

IMPLEMENTASI ALGORITMA I-SOS DALAM PENYELESAIAN TRAVELING SALESMAN PROBLEM (TSP)

(IMPLEMENTATION OF I-SOS ALGORITHM IN SOLUTION THE TRAVELLING SALESMAN PROBLEM TSP)

Zulkarnaen¹⁾, Muhammad Azmi²⁾

¹⁾Teknik Informatika, Sistem Informasi²⁾. STMIK Syaikh Zainuddin NW
Jl. Raya Mataram Lb. Lombok KM.49 Anjani Lombok Timur - NTB

e-mail: zolcakep@gmail.com¹⁾, muhammad4zmi@gmail.com²⁾

ABSTRAK

Permasalahan pada TSP merupakan upaya untuk menemukan jarak tempuh dengan jarak perjalanan terpendek yang harus dilalui oleh salesman dalam mengunjungi setiap kota tanpa harus mendatangi kota yang sama lebih dari satu kali. Tujuan dari penelitian ini untuk dapat menemukan formulasi rute terpendek dalam permasalahan jarak tempuh pada kasus TSP menggunakan algoritma I-SOS. Solusi dapat diperoleh setelah melalui perhitungan fase mutualisme, fase komensalisme, fase parasitisme dan fase predasi dijalankan. Penyelesaian permasalahan TSP pada penelitian diawali dengan proses identifikasi setiap kota dengan cara memberikan nilai random untuk mewakili setiap kota tujuan. Nilai random yang digunakan antara 0 sampai 1, hasil random yang diperoleh kemudian akan di sorting dengan ketentuan nilai random terkecil akan dijadikan sebagai inisial terhadap kota A sedangkan nilai random terbesar dijadikan sebagai inisial terhadap kota D. Hasil akhir dari proses algoritma yang dilakukan apabila kota keberangkatan tidak ditentukan dapat menghasilkan total jarak tempuh terpendek untuk ke empat solusi adalah sejauh 45km dengan representasi solusi pertama $|0.25|0.27|0.5|0.75|$ dengan urutan kota $|A - B - C - D|$, solusi kedua $|B - A - D - C|$ dengan nilai random $|0.27|0.25|0.75|0.5|$, solusi ke tiga $|B - C - D - A|$ urutan nilai random $|0.27|0.5|0.75|0.25|$ dan solusi ke empat yakni $|D - A - B - C|$ dengan nilai random $|0.75|0.25|0.27|0.5|$.

Kata Kunci: Algoritma I-SOS, Komensalisme, Mutualisme, Parasitisme, Predasi, TSP.

ABSTRACT

The problem with TSP is an attempt to find the shortest distance traveled by a salesman in visiting each city without having to visit the same city more than once. The purpose of implementing the I-SOS algorithm in this case is to find the minimum distance traveled, a solution can be obtained after going through the calculation of the mutualism phase, commensalism phase, parasitism phase and predation phase. The resolution of TSP problems in the study begins with the process of identifying each city by providing a random value to represent each destination city. The random value used is between 0 and 1, the random results obtained will then be sorted with the provision that the smallest random value will be used as the initial for city A while the largest random value is used as the initial for city D. The final result of the algorithm process that is carried out if the departure city is not determined can produce the shortest total distance traveled for the four solutions as far as 45km with the first solution representation $|0.25|0.27|0.5|0.75|$ with the order of cities $|A - B - C - D|$, the second solution $|B - A - D - C|$ with a random value $|0.27|0.25|0.75|0.5|$, the third solution $|B - C - D - A|$ sequence of random values $|0.27|0.5|0.75|0.25|$ and the fourth solution, namely $|D - A - B - C|$ with a random value $|0.75|0.25|0.27|0.5|$.

Keywords: Algoritma I-SOS, Commensalism, Mutualism, Parasitism, Predation, TSP.

I. PENDAHULUAN

TSP merupakan suatu permasalahan optimasi klasik dalam penentuan rute sebagai jarak tempuh terpendek pada suatu lintasan distribusi. Permasalahan ini dapat analogikan pada seorang kurir harus mengunjungi setiap kota dalam lintasan sebanyak satu kali dalam satu kali perjalanan. Permasalahan *traveling salesman* (TSP) termasuk dalam persoalan optimasi yang dinyatakan sebagai penentuan rute perjalanan termurah seorang kurir untuk mengunjungi setiap node atau konsumen dengan syarat setiap

konsumen dapat dipastikan untuk dikunjungi satu kali dalam setiap perjalanan. Hingga kini belum ada metode baku yang dianggap paling optimal dalam penyelesaian traveling salesman problem TSP selain dengan mencoba seluruh kemungkinan yang ada sebagai solusi. Penentuan rute perjalanan merupakan salah satu permasalahan yang sering dihadapi dalam kehidupan sehari-hari. Salah satu contoh yaitu rute manakah yang memiliki biaya paling murah untuk dilalui seorang salesman ketika harus mengunjungi sejumlah daerah. Setiap daerah tersebut harus dikunjungi

tepat satu kali kemudain kembali lagi ke tempat asal perjalan atau kota keberangkatan.

II. STUDI PUSTAKA

TSP merupakan suatu permasalahan optimasi yang berifat klasik dimana tidak ada penyelesaian yang paling optimal selain mencoba seluruh kemungkinan penyelesaian yang ada. Permasalahan dalam TSP melibatkan seorang salesman yang harus melakukan kunjungan sekali pada semua kota dalam sebuah lintasan atau jalur sebelum dia kembali ke kota asal, supaya perjalanan yang dilakukan dapat dikatakan sempurna[2]. TSP merupakan salah satu permasalahan kombiantorial di bidang transfortasi untuk menemukan suatu rute perjalanan terpendek yang dapat di tempuh dari titik awal keberangkatan menuju titik tujuan, serta meminimumkan biaya perjalanan dan waktu tempuh perjalanan[4]. Teknik optimasi yang paling baru adalah *symbiotic organism search* atau yang dikenal dengan metode SOS. Metode ini dikembangkan oleh cheng dan prayogo pada tahun 2014 yang terinspirasi oleh hubungan biologis antar organisme pada sistem ekologi[1]. Algoritma I-SOS merupakan pengembangan dari algoritma sebelumnya yaitu Algoritma *Symbiotic Organism Search (SOS)* yang merupakan salah satu metode metaheuristik terbaru yang terinspirasi dari perilaku interaksi yang terlihat antara organisme di alam semesta[3]. Algoritma ini berbeda dari algoritma metaheuristik berbasis *evolutionary*, I-SOS tidak memproduksi atau menciptakan keturunan, namun layaknya algoritma yang berbasis populasi I-SOS menciptakan populasi awal yang disebut sebagai ekosistem dan akan melalui berbagai operator pencarian secara iteratif akan memodifikasi populasi untuk menghasilkan variabel solusi yang disebut dengan organisme yang optimal[5]. berikut merupakan aturan-aturan yang mengidentifikasi bahwa permasalahan tersebut adalah TSP:

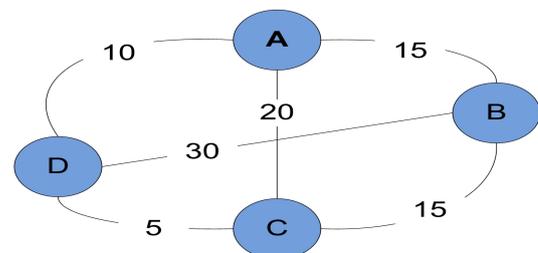
1. Perjalanan dimulai dan diakhiri di kota yang sama sebagai kota asal keberangkatan.
2. Seluruh kota harus dikunjungi tanpa satu kota terlewatkan
3. Salesman tidak boleh kembali ke kota asal sebelum seluruh kota terkunjungi.

III. METODE PENELITIAN

Metode *Improved Symbiotic Organism Search (I-SOS)* merupakan hasil modifikasi dari

metode SOS berupa penambahan sebuah fase baru yaitu fase predasi. Modifikasi algoritma ini dilakukan oleh (Sukanta, dkk. 2016) dalam penelitian yang berjudul "*Improved Symbiotic Organisms Search (I-SOS) Algorithm For Solving Unconstrained Function Optimization*" dengan tujuan untuk meningkatkan kemampuan pencarian dari metode SOS dengan teknik *Random Weighted Reflection Vektor (RWRV)*. Algoritma I-SOS bekerja dengan mensimulasikan perilaku interaksi antara dua organisme dalam ekosistem. Hubungan simbiosis yang terjadi dalam ekosistem dibagi menjadi empat fase yaitu *fase mutualisme, comensalisme, fase parasitisme* dan *predasi*. Calon solusi dari masalah merupakan organisme dalam ekosistem dimana nilai fitness dari setiap organisme mencerminkan tingkat adaptasi terhadap tujuan yang diinginkan. Algoritma I-SOS bekerja dengan mensimulasikan perilaku interaksi antara dua organisme dalam ekosistem. Hubungan simbiosis yang terjadi dalam ekosistem dibagi menjadi empat fase yaitu *fase mutualisme, comensalisme, fase parasitisme* dan *predasi*. Calon solusi dari masalah merupakan organisme dalam ekosistem dimana nilai fitness dari setiap organisme mencerminkan tingkat adaptasi terhadap tujuan yang diinginkan.

Berikut ini merupakan contoh implementasi algoritma I-SOS pada kasus TSP dengan 4 kota tujuan. Pada contoh ini, kota keberangkatan atau kota asal tidak ditentukan terlebih dahulu, dalam sekali perjalanan atau tour tidak boleh mengunjungi setiap kota lebih dari satu kali. Tujuan dari penerapan algoritma I-SOS pada kasus ini adalah untuk menemukan jarak tempuh paling minimum yang akan dilalui oleh seorang salesman. Peta kota dari rute perjalanan yang dimaksud pada penelitian ini seperti yang terdapat pada gambar 1 sebagai berikut.



Gambar 1 : Peta Perjalanan Salesman

Operator pada algoritma I-SOS terdiri dari ke empat fase yaitu fase mutualisme, fase komensalisme, fase parasitisme dan, fase predasi. Penjelasan dari ke empat fase tersebut sebagai

berikut:

$$X_{jnew} = X_j + RWRV * (X_{best-Mutual_Vector} * BF2) \quad (7)$$

1. Fase Mutualisme

Pada fase ini, organisme ke-i di dalam ekosistem, (x_i) akan berinteraksi secara mutualisme dengan organisme lain yang dipilih secara acak (x_j) dengan tujuan untuk meningkatkan kualitas hidup masing-masing di ekosistem. Selanjutnya, dua variabel solusi, x_{i_new} dan x_{j_new} akan diciptakan melalui hasil modifikasi setelah simbiosis mutualisme dilakukan dengan operasi matematis yang juga melibatkan X_{best} (organisme dengan nilai fitness terbaik dalam ekosistem). apabila nilai fitness dari kandidat solusi, x_{i_new} dan x_{j_new} hasil modifikasi X_i dan X_j lebih baik dari nilai fitness organisme sebelumnya maka seluruh nilai pada organisme X_i dan X_j akan diperbarui dengan nilai yang terdapat pada organisme x_{i_new} dan x_{j_new} .

- **RWRV (Random Weighting Reflection Vektor)** merupakan nomor acak yang terdistribusi secara seragam dalam kisaran $[0,1]$ untuk meningkatkan kemampuan pencarian dari algoritma i-sos dengan persamaan berikut.

$$WRV = 1 - 0.5 * (rand(1: d)) \quad (1)$$

- **Benefit Vektor (BF)** Nilai benefit vektor merupakan tingkat keuntungan yang diperoleh masing-masing organisme, apakah suatu organisme diuntungkan sebagian atau sepenuhnya dari interaksi. Nilai benefit vektor terdiri dari angka random antara 1 dan 2 sebagai berikut.

$$BF1 = rand(1:2) \quad (2)$$

$$BF1 = rand(1:2) \quad (3)$$

- **Mutual Vektor** menandakan karakteristik hubungan antara organisme X_i dan X_j serta X_{best} mewakili suatu organisme dengan nilai fungsi objektif terbaik dalam ekosistem.

$$mutual_vektor = (x_i + x_j) / 2 \quad (4)$$

$$BF1 = rand(1:2) \quad (5)$$

sedangkan pada persamaan 6 dan 7 merupakan fungsi yang digunakan untuk membentuk organisme X_{i_new} dan X_{j_new} .

$$X_{i_new} = X_i + RWRV * (X_{best-Mutual_Vector} * BF1) \quad (6)$$

2. Fase Komensalisme

Pada fase ini, organisme X_i akan berinteraksi secara komensalisme dengan organisme lain yang dipilih secara acak X_j . pada fase komensalisme, organisme X_i mendapatkan keuntungan dari interaksi dengan X_j namun organisme X_j sendiri tidak diuntungkan maupun dirugikan. Variabel solusi X_{i_new} diciptakan melalui hasil modifikasi setelah simbiosis komensalisme dilakukan dengan persamaan 8 sebagai berikut:

$$X_{i_new} = X_i + RWRV * (X_{best} - X_j) \quad (8)$$

3. Fase Parasitisme

Fase parasit dimulai dengan organisme X_i yang memproduksi parasit buatan bernama **parasit_vektor**. Variabel solusi **parasit_vektor** tersusun dari hasil perpaduan antara kloning terhadap organisme X_i menggunakan persamaan 9. Evaluasi terhadap nilai fitness akan dilakukan terhadap **parasit_vektor** dan organisme X_j . **parasit_vektor** akan menggantikan posisi organisme X_j dalam ekosistem hanya jika nilai fitnessnya lebih baik, Sebaliknya X_j akan bertahan dari **parasit_vektor** bila nilai fitnessnya lebih baik dari **parasit_vektor**.

$$Parasit_vektor = \begin{cases} X_{i,krk} & \text{if } rand(0,1) < k, k=1,2,3, \dots, nvar \\ arandompositionlainnya & \text{otherwise} \end{cases} \quad (9)$$

Dimana $nvar$ pada persamaan 2.8 merupakan jumlah elemen dalam variabel solusi, $x_{i,k}$ menunjukkan nilai variabel dari elemen organisme ke-i dalam ekosistem. Variabel solusi **parasit_vektor** tersusun dari hasil perpaduan antara kloning terhadap organisme x_i dan variabel acak. Organisme x_j yang berfungsi sebagai inang, dipilih secara acak dari ekosistem. Evaluasi terhadap nilai fitness akan dilakukan terhadap **parasit_vektor** dan organisme x_j .

4. Fase Predasi

Pada fase predasi merupakan upaya untuk memperbaiki organisme dengan nilai fitness terburuk dalam ekosistem. Organisme dengan nilai fitness terkecil akan digantikan oleh predasi vektor. Organisme predasi terbentuk berdasarkan

persamaan 10 berikut.

$$i = rand(0:1) * (Xi^{max} - Xi^{min}) \tag{10}$$

Dimana :

- X_i : Nilai sel ke-i dari organisme x_i
- Rand (0:1) : Nilai random antara 0 hingga 1
- X_{imin} : Nilai sel terkecil dari organisme x_i
- X_{imax} : Nilai sel terbesar dari organisme x_i

IV. HASIL DAN PEMBAHASAN

Berdasarkan peta rute perjalanan yang terdapat pada gambar 1 dapat ditentukan jarak antar kota seperti yang terdapat pada matrix jarak pada table 1.

Table 1 Matrix Jarak Antar Kota

	Kota A	Kota B	Kota C	Kota D
Kota A	0	15	20	10
Kota B	15	0	15	30
Kota C	20	15	0	5
Kota D	10	30	5	0

Tahapan penyelesaian permasalahan TSP menggunakan algoritma I-SOS pada penelitian diawali dengan proses identifikasi setiap kota yang terdapat pada gambar 3.1 dengan cara memberikan nilai random untuk mewakili setiap kota tujuan. Nilai *random* yang digunakan antara 0 sampai 1, hasil *random* yang diperoleh kemudian akan *sorting* dengan ketentuan nilai random terkecil akan dijadikan sebagai inisial terhadap kota A sedangkan nilai random terbesar dijadikan sebagai inisial terhadap kota D, sedangkan jarak antar kota yang terdapat pada tabel 1 akan digunakan pada tahap identifikasi nilai fitness organisme dalam ekosistem.

Dari Tabel 1 diatas, dapat ditentukan sejumlah kota yang akan dikunjungi yaitu kota A, kota B, kota C dan, kota D. dari ke empat kota tersebut representasi solusi untuk setiap rute akan dibuat sebanyak empat organisme. Panjang organisme adalah sebanyak kota yang akan dikunjungi, angka dalam setiap sel pada tiap-tiap organisme merupakan inisial dari masing-masing kota yang direpresentasikan dengan angka random. Pada random pertama yang dilakukan dapat diperoleh

nilai random | 0.5 | 0.27 | 0.75 | 0.25 | representasi kota dari angka random tersebut adalah | C | B | D | A | atau jika nilai tersebut di urutkan maka akan diperoleh urutan kota yaitu A=0.25, B=0.27, C=0.5 dan D=0.75, proses ini akan terus dijalankan hingga semua organisme yang ditentukan dalam ekosistem terbentuk.

Hasil dari proses inialisasi kota yang dilakukan dengan angka random tersebut sebagai berikut:

Tabel 2 Inialisasi Kota dengan Nilai Random

1	Random 1	0.25	0.27	0.5	0.75
	Urutan Kota	A	B	C	D
2	Random 2	0.80	0.50	0.75	0.3
	Urutan Kota	D	B	C	A
3	Random 3	0.45	0.25	0.60	0.20
	Urutan Kota	C	B	D	A
4	Random 4	0.90	0.25	0.60	0.80
	Urutan Kota	D	A	B	C

Dengan adanya nilai random yang diinputkan, maka dapat ditentukan identitas dari masing-masing kota pada tiap-tiap random yang dilakukan. Angka random tersebut merupakan representasi kota yang akan mengisi setiap sel organisme sesuai dengan ketentuan urutan nilai random yang digunakan.

A. Perhitungan Nilai Fitness

Nilai fitness yang digunakan pada penelitian ini merupakan inversi dari total jarak yang terdapat pada masing-masing organisme. Nilai fitness yang ditentukan terdiri dari range 0 hingga 1. Semakin mendekati 1 maka nilai yang ditampilkan akan semakin baik, sebaliknya apabila nilai fitness yang diperoleh semakin mendekati angka 0 maka nilai yang ditampilkan semakin buruk. Penentuan nilai fitness organisme yang dilakukan pada penelitian ini berdasarkan fungsi sebagai berikut:

$$f(x) = \frac{1}{\sum_{i=0}^n} \tag{10}$$

Ket:

x = Organisme ke-x

n = Jarak antar kota ke-i

Berikut ini merupakan perhitungan nilai fitness untuk ke empat organisme dalam ekosistem:

a. Perhitungan Fitness Organisme ke-1

$$\begin{aligned}\text{Organisme -1} &= A - B - C - D \\ \text{Fitnes(1)} &= 1 / (A-B + BC + CD + DA) \\ &= 1 / (15 + 15 + 5 + 10) \\ &= 1/45 \\ &= 0.0222222\end{aligned}$$

b. Perhitungan Fitnes Organisme ke-2

$$\begin{aligned}\text{Organisme -2} &= D - B - C - A \\ \text{Fitnes(2)} &= 1 / (DB + BC + CA + AD) \\ &= 1 / (30 + 15 + 20 + 10) \\ &= 1/75 \\ &= 0.0133333\end{aligned}$$

c. Perhitungan Fitnes Organisme ke-3

$$\begin{aligned}\text{Organisme -3} &= C - D - B - A \\ \text{Fitnes(3)} &= 1 / (CD + DB + BA + AC) \\ &= 1 / (5 + 30 + 15 + 20) \\ &= 1/70 \\ &= 0.0142857\end{aligned}$$

d. Perhitungan Fitnes Organisme ke-4

$$\begin{aligned}\text{Organisme -4} &= D - A - C - B \\ \text{Fitnes(4)} &= 1 / (DA + AC + CB + BA) \\ &= 1 / (10 + 20 + 15 + 15) \\ &= 1/75 \\ &= 0.0133333\end{aligned}$$

Setelah diperoleh nilai fitnes untuk masing-masing organisme, langkah selanjutnya akan dilakukan identifikasi terhadap nilai fitnes organisme untuk mencari organisme terbaik yang akan dijadikan sebagai x_{best} .

Dari hasil perhitungan nilai fitness dapat ditentukan organisme yang akan dijadikan sebagai x_{best} terdapat pada organisme ke-1 dengan nilai fitnes terbaik yaitu 0.0222222, untuk itu dapat ditentukan $x_{best} = x_1$ (organisme ke-1).

B. Tahapan Penyelesaian TSP pada algoritma I-SOS

Pencarian solusi optimal pada algoritma I-SOS, solusi akan diperoleh setelah melalui perhitungan fase mutualisme, fase komensalisme, fase parasitisme dan fase predasi dijalankan. jumlah organisme yang ditentukan dalam ekosistem = 4 dan, jumlah iterasi yang dilakukan = 2 iterasi. Berikut ini merupakan tahapan perhitungan fase-fase yang ada pada algoritma

I-SOS.

a. Fase Mutualisme Iterasi-I

Pemilihan organisme X_i dalam ekosistem, organisme yang terpilih adalah organisme ke-1, kemudian akan dilakukan pemilihan organisme secara random selain dari organisme ke-1 yang akan di inialisasi sebagai (X_j). Dari hasil pemilihan organisme yang dilakukan diperoleh organisme ke-3 sebagai (X_j) sehingga dapat dilakukan perhitungan pada fase mutualisme antara X_i dan X_j dengan tahapan sebagai berikut:

$$X_i = (x_1) : 0.25 | 0.27 | 0.5 | 0.75$$

$$X_j = (x_3) : 0.25 | 0.30 | 0.45 | 0.60$$

$$\text{Mutual_vektor} = 0.25 | 0.285 | 0.475 | 0.675 |$$

$$\text{Benefit_vektor} = \text{Random} (1:2).$$

$$\text{RWRV} = 1 - 0.5 * (1 + \text{rand} (0:D))$$

$$1. \quad X_{1(\text{new})} = x_1 + \text{RWRV} * (X_{\text{best}} - \text{Mutual_vektor} * \text{BF1})$$

$$\begin{aligned}\text{Cell_1} &= 0.25 + 0.08534 * (0.25 - 0.25 * 1) \\ &= 0.24206\end{aligned}$$

$$\begin{aligned}\text{Cell_2} &= 0.27 + 0.03481 * (0.27 - 0.285 * 2) \\ &= 0.08530\end{aligned}$$

$$\begin{aligned}\text{Cell_3} &= 0.5 + 0.33716 * (0.50 - 0.475 * 1) \\ &= 0.508429\end{aligned}$$

$$\begin{aligned}\text{Cell_4} &= 0.75 + 0.34097 * (0.75 - 0.675 * 2) \\ &= 0.86251\end{aligned}$$

Dari hasil perhitungan tersebut akan muncul organisme baru yang akan diujikan sebagai $X_{i(\text{new})}$ sebagai berikut:

$$X_{i(\text{new})} = 0.24206 | 0.08530 | 0.508429 | 0.86251$$

$$\begin{aligned}\text{Fitnes } X_{i(\text{new})} &= 1 / (\text{BA} + \text{AC} + \text{CD} + \text{DB}) \\ &= 1 / (15 + 20 + 5 + 30) \\ &= 1/70 \\ &= 0.014286\end{aligned}$$

$$2. \quad X_{3(\text{new})} = x_3 + \text{RWRV} * (X_{\text{best}} - \text{Mutual_vektor} * \text{BF2})$$

$$\begin{aligned}\text{Cell_1} &= 0.25 + 0.88227 * (0.25 - 0.25 * 2) \\ &= 0.20782\end{aligned}$$

$$\begin{aligned}\text{Cell_2} &= 0.30 + 0.22889 * (0.27 - 0.285 * 1) \\ &= 0.23985\end{aligned}$$

$$\begin{aligned}\text{Cell_3} &= 0.45 + 0.06699 * (0.50 - 0.475 * 1) \\ &= 0.43809\end{aligned}$$

$$\begin{aligned}\text{Cell_4} &= 0.60 + 0.48234 * (0.75 - 0.675 * 2) \\ &= -0.04044\end{aligned}$$

Dari hasil perhitungan tersebut akan muncul organisme baru yang akan diujikan sebagai $X_{j(\text{new})}$ sebagai berikut:

$$X_{j(\text{new})} = 0.20782 | 0.23985 | 0.43809 | -0.04044$$

$$\text{Fitnes } X_{j(\text{new})} = 1 / (\text{BC} + \text{CD} + \text{DA} + \text{AB})$$

$$= 1/(15 + 5 + 10 + 15)$$

$$= 1/45$$

$$= 0.022222$$

Setelah diperoleh nilai fitness untuk $X_i(\text{new})$ dan $X_j(\text{new})$, selanjutnya akan dilakukan perbandingan nilai fitness antara X_i dan $X_i(\text{new})$ serta X_j dan $X_j(\text{new})$. Karena nilai fitness $X_i(\text{new})$ tidak lebih baik dari nilai fitness X_i maka nilai fitness $X_i(x1)$ tetap dipertahankan, sedangkan nilai fitness $X_j(\text{new})$ lebih baik dari nilai fitness X_j maka seluruh nilai pada organisme $X_j(x3)$ akan digantikan oleh organisme $X_j(\text{new})$.

b. Fase Komensalise Iterasi-I

Pada fase ini, akan dilakukan pemilihan organisme dalam ekosistem selain dari organisme yang sedang diproses yaitu organisme ke-1 (X_i). Hasil pemilihan organisme yang dilakukan diperoleh organisme ke-2 sebagai X_j . Organisme X_i akan dipengaruhi oleh organisme X_j dengan perhitungan sebagai berikut:

$$X_i(x1) = 0.25|0.27|0.5|0.75$$

$$X_j(x2) = 0.3|0.50|0.75|0.80$$

Tahap selanjutnya dilakukan modifikasi terhadap $X_i(x1)$ yang akan dipengaruhi oleh $X_j(x2)$.

$$X_i(x1) = 0.25|0.27|0.5|0.75$$

$$= X_i + RWRV * (X_{\text{best}} - X_j)$$

$$\text{Cell}_1 = 0.25 + RWRV * (0.25 - 0.3)$$

$$= 0.52778$$

$$\text{Cell}_2 = 0.27 + RWRV * (0.27 - 0.50)$$

$$= -0.17778$$

$$\text{Cell}_3 = 0.5 + RWRV * (0.50 - 0.75)$$

$$= 0.99778$$

$$\text{Cell}_4 = 0.75 + RWRV * (0.75 - 0.80)$$

$$= 1.52778$$

Dari hasil perhitungan yang dilakukan akan muncul organisme baru untuk organisme yang diujikan, selanjutnya dilakukan perbandingan nilai fitness antara organisme $X_i(\text{new})$ dan X_j .

$$X_i(\text{new}) = 0.52778 | -0.17778 | 0.99778 | 1.52778$$

$$\text{Fitnes}_{S_{X_i(\text{new})}} = 1/(BA + AC + CD + DB)$$

$$= 1/(15 + 20 + 5 + 30)$$

$$= 1/70$$

$$= 0.014286$$

Setelah dilakukan perhitungan nilai fitness untuk organisme $X_i(\text{new})$, diperoleh nilai fitness lebih kecil dari nilai fitness organisme sebelumnya (X_i), maka tidak terjadi perubahan nilai terhadap

organisme X_i dan nilai $X_i(x1)$ tetap dipertahankan dalam ekosistem.

c. Fase Parasit Iterasi-I

Pada fase ini akan dilakukan duplikat terhadap organisme ke-1 (x_i) hasil duplikat tersebut selanjutnya dilakukan modifikasi sel organismenya yang akan dijadikan sebagai parasit_vektor. Modifikasi organisme X_i sebagai parasit vektor dilakukan sebagai berikut:

$$X_i(x1) = 0.25 | 0.27 | 0.5 | 0.75$$

$$\text{Parasit_vektor} = 0.5 | 0.75 | 0.25 | 0.27$$

$$\text{Fitnes}_{(\text{parasit_vektor})} = 1/(BD + DA + AC + CB)$$

$$= 1/(30 + 10 + 20 + 15)$$

$$= 1/75$$

$$= 0.01333333$$

Selanjutnya Parasit_vektor akan dibandingkan dengan organisme X_j yang di pilih secara acaka dalam ekosistem. Organisme yang dipilih selain dari organisme yang sedang diproses. Misalkan organisme yang terpilih adalah organisme ke-4.

$$X_j(x4) = 0.90 | 0.25 | 0.80 | 0.60$$

$$= 1/(DA + AC + CB + BD)$$

$$= 1/(10 + 20 + 15 + 30)$$

$$= 1/75$$

$$= 0.01333333$$

Setelah dilakukan perhitungan nilai fitness untuk kedua organisme (parasit_vektor dan X_j) didapatkan nilai fitness yang sama untuk kedua organisme maka parasit_vektor tidak bisa menggantikan nilai yang terdapat pada organisme X_j sehinga nilai $x_j(x4)$ tetap dipertahankan dalam ekosistem.

d. Fase Predasai Iterasi-I

Fase predasi merupakan upaya untuk memperbaiki organisme dengan nilai fitness terburuk dalam ekosistem dengan organisme predasi_vektor. Predasion_vektor dibentuk sebagai berikut.

$$\text{Predasi} = x_i + \text{rand}(0:1) * (X_{i_{\text{max}}} - X_{i_{\text{min}}})$$

$$\text{Cell}_1 = 0.25 + 0.5790 * (0.75 - 0.25)$$

$$= 0.53950$$

$$\text{Cell}_2 = 0.27 + 0.2543 * (0.75 - 0.25)$$

$$= 0.39716$$

$$\text{Cell}_3 = 0.5 + 0.44043 * (0.75 - 0.25)$$

$$= 0.72015$$

$$\text{Cell}_4 = 0.75 + 0.30562 * (0.75 - 0.25)$$

$$= 0.90281$$

Hasil dari perhitungan tersebut dapat diperoleh organisme predasi sebagai berikut

$$\text{Predasi} = 0.53950 | 0.39716 | 0.72015 | 0.90281$$

$$\begin{aligned} \text{Fitnes}_{\text{predasi}} &= 1/(\text{BA} + \text{AC} + \text{CD} + \text{DB}) \\ &= 1/(15 + 20 + 10 + 30) \\ &= 1/75 \\ &= 0.013333 \end{aligned}$$

C. Hasil Perhitungan Organisme Pada Iterasi-I

Dari hasil identifikas nilai fitnes organisme pada fase predasi Iterasi-I dapat ditampilkan susunan organisme dalam ekosistem seperti yang terdapat pada tabel 3 berikut.

Tabel 3 Representasi Organisme Iterasi-I

Organisme	Sel Organisme	Rute	Nilai Fitnes
ke-1	0.25 0.27 0.5 0.75	A-B-C-D	0.0222222
ke-2	0.80 0.50 0.75 0.3	D-B-C-A	0.0133333
ke-3	0.20782 0.23985 0.43809 -0.04044	B-C-D-A	0.0222222
ke-4	0.90 0.25 0.60 0.80	D-A-C-B	0.0133333

D. Kondisi Berhenti Algoritma I-SOS pada kasus TSP

Kondisi berhenti atau stoping criteria yang digunakan dalam penyelesaian kasus TSP menggunakan algoritma I-SOS pada penelitian ini yaitu berdasarkan nilai fitnes atara organisme sama atau perhitungan semua organisme sudah mencapai perhitungan pada iterasi yang terakhir. Kedua kriteria tersebut digunakan sebagai batasan dalam pencarian solusi. Pencarian solusi pada algoritma I-SOS akan dihentikan jika salah satu dari kedua kriteria tersebut terpenuhi.

Berikut ini merupakan hasil perhitungan seluruh oragnisme dalam ekosistem, representasi organisme terbaik yang diperoleh setelah melalui tahapan algoritma dapat diperoleh empat solusi rute terbaik yaitu dengan nilai fitness sebesar 0.022 seperti yang terdapat pada gambar 2 berikut.

X1			X2			X3			X4		
Cell	Kota	Jarak									
0.25	A	15	0.473	B	15	0.271	B	15	1.388	D	10
0.27	B	15	0.363	A	10	0.28	C	5	0.565	A	15
0.5	C	5	0.601	D	5	0.438	D	10	0.89	B	15
0.75	D	10	0.568	C	15	-0.4	A	15	0.893	C	5
Fitnes	0.022		Fitnes	0.022		Fitnes	0.022		Fitnes	0.022	

Gambar 2 Hasil Perhitungan Algoritma I-SOS Pada Kasus TSP

Berdasarkan dari gambar 2 dapat dijelaskan

Total jarak tempuh terpendek untuk ke empat solusi rute yang dihasilkan adalah sejauh 45km. Apabila kota keberangkatan tidak ditentukan maka, urutan kota yang dapat dilalui adalah sebagai berikut:

- Rute 1 : A – B – C – D (Organisme ke-1)
- Rute 2 : B – A – D – C (Organisme ke-2)
- Rute 3 : B – C – D – A (Organisme ke-3)
- Rute 4 : D – A – B – C (Organisme ke-4)

V. KESIMPULAN

Penyelesaian permasalahan TSP menggunakan algoritma I-SOS pada penelitian diawali dengan proses identifikasi setiap kota dengan cara memberikan nilai random untuk mewakili setiap kota tujuan. nilai random yang digunakan antara 0 sampai 1, hasil random yang diperoleh kemudian akan di sorting dengan ketentuan nilai random terkecil akan dijadikan sebagai inisial terhadap kota Pertama dimana dalam penelitian ini ditentukan sebagi kota A sedangkan nilai random terbesar dijadikan sebagai inisial terhadap terakhir yaitu kota D. dari hasil perhitungan seluruh fase pada algoritma I-SOS terhadap organisme yang terdapat dalam ekosistem dimana terdapat empat kota yang akan dikunjungi yaitu kota A-B-C-D diperoleh empat rute terbaik dengan total jarak tempuh sejauh 45km jika kota keberangkatan tidak ditentukan.

DAFTAR PUSTAKA

- [1] Dody Prayogo, dkk. Implementasi Metode Symbiotic Organism Search Dalam Penentuan Tata Letak Fasilitas Proyek Konstruksi Berdasarkan Jarak Tempuh Pekerja. *JTI*, Vol. 19, No. 2, hal. 103–114, Desember 2017.
- [2] Muhammad Isnaini Hidayatul Umam, dkk. *Modifikasi algoritma Symbiotic Organism Search untuk Traveling Salesman Problem*. Prosiding Seminar Nasional Manajemen Teknologi XXIV 2016.
- [3] Sukanta Nama, Dkk. 2016. *Improved Symbiotic Organisms Search (I-SOS) Algorithm For Solving Unconstrained Function Optimization*, *DSC, Vol. 5, No. 361 – 380, September 2016*.
- [4] T. W. Dian, “Algoritma Optimasi untuk penyelesaian Travelling Salesman Problem,” *Transformatika*, vol. 11, no. 1, hal. 1–6, Jul. 2013.
- [5] Zulkarnaen, “Algoritma Improved Symbiotic Organism Search (I-SOS) Sebagai Pendukung Keputusan dalam Penentuan Dosen Pengampu Matakuliah,” *JTIM*, vol. 1, no. 4, hal. 279-286, februari, 2020.