Solving Hub Location Problem with Genetic Algorithm

Aydin Ozmen
Turkish Air Force Academy
Computer Engineering Department
Yesilyurt, 34149, Istanbul, Turkey
Email: aydin.ozmen@tubitak.gov.tr

Ozgur Koray Sahingoz
Turkish Air Force Academy
Computer Engineering Department
Yesilyurt, 34149, Istanbul, Turkey
Email: sahingoz@hho.edu.tr

Abstract—In this paper, we present an effective solution using the genetic algorithm for the uncapacitated single allocation hub location problem (USAHLP). This solution was published before, but today it still gives the really effective results, although there are many studies have been done after it. We also make an application using this method and present the performance results of this method.

I. INTRODUCTION

Transportation is very important issue for telecommunication and transportation networks such as passenger airlines, express package delivery firms, message delivery networks etc. Because the commodities (data transmissions, passengers, express packages, mail, etc.) have to be delivered from origin to destination using the cheapest and easiest way. In large networks, this transport process is very difficult because of long distance between origin and destination. So some other transshipment points are needed on this long way between origin and destination. Problem is, for the cheapest and the most effective transportation, where these transshipment points (hubs) are located on network. Hub Location Problem is concerning about this issue.

Hub Location Problems (HLP) are kind of optimization problems and the main purpose is how the cost must be lowest using the transshipment points. Selecting the place of these points on the network is very important and using various methods or some algorithms for this. Therefore, transshipment points play an important role reducing the cost. The transshipment points are called hub; destination and origin points are called spoke. Therefore the whole network is called hub-spoke network.

Hub-spoke networks have application in many areas such as passenger airlines [1], express package delivery firms [2], message delivery networks [3], telecommunication systems [4], supply-chain of chain stores etc. All these applications are commercial and widespread. So there are lots of works have been done about HLPs in literature.

There are different varieties of HLP in literature [5]. These different varieties were determined according to the constraints such as the way of selecting the number of hubs to be located, the way the spokes are assigned to hubs, the existence of capacity limits on hubs, etc. But for all these varieties of HLPs main purpose is locating the hubs on network and allocating the nodes to right hubs with minimizing the cost.

In the Single Allocation Hub Location Problem (SAHLP), a spoke is allocated to exactly one hub and the number of hubs to be used is not known in advance. There are two types of SAHLP. If the hubs number is capacitated and known, this problem is called Capacitated Single Allocation Hub Location Problem (CSAHL). The Un-capacitated Single Allocation Hub Location Problem (USAHL) involves hubs with unlimited capacities. In this problem, the hub number is decision variable. If the hub number is fixed, this problem is called Uncapacitated Single Allocation pHub Median Problem (USApHMP). Also if a spoke is allocated more than one hub, the problem is called Un-capacitated Multiple Allocation pHub Median Problem (UMApHMP).

Genetic Algorithms (GA) are optimization methods which are inspired from theory of evolution and using heuristic search. There are lots of implementations to many combinatorial optimization problems including the SAHLP [5]. Especially, on USAHLP, the GAs give the best solution found in the literature. But on the other types of SAHLP, GAs are not as successful as USAHLP. One of the best and simple solution of USAHLP in the literature is the “Solving the uncapacitated hub location problem using genetic algorithms” by H. Topcuoglu in 2005 [6]. So we select and analyze this method for our paper.

II. BACKGROUND

In the Hub-Spoke systems, transshipment centers which are also called hubs using for switch or sort the commodities or decide where the commodities go after from this point. The other points excluding hubs are called spokes which are including origin and destination nodes. The main purpose of this system is to access the all nodes on network. The easiest way of this, connect the every couple of nodes with each other. But this is not an efficient method because of the total cost. For efficiency, there is a need for the hubs. An example of a hub-spoke network is shown in Figure 1. In this figure, nodes i, j, k, and I are hubs whereas the rest of the nodes are spokes and spokes are connected only hubs, not connected each other. But the some nodes are connected to two hubs. So this is not a single allocation network.
A. Single Allocation Hub Location Problem (SAHLP)

The Single Allocation Hub Location Problem is a type of HLP in which a spoke can be assigned to only a single hub. In SAHLP, to determine the hub number is an important issue. Therefore, the number of hubs is a decision variable for SAHLP. The other problem for SAHLP is allocation of spokes to hubs. There must be a decision mechanism for spoke-hub assigning process.

All these decisions for how to minimize the total cost of network. A SAHLP example is shown in Figure 2. From origin spoke i to destination spoke j flow is shown as $W_{ij}$ and total cost of this flow consist of three costs:

1. Collection Cost ($\chi$): It is a spoke to hub flow (i to k) and showed as $W_{ik} = \chi C_{ik}$.
2. Transfer Cost ($\alpha$): It is a hub to hub flow (k to l) and showed as $W_{kl} = \alpha C_{kl}$.
3. Distribution Cost: It is a hub to spoke flow (l to j) and showed as $W_{lj} = \delta C_{lj}$.

Therefore total cost is:

$$C_{iklj} = \chi C_{ik} + \alpha C_{kl} + \delta C_{lj}$$

This is cost of flow of from i to j. But for all network total cost, this formula has to be calculated for every couple of nodes.

There are two types of SAHLP found in literature: The uncapacitated Single Allocation Hub Location Problem (US-AHLP) and the capacitated Single Allocation Hub Location Problem (CSAHLP). In the USAHLP, there is no limit for the assigned spoke number to one hub while in the CSAHLP, the number of assigned spoke is fixed to one hub. In this work, we study on USAHLP. Also we use the O’Kellys formulation for USAHLP. According to this formula the integer decision variable $X_{ik}$ is 1 if node i is allocated to the hub located at node k, and it is equal to 0 otherwise. Each hub is assigned to itself. With these definitions, USAHLP is stated as follows (n is node number and V is a set of node numbers):[6]:

$$min f(x) = \sum_{i} \sum_{j} W_{ij} + \sum_{i} \sum_{j} W_{ji} - \sum_{i} \sum_{k} \sum_{l} X_{ik} X_{jl}$$

$$+ \sum_{i} \sum_{k} X_{ik} \sum_{j} X_{jl} (\chi O_{i} + \delta D_{j}) +$$

$$\sum_{j} X_{ii} f_{j}$$

(2)

$$\sum_{k} X_{ik} = 1 \forall i \in V$$

(3)

$$X_{kk} - X_{ik} \geq 0 \forall i, k \in V$$

(4)

$$X_{ik} \in 0, 1 \forall i, k \in V$$

(5)

The equation (2) the total cost that includes collection, transfer, distribution and the costs of the each hub. The (3) equation is shown that each node is assigned to exactly one hub. This is constraint of USAHLP. The fourth one is about that node to hub assignment. But for this model there is not accepted spoke to spoke assignment. The last one is shown that the value set of assignment. [6].

Total amount of originating and terminating are:

$$O_{i} = \sum_{j} W_{ij} and D_{i} = \sum_{j} W_{ji}$$

(6)

According to all these informations and constraints the objective function is written as:

$$f(x) = \sum_{i} \sum_{k} X_{ik} C_{ik} (\chi O_{i} + \delta D_{i}) +$$

$$\sum_{i} \sum_{k} X_{ik} \sum_{j} X_{jl} (\alpha C_{kl}) W_{ij} + \sum_{j} X_{ii} f_{j}$$

(7)

B. GA-Formulation of the Problem

Genetic Algorithm (GA) is search algorithm which is inspired from theory of evolution and using heuristic search. It introduced by James Holland in his seminal work "Adaptation in natural and artificial systems" in 1975. The workflow of GA consists of 6 steps:

1. Initialize Population
2. Compute Fitness
3. Selection
4. Crossover
5. Mutation
6. Local Optimization

We formulate our problem according to these steps. Firstly, for Initialize Population, we need the population format which also called chromosome in GA. In our approach, GA chromosome consists of two arrays: HubArray and AssignArray. HubArray is only got two values: 1 and 0. 1 is for nodes and 0 is for spokes. The AssignArray is using for assignments of the spokes to hubs. If a spoke is assigned to hub, this hubs value is written to AssignArray.

1) Initialize Population: After determination of chromosome, the population has to be selected. The steps of initialization are below:

- There are two initial populations are determined for this problem. First one is %75 of initial population; second one is %25 of initial population. Firstly, the %75 of hubs are selected from all hubs randomly. This is first hub set and the remaining %25 is second hub set. Then a number is selected from the range [1, n/4] randomly and the selected number is set to any hub of first hub set (n is denoted total number of nodes). This process is repeated for the every hub in the first hub set. For the second hub set, a number is selected from the range [n/4, n/2] randomly and selected number is set to any hub of second hub set. Thus, the every hub is had a number. These hubs are determined in the HubArray.

- After setting number process, the hubs are selected from HubArray. For this process the flow amount of each nodes is using. The flow amount is consist of total amount of traffic originated by \((O_i)\) and destined for \((D_i)\) node. So it is equal to \(O_i + D_i\). For every nodes, this value is calculated and then nodes are sorted a list (flow list) in decreasing order of total flows. For %75 of the initial population, the hubs in the HubArray are selected from two thirds of the nodes from the flow list starting from the node with the highest total flow. This strategy provides the node which has higher amount of flow to become a hub with higher probability. For the remaining %25 of the population, hubs are selected from the whole set of nodes. These percentage values were determined with a set of experiments for specifying the control parameters of GA.

2) Compute Fitness: Fitness value is calculated using the equation (2).

3) Selection: Proportional selection (with roulette) is used for selection process. Also elitism is used for the new generations.

4) Crossover: Crossover process is applied to HubArray and AssignArray with the same method. Firstly, the crossover point is selected from input strings of parent arrays and then the childs are generated the right and left parts of parents. In Figure 3, there is shown an example crossover:

Sometimes there is some adjustments needed for the children.

5) Mutation: Mutation process is applied for the assignments of nodes. There are two types of mutation methods using for this problem:

- **Shift:** Firstly the spoke is selected and the selected spoke is reassigned to another hub which is selected randomly. If there is only one hub in the string, this mutation operator is not applied. (Figure 4).

- **Exchange:** The selected two spokes which are assigned to different hubs are switched their assignment. (Figure 5).

6) Local Optimization: This process is not used for our method.

C. Related Works

Originally, Hakimi (1964) [7] published the first paper in the area of node optimality which was motivated by similar concepts to the Hub Location Problem. But the first paper about the HLP was published by OKelly in 1987 [8]. He formulated the HLP problem as a quadratic integer form. After this work, the researchers mostly studied on the uncapacitated single allocation p-hub median problem (USApHMP) such as OKelly [9], Smith [10]. Later, in 1994 Campbell [11] proposed multiple mathematical formulations for HLP. Moreover, Aykin [12] and Klincewicz [13] had important roles about this field. In 2008, Alumur and Kara [5] classified all works about HLP until 2007. The recent years, there are lots of works have been published. Therefore, the old classification [5] is not
The GA algorithm was used on HLP firstly by Abdinnour-Helm [15]. He used the Tabu Search and GA together (GATS) for USAHLP. After this work, in 2005, Topcuoglu [6] developed a GA-based approach to the USAHLP and he found the best solutions to some Civil Aviation Board (CAB) problems. He also used Australian Post (AP) data in their experiments that had not been previously used in any study on USAHLP. Also in 2007, Cunha and Silva [16]; in 2009, Filipovic, Kratica, Tosic, and Dugosija [17] used GA for USAHLP. These works show the good effects of GA on USAHLP. But the Topcuoglus method [6] is really effective and simple. Even today, it is the one of the algorithms which are given the best results. Therefore, we use this method for our paper. There are also a lot of works have been published about CSAHLP with GA in the literature.

D. Computational Study

After these explanation about the selected method, we present the results and computations on selected problem area in this section. We show two metrics about our work. First one is the total cost that given by equation (2) and the second one is the selected hubs for the best solution.

We use the CAB Data Set [8] for the problem area. We coded the algorithm in C# programming language and the performance results performed on the PC which has Intel I7 2.40 Ghz processor with Windows 8 Operating System.

1) CAB Data Set: The CAB data set comes from The Civil Aeronautics Board, USA and is one of the most commonly used data set for hub location problems. It has 25 cities in 1970 for airline passenger flow. On CAB data set, the flows between cities are symmetric.

In our referenced study [6], the CAB data set is classified four problem sizes which are 10, 15, 20 and 25 and each problem sizes have different five discount factors for transfer cost ($\alpha \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$), but for collection and distribution cost are keeping 1 ($\chi = \delta = 1$). Also the fixed cost given as $f \in \{100, 150, 200, 250\}$.

There are also some probability values given for mutation and crossover. So mutation and crossover processes are applied only for some generations. These probabilities are given $Pm = 0.4$ and $Pc = 0.7$. For the best solution, the population size is selected 200 ($P = G = 200$).

2) Algorithm: The algorithm have three important parts. First one is initial population, second one is mutation and the last one is crossover. These processes are applied to problem size. Here, n is the problem size, and p is the number of hubs, $Pc$ is crossover probability and $Pm$ is mutation probability.

You can see our algorithm’s pseudo code below:

GA Algorithm:

Procedure Initial Population(n)

while ((convergence is not achieved) and (Number of Generation ≤ GenerationSize)) do

for i = 1 to PopulationSize/2 do

Randomly select two chromosomes: S1; S2

Procedure Crossover($S_1; S_2; S_1^2; S_2^2$)

Procedure Mutation($S_1^1$)

Procedure Mutation($S_2^2$)

Insert offsprings into the new generation

endfor

Apply elitism for the new generation

endwhile

Report the best chromosome as the final solution

Initial Population(n):

Compute the total amount flow of each node, which is $Flow_k = O_k + D_k$.

Sort the nodes in decreasing order of $Flow_k$ values in a list, flowlist

Hub-Set = $\{n_k|n_k = flowlist[i] \forall i, 1 \leq i \leq \frac{7}{9} \times n\}$

for i=1 to ($\frac{7}{9} \times PopulationSize$) do

$p \leftarrow random\left[1, \frac{7}{9}\right]$

Select p hubs from Hub-Set, randomly.

Assign each spoke to the nearest hub based on distance values.

endfor

for i = (\frac{7}{9} \times PopulationSize) + 1 to PopulationSize do

$p \leftarrow random\left[\frac{7}{9}, \frac{2}{3}\right]$

Select p hubs from all nodes randomly.

Assign each spoke to the nearest hub based on distance values.

endfor

Procedure Crossover($S_a, S_b, S_a^c, S_b^c$):

If $random(0, 1) \leq P_c$ then

Select crossover point randomly.

Apply single-point crossover on HubArray and AssignArray of $S_a$ and $S_b$ to procedure $S_a^c$ and $S_b^c$.

If (p=0 or p=n) in ($S_a^c, S_b^c$) then

Discard $S_a$ and $S_b$ and repeat the selection step.

For each spoke $n_k$ in ($S_a^c, S_b^c$) assigned to non-hub do

Re-assign $n_k$ to the nearest hub in the corresponding offspring

endif

else $S_a^c = S_a$ and $S_b^c = S_b$

Procedure Mutation($S_c$):

If $random(0, 1) \leq P_m$ then

Select a spoke $n_k$ at random.

Assign $n_k$ to another hub selected at random.

Call this solution $S_b^c$. // result of shift move

Select two spokes $n_x$, $n_y$ (that are assigned to different hubs) at random.

Switch the assignments of $n_x$ and $n_y$

Call this solution $S_c^c$. // result of exchange move

If (shift or exchange not applicable) then

Return $S_c$

else

Return best of ($S_c^c, S_b^c$) with respect to fitness values

endif

else Return $S_c$
TABLE I
RESULTS FOR N=10

<table>
<thead>
<tr>
<th>α</th>
<th>f</th>
<th>GA cost</th>
<th>GA hubs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>100</td>
<td>321659494.30636</td>
<td>1,2,3,4,9,10</td>
</tr>
<tr>
<td>0.4</td>
<td>100</td>
<td>350615499.74388</td>
<td>1,3,4,6,7,8</td>
</tr>
<tr>
<td>0.6</td>
<td>100</td>
<td>465501276.96272</td>
<td>1,3,4,6,7,8</td>
</tr>
<tr>
<td>0.8</td>
<td>100</td>
<td>580387054.18156</td>
<td>1,3,4,6,7,8</td>
</tr>
<tr>
<td>1.0</td>
<td>100</td>
<td>695272831.4004</td>
<td>1,3,4,6,7,8</td>
</tr>
</tbody>
</table>

TABLE II
RESULTS FOR N=15

<table>
<thead>
<tr>
<th>α</th>
<th>f</th>
<th>GA cost</th>
<th>GA hubs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>100</td>
<td>825452706.63796</td>
<td>1,2,3,4,7,8,9,10,12,15</td>
</tr>
<tr>
<td>0.4</td>
<td>100</td>
<td>1686265612.44888</td>
<td>1,2,4,8,9,11,14</td>
</tr>
<tr>
<td>0.6</td>
<td>100</td>
<td>1753577850.98156</td>
<td>1,2,4,6,9,10,11,12,15</td>
</tr>
<tr>
<td>0.8</td>
<td>100</td>
<td>1911995193.88984</td>
<td>2,3,4,5,6,12,13,14</td>
</tr>
<tr>
<td>1.0</td>
<td>100</td>
<td>2575947661.7142</td>
<td>1,3,4,9,11,13,15</td>
</tr>
</tbody>
</table>

TABLE III
RESULTS FOR N=20

<table>
<thead>
<tr>
<th>α</th>
<th>f</th>
<th>GA cost</th>
<th>GA hubs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>100</td>
<td>2078946035.17276</td>
<td>2,3,4,6,7,8,9,17,18,20</td>
</tr>
<tr>
<td>0.4</td>
<td>100</td>
<td>3356813011.30016</td>
<td>1,2,4,5,7,8,11,14,15,19</td>
</tr>
<tr>
<td>0.6</td>
<td>100</td>
<td>3535264464.04736</td>
<td>1,2,3,4,6,8,12,14,15,17,19,20</td>
</tr>
<tr>
<td>0.8</td>
<td>100</td>
<td>4338834246.40456</td>
<td>1,2,3,4,5,7,11,12,13,14,17,20</td>
</tr>
<tr>
<td>1.0</td>
<td>100</td>
<td>5445476448.6246</td>
<td>1,2,3,4,5,7,10,11,14,16,17,18</td>
</tr>
</tbody>
</table>

TABLE IV
RESULTS FOR N=25

<table>
<thead>
<tr>
<th>α</th>
<th>f</th>
<th>GA cost</th>
<th>GA hubs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>100</td>
<td>2774798178.08988</td>
<td>1,2,3,5,6,7,9,10,13,15,16,22,23,25</td>
</tr>
<tr>
<td>0.4</td>
<td>100</td>
<td>4497231823.99708</td>
<td>1,3,4,6,7,8,9,15,16,17,22,23,24,25</td>
</tr>
<tr>
<td>0.6</td>
<td>100</td>
<td>6594295551.67256</td>
<td>1,6,9,11,14,15,17,18,19,22,23,25</td>
</tr>
<tr>
<td>0.8</td>
<td>100</td>
<td>8845278621.52234</td>
<td>1,5,6,9,15,17,19,20,21,22,25</td>
</tr>
<tr>
<td>1.0</td>
<td>100</td>
<td>10446420420.87923</td>
<td>1,2,4,7,8,17,18,19,22,23,25</td>
</tr>
</tbody>
</table>

3) Results: We generated our method with 10, 15, 20, 25 problem sizes. So for CAB data set, we selected 10, 15, 20, 25 cities. Tables 1-4 gives the GA.Cost and GA.hubs results according to transfer cost and fixed cost values. Table 1 gives the results for 10 cities, Table 2 gives the results for 15 cities, Table 3 gives the results for 20 cities and Table 4 gives the results for 25 cities for different transfer costs.

III. CONCLUSION

This paper presents an efficient method using the genetic algorithm for the uncapacitated single allocation hub location problem (USAHLHP). This method was published by H. Topcuoglu in 2005 [6]. We coded this algorithm in C# and we present the performance results of this method on CAB data set for fourth different problem sizes.

According to these performance results, this referenced method still gives an effective solution for the USALP.

REFERENCES


