: optimal distance based routing 2: optimal pollution based routing 3: hybrid optimal based routing with different initialization techniques Random technique, NN technique and ODV-EV technique. We have been analyzed our algorithm with standard TSP bench marks and we created the corresponding Air pollution matrix, for the instances ulysses16, ulysses22, bays29, att48, eil56, eil76 and kroA100. In the result analysis, we have been analyzed our algorithm with different validation criteria's like Best convergence rate, worst convergence rate, average convergence rate, Best error rate, worst error rate and convergence diversity. The three algorithms are performed well in the ODV-EV technique for all the TSP instances. Next The NN technique is performing better in many instances than the random technique. The ODV-EV technique for optimal distance based routing yielding the best distance convergence of 95.560 % in the instance eil51 and for the optimal pollution based routing yields the best pollution convergence 98.29% in the instance eil76. The hybrid optimal based routing the best convergence rate of distance and pollution are 98.830 % and 100 % in the instance eil51. Since we are calculating the distance from the display coordinates, we are not getting 100% best convergence rate. From results we analyzed that the ODV-EV technique is performing well.

CHAPTER 7

CONCLUSIONS AND FUTURE ENHANCEMENTS

7.1 CONCLUSIONS

In this research, an intelligent routing strategy for VRP has been proposed based on the values of distance and air pollution between the corresponding cities. The proposed routing strategy uses a recent enhanced model of Genetic Algorithm to find the optimal route under two different scenarios. Experiments were performed on the benchmark TSP instances obtained from the TSPLIB and pollution matrices for the corresponding instances are formulated. The performance of two different scenarios of intelligent routing strategy for VRP is investigated using three population seeding techniques namely random, nearest neighbor and ODV based EV method. In the viewpoint of effective intelligent routing strategy for VRP, the maximum combined convergence rate obtained for the route using the Scenario 1 is 70.5% for the instance eil76 using ODV-EV population seeding technique with computation time of 42.9 seconds and for the same instance and the seeding technique, the maximum combined convergence rate is improved to 94% with 43.9 seconds of computation time in the Scenario 2.

In brief our first module objective is consider the total distance of the route" as a primary factor. The total air pollution of the derived route is considered as the secondary factor for the optimal solution selection and our experimental solution provides the best results towards the optimal distance based routing. Second module's objective is the total air pollution of the route" as a primary factor and the total distance of the derived route is considered as the secondary factor for the optimal solution selection and our new trial shows best results towards optimal pollution based routing. Our third and final objective is a Hybrid model, we consider it is Multi Objective Multi Criteria Optimization Model (Hybrid) for an Effective Integrated Routing Strategy in Transportation Systems. The results of final model shows that, we achieved optimal distance and optimal pollution route path for an effective Integrated Routing Strategy in Transportation Systems.

7.2 FUTURE ENHANCEMENTS

This research is focused on a particularly to provide optimal solution for VRP w.r.t distance and air pollution rates, which was translated into three distinct models. In this line of research, it is also possible to combine GA with SI algorithms like Ant and Bee colony and the same could be incorporated with the existing models including the proposed ones in this research. This may improve the global efficiency of the related solution models w.r.t. the problem domains.

On the other hand, it is also possible that further research work can be carried out to promote the proposed Environmental Oriented Optimization models for Vehicle Routing Problems with Vehicular Ad-Hoc Network to achieve efficient Intelligent Transportation System. EVRP with VANET may also helpful to identify the more emission vehicle. This may also helpful to reduce the Global Warming.

REFERENCES

1 Ascheuer. N, L. F. Escudero, M. Groetschel, and M. Stoer, “A cutting plane approach to the sequential ordering problem (with applications to job scheduling in manufacturing)”, SIAM Journal on Optimization, (1993), 3(1):25–42.

2 Merz. P, Freisleben. B. “Genetic local search for the TSP: new results”, Proceedings of the IEEE International Conference on Evolutionary Computation. IEEE Press, Indianapolis, (1997) IN,pp 159–164.

3 Mostepha. R, Khouadjiaa, BriseidaSarasolab, Enrique Albab, LaetitiaJourdana, El GhazaliTalbia, “A comparative study between dynamic adapted PSO and VNS for the vehicle routing problem with dynamic requests”, Applied Soft Computing 12 (2012) 1426–1439.

4 Yannis Marinakisa, Magdalene Marinakib, GeorgiosDouniasc, “A hybrid particle swarm optimization algorithm for the vehicle routing problem”, Engineering Applications of Artificial Intelligence 23 (2010) 463–472.

5 YannisMarinakis, Georgia-RoumbiniIordanidou, Magdalene Marinaki, “Particle Swarm Optimization for the Vehicle Routing Problem with Stochastic Demands”, Applied Soft Computing 13 (2013) 1693–1704

6 “The traveling salesman problem: a computational study”, (2007), Princeton (NJ): Princeton University.

7 Albayrak, N. Allahverdi, “Development a new mutation operator to solve the traveling salesman problem by aid of genetic algorithms”, Expert Systems with Applications (2011) 1313–1320.

8 Anand Subramanian. A, Eduardo Uchoab, Luiz Satoru Ochi c, “A hybrid algorithm for a class of vehicle routing problems”, Computers& Operations Research 40 (2013) 2519–253.

9 Aras N, Oommen BJ, Altinel IK, “Kohonen Network incorporating explicit statistics and its application to the traveling salesman problem”, Neural Networks, Volume 12 Issue 9, Nov.1999, pp. 1273 – 1284.

10 Ascheuer, “Hamiltonian path problems in the on-line optimization of flexible manufacturing systems”, Technical Report 3, Konrad-Zuse-Zentrum, Information’s tech Berlin, Heilbronner Str. 10, D-10711, (1996), Berlin - Wilmersdorf.

11 Bak, Per, “How Nature Works: The Science of Self-Organized Criticality”, Copernicus Press, New York 1996, XIII, pp. 212.

12 Balling RJ, Briggs RR, Gillman K, “Multiple optimum size/shape/topology designs for skeletal structures using a genetic algorithm”, Journal of Structural Engineering, Vol. 132, No. 7, July 2006, pp. 1158-1165, ISSN: 1943-541X.

13 Bernard Angeniol, Gael de La Croix Vaubois, Jean-Yves Le Texier, “Self-organizing feature maps and the traveling salesman problem”, Neural Networks, Volume 1, Issue 4, 1988, pp. 289–293.

14 Berrettini. E, D'Angelo. G, Delling. D, “Arc-Flags in Dynamic Graphs”, In 9th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems (ATMOS 2009), Volume 09002 of Dagstuhl Seminar Proceedings, Germany.

15 Bianchi L, Birattari M, Manfrin M , “Hybrid metaheuristics for the vehicle routing problem with stochastic demands”, J Math Model Algorithms (2006), 5(1):91–110.

16 Chang Wook Ahn, Ki Pyo Kim, R. S. Ramakrishna, “A memory-efficient elitist genetic algorithm”, Parallel Processing and Applied Mathematics, Volume 3019, 2004, pp. 552-559.

17 D. Feillet, P. Dejax, M. Gendreau and C. Gueguen, “An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problem, Network, (2004), 44 (3) :216 – 229.

18 D. Whitley, D. Hains, A.E. Howe, "A hybrid genetic algorithm for the traveling salesman problem using generalized partition crossover", Parallel Problem Solving from Nature, PPSN XI, Volume 6238, 2010, pp. 566-575.

19 David L. Applegate, Robert E. Bixby, Vasek Chvátal & William J. Cook,

20 Desaulniers, J. Desrosiers, A. Erdmann, M.M. Solomon, F. Soumis, "The VRP with Pickup and Delivery", (2002), SIAM, Philadelphia, 225-242.

21 Dimitris Fotakis ,Epameinondas Sidiropoulos, "A new multi-objective self-organizing optimization algorithm (MOSOA) for spatial optimization problems", Applied Mathematics and Computation, Volume 218, Issue 9, January 2012, pp. 5168–5180.

22 Donati AV, Montemanni R, Casagrande N et al., "Time dependent vehicle routing problem with a multi ant colony system", Euro Journal of research (2008) 185(3):1174–1191.

23 Eisuke Kita, ShenKan, ZhaiFei, "Investigation of self-organizing map for genetic algorithm", Advances in Engineering Software, Volume 41, Issue 2, February 2010, pp. 148–153.

24 Eyckelhof CJ, Snoek M, "Ant systems for a dynamic TSP: Ants caught in a traffic jam", Volume 2463, Ant Algorithms: 3rd International Workshop, ANTS 2002, Lecture Notes in Computer Science. Heidelberg: Springer; 2002.pp. 88–99.

25 Favata S, Walker R, "A study of the application of Kohonen-type neural networks to the traveling salesman problem", Biological Cybernetics, April 1991, Volume 64, Issue 6, pp. 463-468.

26 Fogarty T. C, Huang R, "Implementing the genetic Algorithm on Transputer Based Parallel Processing Systems", Parallel Problem Solving from Nature, (1991), p. 145–149.

27 Fort JC, "Solving combinatorial problem via self-organizing process: an application of the Kohonen algorithm to the traveling sales-man problem", Biological Cybernetics, June 1988, Volume 59, Issue 1, pp. 33-40.

28 Garey. M. R, D. S. Johnson, R. Sethi, "The complexity of flowshop and jobshop scheduling", Mathematics of Operations Research, (1976), 2(2):117–129.

29 Ghiani. G, F. Guerriero, G. Laporte, R. Musmanno, "Real-time vehicle routing: solution concepts, algorithms and parallel computing strategies", European Journal of Operational Research 151 (2003) 1–11.

30 Goldberg, A.V., Kaplan, H., Werneck, R.F. "An Efficient Point-to-Point Shortest Path Algorithms. In Workshop on Algorithm Engineering and Experiments (ALNEX), (2006), 129–143.

31 Gorges-Schleuter M, "Explicit Parallelism of genetic Algorithms through Population Structures", Parallel Problem Solving from Nature Lecture Notes in Computer Science Volume 496, 1991, pp 150-159.

32 Gorges-Schleuter. M, Asparagos96 and the traveling salesman problem. Proceedings of IEEE International Conference on Evolutionary Computation. IEEE Press, Indianapolis, IN, (1997), pp. 171–174.

33 Grosso P. B, "Computer simulations of genetic adaptation: Parallel subcomponent interaction in a multilocus model", Doctoral dissertation, The University of Michigan, 1985. (University Microfilms No. 8520908).

34 H. Al-Mulhem, T. Al-Maghrabi, "Efficient convex-elastic net algorithm to solve the Euclidean traveling salesman problem", IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, 1998;28(4), pp. 618-20.

35 H. Li, A. Lim, "A Metaheuristic for the Pickup and Delivery Problem with Time Windows", The 13th IEEE Conference on Tools with Artificial Intelligence, ICTAI-2001, Dallas, USA, 160-170

36 Han J, "Local Evaluation Functions and Global Evaluation Functions for Computational Evolution", doi: SFI-WP 03-09-048.

37 Han, J, Q.S. Cai, "Emergence from Local Evaluation Function", Journal of Systems Science and Complexity, (2003), 16(3): 372–390.

38 Hauser R, Männer R, "Implementation of standard genetic algorithm on MIMD machines", Parallel Problem Solving from Nature — PPSN III, Lecture Notes in Computer Science Volume 866, 1994, pp 503-513.

39 Helsgaun. "An effective implementation of the Lin–Kernighan traveling salesman heuristic", European Journal of Operational Research 126 (1) (2000) 106–130.

40 Heni Ben Amor, "Intelligent Exploration for Genetic Algorithms Using Self Organizing Maps in Evolutionary Computation", GECCO'05, June 25–29, 2005, Washington, DC, pp. 1531-1538.

41 Hong and Y. B. Park, "A heuristic for bi-objective vehicle routing with time window constraints", International Journal of Production Economics, vol.62, no.3, (1999), pp.249-258.

42 Hueter, G.J, "Solution of the traveling salesman problem with an adaptive ring", IEEE International Conference on Neural Networks, vol., no., pp.85,92 vol.1, 24-27 July 1988, doi: 10.1109/ICNN.1988.23832.

43 Irnich and D. Villeneuve. "The shortest path problem with resource constraints and k-cycle elimination for $k \geq 3$", Technical Report G-2003-55, GERAD, (2003), Montreal, Canada.

44 J. Lysgaard, A. N. Letchford, and R. W. Eglese. "A new branch-and-cut algorithm for the capacitated vehicle routing problem", Mathematical Programming, Ser. A, (2004), 100:423–445.

45 J.-F. Cordeau and G. Laporte. "A Tabu Search Algorithm For The Site Dependent Vehicle Routing Problem With Time Windows", (2001), INFOR, 39:292–298.

46 J.-F. Cordeau, G. Laporte, and A. Mercier. "A unified tabu search heuristic for vehicle routing problems with time windows", Journal of the Operational Research Society, (2000), 52:928–936.

47 J.-F. Cordeau, M. Gendreau, A. Hertz, G. Laporte, and J.-S. Sormany. “New heuristics for the vehicle routing problem”, Technical Report G-2004-33, GERAD, Montreal, Canada.

48 J.-F. Cordeau, M. Gendreau, and G. Laporte. “A tabu search heuristic for periodic and multi-depot vehicle routing problems”, Networks, (1997), 30:105–119.

49 J.S. Lawrence, M.M. Amini, “Performance characteristics of alternative genetic algorithmic approaches to the traveling salesman problem using path representation: an empirical study”, European Journal of Operational Research, April 1998, pp. 551–570.

50 Jean-François Cordeaua, MirkoMaischbergerb, “A parallel iterated tabu search heuristic for vehicle routing problems”, Computers & Operations Research 39 (2012) 2033–205

51 Jepsen, S. Spoorendonk and B. Petersen, “A branch- and- cut - and- price framework for VRP applied on CVRP and VRPTW, Master’s Thesis, DIKU Dept. of Computer Science, (2005), University of Copenhagen.

52 John H. Holland, “Adaptation in Natural and Artificial Systems”, MIT Press Cambridge, MA, USA (1992).

53 K. Katayama, H. Sakamoto, H. Narihisa, “The efficiency of hybrid mutation Genetic Algorithm for the Travelling Salesman Problem”, Mathematical and Computer Modeling, Elsevier, Issue 31, October, 2000, pp. 197–203.

54 K. Socha and M. Dorigo, “Ant colony optimization for continuous domains”, European Journal of Operations Research, (2008), 185(3):1155–1173.

55 K. Socha, “ACO for continuous and mixed-variable optimization, Ant Colony Optimization and Swarm Intelligence”, 4th International Work- shop, (2004), ANTS.

56 Kallehauge, J. Larsen, and O. B. G. Madsen, “Lagrangean duality applied on vehicle routing with time windows” - experimental results. Technical Report IMM-REP-2000-8, Informatics and Mathematical Modelling, Technical University of Denmark.

57 Kennedy. J, Eberhart. R. “Particle swarm optimization”, Proceedings of the IEEE International Conference on Neural Networks 4, (1995)1942–1948.

58 Knopp. S, Sanders. P, Schultes. D, Schulz. F, Wagner. D, “Computing Many-to-Many Shortest Paths Using Highway Hierarchies”, In Proceedings of the Workshop on Algorithm Engineering and Experiments, ALENEX 2007, New Orleans, Louisiana, USA, 36–45.

59 Koehler, E., Moehring, R. H., Schilling, H. “Fast Point-to-Point Shortest Path Computations with Arc-Flags”, In Proc. of 9th DIMACS (2006) Implementation Challenge – Shortest Paths.

60 KusumDeep, Dipti, “A self-organizing migrating genetic algorithm for constrained optimization”, Applied Mathematics and Computation, Volume 198, Issue 1,, 15thApril 2008, pp. 237–250.

61 Kwon Jeong, Jang Lee, “A self-organizing genetic algorithm for multimodal function optimizer”, Artif Life and Robotics, 1998, Volume 2, Issue 1, pp. 48-52.

62 Kwong-SakLeunga, Hui-Dong Jinb, Zong-Ben Xuc, “An expanding self-organizing neural network for the traveling salesman problem”, Neurocomputing, Volume 62, December 2004, pp. 267–292.

63 Laporte and F. Semet. Classical heuristics for the capacitated vrp. In P. Toth and D. Vigo, ed-itors, The Vehicle Routing Problem, volume 9 of SIAM Monographs on Discrete Mathematics and Applications, chapter 5, pages 109–128. SIAM, Philidelphia, 2002.

64 Lawler EL, Lenstra JK. Kan AHGR, “The travelling salesman problem”, (1985), Chichester: John Wiley & Sons, Ltd.

65 Leila Kallel, MarcSchoenauer, “Alternative Random Initialization in genetic algorithm”, Proceedings of the 7th International Conference on Genetic Algorithms, 1997.

66 Ling-NingXing , Ying-Wu Chen, Ke-Wei Yang, FengHou, Xue-Shi Shen, Huai-Ping Cai. “A hybrid approach combining an improved genetic algorithm and optimization strategies for the asymmetric traveling salesman problem, January 2008.

67 Liu J, Y.Y. Tang, Y. C. Cao, “An Evolutionary Autonomous Agents Approach to Image Feature Extraction”, IEEE Trans. on Evolutionary Computation, (1997) 1(2): pp.141–158.

68 Liu, J., and K.C. Tsui, “Toward Nature-Inspired Computing”, Communications of the ACM, 49(10): 59 – 64.

69 M. Budinich, “A self-organizing neural network for the traveling salesman problem that is competitive with simulated annealing”, Neural Computation, Volume 8 Issue 2, February, 1996, pp.416-42.

70 M. Gendreau, F. Guertin, J.-Y. Potvin, R. Séguin, “Neighborhood Search Heuristics for a Dynamic Vehicle Dispatching Problem with Pick-ups and Deliveries”, Tech. Rep. CRT-98-10, Centre de Recherche sur les Transport, University de Montreal.

71 Marinakis. Y, Migdalas. A, Pardalos. P.M. “A hybrid genetic—GRASP algorithm using Lagrangian relaxation for the traveling salesman problem”, Journal of Combinatorial Optimization 10, (2005b) 311–326.

72 Marinakis. Y, Migdalas. A, Pardalos. P.M. “Expanding neighborhood GRASP for the traveling salesman problem”, Computational Optimization and Applications 32, (2005a) 231–257.

73 Marinakis. Y, Migdalas. A, Pardalos. P.M. “Multiple phase neighborhood search GRASP based on Lagrangian relaxation and random backtracking Lin Kernighan for the traveling salesman problem”, Journal of Combinatorial Optimization 17, (2009) 134–156.

74 Martin Josef Geiger. “Fast Approximation Heuristics for Multi-Objective Vehicle Routing Problems”, pp. 441–450, 2010. Springer-Verlag Berlin Heidelberg.

75 Milthers. N.P.M. “Solving VRP using Voronoi Diagrams and Adaptive Large Neighborhood Search”, Master’s thesis, University of Copenhagen, (2009), Denmark.

76 Montemanni R, Gambardella LM, Rizzoli AE et al., “Ant colony system for a dynamic vehicle routing problem”, J Comb Optim (2005), 10: 327–343.

77 Moon. C, J. Kim, G. Choi, and Y. Seo, “An efficient genetic algorithm for the traveling salesman problem with precedence constraints”, European Journal of Operational Research, (2002) 140(3): 606-617.

78 Norman Fenton, James Bieman, “Software Metrics: A Rigorous and Practical Approach”, Third Edition, CRC Press, 2013.

79 Norstad. I, Fagerholt. K, Laporte. G, “Tramp ship routing and scheduling with speed optimization”, Transportation Research Part C: Emerging Technologies 19 (5), (2010), 853–865.

80 O. Yugay, I. Kim, B. Kim, F.I.S. Ko, “Hybrid genetic algorithm for solving Traveling Salesman Problem with sorted population”, Third IEEE International Conference on Convergence and Hybrid Information Technology, 2008.

81 P. Toth and D. Vigo, “The Vehicle Routing Problem with Time Windows ”, volume 9 of SIAM Monographs on Discrete Mathematics and Applications, chapter 7, (2002) pages 157–193. SIAM, Philadelphia.

82 P. Victer Paul, A. Ramalingam, R. Baskaran, P. Dhavachelvan, K. Vivekanandan, R. Subramanian, “A new population seeding technique for permutation-coded Genetic Algorithm: Service transfer approach”, Journal of Computational Science, Volume 5, Issue 2, March 2014, Pages 277-297.

83 Pisinger. D, Ropke. S. “A General Heuristic for Vehicle Routing Problems’, Technical Report, DIKU - Department of Computer Science, University of Copenhagen, 2005.

84 Pisinger. D, Ropke. S. “General Heuristic for Vehicle Routing Problems”, Computers & Operations Research 34 (8), (2007) 2403–2435.

85 R. Bent and P. V. Hentenryck, “A two-stage hybrid local search for the vehicle routing problem with time windows”, 2004. Transportation Science.

86 R. Bent, P. Van Hentenryck, “A Two-Stage Hybrid Algorithm for Pickup and Delivery Vehicle Routing Problems with Time Windows”, Proceedings of the International Conference on Constraint Programming (CP-2003), Lecture Notes in Computer Science 2833, 123-137.

87 R. Fukasawa, J. Lysgaard, M. P. de Aragão, M. Reis, E. Uchoa, and R. F. Werneck. “Robust branch-and-cut-and-price for the capacitated vehicle routing problem”, In Proceedings of IPCO X, (2004), Columbia University.

88 R. Geetha, “Ant Colony Optimization in Diverse Engineering Applications: an Overview”, IJCA, (2012), vol. 49, no. 17.

89 Rae-Goo Kang and Chai-Yeoung Jung, “The Improved Initialization Method of Genetic Algorithm for Solving the Optimization Problem”, Neural Information Processing, Volume 4234, 2006, pp. 789-796.

90 Ren,Y, Dessouky. M, & Ordo9ñez. F. (2010). “The multi-shift vehicle routing problem with overtime.” Computers and Operations Research, 37, 1987–1998.

91 Renato Tinos and Shengxiang Yang, “Genetic Algorithms with Self-Organizing Behaviour in Dynamic Environments”, Studies in Computational Intelligence, Volume 51, 2007, pp. 105-127.

92 Renato Tinos, Shengxiang Yang, “A Self-Organizing Random Immigrants Genetic Algorithm for Dynamic Optimization Problems”, Genetic Programming and Evolvable Machines, September 2007, Volume 8, Issue 3, pp. 255-286.

93 Romeijn. H. E, Smith. R. L, “Parallel Algorithms for Solving Aggregated Shortest Path Problems”, Computers & Operations Research, (1999), (26), 941–953.

94 Ropke. S, Pisinger. D. “An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows”, Transportation Science 40 (4), (2006a) 455–472.

95 Ross. B.J. "A Lamarckian evolution strategy for genetic algorithms." In: Chambers, L.D (Ed.), Practical Handbook of Genetic Algorithms: Complex Coding Systems, vol. 3. CRC Press, Boca Raton, FL, (1999), pp. 1–16.

96 S. Abe, "Convergence acceleration of the Hopfield neural network by optimization integration step sizes", IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, Volume 26 Issue 1, February 1996, pp. 194-201.

97 S. Gelinas, M. Desrochers, J. Desrosiers, M.M. Solomon., "A new branching strategy for time constrained routing problems with application to backhauling", Annals of Operations Research, (1995), 61 91–109.

98 Schoonderwoerd R, Holland O, Bruten J, "Ant-based load balancing in telecommunications networks", Adapt Behav, (1996), 5(2):169–207.

99 Scott Camazine, Jean-Louis Deneubourg, Nigel R. Franks, James Sneyd, Guy Theraulaz, Eric Bonabeau, "Self-Organizing in Biological Systems", Princeton University Press: New Jersey, pp 560, 2003.

100 Seeley T. D, "When Is Self-Organizing Used in Biological Systems", Biol. Bull. 202, pp.314–318, (June 2002).

101 Shubhra Sankar Ray, Sanghamitra Bandyopadhyay, Sankar K. Pal, "Genetic operators for combinatorial optimization in TSP and microarray gene ordering", Applied Intelligence, June 2007, Volume 26, Issue 3, pp. 183-195.

102 Shubhra, B. Sanghamitra, K.P. Sankar, "Genetic operators for combinatorial optimization in TSP and microarray gene ordering", Journal of Applied Intelligence (2007) 183 195, Kluwer Academic Publishers

103 Somhom S, Modares A, EnkawaT, "Competition-based neural network for the multiple traveling salesman problem with minmax objective", Computer Res 1999, pp. 395–407.

104 Subramanian A, Penna PHV, Uchoa E, Ochi LS. “A hybrid algorithm for the heterogenous fleet vehicle routing problem”, European Journal of Operational Research (2012), 221:285–95.

105 Sumathi. S, Surekha P, “Computational Intelligence Paradigms Theory and Applications Using MATLAB”, CRC (2010), Boca Raton, FL, USA.

106 T. Kohonen, M. R. Schroeder, and T. S. Huang, “Self-Organizing Maps”, Springer-Verlag New York, Inc. Secaucus, NJ, USA (2001).

107 Thierry Garaixa, Christian Artiguesb, Dominique Feilletc, Didier Josselind, “Vehicle routing problems with alternative paths: An application to on-demand transportation.” European Journal of Operational Research 204 (2010) 62–75.

108 Thomas Stützle, Holger H. Hoos, “MAX–MIN Ant System”, Future Generation Computer Systems, Volume 16, Issue 8, June 2000, 889-914,

109 Togan V, Daloglu AT, “Adaptive approaches in genetic algorithms to catch the global optimum”, Proc. 7th international congress advances in civil engineering, Istanbul, 2006.

110 Togan V, Daloglu AT, “Optimization of 3d trusses with adaptive approach in genetic algorithms”, Engineering Structures, Volume 28, Issue 7, June 2006, pp. 1019–1027.

111 V. Togan, A.T. Daloglu, “An improved genetic algorithm with initial population strategy and self-adaptive member grouping”, Journal of Computers and Structures, June 2008, pp. 1204–1218.

112 Victer Paula, A.Ramalingamb, R. Baskaranc, P. Dhavachelvana, K. Vivekanandand, R. Subramanian a, “A new population seeding technique for permutation-coded Genetic Algorithm: Service transfer approach”, http://dx.doi.org/10.1016/j.jocs.2013.05.009.

113 Vose MD. “The simple genetic algorithm: foundations and theory”, Cambridge, MA: MIT (1998).

114 W. Cook and J. L. Rich. “A parallel cutting-plane algorithm for the vehicle routing problem with time windows.” Technical Report TR99-04, Department of Computational and Applied Mathematics, (1999), Rice University.

115 Winter G, Périaux J, Galaán M, Cuesta P. “Genetic algorithms in engineering and computer science”, New York: Wiley, (1995).

116 Y. Lu, S. Lu, F. Fotouhi, Y. Deng, S. Brown, “Incremental genetic K-means algorithm and its application in Gene Expression Data Analysis”, BMC Bioinformatics, 2004, 5:172.

117 YiyoKuoa, Chi-Chang Wang b. “A variable neighborhood search for the multi-depot vehicle routing problem with loading cost”, Expert Systems with Applications 39 (2012) 6949–6954

118 Zhang Jinhua , ZhuangJian, Du Haifeng, Wang Sun’an, “Self-organizing genetic algorithm based tuning of PID controllers”, Information Sciences, Volume 179, Issue 7, March 2009, pp. 1007–1018.

List of Publication

International Journals

1. N. Saravanan, R. Baskaran, M. Shanmugam and M.S. Saleem Basha, "An Effective Model for QoS Assessment in Data Caching in MANET Environments", International Journal of Wireless and Mobile Computing, Inderscience Publishers, Vol.6 Iss.5, pp.515-527. (2013) (SCOPUS INDEXED)
2. M.Shanmugam, M.S. SaleemBasha, P.Dhavachalvan, “A Study on Communication Protocols and Applications in VANET”, International Journal of Applied Engineering Research, Vol.8 Iss.10, pp.1185-1204. (2013) (SCOPUS INDEXED)
3. M.Shanmugam, M.S. Saleem Basha, P. Dhavachelvan and R. Baskaran, “Performance Assessment over Heuristic Population Seeding Techniques of Genetic Algorithm: Benchmark Analyses on Traveling Salesman Problems”, International Journal of Applied Engineering Research, Research India Publications, Vol.8 Isss.10, pp.1171-1183. (2013) (SCOPUS INDEXED)
4. M.Shanmugam, M.S. SaleemBasha, “DDoS Attack Traceback and Chaos in a Distributed Network a Survey”, International Journal of Applied Engineering Research, Vol.8 Iss.10, pp.1159-1169. (2013) (SCOPUS INDEXED)
5. M.Shanmugam, M.S. Saleem Basha, “Intelligent Routing Strategies for Vehicle Routing Problems using Enhanced Genetic Algorithms”. Springer Publication, Communicated, ISSN Online: 1319-8025, (2014). (SCOPUS INDEXED).

International conferences

1. M. Shanmugam, M.S. Saleem Basha, D. Chandramohan, "SVIP-enhanced security mechanism for SIP based VoIP systems and its issues", Advances in Intelligent Systems and Computing, 176 AISC (VOL. 1), pp. 81-86, 2012. Springer Book Series. (SCOPUS INDEXED)- Cited 7
2. U. Prabu, G. Priyadharshini, M. Saranya, N.R. Parveen, M. Shanmugam, J. Amudhavel “Efficient personal identification using multimodal biometrics”, International Conference on Circuit, Power and Computing Technologies, 978-1-4799-7075-9/15 –(2015) (IEEE)