A Viral System to Solve Generalized Orienteering Problem

Budi Santosaa and Artya Lathifah

"Teknik Industri, ITS, Kampus ITS Surabaya, 60111, Indonesia"

Abstract

Transportation problem is one of combinatorial problems interesting to discuss. Generalized Orienteering Problem (GOP) is a transportation problem. GOP is the development of Traveling Salesman Problem. The purpose of GOP is to maximize total score with a maximum mileage limit. In GOP, there is set of scores which represents level of satisfaction that can be achieved by an orienteer by visiting some places. GOP is a combinatorial optimization problem where metaheuristics is suitable tool to solve it. In this paper, Viral System (VS) is used to solve GOP. VS is a new algorithm that mimics the pattern of virus life. The algorithm performed good enough for other combinatorial cases. To evaluate our VS on GOP, the test data of 27 cities were used. Five different weights have been tried and VS showed promising performance. Therefore, VS still needs further modification in order to compete with other algorithms.

Keywords: Generalized Orienteering Problem, Combinatorial Optimization, Traveling Salesman Problem, Metaheuristics, Viral Systems

1. Introduction

One of interesting issue in transportation is the Traveling Salesman Problem. TSP is a combinatorial problem where a salesman should visit a number of cities, each city only visited once, the total distance traveled should be minimized and he has to start and end in the same city. The problem is how he finds the optimal route. Optimal route is a route with the shortest total distance he traverses. Orienteering Problem (OP) is one type of TSP with score. Score is a benefit or a satisfaction that an orienteer gained from visiting a node or city. Score can be in the form of satisfaction, profits, and other aspects. The objective of an OP is to find a tour that start and end on the same node with maximum total score without violating the maximum distance constraint. Generalized Orienteering Problem (GOP) an OP where an orienteer will visit several cities without violating maximum total distance constraint. Each city has its own score. The problem is which cities an orienteer should visit and what is the best route to obtain maximum score.

Viral Systems (VS) is a new metaheuristics which is very effective to solve combinatorial optimization problem in medium to large size [2]. VS had been used to solve discrete combinatorial
problems, such as the Knapsack Problem, Shortest path [9] and, Steiner Tree [2]. VS performed quite good in these cases. In this paper VS is used to solve GOP. This problem is a discrete combinatorial which is quite complex and suitable to be solved using VS. Therefore, this paper proposes the development of VS to solve GOP problem. The next two sections present literature review and description of the data set. In section 4 it is presented VS procedure to solve GOP. The next section discusses about experiments and finally we concluded the results.

1. Literature Review

2.1 Generalized Orienteering Problem

Generalized Orienteering Problem (GOP) is introduced by Ramesh [6]. An orienteer will visit several cities without violating the constrained distance. The optimal solution is the route with maximum score. An orienteer will go to the place or node with several reasons, and each reason has a score. An orienteer has a limit to travel, for example distance or cost. On the other hand an orienteer wants to get optimal satisfaction from the places that were visited. This problem can be implemented to the touring problem in which a tourist has a limit to trip to visit tourist spots. Tourist spot has four scores, namely: natural beauty, historical value, cultural events, and business opportunities. The selection of tourist attraction on the basis of satisfaction and the limit of the distance can be an interesting problem to solve. This problem belongs to the NP-Hard problems, which has a lot of alternative solutions. Ramesh [6] using a heuristic method namely Four Phase Heuristic to solve GOP. The other approach to solve GOP was dynamic programming [3]. Despite the good results obtained, exact and heuristic methods need very long time to do, especially in large cases. Besides exact methods and heuristic, the GOP has been tried to resolve using Artificial Neural Network by Wang et al. in [4], Harmony search [4], Genetic Algorithm [12] and Cross entropy [7].

2.2 Viral Systems (VS)

Viral Systems is a metaheuristics which imitates the life of virus. A virus is a transition creature between living things and inanimate objects. Viruses are unique in terms of reproduction. A virus uses the cells of organisms as a container for replication. When the virus injects DNA into the host cell in the body, there are two types of replication: lytic cycle and lysogenic cycle. Lytic cycle allows the virus to replicate in the host cell. At a certain amount of viral replication will break the wall and out of the host cell. While the lysogenic cycle allows the virus to remain in the host cell is switched off until there is something to activate it. Then the cell of organism that has been injected was broken or mutated, it depends on the health of the cell itself. In the system of the virus, the virus always seeks and infects susceptible cells. Each cell was trying to produce antibodies to fight the viral infection. The cell that fails to produce antibodies will be grouped into the clinical picture. VS define the clinical picture of the infected population as a description of all the cells infected by viruses. In terms of computing, Clinical Picture containing solutions are being explored (the genome of infected cells, in biological terms) and total nucleus-capsids were replicated in the form of NR (for lytic replication) or IT (for lysogenic replication) [1].

In the infected cells, there are virus replication in the lytic or lysogenic. Both replication is done based on probability of lytic \( p_l \) and the probability of lysogenic \( p_g \) and \( p_l + p_g = 1 \). Initially, the number of nucleus capsid is zero (\( NR = 0 \) and \( IT = 0 \)).
2.2.1 Lysogenic replication

In the lysogenic replication, the activation process of mutation occurs after iteration reaches the maximum limit (LIT). LIT value depends on the health condition of the cells, so the cells are healthy (minimization function \( f(x) \) of high value) would have a lower probability of infection so the value of LIT becomes higher. In contrast, unhealthy cells will have a lower value of LIT [1]. The value of LIT can be calculated from the value of initial LIT \( LIT^0 \) based objective function value relative to the best objective function has been found so far, which can be calculated by the following equation.

\[
LIT_{sel-x} = LIT^0 \left( \frac{f(x) - f(\bar{x})}{f(\bar{x})} \right) \quad (1)
\]

where:

- \( LIT_{sel-x} \) = LIT value of cell-x,
- \( LIT^0 \) = initial LIT value,
- \( f(x) \) = Objective function value of cell-x,
- \( f(\bar{x}) \) = The best value of objective function has been found so far.

1.2. Lytic replication

In lytic replication, first we need to obtain the number of replication nucleus capsid (NR). NR is calculated for each iteration as a function of the binomial variable (Z), its value is added to the existing NR on the clinical picture. Z is calculated using the binomial distribution is given by the maximum value of the nucleus replication capsid (LNR) and the probability of the single replication (p_r).

\[
z = \text{bin}(\text{LNR}, p_r) \quad (1)
\]

\[
\text{NR}_{iter} = \text{NR}_{iter-1} + z \quad (2)
\]

LNR represents the limit cell wall rupture and viruses that inhabit it out. LNR value depends on the value of the objective function \( f(x) \). Cells with the value \( f(x) \) which has a high probability of infection is lower, then the value will be higher LNR [9]. LNR Value can be calculated from the value of initial LNR.
(LNR\(^0\)) based objective function value relative to the best objective function has been found, which can be calculated by the following equation.

\[
LNR_{sel-x} = LNR^0 \left( \frac{f(x) - f(\bar{x})}{f(\bar{x})} \right)
\]

(4)

where:
- \(LNR_{sel-x}\) : LNR value of cell-x
- \(LNR^0\) : initial LNR value
- \(f(x)\) : Objective function value of cell-x
- \(f(\bar{x})\) : The best value of objective function has been found so far.

After that, the cells that have a value exceeding NR LNR be broken so that the virus released. Each virus liberated would have probability \((p_i)\) to infect new cells other being around. If the cardinality of \(x\) is defined as \(|V(x)|\), the number of cells infected by the virus in the environment can be obtained by using the binomial random value \(|V(x)|\) and the probability \(p_i\).

\[
y = \text{bin}(|V(x)|, p_i)
\]

(3)

On the other hand, to maintain from the growth of a viral infection, the organism (combined cell) responds by producing antigen. In the clinical picture, individual infected cells produce antibodies with Bernoulli probability distribution \(A(x) = \text{Ber}(p_{an})\), which is the probability \(p_{an}\) antibodies released by cells in the clinical x-picture. Therefore, the population of infected cells produce antibodies based on a binomial distribution with parameters characteristic clinical picture size \((n)\) and the probability of producing antibody \(p_{an}\) [8].

\[
A(\text{populasi}) = \text{Bin}(n, p_{an})
\]

(4)

2.2.3 Mutation

Lysogenic replication allows the algorithms to perform searching by mutation. Mutation has a goal to bring new solutions that are totally different from the previous solution in order to get out of the local optimal solution [8]. A feasible solution can be mutated and transferred to obtain other feasible solutions. Given \(n\) jobs, each job is represented by a gene, there are several types of mutation, for examples [9]: flip, swap, slide, insertion and etc.

2.2.4 Neighborhood

The lytic replication allows the algorithms to perform searching by neighborhood. Neighborhood tracking is a method of solution adjacent to a good solution that has been found before. Therefore, this method makes the algorithm to do local search faster. Neighboring cells can be defined by exchanging a pair of genes in the genome of the cell before. Starting from the left gene first, interchange with a second gene that produced the first neighboring cells. For cells whose genome consists of \(n\) genes, the procedure is repeated until the \((n-1)\) neighboring cells are obtained [9].

![Figure 2. Cities in Eastern China](image-url)
2. Description of the Data

In this paper, the problem is an orienteer is going through 27 cities as showed in Figure 2. Each city has four scores namely: natural beauty, historical value, cultural events, and business opportunities. An orienteer has a limited maximum distance that can be passed, that is 5000 kms. The purpose of the GOP is to maximize the score which is obtained by combining those four scores. The objective function is formulated as follows:

$$\max Z = \sum_{g=1}^{m} W_g \left( \sum_{i \in P} \left[ S_i (i) \right]^{1/k} \right)$$ \hspace{1cm} (7)

While the constraints are as follows:

$$\sum_{j=2}^{n} X_{1j} = \sum_{j=2}^{n-1} X_{ij} = 1 \hspace{1cm} (8) \quad \sum_{i=2}^{n-1} X_{1k} = \sum_{j=2}^{n-1} X_{kj} \leq 1 \quad k=2,\ldots,n-1 \hspace{1cm} (9)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} X_{ij} \leq DMAX \hspace{1cm} (10) \quad X_{ij} \in \{0,1\} \quad i,j = 1,\ldots,n \hspace{1cm} (11)$$

$W_g$ is a weight for each score.

$S$: is score of each city on path of the orienteer.

Value of $k$ at this research is set to 5. This number was investigated by Wang [12] by testing on $k=1, k=3, k=4, k=5,$ and $k=10$, then indicating the premises $k=5$, the total score obtained is influenced by nodes that have big score. Constraint (8) to ensure the route begins and ends in the same node (city). Constraint (9) is to ensure the relationship between the cities and make sure that every city is visited only once. Constraint (10) indicates that the journey should not exceed the maximum distance (MAX). Constraint (11) shows that travel from city 1 to city n has two possibilities, which are skipped or no.

The distance for each city is calculated by following equation

$$d(x,y) = r \cdot \arcsin \left( \frac{\sin(c_1) \sin\left(\frac{\pi(90-b_1)}{180}\right)}{\sin(e)} \right) \hspace{1cm} (12)$$

where: $e = \arctan(d_1) + \arctan(d_2)$, $d_1 = \cos \frac{c_2}{2} \left( \cos \frac{c_3}{2} \tan \frac{c_1}{2} \right)$, $d_2 = \sin \frac{c_2}{2} \left( \sin \frac{c_3}{2} \tan \frac{c_1}{2} \right)$,

$c_1 = (a_1 - a_2) \frac{\pi}{180}$, $c_2 = (b_2 - b_1) \frac{\pi}{180}$, $c_3 = \pi - (b_1 + b_2) \frac{\pi}{180}$

$d(x,y)$ distance from city $x$ to city $y$, $a_1$: longitude city of $x$, $b_1$: latitude of $x$, $a_2$: longitude of $y$, $b_2$: latitude of $y$ and $r$ is radius of earth =6371 km. Each data point consists of city, longitude, latitude, and scores at each criteria ($S_i$).

4. Procedure of Viral Systems on GOP

Below is the procedure of VS to solve GOP. For simplicity reason, in this example only 5 cities are used. Step 1. Generate the routes

First of all, generates empty clinical picture (CP) as many as the size of sample. Set NR and IT = 0. Genome is filled by random permutations of routes
Step 2. Sub tour Phase
In this sub tour, it will cut route based on the maximum distance that can be taken. Total distance that is allowed is 1000.

Step 3. Objective Function Value
Calculate the objective Function (score of each route) for all cells in the clinical picture.

Step 4. Determine the type of cell replication
Determining the type of virus replication in each cell, if random value < plt is lytic replication, while the lysogenic is the contrary.

Step 5. Lytic Procedure
1) LNR cells are calculated based on the value of objective function relative to the best of objective function that has been found
2) Calculate replication of nucleus capsids with scores at random binomial probability of a single value of the LNR and replication probability (pr).
3) Update the value of NR in cells by adding z at the previous NR
4) If the value of NR>LNR in a genome, then the cell wall will be broken and removed from CP
5) Then list all cell neighbors (neighborhood). Neighborhood is conducted of the route before the sub tour.
   Route 2 → initial genome 1-2-5-3-4-1; Neighborhoods for genome 2 are: 1-5-2-3-4-1, 1-2-3-5-4-1, 1-2-5-4-3-1
6) Each cell has a probability to release antigen (pan). It also can replicate viruses to infect cells, so it has pi. For example set pi = 0.4. If random value < pi then the neighbor cells are removed. Thus, the third neighborhood solution is removed from the neighbor list. Then the infected cells would also have pan. For example pan = 0.2, when the value of random_value>pan, then the cell remains in the neighbor list. If the random_value < pan then the cell is removed from the neighbor list.
7) Phase sub tour neighbor
   In this sub tour, it will cut routes from the neighbor based on the maximum total distance that can be taken, which is 1000.
8) Updates the clinical picture and remove worst solution
9) Then set the type of replication for the new genome

Step 6. Lysogenic Procedure
1) Updating IT, ITi = ITi-1+1
2) If the value of IT > LIT the cell wall will broke and there will be mutation. LIT cells calculated based on the objective function of the cell relative to the objective function that has been found.
3) Do mutation in the initial route 1-4-5-3-2-1 become 1-2-5-4-3-1. Then do the sub tour on a route that is less than max. So, the route becomes 1-2-5-1, and the score is 9.5. Before doing mutation, the score is 8.75, since the mutated cell’s score is greater than the score before, remove initial route with the mutation route. Then set the replication type for this new route.
4) Updating the clinical picture

Step 7 Termination Criteria
Check the termination criteria, if the termination criteria has not been met, then repeat those two procedures, lytic and lysogenic. In this paper we used maximum number of iteration as termination criteria.

5. Experiments and Results
The experiments was run using data set with 27 cities. Before VS was applied to the whole data set, the algorithm first was validated on 5 cities. VS produced the same result as those of enumeration approach. By this result we confirmed that the algorithm is valid and can be applied for the whole data set. In this study, five different vector of weight were used. Wg the weight for 4 different criteria are given below ([4] and [7]):
W0 = [0.25; 0.25; 0.25; 0.25], W1 = [1; 0; 0; 0], W2 = [0; 1; 0; 0], W3 = [0; 0; 1; 0], W4 = [0; 0; 0; 1].
The first vector W0 indicates that for each criteria we applied the same weight. For the other weight vectors we applied binary weight which means only one criteria works for each selected vector. First of all experiments were done to find good parameters which include Plt, Pan, CPS, neighborhood, pi, LNR and LIT. The first experiments was done by changing the value of Plt and Pan with 5 replications. Then, the changes were applied on CPS and type of neighborhood. The Plt and Pan produced high Z indicate good value of parameters. From the experiments on some different combination of parameters we got the following best parameters: Plt = 0.6, pi=0.2, pan = 0.2, pr = 0.2, LNR = 10, and LIT = 5 and a modified type of neighborhood that is by replacing three neighborhood nodes.

Using the best parameters combination Table 1 shows the results of running VS on GOP data set. Compared to Cross Entropy algorithms VS has good performance on computing time. But for the objective function values, the results are not better than CE. Only on W2, VS was able to find solution similar to Cross Entropy. But on the other weight vectors, VS has not been able to find better solutions.

<table>
<thead>
<tr>
<th>Weight</th>
<th>Method</th>
<th>Distance</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>W0</td>
<td>VS</td>
<td>4543.8</td>
<td>11,7139</td>
</tr>
<tr>
<td></td>
<td>CE</td>
<td>4993.4</td>
<td>12,3793</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>4833.5</td>
<td>12,28</td>
</tr>
<tr>
<td></td>
<td>HS</td>
<td>4993.4</td>
<td>12,3793</td>
</tr>
<tr>
<td></td>
<td>ANN</td>
<td>4993.4</td>
<td>12,3793</td>
</tr>
</tbody>
</table>

Table 2 shows the comparisons of VS and other algorithms which are Cross Entropy, Genetic Algorithm, harmony Search and ANN. Again here it shows that VS produced solutions not better than the others.

<table>
<thead>
<tr>
<th>Weight</th>
<th>Method</th>
<th>Distance</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>W3</td>
<td>VS</td>
<td>4926.3</td>
<td>12,540</td>
</tr>
<tr>
<td></td>
<td>CE</td>
<td>4987.5</td>
<td>12,7826</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>4996.8</td>
<td>12,78</td>
</tr>
<tr>
<td></td>
<td>HS</td>
<td>4987.5</td>
<td>12,7826</td>
</tr>
<tr>
<td></td>
<td>ANN</td>
<td>4987.5</td>
<td>12,7826</td>
</tr>
</tbody>
</table>
6. Conclusions
In this paper, Viral Systems was successfully developed to solve Generalized Orienteering Problem. Viral Systems is faster in terms of computational time than CE, but in terms of accuracy the results are worse. For weight vector W2, VS produced objective function value as good as CE. For the same weight vector, VS also produced a better solution than those of Harmony Search, while on the other weight vectors, VS did not produce better solution than those of other algorithms. For future studies, in order to gain better performance, VS algorithm can be modified further on mutation and neighborhood searching processes. By these steps, it might produce more varied solutions and it will not stuck in the local optima. The combination (hybrids) with other algorithms is also possible way to improve the performance.

References