

**PENERAPAN *ARTIFICIAL FISH SWARM ALGORITHM*
(AFSA) PADA *MULTIPLE TRAVELLING SALESMAN*
*PROBLEMS (m-TSP)***
*(Implementation Artificial Fish Swarm Algorithm (AFSA) on Multiple Travelling
Salesman Problems (m-TSP))*

Florenzia Wahyu Ganda Fismaya, Abduh Riski, Ahmad Kamsyakawuni

Jurusan Matematika, Fakultas MIPA, Universitas Jember
Jl. Kalimantan 37 Jember 68121, Indonesia
gflorenciawahyu@gmail.com, {riski, kamsyakawuni}.fmipa@unej.ac.id

Abstract. Selling or trading in the industrial 4.0 era as it can now be done by opening a shop online. Therefore, shopping at this time can also be done online also, so that the online shop owners require orders that do not allow for Cash On Delivery (COD) transactions using package delivery services. This research discusses about finding a solution for good shipping with a minimum total mileage of several couriers at PT. Titipan Kilat District Banyuwangi uses AFSA as a settlement algorithm. The experimental process is carried out by using several parameter values to determine the parameters that affect the final solution. Each parameter will be tested with a maximum of 1000 iterations, then the best results will be tested again with a maximum iteration of 2000, and 5000 and will be compared with the original distance traveled by the couriers. The final solution offered in the form of a delivery route by three couriers with the total distance (Z) of the third courier is 87.28 Km with the smallest iteration value reaching the local minimum in iteration 1169.

Keywords: artificial fish swarm algorithm (afsa), multiple travelling salesman problem (m-tsp), route, total mileage.

MSC 2010: 90B06

1. Pendahuluan

Berjualan atau berdagang pada era industrial 4.0 seperti saat ini bisa dilakukan dengan cara membuka toko secara online [2]. Oleh karena itu, berbelanja saat ini juga dapat dilakukan secara *online*, sehingga menuntut pemilik toko *online* mengirimkan pesanan yang tidak memungkinkan untuk transaksi *Cash On Delivery* (COD) dengan menggunakan jasa pengiriman paket. *Travelling Salesman Problem* (TSP) merupakan masalah seorang kurir dalam mengantarkan barang untuk mencari rute perjalanan yang optimal. Sedangkan, jika suatu masalah yang melibatkan lebih dari satu kurir, maka permasalahan tersebut disebut dengan *Multiple Travelling Salesman Problem* (m-TSP). *Artificial Fish Swarm Algorithm* (AFSA) merupakan salah satu jenis algoritma optimisasi yang termasuk ke dalam *swarm intelligence algorithm*. AFSA merupakan algoritma yang

terinspirasi dari kecerdasan kolektif tingkah laku kumpulan ikan di alam seperti memangsa, bergerak bebas, berkumpul, dan mengikuti [4]. Penelitian kali ini penulis tertarik untuk menerapkan *Artificial Fish Swarm Algorithm* (AFSA) untuk menyelesaikan masalah *Multiple Travelling Salesman Problem* (m-TSP) dan mendapatkan rute pengantaran barang dengan jarak minimum.

Travelling Salesman Problem (TSP)

Travelling Salesman Problem (TSP) merupakan permasalahan perjalanan seorang *salesman* yang harus mengunjungi n tempat tujuan pengiriman dalam satu kali kunjungan, kemudian kembali lagi ke tempat asal [3]. Secara umum, pada saat ini penyedia jasa pengantaran barang atau dokumen yang besar tidak hanya memiliki satu *salesman* atau kurir untuk mengantarkan barang konsumen. *Multiple Travelling Salesman Problem* (m-TSP) merupakan sebuah masalah optimasi yang merupakan perluasan dari permasalahan TSP yang memiliki perbedaan dari jumlah *salesman* yang melakukan kunjungan. m-TSP juga merupakan aturan optimasi dalam penyusunan rute dari m *salesman* dan n kota [5], dimana setiap m *salesman* mengunjungi n kota dan setiap *salesman* memulai dan berakhir di titik yang sama. Dalam kasus ini, setiap kota harus dikunjungi setidaknya satu kali oleh satu *salesman* s_i , dengan i bernilai 1 sampai m . M-TSP ini memiliki dua jenis permasalahan stasiun pemberangkatan, seperti permasalahan stasiun tunggal dan *multiple* stasiun. M-TSP juga dapat diklasifikasikan menjadi simetrik dan asimetrik berdasarkan *cost* atau jarak antar titik pengantaran.

Artificial Fish Swarm Algorithm (AFSA)

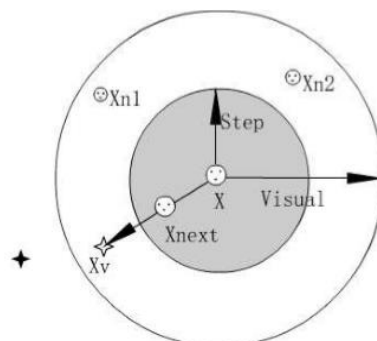
Artificial Fish Swarm Algorithm (AFSA) merupakan salah satu algoritma metaheuristik yang berbasis *swarm intelligence* dan mengembangkan teknik optimasi yang diusulkan oleh Li Xiao-lei pada tahun 2002 [1]. Konsep AFSA pada umumnya adalah meniru perilaku, seperti memangsa, bergerombol atau berkoloni, berburu, dan lain-lain dari kelompok atau kumpulan ikan di alam. Dalam kehidupan bawah air, ikan dapat menemukan area yang memiliki makanan lebih, dimana untuk mendapatkan makanan tersebut dapat dilakukan oleh seekor maupun sekelompok ikan. Pada saat seekor ikan muncul untuk makan, maka ada daya tarik bagi ikan-ikan lain, kemudian dengan cepat seluruh kelompok ikan berkumpul dan mulai untuk makan. Pada dasarnya koloni atau populasi dinyatakan dengan N ikan buatan. Posisi ikan buatan dapat dinyatakan sebagai berikut:

$$X = (X_1, \dots, X_n) \text{ untuk } i = 1, \dots, n \quad (1)$$

dimana i menyatakan posisi dari populasi ikan. Jarak antara dua ikan buatan pada posisi X_i dan X_j adalah $Dis_{ij} = \|X_i - X_j\|$ (*Euclidean Distance*).

Model ikan buatan adalah beberapa bagian fungsi dan variabel atau parameter. Variabel atau parameternya meliputi X (posisi ikan buatan), *step* (panjang langkah maksimum), *visual* (area jarak pandang), *try_number* (maksimum pengulangan perhitungan), *bulletin*, dan *crowd factor* δ ($0 < \delta < 1$). Fungsi juga terdiri dari perilaku memangsa, perilaku

berkoloni, dan perilaku mengikuti. Setiap *step* dari proses optimasi, ikan buatan mencari tempat dengan nilai *fitness* terbaik, dalam masalah mencari tempat yang disebutkan oleh tiga perilaku dasar dari prosedur algoritma ini [1].



Gambar 1. Beberapa ikan buatan dan lingkungannya

Perilaku-perilaku dasar dari *artificial fish* atau ikan buatan antara lain:

1. Perilaku Memangsa

Jika X_i merupakan posisi ikan buatan i , Posisi X_j dapat dihitung dengan persamaan (2).

$$X_{j,d}(t+1) = X_{i,d}(t) + visual \times Rand_d(-1,1) \quad (2)$$

Setelah itu, kepadatan makanan di X_i dibandingkan dengan posisi awal, jika $f(X_i) \geq f(X_j)$, ikan buatan bergerak satu *step* dari posisi terakhir ke X_j , dijelaskan oleh persamaan (3).

$$\vec{X}_i(t+1) = \vec{X}_i(t) + Rand(0,1) \times step \times \frac{\vec{X}_j - \vec{X}_i(t)}{Dis_{i,j}} \quad (3)$$

2. Perilaku Berkumpul atau Berkoloni

Salah satu dari anggota sekelompok ikan, selalu mencoba untuk bergerak mengelilingi anggota ikan lainnya Posisi tengah dari kelompok, seperti persamaan (4).

$$X_{center,d} = \frac{1}{N} \sum_{i=1}^n X_{i,d} \quad (4)$$

Jika $f(X_{center}) \leq f(X_i)$ dan $\delta > (nc/N)$, posisi tengah memiliki konsistensi makanan yang lebih baik dibandingkan posisi saat itu dan kepadatan populasi di tetangga tidak banyak, jadi ikan buatan i bergerak ke posisi tengah dengan persamaan (5).

$$\vec{X}_{i(t+1)} = \vec{X}_i(t) + \frac{\vec{X}_{center} - \vec{X}_i(t)}{Dis_{i,center}} \times step \times Rand(0,1) \quad (5)$$

Jika $n_c = 0$ atau kondisi saat bergerak ke posisi tengah tidak memuaskan, perilaku memangsa sama dengan ikan buatan i .

3. Perilaku Mengikuti

Saat proses dari kelompok ikan bergerak, ketika seekor atau beberapa ekor ikan menemukan makanan, ikan-ikan tetangga akan ikut untuk mengambil makanan. Jika X_i posisi dari ikan buatan i , ikan tersebut akan memeriksa tetangga X_{best} , jika nn adalah jumlah dari ikan buatan di area *visual* dari ikan buatan $best$ atau terbaik, dan jika $f(X_{best}) \leq f(X_i)$ dan $\delta > (nn/N)$, maka posisi X_{best} memiliki konsistensi makanan yang lebih baik dibandingkan posisi dari ikan buatan i dan kepadatan populasi di tetangga nya tidak banyak, oleh karena itu, ikan buatan i bergerak sebanyak *step* dengan persamaan (6).

$$n\vec{X}_{i(t+1)} = \vec{X}_i(t) + \frac{\vec{X}_{best} - \vec{X}_i(t)}{Dis_{i,best}} \times step \times Rand(0,1) \quad (6)$$

Jika ikan buatan i tidak memiliki tetangga atau tidak ada tetangga yang mengikutinya, perilaku memangsa sama dengan ikan buatan i .

2. Metodologi

Data penelitian yang digunakan dalam penelitian ini merupakan data primer berupa 60 alamat konsumen pengguna jasa pengiriman paket dari PT. Titipan Kilat (TIKI) Kantor Cabang Banyuwangi Kota yang diolah menjadi matriks 61×61 berisi jarak antar titik stasiun dan alamat konsumen. Langkah-langkah penelitian yang dilakukan adalah menerapkan AFSA pada data yang telah diidentifikasi dengan langkah perhitungan pada perilaku memangsa, berkumpul atau berkoloni, dan mengikuti untuk masing-masing ikan yang ditentukan di awal. Menjalankan program yang telah dibuat menggunakan *software* MATLAB dengan perubahan beberapa parameter dan iterasi kemudian membandingkan hasil perhitungan AFSA dengan total jarak tempuh asli dari tiga kurir. Sehingga didapatkan solusi akhir berupa rute perjalanan dari masing-masing kurir dengan total jarak tempuh terpendek

3. Hasil dan Pembahasan

Proses percobaan dilakukan dengan menggunakan beberapa nilai parameter untuk mengetahui parameter yang berpengaruh pada solusi akhir. Nilai parameter N yang digunakan yaitu 10, 20, 30, 40, dan 50. Nilai parameter δ , *visual*, dan *Step* yang digunakan berurut-urut 0,5, 5, dan 0,01. Serta nilai parameter *TryNumber* yang digunakan 5, 10, 15, 20, dan 25. Setiap parameter akan diuji dengan maksimal iterasi 1000, kemudian hasil terbaik akan diuji kembali dengan maksimal iterasi 2000, dan 5000. Perhitungan tersebut dilakukan menggunakan program yang dibuat dengan *software* MATLAB R2015b dan dijalankan pada Komputer dengan CPU Intel Core i7, RAM 8GB dan 64-bit OS. Hasil percobaan yang telah dilakukan sebagai berikut:

- a. Hasil percobaan dengan beberapa nilai parameter n pada 1000 iterasi.
 Hasil percobaan beberapa nilai parameter populasi ikan (N) dilakukan pada maksimal iterasi 1000 ini akan diambil nilai rata-rata total jarak (Z) terpendek untuk dilanjutkan dengan percobaan selanjutnya dapat dilihat pada Tabel 1.

Tabel 1. Hasil percobaan populasi

No.	Jumlah N	Rata-rata total jarak Z (Km)	Rata-rata nilai iterasi terkecil mencapai minimum lokal (t)	Rata-rata <i>Running Time</i> (s)
1.	10	114,88	592	15,9962
2.	20	113,61	688	32,5901
3.	30	110,38	791	31,5435
4.	40	108,62	763	33,0249
5.	50	107,92	690	49,3145

- b. Hasil percobaan dengan beberapa nilai parameter *try_number* pada 1000 iterasi.
 Setelah melakukan percobaan dengan beberapa parameter N diatas, didapatkan total jarak (Z) terpendek dengan populasi 50. Kemudian, populasi tersebut digunakan untuk uji pada percobaan dengan beberapa parameter *Try Number* dengan maksimal iterasi yang digunakan sebanyak 1000 dan dilakukan sebanyak 10 kali *running* seperti pada Tabel 2.

Tabel 2. Hasil percobaan *try number*

No.	Jumlah <i>Try Number</i>	Rata-rata total jarak Z (Km)	Rata-rata nilai iterasi terkecil mencapai minimum lokal (t)	Rata-rata <i>Running Time</i> (s)
1.	5	107,92	690	49,3145
2.	10	104,73	738	63,1756
3.	15	103,22	689	63,8845
4.	20	100,89	876	83,0682
5.	25	99,48	718	101,1175

- c. Hasil percobaan dengan 2000 iterasi.
 Setelah percobaan dengan iterasi 1000, dilanjutkan percobaan dengan maksimum iterasi 2000 dilakukan sebanyak 10 kali *running* program dengan hasil pada Tabel 3.

Tabel 3. Hasil percobaan pada iterasi 2000

No.	Total jarak Z (Km)	Nilai iterasi terkecil mencapai minimum lokal (t)	<i>Running Time</i> (s)
1.	87,28	1169	188,3030
2.	88,270	1641	187,3476
3.	94,510	1694	187,3013
4.	101,65	1579	188,1293
5.	101,242	1968	189,1425
6.	96,330	926	189,1778
7.	98,450	1341	189,1773
8.	96,210	1836	189,3020
9.	95,200	1857	189,4593
10.	104,190	533	189,6436

Hasil percobaan 10 kali *running* dengan parameter yang sama pada maksimum iterasi 2000 dapat dihitung rata-rata total jarak (Z) yang semakin pendek dari percobaan sebelumnya yaitu 96,333 Km.

d. Hasil Percobaan dengan 5000 iterasi.

Percobaan selanjutnya dengan maksimal iterasi 5000 dilakukan sebanyak 10 kali seperti pada Tabel 4.

Tabel 4. Hasil percobaan pada iterasi 5000

No.	Total jarak Z (Km)	Nilai iterasi terkecil mencapai minimum lokal (t)	<i>Running Time</i> (s)
1.	100,04	4381	475,2457
2.	93,575	3888	471,2818
3.	92,83	4666	470,7493
4.	99,866	4349	471,2343
5.	94,85	4787	474,8148
6.	88,836	3868	477,5742
7.	94,51	3821	475,4375
8.	93,41	4078	470,9247
9.	92,96	4533	457,8044
10.	96,5	2041	480,2494

Hasil percobaan 10 kali *running* dengan parameter yang sama pada maksimum iterasi 5000 dapat dihitung rata-rata total jarak (Z) yang semakin pendek dari percobaan sebelumnya yaitu 94,7377 Km. Proses percobaan dalam penyelesaian M-TSP dengan menerapkan perhitungan dari AFSA dilakukan menggunakan data jarak antar 62 titik (konsumen dan stasiun pemberangkatan) yang akan menghasilkan total jarak terpendek yang akan ditempuh dan masing-masing rute pengantaran kurir, serta *Running Time* (waktu komputasi). Percobaan tersebut dilakukan menggunakan beberapa parameter, seperti parameter populasi (N), *Step*, *Try Number*, *visual*, faktor keramaian δ dengan batasan $0 < \delta < 1$, *Max Iter*, serta jumlah kurir. Nilai parameter N yang digunakan yaitu 10, 20, 30, 40, dan 50. Nilai parameter δ , *visual*, dan *Step* yang digunakan berurut-urut 0,5, 5, dan 0,01. Serta nilai parameter *Try Number* yang digunakan 5, 10, 15, 20, dan 25. Jumlah kurir yang digunakan sebanyak 3 kurir. Setiap parameter akan diuji dengan maksimal iterasi 1000, kemudian hasil total jarak (Z) terbaik akan diuji kembali dengan maksimal iterasi 2000, dan 5000. Hasil-hasil percobaan dapat dilihat pada tabel percobaan sebelumnya. Dilakukan uji coba parameter populasi (Z) dengan *Try Number* = 5 pada iterasi maksimal 1000, lalu diambil nilai rata-rata dari masing-masing populasi (N). Hasil pada Tabel 1 dapat dilihat bahwa, semakin banyak populasi (N) yang digunakan maka hasil yang didapatkan semakin optimal. Hal ini dikarenakan, semakin banyak populasi (N) yang digunakan, maka semakin banyak kandidat solusi yang ditawarkan dari masing-masing populasi ikan buatan tersebut. Hal tersebut juga menyebabkan *Running Time* pada program semakin lama, seperti halnya pada hasil percobaan pada tabel hasil. Hasil rata-rata total jarak (Z) yang paling optimal yaitu pada populasi ikan buatan (N) = 50 dengan rata-rata total jarak (Z) 107,92 Km.

Selanjutnya, hasil percobaan dengan populasi ikan buatan (N) = 50 akan diuji coba dengan perubahan nilai parameter *Try Number* pada iterasi maksimal 1000. Percobaan dilakukan 10 kali *running*, dan hasilnya seperti pada Tabel 4.2. seperti pada percobaan sebelumnya dapat dilihat bahwa semakin banyak nilai parameter *Try Number* yang digunakan, maka semakin kecil total jarak (Z) yang dihasilkan. Hal ini dikarenakan, parameter *Try Number* digunakan untuk jika hasil perhitungan pada suatu perilaku dasar tidak memuaskan, maka akan dihitung kembali perilaku memangsanya sampai hasilnya memuaskan dengan batasan parameter *Try Number*. Jadi, memungkinkan adanya perubahan atau *update* posisi ikan buatan dan kandidat solusi yang ditawarkan juga lebih baik dari sebelumnya. Hasil percobaan perubahan nilai parameter *Try Number* = 25 yang memiliki hasil paling optimal, dengan rata-rata total jarak (Z) = 99,48 Km.

Setelah itu, hasil dengan nilai parameter populasi ikan buatan (N) = 50 dan *Try Number* = 25 akan diuji kembali dengan maksimal iterasi 2000 dan 5000. Percobaan tersebut dilakukan sebanyak 10 *running* masing-masing iterasi maksimumnya. Dari hasil pada Tabel 2 dan Tabel 3 dapat dilihat bahwa semakin besar iterasi maksimum yang digunakan maka semakin kecil juga hasil rata-rata total jarak (Z) yang dihasilkan. Karena hasil pada iterasi 2000 lebih baik dibandingkan dengan hasil pada iterasi 5000. Oleh karena itu, hasil total jarak (Z) terpendek pada iterasi maksimum 2000 yang digunakan sebagai solusi akhir yang diberikan dengan total jarak (Z) yang akan ditempuh oleh kurir-kurir adalah 87,28 Km. Pembagian rute yang diberikan kepada masing-masing kurir yaitu:

Kurir 1 : 0-15-50-19-61-3-49-34-29-42-41-53-31-26-46-40-44-39-57-33-32-22-0

Kurir 2 : 0-23-5-14-18-54-11-7-4-60-55-21-51-12-9-59-17-16-8-2-13-0

Kurir 3 : 0-27-6-56-20-52-35-45-47-24-43-30-37-48-25-38-10-36-58-28-1-0

dengan masing-masing total jarak tempuhnya 29,13 Km untuk kurir 1, 30,1 Km untuk kurir 2, dan 28,05 Km untuk kurir 3. Sedangkan, rute asli pengantaran dari 3 kurir ini memiliki total jarak (Z) = 93,8 Km dengan masing-masing kurir menempuh 30,9 Km untuk kurir 1, 24,6 Km untuk kurir 2, dan 38,3 Km untuk kurir 3.

4. Kesimpulan

- a. Berdasarkan Hasil percobaan pada kombinasi parameter populasi (N) dan *Try Number*, jika semakin besar nilai parameter ikan buatan (N), *Try Number*, dan jumlah iterasi yang digunakan, maka AFSA mampu memberikan hasil rute dengan total jarak tempuhnya semakin kecil dan.
- b. Solusi rute untuk masing-masing kurir yaitu
Kurir 1: 0-15-50-19-61-3-49-34-29-42-41-53-31-26-46-40-44-39-57-33-32-22-0
Kurir 2: 0-23-5-14-18-54-11-7-4-60-55-21-51-12-9-59-17-16-8-2-13-0
Kurir 3: 0-27-6-56-20-52-35-45-47-24-43-30-37-48-25-38-10-36-58-28-1-0

Dengan total jarak (Z) dari ketiga kurir tersebut adalah 87,28 Km dengan nilai iterasi terkecil mencapai minimum lokal pada iterasi 1169.

Daftar Pustaka

- [1] Li, X. L, Shao, Z. J., dan Qian, J. X. (2002). An Optimizing Method Based On Autonomous Animates: Fish-Swarm Algorithm. *Systems Engineering-Theory & Practice*, 11, pp. 32-38.
- [2] Prasetyo, H dan Sutopo, W. (2018). Industri 4.0: Telaah Klasifikasi Aspek Dan Arah Perkembangan Riset. Surakarta. Universitas Sebelas Maret
- [3] Suprayogi, D. dan Mahmudy, W., (2015). Penerapan Algoritme Genetika Travelling Salesman Problem with Time Window:Studi Kasus Rute Antar Jemput Laundry. *Jurnal Buana Informatika*, Volume 6, pp. 121-130.
- [4] Yazdani, Danial, Toosi, Adel, N., Meybodi, dan Mohammad, R. (2010). “Fuzzy Adaptive Artificial Fish Swarm Algorithm”. Qazvin, Mashhad, Tehran, Iran: IAU of Qazvin, IAU of Mashhad and Amirkabir University of Technology
- [5] Zhou, W. dan Li, Y. (2010). An Improved Genetic Algorithm for Multiple Traveling Salesman Problem. pp.493– 495.