INTEGRATION OF GIS AND OPTIMIZATION Routines for the Vehicle Routing Problem

Takwa Tlili¹, Sami Faiz² and Saoussen Krichen³

¹LARODEC Laboratory, High Institute of Management, Tunisia
takwa.tlili@gmx.fr
²LTSIRS Laboratory, National engineering school of Tunis, Tunisia
sami.faiz@insat.rnu.tn
³LARODEC Laboratory, High Institute of Management, Tunisia
saoussen.krichen@isg.rnu.tn

ABSTRACT
This paper evokes the vehicle routing problem (VRP) which aims to determine the minimum total cost pathways for a fleet of heterogeneous vehicles to deliver a set of customers' orders. The inability of optimization algorithms alone to fully satisfy the needs of logistic managers become obvious in transportation field due to the spatial nature of such problems. In this context, we couple a geographical information system (GIS) with a metaheuristic to handle the VRP efficiently then generate a geographical solution instead of the numerical solution. A real-case instance in a Tunisian region is studied in order to test the proposed approach.

KEYWORDS
Vehicle routing problem; Dijkstra-based approach; Geographical information system.

1. INTRODUCTION
In industrial companies, applications that involve a distribution network are required to be illustrated as a map. For this reason, the use of the GIS is recommended but this technique cannot evoke the optimization aspect of the distribution problems. Also, the optimization is unable to result a map. Vehicle Routing Problems (VRP) introduced by Dantzig et al. (1959), are the basic problems in the vehicle routing class. They have been extensively studied since the sixties and have received the greatest attention in the scientific literature (i.e. Ai et al, 2009, Cheang et al., 2012, and Riera-Ledesma and Salazar-Gonzalez, 2012). In the VRP, all the customers correspond to deliveries and the deterministic demands cannot be split. The vehicles are identical and based at a single central depot, and only the capacity constraints for the vehicles are imposed. The CVRP, considered as the standard version of the VRP (Chen et al. (2010), Szeto et al. (2011) and Marinakis (2013)), is applied in different areas such as fuel consumption optimization (Xiao et al., 2012) and school bus routing problems (Riera-Ledesma and Salazar-Gonzalez, 2012). According to Toth et al. (2002), the largest problems that contain about 50 customers can be consistently solved by the most effective exact algorithms. Over this number of customers the VRP may only be solved with heuristic methods. Basically, three main integration strategies were proposed in the literature (Karimi and Houston, 1996, Faiz and Krichen, 2013) to combine optimization routines in GIS. We can point out, the loose integration, the tight integration and the full integration.

The present paper is organized as follows: section 2 describes in details the Dijkstra-based PSO approach for solving the VRP. Section 3 is a statement of a real-life VRP application in Ezzahra-Tunisia as well as the resolution of the proposed instance.
2. **The Dijkstra-based PSO approach**

To solve the VRP using the dataset previously generated, we apply the Dijkstra-based PSO (DPSO) approach. The PSO algorithm is applied to load the minimum number of vehicles and the dijkstra algorithm is adapted to handle the routing process.

2.1. **Particle swarm optimization concepts**

Particle swarm optimization (PSO) is a population-based evolutionary algorithm simulating the social behavior of bird flocking. The PSO initializes a population of random particles, every particle is considered as a potential solution. Firstly, particles fly spontaneously through the problem space at a random velocities. Next, velocities are updated based on the best previous particle's experience and the best previous group's experience, i.e. the behavior inside a population is a compromise between individual and collective memory. The PSO script is described as follows.

Given a $d$-dimensional search space and a swarm of $S$ particles ($p = 1, \ldots, S$). To each particle $p$ in generation $t$ corresponds a position-vector (1) and a velocity-vector (2).

$$ X_p^t = (x_{p1}^t, x_{p2}^t, \ldots, x_{pd}^t) \quad (1) $$

$$ V_p^t = (v_{p1}^t, v_{p2}^t, \ldots, v_{pd}^t) \quad (2) $$

where $x_{pn}^t$ represents the location and $v_{pn}^t$ is the flying velocity of particle $p$ in generation $t$ in the $n$th dimension of the search space ($n = 1, \ldots, d$).

Particles memorize every reached position and save the one with the best fitness. This individual best position is denoted by $B_p^t = (b_{p1}^t, b_{p2}^t, \ldots, b_{pd}^t)$. In each iteration, particles record the whole best position's fitness until generation $t$. This global best position is referred to as $G^t = (g_1^t, g_2^t, \ldots, g_d^t)$. The new velocity and position are updated, respectively, by those following formulas:

$$ v_{pn}^t = w v_{pn}^{t-1} + c_1 r_1 (b_{pn}^{t-1} - x_{pn}^{t-1}) + c_2 r_2 (g_n^{t-1} - x_{pn}^{t-1}) \quad (3) $$

$$ x_{pn}^t = x_{pn}^{t-1} + v_{pn}^t \quad (4) $$

$$ p = 1, \ldots, S \quad ; \quad n = 1, \ldots, d \quad ; \quad t = 1, \ldots, T \quad (5) $$

Where $t$ is the iteration counter ($t = 1, \ldots, T$); $w$ is the inertia weight that controls the influence of the precedent velocity on the current velocity; $r_1$ and $r_2$ are two random variables uniformly distributed in [0, 1]. The acceleration coefficients, referred to as $c_1$ and $c_2$, control how far a particle will move in a single iteration.

The pseudo-code of PSO (Perez et al., 2007 and Van den Bergh et al., 2006) can be stated as follows.

**Algorithm 1:** PSO algorithm

1: Initialize population ;
2: for $t := 1$ to $T$ do
3: for $s := 1$ to $S$ do
4: Evaluate Fitness ($X_p^t$)

10
5: Select the individual best position $B^*_j$;
6: Select the global best position $G^*$;
7: Update Velocity using Equation (3);
8: Update Positions using Equation (4);
9: end for
10: end for

2.2. Dijkstra algorithm concepts

Dijkstra's algorithm is a well known approach for solving shortest path problems. This algorithm can be described as an iterative procedure which starts from a source node. In each iteration, another vertex from the graph is added to the shortest-path. The pseudo-code of Dijkstra can be stated as follows.

Algorithm 2: Dijkstra algorithm

1: Initialize the cost of each node to 1
2: Initialize the cost of the source to 0
3: while the queue of nodes is not empty
4: Select the unknown node with the lowest cost, node*
5: Mark node* as known
6: for each node n which is adjacent to node*
7: cost(n) = Min (old-cost(node*), cost(node*)+cost(node*, n))

3. The VRP applied to Ezzahra-Tunisia

3.1 The studied area description

Ezzahra is a coastal town on the outskirts of Tunis located six kilometers south of the capital and covers 750 hectares. It is bounded by the Mediterranean Sea and the municipalities of Rades, Hammam Lif and El-Bou Mhel Bassatine. Administratively attached to the governorate of Ben Arous, it is the seat of a delegation and a municipality of 31 792 inhabitants (2006) whereas the city itself has a population of 6000 inhabitants.

3.2 The used GIS tool

We used Quantum GIS (QGIS) that is an Open Source GIS that runs on Linux, Unix, Mac OS X and Windows. It is released under the GNU General Public License (GPL). QGIS aims to be an easy-to-use GIS, providing common functions and features. The initial goal was to provide a GIS data viewer. QGIS has reached the point in its evolution where it is being used by many for their daily GIS data viewing needs. It also supports a number of raster and vector data formats.

Numerous features can be provided by this tool including the following:
- View and overlay vector and raster data in different formats and projections without conversion to an internal or common format
- Explore spatial data with a friendly GUI and compose maps
- Create, edit, manage and export vector maps in several formats.
- Analyze data
3.3 The dataset

To generate the map of Ezzahra, we used QGIS 1.4 under Linux. The input is a dataset which consists on the descriptive data (in .dbf format) and the spatial data (in .shp format). As shown in figure 1 we present mainly roads, sea, islets and buildings of this town.

Figure 1. The Ezzahra (Tunisia) city map created by QGIS

In order to solve the VRP in this context, we enrich our data with a new layer called Customer that contains nine clients and one depot. In figure 2 we can observe the dispersion of customers on the chosen area.

Figure 2. The dispersion of customers on Ezzahra (Tunisia) territory

The available database of the studied area contains a layer of roads that does not allow the movement from customer to another. So, we create a new vector layer expressing the transport
network consisting on a set of arcs connecting the set of customers. This layer represents the roads by which a vehicle can deliver items to a set of customers. Figure 3 shows the positions of customers onto the transport network.

Figure 3. The transport network

3.4 SOLVING THE VRP APPLICATION

The VRP is an optimization problem in which objects, already stored in depots, are required to be loaded into vehicles, then delivered to some geographically dispersed selling points all over the Tunisian territory, in order to fulfill known customer requests.

The scenario of VRP is described below:

The choice of the cost saving vehicles: That depends on the total distance and the quantity of freight.

The cargo loading: In a single depot, a set of orders has to be charged in each vehicle.

The cargo delivery: According to orders packed in the vehicle, this latter has to visit the corresponding customers.

Before stating the mathematical formulation, we have to introduce some key terms of this problem:

-A box: It is a parallelepiped object that contains an order or a part of an order. Each box is characterized by a weight and a volume.
-An order: It corresponds to only one customer and each order is packed into one or more boxes.
-An invoice: It includes the type of products, the corresponding quantity and the customer order.
-A vehicle: It is characterized by a cost, a capacity and a volume.
-A warehouse: It is a central depot that contains the orders already packed into boxes.

Thereby, the problem is about planning the loading of every vehicle then its round in an
optimal way that minimizes the total cost.

- Every order has to be carried by exactly one vehicle.
- The capacity of each vehicle should be respected.
- The total volume of orders packed in a vehicle should be in the volume interval of that vehicle.
- Each vehicle should not exceed a prefixed distance and can not accept a path less than a minimum threshold.

Our motivation behind using the QGIS is to use a real-case while performing distances between customers. There is an additional plugin integrable on QGIS called ROAD GRAPH (Figure 4) that calculates the shortest path between two points on any polyline layer and plots this path over the road network.

![Figure 4. The ROAD GRAPH plugin](image)

By means of ROAD GRAPH we denote the shortest paths between each pair of customers. For example, in figure 5 the red line represents the shortest path of transport network between C1 and C2.

![Figure 5. The detection of the shortest path](image)

In the same manner as shown in figure 5, we denoted each shortest distance between every couple of customers. The result is a distance-matrix illustrated in figure 6. The matrix consists on round distances in meter.
Our objective is to minimize the total cost. The solution of our application is modeled on two parts. The loading of orders and the routing of vehicles.
Figure 7 illustrates the distribution of the orders on vehicles while respecting the capacity of weight, the maximum volume and distance of each vehicle.

Figure 8 is the map that guides the driver of vehicle 1 to deliver the loaded orders to corresponding customers. The yellow arrows form the shortest itinerary to distribute the cargo.

For the travel of the vehicle 1, the cost is equal to $149.28DT$ in the following representation we can observe the path with the cost in each arc.

\[
\text{Depot} \quad 37.22 \to C2 \quad 37.05 \to C1 \quad 37.45 \to C4 \quad 37.56 \to C5
\]

Figure 9 is the map of the second vehicle that begins from the depot point and travels through the optimal itinerary in terms of cost. While observing the following figure, we obtain a cost of travel equal to $186.73DT$.

\[
\text{Depot} \quad 37.48 \to C3 \quad 37.37 \to C5 \quad 37.29 \to C6 \quad 37.12 \to C7 \quad 37.58 \to C9
\]

Figure 9: The routing map of vehicle 1

Figure 10: The routing map of vehicle 2
3. CONCLUSIONS

We introduced in this paper the vehicle routing filed and its relation with GIS techniques. To apply our elaborated approach in a real-case, we used GIS tool for purposes of precision. So we presented in the third section a detailed description of the studied area, the GIS tool, the dataset and the scenario of the routing process of a company. Finally, we solved our application using the DPSO approach, then we reported the solution consisting in two vehicle maps which indicate the optimal itinerary.

REFERENCES


