

A Hybrid Symbiotic Organisms Search Algorithm with Variable Neighbourhood Search for Solving Symmetric and Asymmetric Traveling Salesman Problem

M I H Umam^{1,*} and B Santosa¹

¹Departement of Industrial Engineering, Institut Teknologi Sepuluh Nopember
Kampus ITS Sukolilo-Surabaya 60111, Indonesia

*ISNAINI.30@gmail.com

Abstract. Combinatorial optimization has been frequently used to solve both problems in science, engineering, and commercial applications. One combinatorial problems in the field of transportation is to find a shortest travel route that can be taken from the initial point of departure to point of destination, as well as minimizing travel costs and travel time. When the distance from one (initial) node to another (destination) node is the same with the distance to travel back from destination to initial, this problems known to the Traveling Salesman Problem (TSP), otherwise it call as an Asymmetric Traveling Salesman Problem (ATSP). The most recent optimization techniques is Symbiotic Organisms Search (SOS). This paper discuss how to hybrid the SOS algorithm with variable neighborhoods search (SOS-VNS) that can be applied to solve the ATSP problem. The proposed mechanism to add the variable neighborhoods search as a local search is to generate the better initial solution and then we modify the phase of parasites with adapting mechanism of mutation. After modification, the performance of the algorithm SOS-VNS is evaluated with several data sets and then the results is compared with the best known solution and some algorithm such PSO algorithm and SOS original algorithm. The SOS-VNS algorithm shows better results based on convergence, divergence and computing time.

1. Introduction

Combinatorial optimization from the past has often been used to solve problems in science, engineering, and commercial applications. Since the Mathematical model began to be used to help solve problems in life real, algorithms have become important in recent years due to their flexibility to find a solution. One of the combinatorial problems in the field of transportation is looking for a shortest route that can be traveled from the starting point to the destination point, as well as minimize travel costs and travel time. Such problems such as this is known as Traveling Salesman Problem (TSP). Author **Error! Reference source not found.** suggests that TSP issues have become a concern over the past two decades and various approaches have been proposed for solve the problem. Then research to solve these TSP issues continue to be developed with modifications and with new approaches such as particle swarm optimization [2], Neural network for Euclidean TSP [3], 2-opt based approach on differential evolution [4]. Simulated annealing with greedy search [5], Ant colony with multidepot Multiple TSP [6], Evolutionary Algorithm [7], Lower tolerance branch and bound [8], and Genetic algorithm [9].



The asymmetric traveling salesman problem (ATSP) is a well-known combinatorial optimization problem, in which bi-directional distances between a pair of cities are not necessarily identical. An optimal solution to the ATSP minimizes the total distance traveled to visit all cities in a list [10][11]. The asymmetric traveling salesman problem is defined on a directed graph $G = (V, A)$, where $V = \{1, \dots, n\}$ is the vertex set, $A = \{(i, j) : i, j \in V\}$ is the arc set, and a non-symmetric cost matrix (c_{ij}) is defined on A [12]. For the simple equation [13] describe Let c_{ij} be the distance from city i to city j . If $c_{ij} = c_{ji}$ for all i and j , then this problem is known as the symmetric traveling salesman problem; otherwise, it is designated as an asymmetric traveling salesman problem (ATSP). According [14] and [15], the asymmetric TSP, which is more general in principle and has numerous real-world applications. In contrast to the symmetric TSP, the distance from one city to another in the ATSP may not necessarily be the same as the distance on the reverse direction. Asymmetric traveling salesman problem is one of a class of difficult problems in combinatorial optimization that is representative of a large number of scientific and engineering problems. ATSP and its variants are commonly used models for formulating many practical applications in manufacturing scheduling problem. For example, the scheduling problem in a discrete manufacturing is mainly concerned with how to determine the sequence of jobs so as to minimize the total set-up cost [16].

One of new metaheuristics is the Symbiotic Organism Search (SOS). This method was developed by Cheng and Prayogo [14] that is inspired by the biological relationship between organisms on ecological systems. Obstacles which is encountered in the application of SOS algorithm to solve the combinatorial problem is on the parasitic phase of SOS algorithm requires to create a parasitic organism using the dimensions of the objective function to be sought (Continuous function), whereas TSP and ATSP does not have dimension of purpose function for it is necessary there is another suitable mechanism to replace the mechanism in the parasitic phase. Authors [17] also suggested modifications should be made to the phase parasites by adding more specific search algorithms to overcome combinatorial problems. The proposed mechanism for replacing this parasitic phase is to use mutation mechanism. The mutation mechanism allows for bringing in new individuals who are not from the previous population with refers to the change or replacement of elements of the solution vector [18].

This paper focusses on solving TSP and ATSP using SOS algorithm by modifying the SOS algorithm to be applicable to TSP and ATSP. The modification is done by adding mutation mechanism to replace the mechanism at that parasitic phase to find the best route. SOS is hybridized with Variable Neighbourhood Search algorithm as a local search to generate the better initial leader.

2. Research Methods

2.1 Symbiotic Organisms Search (SOS)

SOS is one of new metaheuristics that is inspired by the interaction behavior seen among organisms in universe. The organism has the basic nature of being unable to live alone so have a dependence on other species to maintain sustainability. This dependence-based relationship is known as a symbiosis. There is some forms of symbiosis i.e symbiotic of mutualism, symbiotic of commensalism and symbiotic of parasitism. Here is the flow of SOS:

1. Initialization

$$\text{Ecosystem} = \text{rand} \times ((\text{ub}-\text{lb}) + \text{lb}) \quad (1)$$

Where :

Rand= a number randomly between 0 to 1

Ub = upper bound value

Lb = lower bound value

2. REPEAT

- The phase of mutualism
- Commensalism Phase

- Phase Parasitism

3. UNTIL (Perform until termination criteria are met)

- Mutualism Phase

This SOS phase mimics mutualistic relationships between organisms in nature. SOS describes X_i is an organism compatible with members of the ecosystem. Other organisms X_j Then randomly selected from the ecosystem to interact with X_i . Both organisms Engage in mutualistic relationships with the goal of improving survival together with benefits in the ecosystem. New candidate solutions for X_i and X_j are calculated based on the mutualistic symbiosis between the organisms X_i and X_j , which are modeled on equations (2) and (3) follows [14]:

$$X_{i\text{new}} = X_i + \text{rand}(0,1) * (X_{\text{best}} - \text{Mutual_Vector} * \text{BF1}) \quad (2)$$

$$X_{j\text{new}} = X_j + \text{rand}(0,1) * (X_{\text{best}} - \text{Mutual_Vector} * \text{BF2}) \quad (3)$$

$$\text{Mutual Vector} = \frac{X_i + X_j}{2} \quad (4)$$

- Commensalism Phase

Similar to the phase of mutualism, an organism X_j is randomly selected from the ecosystem to interact with X_i . In this case, the X_i organism tries to get advantage of interaction. However, organism X_j alone does not profit or suffer relationship. New candidate solutions from X_i are calculated according to the symbiosis between organisms commensal X_i and X_j , which are modeled in Eq. (2). Following the rules, X_i organism updated only if the new fitness are better than the previous interaction fitness.

$$X_{i\text{new}} = X_i + \text{rand}(-1,1) * (X_{\text{best}} - X_j) \quad (5)$$

- Parasitism Phase

In SOS, the X_i organism is given a role similar to anopheles mosquito through the creation of an artificial parasite called " Parasite Vector ". Parasite Vector is made in space search by duplicating the X_i organism, then modify selected at random dimensions using random numbers. Organism X_j is randomly selected from the ecosystem and serves as the host for vector parasites. Parasite Vector tries to replacing X_j in the ecosystem. Both organisms are then evaluated to measure their fitness value. If Parasite Vector has a better fitness value, it will kill the organism X_j and assume its position in the ecosystem. If the fitness value of X_j is better, X_j will have immunity from parasites and Parasite Vector will no longer can live in the ecosystem.

- Mutation on TSP

Mutations are possible to elicit new non-originating individuals of cross breeding results. Mutation refers to the change of order or replacement of elements from solution vector (on the TSP problem) by generating a new value (function optimization). This process is done by choosing which individual to mutate with using random numbers. Individuals who are selected then mutate by placing a sequence of routes from the individual. Mutation can be swap, slide and flip .

2.2 Variable Neighbourhood Search (VNS)

According to [19], VNS's basic idea is to change the environment in searching for better solutions. VNS starts from a descent method to reach the local minimum, then investigate systematically or randomly, causing the environment to be further away from this solution. Each time, one or more points in the current environment is used as the initial solution for local descent. One jump point from the current solution as a new reference if and only if a better solution is found. VNS is not like the trajectory method on Simulated Annealing or Taboo Search and does not include prohibited steps in the process. Although simplicity is more specific.

3. Discussion

The SOS-VNS algorithm is a hybrid of SOS and VNS algorithm. The VNS is a local search algorithm widely used for solving combinatorial problem. The steps of the hybrid SOS-VNS algorithm are then described in algorithm shown in Figure 1.

Table 1. Data Set, Type, Number of city and best known solution.

Data Set	Type	Number of City	Best Known Solution
Co04	Asymmetric	4	21
SP11	Symmetric	11	133
UK12	Symmetric	12	1733
LAU15	Symmetric	15	291
GR17	Symmetric	17	2085
BR17	Asymmetric	17	39
WG22	Symmetric	22	842

Table 2. Result of computation.

Data set	PSO			SOS			SOS-VNS		
	Convergence	divergence	comp time	Convergence	divergence	comp time	convergence	Divergence	comp time
Co04	0	0	0,014 9	0	0	0,012 4	0	0	0
SP11	0,00861	77	10,48	0,0015	1	13,8	0,0015	1	0,648
UK12	0,0547	215	17,14	0,0084	67	14,18	0,0051	6	6,489
LAU15	0,29	204	39,91	0,016	24	16,83	0,0109	12	6,363
GR17	0,1876	570	115,9 7	0,01813	107	63,99	0,0162	73	8,081
BR17	0,1743	11	49,39 9	0,0512	7	12,45 3	0,01025	2	11,97 5
WG22	0,627	621	63,35	0,114	249	33,55	0,0762	165	14,92 5

Table 2 compares the running results of Particle Swarm Optimization (PSO), original Symbiotic organisms Search (SOS) and Symbiotic organisms Search with Variable Neighbourhood Search (SOS-VNS) on a set of benchmark TSP and ATSP in TSPLIB. Convergence is used to measure the ability of the algorithm to find result near the optimal solution. The smaller of convergence value shown the algorithm is able to find the near optimal solution. While divergence is used to see how much deviation generated from the algorithm. Smaller level of divergence shown the more robust algorithm. Computation time (comp time) is used to evaluate the efficiency of optimization algorithm. All of algorithm are written in MATLAB code and limit of iteration of each is 50000 on Intel(R) Core(TM) i3 processor 2.27GHz.

Based on the results of 10 runs are carried out for each algorithm, overall SOS-VNS algorithm can be said to be superior than PSO and SOS original. Convergence value of the SOS-VNS algorithm is superior to all data sets. This proves that the SOS-VNS algorithm is capable of generating a set of solutions closer to the optimal solution than PSO and SOS original. In terms of SOS-VNS divergence it also shows that better than PSO and SOS original, with smaller divergence values than for all data sets. For computing time SOS-VNS also shown that it more efficient than PSO and SOS original.

4. Conclusion

In this paper, a novel and hybrid SOS with VNS is proposed as a new approach for solving symmetric and asymmetric TSP. This new algorithm is compared with PSO and the original SOS. The result show excellent results. Using convergence, divergence, and computation time, SOS-VNS is better than PSO and original SOS. From 7 data sets tested, SOS-VNS is able to solve all data sets.

Pseudocode SOS-VNS Algorithm for TSP

```

Input  : Number of population, maximum iteration
Output : Best Route, Total distance
Begin
Generate ecosystem, calculation initial distance,
Generate neighbourhood set
    Shake. Random solution x' of Nk (x).
    Local search.
    Improve or not. if x'' better than x*, set (x ← x'').
calculation distance
set IT=1
Repeat
    while it < itmax
        for i = 1: ecosize; update best distance and best route
            Mutualism Phase
                If distance better than before
                    | eco(i; j) ← x (new1; new2)
                end
            Commensalism Phase
                If distance better than before
                    | eco(i) ← x (new1)
                end
            Parasitism Phase
            Rand
                If rand < 0,33
                    Flip
                Elseif rand > 0,67
                    Swap
                Else
                    Slide
            end
            If distance better than before
                | eco(i) ← x (new1)
            end
        end
    End
    It=it+1
    If it == itmax
        | break
    end
end
end

```

Figure 1. Pseudocode Algorithm of SOS-VNS for TSP**References**

- [1] Moon C, Kim J, Choi G and Seo Y 2002 An Efficient Genetic Algorithm for the Traveling Salesman Problem with Precedence constraints *European Journal of Op Res.* **140** 606-617
- [2] Shi X H, Liang Y C, Lee H P, Lu C, and Wang Q X 2007 Particle Swarm Optimization-based Algorithm for TSP and Generalized TSP *Information Process Letters* **103** 169-176
- [3] Creput J C and Koukam A 2009 A Memetic Neural Network for The Euclidean Traveling Salesman Problem *Neurocomputing* **72** 1250-1264
- [4] Chiang C W, Lee W P and Heh J S 2010 A 2-opt Based differential Evolution for Global Optimization *Applied Soft Computing* **10** 1200-1207

- [5] Geng X, Chen Z, Yang W, Shi D and Zhao K 2011 Solving the Traveling Salesman Problem Based On An Adaptive Simulated Annealing Algorithm with Greedy Search *Applied Soft Computing* **11** 3680-3689
- [6] Ghaufurian S and Javadian N 2011 An Ant Colony Algorithm for Solving Fixed Destination Multi-depot Multiple Traveling Salesman Problems *Applied Soft Computing* **11** 1256-1262
- [7] Liao Y F, Yau D H and Chen C L 2012 Evolutionary Algorithm to Traveling Salesman Problem *Computers and Mathematics with Applications* **64** 788-797
- [8] Germs R, Goldenngorin B and Turkensteen M 2012 Lower Tolerance-based Branch and Bound Algorithms for ATSP *Computers & Operations Research* **39** 291-298
- [9] Rani K and Kumar V 2014 Solving traveling salesman problem using genetic algorithm based on heuristic crossover and mutation operator *Int Journal of Research in Engineering & Technology* **2** 27-33.
- [10] Nagata Y and Soler D 2012 Expert Systems with Applications A new genetic algorithm for the asymmetric traveling salesman problem. *Expert Systems With Applications* **39** 8947–8953
- [11] Xing A L, Chen Y, Yang K, Hou F, Shen X and Cai H 2008 A hybrid approach combining an improved genetic algorithm and optimization strategies for the asymmetric traveling salesman problem *Engineering Applications of Artificial Intelligence* **21** 1370–1380
- [12] Öncan T, Altinel I K and Laporte G 2009 A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers & Operation Research* **36** 637–654
- [13] Sherali H D, Sarin S C and Tsai P 2006 A class of lifted path and flow-based formulations for the asymmetric traveling salesman problem with and without precedence constraints, *Discrete Optimization* **3** 20–32.
- [14] Cheng M Y and Prayogo 2014 Symbiotic Organisms Search : A New Metaheuristic Optimization Algorithm *Computers and Structures* **139** 98-112.
- [15] Ausiello G, Bonifaci V and Laura L 2008 The on-line asymmetric traveling salesman problem. *Journal of Discrete Algorithm* **6** 290–298.
- [16] Bai J, Yang G, Chen Y, Hu L and Pan C 2013 A model induced max-min ant colony optimization for asymmetric traveling salesman problem *Applied Soft Computing Journal* **13** 1365–1375
- [17] Ruskartina E, Vincent F Y, Santosa B and Perwira A A N R 2015 Symbiotic Organism Search (SOS) for Solving Capacited Vehicle Routing Problem *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering* **9** 873-877
- [18] Santosa B and Willy P 2011 *Metode Metaheuristik Konsep dan Implementasi*, Guna Widya, Surabaya
- [19] Alba E 2005 *Parallel Metaheuristics : A New Class of Algorithms*. Hoboken, New Jersey: John Wiley & Sons