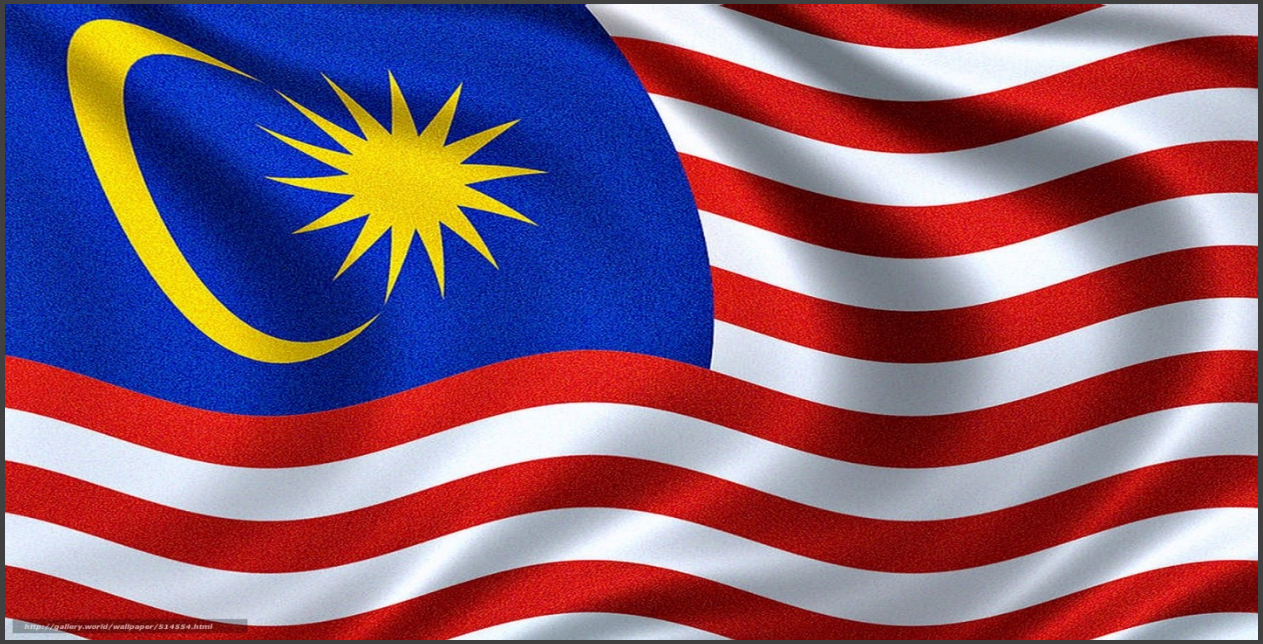
Briefly Introduce Myself

- Name: Chieng Hock Hung
- Chinese Name: 钱学宏
- Nick Name: Joshua or DAMAGE
- Ancestral Home: Fujian
- Nationality: Malaysia
- Home state: Sarawak (Borneo Island)
- Hometown: Limbang (林梦)




Kuala Lumpur (吉隆坡), the capital of Malaysia.
Limbang (林梦), Where I grow up!
Bintulu (民都鲁), Where I was born!
Batu Pahat (峇株巴辖), Where I got my Bachelor and Master Degree.
Regional capital
Town, village
Airport
International boundary
Regional boundary
Main road
Secondary road
Railroad
South China Sea
VIET NAM
THAILAND
Thailand
PHILIPPINES
Palawan Is
BRUNEI DARUSSALAM
Bandar Seri Begawan
Sabah
Sarawak
MALAYSIA
Borneo Is
Sumatra Is
INDONESIA
Singapore
Kuala Lumpur
Ho Chi Minh City
Kota Kinabalu
Mount Kinabalu 4,095 m
Kuching
Bintulu
Mukah
Sibu
Miri
Kuala Terengganu
Kuantan
Kota Bahar
Ipoh
Seremban
Malacca
Johor Baharu
Shah Alam
Port Klang
Port Dickson
Keluang
Mersing
Sandakan
Natuna Besar
Kepulauan Anambas (Indonesia)
0 50 100 200 km
0 50 100 150 mi
© Nations Online Project

Previous Academic Degree

- **Bachelor Degree (2009-2013)**

  Institute: Universiti Tun Hussein Onn Malaysia

  Field of study: Information Technology (Multimedia)

  Final year project: The development of a modern musical instruments application for Android platform

- **Master Degree (2013-2015)**

  Institute: Universiti Tun Hussein Onn Malaysia

  Field of study: Information Technology (Application)

  Topic: (I will show to you now….)

VIVA PRESENTATION

TITLE:

GENETIC SIMPLIFIED SWARM ALGORITHM (GSSA) FOR OPTIMIZING *n* -CITIES OPEN LOOP TRAVELLING SALESMAN PROBLEM (*n*OTSP)

Presenter:
Chieng Hock Hung
M.Sc. in Information Technology

Supervisor:
Dr. Noorhaniza Binti Wahid
Head of Department
Department Multimedia
UNIVERSITI TUN HUSSEIN ONN MALAYSIA

Presentation outline

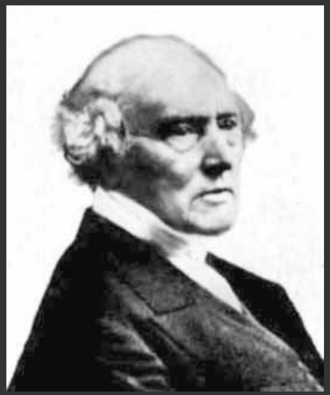
1. Introduction
2. Objectives of Study
3. Methodology
4. Analysis & Result
5. Conclusion & Future Work

Introduction

Travelling Salesman Problem (TSP)

1800s

W. R. Hamilton



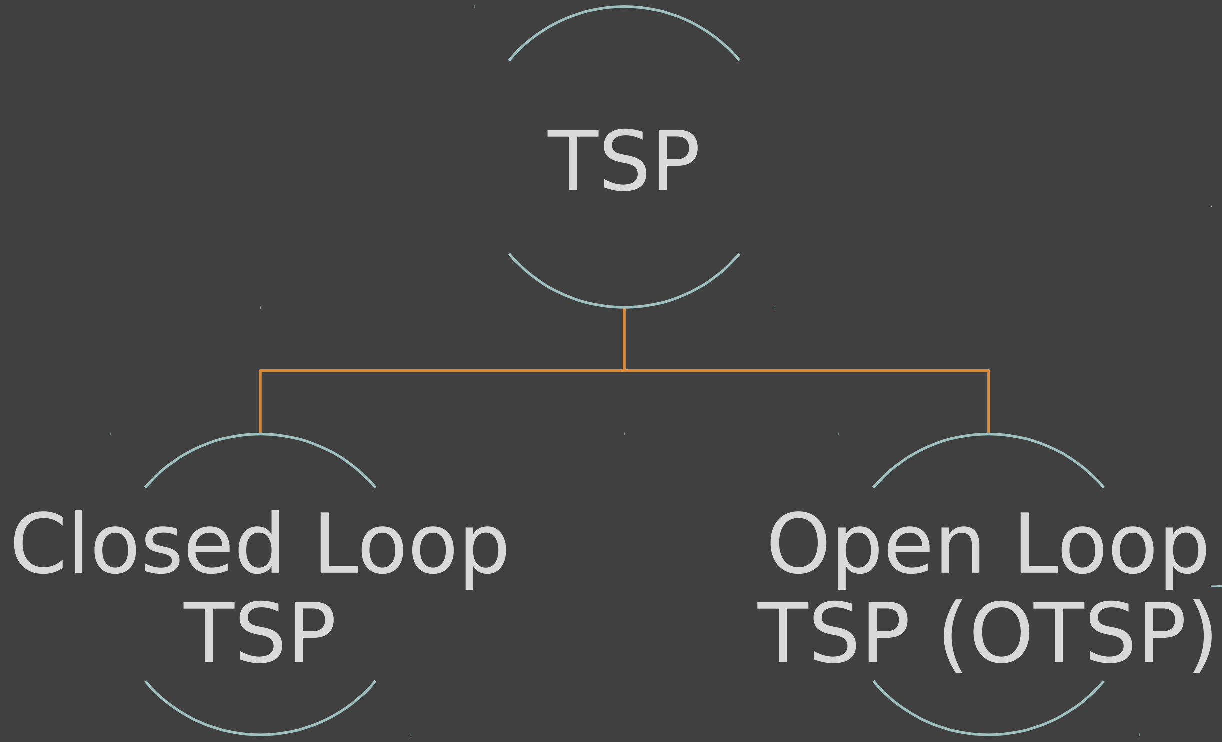
P. Krikman

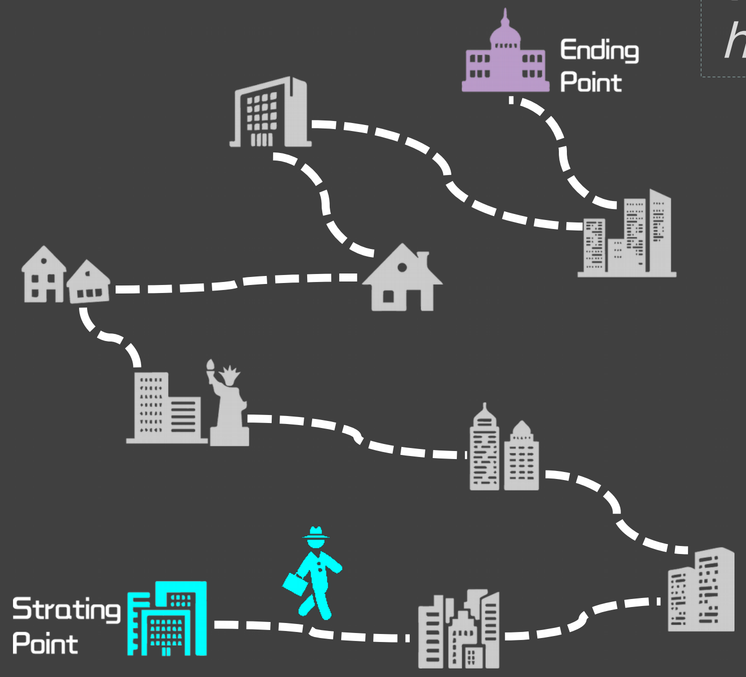
Definition:

A salesman who must travel and visit each one of the given cities exactly once and return to the city he started with the total of the route which is shortest.

Total number of city, m=10.

Introduction





*** Salesman does not return to the city where he departed.*

Total number of city, m=10.

Introduction

- In today real-life transportation scenarios, the problems are not exactly similar as what has been described in TSP & OTSP.

- In daily practical cases…
  - There are vehicles may only travel from a starting point and end the journey in another destination.
  - Unnecessary to visit all the cities (or nodes, locations).
  - Yet they travel only to a certain number of “cities” with minimum total travelling distance.
  - For example:
    - ✓ Logistics of merchandise delivery services – Fuel saving, time saving & environment friendly.
    - ✓ Emergency & evacuation transportation route planning (Short

*n*OTSP possible solutions (combinations)

$$\frac{m!}{2(m-n)!} = \text{Possible solutions}$$

*****m=50***

n	*Possible solutions*
10	
20	
30	
40	

n	*Possible solutions*
10	1.86×10^{16}
20	5.73×10^{31}
30	6.25×10^{45}
40	4.19×10^{57}

n	*Possible solutions*
10	1.86×10^{16}
20	5.73×10^{31}
30	6.25×10^{45}
40	4.19×10^{57}

n	*Possible solutions*
10	1.86×10^{16}
20	5.73×10^{31}
30	6.25×10^{45}
40	4.19×10^{57}

n	*Possible solutions*
10	1.86×10^{16}
20	5.73×10^{31}
30	6.25×10^{45}
40	4.19×10^{57}

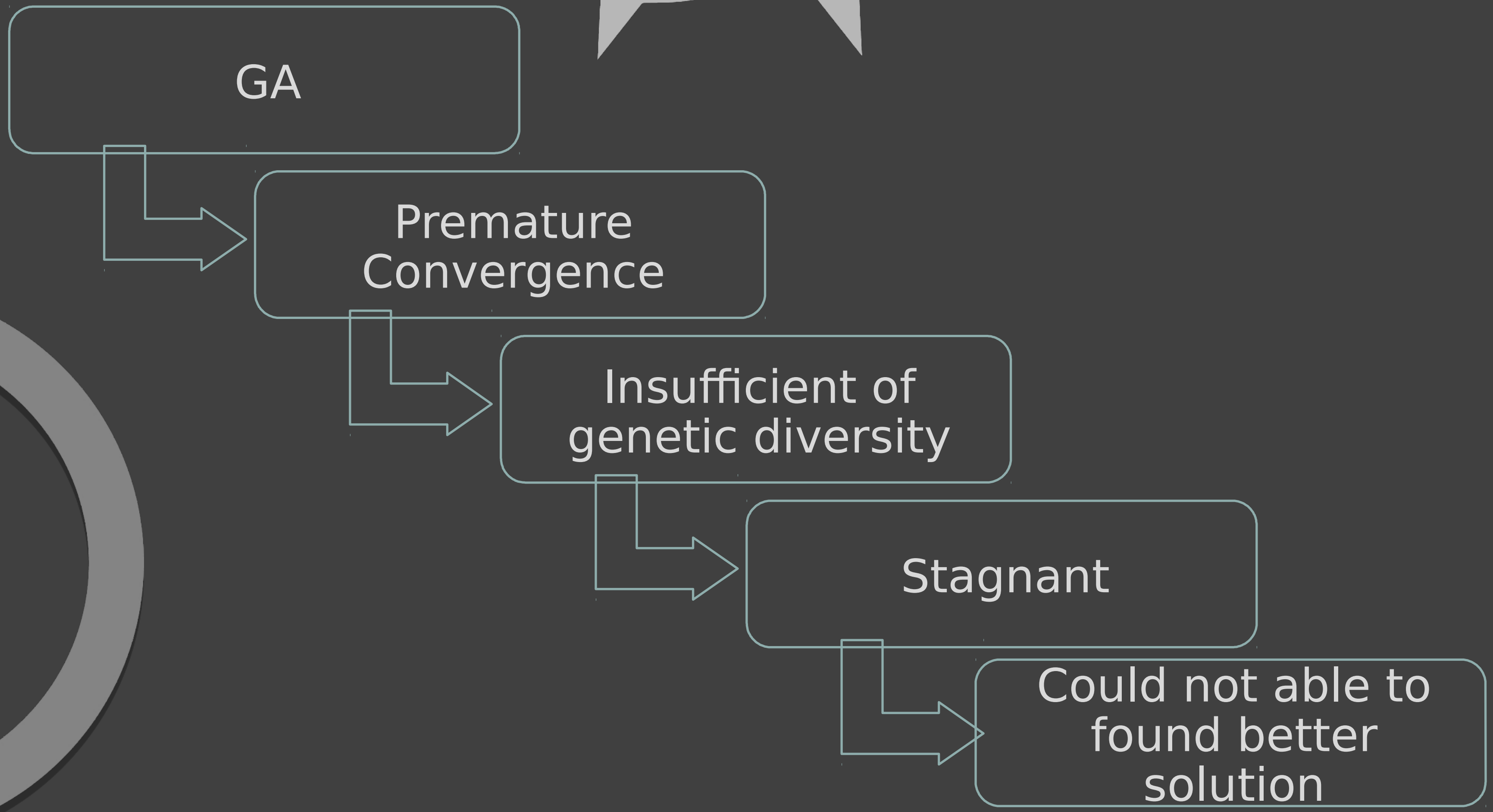
Introduction

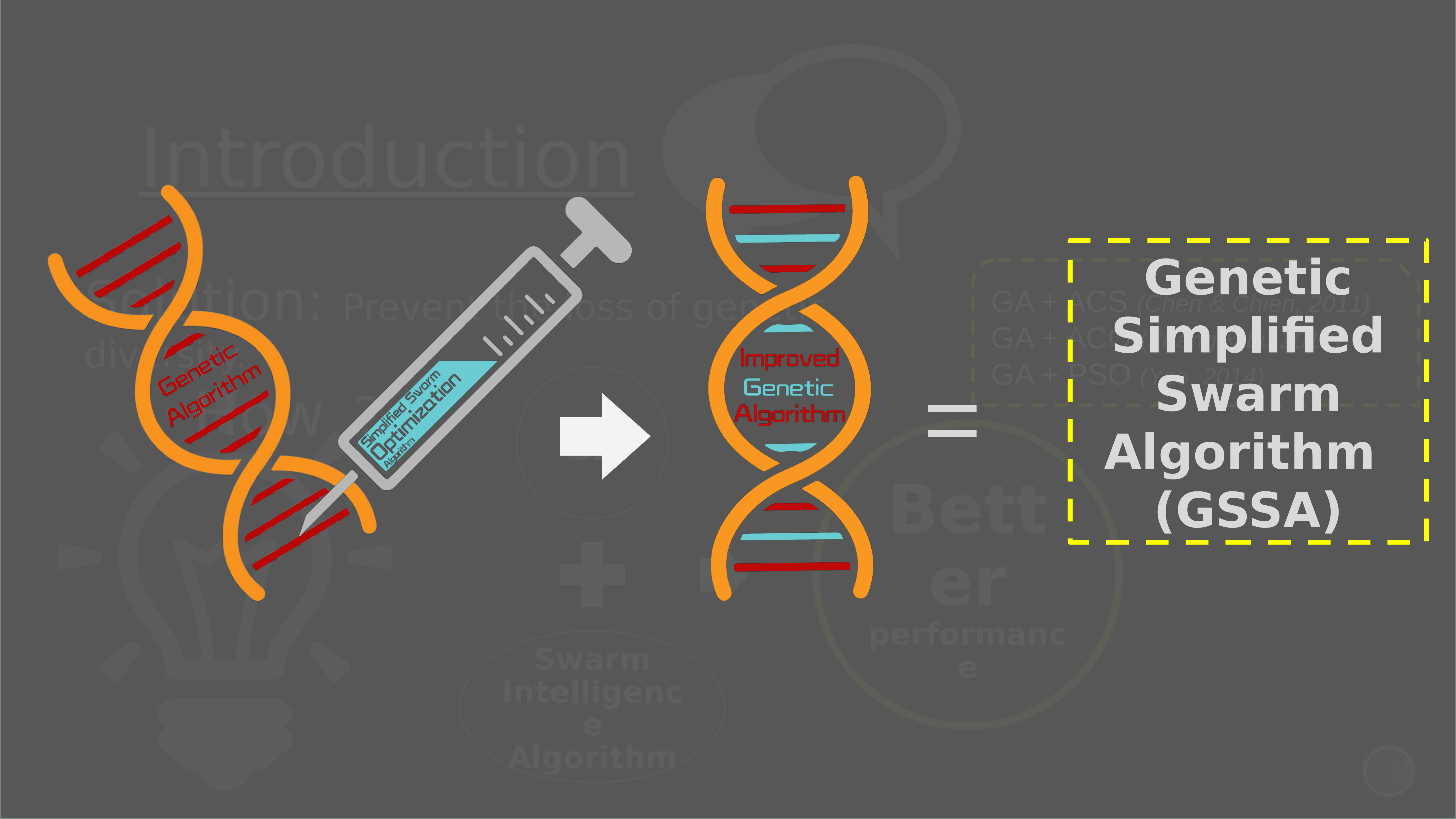
Solution for TSP and its variant:

Natural-inspired metaheuristic algorithms

- *Ant Colony Optimization (ACO)*
  (Hliang & Khine, 2011)
- *Particle Swarm Optimization (PSO)*
  (Gao et al., 2012)
- *Genetic Algorithm (GA)*
  (Nagata & Soler, 2012)
  - Khan *et al.*, 2009
  - Yang *et al.*, 2013
  - Liu, 2014
  - Etc….
- *Simulated Annealing (SA)*
  (Wang et al., 2012)
- *Tabu Search (TS)*
  (Pedro et al., 2013)

Introduction




Genetic Algorithm
Simplified Swarm Optimization Algorithm
Improved Genetic Algorithm
=
Genetic Simplified Swarm Algorithm (GSSA)

Introduction

Comparison Strategy

#SSO's characteristic

Swarm Intelligence Algorithm

Variant of PSO

Simplified Swarm Optimization

(SSO)

(Yeh, 2009)

#Better convergent rate

#Better accuracy

#Simpler, efficient and flexible

Developed to overcome the drawback of PSO.

#Premature Convergence

To apply and solve the data classification problem

Objectives

1. To propose a new extension of OTSP variant named *n*-Cities Open Loop Travelling Salesman Problem (*n*OTSP) based on transportation problem in today's reality.

2. To propose an improved technique of Genetic Algorithm (GA) with Simplified Swarm Optimization (SSO) algorithm's characteristic to prevent the loss of genetic diversity in the population.

3. To develop the propose technique in (2) for optimizing the *n*OTSP in term of finding the shortest path.

4. To evaluate the performance of the proposed technique with other GA variants in terms of shortest distance and population size.

Methodology

*n*OTSP Topology development

Development of Genetic Simplified Swarm Algorithm (GSSA)

Experimental Setup

Performance Evaluation

Methodology

nOTSP Topology development

- Nodes Generation.
- Nodes Mapping.
- Determining the starting and ending points.
- *n*OTSP Creation.

$$Node_{xy} = a*rand(m,n)$$

Parameter setting

a=50, m=50 and n=2

Outcome:

Node	**Coordinate (x,y)**
1	6.5961 4.5380
2	32.7739 30.8022
3	8.5593 23.6644
:	:
:	:
48	46.4632 40.8652
49	17.4992 43.4347
50	8.0033 1.3317

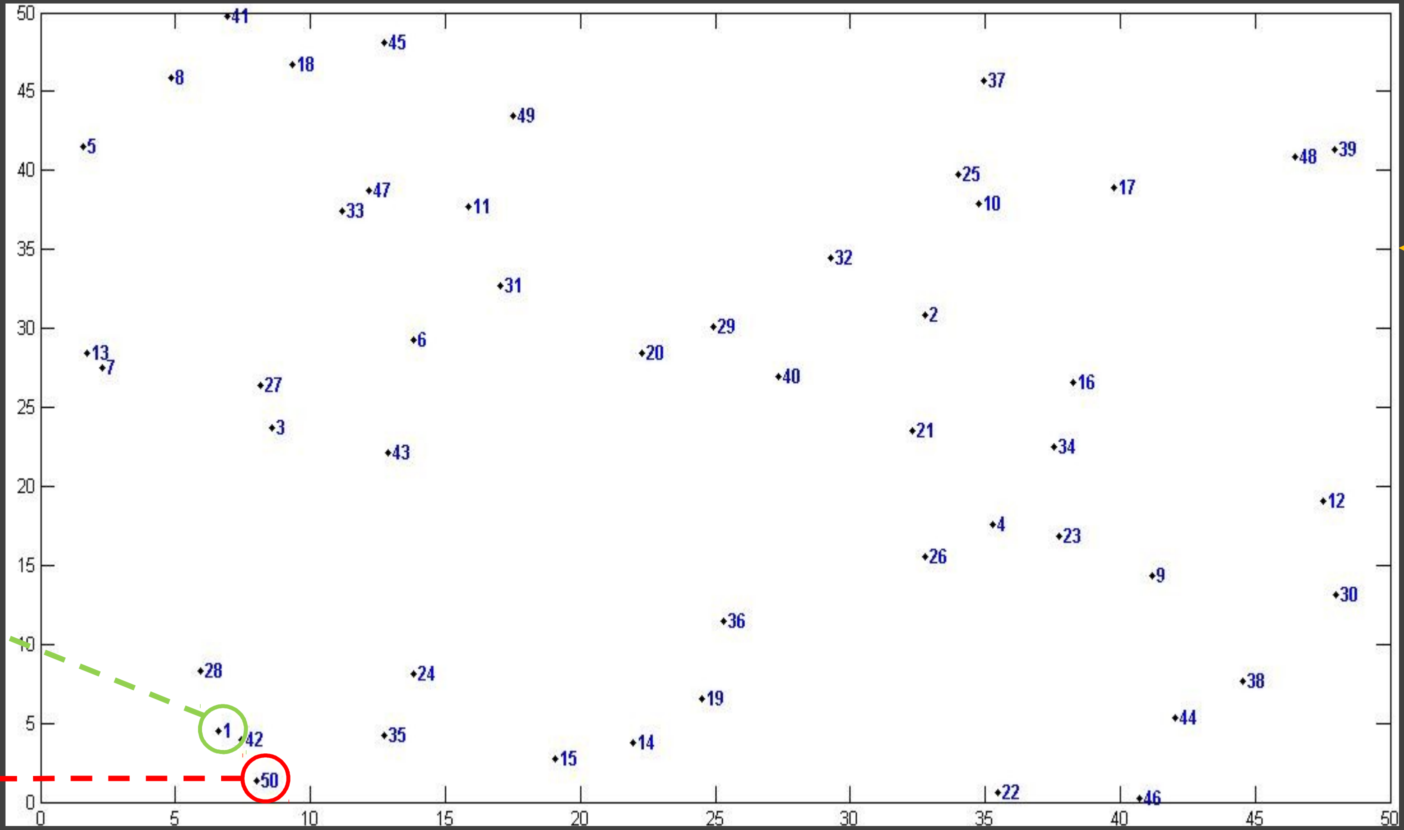
Starting point (node 1)

Ending point (node 50)

Methodology

nOTSP Topology development

- Nodes Generation.
- Nodes Mapping.
- Determining the starting and ending points.
- *n*OTSP Creation.



Methodology

_n_OTSP Topology development

- Nodes Generation.
- Nodes Mapping.
- Determining the starting and ending points.
- _n_OTSP Creation.

Before:

Node	Coordinate (x,y)
1	6.5961 4.5380
2	32.7739 30.8022
3	8.5593 23.6644
:	:
:	:
48	46.4632 40.8652
49	17.4992 43.4347
50	8.0033 1.3317

After:

Node	Coordinate (x,y)
1	0.00 0.00
2	32.7739 30.8022
3	8.5593 23.6644
:	:
:	:
48	46.4632 40.8652
49	17.4992 43.4347
50	50.00 50.00

Starting point

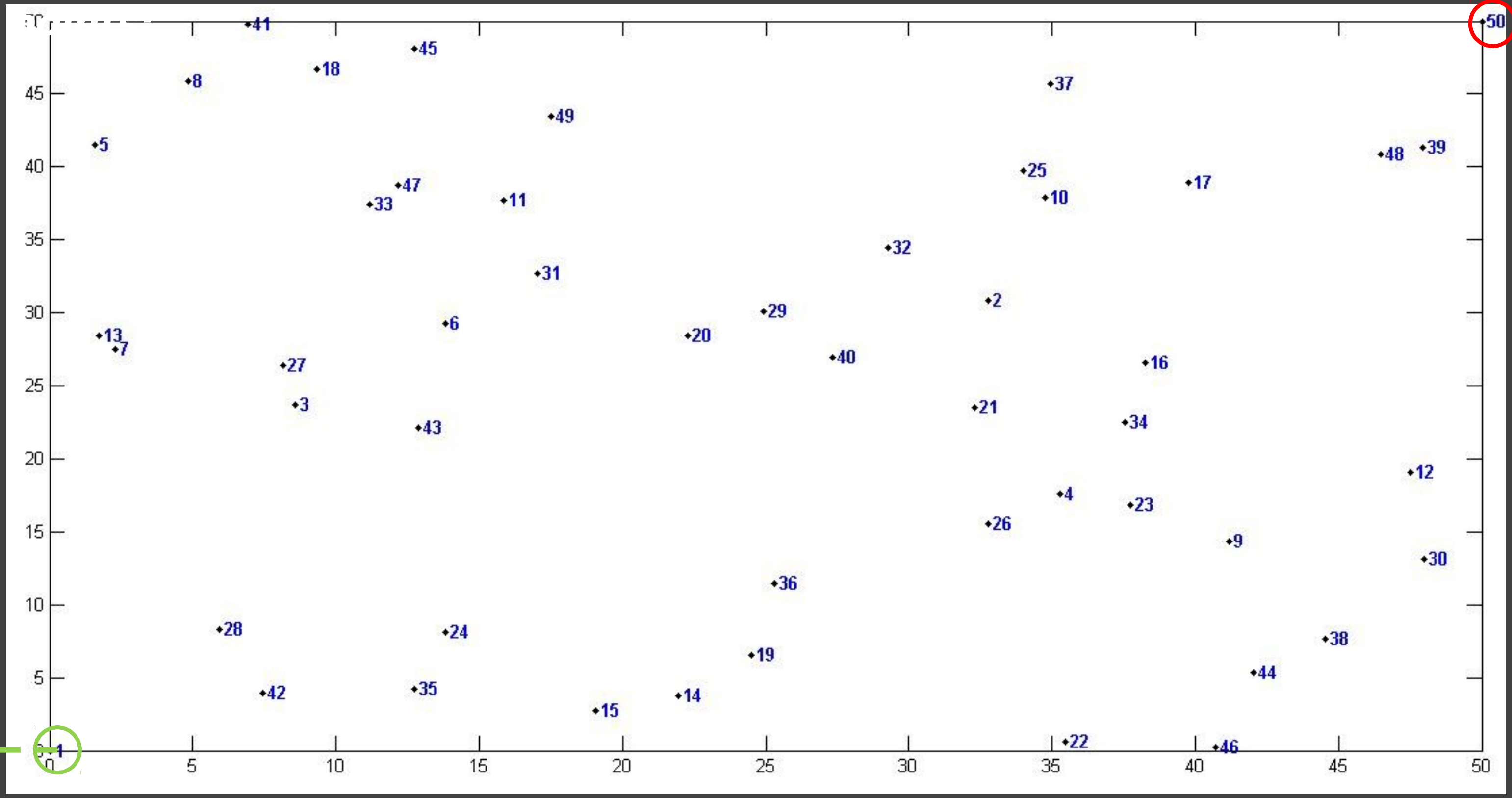
Ending point

Methodology

*n*OTSP Topology development

- Nodes Generation.
- Nodes Mapping.
- Determining the starting and ending points.
- *n*OTSP Creation.

New location of the starting and ending



Ending point

Starting point

Methodology

nOTSP Topology development

- Nodes Generation.
- Nodes Mapping.
- Determining the starting and ending points.
- *n*OTSP Creation.

Eu … $y_j)^2$

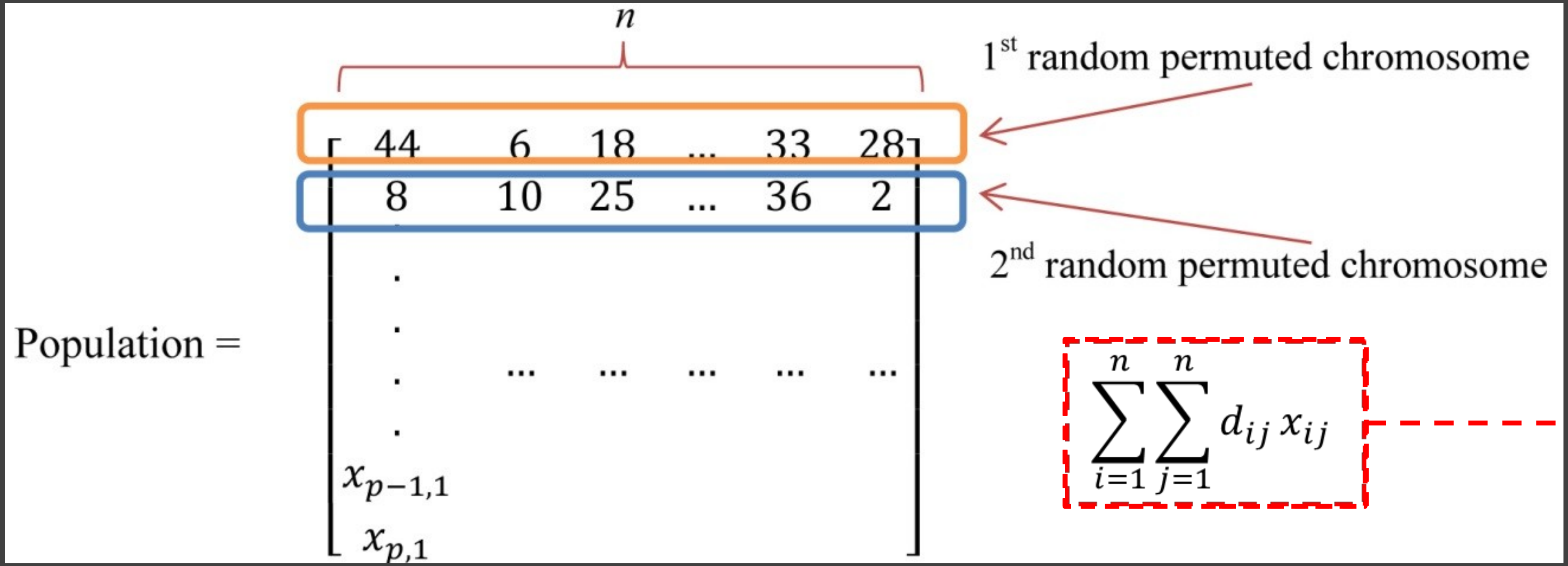
City	1	2	3	4	….	47	48	49	50
1	0	44.97	25.16	39.43	….	40.61	61.87	46.82	70.71
2	44.97	0	25.24	25.24	….	22.07	16.99	19.82	25.79
3	25.16	25.24	0	27.42	….	15.50	41.62	21.69	49.10
4	39.43	13.45	27.42	0	….	25.81	25.81	31.38	35.59
….	….	….	….	….	0	….	….	….	….
47	40.61	22.07	15.50	25.81	….	0	34.35	7.09	39.46
48	61.87	16.99	41.62	25.81	….	34.35	0	29.07	9.79
49	46.82	19.82	21.69	31.38	….	7.09	7.09	0	33.15
50	70.71	25.79	49.10	35.59	….	39.46	9.79	33.15	0

Symmetric distance-matrix

Methodology

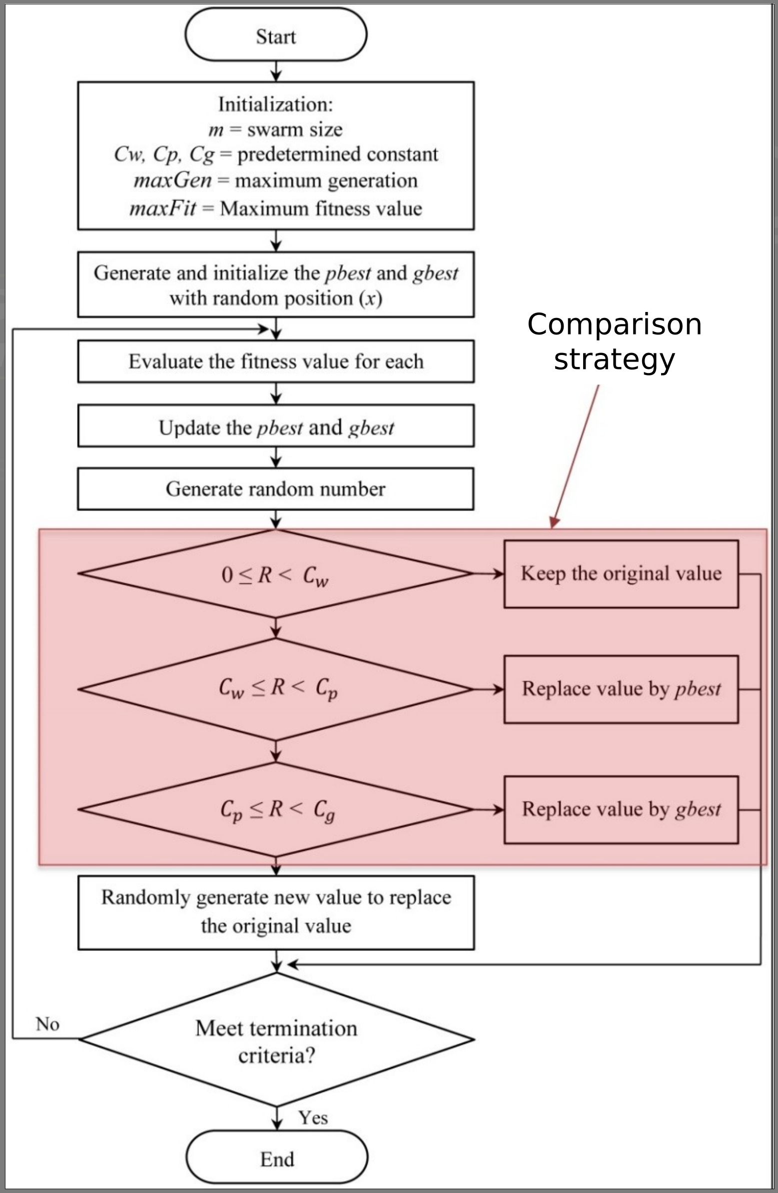
nOTSP Topology development

- Nodes Generation.
- Nodes Mapping.
- Determining the starting and ending points.
- *n*OTSP Creation.



Summation of the total distance of *n* cities.

SSO's flowchart



Comparison strategy

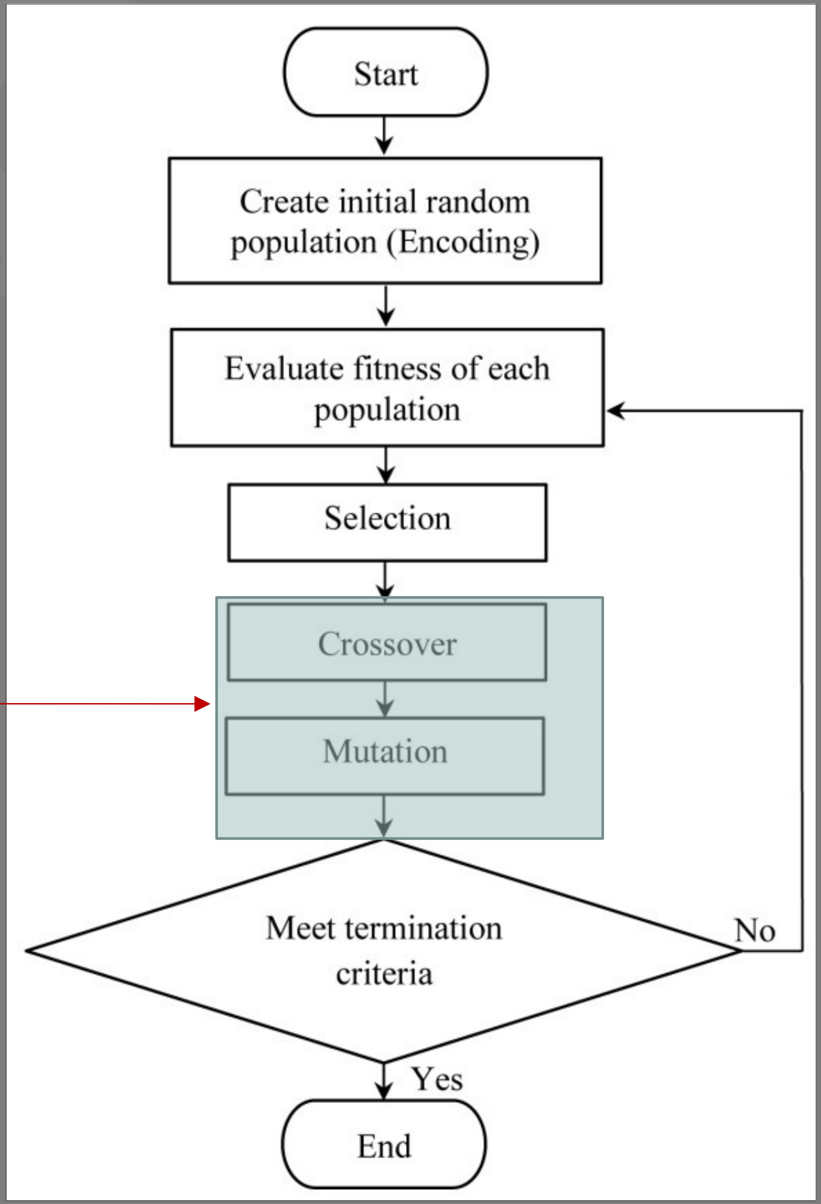
Mutation Operators:

1. Inversion mutation
2. Displacement mutation
3. Pairwise swap mutation

Adopted into comparison strategy

Embedded into GA

GA's flowchart



**comparison strategy is renamed to Solution Update Mechanism (SUM) after the modification

Methodology

Genetic Simplified Swarm Algorithm (GSSA)

- GSSA development.
- Process of GSSA.

Inversion mutation:

Before mutation	1 5 9 3 7 4 6 2 8 0
After mutation	1 5 2 6 4 7 3 9 8 0

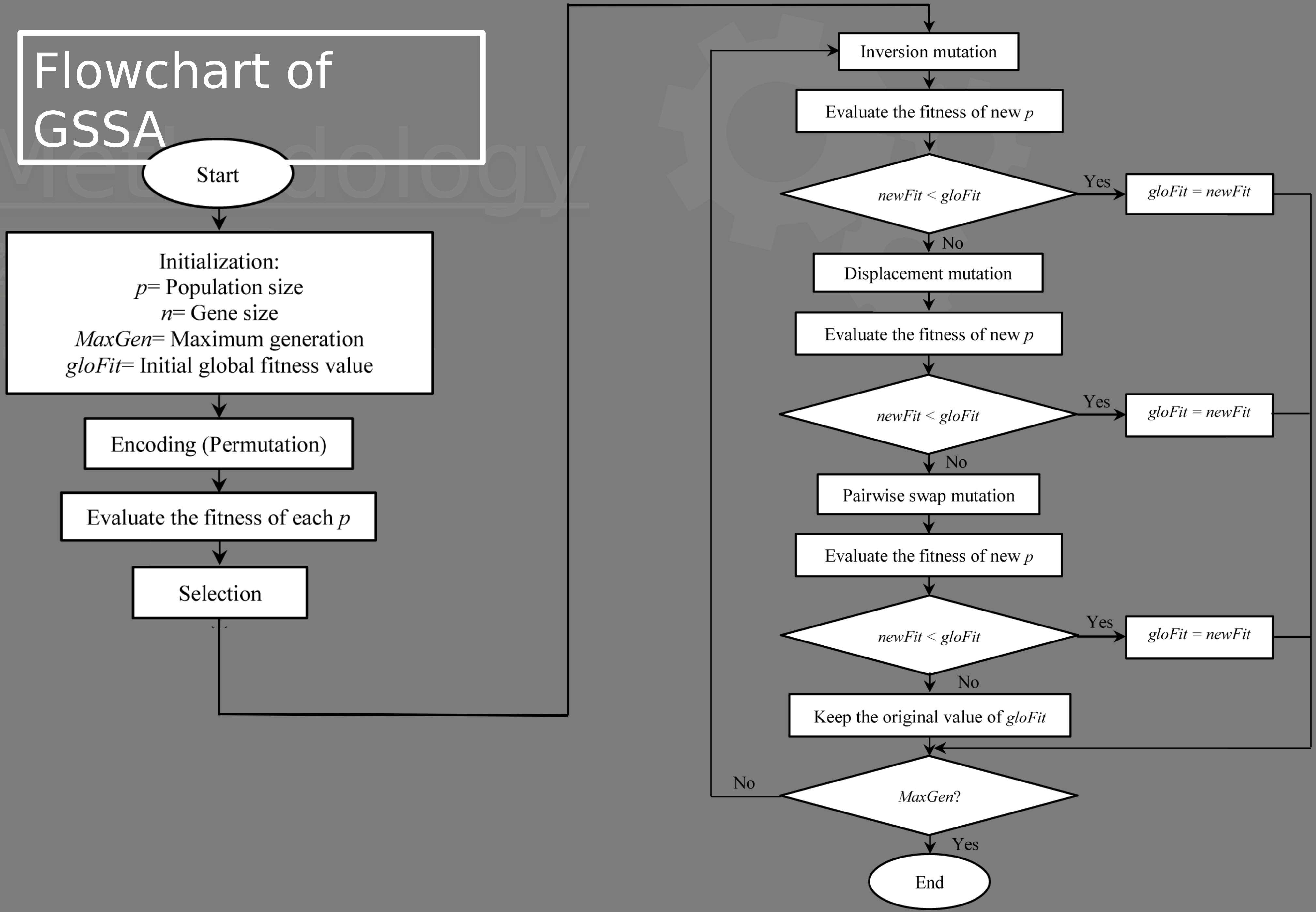
Displacement mutation:

Before mutation	1 5 9 3 7 4 6 2 8 0
After mutation	1 5 3 7 4 6 2 9 8 0

Pairwise Swap mutation:

Before mutation	1 5 9 3 7 4 6 2 8 0
After mutation	1 5 2 3 7 4 6 9 8 0

Flowchart of GSSA



Pre-processing results of determine *MaxGen* for *n*=10, 20, 30 and 40

n	*Maximum Break iteration*	*Average break iteration*	*MaxGen*
10	27	18	30
20	146	103	150
30	248	218	250
40	496	436	500

Methodology

Experimental Setup

- Pre-processing: Determining the *MaxGen*.
- *Parameter Setting and Data Collection.*

$m=50$

n

10

20

30

40

Execute for 10 times in all the population p, where p=1000, 2000, 3000, 4000 and 5000.

Methodology

Performance Evaluation

- The results (optimal solution and average) of GSSA will be compared with the results obtained by GA without crossover operator (GA-XX) and GA with one-point crossover operator (GA-1X).

Characteristics of the algorithms:

Algorithms	***Crossover Operator***	***Mutation Operator***
GSSA	No	Three mutation operators: 1. Inversion 2. Displacement 3. Pairwise Swap
GA-XX	No	
GA-1X	Yes, single point crossover	

- The research had also conducted an experiment to see the effects of the population size toward the algorithms.

Analysis & Result

Shortest distance (optimal solution)

		Population, p				
n	***Algorithm***	***1000***	***2000***	***3000***	***4000***	***5000***
10	***GSSA***	98.003	**88.511**	**95.289**	**87.922**	91.0164
	GA-XX	**94.373**	101.177	96.446	94.617	**89.760**
	GA-1X	130.389	113.874	124.686	115.113	122.958
20	***GSSA***	**157.972**	**149.713**	**158.063**	**152.154**	**139.064**
	GA-XX	163.740	162.133	164.168	165.537	146.251
	GA-1X	292.819	299.943	281.290	287.963	291.253
30	***GSSA***	**198.922**	**190.485**	**189.101**	**205.128**	**204.465**
	GA-XX	213.169	198.643	202.315	207.311	207.633
	GA-1X	498.971	481.285	477.500	495.247	496.988
40	***GSSA***	**243.425**	**235.712**	**228.921**	**236.649**	**239.535**
	GA-XX	251.667	239.178	249.184	249.995	251.859
	GA-1X	679.322	714.039	696.956	660.416	682.911

Analysis & Result

Average distance (average solution)

n	Algorithm	Population, p				
		1000	***2000***	***3000***	***4000***	***5000***
10	***GSSA***	**107.23**	**100.994**	**101.658**	**96.686**	**95.865**
	GA-XX	113.584	112.511	105.342	106.946	104.419
	GA-1X	144.732	133.404	132.311	130.761	131.565
20	***GSSA***	**172.704**	**169.517**	**167.28**	**165.464**	**154.146**
	GA-XX	178.467	174.248	175.114	178.381	169.444
	GA-1X	315.246	310.311	298.123	303.507	298.444
30	***GSSA***	**214.061**	**212.557**	**211.482**	**215.406**	**211.02**
	GA-XX	229.145	224.96	219.822	219.309	225.971
	GA-1X	524.062	522.027	504.504	514.896	512.536
40	***GSSA***	**257.427**	**246.475**	**245.556**	**247.43**	**246.636**
	GA-XX	261.962	257.345	260.275	256.422	261.82
	GA-1X	725.898	732.294	731.897	712.668	711.656

Shortest average distance

Analysis & Result

4 reasons why GSSA can produce better solutions?

1 **New characteristic of the algorithm**

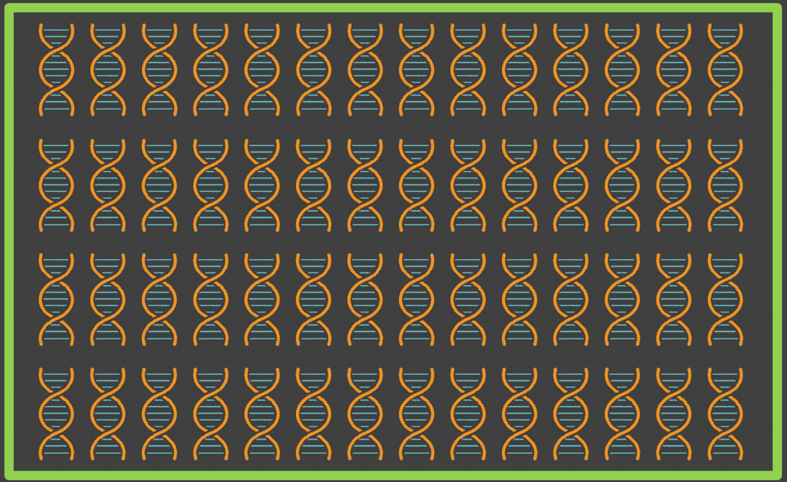
Comparison Strategy

+

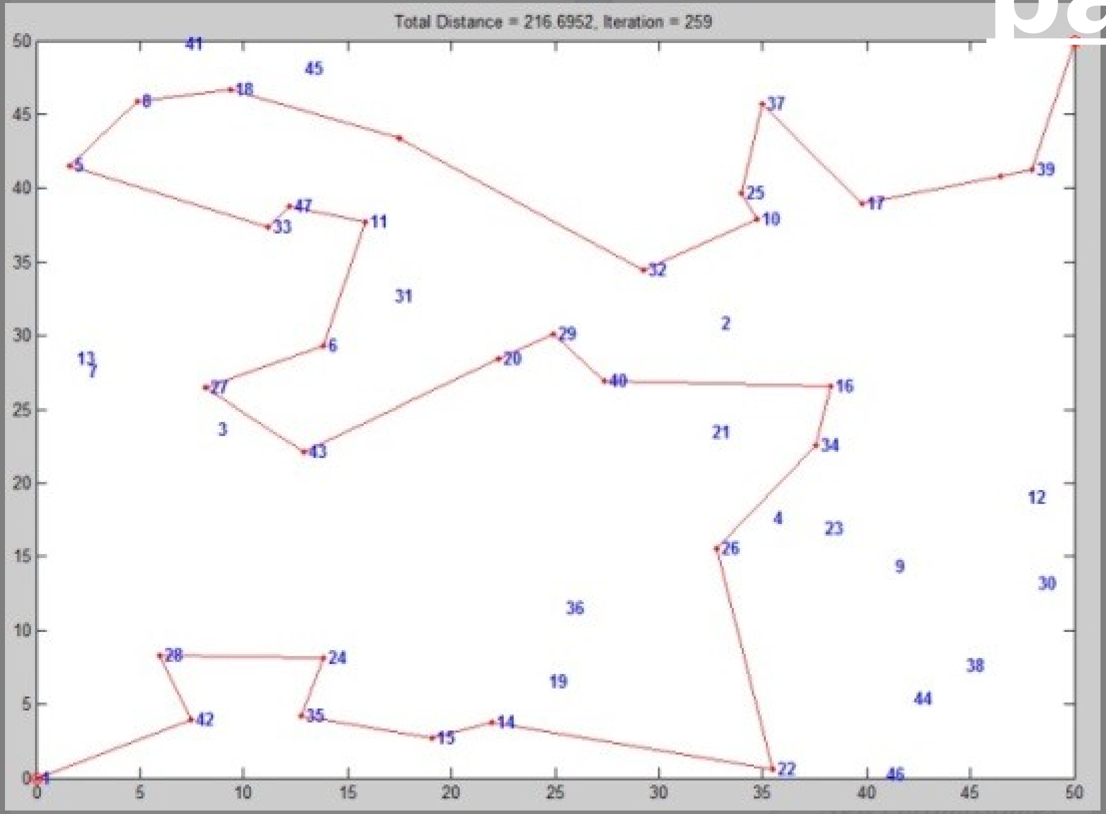
3 mutation operators

2 **Adequate genetic diversity**

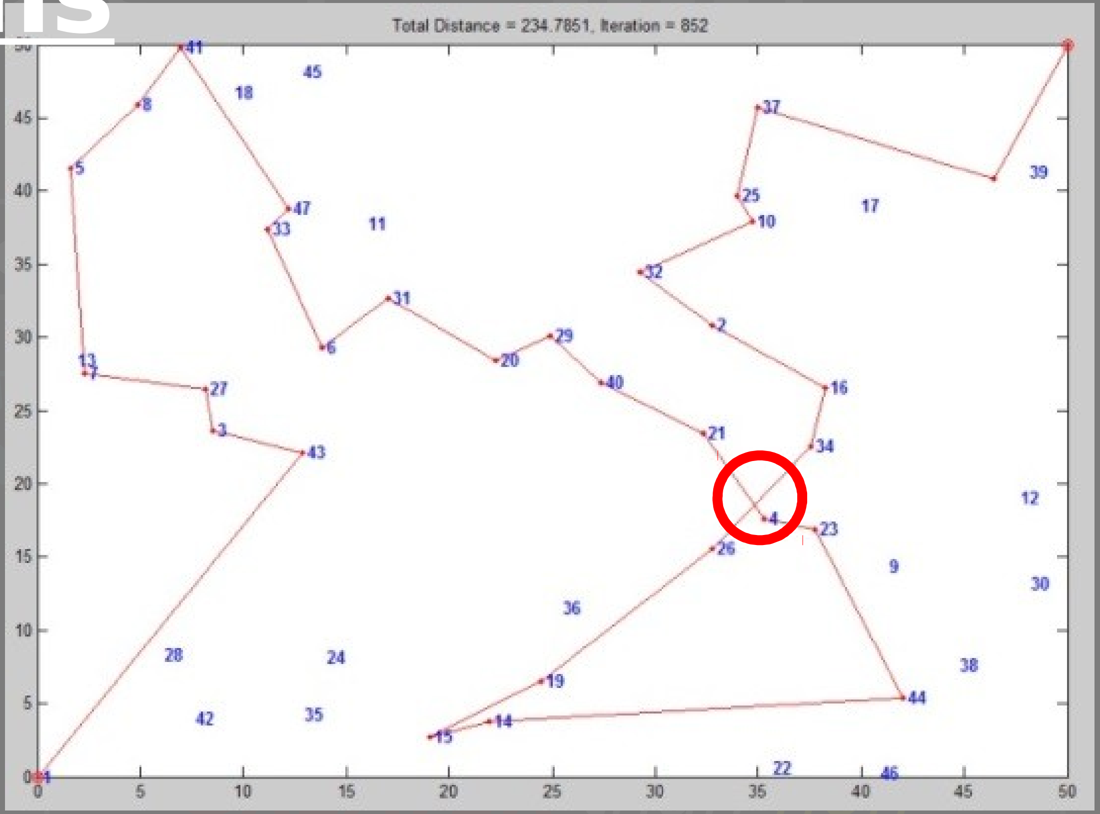
Search space:



Simulation of *n*OTSP's paths



Non-crossing path obtained by GSSA



Crossing path obtained by GA-XX

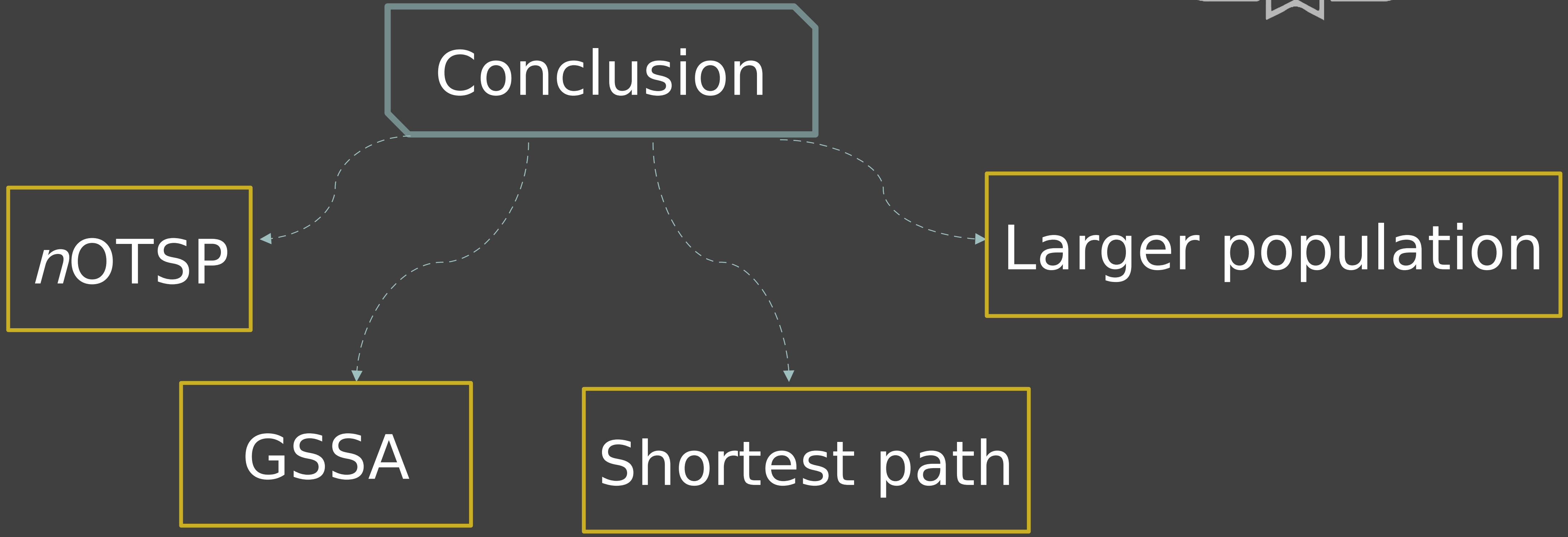
3000 ≤ p ≤ 5000

Algorithm: GSSA					
n	***Population, p***				
	1000	***2000***	***3000***	***4000***	***5000***
10	107.23	100.994	101.659	96.686	**95.865**
20	172.704	169.517	167.28	165.464	**154.146**
30	214.061	212.557	211.482	215.406	**211.02**
40	257.427	246.475	**245.557**	247.43	246.636

Algorithm: GA-XX					
n	***Population, p***				
	1000	***2000***	***3000***	***4000***	***5000***
10	113.584	112.511	105.342	106.946	**104.42**
20	178.467	174.248	175.115	178.381	**169.444**
30	229.145	224.960	219.822	**219.31**	225.971
40	261.962	257.345	260.275	**256.423**	261.820

Algorithm: GA-1X					
n	***Population, p***				
	1000	***2000***	***3000***	***4000***	***5000***
10	144.732	133.404	132.311	**130.761**	131.565
20	315.246	310.311	**298.123**	303.507	298.444
30	524.062	522.027	**504.504**	514.896	512.5367
40	725.898	732.294	731.897	712.668	**711.657**

Conclusion & Future Work



Conclusion & Future Work

Future works

- *Greedy Sub Tour Mutation (GSTM)*
- Improve/modify structure of the algorithm
- Invent new permutation based algorithm
- Modified SSO to optimize *n*OTSP

Publications

1. Chieng, H. H., and Wahid, N. (2014). *A Performance Comparison of Genetic Algorithm's Mutation Operators in n-Cities Open Loop Travelling Salesman Problem*. Recent Advances on Soft Computing and Data Mining (SCDM) pp. 89-97. Springer International Publishing. (Indexed by ISI, DBLP, EI-Compendex, Scopus).

2. Chieng, H. H., and Wahid, N. (2015). *An Improved GA-Based Algorithm for Travelling Salesman Problem*. International Journal of Advancements in Computing Technology (IJACT), (ISSN: 2005-8039). Vol. 8, No. 3.

3. Chieng, H. H., and Wahid, N. (2016). *Genetic Simplified Swarm Optimization for n-Cities Open Loop travelling salesman problem*. **(Submitting)**

4. Chieng, H. H., and Wahid, N. (2016). *Animal-Inspired Metaheuristic Algorithms' Directory*. **(Progressing…)**

{ Thank you for your attention }