Solving Vehicle Routing Problem with Simultaneous Pickup and Delivery using Genetic Algorithm and Prins Splitting Procedure

(case study: TEPI's Helicopter Route Optimization in Off-shore Location)

By :

Carry Prameswari

13607039

Supervisors:

Hisar M. Pasaribu Ph.D

Pramudita Satria S.T

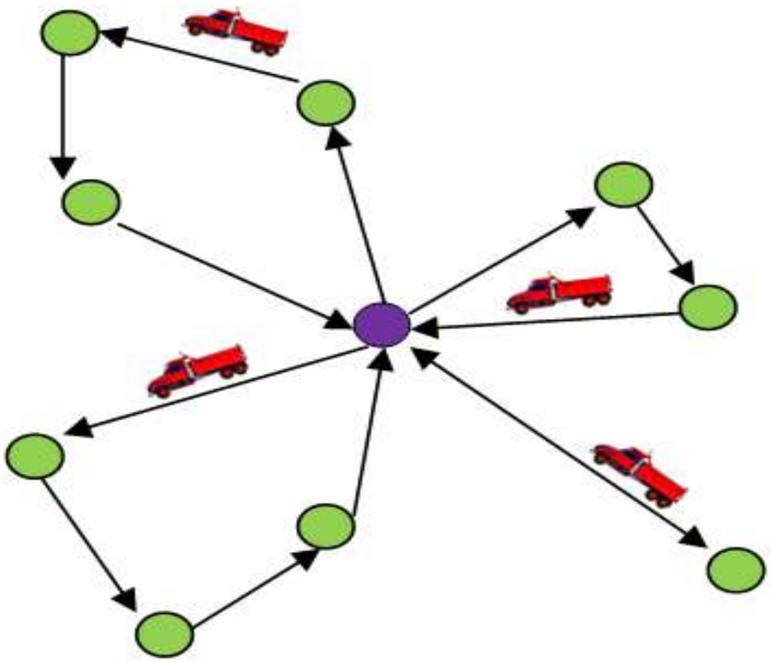
Outline

- Introduction
- Literature Review
- Mathematical Model and Software Implementation
- Result and Discussion
- Conclusion and Future Works

Introduction

What is VRP ?

- Combinatorial Optimization Problem
- Consist of :
  - Depot
  - Destination point
- Example : soft drink company



History of VRP ?

- First formally introduced by Dantzig and Ramser in 1959.
- They described a real-world application concerning the delivery of gasoline to service station
- and proposed the first mathematical programming formulation and algorithmic approach.

Introduction

How to Solve VRP ?

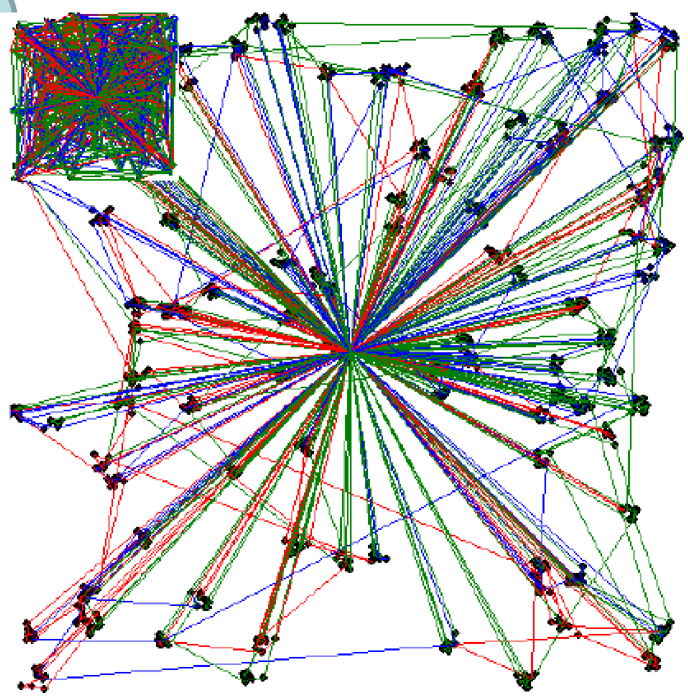
- Brute Force Method
- Linear Programming
- Heuristic Algorithm

This method works by generating all the combination possibilities, and pick the best one.

What if the problem becomes large and complex ?

Heuristic method can handle a very large and complex problem with more effective computation time.

A method that will be used in this research is Genetic Algorithm.



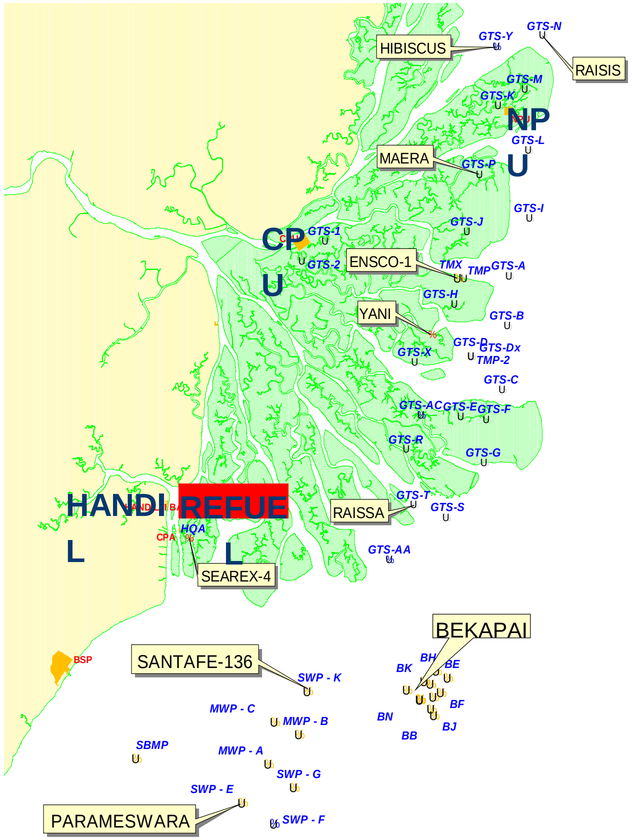
Introduction

Case study

TEPI’s helicopter routing problem in off-shore location.

Problem Statement:

- 15 landing points
- Three helicopters available with max. capacity is 12 pax.
- 3 Flights / day per helicopter
- Average 2600 passenger flown / month
- Different composition of passengers each day (site visit, meeting, crew changes, etc)



Introduction

Research Objectives

- Solving VRP
- Applying the genetic algorithm, prins splitting, etc.
- Output: VRP-SPD solver

Scope

- Vehicle Routing Problem with Simultaneous Pickup and Delivery.
- The data which is used for the case study, is the given from the TEPI company.
- Tools used: Matlab.

Literature Study

Vehicle Routing Problem (VRP)

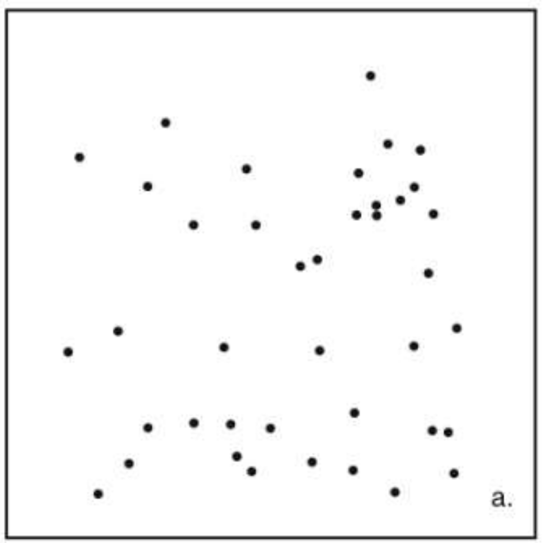
Genetic Algorithm (GA)

Prins Splitting Procedure

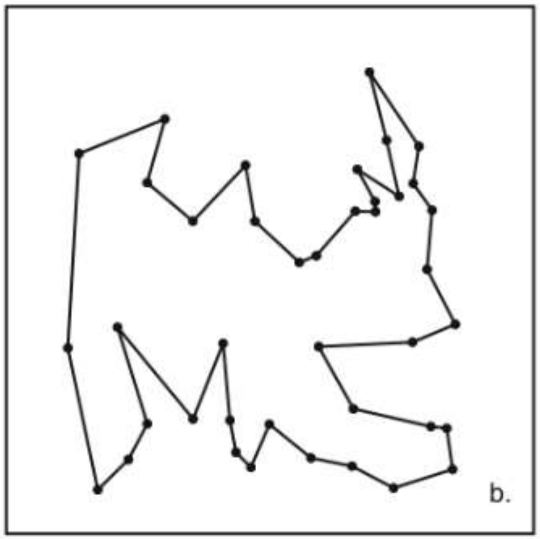
Literature Study

Vehicle Routing Problem (VRP)

Traveling Salesman Problem (TSP) as The Basis of The VRP



Which path has the shortest travel distance?

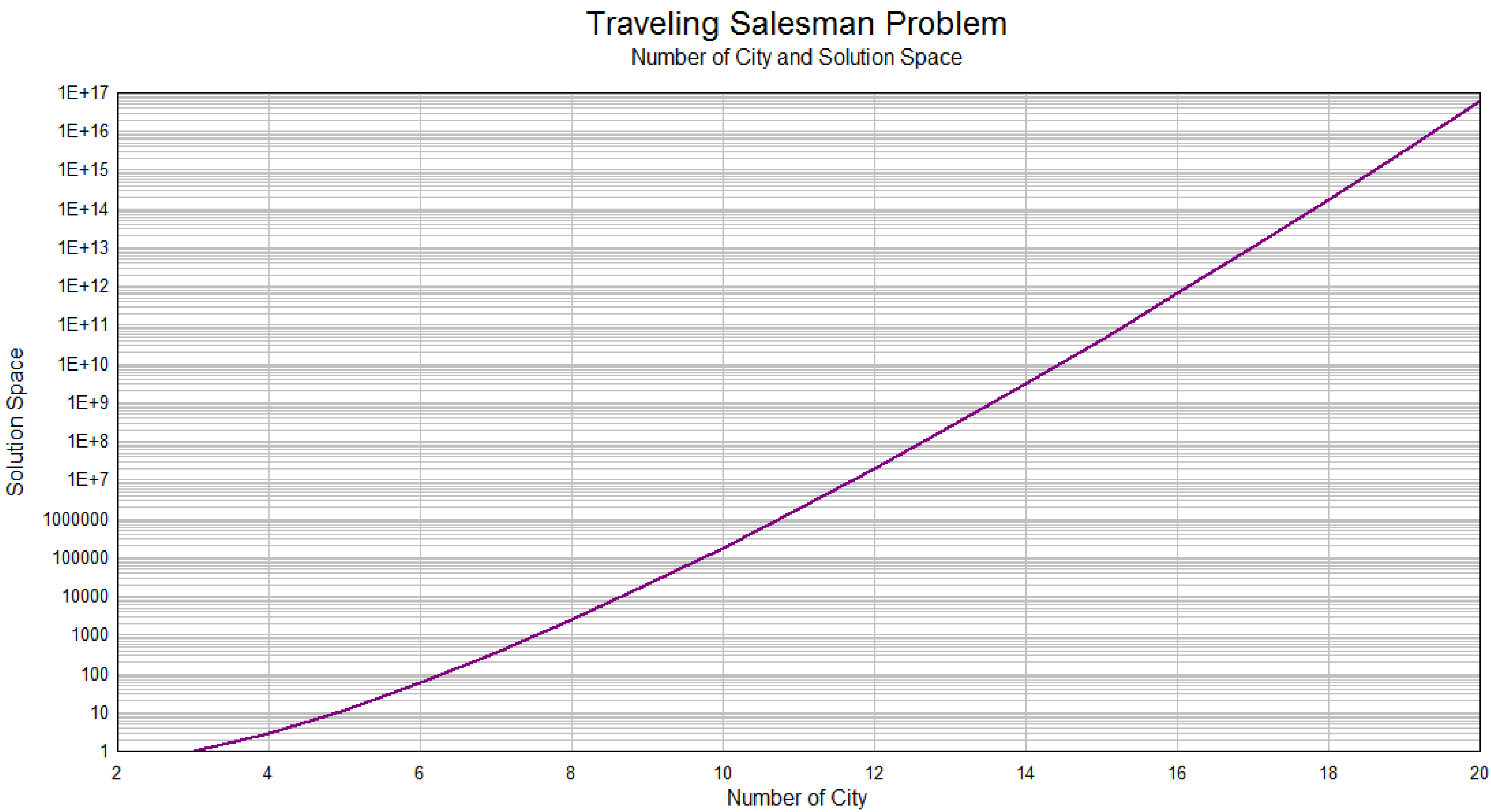


- There are numbers of (n) cities that must be visited.
- Start and end in the same point.

The solution space size can be calculated by this simple equation $\frac{1}{2}(n-1)!$ for n >2

Literature Study

Traveling Salesman Problem (TSP) as The Basis of The VRP



The growth of solution space in TSP

Literature Study

The Vehicle Routing Problem (VRP)

TSP

Similarity :
find the most minimum total traveling distance

Differences:
the constraints that must be fulfilled

VRP

Only one constraint:
the whole cities must be visited.

Multi Constrained Optimization Problem:
- The whole cities must be visited.
- Capacity of vehicle
- Max. Vehicle range
- Operation duration, etc

Literature Study

Variations Of The VRP :

- Vehicle Routing Problem with Pickup and Delivery (VRPPD)
- Vehicle Routing Problem with Simultaneous Pickup and Delivery (VRPSPD)
- Vehicle Routing Problem with Time Windows (VRPTW)
- Capacitated Vehicle Routing Problem (with or without Time Windows, CVRP or CVRPTW)

Within the increasing constraints

Solution space of VRP become very huge

Using Brute Force or Linear Programming are not impossible but very exhaustive & time consuming

Heuristic method is mostly chosen, in this case GA

Literature Study

Genetic Algorithm (GA)

- Developed by John Holland (1975)
- optimization techniques which is based on the principles of genetic and natural selection
- GA optimization technique is inspired by the theories in the biological sciences (population, reproduction, crossover, mutation, etc)
- The most important concept is heredity, *fitness* and *selection* for reproduction.
- Only the best individuals who can survive.

Literature Study

Genetic Algorithm (GA)

This method is selected to solve VRP because GA can handle:

- Multi-constrained problems, with lots of constraints.
- A problem with a huge solution space in relevant time.
- The problem which not has exact or known algorithm.
- A has time limit or a resource limit of the problem

Literature Study

Genetic Algorithm (GA)

Components of GA:

- A population with N chromosome (individual)
- Fitness Function
- Selection and Inheritance
- Crossover
- Mutation
- Crossover and mutation probability
- Elitism

Yes

No

Literature Study

Dynamic Programming

Fitness Function

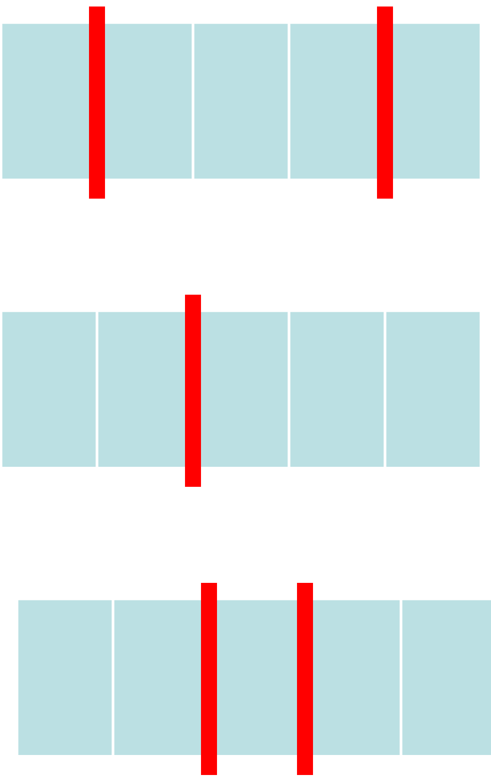
Bellman Ford Algorithm

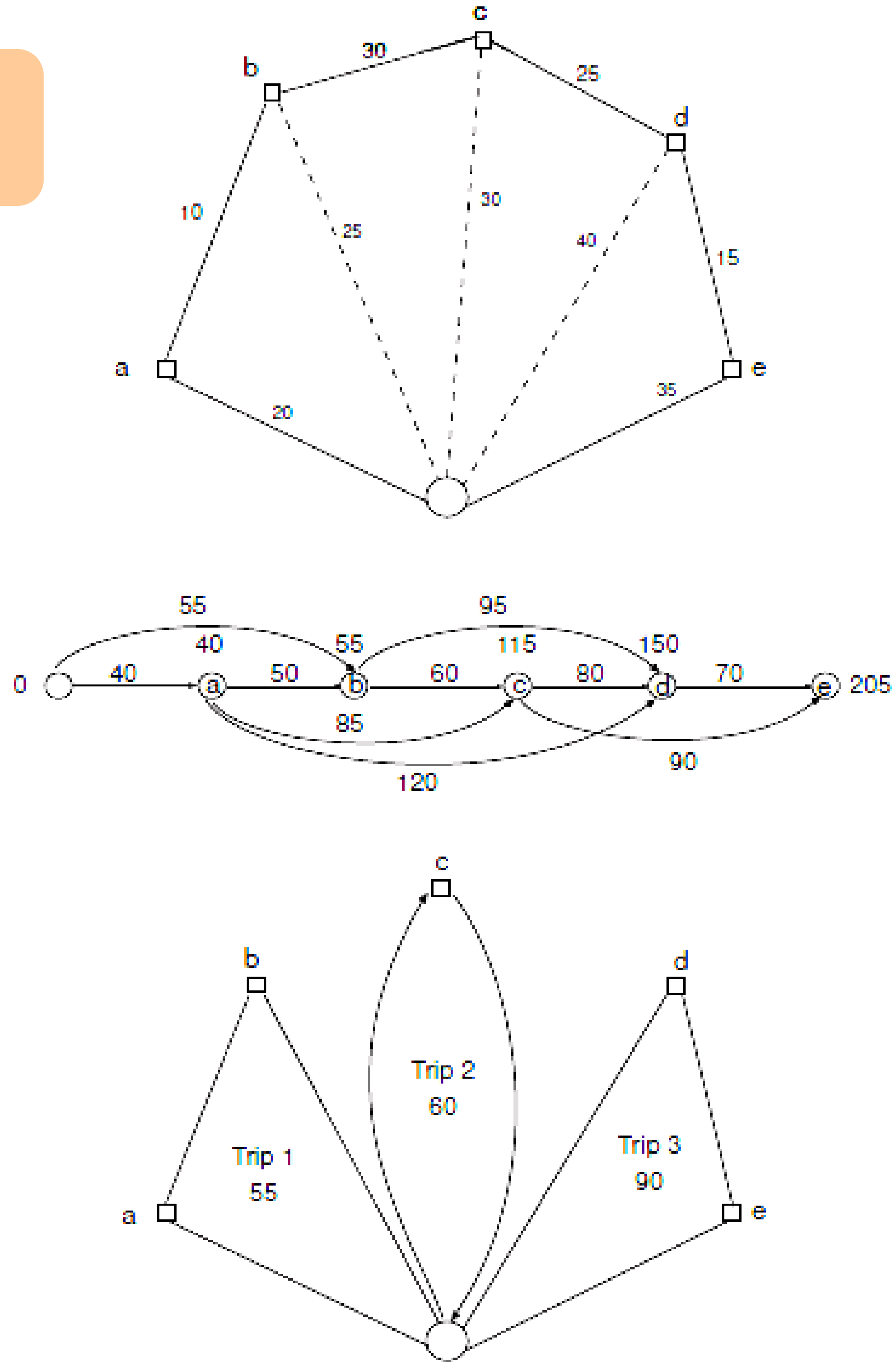
Prins Splitting Algorithm

method for solving complex problems by breaking them down into simpler sub problems.

algorithm to find the shortest path by calculating the shortest distance for every edges (point to point) in a weighted digraph.

Literature Study





Mathematical Model and Software Implementation

Mathematical Model

- On this research TEPI case is categorized as VRPSPD case, make the problem is simplified as follow:
  - Passenger demand which is served just demands from base to oil rig destination or from oil rig to base point.
  - Total passenger demand from one point will be carried all at once in one helicopter.
  - Multiple Trips is not part of this research, so if there is any demand that exceed the helicopter capacity, the solver cannot be executed.

Mathematical Model

Paolo Toth & Daniele Vigo (2002.) “Vehicle Routing Problem”.

The objective function seeks to minimize total distance traveled

$$Minimize \sum_{j=0}^{N}\sum_{j=0}^{N} c_{ij}x_{ij}$$

assure that each Client visited by exactly one vehicle.

$$\sum_{j=0}^{N} x_{ij} = 1 \qquad j = 1, \dots, N$$

$$\sum_{i=0}^{N} x_{ij} = 1 \qquad i = 1, \dots, N$$

flow equations for pick-up and delivery demands, respectively they guarantee that both demands are satisfied for each client

$$\sum_{j=0}^{N} y_{ji} - \sum_{j=0}^{N} y_{ij} = pj \qquad \forall j \neq 0$$

$$\sum_{i=0}^{N} z_{ji} - \sum_{i=0}^{N} z_{ij} = dj \qquad \forall j \neq 0$$

sets a limit in the number of vehicles used.

define the nature of the decision variables

$$x_{ij} \in \{0,1\}$$

$$y_{ij} \geq 0 \qquad i, j = 0, \dots, N$$

$$z_{ij} \geq 0 \qquad i, j = 0, \dots, N$$

establish that pickup and delivery demands will only be transported using arcs include in the solution

$$y_{ij} + z_{ij} \leq Q x_{ij} \qquad i, j = 0, \dots, N$$

Software Implementation

Population Initiation

Population consists of:

- Some individuals/chromosomes → Route combinations
  - Each chromosom consist of some gens → Destination points

Individual Evaluation

The purpose of this phase is calculate fitness function: finding a combination of these points which have a minimum total traveling distance.

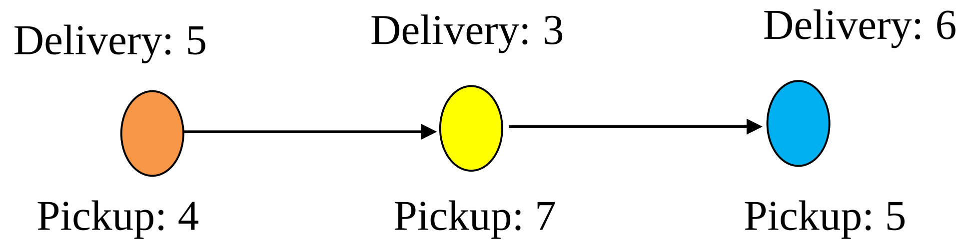
Should not violate the constraints

Software Implementation

Individual Evaluation

Constraints:

1. Vehicle capacity.
2. The maximum range of vehicle.
3. The available number of vehicle.
4. All passenger demand must be satisfied.
5. Pickup and delivery process should be conducted simultaneously.



The encoding scheme of individual evaluations is the result of modification from the previous work developed by Prins. The modification which applied in Prins Splitting Algorithm is adding a load calculation carried by each vehicle from point to point.

Software Implementation

Elitism

"save" the chromosomes/individuals which have the best fitness value

in order not to go through the cross-over and mutation process.

Parent Selection

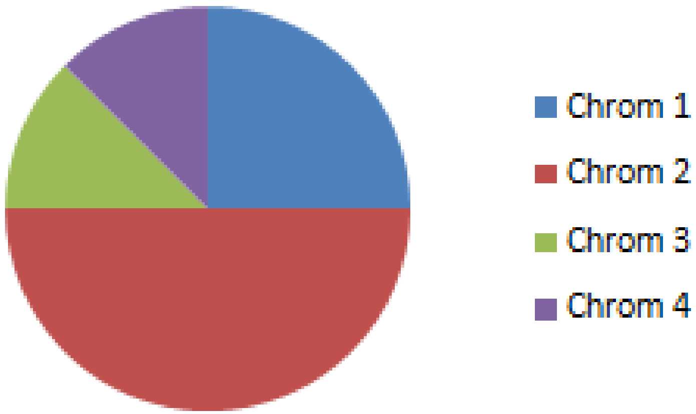
The parent selection process is done using the Roulette Wheel Selection method. To avoid the tendency of convergence in local optimum, scaling the fitness value is performed. This scaling is based on the following equation:[6]

$$f(i) = f_{max} - (f_{max} - f_{min})\left(\frac{R(i) - 1}{N - 1}\right)$$

Software Implementation

Parent Selection: Roulette Wheel Selection

Chromosome	Fitness Value
Chrom1	1
Chrom2	2
Chrom3	0.5
Chrom4	0.5
Total	4



Chrom2 with the biggest fitness value occupies a half circle.

Thus *Chrom2* have chance of 0.5 or 50% (2 divided by 4)

to elect as a parent. [6]

Software Implementation

Cross Over

The crossover method used for this case is novel order crossover method (NOX). NOX selected as a crossover method because it may be the best method to deal with permutation problems

parent 1

1	2	3	4	5	6	7	8	9	10

parent 2

10	9	8	7	6	5	4	3	2	1

children 1

4	5	6	7	1	2	3	8	9	10

children 2

7	6	5	4	10	9	8	3	2	1

Software Implementation

Mutation

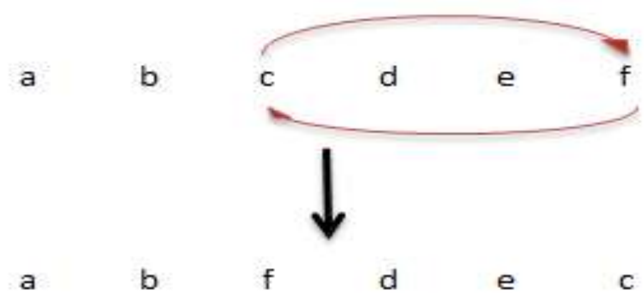
Mutation method used for this case is swapping mutation and inversion mutation. Both of this mutation method is used simultaneously in order to avoid the program stuck at the local optimum point.

inversion mutation

1	2	3	4	5	6	7	8	9	10

1	2	3	8	7	6	5	4	9	10

swapping mutation



Software Implementation

All the functions described above is then united in a main program. The pseudo code:

```
Function GeneticAlgorithm()
InitializeThePopulation();
IndividualsEvaluation();
       While(CriteriaHasNotReached)
       Repeat
       ParentSelection();
       CrossoverToCreateChildren();
       MutationToCreateChildren();
       ChildrenIndividualsEvaluation();
       ReplaceTheWorseOfThePopulationWithChildren();
       end
```

RESULT AND DISCUSSION

Program Validation

- This validation is done by inserting a test function case as an input which the answer is already known.
- If the result of the program has the same solution with the known test function solution, it can be said that the program is valid.
- A chromosome is accepted as valid if, and only if [17]:
  The point is that the resulting solution is valid as long as it does not violate any constraints.

RESULT AND DISCUSSION

Program Validation

<u>Test scenario</u>

Vehicle available: 3

Capacity: 25 passengers.

Maximum range: 50 km,

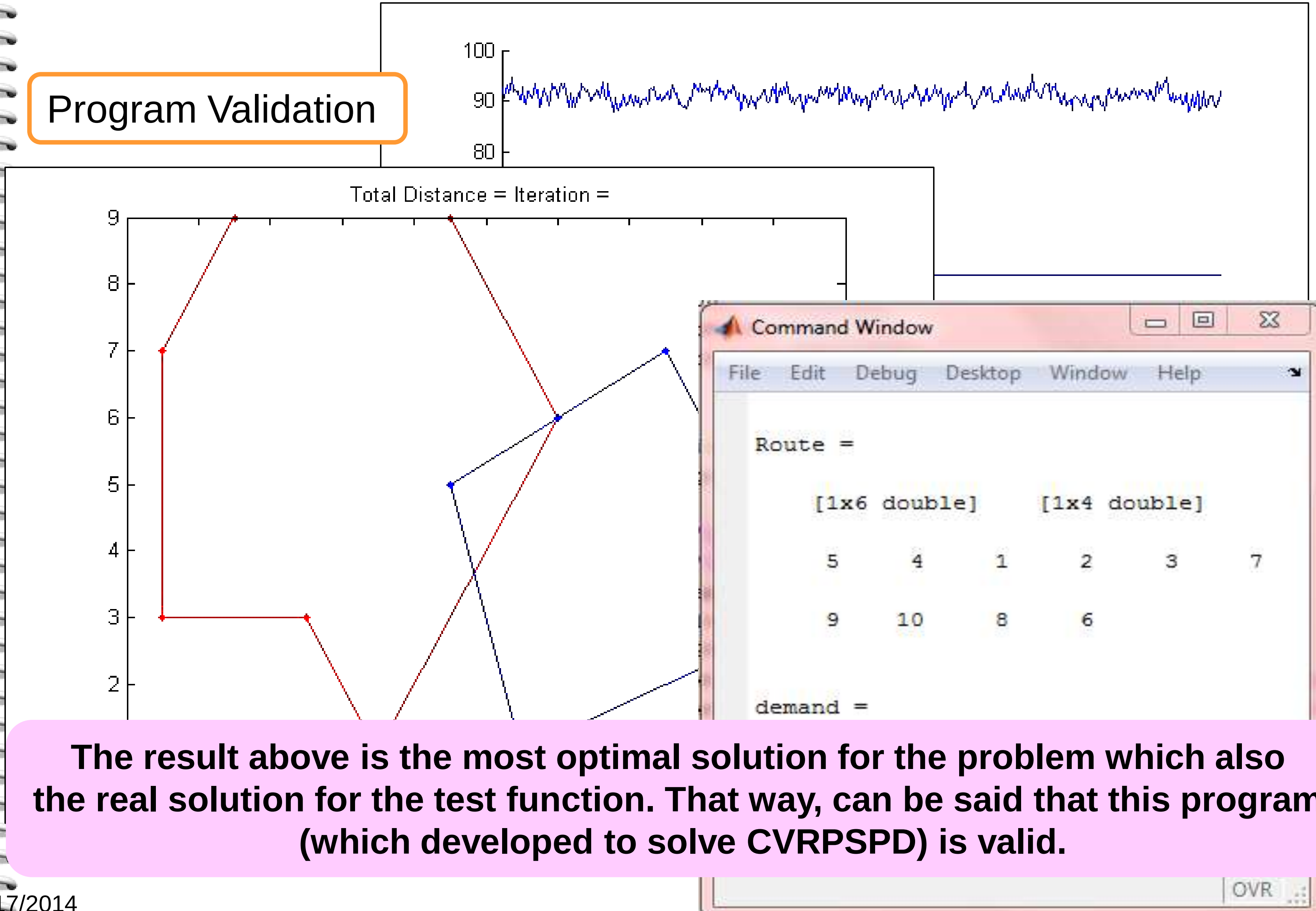
```
PopSize  = 80;
MaxG     = 500;
Pcrsovr  = 0.9;
Pmut     = 0.08;
PF       =5000;
```

point	coordinates		demand	
	X	Y	delivery	pick-up
1	1	3	4	5
2	1	7	3	7
3	3	9	6	3
4	5	3	2	2
5	7	1	5	1
6	9	5	6	5
7	9	9	4	6
8	11	1	1	4
9	15	7	3	4
10	19	3	5	2
base	12	6		

Table 4.1 Coordinates and Demand of Each Point

RESULT AND DISCUSSION

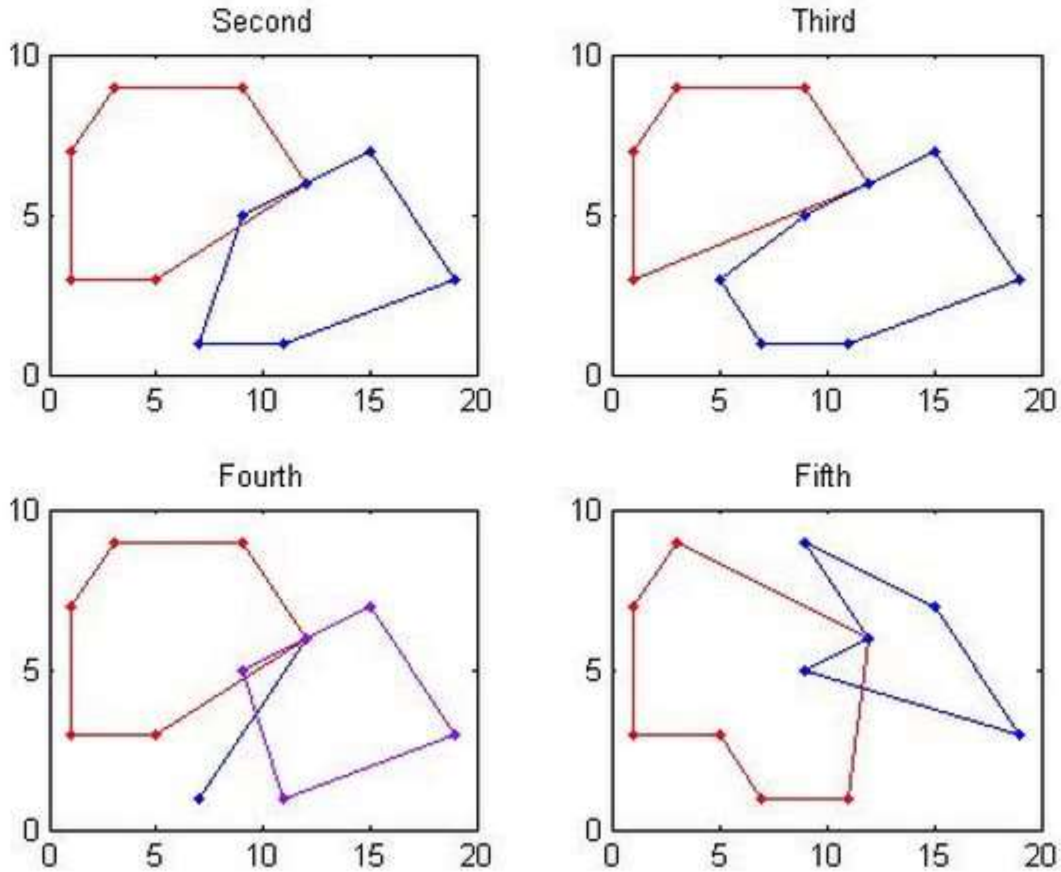
Program Validation



The result above is the most optimal solution for the problem which also the real solution for the test function. That way, can be said that this program (which developed to solve CVRPSPD) is valid.

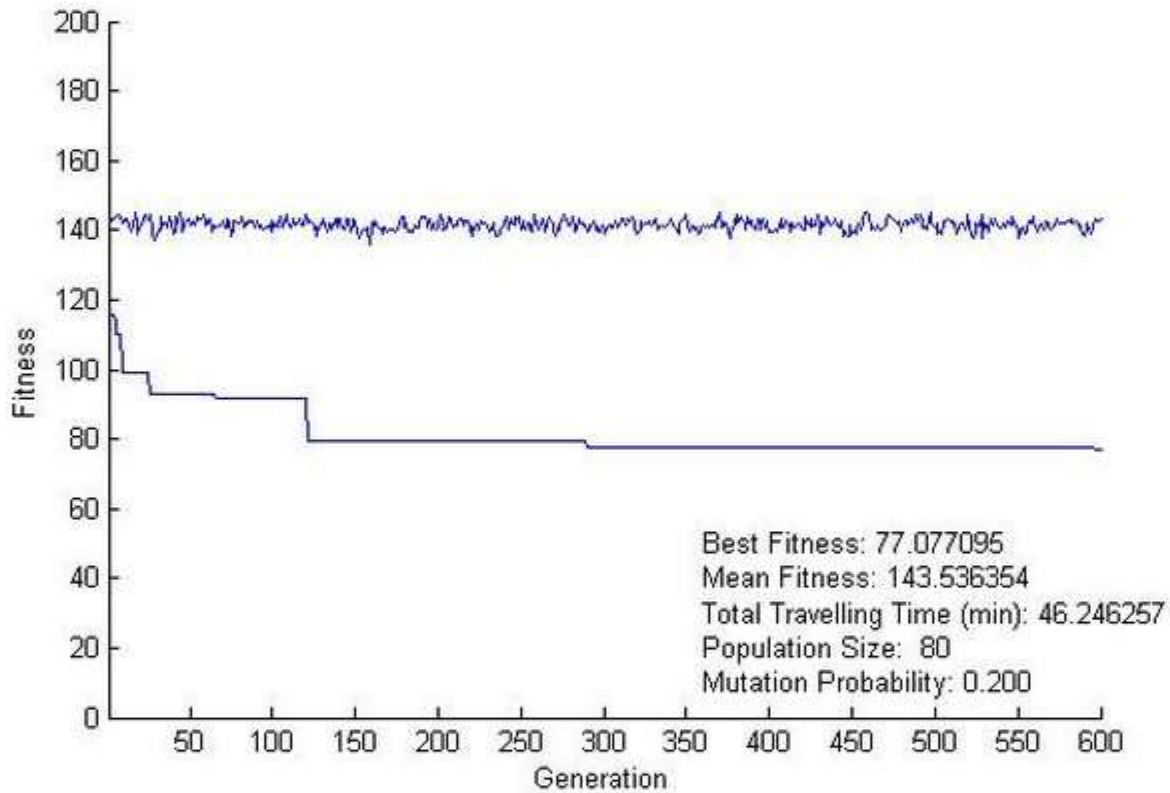
RESULT AND DISCUSSION

Test Function Itteration Process



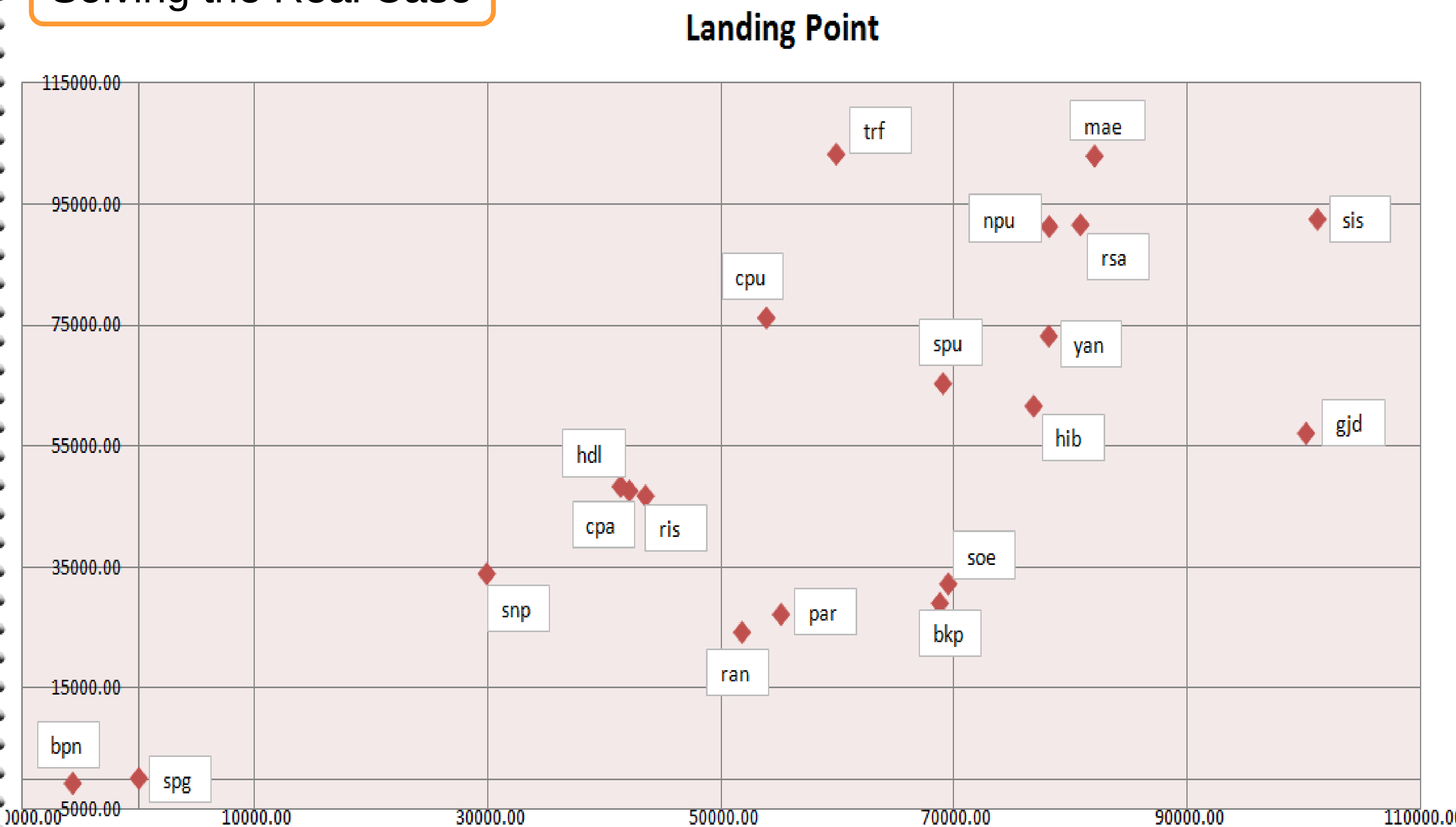
RESULT AND DISCUSSION

Example case



RESULT AND DISCUSSION

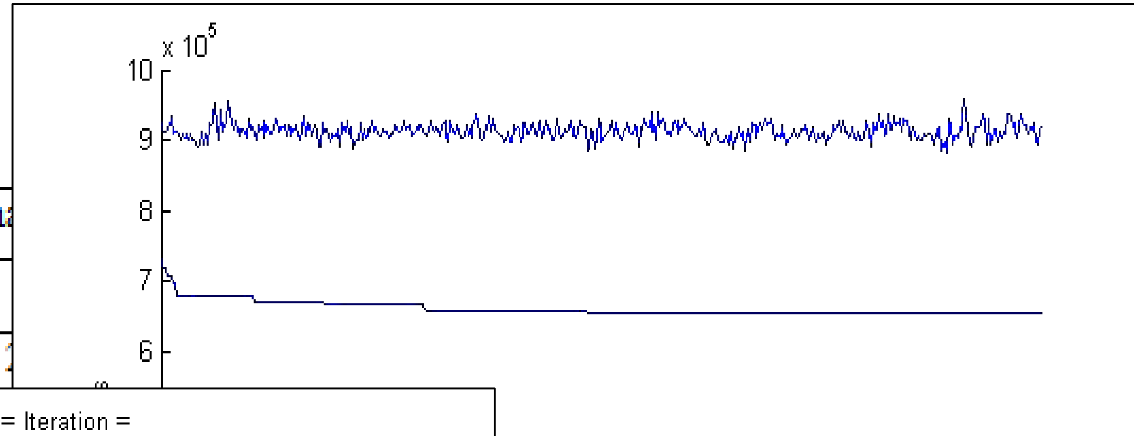
Solving the Real Case

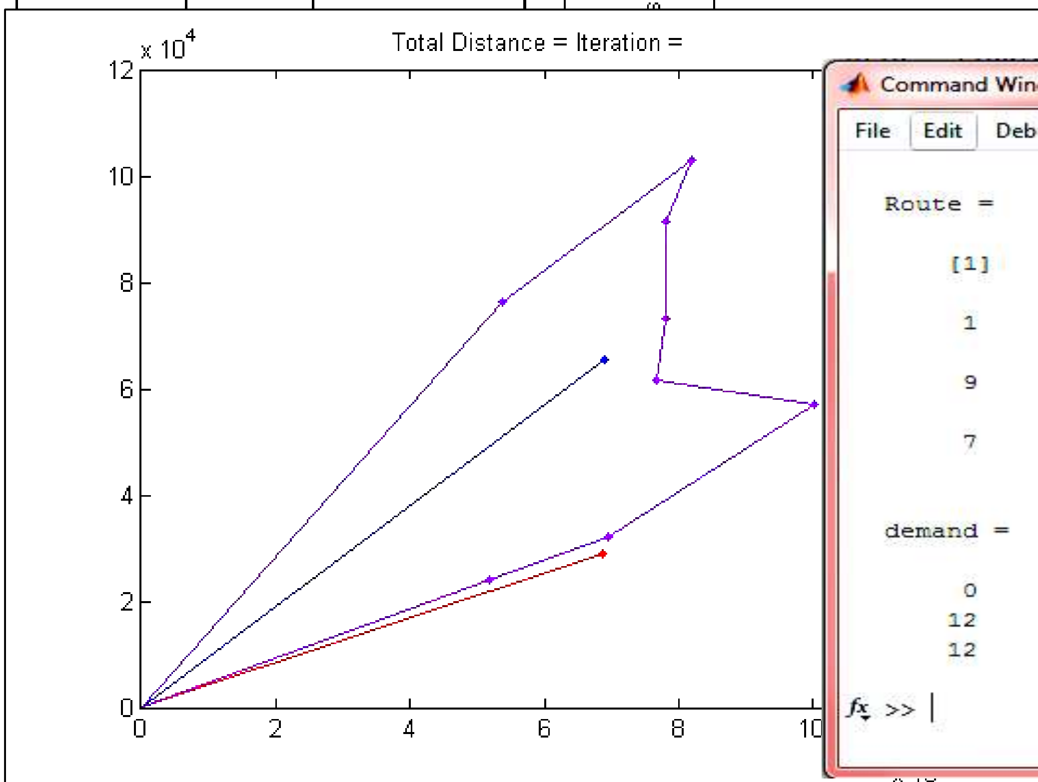


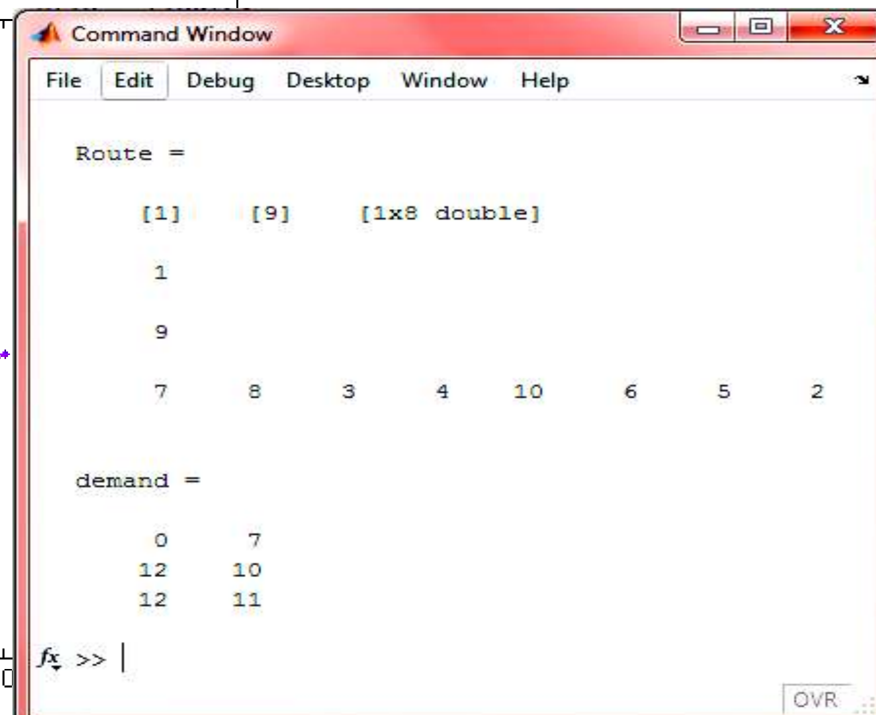
RESULT AND DISCUSSION

Solving the Real Case

Oil-rig name	index	coordin	
		X	
BKP	1	68780.00	1







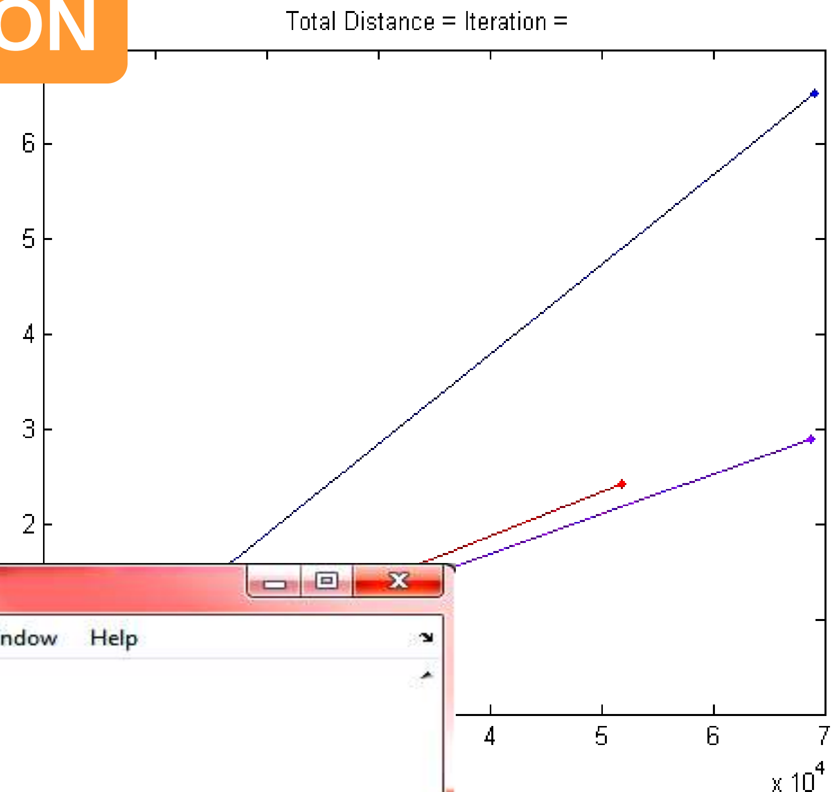
RESULT AND DISCUSSION

Solving the Real Case

7:30:00 AM

Oil-rig name	index	coordinates		
		X	Y	deliv
BKP	1	68780.00	28960.00	11
RAN	2	51790.00	24160.00	6
SPU	3	69050.00		
SPG	base	0.00		

Table 4.3 Rese



Command Window

File Edit Debug Desktop Window Help

```
Route =

    [1]     [2]     [3]

     1

     2

     3


demand =

    11     5
     6     7
    12    12

>>
```

OVR

RESULT AND DISCUSSION

Finding the Optimal Parameters

The main variables are:

-population size,

-mutation probability and

-crossover probability.

In this research, only value of population size and mutation probability which will be analyzed in order to find better solution.

This observation is done by varying 5 population size: [30, 40, 50, 80, 100] and 6 mutation probability: [0.025, 0.05, 0.08, 0.1, 0.15, 0.2].

These variations will produce 30 parameter combinations.

For each combination, it is done by10 times running with the number of evaluation

individual most widely 50000.

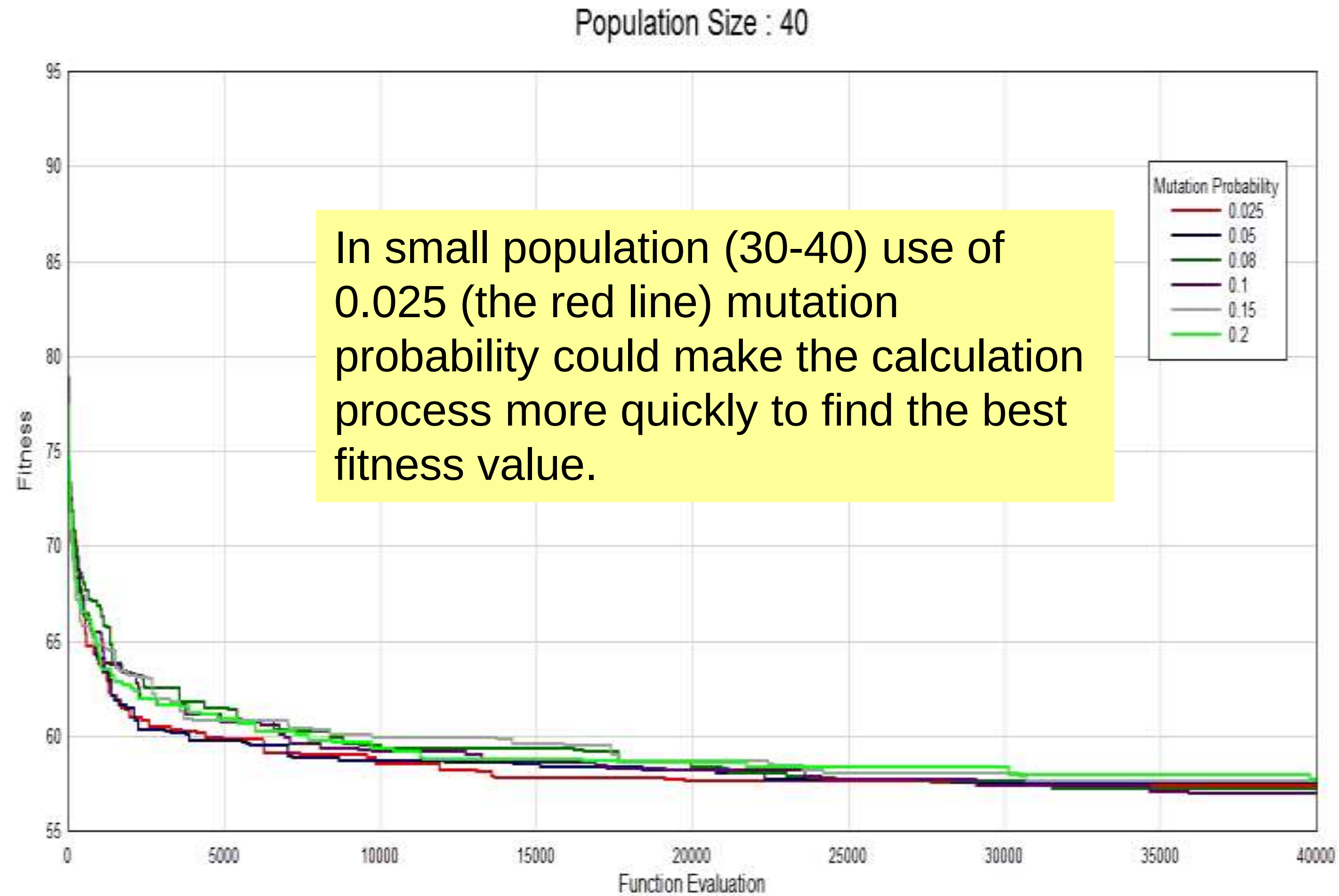
RESULT AND DISCUSSION

Finding the Optimal Parameters

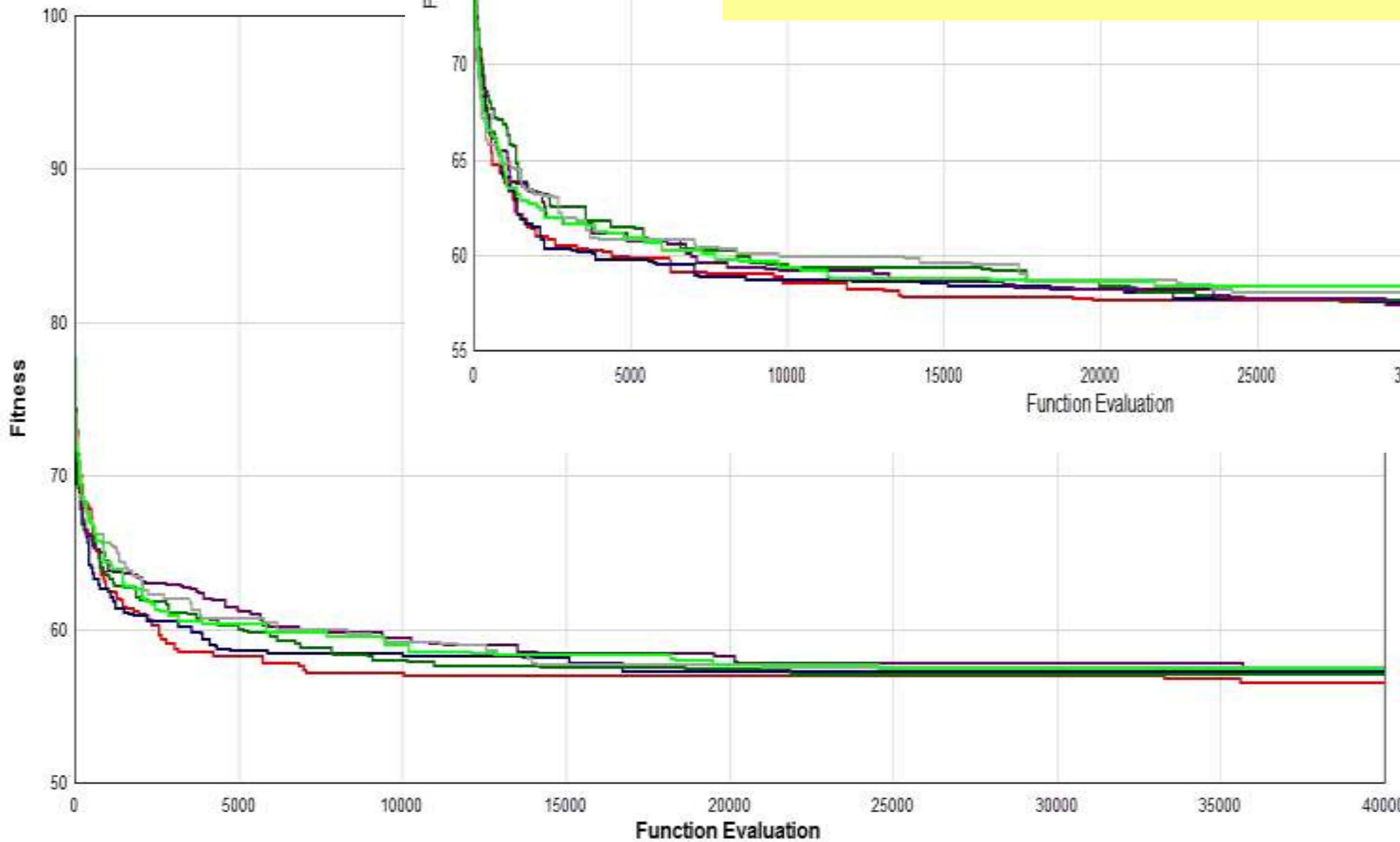
The greater mutation probability value makes the difference between mean fitness and the optimal fitness become greater too.

Population Size	Mutation Probability	Mean Fitness	Max Individual Number	Population Size	Mutation Probability	Mean Fitness	Max Individual Number
30	0.025	56.013575	49980	50	0.1	57.68957	50000
	0.05	56.690016	49980		0.15	57.607142	50000
	0.08	57.200786	49980		0.2	57.504887	50000
	0.1	56.990808	49980	80	0.025	56.516971	50000
	0.15	56.518388	49980		0.05	57.517942	50000
	0.2	56.669874	49980		0.08	56.567006	50000
[illegible]	0.025	56.34676	50000		0.1	57.686865	50000
	0.05	56.34676	50000		0.15	56.9568	50000
	0.08	57.174686	50000		0.2	56.82556	50000
	0.1	57.607941	50000	100	0.025	57.859781	50000
	0.15	56.940772	50000		0.05	57.1132	50000
	0.2	57.316184	50000		0.08	57.369724	50000
50	0.025	56.003504	50000		0.1	57.063165	50000
	0.05	56.34676	50000		0.15	58.20852	50000
	0.08	57.392046	50000		0.2	57.530727	50000

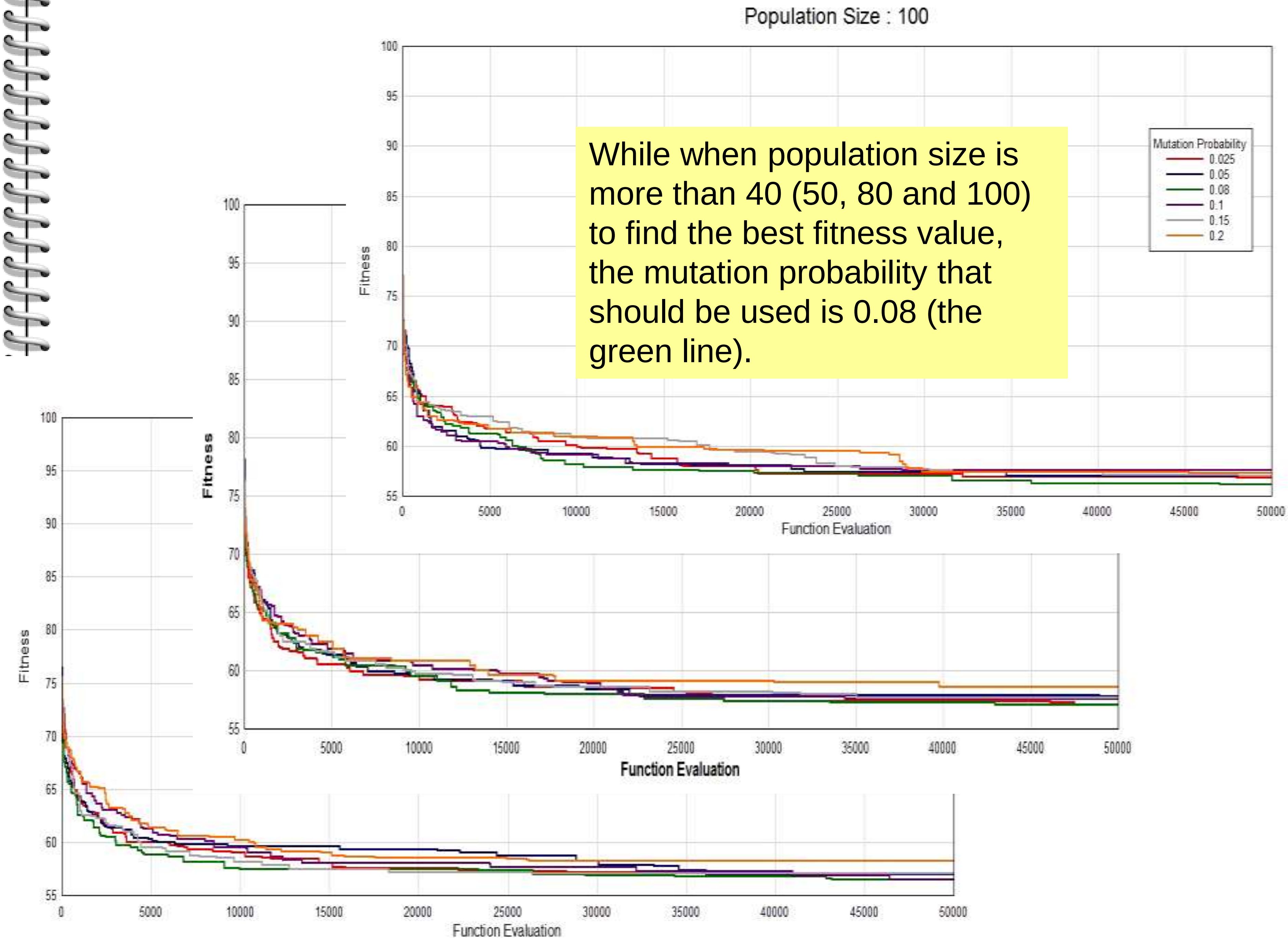
the greater population size, the greater difference in mean fitness to the optimal fitness value.

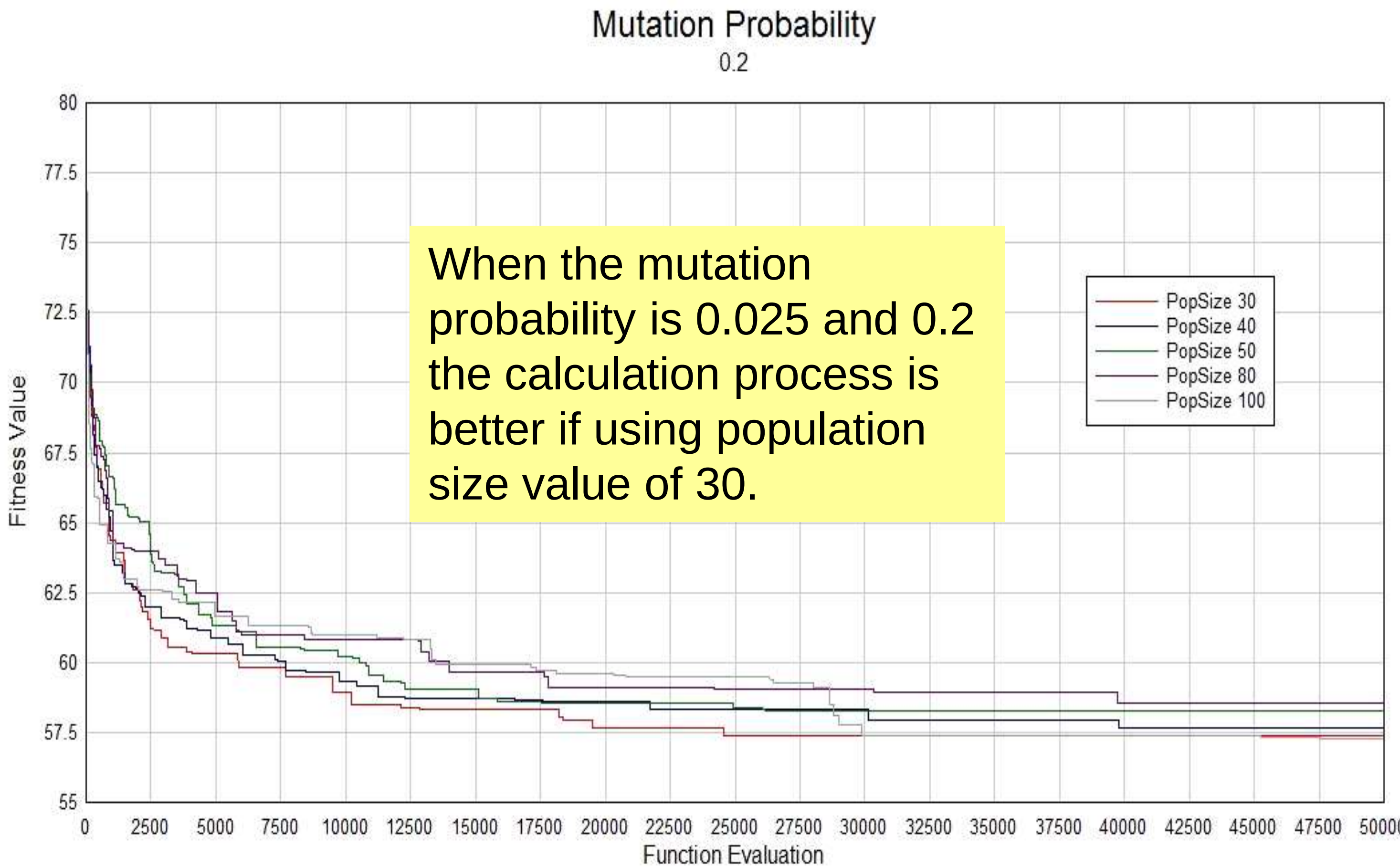


In small population (30-40) use of 0.025 (the red line) mutation probability could make the calculation process more quickly to find the best fitness value.

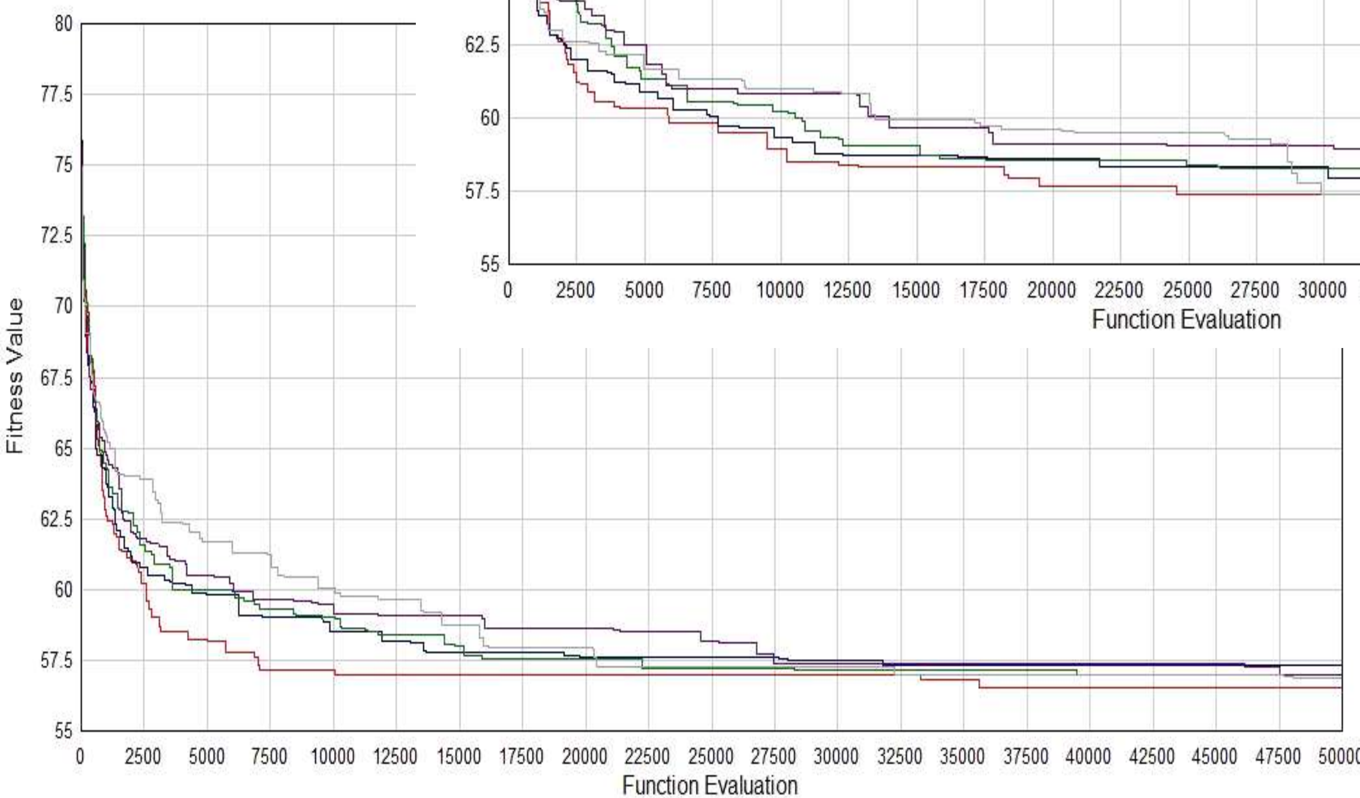


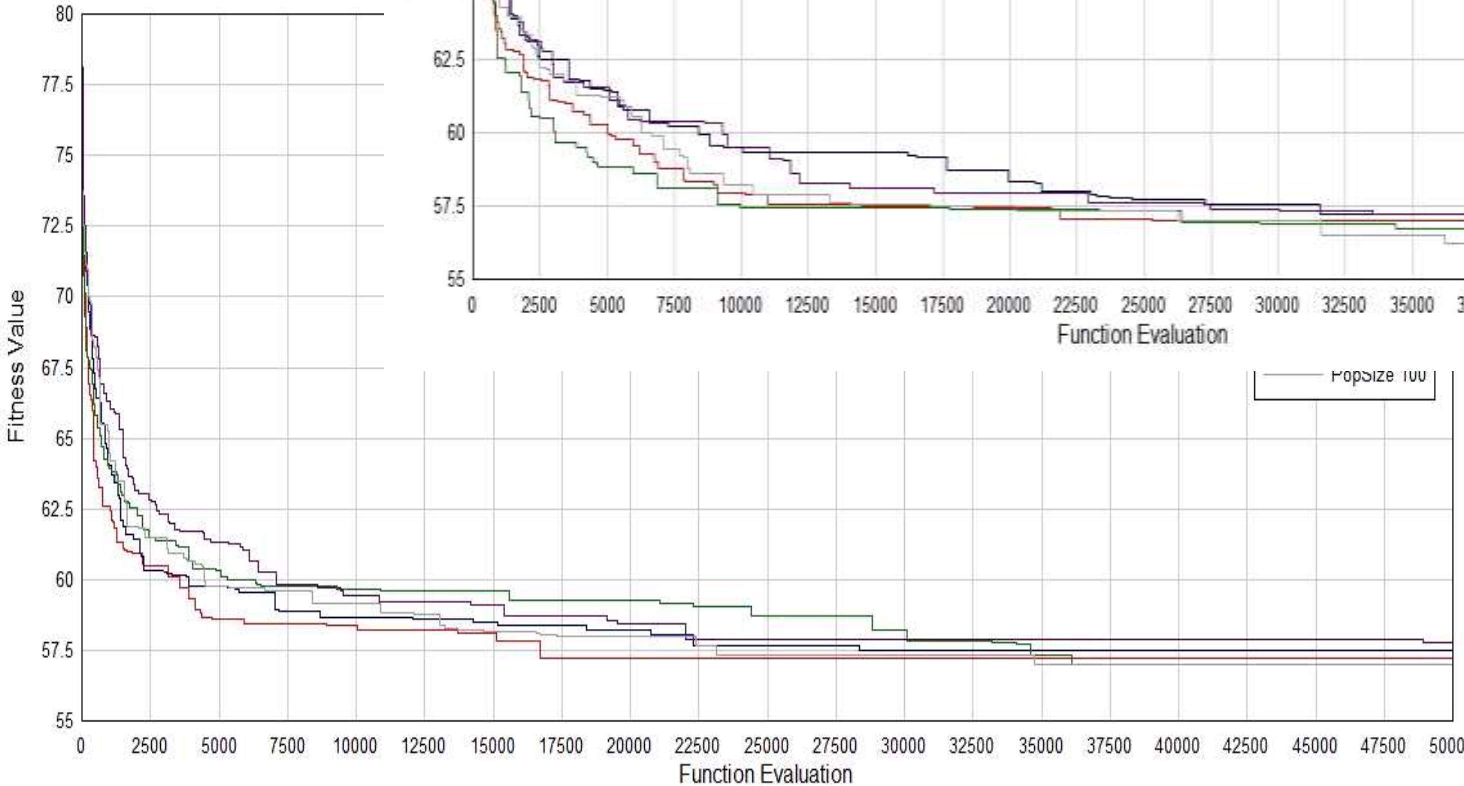
While when population size is more than 40 (50, 80 and 100) to find the best fitness value, the mutation probability that should be used is 0.08 (the green line).

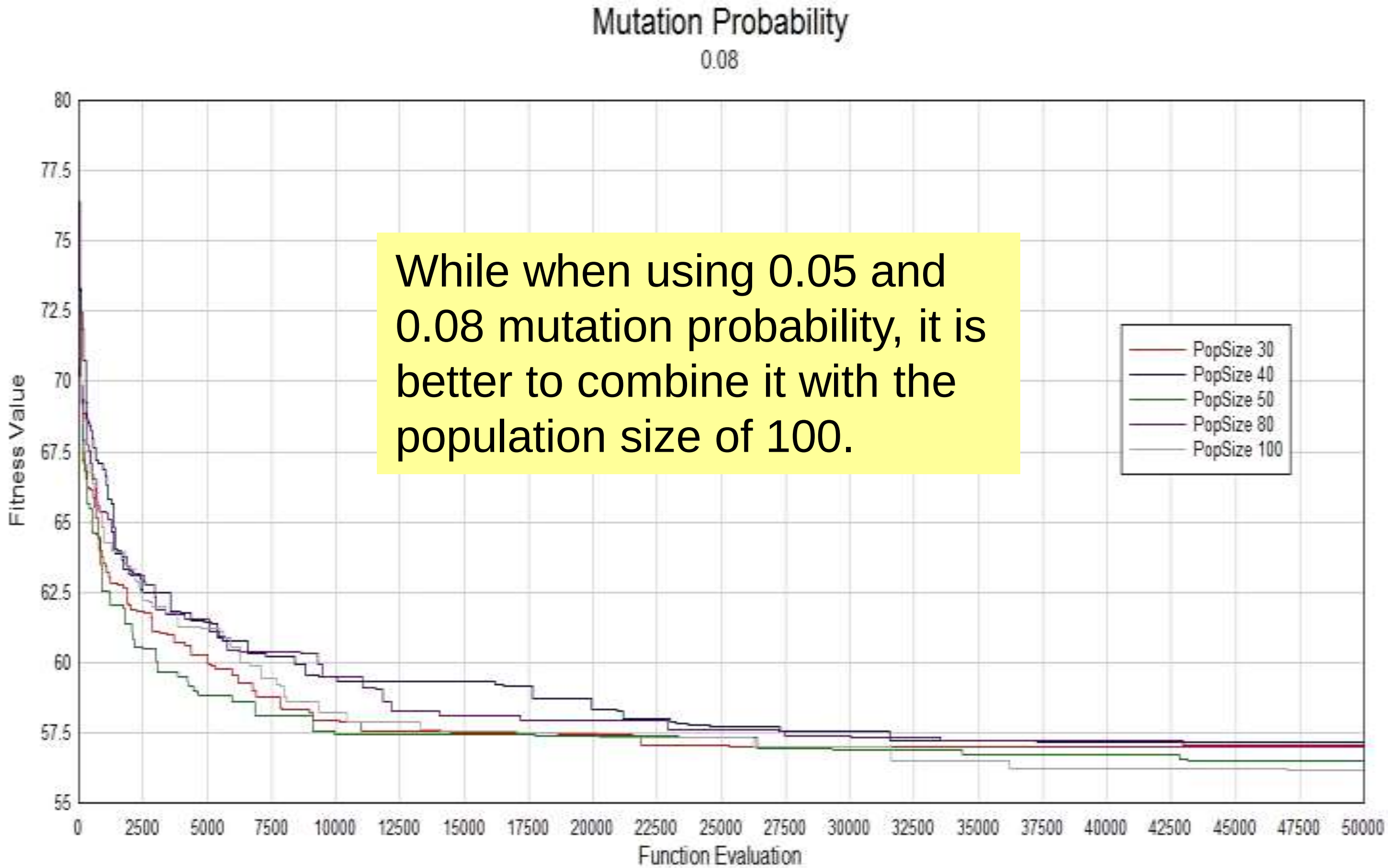




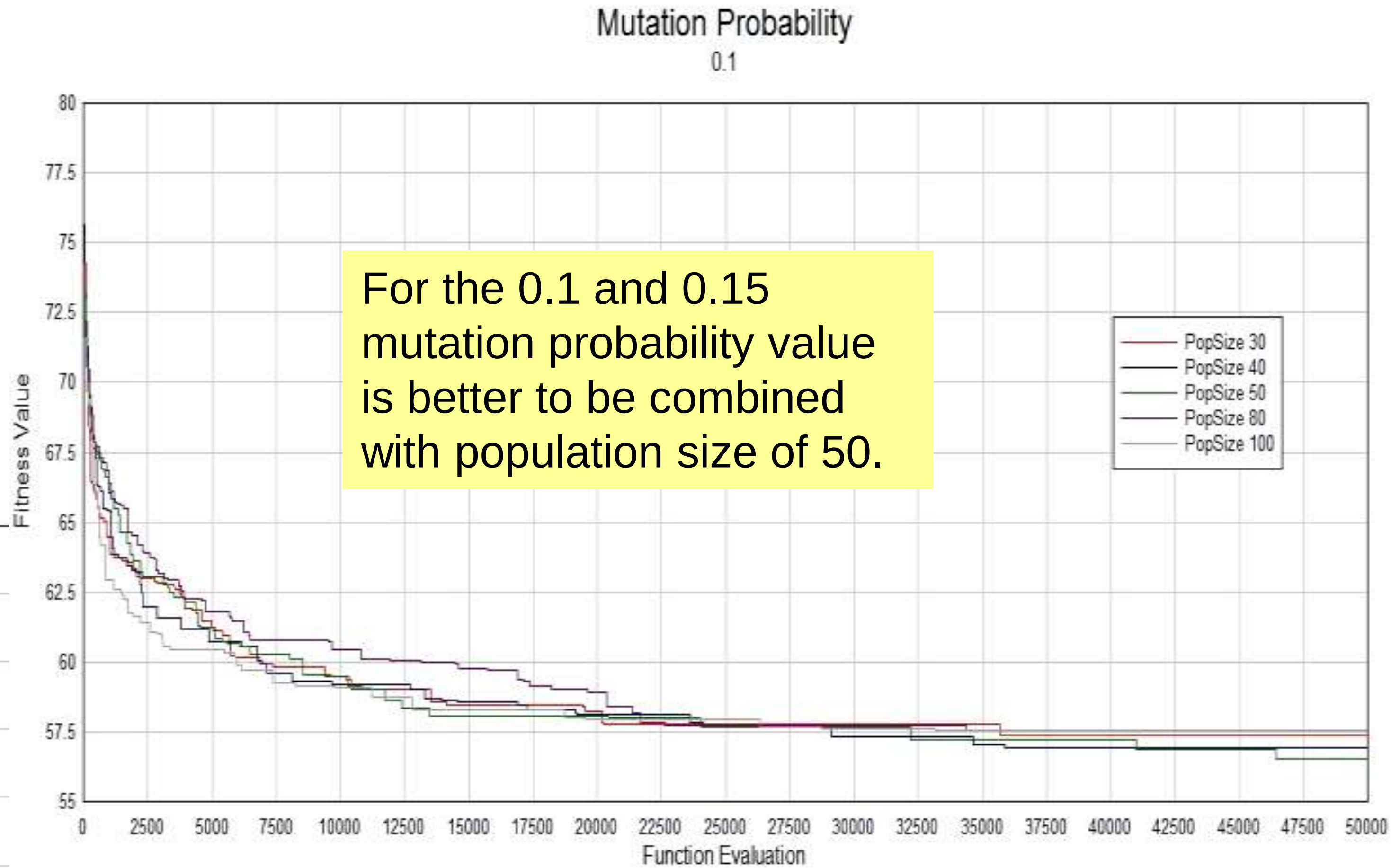
When the mutation probability is 0.025 and 0.2 the calculation process is better if using population size value of 30.



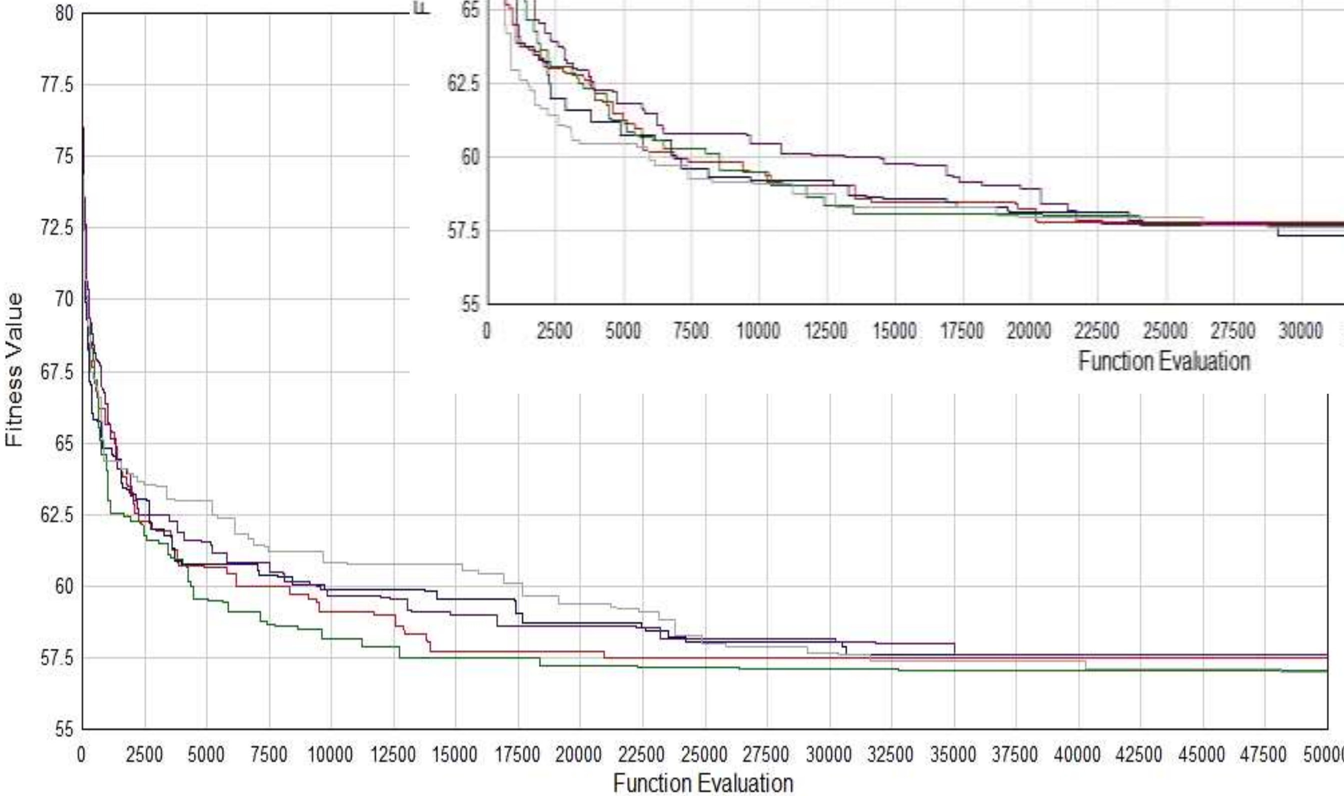




While when using 0.05 and 0.08 mutation probability, it is better to combine it with the population size of 100.



For the 0.1 and 0.15 mutation probability value is better to be combined with population size of 50.



RESULT AND DISCUSSION

Finding the Optimal Parameters

Population size

The smaller population would only cover a small search space and may results in poor performance.

A larger population would cover more space and prevent premature convergence to local solutions. a large population needs more evaluation per generations and may slow down the convergence rate.

Crossover

On the other hand, for small population size the choice of crossover operators plays a more dominant role.

Mutation

Probability of mutation or mutation rate is the probability with which each bit position of each string in the new population undergoes a random change after a selection process.

A low mutation rate helps to prevent any bit positions from getting stuck to single values, where as a high mutation rate results in essentially random search.

CONCLUSIONS AND FUTURE WORKS

Conclusions

Genetic Algorithm is a suitable method to solve combinatorial optimization such Vehicle Routing Problem which has a very large solution space.

The more specific VRP problem which is discussed in this thesis—the VRPSPD—can be solved using Genetic Algorithm coupled with modified Prins Splitting Algorithm.

The use of Prins Splitting Algorithm as an individual evaluation function in pickup and delivery problem is very effective, it is also used to minimize the solution space which will be evaluated

To obtain an optimal solution in GA, further analysis should be done to determine the parameters value corresponding to each case which will be resolved.

CONCLUSIONS AND FUTURE WORKS

Conclusions

For VRPSPD test function case discussed in this thesis, parameters value that can be used to obtain the optimal results are:

- Population size value ranges from 80 to 100 uses mutation probability 0.08
- Population size value ranges from 30 to 40 uses mutation probability 0.025

VRPSPD test function case discussed in this thesis can be completed in 5.027645 seconds by using personal computer with 2.10 GHz processor AMD Turion X2 Dual-Core Mobile RM-72 and 4 GB of RAM.

CONCLUSIONS AND FUTURE WORKS

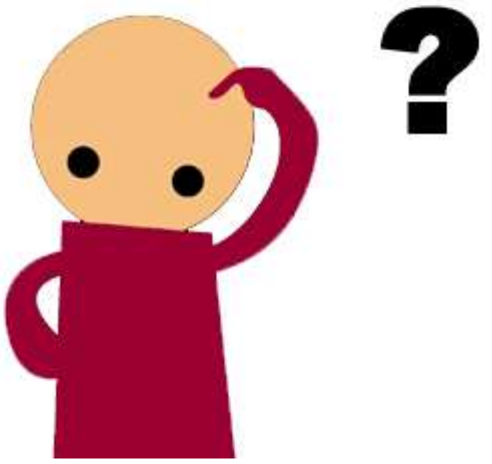
Future Works

For VRP case GA, GA performance can be improved by adding local search method.

For future work, additional constraints can be added, such as time window, multi vehicle, multi trips, inter-point demand and etc.

In addition, comparison between heuristic method should be done so the VRP problem can be solved more effective and produce more optimal solution.

Any question ??




Good-bye!
See you later, alligator!
After while, crocodile!
In an hour, sunflower!
Maybe two, kangaroo!
Gotta go, buffalo!
Adios, hippos!
Chow, chow brown cow!
See you soon, baboon!
Adieu, cockatoo!
Better swish, jellyfish!
Chop, chop, lollipop!
Gotta run, skeleton!
Bye-bye, butterfly!
Better shake, rattle snake!
Our school day now ends.
So, good-bye good friends!

Thank You