

Implementasi Cuucko Search Terhadap Jarak Terbaik Antar Kota

METODE TRAVELLING SALESMAN PROBLEM



SWARM INTELLIGENCE
KELOMPOK 4 - RB

Our Great Team



Vicky Aditia
120450032



M Nabil Azizi
120450090



M Dafha Syahrizal
120450050



M Aditya P Suhendar
120450058

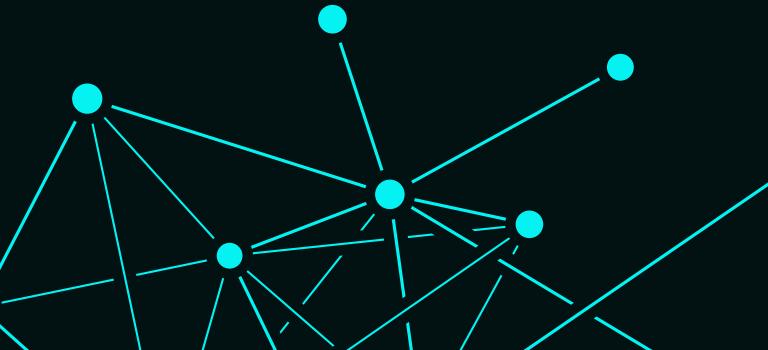


Table Of Content

01

Introduction

02

Problem
Statement

03

Data
Description

04

Method

05

Individual
Modelling

06

Results &
Analysis

07

Summary



Introduction

Algoritma Cuckoo Search

Terinspirasi dari cara burung cuckoo berkembang biak

Meniru tingkah laku dari spesies cuckoo yaitu sebuah parasit yang meletakkan telurnya di sarang burung lain



Problem Statement

Ada banyak kombinasi rute yang mungkin muncul dan harus dihitung

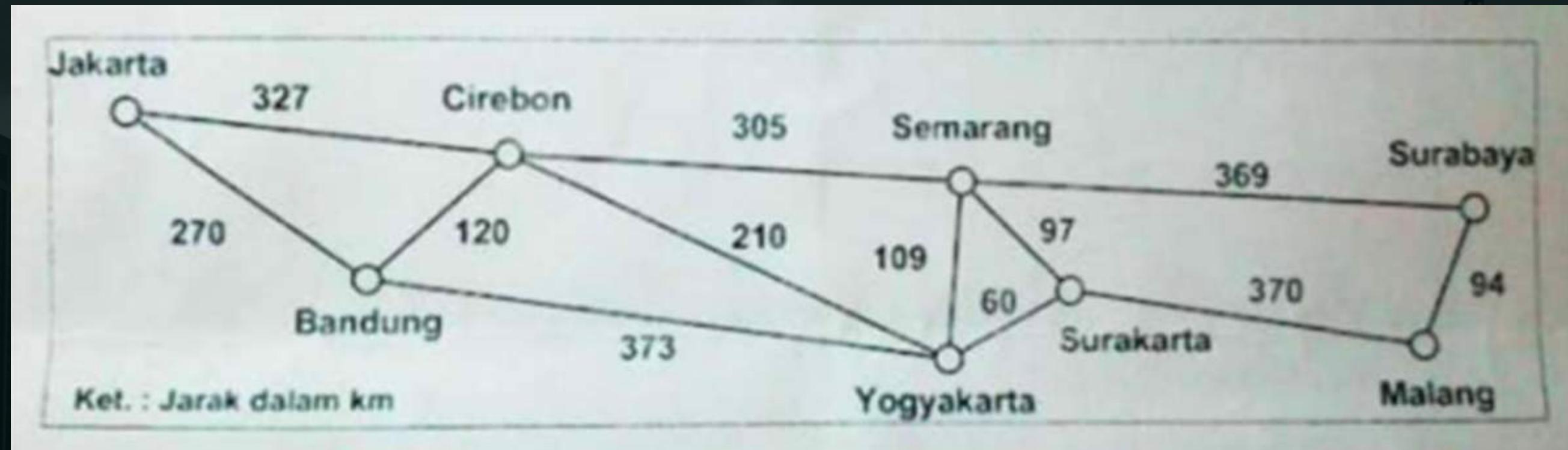
Pendekatan heuristic mempersingkat waktu

Diperlukan sebuah sistem yang memanfaatkan teknologi informasi guna melakukan proses pencarian jalur optimum secara otomatis.

Masalah yang Dirumuskan

1. Berapa banyak rute dari jakarta ke malang?
2. Bagaimana algoritma CS menyeleksi rute terbaik dari jakarta ke malang?
3. Apa rute terbaik yang disimpulkan berdasarkan hasil algoritma CS?

Data Description



Kami menggunakan data beberapa kota yang ada di pulau jawa, beserta jaraknya.

Data Description

Tahun	Nama Provinsi	Nama Kota	Jarak
2015	Jawa Barat	Bandung	270
2015	Jawa Barat	Cirebon	327
2015	Yogyakarta	Yogyakarta	582
2015	Jawa Tengah	Semarang	482
2015	Jawa Tengah	Surakarta	657
2015	Jawa Timur	Surabaya	789
2015	Jawa Timur	Malang	964

Tabel Data Jarak Jakarta dengan Masing-Masing Kota.

Data Description

	JKT	BDG	Cirebon	Yogya	SMRG	SRKT	SRBY	MLG
Jakarta	0	1	1	0	0	0	0	0
Bandung	1	0	1	1	0	0	0	0
Cirebon	1	1	0	1	1	0	0	0
Yogyakarta	0	1	1	0	1	1	0	0
Semarang	0	0	1	1	0	1	1	0
Surakarta	0	0	0	1	1	0	0	1
Surabaya	0	0	0	0	1	1	0	1
Malang	0	0	0	0	0	1	1	0

Matriks Ketetanggaan

Method

Ada tiga pendekatan yang digunakan dalam Cuckoo Search Algorithm (CSA) dirumuskan sebagai berikut :



- Setiap cuckoo meletakkan satu telur pada satu waktu, dan membuang telur dalam sarang yang dipilih secara acak.
- Sarang terbaik dengan telur yang berkualitas sebagai solusi yang akan digunakan untuk generasi selanjutnya.
- Jumlah sarang burung lain yang tersedia adalah tetap, dan burung pemilik sarang dapat menemukan telur cuckoo dengan probabilitas $pa \in (0,1)$. Dalam kasus ini, burung pemilik sarang dapat membuang telur cuckoo atau meninggalkan sarang sehingga pemilik sarang dapat membangun sarang baru di lokasi yang baru.

Pseucode

Cuckoo search algorithm

Define Objective function $f(x)$, $x = (x_1, x_2, \dots, x_d)$

Initial a population of n host nests $x_i(i=1,2,\dots,d)$

while ($t < \text{MaxGeneration}$) or (stop criterion);

Get a cuckoo (say i) randomly

and generate a new solution by Lévy flights;

Evaluate its quality/fitness; F_i

Choose a nest among n (say j) randomly;

if ($F_i > F_j$),

Replace j by the new solution;

end

Abandon a fraction (P_a) of worse nests

[and build new ones at new locations via Lévy flights];

Keep the best solutions (or nests with quality solutions);

Rank the solutions and find the current best;

end while

Post process results and visualization;

End

Individual Modelling



```
array([[2.1952072 , 4.86623395, 9.90119091, 7.04606005, 9.71849951,
       7.68734731, 3.10934122, 4.16749171],
      [7.03850973, 9.63339026, 8.49386438, 3.31397264, 3.26182791,
       2.08474803, 6.54860587, 6.80566077],
      [4.66820429, 4.81393559, 4.20751264, 3.17222867, 9.76361527,
       6.48667795, 8.76366454, 5.62670583],
      [4.31404143, 6.05331384, 9.22002875, 3.98041455, 7.69161805,
       3.65380236, 6.3205878 , 9.14609692],
      [8.61234482, 5.71712611, 4.723789 , 6.22865074, 8.61397726,
       7.75679775, 7.38802192, 7.40943861],
      [7.31005057, 4.35548349, 9.64627757, 9.87030778, 5.05081938,
       8.91898704, 6.37742276, 9.35140215],
      [4.62528434, 8.89425131, 6.54451284, 6.52197027, 2.30006366,
       3.31163209, 9.2277901 , 4.85800229],
      [8.73118608, 5.31545804, 8.64933676, 6.11231348, 4.29046226,
       4.77577809, 6.90242358, 6.61697674],
      [8.47063558, 3.23121078, 6.54899196, 4.18222132, 7.36347669,
       4.96164536, 6.08005521, 5.74837665],
      [8.4734408 , 9.57396743, 7.02305139, 2.95964903, 4.05486897,
       6.56760438, 9.98236226, 7.02410509],
      [9.24158616, 8.73688964, 3.81101818, 4.86246986, 4.69175746,
       9.27183316, 3.43377834, 3.38551956],
      [5.90096574, 9.2812843 , 7.32212024, 4.56169444, 9.20410198,
       2.47244888, 8.30908518, 2.56087627],
      [6.63851075, 2.06635322, 8.63008092, 3.32262126, 4.35188742,
       7.40051394, 5.97560453, 8.78998864],
      [9.07848406, 8.04649112, 2.81410197, 4.03707042, 9.01595877,
       7.44567869, 6.60925679, 8.81521558],
      [3.74607599, 3.03273079, 9.83979158, 5.87161527, 8.46537534,
       4.18793154, 7.96239442, 7.41711519]])
```

Results & Analysis

Visualisasi hubungan antara parameter dan solusi terbaik.
Parameter yang diobservasi: parameter metode , jumlah individu , maksimal iterasi

Hasil n individu 50 dan 100

Urutan 1	Urutan 2	Urutan 3	Urutan 4	Urutan 5	Urutan 6	Urutan 7	Urutan 8	Jarak	
0	3	4	5	2	7	0	1	6	476.0
1	2	3	6	5	1	7	4	0	537.0
2	5	3	0	1	6	4	7	2	699.0
3	5	4	3	6	7	0	2	1	747.0
4	4	5	0	7	3	2	1	6	796.0
5	1	3	4	6	0	5	2	7	851.0
6	4	5	1	3	7	6	2	0	891.0
7	5	7	3	0	2	6	4	1	1066.0
8	1	0	5	3	7	6	4	2	1218.0
9	0	6	7	1	3	5	4	2	1256.0
10	2	1	3	6	7	5	0	4	1262.0
11	1	4	6	0	5	7	2	3	1322.0
12	1	3	5	7	6	2	4	0	1472.0
13	7	6	2	4	3	1	0	5	1521.0
14	3	2	0	1	7	5	6	4	1655.0

Urutan 1	Urutan 2	Urutan 3	Urutan 4	Urutan 5	Urutan 6	Urutan 7	Urutan 8	Jarak	
0	1	5	3	6	4	0	2	7	109.0
1	0	4	1	2	5	6	3	7	120.0
2	4	0	6	2	7	1	5	3	169.0
3	5	3	6	0	4	7	2	1	180.0
4	6	7	4	1	2	5	0	3	214.0
5	7	5	3	6	0	4	7	2	225.0
6	6	4	0	2	5	7	3	1	225.0
7	3	4	2	0	7	5	6	1	225.0
8	5	6	4	0	2	7	3	1	1439.0
9	4	3	7	5	2	0	1	6	1445.0
10	3	4	2	0	7	5	6	1	1484.0
11	6	5	4	0	2	7	0	1	1687.0
12	2	3	1	0	7	5	4	6	1746.0

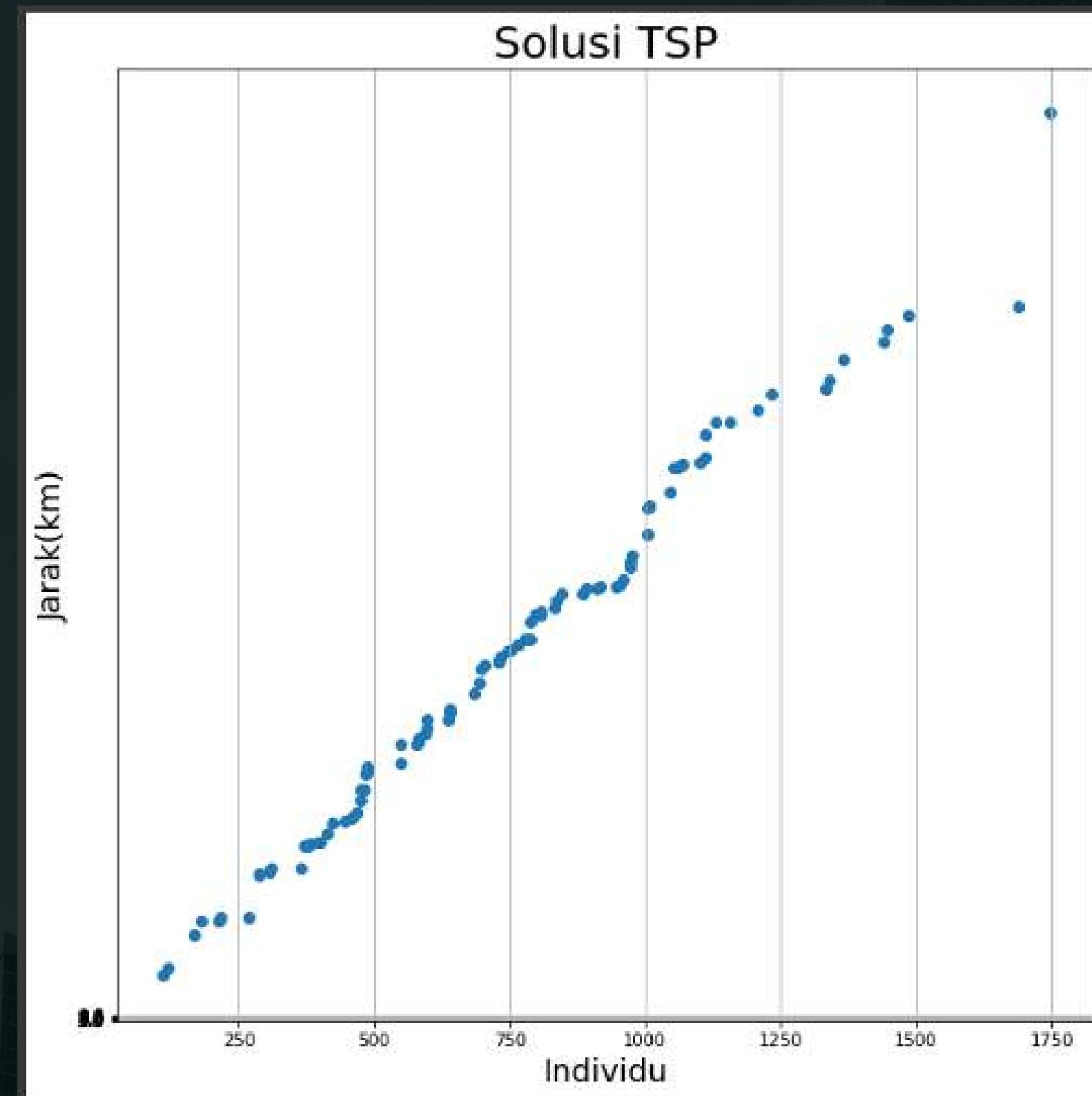
Results & Analysis

Solusi Terbaik, parameter terbaik , iterasi berhenti

	Urutan 1	Urutan 2	Urutan 3	Urutan 4	Urutan 5	Urutan 6	Urutan 7	Urutan 8	Jarak
0	1	6	2	5	0	4	3	7	109.0
1	0	4	1	2	5	6	3	7	120.0
2	4	0	6	2	7	1	5	3	169.0
3	5	3	6	0	4	7	2	1	180.0
4	6	7	4	1	2	5	0	3	214.0
...
95	6	4	0	2	5	7	3	1	1439.0
96	4	3	7	5	2	0	1	6	1445.0
97	3	4	2	0	7	5	6	1	1484.0
98	3	6	4	2	5	7	0	1	1687.0
99	2	3	1	7	5	4	6	0	1746.0

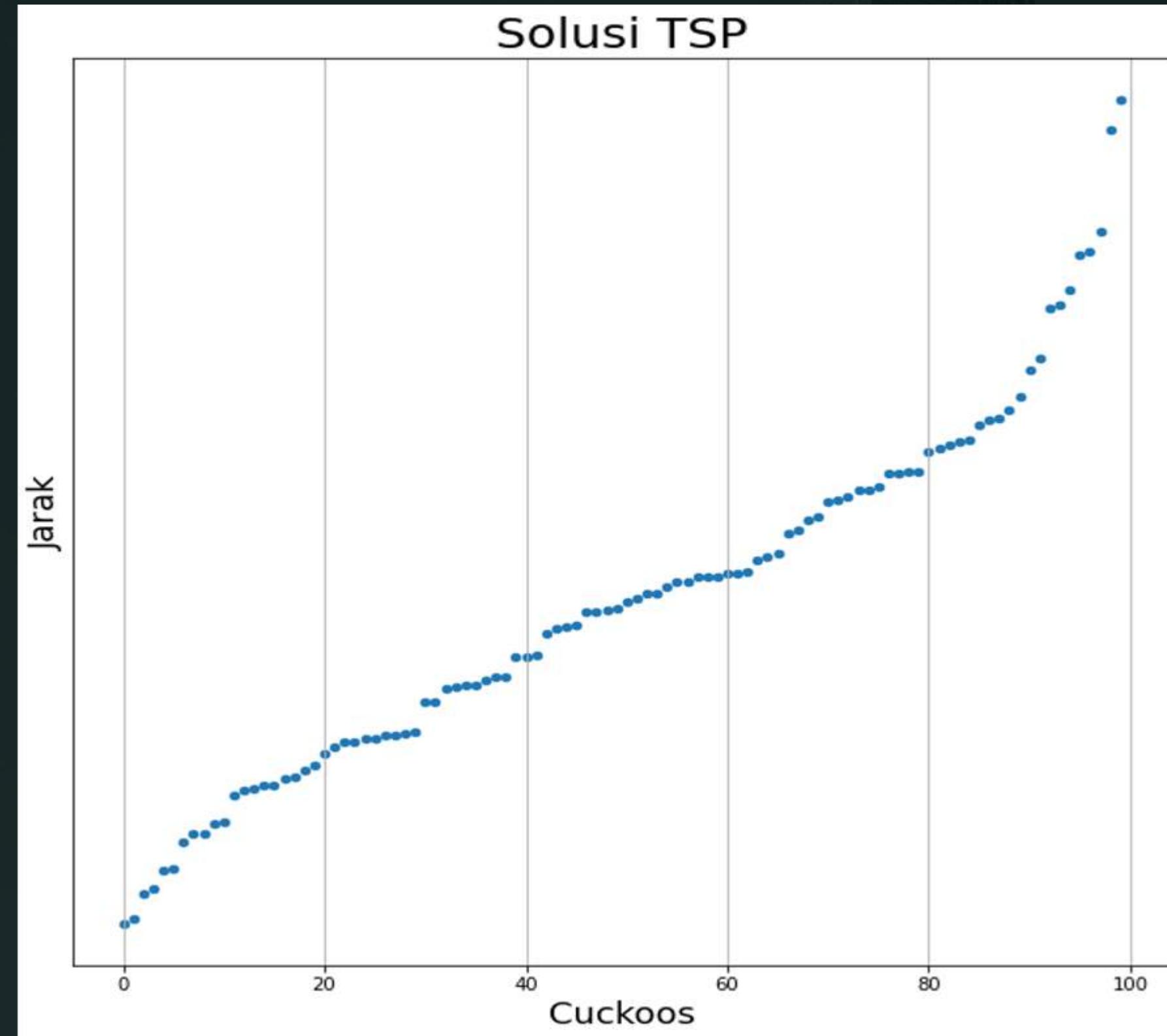
Results & Analysis

Visualisasi hubungan antara parameter dan solusi terbaik.
Parameter yang diobservasi: parameter metode , jumlah individu , maksimal iterasi



Results & Analysis

Visualisasi solusi terhadap iterasi



Summary

Pada permasalahan ini dengan membawa kasus jasa pengiriman barang semakin panjang jarak tempuh sebuah rute maka biaya dan waktu yang dibutuhkan semakin besar, sebaliknya semakin pendek jarak yang tempuh sebuah rute maka biaya dan waktu yang dibutuhkan semakin kecil.

Menggunakan TSP dengan metode cuckoo search didapatkan penyelesaian dengan hasil yaitu dari percobaan algoritma Cuckoo Search dengan metode TSP **didapatkan bahwa rute terpendek dengan jarak terbaik 109 KM adalah Bandung - Surabaya - Cirebon - Surakarta - Jakarta - Semarang - Yogyakarta - Malang**

TERIMA KASIH

Bapak/ibu dan teman-teman

Semoga

SEHAT DAN SUKSES SELALU

SALAM SEHAT & TETAP SEMANGAT

dari vicky, nabil,
dafha, adit