Symbiotic Organism Search (SOS) for Solving the Capacitated Vehicle Routing Problem

Ruskartina Eki, Vincent F. Yu, Santosa Budi, A. A. N. Perwira Redi

Abstract—This paper introduces symbiotic organism search (SOS) for solving capacitated vehicle routing problem (CVRP). SOS is a new approach in metaheuristics fields and never been used to solve discrete problems. A sophisticated decoding method to deal with a discrete problem setting in CVRP is applied using the basic symbiotic organism search (SOS) framework. The performance of the algorithm was evaluated on a set of benchmark instances and compared results with best known solution. The computational results showed that the proposed algorithm can provide good solution as a preliminary testing. These results indicated that the proposed SOS can be applied as an alternative to solve the capacitated vehicle routing problem.

Keywords—Symbiotic organism search, vehicle routing problem, metaheuristics, Solution Representation.

I. INTRODUCTION

CAPACITATED vehicle routing problem (CVRP) or Vehicle Routing Problem (VRP) was first introduced by [1]. It had been proven by [2] that this problem is NP-hard problem since it is not solvable in polynomial time. CVRP is one of the most widely-studied problems in combinatorial optimization and the literature provides an extensive stream of heuristics and metaheuristics solution techniques. CVRP can be briefly described as a set of n customers that must be served by m number of homogeneous vehicles. Let G = (V,A) be a complete graph where V = {0,...,n} is the vertex set and A is arc set. Vertex i = 1,...,n correspond to the customer and vertex 0 correspond to the depot. Each customer must be assigned to exactly one vehicle to be visited at exactly once. At each visit, vehicle capacity Q and maximum allowable route length L should not be violated. Each customer i is associated with a non-negative demand d_i, service time s_i, and travel cost c_{ij}. The objective is to satisfy total demand of customers while minimizing total network cost [3].

Several classical heuristics approaches were proposed to solve capacitated vehicle routing problem (CVRP). Clarke and Wright [4] used savings algorithm to solve CVRP. This algorithm usually applied to CVRP that emphasizes number of vehicle as decision variables. Sequential insertion heuristics was proposed by [5] and [6] to solve CVRP with undefined number of vehicles. Gillet and Miller [7], Wren and Holliday [8] presented elementary clustering method called as sweep algorithm; and [9] proposed generalized assignment-based algorithm and [10] proposed location-based heuristics. These last two heuristics assume that the number of vehicles is fixed. In recent years, several metaheuristics algorithm have been proposed for CVRP. Robuste, Daganzo, and Souleyrette [11] implemented simulated annealing with neighborhood structure in the context of VRP, the algorithm was tested on several instances but no comparisons are available to verify the performance. Then, [12] proposed a Simulated Annealing with Tabu Search algorithm and were more successful in his implementation. Some examples of other metaheuristics were also proposed such as Genetic algorithms [13], [14]; Ant colony optimization [15], Artificial Bee Colony algorithm [16], and Particle Swarm Optimization [17].

Cheng and Prayogo [18] presented a new metaheuristics called Symbiotic Organism Search (SOS) inspired from interaction among organisms in the ecosystem known as symbiotic relationship. This algorithm was proposed initially to solve continuous engineering optimization problems. SOS showed considerable robustness in its performance when tested on complex mathematical benchmark problems. SOS has never been used to solve discrete problem, such as, routing, scheduling, or assignment problems. This could be our motivation to first introduce SOS to solve discrete problem. In this paper, we introduce SOS to solve CVRP. We evaluate the performance of the SOS using classical benchmark instances. In order to enhance the solution quality, we adopted the use of solution representation 1 (SR-1) presented by [19]. Originally SR-1 is used for Particle Swarm Optimization (PSO) to deal with CVRP. Solution representation is defined as an encode solution for each particle or candidate solution, while the method to transform it to problem specific solution is called decoding method. This solution representation is constructed with original PSO framework which uses real-valued particle positions instead of discrete-valued representation. The
preaminatory test runs show that SOS algorithm with SR-1 can obtain promising results for the CVRP.

The remainder of this paper is organized as follows: Section II describes in detail the symbiotic organism search algorithm. Section III discusses SOS with SR-1. Section IV presents the computational results. Finally, section V gives the conclusions and directions for future researches.

II. SYMBIOTIC ORGANISM SEARCH ALGORITHM

Almost all metaheuristics algorithm were inspired from natural biological phenomena. Particle swarm optimization simulates the social behavior of bird flocks. Genetic algorithm, a type of evolutionary algorithm mimics the process of natural selection [20]. SOS simulates interactions between two organisms survival in the ecosystem. This interaction is also known as Symbiosis. Symbiotic relationship is defined as reliance-based relationship among organism in order to fulfill their sustenance or even survive in nature. Generally, there are three kinds of symbiotic relationships namely, mutualism, commensalism, and parasitism. In mutualism, organisms benefit from each other. Like for example, the interaction between starlings and buffalo. Starlings get ticks from buffalo’s skin for sustenance. The itching on buffalo’s skin will be reduced in return. Commensalism takes place when an organism gets benefits while the other is not significantly harmed and helped. Like for example, the interactions between remora fish and shark. Remora fish eats leftovers from the shark without bothering the shark at all. Parasitism takes place when an organism obtains benefits from the interaction while other is harmed. Like for example, the anopheles mosquito and human body. Anopheles inducts parasite into the human body which poses fatal threats causing the body to eventually die. Generally, most organisms in ecosystem are doing these kinds of symbiotic relationships to adapt to environmental changes and to create survival strategies over long periods of times [18].

Like other population-based metaheuristics, SOS utilizes an initial population called ecosystem in order to provide candidate solutions in the searching space to obtain optimal solution. Organisms are generated randomly from the searching space that has upper and lower bounds. Then, the best solution ($X_{\text{best}}$) is selected among all organisms. Commonly, metaheuristics have operators in order to generate a new solution in each iteration. The phases in SOS such as mutualism, commensalism, and parasitism serve as the operators. Each organism interacts with other organisms randomly in the population through all phases. For example, with an eco_size of n member organisms, each member will go through each phase of the algorithm. This is the inner loop of the algorithm. Then, the process repeats until the termination criteria is reached. This is the outer loop of the algorithm specifically using maximum number of iterations as a stopping policy. Fig. 1 describes the simple SOS algorithm procedure.

![Fig. 1 SOS Algorithm Procedure](image)

III. SOS FRAMEWORK FOR SOLVING CVRP

The proposed algorithm to solve CVRP is based on SOS proposed by [18] and added with solution representation 1 proposed by [19]. This algorithm uses four parameters, namely $I_{\text{size}}$, $\text{max}_it$, $\text{Non}_improve$, and $\text{Parasite force (PF)}$. $I_{\text{size}}$ is the number of organisms in the ecosystem usually called population size. $\text{max}_it$ is maximum number of iteration and $\text{Non}_improve$ is the allowable number of update organisms during which $X_{\text{best}}$ has not been improved. $\text{Parasite force}$ is a parameter from 0 to 1 that represents the probability of the creation of $\text{Parasite}_{\text{vector}}$.

Algorithm 1 presents the proposed SOS algorithm. It begins with generating $I_{\text{size}}$ of organisms with a particular representation that corresponds to $I_{\text{size}}$ different sets of vehicles. Then, we select the best organism from the ecosystem to be the $X_{\text{best}}$. Then, a movement procedure which is called update organism is applied to each organism. Whenever a better set of route is found after all phase done, $X_{\text{best}}$ is updated. In addition, in order to improve the solution quality, we apply local search improvement using neighborhood operator such as swap.
reverse, and insert to each organism. The algorithm is terminated when the best solution \((X_{best})\) has not been improved in \textit{Non improve} or the iteration reached \textit{max it}.

**Algorithm 1.** SOS framework for CVRP

A. Mutualism Phase

Two organisms interact in mutualism phase. \(X_i\) is an organism selected sequentially while \(X_j\) is selected randomly from the ecosystem. Both organisms interact with the purpose of increasing mutual survival advantage in the ecosystem. The relationship is represented by \textit{Mutual vector} equation.

\textit{Mutual vector} represents the relationship characteristics between organism \(X_i\) and \(X_j\). BF\(_1\) and BF\(_2\) represent the beneficial factor from both organisms. The value of BF\(_1\) and BF\(_2\) are determined randomly as either 1 or 2. Then new candidate solution from \(X_i\) and \(X_j\) are calculated as in step 1.3 in Algorithm 2. Organisms are updated if their fitness is better than their previous. Finally, only the fittest organism is selected to go to the next phase. By doing this phase, we update two organisms at once. The pseudocode for mutualism phase is presented in Algorithm 2.

B. Commensalism Phase

The fittest \(X_i\) from mutualism phase becomes the input for commensalism phase. Similar to Mutualism phase, an organism \(X_j\) is selected randomly from the ecosystem. Then the interaction between \(X_i\) and \(X_j\) is calculated as step 1.2. The new candidate solution from \(X_i\) and \(X_j\) are calculated as in step 1.3 in Algorithm 3. Different from Mutualism phase, output of commensalism is only one organism \((X_j)\). \(X_i\) is updated only if its new fitness is better than previous. The pseudocode of Commensalism is shown in Algorithm 3.

**Algorithm 2.** Pseudocode for Mutualism Phase

**Algorithm 3.** Pseudocode for Commensalism Phase

C. Parasitism Phase

Parasitism phase has the same structure with mutation operator in Genetic Algorithm. First, \(X_i\) from commensalism phase becomes the input for parasitism phase. From \(X_i\), we create an artificial parasite called “Parasite vector”.

Parasite vector is created by duplicating organism \(X_i\), then modifying it based on the Parasite force.

Then, \(X_j\) is selected from the ecosystem to interact with \textit{Parasite vector}. If \(X_j\) is better than the Parasite vector, then \(X_j\) will survive. Otherwise, \textit{Parasite vector} kills \(X_j\) and replaces its current position. The steps of this phase are shown in Algorithm 4.

D. Solution Representation and Decoding Method

The solution representation 1 or SR-1 of CVRP consists of \((n+2m)\) dimension. The first \(n\) dimensions refer to customers and the last \(2m\) dimensions are related to vehicle. In each phase, the decoding method is applied after the organism is updated. Each organism is encoded as a real number. The notations and decoding algorithm for SR-1 are presented in Algorithm 5.
The SOS algorithm is implemented in C++ using Microsoft Visual Studio on a Windows PC with Intel Core 2 Duo CPU, 3 GHz, and 4 GB RAM. The SOS parameters are as follows: number of organism (Ghz) = 100, number of iteration (ii) = 100, Non-improvement = 50, and Parasite Force = 0.8.

The characteristics of SOS have some resemblance with PSO algorithm. The movement mechanism of PSO called velocity is relatively small at 3.29%. In addition, the result for vrpnc1 is better. However, the average deviation for all instances with 120 - 199 numbers of customers show differences ranging from 3.11 – 11.34. The result from SOS, especially for the large instances still needs further improvement. However, the average deviation for all instances is relatively small at 3.29%. The result for vrpnc1 is exactly the same with best known solution. With this preliminary test, SOS for CVRP shows promising results.

V. CONCLUSION AND FUTURE STUDY

Symbiotic organism search for solving capacitated vehicle routing problem is presented in this paper. The SOS was initially proposed to solve engineering optimization problem, a continuous mathematical problem. This paper proposed a new approach of SOS to solve the discrete problem, CVRP. SOS is implemented for CVRP with Solution representation 1 decoding method. The computational result shows that with the preliminary tests, SOS can obtain reasonable and promising results.

Future researchs may focus on developing SOS for solving CVRP that can handle both small and large instances efficiently. In addition, the researcher may also consider using SOS to solve another discrete problem.
### TABLE I

**Computational Result of SOS for CVRP**

<table>
<thead>
<tr>
<th>Instance</th>
<th>n</th>
<th>m</th>
<th>Q</th>
<th>s</th>
<th>L</th>
<th>BKS*</th>
<th>SOS best</th>
<th>Diff (%)</th>
<th>Comp. Time (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>vrpnc1</td>
<td>50</td>
<td>5</td>
<td>160</td>
<td>0</td>
<td>∞</td>
<td>524.61</td>
<td>524.61</td>
<td>0.00</td>
<td>0.35</td>
</tr>
<tr>
<td>vrpnc2</td>
<td>75</td>
<td>10</td>
<td>140</td>
<td>0</td>
<td>∞</td>
<td>835.26</td>
<td>842.39</td>
<td>0.85</td>
<td>2.28</td>
</tr>
<tr>
<td>vrpnc3</td>
<td>100</td>
<td>8</td>
<td>200</td>
<td>0</td>
<td>∞</td>
<td>826.14</td>
<td>839.70</td>
<td>1.64</td>
<td>2.31</td>
</tr>
<tr>
<td>vrpnc4</td>
<td>150</td>
<td>12</td>
<td>200</td>
<td>0</td>
<td>∞</td>
<td>1028.42</td>
<td>1060.38</td>
<td>3.11</td>
<td>18.16</td>
</tr>
<tr>
<td>vrpnc5</td>
<td>199</td>
<td>17</td>
<td>200</td>
<td>0</td>
<td>∞</td>
<td>1291.29</td>
<td>1378.83</td>
<td>6.78</td>
<td>46.36</td>
</tr>
<tr>
<td>vrpnc6</td>
<td>50</td>
<td>6</td>
<td>160</td>
<td>10</td>
<td>200</td>
<td>555.43</td>
<td>557.89</td>
<td>0.44</td>
<td>0.49</td>
</tr>
<tr>
<td>vrpnc7</td>
<td>75</td>
<td>11</td>
<td>140</td>
<td>10</td>
<td>160</td>
<td>909.68</td>
<td>936.85</td>
<td>2.99</td>
<td>1.60</td>
</tr>
<tr>
<td>vrpnc8</td>
<td>100</td>
<td>9</td>
<td>200</td>
<td>10</td>
<td>230</td>
<td>865.94</td>
<td>872.29</td>
<td>0.73</td>
<td>3.16</td>
</tr>
<tr>
<td>vrpnc9</td>
<td>150</td>
<td>14</td>
<td>200</td>
<td>10</td>
<td>200</td>
<td>1162.55</td>
<td>1294.37</td>
<td>11.34</td>
<td>14.44</td>
</tr>
<tr>
<td>vrpnc10</td>
<td>199</td>
<td>18</td>
<td>200</td>
<td>10</td>
<td>200</td>
<td>1395.85</td>
<td>1494.66</td>
<td>7.08</td>
<td>39.83</td>
</tr>
<tr>
<td>vrpnc11</td>
<td>120</td>
<td>7</td>
<td>200</td>
<td>0</td>
<td>∞</td>
<td>1042.11</td>
<td>1047.83</td>
<td>0.55</td>
<td>6.95</td>
</tr>
<tr>
<td>vrpnc12</td>
<td>100</td>
<td>10</td>
<td>200</td>
<td>0</td>
<td>∞</td>
<td>819.56</td>
<td>820.25</td>
<td>0.08</td>
<td>6.68</td>
</tr>
<tr>
<td>vrpnc13</td>
<td>120</td>
<td>11</td>
<td>200</td>
<td>50</td>
<td>720</td>
<td>1541.14</td>
<td>1658.00</td>
<td>7.58</td>
<td>10.19</td>
</tr>
<tr>
<td>vrpnc14</td>
<td>100</td>
<td>11</td>
<td>200</td>
<td>90</td>
<td>1040</td>
<td>866.37</td>
<td>891.66</td>
<td>2.92</td>
<td>6.43</td>
</tr>
</tbody>
</table>

*Best Known Solution; Diff = (BKS – SOS Best) / BKS * 100%

### REFERENCES

4. G. Clarke, J.V. Wright. *Scheduling of vehicle from a central depot to a number of delivery points*. Operation research, 12:568-581, 1964