Improving travels of the public transport system of Guadalajara using ACO

[1]Edgar Gonzalo Cossio Franco, [2]José Alberto Hernández Aguilar, [3]Alberto Ochoa-Zezzatti & [4]Julio Ponce Gallegos

[1]Universidad Da Vinci/Universidad Enrique Díaz de León
[2]Universidad Autónoma del Estado de Morelos
[3]Universidad Autónoma de Ciudad Juárez
[4]Universidad Autónoma de Aguascalientes

kofrran@gmail.com, alberto_doctor@hotmail.com, alberto.ochoa@uacj.mx, julk_cpg@hotmail.com

Abstract. This paper presents the application of ant algorithm to improve public transport route in the metropolitan area of Guadalajara.

Keywords: C++, Instances, Traffic, Ant Colony Optimization

1. Introduction

In the last population census of 2010, conducted by the INEGI, identified in the area of mobility growth rates in terms of vehicle fleet it has been increasing: the rate of motorcycles increased by 30% annually between 1990 and 2010, from 16,000 to 177,000. In the same period, the rate grew to 7.31% cars, and trucks and cargo vans to 7.66%. The lower rate is that of passenger buses -3.49% - certainly closer to the growth of housing in the state. ZMG concentrated two-thirds of the vehicle fleet with 65%.
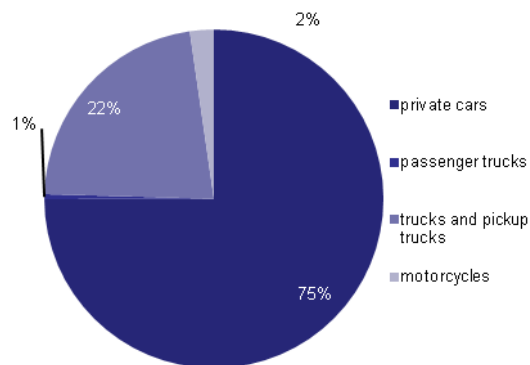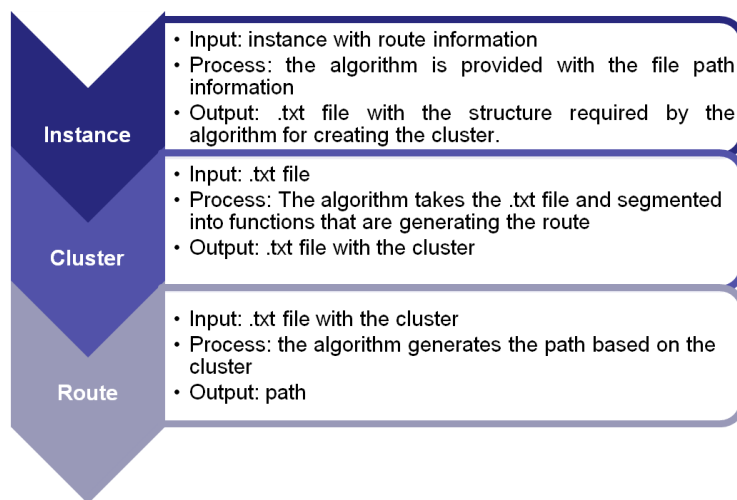

In the figure above 4 types of transport are shown. It is identified as the predominant percentage use of private cars and the least dominant is represented by passenger trucks.

2. General objective

Implement ACO to a problem of public transport in the metropolitan area of Guadalajara with the aim of improving travel

3. Process

The total distance is calculated, the number of routes for each execution and the number of vehicles, with the inputs 25 cities (nodes).

Instance
- Input: instance with route information
- Process: the algorithm is provided with the file path information
- Output: .txt file with the structure required by the algorithm for creating the cluster.

Cluster
- Input: .txt file
- Process: The algorithm takes the .txt file and segmented into functions that are generating the route
- Output: .txt file with the cluster

Route
- Input: .txt file with the cluster
- Process: the algorithm generates the path based on the cluster
- Output: path

The process consists of three elements; an instance as input, clusters are created and finally the route is created.

4. Experiment

The following data is used in the experiment
- 33 units
- 71 drivers
- 37 points
- Daily average per unit:
    - Passengers: 528
    - Laps: 4
    - Time of turns: 50 minutes

The experiment consists of:
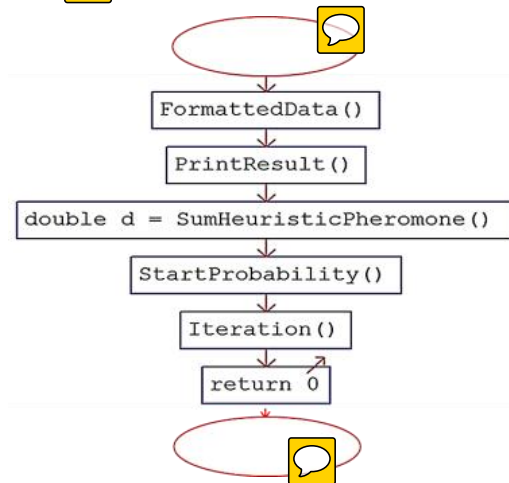1. Identify points along the route

trip	number	punto	lat	long	X	Y
	1	Base 214	20.63128	-103.272	95	73
	2	Farm GDL	20.63564	-103.276	92	68
	3	Rio Nilo	20.63904	-103.277	92	63
	4	Curva Malecon	20.64367	-103.278	92	56
	5	Templo Malecón	20.64762	-103.274	94	50
	6	Pizzería 61	20.65312	-103.28	90	43
	7	Bolería	20.6552	-103.294	80	40
	8	Ramón Val-Pensador	20.65748	-103.309	70	37
g	9	Arena Jalisco	20.66469	-103.317	64	26
o	10	Cuartel colorado	20.67049	-103.334	52	18
i	11	Zapatería prado	20.67298	-103.342	46	15
n	12	Pque. Sn. Fco.	20.67357	-103.348	42	14
g	13	8 de julio- la paz	20.67216	-103.353	38	16
	14	Paztolsa	20.66566	-103.36	34	26
	15	Lab. Piza	20.66386	-103.365	30	28
	16	Retorno 61	20.658	-103.378	20	36
	17	Isla Gomer	20.63682	-103.392	11	66
	18	Reyes Heroles	20.62875	-103.391	12	77
	19	Zansibar-Imss	20.62631	-103.391	11	82
	20	Contestación	20.62345	-103.4	6	84

trip	number	punto	lat	long	X	Y
	1	contestación	20.62345	103.3999	6	84
	2	Zansibar-Imss	20.62631	-103.391	11	82
	3	Reyes Heroles	20.62875	-103.391	12	77
	4	Isla Gomer	20.6368	-103.392	11	66
	5	Retorno 61	20.65792	-103.377	20	36
	6	Enrique D/Washing	20.66463	-103.364	32	33
r	7	Paztolsa	20.66651	-103.359	34	26
e	8	Pque. Sn. Fco.	20.67286	-103.347	44	22
t	9	Med 56	20.67099	-103.341	48	25
u	10	Cv Ruíz Sanchez	20.66268	-103.32	63	37
r	11	Vinos la playa	20.65814	-103.308	72	42
n	12	Mercado Osos	20.65281	-103.294	82	49
	13	Templo Malecón	20.64762	-103.274	94	50
	14	Subida malecón	20.64169	-103.277	94	65
	15	Santa Rosalia- Mal	20.63678	-103.279	93	72
	16	Farm GDL	20.63516	-103.275	92	68
	17	Base 214	20.63128	-103.272	95	73

2. Identify the number of passengers moving from one point to another

Number	X	Y	Demand
1	95	73	14399
2	92	68	9763
3	92	63	9892
4	92	56	5029
5	94	50	16833
6	90	43	23515
7	80	40	25165
8	70	37	13695
9	64	26	30269
10	52	18	12624
11	46	15	9037
12	42	14	21949
13	38	16	30900
14	34	26	29200
15	30	28	31100
16	20	36	33461
17	11	66	6754
18	12	77	5020
19	11	82	4794
20	6	84	0

Number	X	Y	Demand
21	6	84	0
22	11	82	45261
23	12	77	16684
24	11	66	48247
25	20	36	147
26	32	33	59589
27	34	26	40230
28	44	22	36416
29	48	25	20500
30	63	37	16673
31	72	42	21461
32	82	49	10546
33	94	50	16833
34	94	65	10300
35	93	72	7200
36	92	68	928
37	95	73	0

3. Apply the algorithm

```
FormattedData()
PrintResult()
double d = SumHeuristicPheromone()
StartProbability()
Iteration()
return 0
```
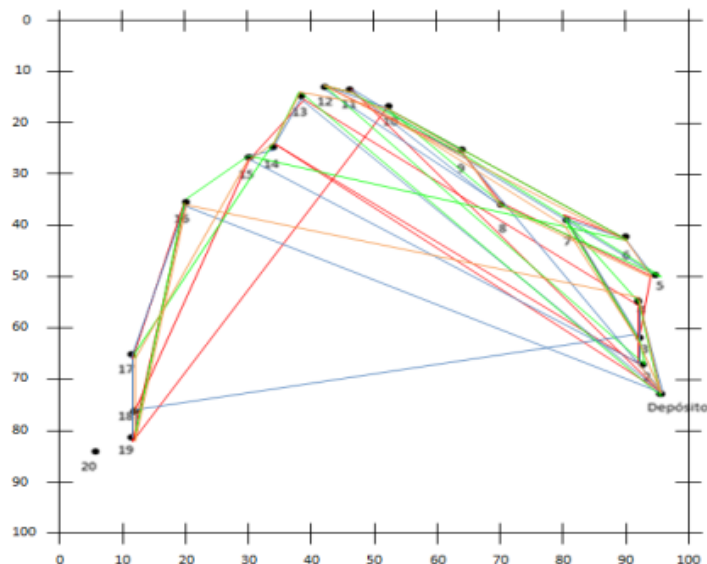
The main method consists of 5 functions in which processing algorithm is performed.

5. Results

The execution of the algorithm shows four routes. Thus it is possible to distribute the units at different times. And gasoline, wages and saves time

5. Conclusions

It is possible to improve the conditions under which public transport units work by applying the ant colony algorithm.

6. Future work

Future work is expected to apply parallelism using CUDA to further improve the algorithm.